9.Phylogenetic Diversity - Communities

Student Name; Z620: Quantitative Biodiversity, Indiana University

07 March, 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.

```
source("./bin/MothurTools.R")
```

```
## Loading required package: reshape
```

```
## Warning: package 'reshape' was built under R version 4.4.2
```

```
setwd("C:/Users/ADMIN/OneDrive/Documents/GitHub/QB2025_Nambiar/Week7-PhyloCom/data")

package.list <- c("picante", "ape", "reshape", "devtools",
                  "seqinr", "vegan", "fossil", "labdsv",
                  "matrixStats", "BiocManager", "ineq",
                  "caret", "pROC")

for (package in package.list) {
  if (!require(package, character.only = TRUE)) {
    install.packages(package, repos="http://cran.us.r-project.org",
                     quietly = TRUE)
    library(package, character.only = TRUE)
  }
}
```

```
## Loading required package: picante
```

```
## Warning: package 'picante' was built under R version 4.4.2
```

```
## Loading required package: ape
```

```
## Warning: package 'ape' was built under R version 4.4.2
```

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

```
## Warning: package 'vegan' was built under R version 4.4.2
```

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

```
## Warning: package 'permute' was built under R version 4.4.2
```

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

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

```
## Loading required package: nlme
```

```
## Loading required package: devtools
```

```
## Warning: package 'devtools' was built under R version 4.4.1

## Loading required package: usethis

## Warning: package 'usethis' was built under R version 4.4.1

##
## Attaching package: 'devtools'

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

## Loading required package: seqinr

## Warning: package 'seqinr' was built under R version 4.4.2

##
## 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

## Loading required package: fossil

## Warning: package 'fossil' was built under R version 4.4.2

## Loading required package: sp

## Warning: package 'sp' was built under R version 4.4.1

## Loading required package: maps

## Warning: package 'maps' was built under R version 4.4.1

## Loading required package: shapefiles

## Warning: package 'shapefiles' was built under R version 4.4.2

## Loading required package: foreign
```

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

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

## Loading required package: labdsv

## Warning: package 'labdsv' was built under R version 4.4.2

## Loading required package: mgcv

## 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

## Loading required package: matrixStats

## Warning: package 'matrixStats' was built under R version 4.4.2

##
## Attaching package: 'matrixStats'

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

## Loading required package: BiocManager

## Warning: package 'BiocManager' was built under R version 4.4.2

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

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

## Loading required package: ineq

## Loading required package: caret

## Warning: package 'caret' was built under R version 4.4.2

## Loading required package: ggplot2

##
## Attaching package: 'caret'

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

## The following object is masked from 'package:vegan':
##
##     tolerance

## Loading required package: pROC

## Warning: package 'pROC' was built under R version 4.4.2

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

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

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.

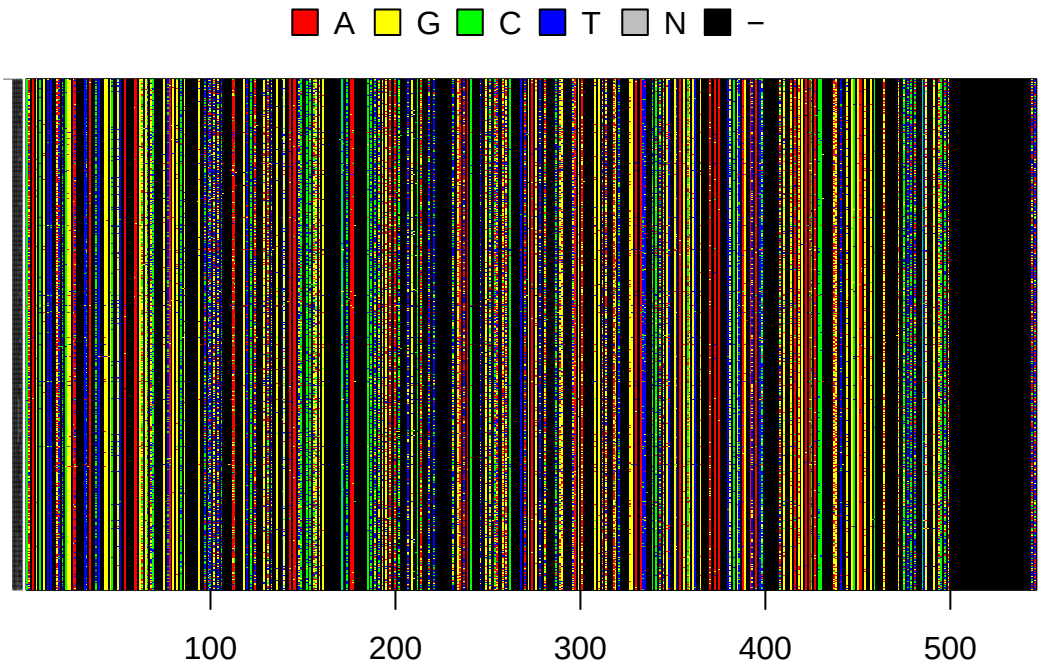
```
env <- read.table("data/20130801_PondDataMod.csv", sep = ",", header = TRUE)

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]

# Import taxonomic information
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

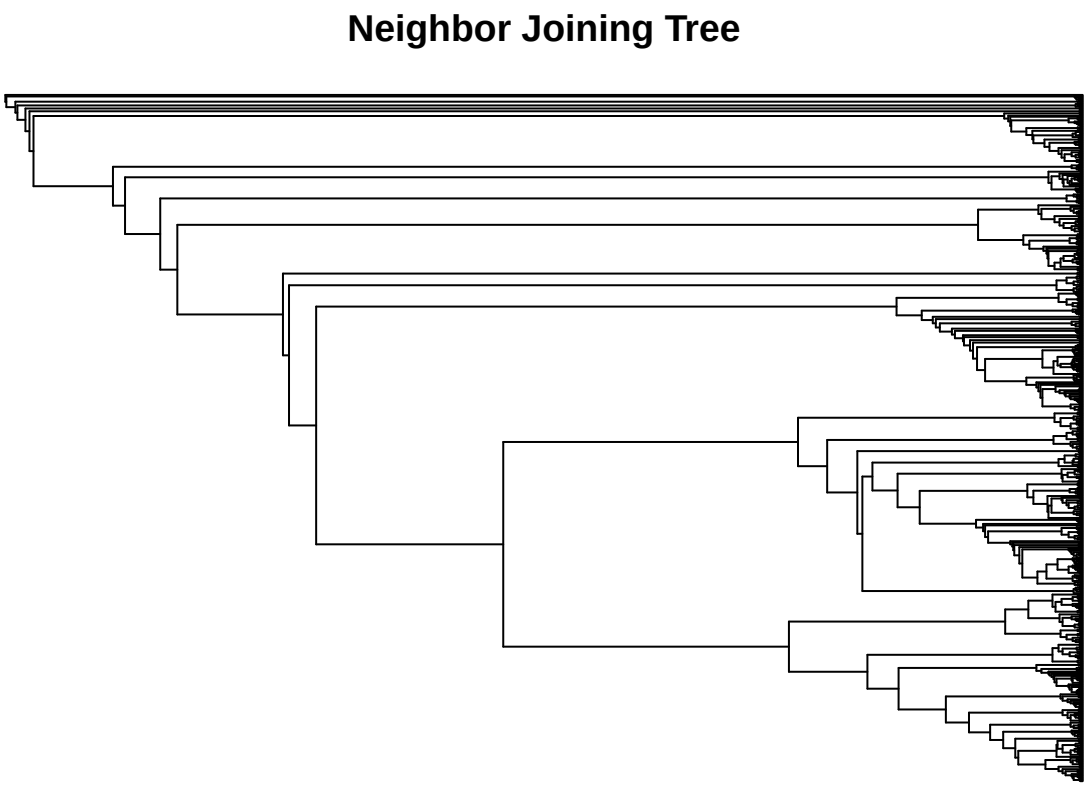
matrix,
7. using the distance matrix above, make a neighbor joining tree,
8. remove any tips (OTUs) that are not in the community data set,
9. plot the rooted tree.

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

# Rename OTUs in the FASTA File
ponds.cons$nam <- gsub("\\|.*$", "", gsub("^.*?\\t", "", 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`)
```

```r
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, 1) + 0.1)
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

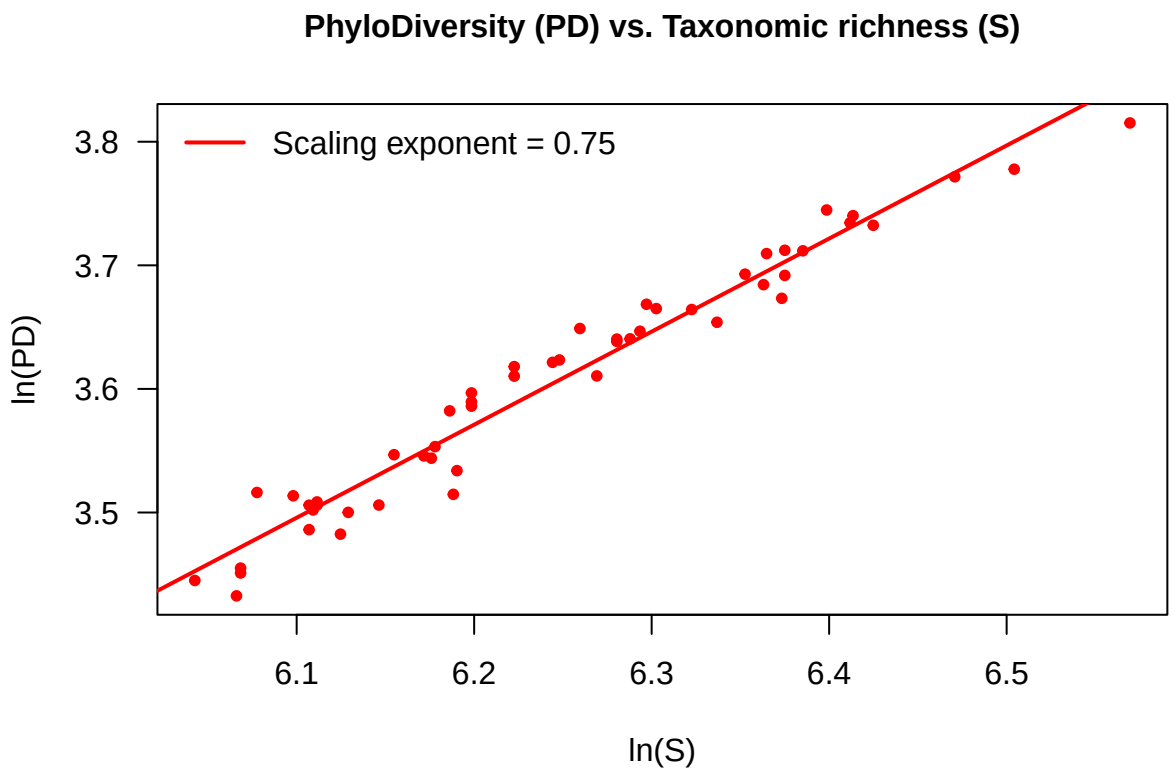
A. Faith's Phylogenetic Diversity (PD)

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

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

# Biplot of S and PD
par(mar = c(5, 5, 4, 1) + 0.1)
```

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.

```
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***: Faith's PD sums the branch lengths spanned by the species in a community. More species often means more total evolutionary historyso higher species richness (S) typically leads to higher PD. ***Answer 1b***: PD and S can diverge if additional species are closely related. In that case, adding more species does not add much extra branch length. Conversely, if new species are distantly related, PD grows faster. ***Answer 1c***: The scaling exponent (~0.75) shows that PD increases with S- at a lower rate than a 1:1 relationship. This suggests reducing returns in total phylogenetic branch length as richness rises.

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

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank   pd.obs.z  pd.obs.p
## BC001   668 43.71912     43.96664  0.8673045          11 -0.2853837 0.4230769
## BC002   587 40.94334     39.85374  0.9372610          24  1.1625375 0.9230769
##       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***: The null hypothesis is that the observed PD could have arisen by randomly assembling species from a regional pool . The alternative hypothesis is that the observed PD is non-random (e.g., significantly higher or lower than the random expectation). ***Answer 2b***: Different null models constrain different aspects of community composition (like total richness vsspecies occurrence frequency. Changing those constraints alters how "expected" values are calculated, so your standardized effect size (ses.pd) may change depending on which null model you use.

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.

```r
# 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.

```r
# 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"
```

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.

```r
# 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"
```

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:NRI (Net Relatedness Index) measures how close or spread out all species in a community are, on average, along the phylogenetic tree. It emphasizes deeper evolutionary relationships.
Answer 3b: NTI (Nearest Taxon Index) focuses on each species' closest relative within the community. It reflects clustering at the tips of the tree i.e., recent common ancestry)

Answer 3c: Interpreting NRI and NTI involves seeing whether communities are more clustered (positive values) or more overdispersed (negative values). A strongly positive NRI or NTI suggests that species in a site are more phylogenetically similar than random.

Answer 3d: When using abundance weighting common species carry more influence (kind of like Simsons Alpha?). That can change NRI or NTI results if frequently occurring species happen to be phylogenetically clustered or overdispersed relative to rare ones.

5) PHYLOGENETIC BETA DIVERSITY

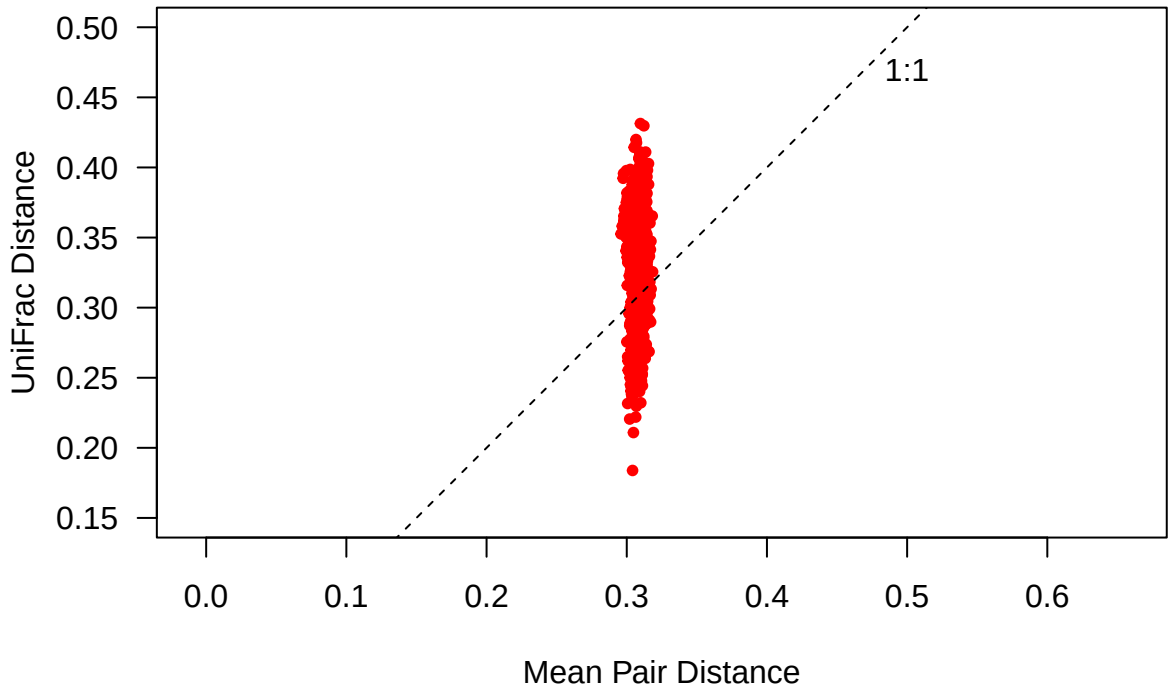
A. Phylogenetically Based Community Resemblance Matrix

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)
```

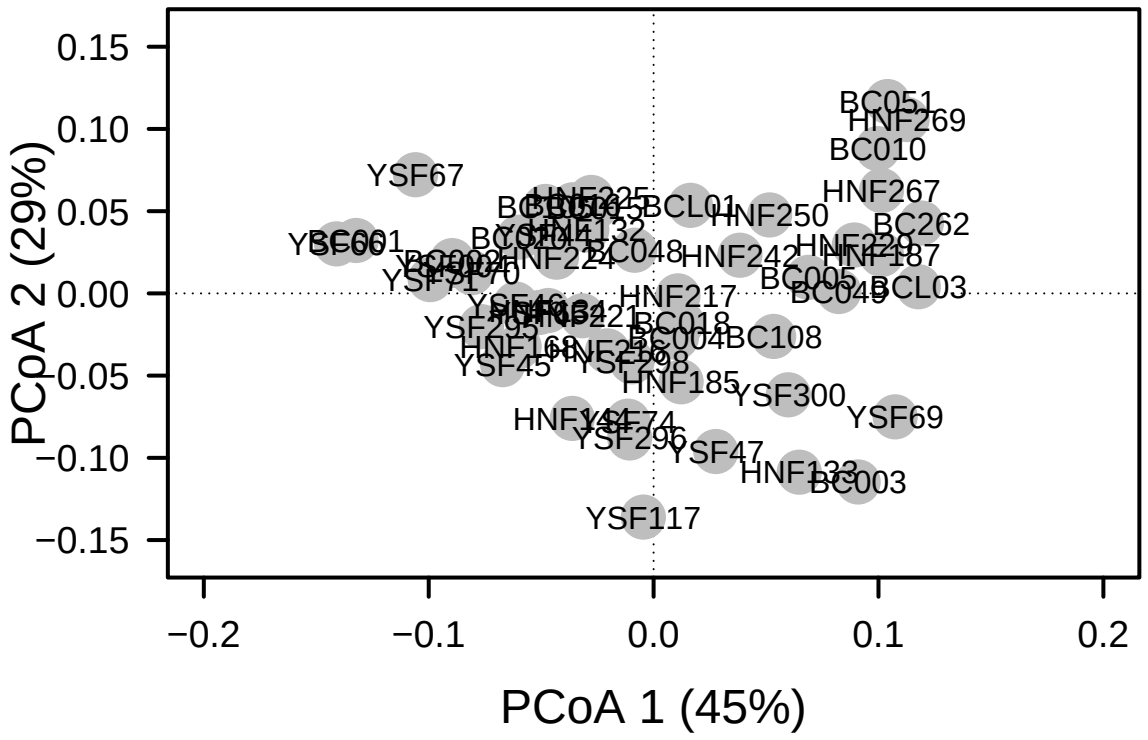
```
## [1] "Dropping taxa from the distance matrix because they are not present in the community data:"
## [1] "Methanosarcina"
```

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

# Compare Mean Pairwise Distance and UniFrac Distance
par(mar = c(6.5, 5, 2, 1) + 0.1)
plot(dist.mp, dist.uf, las = 1, asp = 1,
     pch = 20, col = "red",
     xlab = "Mean Pair Distance", ylab = "UniFrac Distance",
     xlim = c(0.15, 0.5), ylim = c(0.15, 0.5))
abline(b = 1, a = 0, lty = 2)
text(0.5, 0.47, "1:1")
```



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

```
# Perform Principal Coordinates Analysis (PCoA)
pond.pcoa <- cmdscale(dist.uf, eig = T, k = 3)

# Calculate explained variation for each axis
explainvar1 <- round(pond.pcoa$eig[1] / sum(pond.pcoa$eig[1:3]) * 100)
explainvar2 <- round(pond.pcoa$eig[2] / sum(pond.pcoa$eig[1:3]) * 100)
explainvar3 <- round(pond.pcoa$eig[3] / sum(pond.pcoa$eig[1:3]) * 100)

# Define Plot parameters
par(mar = c(6.5, 5, 1, 2) + 0.1)

# Initiate plot
plot(pond.pcoa$points[,1], pond.pcoa$points[,2],
     xlim = c(-0.2, 0.2), ylim = c(-0.16, 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 = TRUE, lwd.ticks = 2, cex.axis = 1.2, las = 1)
axis(side = 2, labels = TRUE, 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 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: Mean Pair Distance (MPD) computes the average phylogenetic distance between species across two communities. UniFrac measures the fraction of branch length unique to one community or shared between communities. UniFrac is more explicit about shared vs. distinct branches on the tree.

Answer 4b: They tend to correlate, but not perfectly. The scatterplot often shows they track similarly (e.g., increasing with differences), but UniFrac can capture differences in branch length distribution that MPD may overlook.

Answer 4c: MPD can compress variation because it uses an average over all pairs. UniFrac can detect subtle differences in which particular branches (lineages) are present or absent, so it may show greater spread in distances.

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.

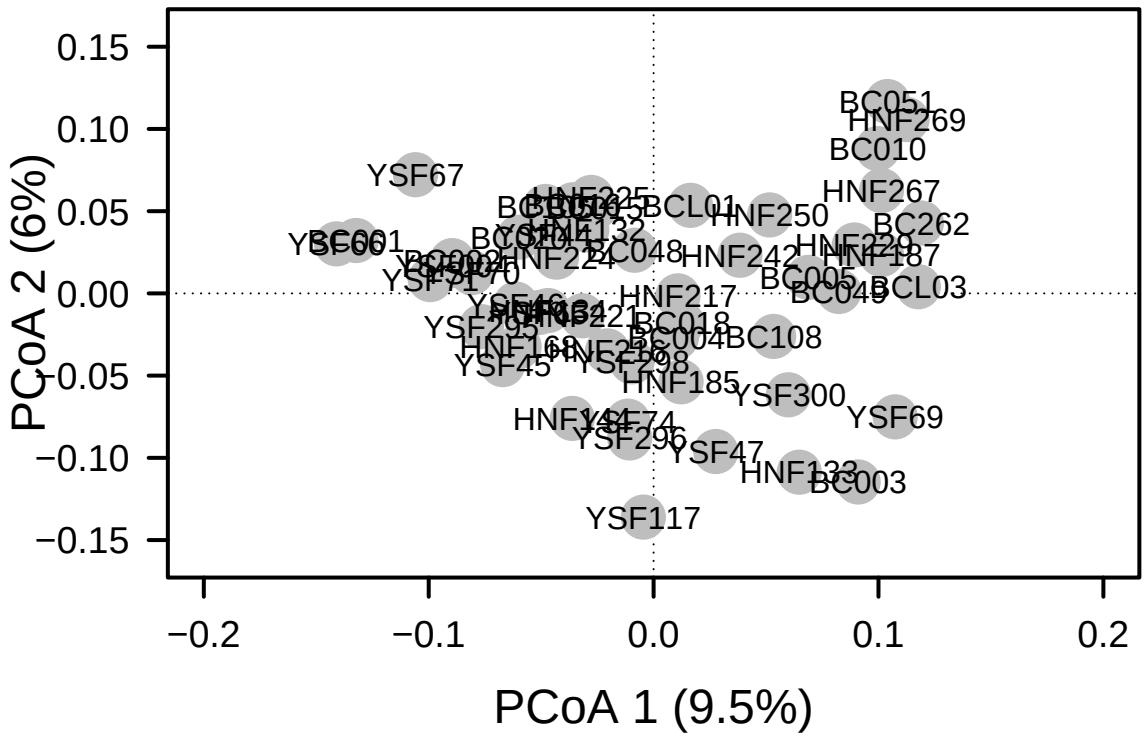
In the R code chunk below, do the following:
1. perform a PCoA based on the UniFrac distances, and
2. calculate the explained variation for the first three PCoA axes.

```
# Perform Principal Coordinates Analysis (PCoA)
pond.pcoa <- cmdscale(dist.uf, eig = TRUE, k = 3)

# Calculate explained variation for each axis
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, 5, 1, 2) + 0.1)

# Initiate Plot
plot(pond.pcoa$points[,1], pond.pcoa$points[,2],
     xlim = c(-0.2, 0.2), ylim = c(-0.16, 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 = TRUE, lwd.ticks = 2, cex.axis = 1.2, las = 1)
axis(side = 2, labels = TRUE, 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.

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***: In many cases, phylogenetically based ordinations group sites differently than taxonomic ordinations. The phylogenetic approach factors in how closely related species are so two sites with different species lists might still cluster together if their species are phylogenetically similar.

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.

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()`).

```
# 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.

```
# 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: 0.062
##
## Upper quantiles of permutations (null model):
##   90%   95% 97.5%   99%
## 0.125 0.176 0.207 0.248
## 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

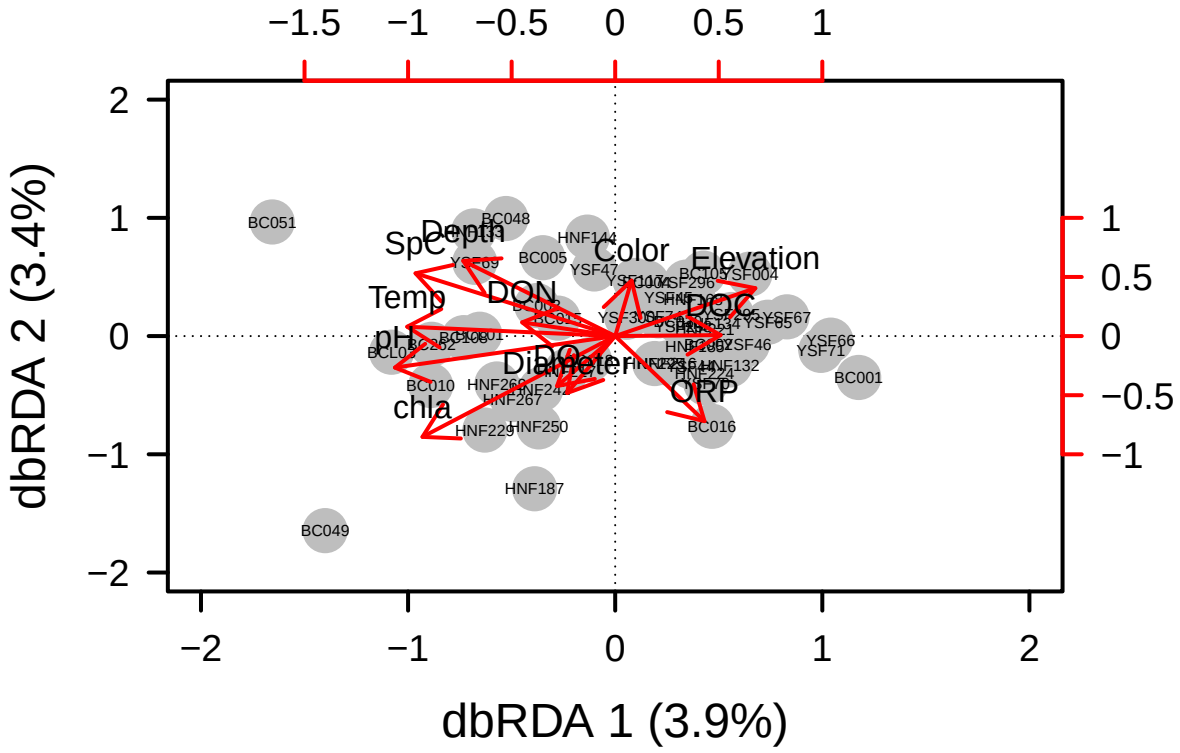
```
# Conduct dbRDA (`vegan`)
ponds.dbrda <- vegan::dbrda(dist.uf ~ ., data = as.data.frame(scale(envs)))

# Permutation test: 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 + (
##          Df SumOfSqs      F Pr(>F)
## dbRDA1    1  0.10566 2.0152  0.423
```

```
## dbRDA2    1  0.09258 1.7658  0.625
## dbRDA3    1  0.07555 1.4409  0.972
## dbRDA4    1  0.06677 1.2735  0.996
## 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
```

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

# Calculate explained variation
dbrda.explainvar1 <- round(ponds.dbrda$CCA$eig[1] /
                           sum(c(ponds.dbrda$CCA$eig, ponds.dbrda$CA$eig)), 3) * 100
dbrda.explainvar2 <- round(ponds.dbrda$CCA$eig[2] /
                           sum(c(ponds.dbrda$CCA$eig, ponds.dbrda$CA$eig)), 3) * 100

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

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

# Initiate plot
plot(ponds_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 = TRUE, lwd.ticks = 2, cex.axis = 1.2, las = 1)
axis(side = 2, labels = TRUE, lwd.ticks = 2, cex.axis = 1.2, las = 1)
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 = 0.5)

# 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 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***: The multivariate analyses (Mantel tests, dbRDA) seem to suggest weaklink between environmental factors and phylogenetic betadiversity. The barely significant p-values and relatively low explained variation imply that while environmental differences are useful for predictions they do not strongly structure the bacterial communities at the phylogenetic level

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***: I think a phylogenetic approach to looking at community structuring especially in the context of conspecific density dependence will be really useful. CNDD or the Janzen

and connell theory posits that distance mortality rate of tree species especailly at the stage of recruitment (seedlings) is a function of distance from other seedlings (or density). Another factor influencing the mrotaility is the phylogenetic relatedness and accounting for density and phylogenetic relatedness would account for biotic affects on the structuring of tree communities. By mapping traits like- , leaf thickness, wood density onto a phylogenetic tree I could test whether certain traits appear together more often than expected by chance, indicating evolutionary or ecological link. This would hel tell us whether current community structure is shaped by ancient lineage traits, recent adaptations, or a mix of both.