9.Phylogenetic Diversity - Communities

Yongsoo Choi; Z620: Quantitative Biodiversity, Indiana University

05 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 5$^{\text{th}}$, 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.

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

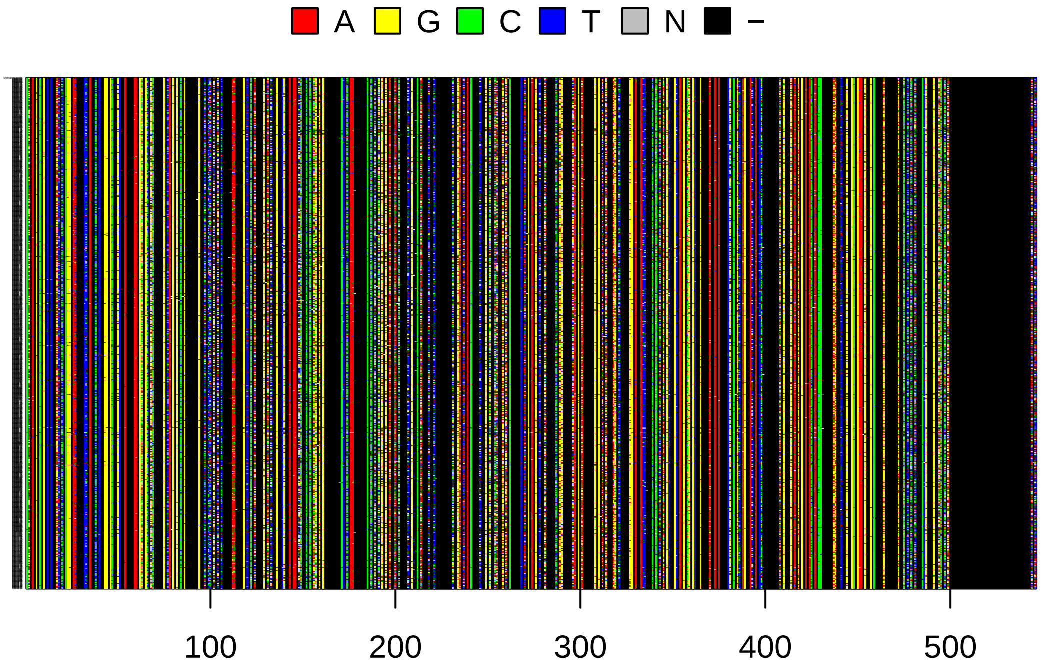
In the R code chunk below, do the following:
1. load the environmental data for the Brown County ponds (*20130801_PondDataMod.csv*),
2. load the site-by-species matrix using the `read.otu()` function,
3. subset the data to include only DNA-based identifications of bacteria,
4. rename the sites by removing 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)
comm <- read.otu(shared = "./data/INPonds.final.rdp.shared", cutoff = "1")
comm <- comm[grep("*-DNA", rownames(comm)), ]
rownames(comm) <- gsub("\\-DNA", "", rownames(comm))
rownames(comm) <- gsub("\\_", "", rownames(comm))
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
```

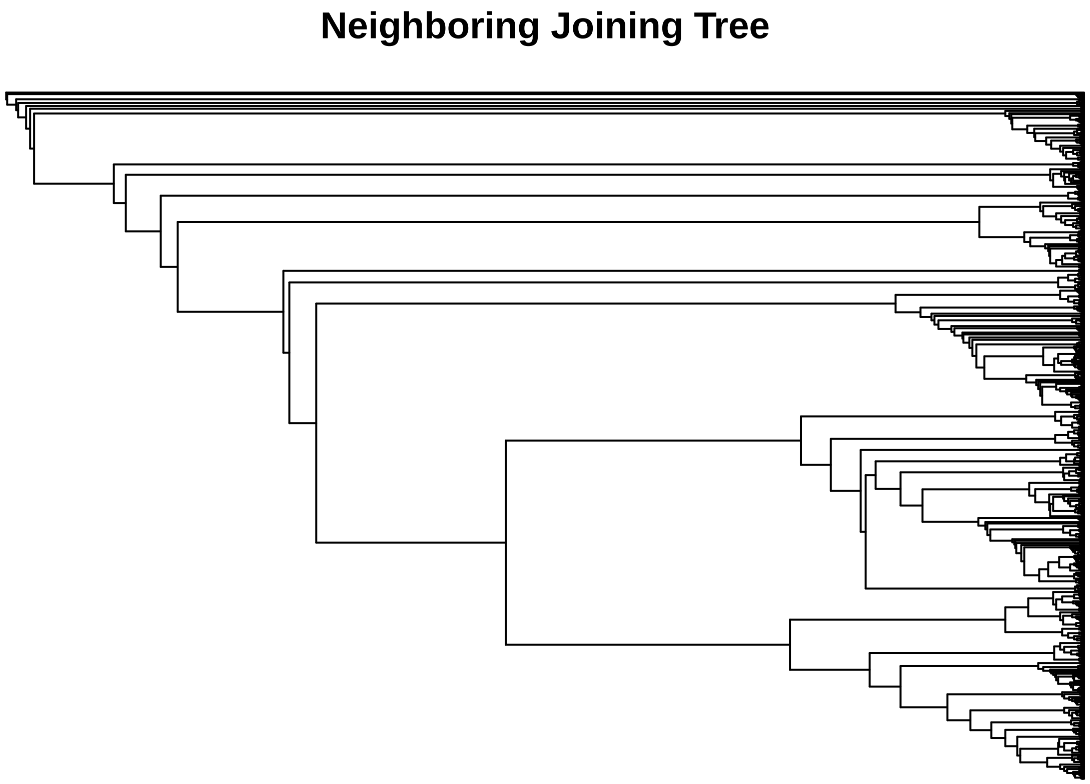
Next, in the R code chunk below, do the following:
1. load the FASTA alignment for the bacterial operational taxonomic units (OTUs),
2. rename the OTUs by removing everything before the tab (\t) and after the bar (|),
3. import the *Methanosarcina* outgroup FASTA file,
4. convert both FASTA files into 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.

```r
ponds.cons <- read.alignment(file = "./data/INPonds.final.rdp.1.rep.fasta",
                             format = "fasta")
ponds.cons$nam <- gsub(".*\t", "", ponds.cons$nam)
ponds.cons$nam <- gsub("\\|.*", "", ponds.cons$nam)

outgroup <- read.alignment(file = "./data/methanosarcina.fasta", format = "fasta")
DNAbin <- rbind(as.DNAbin(outgroup), as.DNAbin(ponds.cons))

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



```r
seq.dist.jc <- dist.dna(DNAbin, model = "JC", pairwise.deletion = F)
phy.all <- bionj(seq.dist.jc)
phy <- drop.tip(phy.all, phy.all$tip.label[!phy.all$tip.label %in%
                                              c(colnames(comm), "Methanosarcina")])
outgroup <- match("Methanosarcina", phy$tip.label)
phy <- root(phy, outgroup, resolve.root = TRUE)
par(mar = c(1, 1, 2, 1) + 0.1)
plot.phylo(phy, main = "Neighboring Joining Tree", "phylogram",
           show.tip.label = F, use.edge.length = F,
           direction = "right", cex = 0.6, lable.offset = 1)
```

```
## Warning in plot.window(...): "lable.offset" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lable.offset" is not a graphical parameter

## Warning in title(...): "lable.offset" is not a graphical parameter
```



4) PHYLOGENETIC ALPHA DIVERSITY

A. Faith's Phylogenetic Diversity (PD)

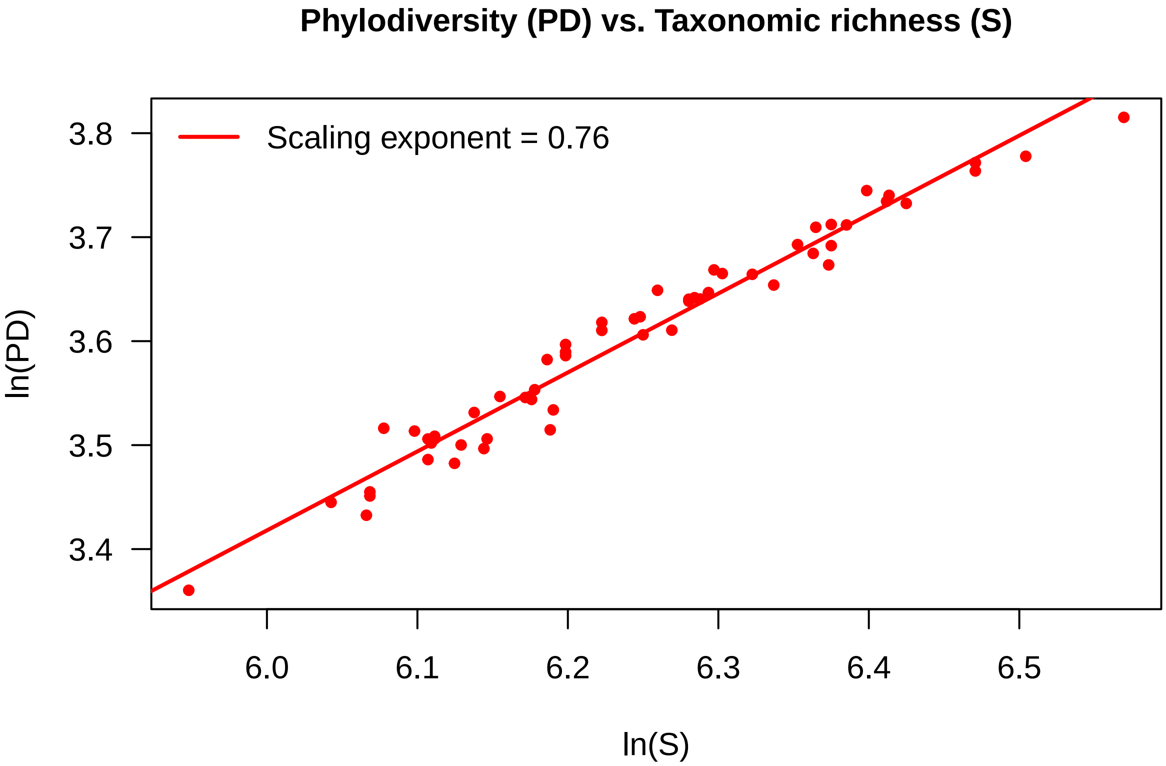
In the R code chunk below, do the following:
1. calculate Faith's D using the `pd()` function.

```
pd <- pd(comm, phy, include.root = F)
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
## HNF189 38.15895 536
## HNF190 28.80133 383
## HNF191 34.17077 463
## HNF216 36.47943 492
## HNF217 36.97870 504
## HNF221 38.11234 538
## HNF224 39.05581 546
## HNF225 38.10391 534
## HNF229 34.25624 488
## HNF236 33.00523 466
## HNF237 36.82098 518
## HNF242 34.92780 482
## HNF250 34.66701 479
## HNF267 33.60526 487
## HNF269 32.54033 457
## HNF279 43.10564 646
## 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.

```r
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)")

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***: PD is calculated by summing the branch lenthgs for each species from root to tips. However, it does not count shared branches more than once. Thus, higher PD value indicates more evolutionary divergent taxa in an assemblage. ***Answer 1b***: If there are many relative related species, PD will be significantly lower than S value. ***Answer 1c***: 0.76 is lower than 1, and it means PD is not linearly increased as S increases. However, it still shows larger S has relatively larger PD than smaller S.

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.

```
ses.pd1 <- ses.pd(comm[1:2, ], phy, null.model = "richness", runs = 25,
                  include.root = F)
ses.pd2 <- ses.pd(comm[1:2, ], phy, null.model = "frequency", runs = 25,
                  include.root = F)
ses.pd1
```

```
##        ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank   pd.obs.z  pd.obs.p
## BC001    668 43.71912     43.82707  0.7585043          15 -0.1423108 0.5769231
## BC002    587 40.94334     39.76377  1.0518409          24  1.1214310 0.9230769
##        runs
## BC001    25
## BC002    25
```

```
ses.pd2
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank  pd.obs.z   pd.obs.p
## BC001   668 43.71912     42.41961  0.5022911          26  2.587173 1.00000000
## BC002   587 40.94334     42.15943  0.5170896           1 -2.351799 0.03846154
##       runs
## BC001   25
## BC002   25
```

```
?ses.pd
```

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

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

   Answer 2a: H0: The observed PD is not different from the expectation from the null model, which means the community does not show any significant phylogenetic clustering or dispersions. H1: The observed PD is significantly different from the null expectation. ***Answer 2b***: Each null model uses different parameters for alpha diversity, and it definitely affects to observed ses.pd values.

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.

```
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.

```
ses.mpd <- ses.mpd(comm, phydist, null.model = "taxa.labels",
                   abundance.weighted = T, runs = 25)
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   0.324216036
## BC002   0.696932666
## BC003   0.743333599
## BC004  -0.167801089
## BC005   0.438429380
## BC010   0.629006791
## BC015   0.210838841
## BC016   0.650500924
## BC018   0.416510932
## BC020   0.405842470
## BC048   0.146804262
## BC049   0.202260904
## BC051  -0.378287103
## BC105  -0.152543516
## BC108   0.276930197
## BC262   0.175343067
## BCL01   0.181946261
## BCL03  -0.004361224
## HNF132 -0.026109069
## HNF133  0.162962878
## HNF134  0.337613615
## HNF144 -0.091684823
## HNF168  0.017511899
## HNF185  0.287512146
## HNF187  0.767417468
## HNF189  0.346983435
## HNF190  0.349618295
## HNF191 -0.032901801
## HNF216  0.724924111
## HNF217  0.297418007
## HNF221 -0.098652491
## HNF224  0.302539575
## HNF225  0.931572299
## HNF229  0.157710498
## HNF236  0.394871850
## HNF237  0.135644211
## HNF242  0.372174778
## HNF250  0.178465260
## HNF267  0.091142580
## HNF269  0.158417956
## HNF279  0.213098696
## YSF004 -0.437792790
## YSF117  0.646206375
## YSF295 -0.609006917
## YSF296  1.072802404
## YSF298  0.893695231
## YSF300  0.312717191
## YSF44   0.591427064
## YSF45   0.701852511
## YSF46   1.178923306
## YSF47   0.289093808
## YSF65   0.526742116
## YSF66   0.140991432
## YSF67   0.003734462
## YSF69   0.045088237
## YSF70  -0.121126899
## YSF71   0.731863656
## YSF74   0.897855233
```

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.

```
ses.mntd <- ses.mntd(comm, phydist, null.model = "taxa.labels",
                   abundance.weighted = T, runs = 25)
```

```
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   0.69859781
## BC002   1.55503453
## BC003   1.62575931
## BC004   1.58734920
## BC005   2.26746192
## BC010   0.47115173
## BC015   0.89011932
## BC016   1.59697339
## BC018   1.72106068
## BC020   0.98992685
## BC048   1.19186707
## BC049   1.88989836
## BC051   1.91626251
## BC105   0.86461058
## BC108   1.02981097
## BC262   0.95345194
## BCL01   1.13161786
## BCL03   0.53064584
## HNF132  1.44809256
## HNF133  1.91357987
## HNF134  1.78330267
## HNF144  0.83808443
## HNF168  0.35087591
## HNF185  1.55011684
## HNF187  0.44369123
## HNF189  0.65965065
## HNF190  1.22872570
## HNF191  0.99973202
## HNF216  0.06158389
## HNF217  0.01811770
## HNF221  0.23300155
## HNF224  1.07049887
## HNF225  0.23505602
## HNF229  1.28885495
## HNF236  0.30001317
## HNF237  0.48691414
## HNF242  1.50696431
## HNF250  1.20130028
## HNF267  0.73721842
## HNF269  0.90213560
## HNF279  0.70650971
## YSF004  0.49768847
## YSF117  1.39967252
## YSF295 -1.08604765
## YSF296  1.62846560
## YSF298  1.76855697
## YSF300  1.84388635
## YSF44   1.02322873
## YSF45   1.31484513
## YSF46   1.97284196
## YSF47   0.89858277
## YSF65   1.31378343
## YSF66   1.32487836
## YSF67   1.20515230
## YSF69   0.84199705
## YSF70   0.97238451
## YSF71   1.32529279
## YSF74   1.78056162
```

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***: It is calculated by the difference between the mean branch length of randomized null model and observed phylogeny. If the value is greater than 0, it means they are clustered, and if the value is lower than 0, it means they are overdispersed. ***Answer 3b***: It is similar to NRI, but it uses phylogenetically closest neighbor to calculates their distance. It also means if the value is greater than 0, it means they are clustered, and if the value is lower than 0, it means they are overdispersed. ***Answer 3c***: Most of the sample showed negative values which means they are phylogenetically overdispered. ***Answer 3d***: Now, these value gave more weight to abundant species compared to the previous results. Thus, I think these values focus more on phylogenetic relationship in dominant species, and its value is getting higher if the dominant species are closely related.

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.

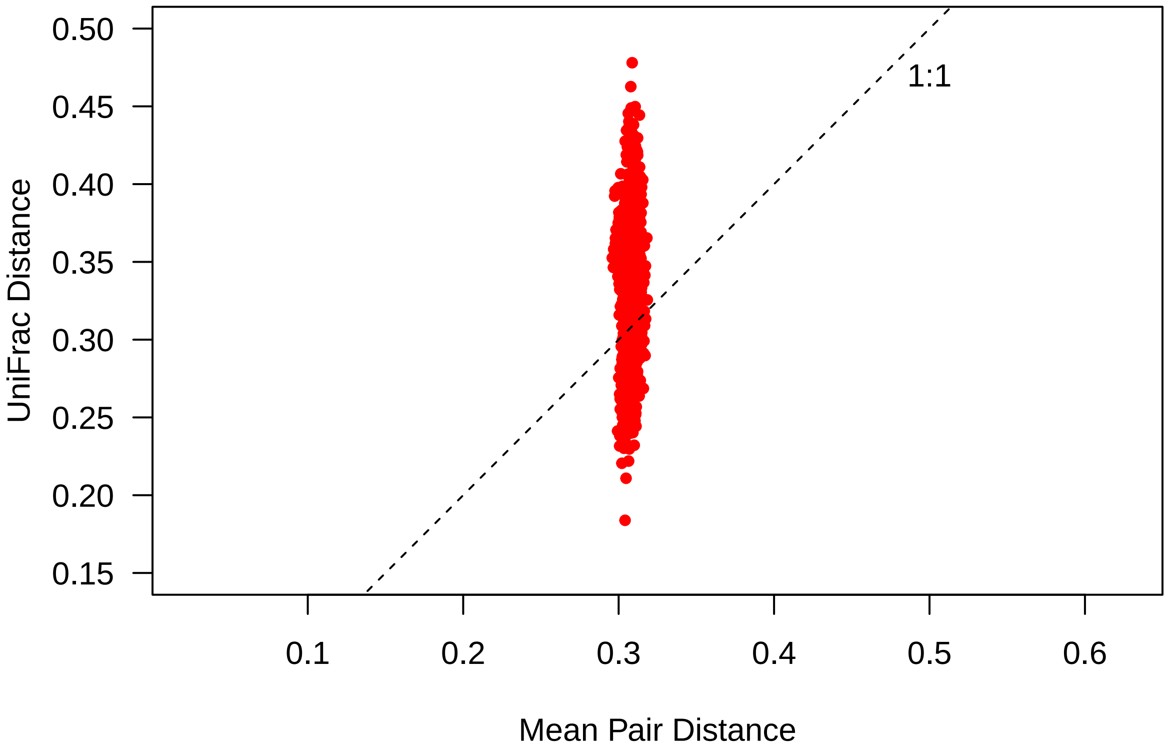
```
dist.mp <- comdist(comm, phydist)
```

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

```
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 Pair Distance is the average of the branch length of all species pairs in paired communities. However, UniFrac is calculated by dividing the sum of unshared branch by the sum of total branch length, which shows dissimilarity of two communities. ***Answer 4b***: The graph shows there is no relationship between these two values. Each community shows different level of dissimilarity (calculated by UniFrac), but their average branch lengths are similar (Mean Pair Distance). ***Answer 4c***: I think this is because MPD does not distinguish shared branch and unshared branch which tell us their evolutionary tracts. Thus, it cannot reflect the relativeness of species in each community and shows more similar values.

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
```

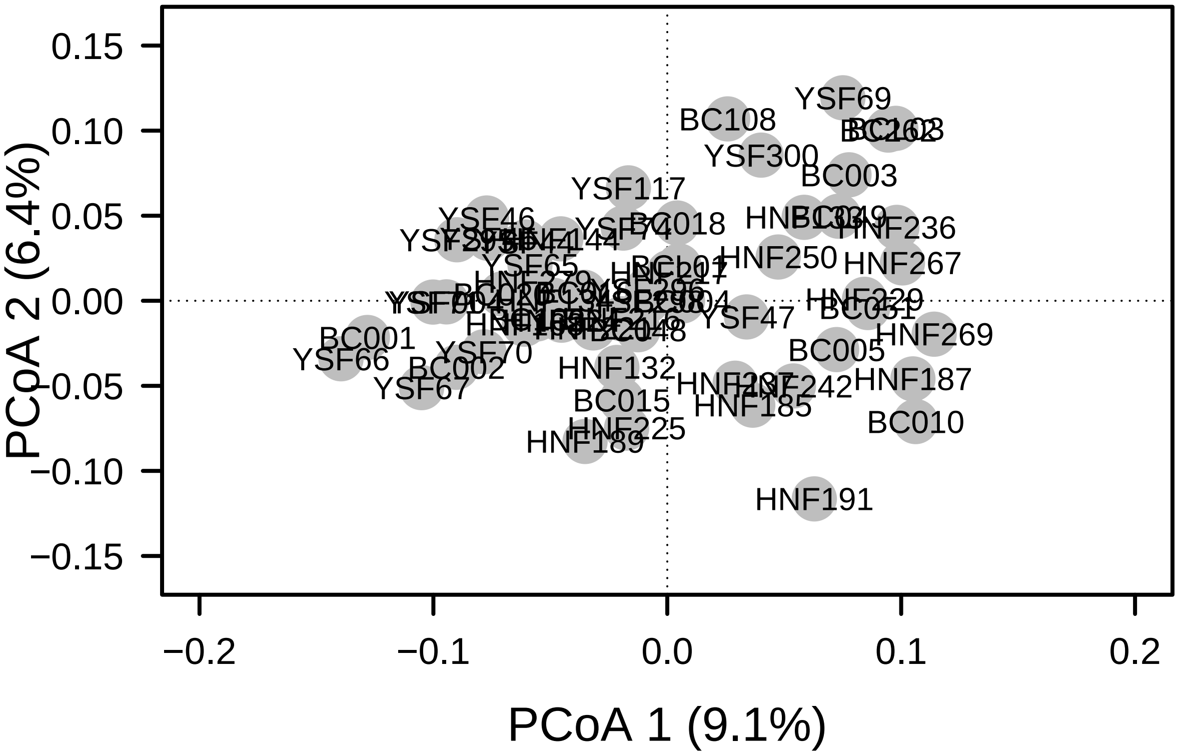
```
sum.eig <- sum(explainvar1, explainvar2, explainvar3)
sum.eig
```

```
## [1] 21.2
```

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

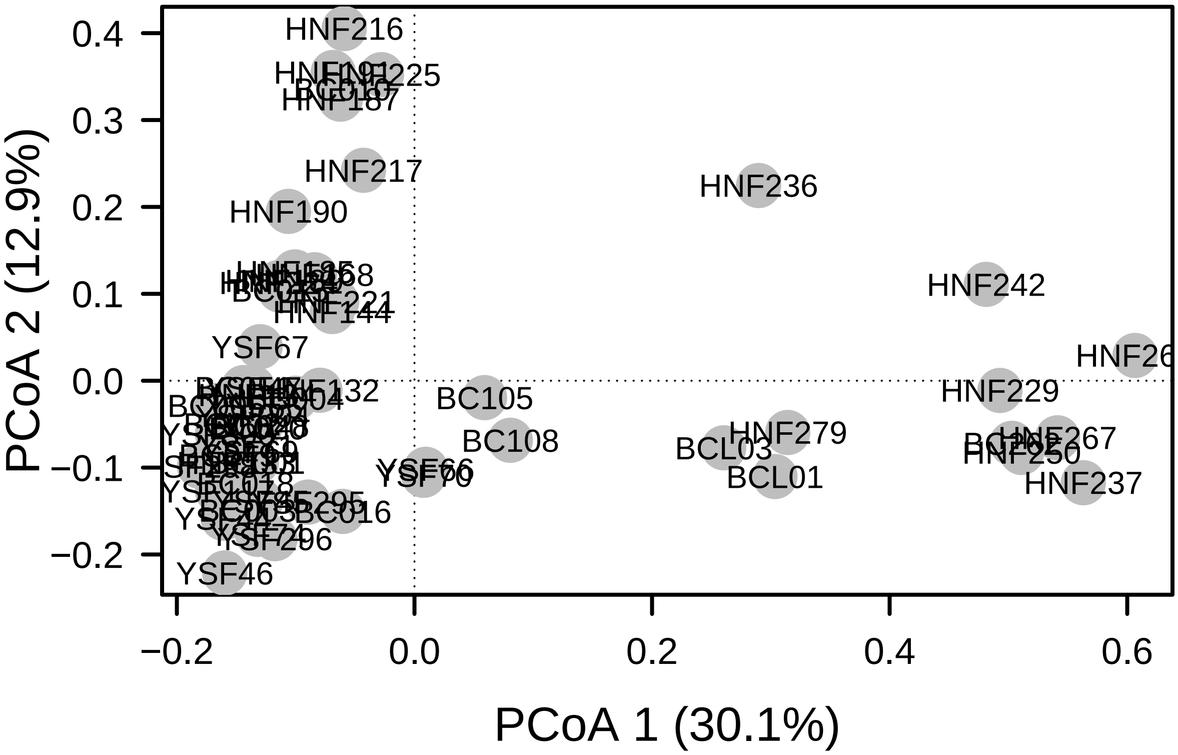
In the R code chunk below, do the following:
1. plot the PCoA results using either the R base package or the `ggplot` package,
2. include the appropriate axes,
3. add and label the points, and
4. customize the plot.

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

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 = F)

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)

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.

```
comm.db <- vegdist(comm, method = "bray", upper = TRUE, diag = TRUE)

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

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

sum.eig <- sum(explainvar1, explainvar2, explainvar3)

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

plot(comm.pcoa$points[, 1], comm.pcoa$points[ ,2],
     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 = F)

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)

points(comm.pcoa$points[ ,1], comm.pcoa$points[, 2],
       pch = 19, cex = 3, bg = "gray", col = "gray")

text(comm.pcoa$points[, 1], comm.pcoa$points[ , 2],
       labels = row.names(comm.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 PCoA plots by phylogenetic data and only taxonomic data showed completely different patterns. If we infer relationships only with taxonomic data, sites are clustered into roughly two way alongo x-axis. However, when we useing phylogenetic data, they do not diverege into distinct clusters. I think this highlights the importance of incorporating phylogenetic data when calculating beta diversity, and alerts us the dangers of inferring betadiversity with only taxonomic data.

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.

```
watershed <- env$Location
phylo.adonis <- adonis2(dist.uf ~ watershed, permutations = 999)
tax.adonis <- adonis2(
  vegdist(
    decostand(comm, method = "log"),
    method = "bray") ~ watershed,
  permutations = 999)
tax.adonis
```

```
## Permutation test for adonis under reduced model
## Permutation: free
## Number of permutations: 999
##
## adonis2(formula = vegdist(decostand(comm, method = "log"), method = "bray") ~ watershed, permutation
##          Df SumOfSqs      R2      F Pr(>F)
## Model     2   0.1822 0.05667 1.6521  0.003 **
## Residual 55   3.0328 0.94333
## Total    57   3.2150 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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()`).

```
envs <- env[, 5:19]
envs <- envs[, -which(names(envs) %in% c("TDS", "Salinity", "Cal_Volume"))]
env.dist <- vegdist(scale(envs), method = "euclid", na.rm = T)
```

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.

```
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: 0.134
##       Significance: 0.094
##
## Upper quantiles of permutations (null model):
##   90%   95% 97.5%   99%
## 0.125 0.167 0.200 0.231
## 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

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:reshape':
##
##     rename

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

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

```
##
##     count

## The following object is masked from 'package:nlme':
##
##     collapse

## The following object is masked from 'package:ape':
##
##     where

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
envs <- envs %>%
  mutate_all(~ifelse(is.na(.), mean(., na.rm = TRUE), .))  #  if I omitted NA value, dist.uf and envs h

ponds.dbrda <- vegan::dbrda(dist.uf ~ ., data = as.data.frame(scale(envs)))
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  0.11488 2.1480  0.327
## dbRDA2    1  0.08797 1.6448  0.858
## dbRDA3    1  0.07967 1.4897  0.955
## dbRDA4    1  0.07518 1.4057  0.966
## dbRDA5    1  0.06626 1.2389  0.993
## dbRDA6    1  0.05831 1.0904  0.999
## dbRDA7    1  0.05307 0.9924  1.000
## dbRDA8    1  0.04321 0.8080
## dbRDA9    1  0.04222 0.7894
## dbRDA10   1  0.03635 0.6797
## dbRDA11   1  0.02671 0.4995
## dbRDA12   1  0.02208 0.4128
## Residual 45  2.40671
```

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

```
##
## ***VECTORS
##
##              dbRDA1   dbRDA2     r2 Pr(>r)
## Elevation  0.03779 -0.99929 0.1145  0.038 *
## Diameter   0.71518  0.69894 0.2103  0.002 **
## Depth     -0.71504 -0.69908 0.2798  0.001 ***
## ORP        1.00000 -0.00310 0.1113  0.037 *
## Temp       0.22503  0.97435 0.2188  0.003 **
## SpC       -0.96137 -0.27525 0.3541  0.001 ***
## DO         0.44068  0.89766 0.1689  0.007 **
## pH        -0.99195 -0.12662 0.1017  0.057 .
## Color     -0.62871 -0.77764 0.0922  0.075 .
## chla      -0.97705 -0.21299 0.1353  0.028 *
## DOC        0.24706 -0.96900 0.1970  0.002 **
## DON       -0.31446 -0.94927 0.1413  0.018 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Permutation: free
## Number of permutations: 999
```

```
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

ponds_scores <- vegan::scores(ponds.dbrda, display = "sites")

par(mar = c(5, 5, 4, 4) + 0.1)
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, tyep = "n", cex.lab = 1.5,
     cex.axis = 1.2, axe = FALSE)
```

```
## Warning in plot.window(...): "tyep" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "tyep" is not a graphical parameter
```

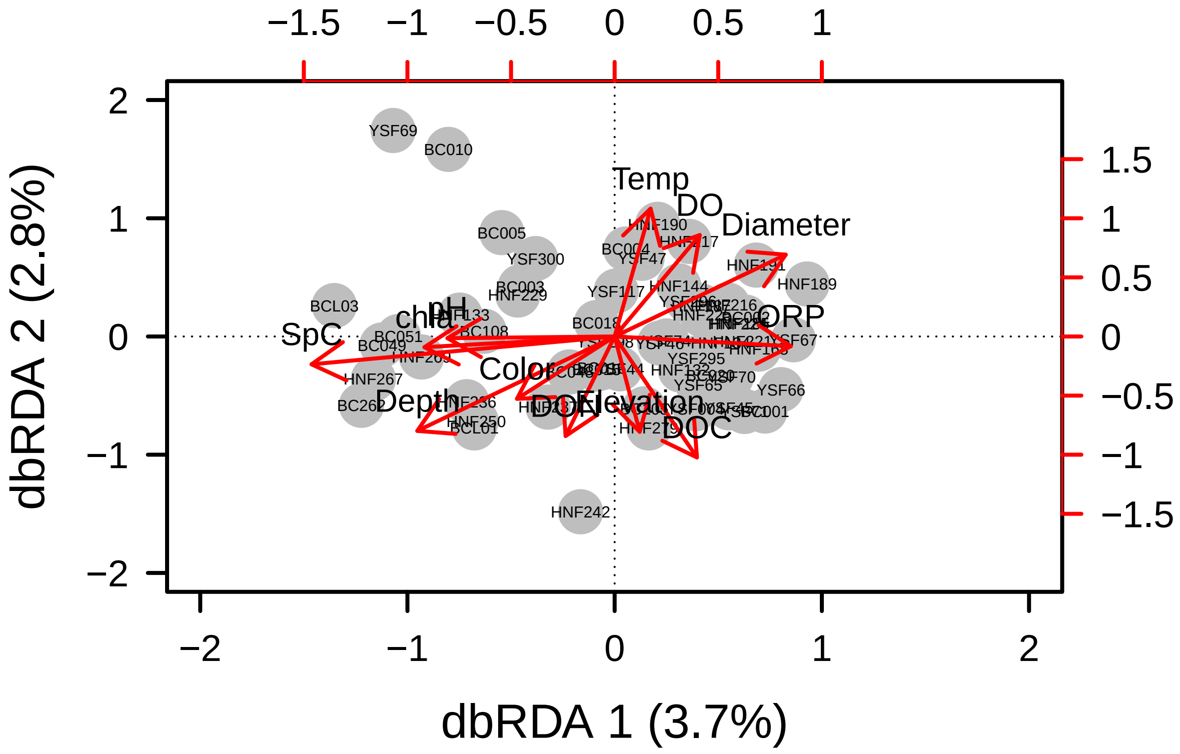
```
## Warning in title(...): "tyep" is not a graphical parameter
```

```
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)

wa_scores <- vegan::scores(ponds.dbrda, display = "sites")
points(wa_scores,
       pch = 19, cex = 3, col = "gray")

text(wa_scores,
     labels = row.names(wa_scores), cex = 0.5)


vectors <- vegan::scores(ponds.dbrda, display = "bp")
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,
     label = row.names(vectors))
axis(side = 3, lwd.ticks = 2, cex.axis = 1.2, las = 1, col = "red", ld = 2,2,
     at = pretty(range(vectors[, 1]) * 2), labels = pretty(range(vectors[,1]) * 2))
```

```
## Warning in axis(side = 3, lwd.ticks = 2, cex.axis = 1.2, las = 1, col = "red",
## : "ld" is not a graphical parameter
```

```r
axis(side = 4, lwd.ticks = 2, cex.axis = 1.2, las = 1, col = "red", ld = 2,2,
     at = pretty(range(vectors[, 2]) * 2), labels = pretty(range(vectors[,2]) * 2))
```

```
## Warning in axis(side = 4, lwd.ticks = 2, cex.axis = 1.2, las = 1, col = "red",
## : "ld" is not a graphical parameter
```



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 bacterial communities in the ponds in Indiana are phylogenetically diverse based on these tests above. Mantel test showed us quite low correlation coefficient (0.134), and not higly significant p-value (0.073), but the distance-based RDA showed some environmental data such as mainly Depth, Temp, SpC have significant effects on their phylogenetic distances.

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***: In my study system, phylogenetic information is very important. My research is mainly focused on how the plastic novel traits are evolved in specific beetle clade. However, even in these specific clade, some of beetles do not possess any horn. These beetles are assumed they have secondarily lost their horns rather than never having developed them. Thus, to distinguish these hornless beetles from the beetles which haven't possessed horns in their evolutionaty history, I should use phylogenetic information from genomic analysis.