9.Phylogenetic Diversity - Communities

Jaeyoung Yoo; Z620: Quantitative Biodiversity, Indiana University

28 February, 2025

OVERVIEW

Complementing taxonomic measures of α- and β-diversity with evolutionary information yields insight into a broad range of biodiversity issues including conservation, biogeography, and community assembly. In this worksheet, you will be introduced to some commonly used methods in phylogenetic community ecology.

After completing this assignment you will know how to:

1. incorporate an evolutionary perspective into your understanding of community ecology
2. quantify and interpret phylogenetic α- and β-diversity
3. evaluate the contribution of phylogeny to spatial patterns of biodiversity

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 today, it is *imperative* that you **push** this file to your GitHub repo, at whatever stage you are. This will enable you to pull your work onto your own computer.
6. When you have completed the worksheet, **Knit** the text and code into a single PDF file by pressing the `Knit` button in the RStudio scripting panel. This will save the PDF output in your '9.PhyloCom' folder.
7. After Knitting, please submit the worksheet by making a **push** to your GitHub repo and then create a **pull request** via GitHub. Your pull request should include this file *9.PhyloCom_Worksheet.Rmd* and the PDF output of `Knitr` (*9.PhyloCom_Worksheet.pdf*).

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

1) 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, provide the code to:
1. clear your R environment,
2. print your current working directory,
3. set your working directory to your `Week7-PhyloCom/` folder,
4. load all of the required R packages (be sure to install if needed), and
5. load the required R source file.

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

```
## [1] "/cloud/project/QB2025_Yoo/Week7-PhyloCom"
```

```
#setwd("~/GitHub/QB-2025/1.HandOuts/11.PhyloCom")

package.list <- c("picante", "ape", "seqinr", "vegan", "fossil",
                  "reshape2", "devtools", "BiocManager", "ineq",
                  "labdsv", "matrixStats", "pROC")
for (package in package.list) {
  if (!require(package, character.only = TRUE, quietly = TRUE)) {
    install.packages(package, repos="http://cran.us.r-project.org")
    library(package, character.only = TRUE)
  }
}
```

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

```
##
## Attaching package: 'seqinr'
```

```
## The following object is masked from 'package:nlme':
##
##     gls
```

```
## The following object is masked from 'package:permute':
##
##     getType
```

```
## The following objects are masked from 'package:ape':
##
##     as.alignment, consensus
```

```
##
## Attaching package: 'shapefiles'
```

```
## The following objects are masked from 'package:foreign':
##
##     read.dbf, write.dbf
```

```
##
## Attaching package: 'devtools'
```

```
## The following object is masked from 'package:permute':
##
##     check
```

```
##
## Attaching package: 'BiocManager'
```

```
## The following object is masked from 'package:devtools':
##
##     install
```

```
## This is mgcv 1.9-1. For overview type 'help("mgcv-package")'.
```

```
## Registered S3 method overwritten by 'labdsv':
##   method       from
##   summary.dist ade4
```

```
## This is labdsv 2.1-0
## convert existing ordinations with as.dsvord()

##
## Attaching package: 'labdsv'

## The following objects are masked from 'package:vegan':
##
##     calibrate, pca, pco, scores

## The following objects are masked from 'package:stats':
##
##     density, loadings

##
## Attaching package: 'matrixStats'

## The following object is masked from 'package:seqinr':
##
##     count

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
source("./bin/MothurTools.R")

## Loading required package: reshape

##
## Attaching package: 'reshape'

## The following objects are masked from 'package:reshape2':
##
##     colsplit, melt, recast
```

2) DESCRIPTION OF DATA

need to discuss data set from spatial ecology!

We sampled >50 forested ponds in Brown County State Park, Yellowood State Park, and Hoosier National Forest in southern Indiana. In addition to measuring a suite of geographic and environmental variables, we characterized the diversity of bacteria in the ponds using molecular-based approaches. Specifically, we amplified the 16S rRNA gene (i.e., the DNA sequence) and 16S rRNA transcripts (i.e., the RNA transcript of the gene) of bacteria. We used a program called `mothur` to quality-trim our data set and assign sequences to operational taxonomic units (OTUs), which resulted in a site-by-OTU matrix.
In this module we will focus on taxa that were present (i.e., DNA), but there will be a few steps where we need to parse out the transcript (i.e., RNA) samples. See the handout for a further description of this week's dataset.

3) LOAD THE DATA

In the R code chunk below, do the following:
1. load the environmental data for the Brown County ponds (*20130801_PondDataMod.csv*),
2. load the site-by-species matrix using the `read.otu()` function,

3. subset the data to include only DNA-based identifications of bacteria,
4. rename the sites by removing extra characters,
5. remove unnecessary OTUs in the site-by-species, and
6. load the taxonomic data using the `read.tax()` function from the source-code file.

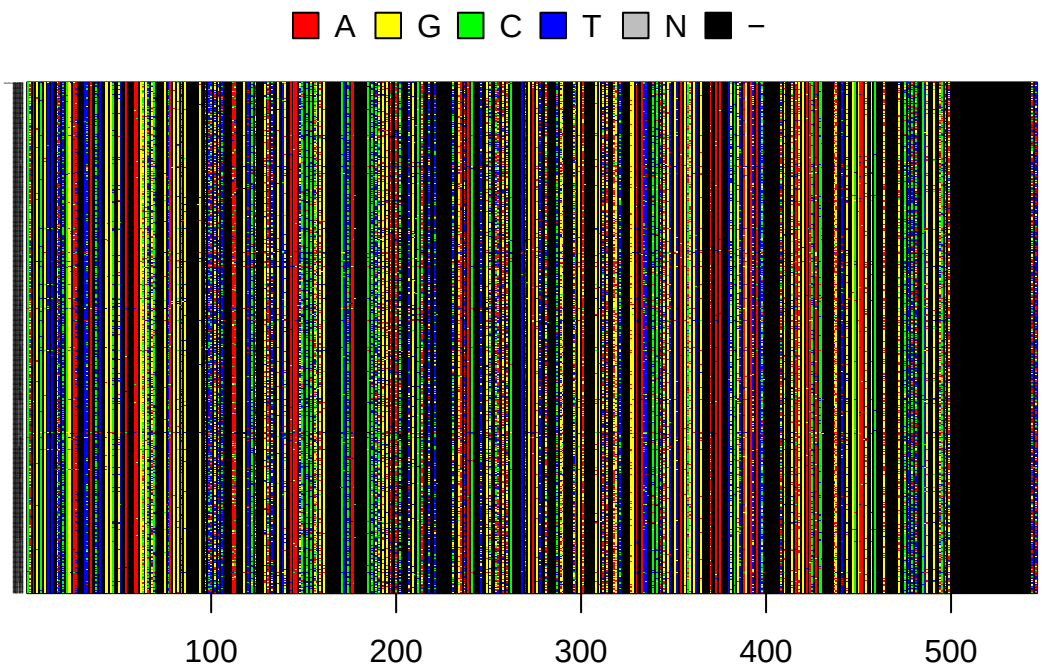
```
env <- read.table("data/20130801_PondDataMod.csv", sep = ",", header = TRUE)
env <- na.omit(env)

# Load Site-by-Species Matrix
comm <- read.otu(shared = "./data/INPonds.final.rdp.shared", cutoff = "1")

# Select DNA data using `grep()`
comm <- comm[grep("*-DNA", rownames(comm)), ]

# Perform replacement of all matches with `gsub()`
rownames(comm) <- gsub("\\-DNA", "", rownames(comm))
rownames(comm) <- gsub("\\_", "", rownames(comm))

# Remove sites not in the environmental data set
comm <- comm[rownames(comm) %in% env$Sample_ID, ]

# Remove zero-abundance OTUs from data set
comm <- comm[, colSums(comm) > 0]

tax <- read.tax(taxonomy = "./data/INPonds.final.rdp.1.cons.taxonomy")
```

```
## Warning in type.convert.default(as.character(x)): 'as.is' should be specified
## by the caller; using TRUE
## Warning in type.convert.default(as.character(x)): 'as.is' should be specified
## by the caller; using TRUE
## Warning in type.convert.default(as.character(x)): 'as.is' should be specified
## by the caller; using TRUE
## Warning in type.convert.default(as.character(x)): 'as.is' should be specified
## by the caller; using TRUE
## Warning in type.convert.default(as.character(x)): 'as.is' should be specified
## by the caller; using TRUE
## Warning in type.convert.default(as.character(x)): 'as.is' should be specified
## by the caller; using TRUE
```

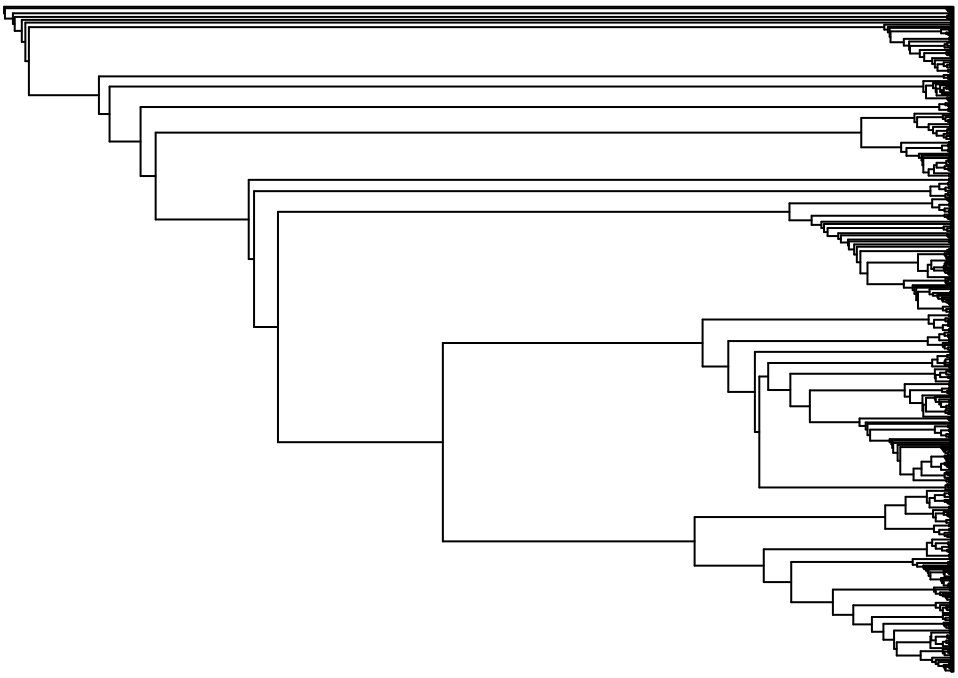
Next, in the R code chunk below, do the following:
1. load the FASTA alignment for the bacterial operational taxonomic units (OTUs),
2. rename the OTUs by removing everything before the tab (\t) and after the bar (|),
3. import the *Methanosarcina* outgroup FASTA file,
4. convert both FASTA files into the DNAbin format and combine using `rbind()`,
5. visualize the sequence alignment,
6. using the alignment (with outgroup), pick a DNA substitution model, and create a phylogenetic distance matrix,
7. using the distance matrix above, make a neighbor joining tree,
8. remove any tips (OTUs) that are not in the community data set,
9. plot the rooted tree.

```
# Import the alignment file (seqinr)
ponds.cons <- read.alignment(file = "./data/INPonds.final.rdp.1.rep.fasta",
                             format = "fasta")

ponds.cons$nam <- gsub(".*\t", "", ponds.cons$nam)
```

```
ponds.cons$nam <- gsub("\\|.*", "", ponds.cons$nam)

# Import outgroup sequence
outgroup <- read.alignment(file = "./data/methanosarcina.fasta", format = "fasta")

# Convert alignment file to DNAbin
DNAbin <- rbind(as.DNAbin(outgroup), as.DNAbin(ponds.cons))

# Visualize alignment
image.DNAbin(DNAbin, show.labels = T, cex.lab = 0.05, las = 1)
```




```
# Make distance matrix (ape)
seq.dist.jc <- dist.dna(DNAbin, model = "JC", pairwise.deletion = FALSE)

# Make a neighbor-joining tree file (ape)
phy.all <- bionj(seq.dist.jc)

# Drop tips of zero-occurrence OTUs (ape)
phy <- drop.tip(phy.all, phy.all$tip.label[!phy.all$tip.label %in%
                c(colnames(comm), "Methanosarcina")])

# Identify outgroup sequence
outgroup <- match("Methanosarcina", phy$tip.label)

# Root the tree (ape)
phy <- root(phy, outgroup, resolve.root = TRUE)

# Plot the rooted tree (ape)
par(mar = c(1, 1, 2, 4) + 0.4)
plot.phylo(phy, main = "Neighbor Joining Tree", "phylogram",
           show.tip.label = FALSE, use.edge.length = FALSE,
           direction = "right", cex = 0.6, label.offset = 1)
```



4) PHYLOGENETIC ALPHA DIVERSITY

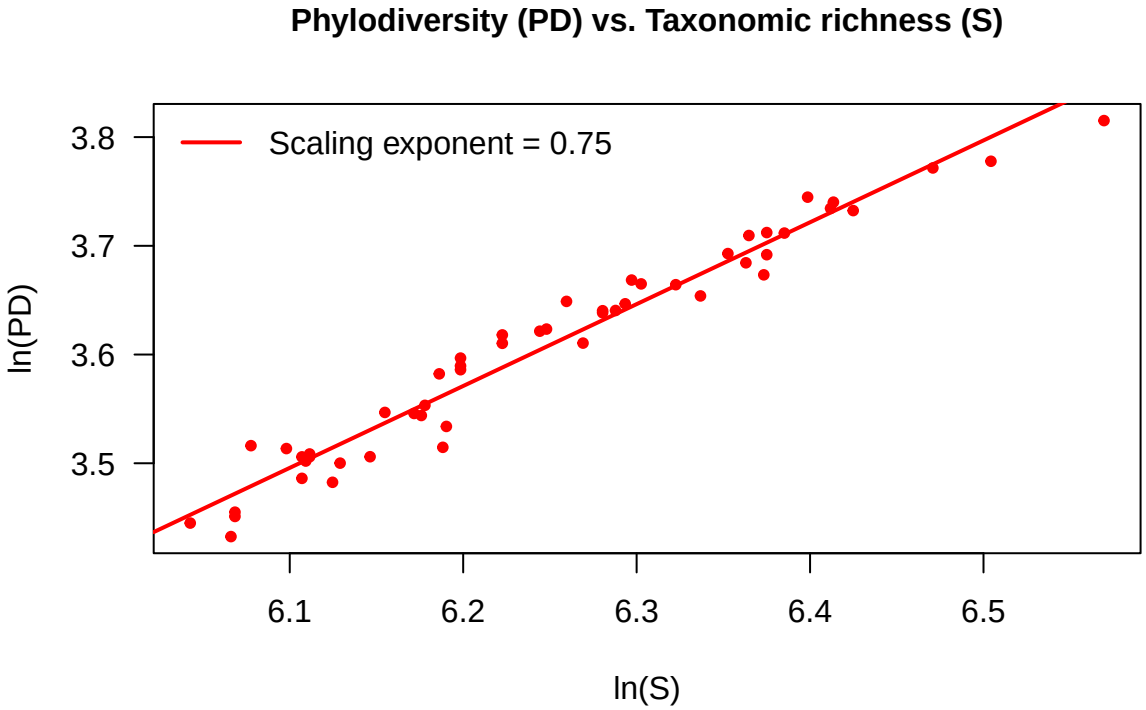
A. Faith's Phylogenetic Diversity (PD)

In the R code chunk below, do the following:
1. calculate Faith's D using the `pd()` function.

```
# Calculate PD and S (picante)
pd <- pd(comm, phy, include.root = FALSE)
```

In the R code chunk below, do the following:
1. plot species richness (S) versus phylogenetic diversity (PD),
2. add the trend line, and
3. calculate the scaling exponent.

```
# Biplot of S and PD
par(mar = c(6, 5, 4, 1) + 0.4)
plot(log(pd$S), log(pd$PD), pch = 20, col = "red", las = 1,
     xlab = "ln(S)", ylab = "ln(PD)", cex.main = 1,
     main = "Phylodiversity (PD) vs. Taxonomic richness (S)")

# Test of power-law relationship
fit <- lm(log(pd$PD) ~ log(pd$S))
abline(fit, col = "red", lw = 2)
exponent <- round(coefficients(fit)[2], 2)
legend("topleft", legend = paste("Scaling exponent = ", exponent, sep = ""),
       bty = "n", lw = 2, col = "red")
```



Question 1: Answer the following questions about the PD-S pattern.
a. Based on how PD is calculated, how and why should this metric be related to taxonmic richness? b. When would you expect these two estimates of diversity to deviate from one another? c. Interpret the significance of the scaling PD-S scaling exponent.

> ***Answer 1a***: The Faith's PD measures branch length of differents species, so when taxonomic richness increases, which means species richness increases, the sum of length will increase as well as Faith's PD. ***Answer 1b***: If there is environmental filtering under harsh environmental conditions, then S can be very high while PD is not that high because of the closeness of those species in the phylogenetic tree. ***Answer 1c***: If the scaling component is 1, then PD and S has a linear relationship, but mostly the scaling component of PD-S is under 1, which is not a linear relationship but showing saturating curve. It means that as richness increases, the phylogenetic diversity increases but the slope of increase decreases.

i. Randomizations and Null Models

In the R code chunk below, do the following:
1. estimate the standardized effect size of PD using the `richness` randomization method.

```
# Estimate standardized effect size of PD via randomization ('picante')
ses.pd(comm[1:2,], phy, null.model = "richness", runs = 25, include.root = FALSE)
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank  pd.obs.z  pd.obs.p
## BC001   668 43.71912     43.56166  0.9890954          14 0.1591993 0.5384615
## BC002   587 40.94334     39.55667  1.0891938          24 1.2731143 0.9230769
##       runs
## BC001   25
## BC002   25
```

```
ses.pd(comm[1:2,], phy, null.model = "frequency", runs = 25, include.root = FALSE)
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank  pd.obs.z   pd.obs.p
## BC001   668 43.71912     42.17015  0.4891871          26  3.166418 1.00000000
## BC002   587 40.94334     42.45806  0.4963430           1 -3.051770 0.03846154
```

```
##       runs
## BC001   25
## BC002   25
```

Question 2: Using `help()` and the table above, run the `ses.pd()` function using two other null models and answer the following questions:

a. What are the null and alternative hypotheses you are testing via randomization when calculating `ses.pd`?
b. How did your choice of null model influence your observed ses.pd values? Explain why this choice affected or did not affect the output.

Answer 2a:
H0: The observed PD is same as the PD of random sample created using a null model. H1: The observed PD is statistically different from the PD of random sample created using a null model. It is not by chance. ***Answer 2b***: The null model influence the observed PD values, because different null models give different assumption and information about expected distribution of PD. The null model based on richness accounts for only the number of species and their presence/absence, while the null model based on frequecy accounts for the probability of species occurrence based on the frequecy of each species.

B. Phylogenetic Dispersion Within a Sample

Another way to assess phylogenetic α-diversity is to look at dispersion within a sample.

i. Phylogenetic Resemblance Matrix

In the R code chunk below, do the following:
1. calculate the phylogenetic resemblance matrix for taxa in the Indiana ponds data set.

```
# Create a Phylogenetic Distance Matrix (picante)
phydist <- cophenetic.phylo(phy)
```

ii. Net Relatedness Index (NRI)

In the R code chunk below, do the following:
1. Calculate the NRI for each site in the Indiana ponds data set.

```
# Estimate standardized effect size of NRI via randomization (picante)
ses.mpd <- ses.mpd(comm, phydist, null.model = "taxa.labels", abundance.weighted = FALSE, runs = 25)

# Calculate NRI
NRI <- as.matrix(-1 * ((ses.mpd[,2] - ses.mpd[,3]) / ses.mpd[,4]))
rownames(NRI) <- row.names(ses.mpd)
colnames(NRI) <- "NRI"
head(NRI)
```

```
##             NRI
## BC001 -2.035378
## BC002 -2.717866
## BC003 -1.411976
## BC004 -2.611653
## BC005 -3.835653
## BC010 -2.286062
```

```
ses.mpd <- ses.mpd(comm, phydist, null.model = "taxa.labels", abundance.weighted = TRUE, runs = 25)
NRI <- as.matrix(-1 * ((ses.mpd[,2] - ses.mpd[,3]) / ses.mpd[,4]))
rownames(NRI) <- row.names(ses.mpd)
```

```
colnames(NRI) <- "NRI"
head(NRI)
```

```
##               NRI
## BC001 -0.01062266
## BC002  0.45303786
## BC003  1.07244755
## BC004 -0.17765516
## BC005  0.71999587
## BC010  0.40726315
```

iii. Nearest Taxon Index (NTI)

In the R code chunk below, do the following: 1. Calculate the NTI for each site in the Indiana ponds data set.

```
# Estimate Standardized Effect Size of NTI via Randomization (picante)
ses.mntd <- ses.mntd(comm, phydist, null.model = "taxa.labels", abundance.weighted = FALSE, runs = 25)

# Calculate NTI
NTI <- as.matrix(-1 * ((ses.mntd[,2] - ses.mntd[,3]) / ses.mntd[,4]))
rownames(NTI) <- row.names(ses.mntd)
colnames(NTI) <- "NTI"
head(NTI)
```

```
##              NTI
## BC001  0.6586114
## BC002 -1.2544767
## BC003 -0.1801059
## BC004 -1.4630012
## BC005 -1.7338526
## BC010 -0.8220208
```

```
# With abundance
ses.mntd <- ses.mntd(comm, phydist, null.model = "taxa.labels", abundance.weighted = TRUE, runs = 25)
NTI <- as.matrix(-1 * ((ses.mntd[,2] - ses.mntd[,3]) / ses.mntd[,4]))
rownames(NTI) <- row.names(ses.mntd)
colnames(NTI) <- "NTI"
head(NTI)
```

```
##             NTI
## BC001 0.8933185
## BC002 1.3251925
## BC003 1.3798647
## BC004 0.9789940
## BC005 1.7237874
## BC010 0.6633280
```

Question 3:

a. In your own words describe what you are doing when you calculate the NRI.
b. In your own words describe what you are doing when you calculate the NTI.
c. Interpret the NRI and NTI values you observed for this dataset.
d. In the NRI and NTI examples above, the arguments "abundance.weighted = FALSE" means that the indices were calculated using presence-absence data. Modify and rerun the code so that NRI and NTI are calculated using abundance data. How does this affect the interpretation of NRI and NTI?

Answer 3a: When calculating NRI, NRI is -(mean phylogenetic distance of an observed sample - mean phylogenetic distance of a random sample)/standard deviation of phylogenetic distance of a

random sample. The mean phylogenetic distance is average of branch length between pairwise taxa. ***Answer 3b***: When calculating NTI, NTI is -(mean nearest phylogenetic neighbor distance of an observed sample - mean nearest phylogenetic neighbor distance of a random sample)/standard deviation of nearest phylogenetic neighbor distance of a random sample. The nearest phylogenetic neighbor distance is an average phylogenetic distance between all taxa and their closest taxa in the tree. ***Answer 3c***: Most NRI values are negative, so the sample is overdispersed than expected phylogenetic distribution based on presence-absence data. Most NTI values are negative, so the sample is overdispersed among nearest taxa than expected phylogenetic distribution. ***Answer 3d***: When using abundance data, most NRI values are positive, so the sample is clustered than expected phylogenetic distribution based on abundance data. Also, when using abundance data, most NTI values are positive, so the sample is clustered than expected phylogenetic distribution based on abundance data.

5) PHYLOGENETIC BETA DIVERSITY

A. Phylogenetically Based Community Resemblance Matrix

In the R code chunk below, do the following:
1. calculate the phylogenetically based community resemblance matrix using Mean Pair Distance, and
2. calculate the phylogenetically based community resemblance matrix using UniFrac distance.

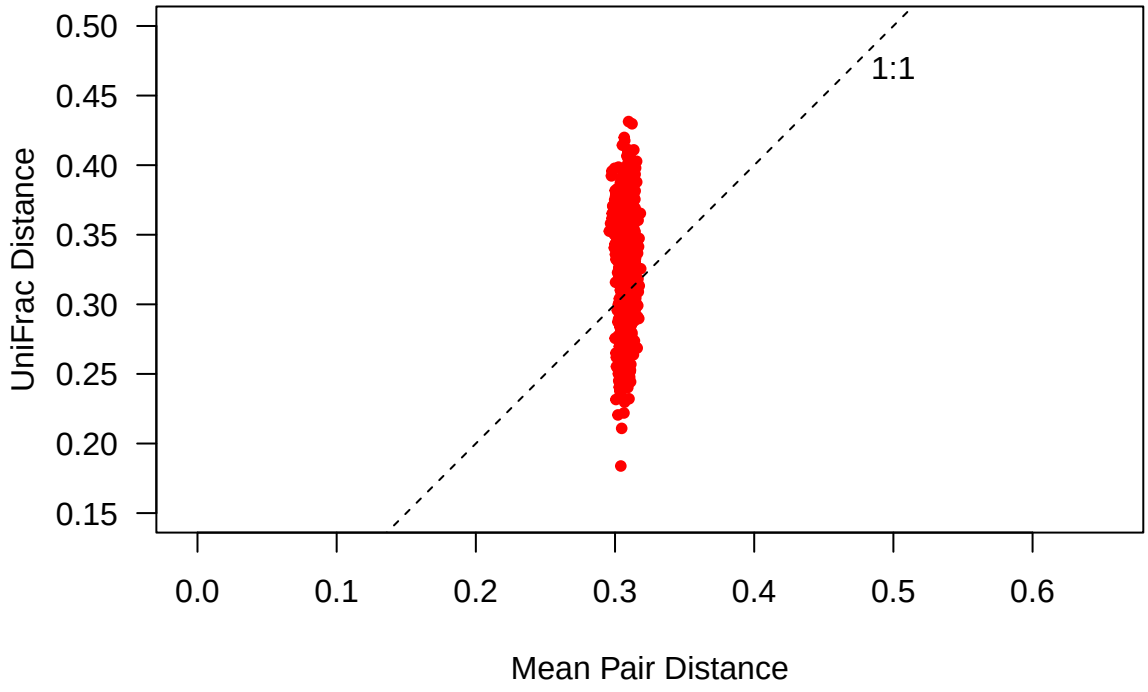
```
# Mean Pairwise Distance
dist.mp <- comdist(comm, phydist)
```

```
## [1] "Dropping taxa from the distance matrix because they are not present in the community data:"
## [1] "Methanosarcina"
```

```
# UniFrac Distance (Note: this takes a few minutes; be patient)
dist.uf <- unifrac(comm, phy)
```

In the R code chunk below, do the following:
1. plot Mean Pair Distance versus UniFrac distance and compare.

```
par(mar = c(6, 5, 2, 1) + 0.4)
plot(dist.mp, dist.uf, pch = 20, col = "red", las = 1, asp = 1,
     xlim = c(0.15, 0.5), ylim = c(0.15, 0.5),
     xlab = "Mean Pair Distance", ylab = "UniFrac Distance")
abline(b = 1, a = 0, lty = 2)
text(0.5, 0.47, "1:1")
```



Question 4:

a. In your own words describe Mean Pair Distance, UniFrac distance, and the difference between them.
b. Using the plot above, describe the relationship between Mean Pair Distance and UniFrac distance. Note: we are calculating unweighted phylogenetic distances (similar to incidence based measures). That means that we are not taking into account the abundance of each taxon in each site.
c. Why might MPD show less variation than UniFrac?

Answer 4a: MPD is just an average phylogenetic distance, and UniFrac accounts for the length of unshared branch, which means it can capture the uniqueness or commonness of the species. ***Answer 4b***: Mean pair distance and UniFrac is not that related. While mean pair distance is clustered around 0.35, UniFrac values are dispersed. ***Answer 4c***: UniFrac is better at capturing and emphasizing the unique evolutionary history of taxa, which share less branch from others, based on phylogenetic tree.

B. Visualizing Phylogenetic Beta-Diversity

Now that we have our phylogenetically based community resemblance matrix, we can visualize phylogenetic diversity among samples using the same techniques that we used in the β-diversity module from earlier in the course.

In the R code chunk below, do the following:
1. perform a PCoA based on the UniFrac distances, and
2. calculate the explained variation for the first three PCoA axes.

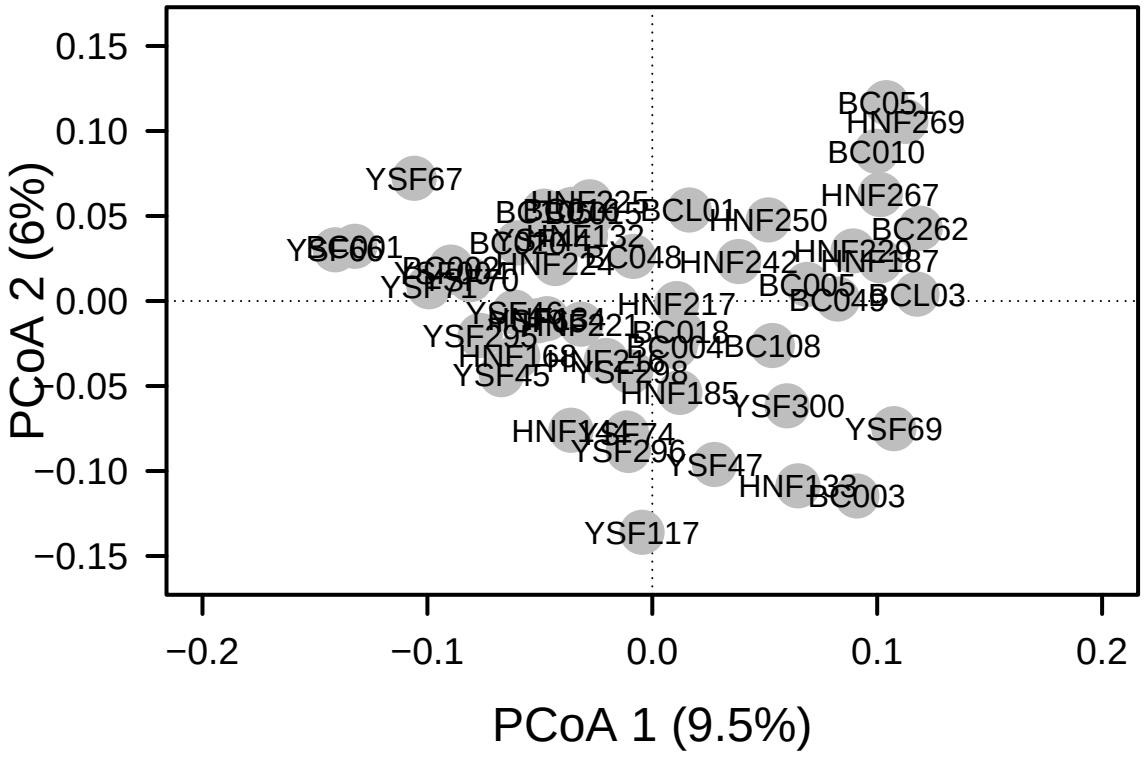
```
pond.pcoa <- cmdscale(dist.uf, eig = T, k = 3)

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

Now that we have calculated our PCoA, we can plot the results.

In the R code chunk below, do the following:

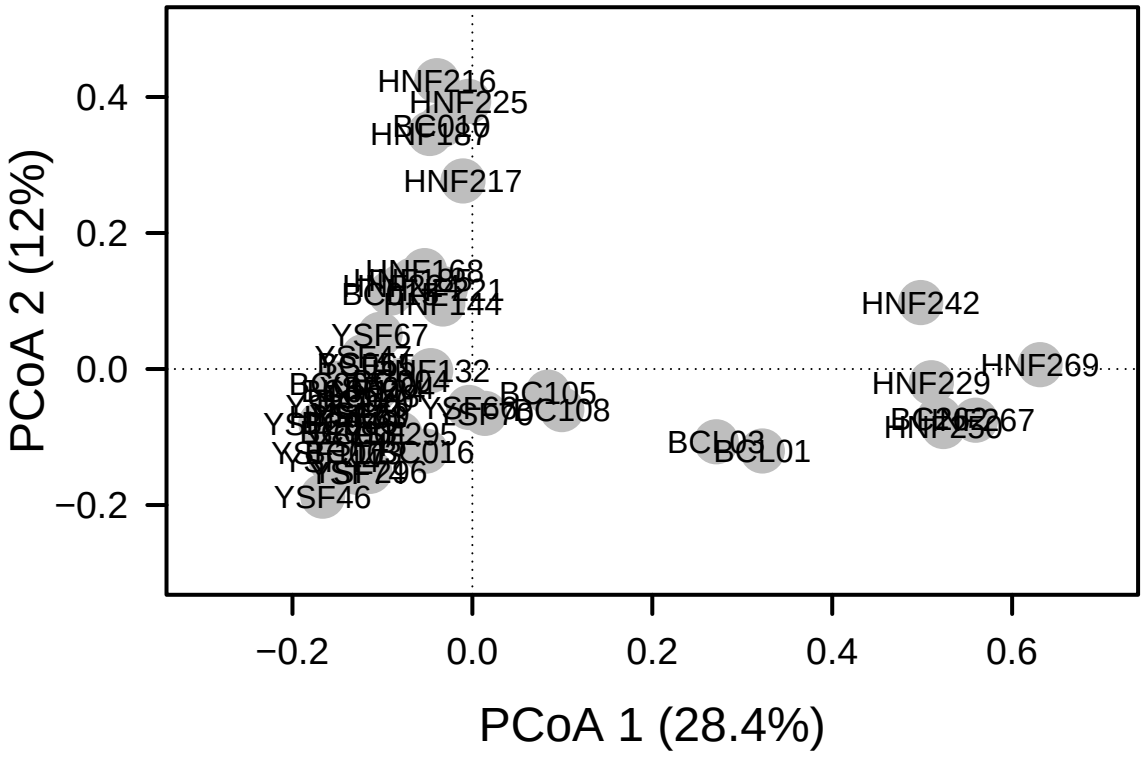
1. plot the PCoA results using either the R base package or the `ggplot` package,
2. include the appropriate axes,
3. add and label the points, and
4. customize the plot.

```
# Define Plot Parameters
par(mar = c(6, 5, 1, 2) + 0.1)

# Initiate Plot
plot(pond.pcoa$points[,1], pond.pcoa$points[,2],
     xlim = c(-0.2, 0.2), ylim = c(-.16, 0.16),
     xlab = paste("PCoA 1 (", explainvar1, "%)", sep = ""),
     ylab = paste("PCoA 2 (", explainvar2, "%)", sep = ""),
     pch = 16, cex = 2.0, type = "n", cex.lab = 1.5, cex.axis = 1.2,
     axes = FALSE)

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

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



In the following R code chunk: 1. perform another PCoA on taxonomic data using an appropriate measure of dissimilarity, and 2. calculate the explained variation on the first three PCoA axes.

```r
# Using Bray-Curtis distance
pond.db <- vegdist(comm, method = "bray", diag = TRUE)

pond.pcoa <- cmdscale(pond.db, eig = T, k = 3)

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

# Define Plot Parameters
par(mar = c(6, 5, 1, 2) + 0.1)

# Initiate Plot
plot(pond.pcoa$points[,1], pond.pcoa$points[,2],
     xlim = c(-0.3, 0.7), ylim = c(-.3, 0.5),
     xlab = paste("PCoA 1 (", explainvar1, "%)", sep = ""),
     ylab = paste("PCoA 2 (", explainvar2, "%)", sep = ""),
     pch = 16, cex = 2.0, type = "n", cex.lab = 1.5, cex.axis = 1.2,
     axes = FALSE)

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

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



Question 5: Using a combination of visualization tools and percent variation explained, how does the phylogenetically based ordination compare or contrast with the taxonomic ordination? What does this tell you about the importance of phylogenetic information in this system?

> ***Answer 5***: Taxonomic ordination has higher PCoA values (percent variation explained) and more dispersed points in the figure. It could mean that the taxonomic distribution of the community has some shared evolutionary history. Also, phylogenetic tree has some ecological insights, and taxonomic data itself is not enough to explain the community structure.

C. Hypothesis Testing

i. Categorical Approach

In the R code chunk below, do the following:
1. test the hypothesis that watershed has an effect on the phylogenetic diversity of bacterial communities.

```
# Define Environmental Category
watershed <- env$Location

# Run PERMANOVA with adonis() Function {vegan}
phylo.adonis <- adonis2(dist.uf ~ watershed, permutations = 999)

# We can compare to PERMANOVA results based on taxonomy
tax.adonis <- adonis2(vegdist(decostand(comm, method = "log"),
                              method = "bray") ~ watershed, permutations = 999)
```

ii. Continuous Approach

In the R code chunk below, do the following: 1. from the environmental data matrix, subset the variables related to physical and chemical properties of the ponds, and
2. calculate environmental distance between ponds based on the Euclidean distance between sites in the environmental data matrix (after transforming and centering using `scale()`).

```r
# Define environmental variables
envs <- env[, 5:19]

# Remove redundant variables
envs <- envs[, -which(names(envs) %in% c("TDS", "Salinity", "Cal_Volume"))]

# Create distance matrix for environmental variables
env.dist <- vegdist(scale(envs), method = "euclid")
```

In the R code chunk below, do the following:
1. conduct a Mantel test to evaluate whether or not UniFrac distance is correlated with environmental variation.

```r
# Conduct Mantel Test {vegan}
mantel(dist.uf, env.dist)
```

```
##
## Mantel statistic based on Pearson's product-moment correlation
##
## Call:
## mantel(xdis = dist.uf, ydis = env.dist)
##
## Mantel statistic r: 0.1604
##       Significance: 0.056
##
## Upper quantiles of permutations (null model):
##   90%   95% 97.5%   99%
## 0.127 0.170 0.197 0.244
## Permutation: free
## Number of permutations: 999
```

Last, conduct a distance-based Redundancy Analysis (dbRDA).

In the R code chunk below, do the following:
1. conduct a dbRDA to test the hypothesis that environmental variation effects the phylogenetic diversity of bacterial communities,
2. use a permutation test to determine significance, and 3. plot the dbRDA results

```r
# Conduct dbRDA {vegan}
ponds.dbrda <- vegan::dbrda(dist.uf ~ ., data = as.data.frame(scale(envs)))

# Permutation tests: axes and environmental variables
anova(ponds.dbrda, by = "axis")
```

```
## Permutation test for dbrda under reduced model
## Forward tests for axes
## Permutation: free
## Number of permutations: 999
##
## Model: vegan::dbrda(formula = dist.uf ~ Elevation + Diameter + Depth + ORP + Temp + SpC + DO + pH + C
##          Df SumOfSqs      F Pr(>F)
## dbRDA1    1  0.10566 2.0152  0.448
## dbRDA2    1  0.09258 1.7658  0.613
## dbRDA3    1  0.07555 1.4409  0.976
## dbRDA4    1  0.06677 1.2735  0.997
## dbRDA5    1  0.05666 1.0807  1.000
```

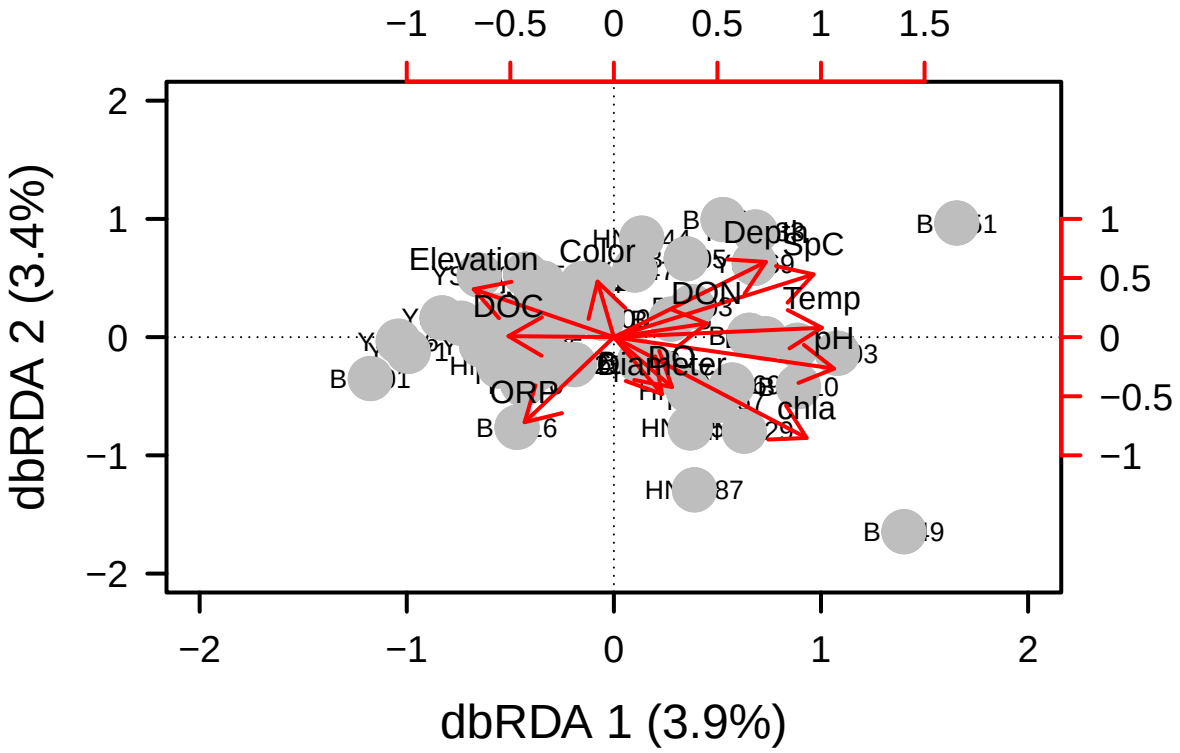
```
## dbRDA6    1  0.05293 1.0095
## dbRDA7    1  0.04750 0.9059
## dbRDA8    1  0.03941 0.7517
## dbRDA9    1  0.03775 0.7201
## dbRDA10   1  0.03280 0.6256
## dbRDA11   1  0.02876 0.5485
## dbRDA12   1  0.02501 0.4770
## Residual 39  2.04482
```

```
ponds.fit <- envfit(ponds.dbrda, envs, perm = 999)
ponds.fit
```

```
##
## ***VECTORS
##
##             dbRDA1   dbRDA2     r2 Pr(>r)
## Elevation -0.77670  0.62986 0.0959  0.076 .
## Diameter   0.27972 -0.96008 0.0541  0.259
## Depth      0.63137  0.77548 0.1756  0.004 **
## ORP       -0.41879 -0.90808 0.1437  0.025 *
## Temp       0.98250  0.18628 0.1523  0.021 *
## SpC        0.77101  0.63682 0.2087  0.009 **
## DO         0.39318 -0.91946 0.0464  0.325
## pH         0.96210 -0.27270 0.1756  0.007 **
## Color     -0.06353  0.99798 0.0464  0.300
## chla       0.60392 -0.79704 0.2626  0.007 **
## DOC       -0.99847 -0.05526 0.0382  0.383
## DON        0.91633  0.40042 0.0339  0.408
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Permutation: free
## Number of permutations: 999
```

```
# Calculate explained variation
dbrda.explainvar1 <- round(ponds.dbrda$CCA$eig[1] / sum(c(ponds.dbrda$CCA$eig, ponds.dbrda$CA$eig)), 3)
dbrda.explainvar2 <- round(ponds.dbrda$CCA$eig[2] / sum(c(ponds.dbrda$CCA$eig, ponds.dbrda$CA$eig)), 3)

# Make dbRDA plot

# Extract scores from the dbRDA object
pond_scores <- vegan::scores(ponds.dbrda, display = "sites")

# Define plot parameters
par(mar = c(5, 5, 4, 4) + 0.1)

# Initiate plot
plot(pond_scores, xlim = c(-2, 2), ylim = c(-2, 2),
     xlab = paste("dbRDA 1 (", dbrda.explainvar1, "%)", sep = ""),
     ylab = paste("dbRDA 2 (", dbrda.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)
```

```r
abline(h = 0, v = 0, lty = 3)
box(lwd = 2)

# Extract site scores
wa_scores <- vegan::scores(ponds.dbrda, display = "sites")

# Add points and labels
points(wa_scores, pch = 19, cex = 3, col = "gray")
text(wa_scores, labels = rownames(wa_scores), cex = 0.8)

# Add points and labels
points(wa_scores, pch = 19, cex = 3, col = "gray")

# Extract environmental vectors (biplot scores)
vectors <- vegan::scores(ponds.dbrda, display = "bp")

# Add environmental vectors to the plot
arrows(0, 0, vectors[,1] * 2, vectors[,2] * 2,
       lwd = 2, lty = 1, length = 0.2, col = "red")

# Add labels for the environmental vectors
text(vectors[,1] * 2, vectors[,2] * 2, pos = 3, labels = rownames(vectors))

# Add axes for the vectors
axis(side = 3, lwd.ticks = 2, cex.axis = 1.2, las = 1, col = "red", lwd = 2.2,
     at = pretty(range(vectors[, 1]) * 2),
     labels = pretty(range(vectors[, 1]) *2))
axis(side = 4, lwd.ticks = 2, cex.axis = 1.2, las = 1, col = "red", lwd = 2.2,
     at = pretty(range(vectors[, 2]) * 2),
     labels = pretty(range(vectors[, 2]) *2))
```



Question 6: Based on the multivariate procedures conducted above, describe the phylogenetic patterns of β-diversity for bacterial communities in the Indiana ponds.

> ***Answer 6***: Temperature, depth and other variables are related to the phylogenetic patterns of beta diversity for bacterial communities in the Indiana ponds. However, environmental factors only can explain 3.9% and 3.4% of variation.

SYNTHESIS

Question 7: Ignoring technical or methodological constraints, discuss how phylogenetic information could be useful in your own research. Specifically, what kinds of phylogenetic data would you need? How could you use it to answer important questions in your field? In your response, feel free to consider not only phylogenetic approaches related to phylogenetic community ecology, but also those we discussed last week in the PhyloTraits module, or any other concepts that we have not covered in this course.

> ***Answer 7***: My research is about how plant and soil fauna interact in terms of resource provision. Plant can support soil food web by providing basal resources (bottom-up control). Root herbivores feed on roots, thereby probably altering the quantity and quality of plant resources by changing root exudate and root turnover rate. Especially, my second chapter is about how different plant resources (aboveground litter, root litter, and living root) alter soil food web which includes from protists, microfauna such as nematodes to macrofauna such as ants and earthworms. I can use phylogenetic information to analyze if different plant resources are more related to certain fauna community which have functional genes more suitable for the resource. For example, in plots only with living root, some nematode species that can get resource directly from living roots will survive (the plant parasitic nematodes have specific structure called stylet that allow them to feed on root tissue or exudate) and thrive, while nematodes without stylet may not thrive.