9.Phylogenetic Diversity - Communities

Joy O'Brien; Z620: Quantitative Biodiversity, Indiana University

01 March, 2023

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 1st, 2023 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 `/9.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] "/Users/joyobrien/GitHub/QB2023_OBrien/2.Worksheets/9.PhyloCom"
```

```
setwd("~/GitHub/QB2023_OBrien/2.Worksheets/9.PhyloCom")

# Loading packages
package.list <- c('picante', 'ape', 'seqinr', 'fossil', 'reshape', 'devtools', 'BiocManager', 'ineq',
                  'labdsv', 'matrixStats', 'pROC')
for (package in package.list) {
  if (!require(package, character.only=TRUE, quietly=TRUE)) {
    install.packages(package)
    library(package, character.only=TRUE)
  }
}
```

```
## This is vegan 2.6-4

##
## 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 1.8-41. For overview type 'help("mgcv-package")'.

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

## This is labdsv 2.0-1
## convert existing ordinations with as.dsvord()

##
## Attaching package: 'labdsv'

## The following object is masked from 'package:stats':
##
##     density

##
## 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
```

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

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.

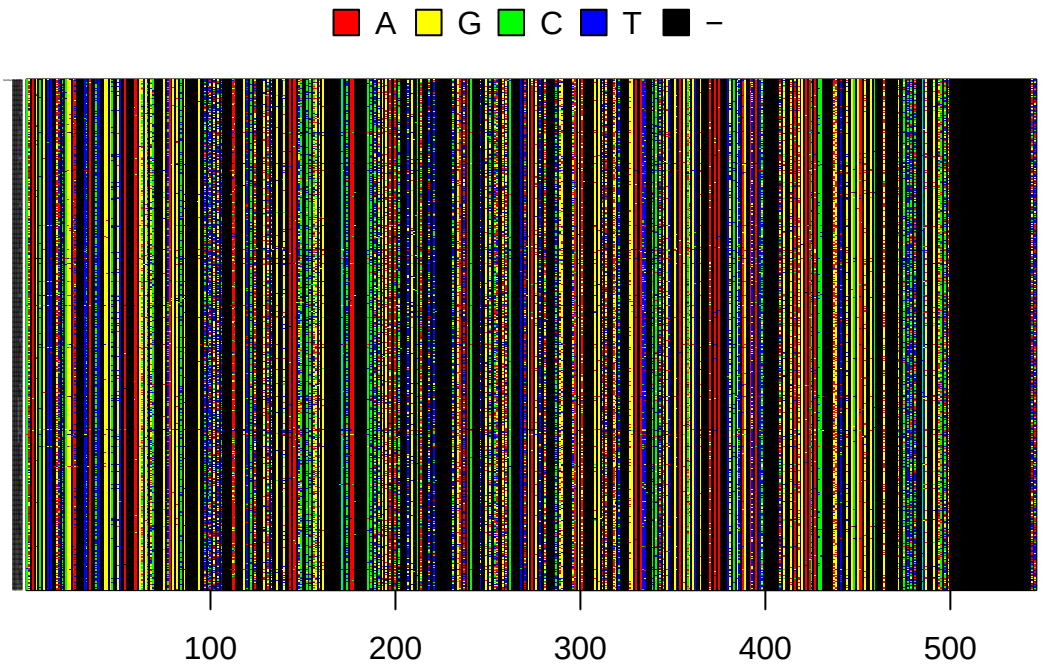
```
env <- read.table("data/20130801_PondDataMod.csv", sep = ",", header = TRUE)
env <- na.omit(env)

# Load site by species matrix
comm <- read.otu(shared = "./data/INPonds.final.rdp.shared", cutoff = "1")

# Select DNA 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
```

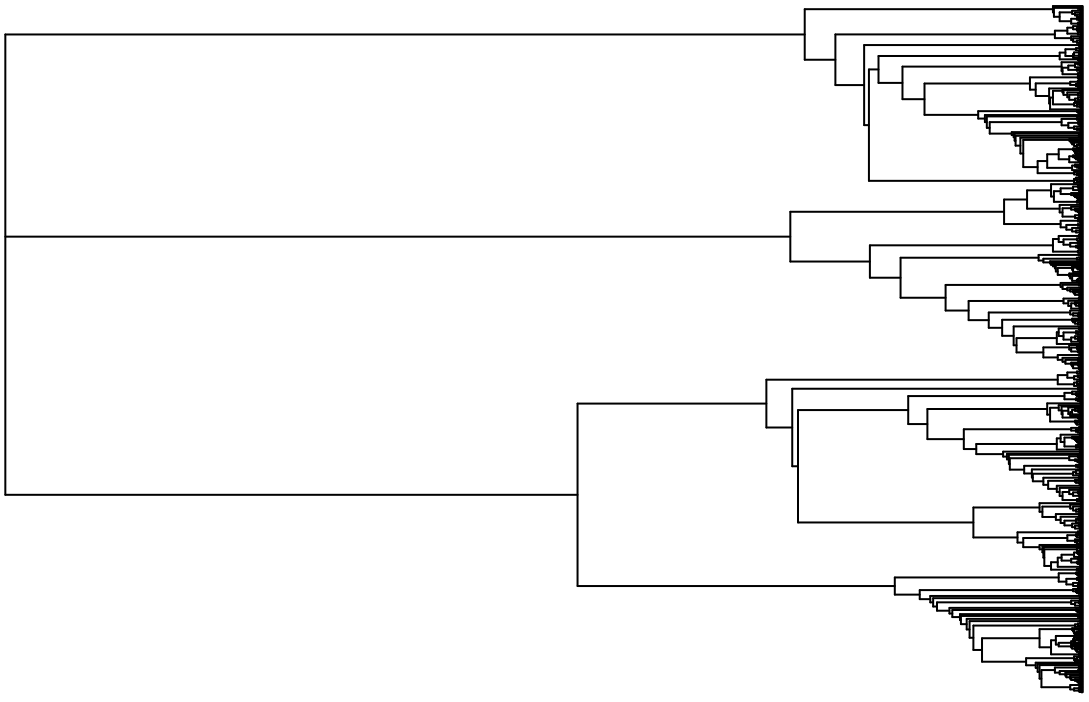
Next, in the R code chunk below, do the following:
1. load the FASTA alignment for the bacterial operational taxonomic units (OTUs),
2. rename the OTUs by removing everything before the tab (\t) and after the bar (|),
3. import the *Methanosarcina* outgroup FASTA file,
4. convert both FASTA files into 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
ponds.cons <- read.alignment(file = "./data/INPonds.final.rdp.1.rep.fasta",
                             format = "fasta")
# Rename OTUs in fasta file
# ponds.cons$nam <- gsub("\\|.*$", "", gsub("^.*?\t", "", ponds.cons$nam)) # this was giving the error
ponds.cons$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)
```



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

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

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

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

# Plot the rooted tree
par(mar = c(1, 1, 2, 1) + 0.1)
plot.phylo(phy.all, 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.

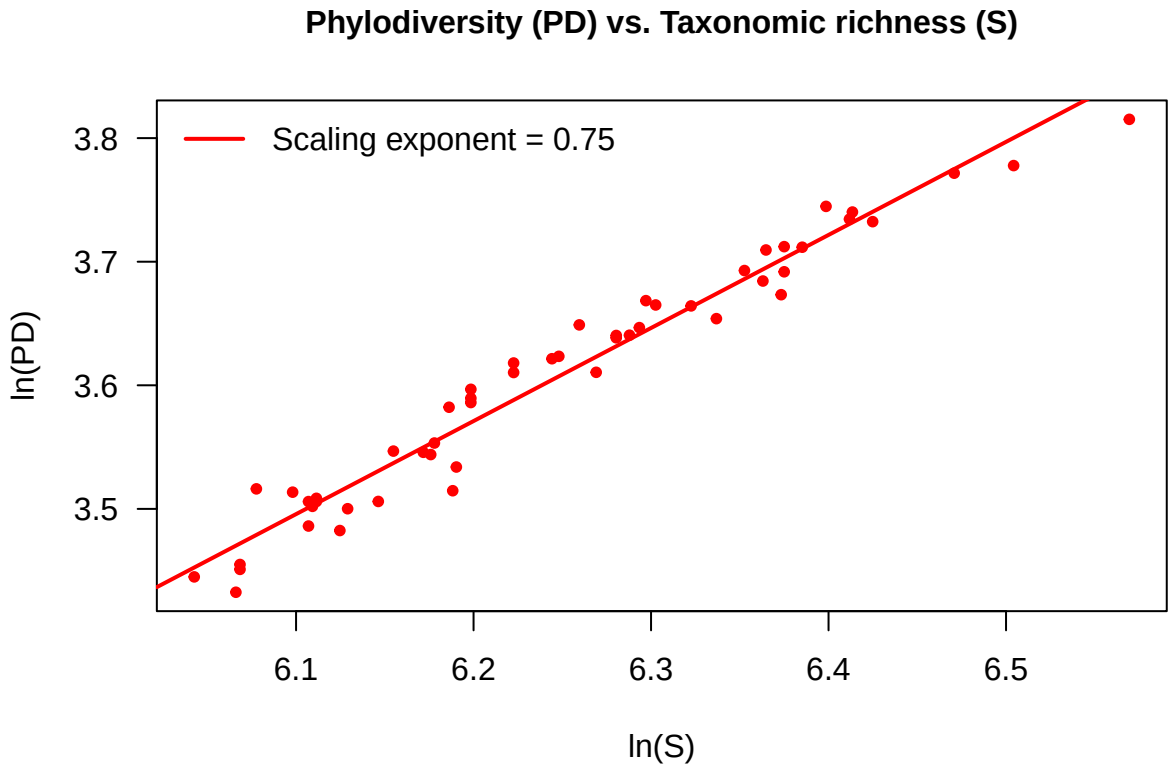
```
# Calculate PD and S
pd <- pd(comm, phy.all, include.root = FALSE)
pd
```

```
##              PD  SR
## BC001  43.71912 668
## BC002  40.94334 587
## BC003  31.53402 432
## BC004  35.95465 486
## BC005  33.65632 436
## BC010  31.34254 421
## BC015  40.15954 574
## BC016  38.62593 565
## BC018  36.98545 528
## BC020  40.92505 593
## BC048  37.39332 515
## BC049  32.65870 449
## BC051  33.56599 445
## BC105  41.86524 609
## BC108  37.46606 517
## BC262  33.31506 467
## BCL01  38.43171 523
## BCL03  33.11983 459
## HNF132 38.03250 534
## HNF133 33.31136 449
## HNF134 38.34699 541
## HNF144 37.26483 504
## HNF168 39.19321 543
## HNF185 33.39952 451
## HNF187 30.95549 431
## HNF216 36.47943 492
## HNF217 36.97870 504
## HNF221 38.11234 538
## HNF224 39.05581 546
## HNF225 38.10391 534
## HNF229 34.25624 488
## HNF242 34.92780 482
## HNF250 34.66701 479
## HNF267 33.60526 487
## HNF269 32.54033 457
## YSF004 42.29978 601
## YSF117 34.70052 471
## YSF295 42.10615 610
## YSF296 34.60291 481
## YSF298 36.21693 492
## YSF300 33.31466 451
## YSF44  40.11835 587
```

```
## YSF45  40.83293 581
## YSF46  39.02660 557
## YSF47  33.18340 450
## YSF65  39.38228 586
## YSF66  45.38554 713
## YSF67  43.45101 646
## YSF69  31.65875 432
## YSF70  39.82013 580
## YSF71  41.77992 617
## YSF74  36.09142 492
```

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.

```
# Create biplot of S and PD
par(mar = c(5, 5, 4, 1) + 0.1)
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, why should this metric be related to taxonmic richness? b. Describe the relationship between taxonomic richness and phylodiversity. c. When would you expect these two estimates of diversity to deviate from one another? d. Interpret the significance of the scaling PD-S scaling exponent.

> ***Answer 1a***: This method is related to taxonomic richness because it sums the branch lengths for each species found in a sample from the root to the tip of the phylogenetic tree. This PD function returns observed richness. ***Answer 1b***: Based on the figure above, as taxonomic richness increases, phylodiversity also increases. ***Answer 1c***: We would expect these esimates to deviate from one another when compared to a null model. ***Answer 1d***: The scaling exponent that we see above is 0.75 which is representitive of how PD scales with S (slope).

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

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank   pd.obs.z  pd.obs.p
## BC001   668 43.71912     44.02562  0.8542398           8 -0.3587939 0.3076923
## BC002   587 40.94334     39.86733  0.7568846          24  1.4216250 0.9230769
##       runs
```

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

```
# Using other null models
ses.pd.freq <- ses.pd(comm[1:2,], phy, null.model = "frequency", runs = 25,
                include.root = FALSE)
ses.pd.freq
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank  pd.obs.z   pd.obs.p
## BC001   668 43.71912     42.35284  0.7011607          25  1.948607 0.96153846
## BC002   587 40.94334     42.24827  0.7094409           2 -1.839380 0.07692308
##       runs
## BC001   25
## BC002   25
```

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

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank   pd.obs.z  pd.obs.p
## BC001   668 43.71912     43.89787  0.4404995           8 -0.4057884 0.3076923
## BC002   587 40.94334     40.66624  0.4161234          21  0.6659144 0.8076923
##       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: When using the 'ses.pd' function, we are testing whether or not our sample is more or less phylogenetically diverse than what is expected when compared to the null distribution.
Answer 2b: I chose frequency and trialswap as my null models. The observed pd.obs value was 43.7 (BC001) and 40.9 (BC002) for both null model types. These values are the same as the richness null model, indicating that the choice of null model did not affect the output. This is because both of our samples are more phylogenetically diverse than expected which is supported by randomization/resampling of the data.

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 phylogenetic distance matrix
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
ses.mpd <- ses.mpd(comm, phydist, null.model = "taxa.labels",
                   abundance.weighted = TRUE, 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"

NRI
```

```
##                 NRI
## BC001   0.156387772
## BC002   0.556703187
## BC003   0.712806593
## BC004  -0.046161537
## BC005   0.585459337
## BC010   0.487252207
## BC015   0.121279153
## BC016   0.363832417
## BC018   0.299690599
## BC020   0.351688692
## BC048  -0.007633230
## BC049   0.415748457
## BC051  -0.711276601
## BC105  -0.454547810
## BC108  -0.095495374
## BC262  -0.103767824
## BCL01  -0.195143596
## BCL03  -0.367297431
## HNF132 -0.117639287
## HNF133  0.229144157
## HNF134  0.368366547
## HNF144 -0.157753380
## HNF168 -0.136698662
## HNF185  0.539973729
## HNF187  0.799215357
## HNF216  0.761112056
## HNF217 -0.065051734
## HNF221 -0.358285371
## HNF224  0.206673559
## HNF225  0.793338079
## HNF229 -0.137576482
## HNF242  0.144919101
## HNF250 -0.117964155
## HNF267 -0.280715743
```

```
## HNF269 -0.129535159
## YSF004 -0.249445886
## YSF117  0.637297736
## YSF295 -1.073054959
## YSF296  0.793154556
## YSF298  1.002278304
## YSF300  0.422884286
## YSF44   0.538359275
## YSF45   0.705272120
## YSF46   1.271186993
## YSF47   0.397749810
## YSF65   0.584154171
## YSF66  -0.230520593
## YSF67  -0.002987735
## YSF69   0.017066567
## YSF70  -0.433932397
## YSF71   0.672141988
## YSF74   0.827324990
```

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 NRI via randomization
ses.mntd <- ses.mntd(comm, phydist, null.model = "taxa.labels",
                     abundance.weighted = TRUE, 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"

NTI
```

```
##               NTI
## BC001   1.7177844
## BC002   2.0264035
## BC003   1.2966823
## BC004   1.3053059
## BC005   2.5919722
## BC010   0.5781323
## BC015   1.4701073
## BC016   1.9880975
## BC018   1.5467693
## BC020   1.5566778
## BC048   1.4634315
## BC049   2.2267063
## BC051   2.6874396
## BC105   1.2673963
## BC108   1.4266580
## BC262   1.1591228
## BCL01   1.5045757
## BCL03   1.3228317
## HNF132  1.4681186
```

```
## HNF133  1.3918113
## HNF134  1.9643587
## HNF144  0.9551916
## HNF168  0.6261667
## HNF185  1.5910330
## HNF187  0.6678411
## HNF216  0.4635686
## HNF217  0.5007698
## HNF221  0.4704541
## HNF224  1.4744642
## HNF225  0.4581301
## HNF229  1.5742386
## HNF242  1.5333254
## HNF250  1.2646376
## HNF267  0.8401898
## HNF269  1.1042402
## YSF004  0.3101223
## YSF117  2.4325820
## YSF295 -0.7727756
## YSF296  2.0512632
## YSF298  1.9893187
## YSF300  1.5717718
## YSF44   1.1255413
## YSF45   1.3694824
## YSF46   1.5010452
## YSF47   1.4837356
## YSF65   1.8583742
## YSF66   1.6384040
## YSF67   1.1875240
## YSF69   1.1676561
## YSF70   1.1264651
## YSF71   1.2843736
## YSF74   1.3895525
```

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 we calculate NRI, we are quantifying alpha phylodiversity with respect to dispersion of species within a sample. NRI is based on mean phylogenetic distance, which is the average branch length between taxa in a pairwise sample (MPD). Once MPD is calculated for the phylogenetic resemblance matrix, we can find NRI by assigning a negative sign to the MPD observation value and subtracting the mean of the MPD values (produced via randomization from a null model) from the MPD observation. We then take this value and divide it by standard deviation of the MPD values generated from the null model which yields the value for NRI.
Answer 3b: When we calculate NTI, we are using the mean nearest phylogenetic neighbor distance. To do this, we take the sum of the minimum values for each taxon of the phylogenetic resemblance matrix followed by randomizations and then the NTI calculation which is the differnce between the negative mean nearest taxon distance observed and the average of the mean

nearest taxon distance all divided by the standard deviation of mean nearest taxon distance. ***Answer 3c***: The NRI values for this dataset range from -0.38 to -2.8 indicating that the samples within these sites are phylogenetically overdispersed meaning that taxa are less related to one another than would be expected from the null model. The NTI values for this dataset range from -2 to 1.7 indicating that some samples are overdispersed (negative values) and some are clustered (positive) meaning that nearest taxa are more closely related than would be expected under the null model. ***Answer 3d***: The NRI values when "abundance.weighted = FALSE" range from -1.06 to 1.7. There are definitely more positive NRI values which indicates that the samples are phylogenetically underdispersed. The NTI values when "abundance.weighted = FALSE" range from -0.9 to 2.1 indicating that taxa are more closely related than expected and therefore are phylogenetically clustered.

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.

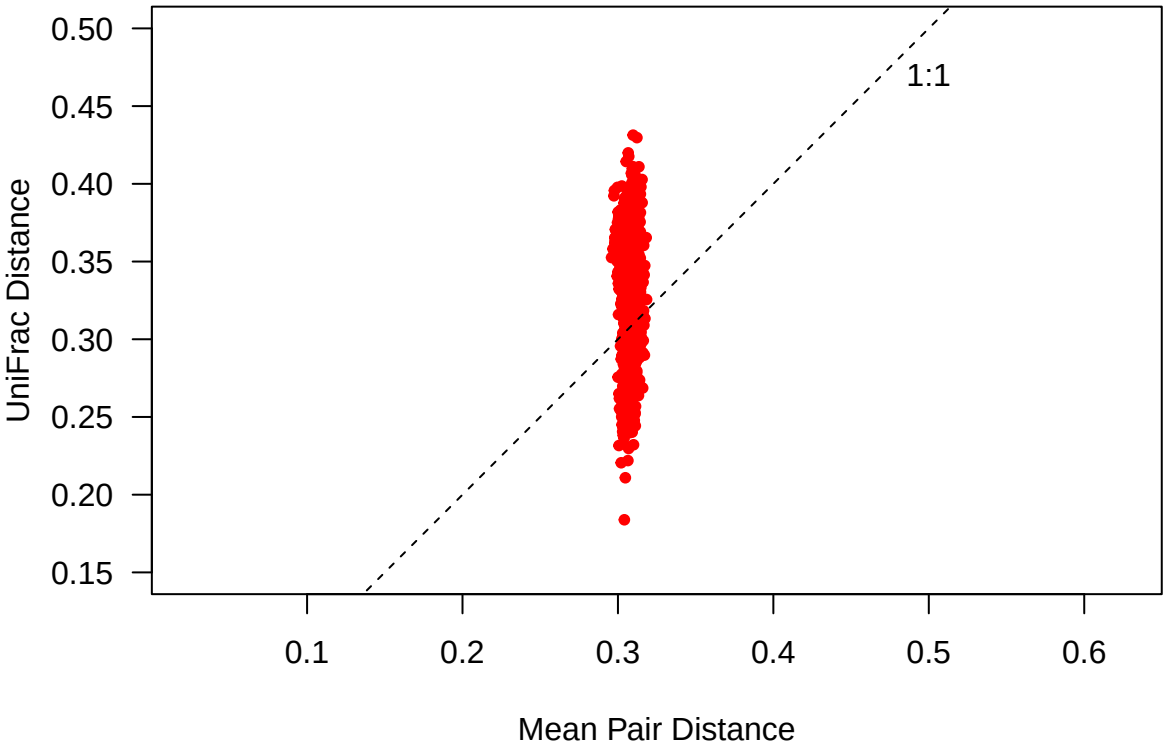
```
# Mean Pairwise Distance
dist.mp <- comdist(comm, phydist)
```

```
## [1] "Dropping taxa from the distance matrix because they are not present in the community data:"
##  [1] "Methanosarcina" "Otu0881"        "Otu0963"        "Otu0969"
##  [5] "Otu0984"        "Otu0990"        "Otu0991"        "Otu0997"
##  [9] "Otu0998"        "Otu1002"        "Otu1004"        "Otu1007"
## [13] "Otu1011"        "Otu1013"        "Otu1019"        "Otu1022"
## [17] "Otu1023"        "Otu1025"        "Otu1029"        "Otu1030"
## [21] "Otu1034"        "Otu1039"        "Otu1049"        "Otu1050"
## [25] "Otu1052"        "Otu1057"        "Otu1058"        "Otu1059"
## [29] "Otu1060"        "Otu1061"        "Otu1062"        "Otu1069"
## [33] "Otu1072"        "Otu1073"        "Otu1074"        "Otu1079"
## [37] "Otu1083"        "Otu1084"        "Otu1085"        "Otu1089"
## [41] "Otu1090"        "Otu1091"        "Otu1093"        "Otu1094"
## [45] "Otu1096"        "Otu1097"        "Otu1098"        "Otu1112"
## [49] "Otu1113"        "Otu1115"        "Otu1116"        "Otu1119"
## [53] "Otu1120"        "Otu1123"
```

```
# UniFrac Distance
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(5, 5, 2, 1) + 0.1)
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: Mean pairwise distance allows us to calculate phylogenetic distances in a community resemblance matrix by taking the average phylogenetic distance between pairs of taxa. UniFrac distance is calculated by dividing the sum of unshared branch lengths between samples by the total branch lengths in the rooted tree (total refers to shared and unshared). The main difference between these two phylogenetic distances is that UniFrac takes into account unshared branch lengths while mean pairwise distance does not. ***Answer 4b***: Mean Pairwise distance is approximately 0.3 while UniFrac distance varies from approx. 0.2 to 0.45. This variation seen within the UniFrac distance is not dependent on large changes in mean pairwise distance. ***Answer 4c***: MPD may show less variation than UniFrac because MPD is working with average phylogenetic distance values and UniFrac is not.

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.

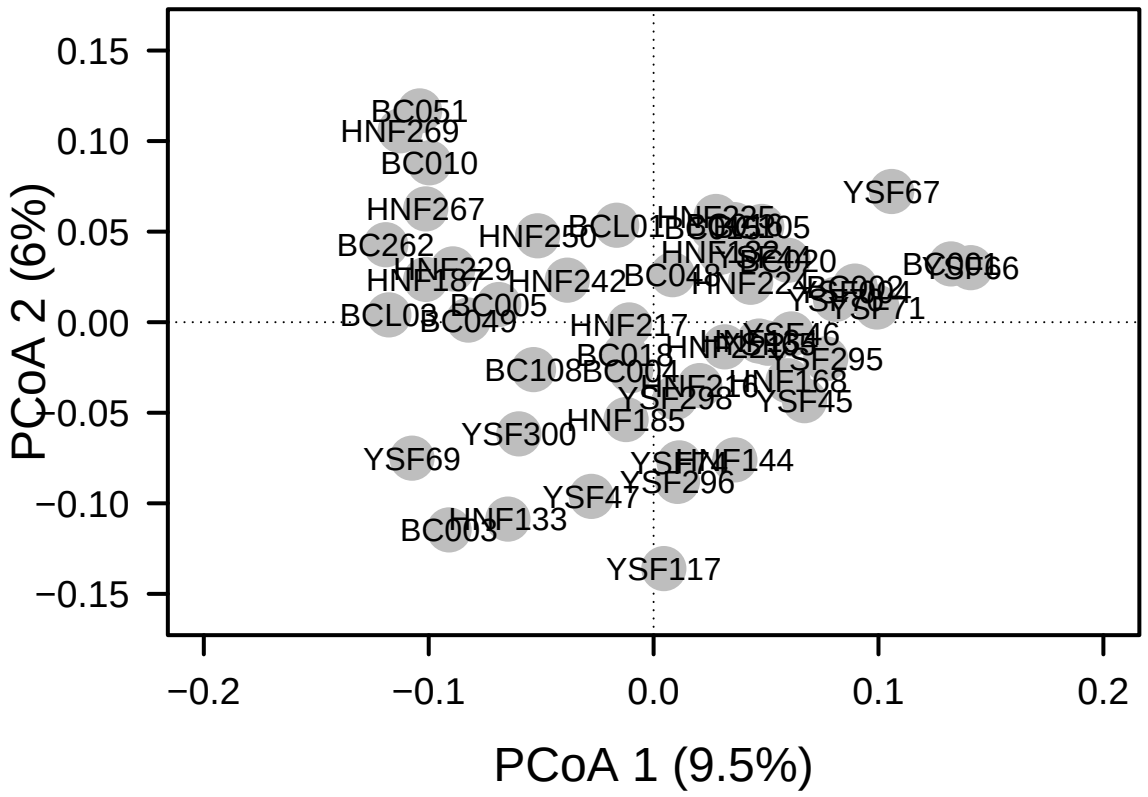
In the R code chunk below, do the following:
1. perform a PCoA based on the UniFrac distances, and
2. calculate the explained variation for the first three PCoA axes.

```
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.

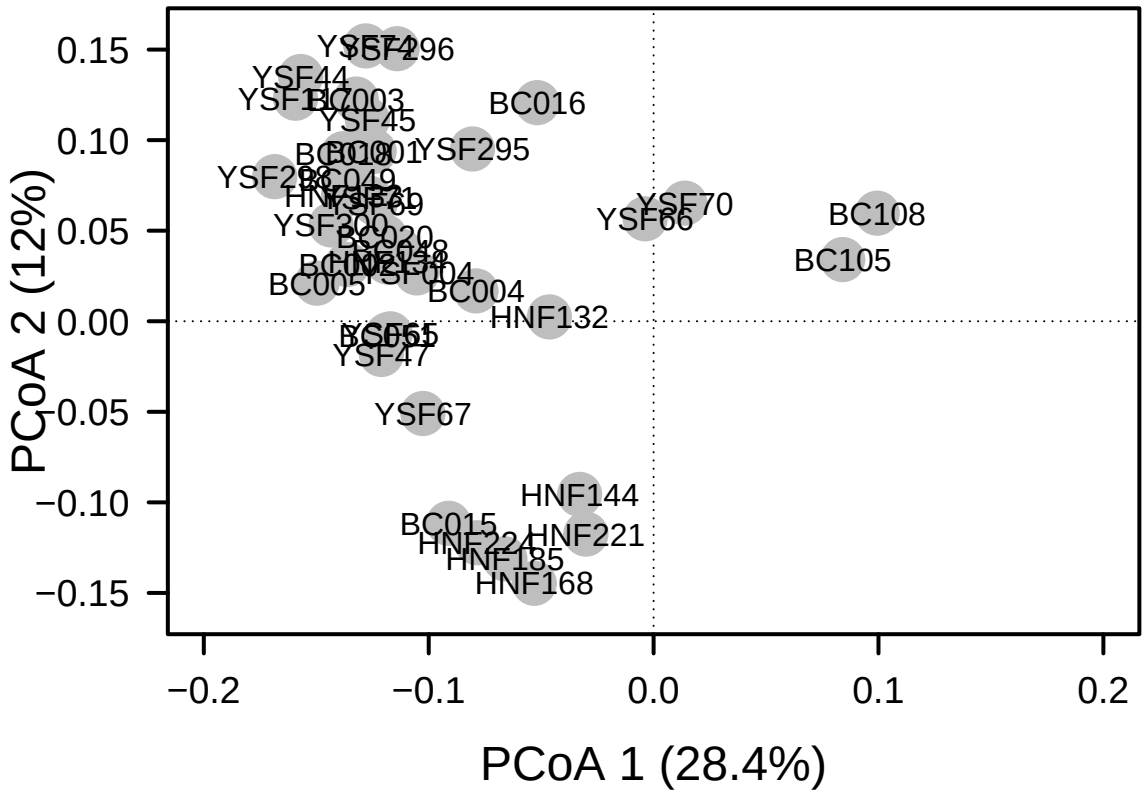
```
# Define plot
par(mar = c(5, 5, 1, 2) + 0.1)

# Initiate plot
plot(pond.pcoa$points[ ,1], pond.pcoa$points[ ,2],
     xlim = c(-0.2, 0.2), ylim = c(-.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.

```
# PCoA on taxonomic data & calculate explained variation
# Making a distance matrix first
comm.df <- vegdist(comm, method = "bray")
tax.pcoa <- cmdscale(comm.df, eig = T, k = 3)
explainvar1 <- round(tax.pcoa$eig[1] / sum(tax.pcoa$eig), 3) * 100
explainvar2 <- round(tax.pcoa$eig[2] / sum(tax.pcoa$eig), 3) * 100
explainvar3 <- round(tax.pcoa$eig[3] / sum(tax.pcoa$eig), 3) * 100
sum.eig <- sum(explainvar1, explainvar2, explainvar3)

# Define plot
par(mar = c(5, 5, 1, 2) + 0.1)

# Initiate plot
plot(tax.pcoa$points[ ,1], tax.pcoa$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(tax.pcoa$points[ ,1], tax.pcoa$points[ ,2],
       pch = 19, cex = 3, bg = "gray", col = "gray")
text(tax.pcoa$points[ ,1], tax.pcoa$points[ ,2],
     labels = row.names(tax.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***: The taxonomic ordination is more clustered based on site than the phylogenetic based ordination and explains more variation within the samples than the phylogenetic ordination. Because we get more information from the taxonomic PCoA, I would say that visualizing phylogenetic information in a PCoA for this dataset is not as important.

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.

```r
# Define environmental category
watershed <- env$Location

# Run PERMANOVA
phylo.adonis <- adonis2(dist.uf ~ watershed, permutations = 999)

# Compare 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 redudnant 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
# Mantel test
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: 0.053
##
## Upper quantiles of permutations (null model):
##   90%   95% 97.5%   99%
## 0.120 0.163 0.200 0.235
## 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
ponds.dbrda <- vegan::dbrda(dist.uf ~ ., data = as.data.frame(scale(envs)))

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

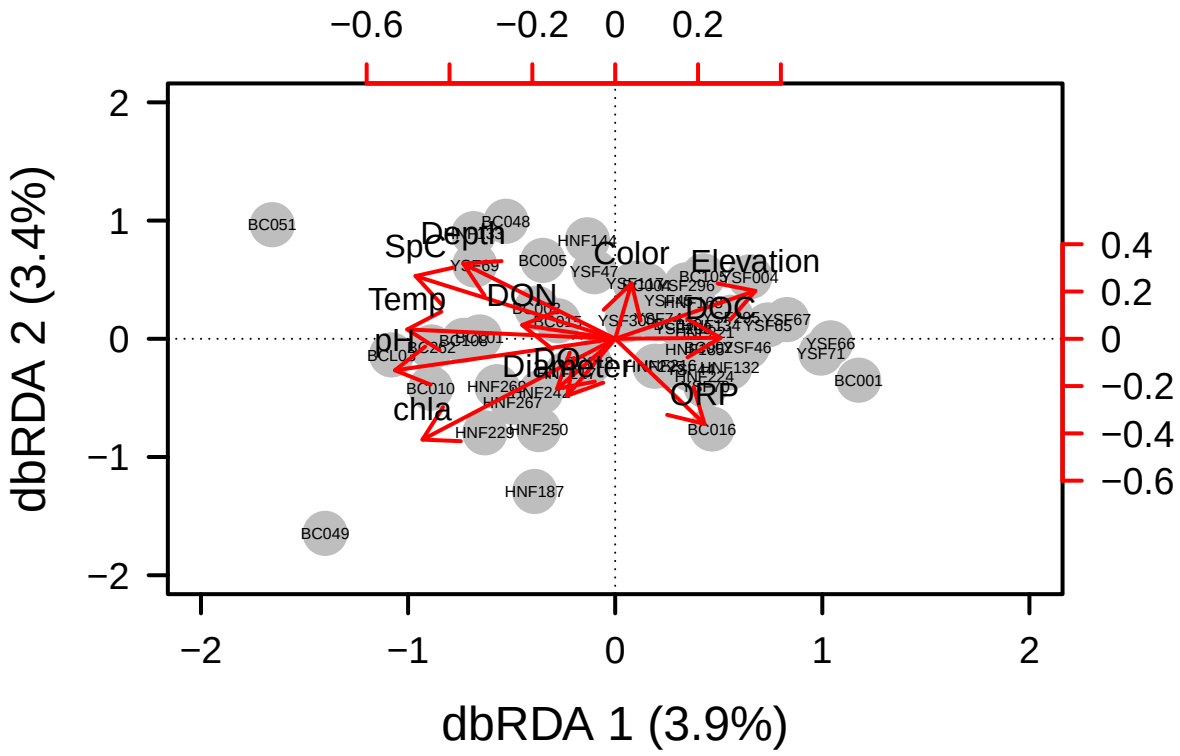
```
## Permutation test for dbrda under reduced model
## Forward tests for axes
## Permutation: free
## Number of permutations: 999
##
## Model: vegan::dbrda(formula = dist.uf ~ Elevation + Diameter + Depth + ORP + Temp + SpC + DO + pH + C
##          Df SumOfSqs      F Pr(>F)
## dbRDA1    1  0.10566 2.0152  0.445
## dbRDA2    1  0.09258 1.7658  0.638
## dbRDA3    1  0.07555 1.4409  0.973
## dbRDA4    1  0.06677 1.2735  0.995
## dbRDA5    1  0.05666 1.0807  1.000
## dbRDA6    1  0.05293 1.0095  1.000
## dbRDA7    1  0.04750 0.9059  1.000
## dbRDA8    1  0.03941 0.7517  1.000
## dbRDA9    1  0.03775 0.7201  1.000
## dbRDA10   1  0.03280 0.6256  1.000
## dbRDA11   1  0.02876 0.5485  1.000
## dbRDA12   1  0.02501 0.4770  1.000
## 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  0.065 .
## Diameter  -0.27972 -0.96008 0.0541  0.260
## Depth     -0.63137  0.77548 0.1756  0.013 *
## ORP        0.41879 -0.90808 0.1437  0.026 *
## Temp      -0.98250  0.18628 0.1523  0.015 *
## SpC       -0.77101  0.63682 0.2087  0.004 **
## DO        -0.39318 -0.91946 0.0464  0.319
## pH        -0.96210 -0.27270 0.1756  0.008 **
## Color      0.06353  0.99798 0.0464  0.312
## chla      -0.60392 -0.79704 0.2626  0.010 **
## DOC        0.99847 -0.05526 0.0382  0.364
## DON       -0.91633  0.40042 0.0339  0.418
## ---
## 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) * 100
dbrda.explainvar2 <- round(ponds.dbrda$CCA$eig[2] /
                     sum(c(ponds.dbrda$CCA$eig, ponds.dbrda$CA$eig)), 3) * 100
# Make dbRDA plot
# Define plot parameters
par(mar = c(5, 5, 4, 4) + 0.1)

# Initiate plot
plot(scores(ponds.dbrda, display = "wa"), 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)
abline(h = 0, v = 0, lty = 3)
box(lwd = 2)

# Add points & labels
points(scores(ponds.dbrda, display = "wa"),
       pch = 19, cex = 3, bg = "gray", col = "gray")
text(scores(ponds.dbrda, display = "wa"),
    labels = row.names(scores(ponds.dbrda, display = "wa")), cex = 0.5)

# Add environmental vectors
vectors <- scores(ponds.dbrda, display = "bp")
#row.names(vectors) <- c("Temp", "DO", "chla", "DON")
arrows(0, 0, vectors[,1] * 2, vectors[, 2] * 2,
       lwd = 2, lty = 1, length = 0.2, col = "red")
text(vectors[,1] * 2, vectors[, 2] * 2, pos = 3,
     labels = row.names(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])))
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])))
```



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***: Based on multiple analyses demonstrated above we can draw multiple conclusions regarding bacterial diversity. First, the taxonomic richness between bacterial communities increases as phylodiversity increases, indicating that the community assemblages may contain evolutionarily divergent taxa. Second, the NRI values for these ponds indicated phylogenetic overdispersion which supports the previous indication of divergent taxa between communities. The NTI values indicated mostly overdispersion within ponds and some clustering, although the results were largely comprised of negative NTI values indicating overdispersion which is consistent with previous results. Third, based on information from the phylogenetic PCoA, we can see that there is not distinct clustering between ponds based on their respective sites, this supports our conclusion that the phylogenetic diversity within the Indiana ponds is characterized by overdispersion, indicating that strong dispersal mechanisms may be at play. However in the taxonomic PCoA there was indication of clustering based on sample site, this may be complimentary to our phylogenetic conclusion of dispersal because ponds within the same fieldsite clustered similarly on the taxonomic PCoA. Lastly, based on the dbRDA analysis performed above, we can see that many environmental factors are correlated with different ponds, which may give us an indication of the environmental filter that is acting within these communities in regards to dispersal.

6) SPATIAL PHYLOGENETIC COMMUNITY ECOLOGY

A. Phylogenetic Distance-Decay (PDD)

A distance decay (DD) relationship reflects the spatial autocorrelation of community similarity. That is, communities located near one another should be more similar to one another in taxonomic composition than

distant communities. (This is analagous to the isolation by distance (IBD) pattern that is commonly found when examining genetic similarity of a populations as a function of space.) Historically, the two most common explanations for the taxonomic DD are that it reflects spatially autocorrelated environmental variables and the influence of dispersal limitation. However, if phylogenetic diversity is also spatially autocorrelated, then evolutionary history may also explain some of the taxonomic DD pattern. Here, we will construct the phylogenetic distance-decay (PDD) relationship

First, calculate distances for geographic data, taxonomic data, and phylogenetic data among all unique pair-wise combinations of ponds.

In the R code chunk below, do the following:
1. calculate the geographic distances among ponds,
2. calculate the taxonomic similarity among ponds,
3. calculate the phylogenetic similarity among ponds, and
4. create a dataframe that includes all of the above information.

```
# Geographic distances among ponds (km)
long.lat <- as.matrix(cbind(env$long, env$lat))
coord.dist <- earth.dist(long.lat, dist = TRUE)

# Taxonomic similarity among ponds (BC)
bray.curtis.dist <- 1 - vegdist(comm)

# Phylogenetic similarity via UniFrac
unifrac.dist <- 1 - dist.uf

# Transform all distances into pairwise long format with melt function
unifrac.dist.mlt <- melt(as.matrix(unifrac.dist))[melt(upper.tri(as.matrix(unifrac.dist)))$value,]
```

```
## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by
## the caller; using TRUE

## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by
## the caller; using TRUE
```

```
bray.curtis.dist.mlt <- melt(as.matrix(bray.curtis.dist))[melt(upper.tri(as.matrix(bray.curtis.dist)))$
```

```
## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by
## the caller; using TRUE

## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by
## the caller; using TRUE
```

```
coord.dist.mlt <- melt(as.matrix(coord.dist))[melt(upper.tri(as.matrix(coord.dist)))$value,]
```

```
## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by
## the caller; using TRUE

## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by
## the caller; using TRUE
```

```
env.dist.mlt <- melt(as.matrix(env.dist))[melt(upper.tri(as.matrix(env.dist)))$value,]
```

```
## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by
## the caller; using TRUE

## Warning in type.convert.default(X[[i]], ...): 'as.is' should be specified by
## the caller; using TRUE
```

```
# Create data frame from the list of distances
df <- data.frame(coord.dist.mlt, bray.curtis.dist.mlt[, 3], unifrac.dist.mlt[, 3],
                 env.dist.mlt[, 3])
names(df)[3:6] <- c("geo.dist","bray.curtis", "unifrac", "env.dist")
```

Now, let's plot the DD relationships:
In the R code chunk below, do the following:
1. plot the taxonomic distance decay relationship,
2. plot the phylogenetic distance decay relationship, and
3. add trend lines to each.

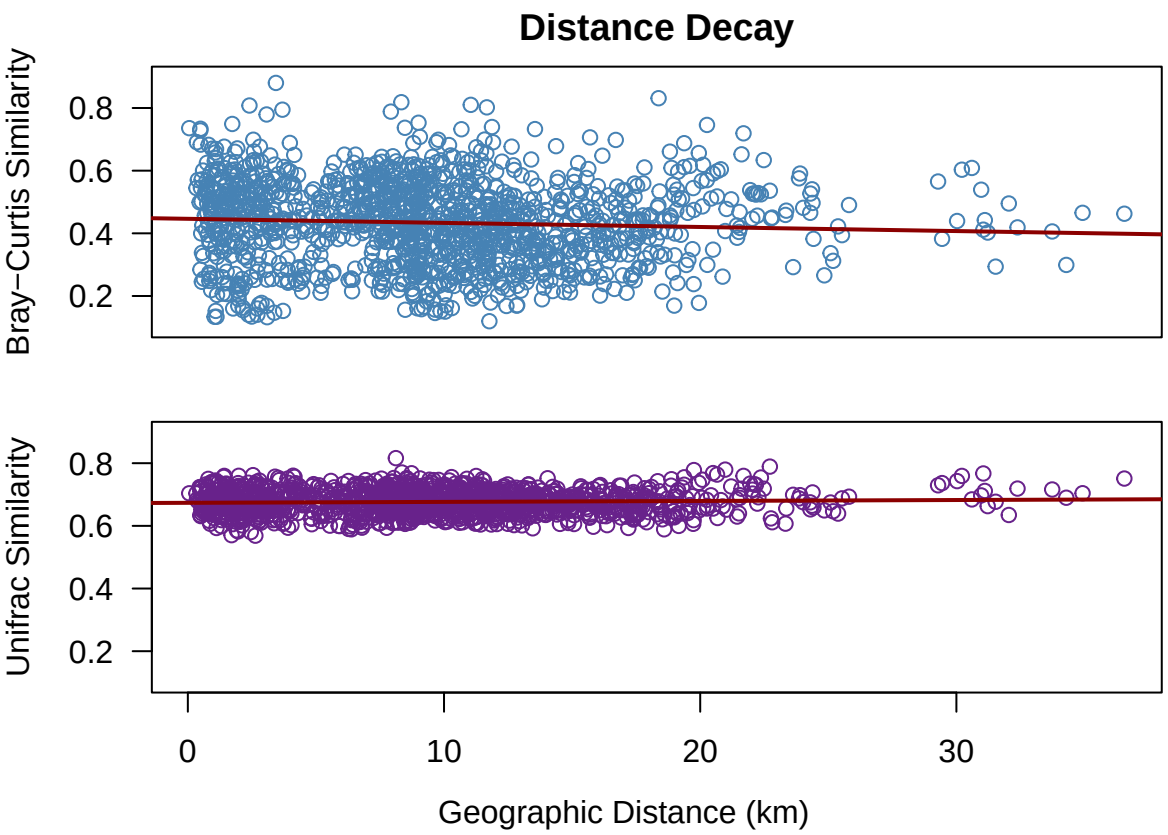
```
# Set initial plot parameters
par(mfrow=c(2, 1), mar = c(1, 5, 2, 1) + 0.1, oma = c(2, 0, 0, 0))

# Make plot for taxonomic DD
plot(df$geo.dist, df$bray.curtis, xlab = "", xaxt = "n", las = 1, ylim = c(0.1, 0.9),
     ylab="Bray-Curtis Similarity",
     main = "Distance Decay", col = "SteelBlue")

# Regression for taxonomic DD
DD.reg.bc <- lm(df$bray.curtis ~ df$geo.dist)
summary(DD.reg.bc)
```

```
##
## Call:
## lm(formula = df$bray.curtis ~ df$geo.dist)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31151 -0.08843  0.00315  0.09121  0.43817
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.4463453  0.0066883  66.735   <2e-16 ***
## df$geo.dist -0.0013051  0.0005864  -2.226   0.0262 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1303 on 1324 degrees of freedom
## Multiple R-squared:  0.003728,   Adjusted R-squared:  0.002975
## F-statistic: 4.954 on 1 and 1324 DF,  p-value: 0.0262
```

```r
abline(DD.reg.bc , col = "red4", lwd = 2)

# New plot parameters
par(mar = c(2, 5, 1, 1) + 0.1)
# Make plot for phylogenetic DD
plot(df$geo.dist, df$unifrac, xlab = "", las = 1, ylim = c(0.1, 0.9),
     ylab = "Unifrac Similarity", col = "darkorchid4")

# Regression for phylogenetic DD
DD.reg.uni <- lm(df$unifrac ~ df$geo.dist)
summary(DD.reg.uni)
```

```
##
## Call:
## lm(formula = df$unifrac ~ df$geo.dist)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.105629 -0.027107 -0.000077  0.026761  0.140215
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.6735186  0.0019206 350.677   <2e-16 ***
## df$geo.dist 0.0002976  0.0001684   1.767   0.0774 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03741 on 1324 degrees of freedom
## Multiple R-squared:  0.002354,   Adjusted R-squared:  0.0016
## F-statistic: 3.124 on 1 and 1324 DF,  p-value: 0.07738
```

```r
abline(DD.reg.uni, col = "red4", lwd = 2)

# Add x-axis label to plot
mtext("Geographic Distance (km)", side = 1, adj = 0.55,
      line = 0.5, outer = TRUE)
```



In the R code chunk below, test if the trend lines in the above distance decay relationships are different from one another.

```
"diffslope" <-
  function(x1, y1, x2, y2, permutations=1000, ic=FALSE, resc.x=FALSE, resc.y=TRUE, trace=FALSE, ...) {
    if (resc.x) {
      maxS <- max(mean(x1), mean(x2)) #das hoehere der beiden means herausfinden
      x1 <- x1+(maxS-mean(x1)) #und auf beide datensaetze anwenden
      x2 <- x2+(maxS-mean(x2))
    }
    if (resc.y) {
      maxD <- max(mean(y1), mean(y2))
      y1 <- y1+(maxD-mean(y1))
      y2 <- y2+(maxD-mean(y2))
    }
    m1 <- data.frame(as.numeric(y1), as.numeric(x1))
    m2 <- data.frame(as.numeric(y2), as.numeric(x2))
    names(m1) <- c("x","y")
    names(m2) <- c("x","y")
    m1.lm <- lm(m1) #die beiden linearen modelle rechnen
    m2.lm <- lm(m2)
    ds0 <- as.numeric(m1.lm$coefficients[2]-m2.lm$coefficients[2]) #deren differenz ausrechnen
    if(ic){
      m12.lmcoeff <- matrix(data=NA, nrow=permutations, ncol=2)
      m21.lmcoeff <- matrix(data=NA, nrow=permutations, ncol=2)
      dic <- as.numeric(m1.lm$coefficients[1]-m2.lm$coefficients[1])
```

```
    if (trace) {cat(permutations, "perms: ")}
    for(i in 1:permutations) {
      tmp1 <- sample(nrow(m1), nrow(m1)/2)
      tmp2 <- sample(nrow(m2), nrow(m2)/2)
      m12 <- rbind(m1[tmp1,], m2[tmp2,])
      m21 <- rbind(m1[-tmp1,], m2[-tmp2,])
      m12.lmcoeff[i,] <- as.numeric(lm(m12)$coefficients)
      m21.lmcoeff[i,] <- as.numeric(lm(m21)$coefficients)
      if (trace) {cat(paste(i,""))}
    }
    perms <- m12.lmcoeff - m21.lmcoeff
    if (ds0 >= 0) {
      signif <- length(perms[perms[,2]>=ds0,2])/permutations
    }
    else {
      signif <- length(perms[perms[,2]<=ds0,2])/permutations
    }
    if (dic >= 0) {
      signific <- length(perms[perms[,1]>=ds0,1])/permutations
    }
    else {
      signific <- length(perms[perms[,1]<=ds0,1])/permutations
    }

    if (signif == 0) {
      signif <- 1/permutations
    }
    if (signific == 0) {
      signific <- 1/permutations
    }
  }
  else{
    perms <- vector("numeric", permutations)
    if (trace) {cat(permutations, "perms: ")}
    for(i in 1:permutations) {
      tmp1 <- sample(nrow(m1), nrow(m1)/2)
      tmp2 <- sample(nrow(m2), nrow(m2)/2)
      m12 <- rbind(m1[tmp1,], m2[tmp2,])
      m21 <- rbind(m1[-tmp1,], m2[-tmp2,])
      perms[i] <- as.numeric(lm(m12)$coefficients[2]-lm(m21)$coefficients[2])
      if (trace) {cat(paste(i,""))}
    }
    if (ds0 >= 0) {
      signif <- length(perms[perms>=ds0])/permutations
    }
    else {
      signif <- length(perms[perms<=ds0])/permutations
    }
    if (signif == 0) {
      signif <- 1/permutations
    }
    perms <- cbind(perms,perms)
  }
```

```
    res <- c(call=match.call())
    res$slope.diff <- as.numeric(ds0)
    res$signif <- signif
    res$permutations <- permutations
    res$perms <- perms[,2]
    class(res) <- "dsl"
    if(ic) {
      res$intercept <- as.numeric(dic)
      res$signific <- as.numeric(signific)
      res$permsic <- as.numeric(perms[,1])
      class(res) <- "dsl2"
    }
    res
  }


diffslope(df$geo.dist, df$unifrac, df$geo.dist, df$bray.curtis)
```

```
## $call
## diffslope(x1 = df$geo.dist, y1 = df$unifrac, x2 = df$geo.dist,
##     y2 = df$bray.curtis)
##
## $slope.diff
## [1] 0.001602746
##
## $signif
## [1] 0.001
##
## $permutations
## [1] 1000
##
## $perms
##    [1] -1.096972e-03 -1.942045e-04 -3.016768e-04 -1.016493e-03  1.415802e-03
##    [6] -1.961433e-04 -4.073683e-04  2.163275e-05 -1.984406e-04  1.454881e-03
##   [11] -2.537816e-04  1.384921e-03  3.780675e-06  6.676900e-05 -2.140625e-04
##   [16] -8.028123e-04  6.372354e-04 -3.533418e-04  1.226846e-04 -3.666524e-04
##   [21]  6.475500e-05 -4.812402e-04  2.581355e-04 -3.984382e-04 -1.909516e-04
##   [26] -3.950040e-04  2.367368e-04  8.587354e-04 -6.158361e-04  5.804591e-04
##   [31]  3.933385e-04 -1.546810e-04  5.217249e-04  3.903718e-04  1.036214e-03
##   [36]  7.067250e-04  5.373023e-04  8.237502e-04  1.263535e-04  1.962060e-04
##   [41]  8.602690e-05  8.894645e-04 -1.153809e-03  2.029224e-04  5.932018e-04
##   [46]  6.452425e-04  3.773584e-04 -7.283404e-04  1.388837e-04  3.415246e-04
##   [51] -3.085638e-04 -6.643894e-05  2.067597e-04  6.694956e-05 -5.281072e-04
##   [56]  8.648724e-04  1.034183e-03  7.676030e-04 -9.305360e-04  3.061486e-04
##   [61]  1.038365e-03 -6.217992e-05  5.598464e-04  1.424788e-03 -3.648734e-04
##   [66] -2.785136e-04  4.386898e-05 -4.483284e-04 -8.467755e-04  7.633127e-04
##   [71] -5.150419e-04  1.540811e-06 -1.770517e-04 -3.216890e-04 -1.853841e-04
##   [76]  1.156455e-04 -1.075770e-03  6.668627e-04 -3.288285e-04  5.052467e-04
##   [81]  3.028383e-04 -1.194593e-04  1.466306e-04 -3.801361e-04 -3.120234e-04
##   [86]  9.053033e-04  2.505430e-05  5.721202e-04  3.736017e-04  2.282505e-04
##   [91]  9.902640e-05  5.638250e-04  1.461106e-03 -1.208483e-04  7.044507e-04
##   [96]  9.864911e-06  8.161205e-06 -4.822628e-04 -4.363147e-04  6.322329e-05
##  [101] -2.531616e-04  2.081825e-04 -6.290148e-04  4.265357e-04  1.131542e-04
```

```
##  [106] -1.285089e-04  2.769567e-04  9.270549e-04 -1.675918e-04  3.188535e-04
##  [111]  4.296793e-04  2.699477e-04 -6.422939e-04 -1.253582e-03  4.454690e-04
##  [116]  5.575554e-04 -2.159187e-04 -1.464097e-03 -2.557809e-04  9.743229e-04
##  [121]  1.492002e-04  1.598211e-04  9.847250e-04  4.402982e-04 -8.307506e-04
##  [126] -6.404991e-04 -4.673051e-04  4.370396e-04  2.607424e-04  7.551526e-04
##  [131] -1.593944e-04 -2.608317e-04 -9.145593e-04  3.039545e-04  4.712702e-05
##  [136] -5.927384e-05  4.409745e-04  6.350350e-04 -4.412684e-04  6.742504e-04
##  [141]  8.244716e-04  8.417432e-04 -3.786937e-05  1.818088e-04 -2.645287e-04
##  [146] -4.228867e-04 -2.997369e-04 -2.410776e-04  1.044171e-04  7.374797e-04
##  [151]  1.028500e-03  5.864229e-04 -5.372659e-04 -1.551856e-04 -1.518313e-04
##  [156]  1.532342e-04  4.106986e-04 -5.543027e-04 -8.558412e-05 -8.472591e-04
##  [161]  6.159450e-04  3.281879e-04 -5.132028e-04  2.573000e-05 -4.855805e-05
##  [166] -7.496243e-04 -3.027560e-04 -2.491430e-04  9.470413e-04  6.687274e-04
##  [171] -6.952459e-04  2.737056e-04 -6.053973e-04  9.999248e-04 -8.094991e-04
##  [176] -2.227129e-04 -2.927304e-04  7.176943e-04  8.316267e-04  1.887195e-04
##  [181]  3.425363e-04 -6.765899e-04  7.264579e-04  8.281226e-04 -5.781735e-04
##  [186] -8.363197e-04  4.061516e-04  1.323711e-04 -3.897208e-04 -4.224908e-04
##  [191]  1.030142e-04  5.460321e-04 -6.901409e-04 -6.475275e-04  1.089378e-03
##  [196]  7.852004e-04  5.204778e-04  1.637525e-04  3.355765e-04 -4.823994e-04
##  [201]  1.288735e-03 -3.146966e-04 -2.457830e-04  8.177459e-04  6.282854e-04
##  [206]  2.247618e-04  3.575867e-04 -3.933500e-04 -8.584169e-05  4.506562e-05
##  [211]  7.885584e-04 -1.168042e-04 -7.078522e-04 -4.661068e-04  3.388027e-04
##  [216]  3.376262e-04 -3.101231e-05  1.226329e-03  2.750911e-04 -6.978379e-04
##  [221]  1.730809e-04 -1.600950e-04 -7.249431e-04 -1.455530e-03  7.359999e-04
##  [226] -5.346532e-04  1.126683e-03  1.226696e-04 -2.185416e-04  4.139994e-04
##  [231] -3.154569e-04 -1.275053e-04 -1.153123e-04  6.733073e-04 -7.216261e-04
##  [236] -7.127768e-04 -2.907570e-04 -8.485671e-04  3.748974e-04 -1.941715e-04
##  [241] -6.476286e-04  3.498713e-04  3.629765e-04  1.115795e-04 -9.415155e-04
##  [246] -1.045450e-03  1.253053e-03  1.031085e-03  8.708869e-05 -3.121777e-04
##  [251]  8.002938e-04 -8.272817e-04 -2.244074e-04  4.733399e-04 -2.867630e-04
##  [256]  7.370476e-05 -8.774215e-04  1.008571e-03 -1.283313e-04  2.688312e-04
##  [261] -1.672441e-03  3.879473e-04 -2.693038e-04  1.427459e-04  8.152499e-04
##  [266]  3.880262e-04 -1.267967e-03  3.217352e-04 -1.850489e-05  7.867238e-04
##  [271]  6.416240e-04  2.243141e-05 -5.501269e-04 -4.084144e-04 -2.908023e-04
##  [276] -2.775604e-04 -9.715653e-04  3.490374e-04 -8.062296e-04  1.742139e-04
##  [281] -6.990080e-04 -2.267548e-05  5.483112e-04 -8.658717e-04 -3.668417e-04
##  [286]  8.551676e-05  1.155352e-03 -1.258203e-03 -5.776875e-04 -2.412249e-04
##  [291] -3.134822e-04 -8.753958e-05  1.026351e-03 -5.944796e-04 -8.014677e-04
##  [296]  2.310978e-04  9.957258e-04  1.006109e-03  2.562244e-04 -1.428551e-04
##  [301]  7.706694e-04  7.024025e-04 -1.373522e-04 -1.972989e-04  2.688896e-04
##  [306] -4.786184e-05  1.268800e-04  4.274488e-04 -6.456271e-04  7.950785e-04
##  [311]  5.917567e-04 -8.699925e-05  3.061332e-04 -3.827463e-04 -1.480126e-05
##  [316]  1.797124e-04  9.789701e-05  4.670900e-07  5.023400e-04  6.477712e-04
##  [321]  8.454955e-04 -2.854728e-04 -1.348041e-03  6.538344e-04  5.426820e-04
##  [326] -9.006085e-04 -9.941445e-05  1.498229e-04  1.039343e-03  1.453514e-04
##  [331] -6.509111e-04  3.645481e-04  1.933219e-04 -5.355928e-04 -3.736936e-04
##  [336]  8.654728e-04  7.655753e-05  4.593231e-04 -3.023533e-04 -9.901451e-04
##  [341]  9.809184e-05 -2.077750e-04  8.952756e-04 -1.907056e-04 -1.453470e-04
##  [346] -7.410150e-04  1.921343e-04 -4.489443e-04  6.395391e-04  1.094810e-03
##  [351]  1.814564e-04 -4.885961e-04  1.134534e-03 -1.087480e-04 -4.235320e-04
##  [356]  2.078129e-05  2.626059e-04 -5.777650e-04  3.625140e-04 -2.783768e-04
##  [361]  9.081267e-04  7.433537e-04 -1.145709e-04  7.110257e-04 -5.685475e-04
##  [366]  8.938959e-04  9.250183e-04 -3.280807e-04 -4.551175e-06  2.208446e-05
##  [371] -2.623364e-04 -1.218155e-03 -6.941827e-04 -1.131174e-03 -3.110530e-04
```

```
##  [376]  1.154858e-04 -2.054419e-04  1.905057e-04 -9.600824e-04 -2.212745e-04
##  [381]  6.332805e-04  9.070141e-04 -4.163512e-04  4.361127e-06  4.193380e-04
##  [386]  8.067921e-04  7.947738e-04 -8.633316e-05  3.431021e-04  2.933480e-04
##  [391] -4.902797e-05  3.179319e-04 -9.902379e-05  1.261345e-05  6.288376e-04
##  [396]  1.081428e-03 -4.222828e-05 -6.313785e-06  1.888834e-04 -2.994771e-04
##  [401] -9.540043e-04 -5.413983e-05 -8.616963e-05 -2.303288e-04  1.029386e-04
##  [406] -5.267172e-04  7.709484e-05 -6.765472e-04  7.482928e-04  1.233859e-04
##  [411]  7.839759e-04  1.871118e-04  1.721849e-04 -1.664067e-04  4.437709e-04
##  [416] -8.767623e-04  9.971777e-04  2.996766e-05 -1.327949e-04 -5.819027e-05
##  [421]  2.308679e-05 -6.607154e-04 -6.833597e-04  5.957464e-04  8.252478e-04
##  [426]  2.678026e-04 -1.106703e-04 -2.861377e-05 -4.134328e-05 -1.542972e-04
##  [431]  2.049623e-04 -3.033141e-04  7.002036e-04  5.794212e-04  1.307853e-05
##  [436]  2.745379e-04  9.675768e-05 -3.495825e-04  1.708687e-04 -5.426364e-04
##  [441] -8.403032e-04 -3.591370e-04  4.427060e-04 -1.263897e-04 -5.807498e-04
##  [446] -1.441288e-04 -6.091153e-04 -5.386608e-04 -1.432262e-03  2.691747e-05
##  [451]  7.245678e-04 -1.346158e-04  4.655692e-04  2.512384e-04 -2.006931e-04
##  [456] -1.202054e-03 -4.484957e-04 -8.939018e-04  7.838205e-04 -1.119796e-03
##  [461]  1.186147e-08 -5.261853e-04 -1.134545e-04  2.644643e-04 -7.899966e-05
##  [466] -2.978786e-04 -5.512795e-04 -1.231118e-03 -1.023368e-03  1.172856e-03
##  [471]  3.733274e-04 -5.589676e-06  6.882833e-05 -7.749330e-04 -6.746763e-04
##  [476]  8.270444e-04  2.995309e-04  9.555858e-04 -1.033046e-03  1.998517e-04
##  [481]  1.741615e-04  9.692953e-04  8.568093e-04  3.979664e-04 -1.240414e-04
##  [486]  3.068459e-04 -2.724098e-04 -1.824947e-04 -9.091081e-04  8.807814e-04
##  [491] -4.489679e-04  2.643624e-04  2.460931e-04 -5.591362e-04  1.344027e-04
##  [496] -4.540495e-04  2.975696e-04  3.222015e-04  2.228798e-04 -3.042301e-04
##  [501] -1.098516e-03  6.529652e-05  1.853996e-04 -5.530347e-04  9.779392e-04
##  [506]  1.133951e-04 -3.472612e-04 -6.723231e-04 -2.195546e-04 -2.750266e-04
##  [511]  3.383451e-04  7.194439e-04  8.044905e-04  3.772862e-04  2.572494e-04
##  [516] -1.062224e-03 -5.422005e-04 -7.354581e-04  2.222741e-04 -4.122034e-04
##  [521]  3.087253e-04 -8.904363e-04 -6.357513e-05 -5.547610e-04  3.558515e-04
##  [526]  1.865065e-04 -1.211714e-03  3.877397e-04 -6.876355e-04 -5.675808e-04
##  [531] -1.386012e-03  8.135815e-04 -7.746865e-05 -9.812884e-04 -3.253270e-04
##  [536] -3.452717e-05 -3.958492e-05 -7.396839e-04 -2.126244e-04  9.435964e-04
##  [541] -2.863581e-05  5.520726e-04 -3.345424e-05  1.321240e-03  3.231041e-05
##  [546] -6.263865e-04 -1.009775e-04 -1.088016e-03 -5.260775e-04  2.811036e-04
##  [551] -2.976048e-04 -9.001781e-04  1.218840e-03 -1.055717e-03  1.119600e-03
##  [556]  5.240201e-04  1.054562e-03  6.703250e-04  5.493275e-04  7.637750e-04
##  [561]  2.389669e-05 -5.823286e-04 -1.215386e-04  8.564542e-04  7.297751e-04
##  [566] -6.084208e-04  1.519869e-04  4.309502e-04 -1.228415e-03 -1.650774e-04
##  [571]  3.373548e-05 -5.542282e-04  6.908162e-04  7.101818e-04  7.935513e-04
##  [576] -1.814237e-04  6.410283e-04  7.528750e-04 -4.069769e-04 -1.067729e-03
##  [581] -9.184007e-04  4.794408e-04  9.900189e-05  1.082862e-03 -2.849483e-04
##  [586]  1.522218e-03 -4.091403e-05  2.365086e-04 -3.368755e-04  5.965137e-04
##  [591]  5.189165e-04 -3.983067e-04 -7.917999e-04 -1.147122e-03  1.050345e-04
##  [596]  6.834090e-04  3.625376e-04 -9.618978e-04 -7.844199e-04  4.586086e-04
##  [601] -3.216681e-04 -2.325242e-04 -2.513048e-04  7.908472e-05 -1.025576e-03
##  [606] -6.165855e-04  1.029918e-03 -2.882398e-04  9.532496e-04 -7.367311e-04
##  [611]  5.007832e-04  5.863531e-04 -3.841126e-04 -7.583665e-04  7.222378e-05
##  [616] -7.570398e-04  3.721041e-04 -3.892595e-04  1.359265e-04  1.299669e-04
##  [621]  4.867253e-04 -1.014655e-03  2.752017e-04  5.321260e-04  1.273962e-03
##  [626] -7.992899e-04  3.189149e-04  1.581196e-04  1.797219e-04  1.367632e-03
##  [631]  6.307861e-04 -3.146909e-04  4.155882e-04  4.752101e-04  3.253918e-04
##  [636] -3.103529e-04 -4.683611e-04 -1.077307e-04  5.821424e-04 -1.457504e-04
##  [641] -1.876544e-04 -2.471805e-05  2.245521e-04 -6.536261e-04  2.256560e-04
```

```
##  [646]  2.442638e-04  4.026573e-05 -4.659977e-04  1.076133e-04  2.843052e-04
##  [651]  1.080936e-03  6.483380e-04 -5.805801e-04 -2.062367e-05  1.106921e-05
##  [656]  8.413901e-04 -5.376171e-04 -2.862521e-04 -5.268614e-04  1.089180e-04
##  [661]  2.088606e-04  2.022842e-04 -5.799469e-04  4.523807e-04  3.886257e-04
##  [666]  8.005551e-04 -1.024120e-03  4.234579e-04 -8.493704e-04  8.347647e-04
##  [671]  1.908606e-04  1.992042e-04 -1.219432e-03 -1.632370e-04 -6.434965e-04
##  [676] -2.944143e-04 -3.232518e-05  3.591451e-04 -5.048753e-04 -3.844313e-04
##  [681]  8.276086e-04  7.660574e-04  9.082660e-05 -9.068509e-04  6.731114e-04
##  [686]  1.843370e-04 -3.299169e-04  1.167418e-03  4.488802e-04  1.092525e-03
##  [691] -6.325633e-04 -2.108265e-04 -1.231545e-04 -1.869085e-04  7.893591e-04
##  [696] -2.944261e-04 -9.243378e-05 -5.936998e-04 -2.315846e-04  5.204389e-04
##  [701]  1.050782e-03  1.083817e-04 -7.397745e-04  1.100610e-03  8.805696e-05
##  [706]  2.560780e-04  3.178242e-04 -7.021235e-04 -2.241627e-04 -2.073228e-05
##  [711] -5.646735e-04  4.619838e-04 -2.932248e-05  1.841936e-04 -4.642810e-04
##  [716] -1.218734e-03  7.407372e-04  6.290947e-04 -3.764556e-04 -2.378139e-04
##  [721] -6.007232e-04  2.020179e-05  8.013145e-04 -5.725294e-04  1.311422e-04
##  [726] -6.175959e-04  9.972405e-05 -6.090256e-04  1.124320e-03  1.772387e-05
##  [731] -8.507009e-05  1.599922e-04 -3.872302e-04 -3.567182e-04 -6.150592e-04
##  [736] -4.860203e-04  3.150216e-04  4.401578e-04 -1.554457e-04 -6.375100e-04
##  [741] -8.752990e-04  3.831527e-04 -1.572265e-04 -2.782285e-04 -2.331773e-04
##  [746] -7.311000e-04 -7.017696e-04  1.436622e-03 -6.570143e-04 -5.940100e-04
##  [751] -2.576579e-04  3.016134e-04  7.435316e-05 -3.058142e-04 -5.641042e-04
##  [756] -1.232896e-04 -1.300421e-03  2.093995e-04 -6.441742e-04  8.939202e-04
##  [761] -3.810024e-04 -1.432806e-04 -1.174202e-03  1.905346e-04  4.568240e-04
##  [766]  5.965905e-04  6.281612e-04 -2.955231e-04  2.046952e-04  1.174428e-04
##  [771] -6.018219e-05  2.060367e-04 -2.236546e-05  4.310718e-04  1.277928e-04
##  [776]  1.527009e-03 -1.916483e-04 -5.853376e-04 -9.298293e-04 -4.787934e-04
##  [781]  8.897247e-04 -6.874943e-04 -1.497617e-04  4.271438e-04 -5.633917e-04
##  [786] -3.197991e-04 -1.052986e-03  1.021845e-03 -1.438143e-04 -1.158602e-03
##  [791] -4.643562e-05 -3.723027e-04  1.068631e-04 -7.960994e-04 -1.599365e-04
##  [796]  5.343008e-04  9.769189e-05 -5.802347e-04 -5.147521e-04 -1.185444e-03
##  [801] -3.709275e-04 -9.217900e-04  1.359248e-03  5.928363e-04  3.254113e-04
##  [806]  2.453740e-04  8.861420e-04 -2.519656e-04 -2.891151e-04 -8.870418e-04
##  [811] -6.686193e-04  1.002362e-03  3.005877e-04  9.602127e-04 -1.300569e-04
##  [816] -1.174971e-03 -6.439534e-04  2.288311e-04 -4.841148e-04  6.803167e-04
##  [821]  2.923283e-04 -7.212517e-04  5.899123e-04 -1.169493e-03  1.375923e-04
##  [826] -2.691535e-05 -5.879591e-05 -2.105201e-05 -7.537127e-05  3.057448e-04
##  [831]  5.263537e-05 -1.079330e-04  1.087924e-04 -9.691554e-04  2.389656e-04
##  [836]  9.444633e-04  4.661849e-04 -1.747866e-04  3.525276e-04  1.209400e-03
##  [841] -2.479373e-04 -5.529787e-04  6.372913e-04  2.331572e-05  2.685276e-04
##  [846]  1.660088e-04 -3.707501e-04 -7.303671e-04  3.778964e-04 -7.705357e-04
##  [851]  2.404277e-04 -5.982989e-04 -5.659225e-04  4.029443e-05 -5.270528e-05
##  [856] -6.259375e-04  1.297373e-03 -1.717549e-04  1.921771e-04  7.889515e-04
##  [861]  7.112933e-04  4.472890e-04  4.204391e-04 -2.980788e-04  1.857416e-04
##  [866]  4.438699e-04 -3.439581e-04  4.140637e-04 -4.763450e-04 -4.089038e-04
##  [871] -2.379031e-04 -5.204653e-05 -8.025344e-04 -1.035690e-03  1.256379e-04
##  [876]  3.230847e-04 -6.801241e-04 -4.439907e-04  2.331386e-04  7.611380e-04
##  [881] -3.239407e-04 -1.449408e-03  4.956821e-04  4.535264e-04  5.290557e-04
##  [886]  7.626911e-04  7.586404e-04  3.436480e-05  8.423659e-05  8.473130e-04
##  [891]  1.170882e-03 -7.223937e-04  5.210063e-04 -5.900176e-05 -9.045143e-04
##  [896] -1.549091e-04  3.970131e-04 -2.099538e-04 -7.552757e-04  1.446084e-04
##  [901] -3.847396e-04 -2.306285e-04  8.758011e-04  3.655735e-04  1.062504e-03
##  [906]  3.911562e-05  1.486218e-04  1.575579e-04 -1.805824e-05 -3.904609e-04
##  [911]  2.247757e-04 -3.120564e-04 -1.070007e-03 -5.381586e-06  3.266786e-04
```

```
##  [916]  6.013201e-04 -9.446140e-05  4.368026e-04 -4.580794e-04  3.019870e-04
##  [921] -1.763919e-03 -1.012394e-03 -1.557084e-04 -1.111472e-03 -3.858061e-04
##  [926]  2.307580e-04 -1.122779e-04 -6.774076e-04  6.349596e-05  1.121438e-03
##  [931]  6.195326e-04  4.894876e-04  1.961063e-04  1.986820e-04  1.939202e-04
##  [936] -2.750389e-04  8.343361e-04 -2.767161e-05  4.466570e-06 -5.111070e-04
##  [941] -3.220482e-04  3.984699e-05 -4.538262e-04  4.037352e-04 -4.465113e-04
##  [946] -7.386730e-04 -2.057454e-04  7.275771e-04  4.462890e-04  6.098632e-04
##  [951] -2.091703e-04 -3.499186e-04  1.485421e-03  5.180121e-04  5.165402e-04
##  [956]  7.048990e-05 -3.478447e-04 -3.608045e-04  1.354916e-03 -2.749908e-04
##  [961]  3.364731e-04  1.217143e-04  2.046628e-04 -1.086268e-03  3.001825e-04
##  [966] -6.403779e-04  3.035495e-05  3.117229e-04 -7.320849e-04  8.822110e-05
##  [971]  1.496499e-03 -3.771744e-04 -1.359558e-03 -6.651057e-04  5.695615e-04
##  [976] -1.096072e-04  5.219153e-04  4.478058e-05 -2.005397e-04 -1.757215e-03
##  [981]  5.401530e-04 -4.392108e-04  4.187576e-04  2.459857e-04 -1.480397e-04
##  [986] -4.043563e-04  5.006208e-04  3.952615e-04 -6.358054e-05  2.278991e-04
##  [991]  2.365655e-05 -2.023354e-05  1.999228e-04 -9.390757e-04 -6.495331e-04
##  [996]  1.407037e-04  2.049718e-04  4.013642e-04 -1.115474e-03 -1.880763e-05
##
## attr(,"class")
## [1] "dsl"
```

Question 7: Interpret the slopes from the taxonomic and phylogenetic DD relationships. If there are differences, hypothesize why this might be.

> ***Answer 7***: There is a significant difference in slopes for the phylogenetic DD relationships between Bray-Curtis distance and UniFrac distance. The Bray-Curtis slope is steeper than the UniFrac slope and this may be due to more taxonomic dissimilarities between the ponds than phylogenetic dissimiliarities.

SYNTHESIS

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.

> Synthesis Answer: I think that phylogenetic information would be useful in determining or making connections between microbial assembly and disturbance (such as permafrost thaw). For example, maybe one could ask whether or not communities that are more resistant or resilient to a disturbance are more or less phylogenetically similar? Is the resulting community post-disturbance more or less phylogenetically similar to the pre-thaw community and what are the important processes or abiotic factors that are playing into that? I don't think this is particularly ground-breaking for the field, but it may give insight into how evolution may or may not dictate assembly patterns. To answer the above, you would need phylogenetic information (probably based on 16S rRNA sequencing) of pre- and post-thaw communities in addition to environmental data.