12.Phylogenetic Diversity - Communities

Dustin Brewer; Z620: Quantitative Biodiversity, Indiana University

24 February, 2019

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 '8.BetaDiversity' 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 *12.PhyloCom_Worksheet.Rmd* and the PDF output of `Knitr` (*12.PhyloCom_Worksheet.pdf*).

1) SETUP

Typically, the first thing you will do in either an R script or an RMarkdown file is setup your environment. This includes things such as setting the working directory and loading any packages that you will need.

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

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

```
## [1] "C:/Users/dusti/GitHub/QB2019_Brewer/2.Worksheets/12.PhyloCom"
```

```
 library("picante", lib.loc="~/R/win-library/3.5")
```

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

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

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

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

```
## This is vegan 2.5-3
```

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

```
library("ape", lib.loc="~/R/win-library/3.5")
 library("seqinr", lib.loc="~/R/win-library/3.5")
```

```
##
## 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
```

```
library("vegan", lib.loc="~/R/win-library/3.5")
library("fossil", lib.loc="~/R/win-library/3.5")
```

```
## Loading required package: sp
```

```
## Loading required package: maps
```

```
## Loading required package: shapefiles
```

```
## Loading required package: foreign
```

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

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

```
library("reshape", lib.loc="~/R/win-library/3.5")
library("simba", lib.loc="~/R/win-library/3.5")
```

```
## This is simba 0.3-5

##
## Attaching package: 'simba'

## The following object is masked from 'package:picante':
##
##     mpd

## The following object is masked from 'package:stats':
##
##     mad
```

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

2) DESCRIPTION OF DATA

need to discuss data set from spatial ecology!

In 2013 we sampled > 50 forested ponds in Brown County State Park, Yellowwood 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)

str(env)
```

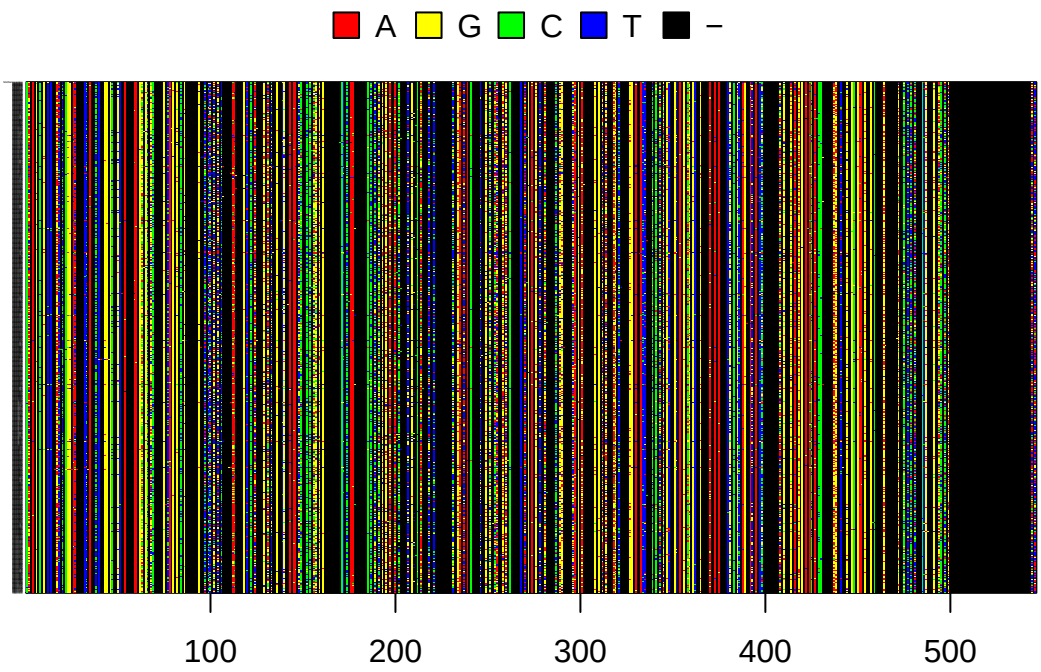
```
## 'data.frame':    52 obs. of  21 variables:
##  $ Location  : Factor w/ 3 levels "BCSP","HNF","YSF": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Sample_ID : Factor w/ 58 levels "BC001","BC002",..: 1 2 3 4 5 7 8 9 10 11 ...
##  $ lat       : num  39.1 39.2 39.2 39.1 39.1 ...
##  $ long      : num  -86.2 -86.2 -86.2 -86.2 -86.2 ...
##  $ Elevation : int  226 276 258 249 241 126 238 244 248 246 ...
##  $ Diameter  : num  11.5 15.5 15.5 13.5 7 12 11.5 11 10 16 ...
##  $ Depth     : num  1 0.5 1.3 0.75 0.75 1.2 2 0.75 1.5 1.5 ...
##  $ Cal_Volume: num  398.2 974.9 974.9 644.1 89.8 ...
##  $ ORP       : num  198 144.5 107.4 84.7 91.8 ...
##  $ Temp      : num  19.9 21.6 21.1 22.6 21.7 ...
##  $ SpC       : int  31 48 61 41 46 61 63 38 60 48 ...
##  $ DO        : num  2.3 1.89 0.81 1.8 2.75 0.99 2.04 1.53 2.13 0.27 ...
##  $ TDS       : num  0.02 0.031 0.04 0.026 0.03 0.039 0.041 0.025 0.039 0.031 ...
##  $ Salinity  : num  0.01 0.02 0.03 0.02 0.02 0.03 0.03 0.02 0.03 0.02 ...
##  $ pH        : num  6.11 6.39 6.31 6.25 6.17 6.31 6.5 6.31 6.6 6.34 ...
##  $ Color     : num  0.0859 0.0362 0.0917 0.0436 0.057 ...
##  $ chla      : num  86.3 93.4 131.3 88.2 110.6 ...
##  $ DOC       : num  8.75 10.01 6.82 6.97 7.84 ...
##  $ DON       : num  0.604 0.383 1.282 0.22 0.293 ...
##  $ canopy    : num  0.152 0.323 0.208 0.506 0.195 ...
##  $ TP        : num  598.1 123.1 291.4 80.1 81.3 ...
##  - attr(*, "na.action")= 'omit' Named int  34 54 55 56 57 58
##   ..- attr(*, "names")= chr  "34" "54" "55" "56" ...
```

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

#Select DNA, disregard cDNA
comm <- comm[grep("*-DNA", rownames(comm)),]

#Perform replacement of all matches
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]

#From the source code, take a look at taxonomic information associated with OTU data
tax <- read.tax(taxonomy = "./data/INPonds.final.rdp.1.cons.taxonomy")
```

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

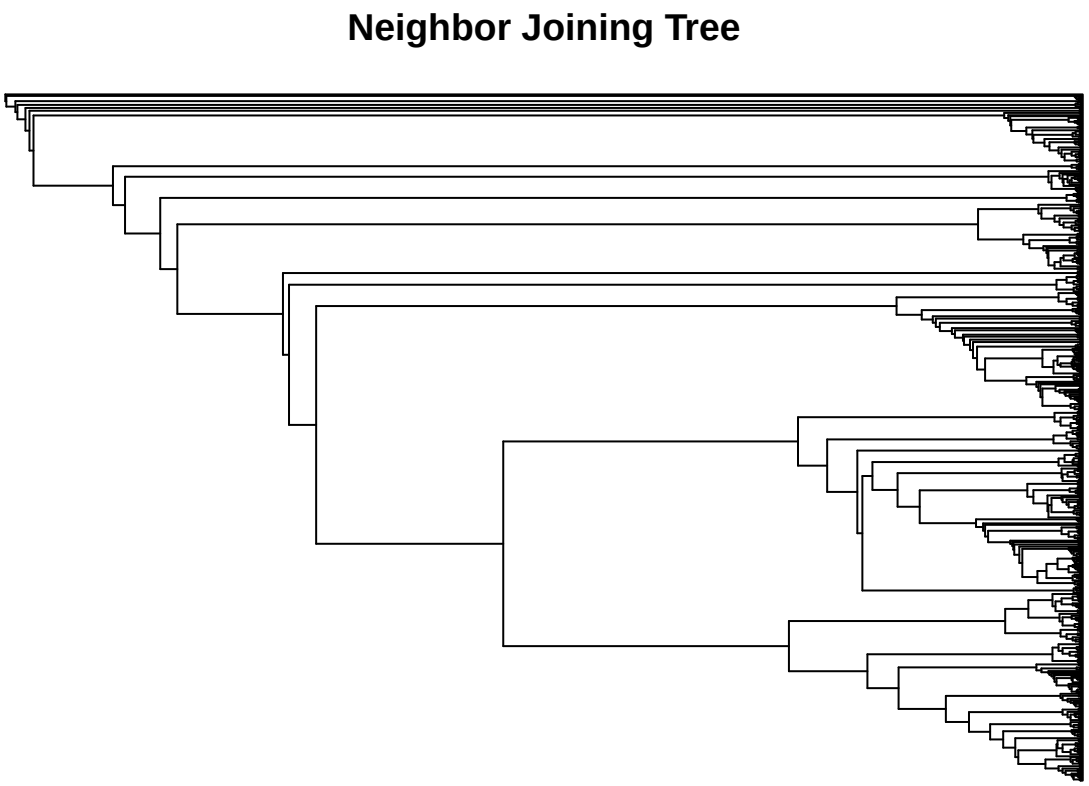
4. convert both FASTA files into the DNAbin format and combine using `rbind()`,
5. visualize the sequence alignment,
6. using the alignment (with outgroup), pick a DNA substitution model, and create a phylogenetic distance matrix,
7. using the distance matrix above, make a neighbor joining tree,
8. remove any tips (OTUs) that are not in the community data set,
9. plot the rooted tree.

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

#rename OTUs
ponds.cons$nam <- gsub("\\|.*$", "", gsub("^.*?\t", "", ponds.cons$nam))

#import the outgroup (archaea)
outgroup <- read.alignment(file = "./data/methanosarcina.fasta", format = "fasta")

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

#have a look at the alignment
image.DNAbin(DNAbin, show.labels=T, cex.lab = 0.05, las = 1)
```




```
#make a distance matrix, using 'jukes cantor' model
seq.dist.jc <- dist.dna(DNAbin, model = "JC", pairwise.deletion = FALSE)
```

```r
#Use the jukes cantor object and make a neighbor-joining tree
phy.all <- bionj(seq.dist.jc)

#Drop tips associated with zero abundance taxa
phy <- drop.tip(phy.all, phy.all$tip.label[!phy.all$tip.label %in%
              c(colnames(comm), "Methanosarcina")])

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

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

#Plot the rooted tree
par(mar = c(1, 1, 2, 1) + 0.1)
plot.phylo(phy, main = "Neighbor Joining Tree", "phylogram", show.tip.label = FALSE,
           use.edge.length = FALSE, direction = "right", cex = 0.6, label.offset = 1)
```



4) PHYLOGENETIC ALPHA DIVERSITY

A. Faith's Phylogenetic Diversity (PD)

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

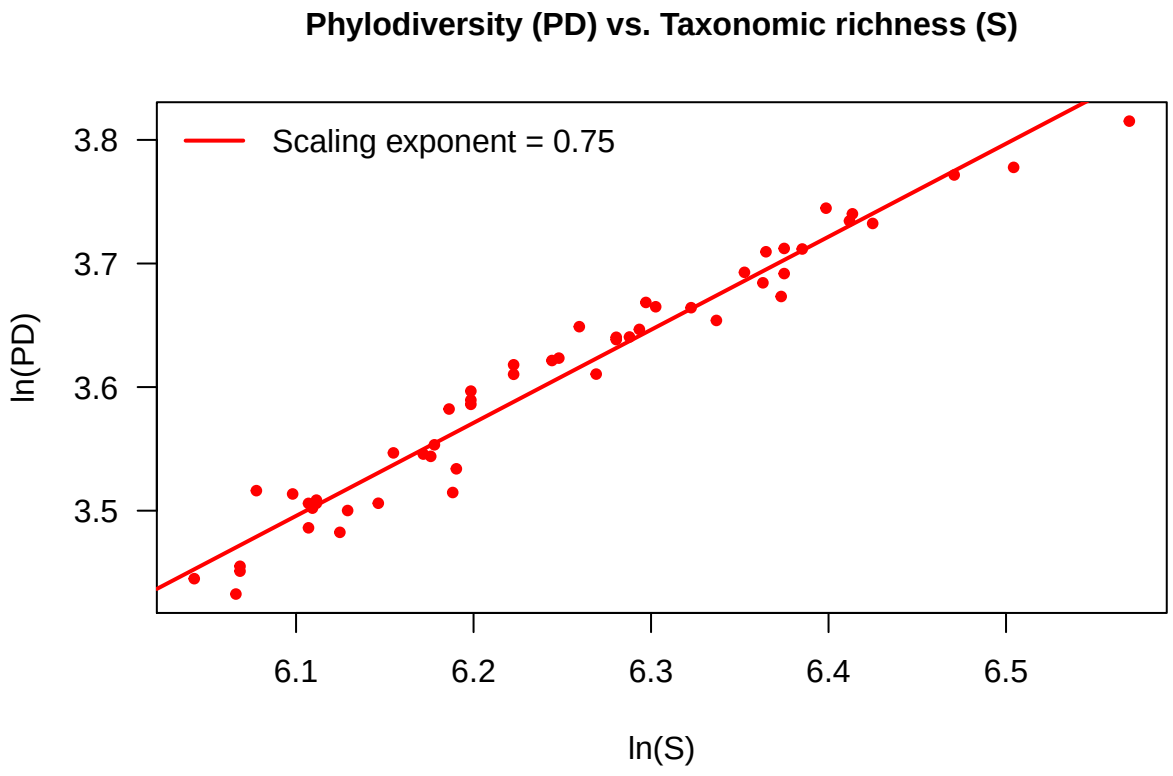
```r
#Calculate Faith's D and species richness
pd <- pd(comm, phy, include.root = FALSE)
pd
```

```
##              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.

```
#Make a biplot of species richness and phylogenetic diversity, using a natural logarithm on both axes s
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 the 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***: PD does not weight the branches of common species more than the branches of rare species, just like a measure of species richness, so it makes sense that the measures will be related. As there are more species at a site, there will be more branch length measured by PD. ***Answer 1b***: There is positive, non-linear relationship between PD and S. As species richness increases, so does phylogenetic diversity (but PD does not increase as fast). ***Answer 1c***: If there were many closely related species in an area, it seems that species richness and phylogenetic diversity there would diverge.
> ***Answer 1d***: Because the scaling exponent is less then 1, I take that to mean that Phylogenetic Diversity increases at a slower rate than species richness.

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.

```
#randomize phylogenetic diversity to create a null model that our data can be compared to, using the pi

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

#do same, but with null model of 'taxa labels'
ses.pd2 <- ses.pd(comm[1:2,], phy, null.model = "taxa.labels", runs = 25, include.root = FALSE)
ses.pd2
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank   pd.obs.z
## BC001   668 43.71912     44.09587  0.6291881           8 -0.5987763
## BC002   587 40.94334     40.15025  0.8639968          22  0.9179286
##        pd.obs.p runs
## BC001 0.3076923   25
## BC002 0.8461538   25
```

```
#now with 'frequency'
ses.pd3 <- ses.pd(comm[1:2,], phy, null.model = "frequency", runs = 25, include.root = FALSE)
ses.pd3
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank  pd.obs.z
## BC001   668 43.71912     42.35170  0.5306079          26  2.577095
## BC002   587 40.94334     42.20036  0.5507459           1 -2.282392
##         pd.obs.p runs
## BC001 1.00000000   25
## BC002 0.03846154   25
```

```
?ses.pd()
```

```
## starting httpd help server ... done
```

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

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

Answer 2a: The null hypothesis being tested when using ses.pd is that there is no difference between the randomized communities generated by the null model and the one that you sampled/are interested in. The alternative hypothesis is that there is a difference.
Answer 2b: The 'richness' null model was not significantly different than BC001 or BC002, nor was the 'taxa.labels' null model. The 'frequency' null model was not significantly different than BC001, but it was signficantly different than BC002. I think that BC002 was significantly different than the 'frequency' null model because in that model, sample species richness is not maintained and BC002 has an above average species richness.

B. Phylogenetic Dispersion Within a Sample

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

i. Phylogenetic Resemblance Matrix

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

```
#Create a phylogenetic distance matrix (which is based on distance between tax on the tree)
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 Net Relatedness Index via randomizations
ses.mpd <- ses.mpd(comm, phydist, null.model = "taxa.labels",
                   abundance.weighted = FALSE, 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  -2.3723893
## BC002  -5.8262520
## BC003  -2.3466152
## BC004  -2.9008320
## BC005  -3.8788327
## BC010  -3.4786466
## BC015  -2.2839453
## BC016  -1.8029222
## BC018  -2.1515191
## BC020  -1.8306328
## BC048  -2.1101893
## BC049  -0.6936663
```

```
## BC051  -3.3351943
## BC105  -3.1088701
## BC108  -2.3005890
## BC262  -1.5664113
## BCL01  -3.4579726
## BCL03  -1.1979707
## HNF132 -5.2120662
## HNF133 -2.9286114
## HNF134 -2.8988167
## HNF144 -5.6483454
## HNF168 -2.2762549
## HNF185 -3.9604820
## HNF187  0.7650403
## HNF216 -2.0149028
## HNF217 -2.3927481
## HNF221 -2.2613453
## HNF224 -3.5646626
## HNF225 -2.2673280
## HNF229 -0.9795959
## HNF242 -1.4931766
## HNF250 -2.7477760
## HNF267 -1.2801997
## HNF269 -1.9646599
## YSF004 -3.3933699
## YSF117 -1.9250585
## YSF295 -2.4917012
## YSF296 -2.5792972
## YSF298 -3.7285921
## YSF300 -3.1584466
## YSF44  -1.5178619
## YSF45  -3.2400811
## YSF46  -1.8240643
## YSF47  -2.3756315
## YSF65  -0.2575981
## YSF66  -1.5911894
## YSF67  -2.8895509
## YSF69  -1.9119604
## YSF70  -2.2064332
## YSF71  -1.5817372
## YSF74  -2.7428188
```

```
#Do the same, but with abundance.weighted = TRUE
ses.mpd2 <- ses.mpd(comm, phydist, null.model = "taxa.labels",
                    abundance.weighted = TRUE, runs = 25)

NRI2 <- as.matrix(-1 * ((ses.mpd2[,2] - ses.mpd2[,3]) / ses.mpd2[,4]))
rownames(NRI) <- row.names(ses.mpd)
colnames(NRI) <- "NRI2"

NRI2
```

```
##               [,1]
##  [1,]  0.107045992
##  [2,]  0.386002907
```

```
##  [3,]  0.866860397
##  [4,] -0.653010789
##  [5,]  0.322618863
##  [6,]  0.034927573
##  [7,] -0.273062378
##  [8,]  0.196293747
##  [9,]  0.104457829
## [10,]  0.130190406
## [11,] -0.578998013
## [12,] -0.144238194
## [13,] -0.960480411
## [14,] -1.136956717
## [15,] -0.212116090
## [16,] -0.326143523
## [17,] -0.501683371
## [18,] -0.792438485
## [19,] -0.837148403
## [20,] -0.138413978
## [21,]  0.062896470
## [22,] -0.600439989
## [23,] -0.696141601
## [24,]  0.130232064
## [25,]  0.339215894
## [26,]  0.400834051
## [27,] -0.377751043
## [28,] -0.431649202
## [29,] -0.069167850
## [30,]  0.436604047
## [31,] -0.419534198
## [32,] -0.006396618
## [33,] -0.337811146
## [34,] -0.461606261
## [35,] -0.294225435
## [36,] -0.501179075
## [37,]  0.644720671
## [38,] -1.495022306
## [39,]  0.841386741
## [40,]  0.941866070
## [41,]  0.225862431
## [42,]  0.357052410
## [43,]  0.609250837
## [44,]  1.577490898
## [45,] -0.044744184
## [46,]  0.377071959
## [47,] -1.033393794
## [48,] -0.413965815
## [49,] -0.107064012
## [50,] -0.791067633
## [51,]  0.823131319
## [52,]  0.985481113
```

iii. Nearest Taxon Index (NTI)

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

```r
# Estimate Standardized Effect Size of NRI via Randomization
ses.mntd <- ses.mntd(comm, phydist, null.model = "taxa.labels",
                     abundance.weighted = FALSE, runs = 25)
# Calculate NTI
NTI <- as.matrix(-1 * ((ses.mntd[,2] - ses.mntd[,3]) / ses.mntd[,4]))
rownames(NTI) <- row.names(ses.mntd)
colnames(NTI) <- "NTI"

NTI
```

```
##                NTI
## BC001   0.37785307
## BC002  -1.40490513
## BC003  -0.54587812
## BC004  -1.71843062
## BC005  -1.96384527
## BC010  -1.21255008
## BC015  -1.35179878
## BC016  -0.18384969
## BC018  -0.60970129
## BC020  -0.64636133
## BC048  -1.42060424
## BC049  -0.37743049
## BC051  -1.20612417
## BC105  -1.26091406
## BC108  -0.95801007
## BC262   0.14481123
## BCL01  -1.79910270
## BCL03   0.29948974
## HNF132 -0.91758192
## HNF133 -1.23535180
## HNF134 -0.94979862
## HNF144 -1.86558051
## HNF168 -1.63863471
## HNF185 -0.98219590
## HNF187  0.35444033
## HNF216 -1.95361580
## HNF217 -1.47222826
## HNF221 -0.97352227
## HNF224 -2.17207623
## HNF225 -1.91396847
## HNF229  0.13129033
## HNF242 -1.03983441
## HNF250 -0.59555901
## HNF267  1.21238035
## HNF269  0.47742008
## YSF004 -1.45587667
## YSF117 -1.46774182
## YSF295 -1.04884788
## YSF296  0.15748189
## YSF298 -0.63793926
## YSF300 -1.29549418
## YSF44  -1.10778505
## YSF45  -1.00948737
```

```
## YSF46  -0.50195275
## YSF47  -0.28891888
## YSF65   0.41660814
## YSF66   0.93629216
## YSF67  -1.12241019
## YSF69  -0.06889161
## YSF70   0.15973602
## YSF71  -0.42486887
## YSF74  -1.05902329
```

```
#Do same, but with abundance.weighted as TRUE
ses.mntd2 <- ses.mntd(comm, phydist, null.model = "taxa.labels",
                      abundance.weighted = TRUE, runs = 25)
# Calculate NTI
NTI2 <- as.matrix(-1 * ((ses.mntd2[,2] - ses.mntd2[,3]) / ses.mntd2[,4]))
rownames(NTI) <- row.names(ses.mntd2)
colnames(NTI) <- "NTI2"

NTI2
```

```
##             [,1]
##  [1,]  0.9469980
##  [2,]  1.8443417
##  [3,]  1.5184895
##  [4,]  1.1327312
##  [5,]  2.0855659
##  [6,]  0.4925556
##  [7,]  1.2793376
##  [8,]  3.2295905
##  [9,]  2.2473722
## [10,]  1.5397157
## [11,]  1.1323440
## [12,]  1.6216115
## [13,]  2.0787994
## [14,]  1.2404296
## [15,]  1.0097678
## [16,]  1.0132653
## [17,]  1.2692713
## [18,]  0.8217138
## [19,]  1.1565925
## [20,]  1.6347691
## [21,]  1.7905206
## [22,]  0.8801955
## [23,]  0.5143108
## [24,]  1.5176135
## [25,]  0.4671737
## [26,]  0.2937807
## [27,]  0.5471485
## [28,]  0.3261341
## [29,]  1.0753738
## [30,]  0.5331342
## [31,]  1.2672706
## [32,]  1.5091813
## [33,]  1.2213619
```

```
## [34,]  0.7608128
## [35,]  0.9244131
## [36,]  0.2091209
## [37,]  2.5827165
## [38,] -1.2794640
## [39,]  2.4318434
## [40,]  3.0973046
## [41,]  1.4070907
## [42,]  1.3725436
## [43,]  1.2202189
## [44,]  2.1121394
## [45,]  0.8367027
## [46,]  1.3450964
## [47,]  1.7317920
## [48,]  1.0727230
## [49,]  0.8923596
## [50,]  1.0446775
## [51,]  1.3590484
## [52,]  1.9227794
```

Question 3:

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

Answer 3a: When calculating NRI, I am trying to determine to what extent my sample is clustered or overdispersed based on a comparison with a null model. This comparison is made based upon pairwise comparisons between all taxa in a sample. ***Answer 3b***: The NTI is similar to the NRI, but is based on nearest neighbor distance, rather than by making pairwise comparisons with all taxa in the tree. Thus, this measure focuses more on clustering that occurs near the tips. ***Answer 3c***: Both the NTI and NRI values suggest that the bacterial communities tend to be overdispersed. ***Answer 3d***: Calculating NRI and NTI using abundance data rather than incidence data changes the interpretation. Rather than tending to indicate overdispersion, both NRI and NTI tend to indicate clustering for OTUs at sites.

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.

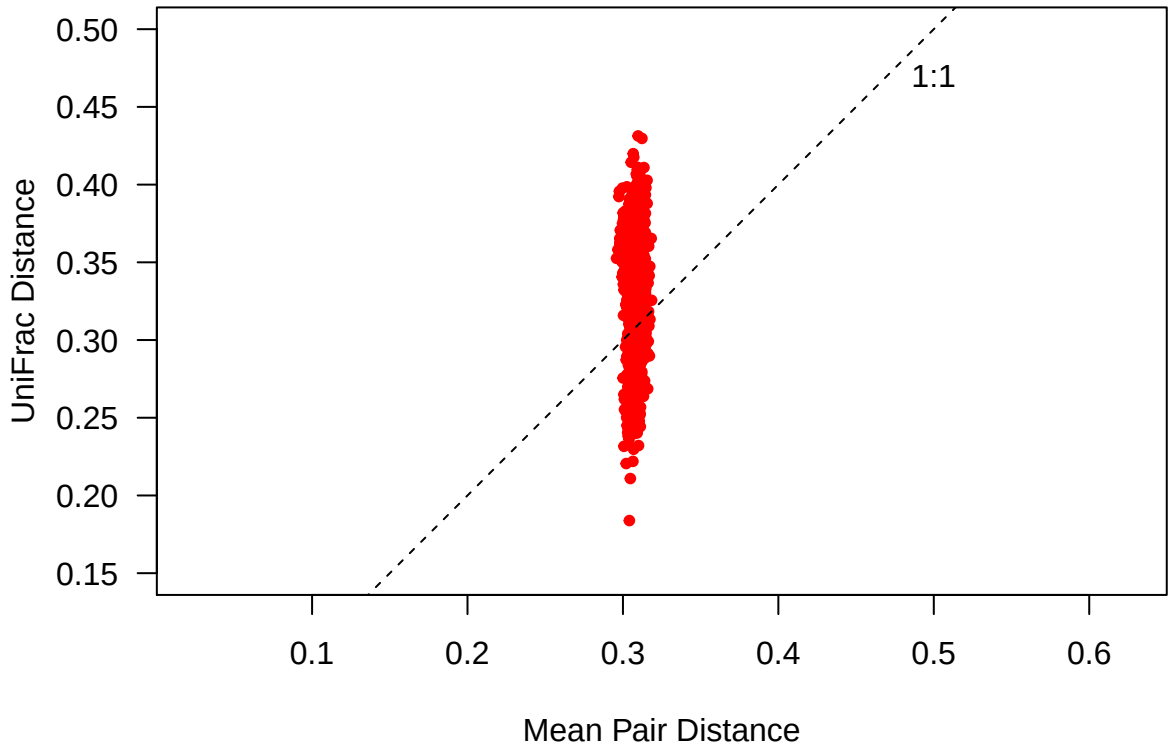
```
#mean pair distance
dist.mp <- comdist(comm, phydist)
```

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

```r
#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.

```r
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 compares the average distance between taxa from two different samples. Unifrac Distance is different than Mean Pair Distance because it compares two samples by computing the proportion of the two trees that are shared with respect to branch topology.
Answer 4b: There is much more variability in Unifrac distance (~0.2 to 0.45) than in Mean Pair

Distance (~0.29 to 0.31). ***Answer 4c***: MPD might show less variation than Unifrac because even if both samples have similar mean distances between pairs of taxa, the phylogenetic trees might vary more with respect to branch lengths, thus causing UniFrac to be more variable.

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.

```
#PCoA based on Unifrac distances
pond.pcoa <- cmdscale(dist.uf, eig = T, k = 3)
pond.pcoa
```

```
## $points
##                [,1]         [,2]         [,3]
## BC001   0.132257551  0.032172014  0.029207578
## BC002   0.089542241  0.019941419  0.000668870
## BC003  -0.090950296 -0.114684057  0.015851099
## BC004  -0.010157519 -0.026778624 -0.090422167
## BC005  -0.068853424  0.009748627 -0.086086860
## BC010  -0.099681511  0.088023141 -0.071630315
## BC015   0.026226866  0.052016768 -0.029688116
## BC016   0.035912745  0.053874421  0.078036314
## BC018  -0.012685648 -0.017974633  0.046959728
## BC020   0.059775546  0.034190866  0.072232695
## BC048   0.008424832  0.026237216 -0.040110169
## BC049  -0.082391835  0.001159213  0.126252559
## BC051  -0.104045365  0.116655423  0.054362863
## BC105   0.048107527  0.052867120 -0.047899814
## BC108  -0.053379299 -0.026180169  0.109744055
## BC262  -0.119102039  0.042768821  0.020031730
## BCL01  -0.016401517  0.053542143 -0.006932119
## BCL03  -0.117725283  0.004033198  0.079411984
## HNF132  0.029706858  0.038605195 -0.046870312
## HNF133 -0.064782803 -0.108722032 -0.035212791
## HNF134  0.046873890 -0.010411845 -0.032912820
## HNF144  0.036144361 -0.076026015 -0.024482416
## HNF168  0.059793430 -0.032327528 -0.030321207
## HNF185 -0.012241296 -0.053756578 -0.088440270
## HNF187 -0.101321166  0.023626499 -0.013329296
## HNF216  0.020506280 -0.035002793 -0.032623487
## HNF217 -0.010839841 -0.001516001  0.023542247
## HNF221  0.031636234 -0.013615059 -0.056853099
## HNF224  0.043164843  0.022043412 -0.019848353
## HNF225  0.027736803  0.058364539 -0.034501610
## HNF229 -0.089263880  0.029431563 -0.020558971
## HNF242 -0.038385503  0.023203854 -0.058136175
## HNF250 -0.051584900  0.047709696  0.024080372
```

```
## HNF267 -0.101334020  0.062550280 -0.004758080
## HNF269 -0.112687520  0.105689733 -0.070016496
## YSF004  0.089191232  0.016996137 -0.002855156
## YSF117  0.004548226 -0.136042378  0.029218678
## YSF295  0.076546417 -0.020238003  0.047675558
## YSF296  0.010637051 -0.087904465 -0.049682247
## YSF298  0.009587482 -0.041603321 -0.055010477
## YSF300 -0.059920769 -0.061666922  0.042332267
## YSF44   0.048802134  0.036117432  0.073918934
## YSF45   0.067100543 -0.043336598  0.021518224
## YSF46   0.061298117 -0.006489543  0.075374989
## YSF47  -0.027667691 -0.096169753 -0.076271125
## YSF65   0.050862102 -0.011610394  0.007377772
## YSF66   0.141000960  0.030113922  0.031201872
## YSF67   0.105851093  0.072168072 -0.022663931
## YSF69  -0.107422620 -0.075083748  0.086892539
## YSF70   0.080889527  0.012597114  0.036378896
## YSF71   0.099308682  0.008377594  0.003904187
## YSF74   0.011392170 -0.077684973  0.011941873
##
## $eig
##  [1] 2.580360e-01 1.635889e-01 1.460151e-01 1.286988e-01 1.224414e-01
##  [6] 1.152644e-01 9.886485e-02 9.371001e-02 9.128046e-02 8.380498e-02
## [11] 7.860618e-02 7.333826e-02 6.612530e-02 6.490344e-02 6.456088e-02
## [16] 6.103984e-02 6.008210e-02 5.474542e-02 5.268350e-02 5.155039e-02
## [21] 4.911369e-02 4.672301e-02 4.347160e-02 4.172799e-02 4.120495e-02
## [26] 3.994070e-02 3.633215e-02 3.435569e-02 3.258808e-02 3.209965e-02
## [31] 3.009472e-02 2.975248e-02 2.890885e-02 2.686516e-02 2.565431e-02
## [36] 2.420778e-02 2.306850e-02 2.120994e-02 2.038193e-02 1.925392e-02
## [41] 1.871748e-02 1.687059e-02 1.609765e-02 1.530220e-02 1.347258e-02
## [46] 1.214111e-02 1.150756e-02 9.714165e-03 8.214255e-03 5.242941e-03
## [51] 2.633958e-03 6.074989e-17
##
## $x
## NULL
##
## $ac
## [1] 0
##
## $GOF
## [1] 0.2097546 0.2097546
```

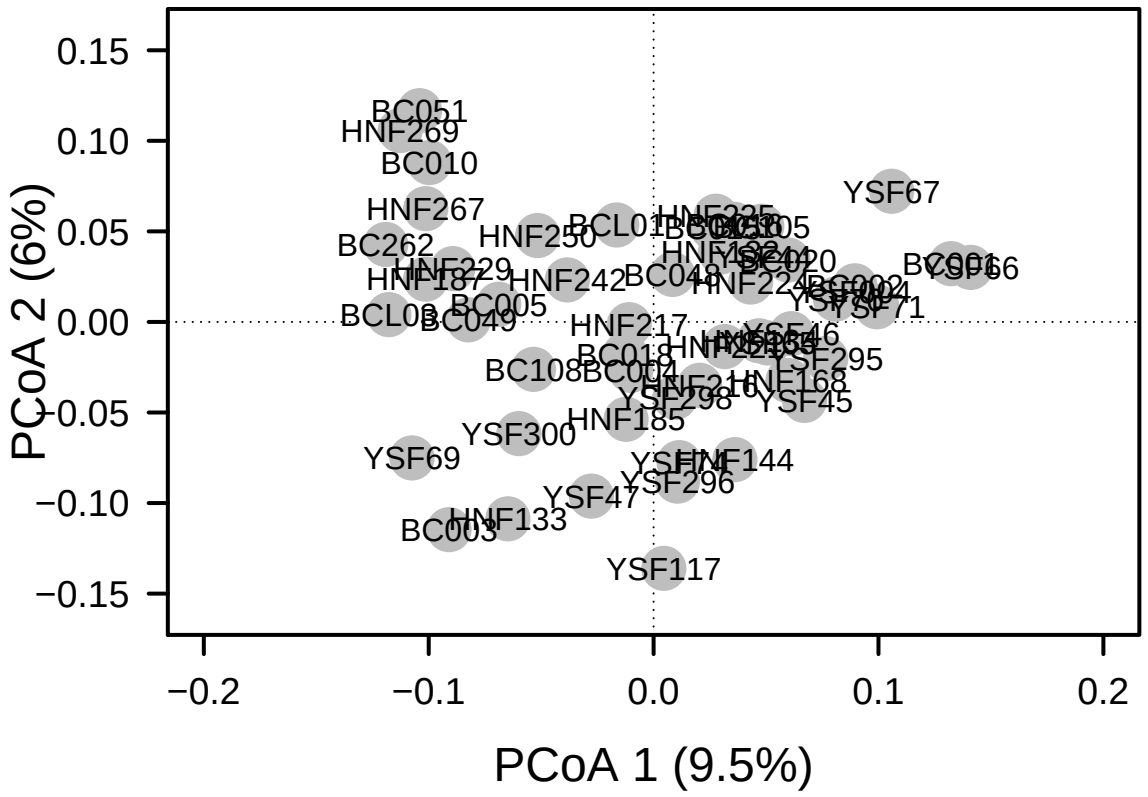
```
?cmdscale()

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

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

In the R code chunk below, do the following:
1. plot the PCoA results using either the R base package or the `ggplot` package,

2. include the appropriate axes,
3. add and label the points, and
4. customize the plot.

```
# Define Plot Parameters
par(mar = c(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.

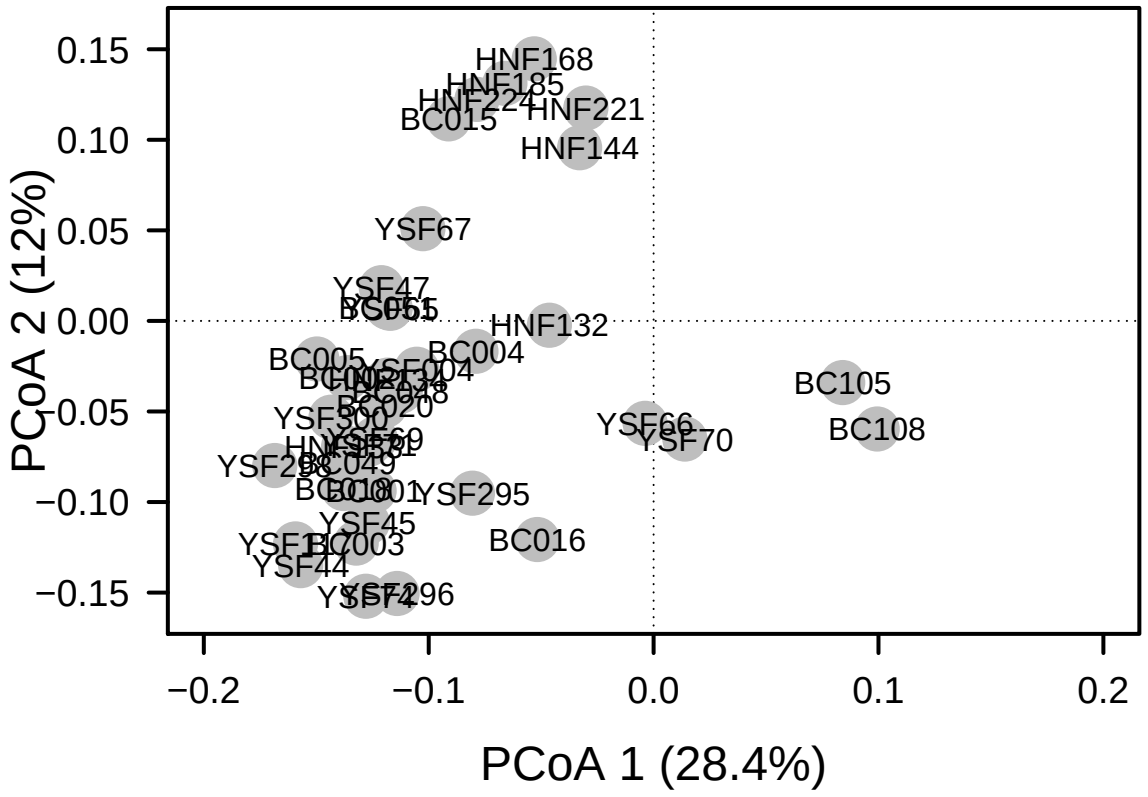
```
bc <- vegdist(comm, method = "bray", binary = FALSE, diag = FALSE, upper = FALSE, na.rm = TRUE)

#PCoA based on bray distances
tax.pcoa <- cmdscale(bc, eig = T, k = 3)
tax.pcoa
```

```
## $points
##                [,1]         [,2]          [,3]
## BC001  -0.124278800 -0.093784403 -1.166177e-01
## BC002  -0.136147625 -0.031305839  1.277629e-02
## BC003  -0.132253740 -0.122683677  2.247354e-01
## BC004  -0.078992297 -0.016750215 -7.120921e-02
## BC005  -0.149717935 -0.021029073 -5.243404e-02
## BC010  -0.034266665  0.358434342  1.186969e-01
## BC015  -0.091160351  0.111554564  1.598349e-03
## BC016  -0.051721412 -0.120700495  5.764055e-02
## BC018  -0.137907811 -0.092531031  9.256824e-02
## BC020  -0.119436957 -0.046730086 -9.137148e-02
## BC048  -0.112649201 -0.038890730 -9.726900e-02
## BC049  -0.136090902 -0.078228735 -7.254878e-02
## BC051  -0.118390603  0.007594530 -1.242228e-01
## BC105   0.084093074 -0.034044965 -1.852660e-01
## BC108   0.099428646 -0.059675096  6.912391e-05
## BC262   0.518873631 -0.072275848 -5.580194e-03
## BCL01   0.322324156 -0.120494387 -7.381776e-02
## BCL03   0.271061702 -0.107012019 -1.252957e-01
## HNF132 -0.046338887 -0.002384683 -7.758244e-02
## HNF133 -0.137863321 -0.069536243  5.148850e-02
## HNF134 -0.118342187 -0.032704672 -2.204020e-01
## HNF144 -0.032889275  0.095615970 -3.971469e-02
## HNF168 -0.053061846  0.144691849 -1.387068e-01
## HNF185 -0.066118528  0.131169590 -7.613902e-02
## HNF187 -0.047439058  0.346363642  1.711673e-01
## HNF216 -0.039383816  0.424016071  9.247741e-02
## HNF217 -0.010444941  0.276566589  3.873096e-03
## HNF221 -0.030121564  0.117524658 -1.564290e-01
## HNF224 -0.078536472  0.122307998 -1.425948e-01
## HNF225 -0.004203634  0.393503910  1.649055e-01
## HNF229  0.510185655 -0.020212570  6.944233e-03
## HNF242  0.498331414  0.097707162  4.956120e-02
## HNF250  0.523639597 -0.084543189  6.012623e-02
## HNF267  0.559368807 -0.075283932  2.427297e-02
## HNF269  0.630988516  0.006404714  2.497916e-02
## YSF004 -0.105330456 -0.026683344 -2.234139e-01
## YSF117 -0.159226525 -0.123138031 -3.764350e-02
## YSF295 -0.080547866 -0.095123698  1.830812e-01
## YSF296 -0.114062948 -0.150500118  1.754112e-01
## YSF298 -0.168450784 -0.079840935 -3.004732e-02
## YSF300 -0.143421080 -0.053162980 -1.834378e-01
## YSF44  -0.156928968 -0.135123450  7.282300e-02
## YSF45  -0.127274082 -0.111381386  2.088572e-01
```

```
## YSF46  -0.166348929 -0.186978116  2.081296e-01
## YSF47  -0.121050749  0.018344038 -5.906552e-04
## YSF65  -0.117101326  0.006943442  4.595799e-02
## YSF66  -0.003764436 -0.056561354  8.110134e-02
## YSF67  -0.102563563  0.050933577 -1.655133e-01
## YSF69  -0.123779335 -0.065043422  1.166382e-03
## YSF70   0.013905404 -0.065223466  5.221681e-02
## YSF71  -0.126662523 -0.068094755  8.706787e-02
## YSF74  -0.127929205 -0.152019703  2.341547e-01
##
## $eig
##  [1]  2.442858e+00  1.036066e+00  7.367706e-01  5.452799e-01  4.989183e-01
##  [6]  3.787233e-01  3.286316e-01  2.651768e-01  2.416208e-01  2.325718e-01
## [11]  2.079450e-01  1.939973e-01  1.739473e-01  1.497483e-01  1.392159e-01
## [16]  1.343515e-01  1.138010e-01  1.089767e-01  9.429539e-02  9.174097e-02
## [21]  8.425808e-02  7.580800e-02  7.325889e-02  6.860878e-02  5.786790e-02
## [26]  5.166945e-02  4.998196e-02  4.385461e-02  4.201294e-02  3.490135e-02
## [31]  3.229466e-02  2.848339e-02  2.156284e-02  2.025208e-02  1.820852e-02
## [36]  1.434080e-02  8.965796e-03  7.869703e-03  3.686532e-03  1.539123e-03
## [41]  8.326673e-17 -2.031246e-03 -2.858999e-03 -5.698683e-03 -1.239255e-02
## [46] -1.577178e-02 -1.666217e-02 -2.206877e-02 -3.018042e-02 -3.510510e-02
## [51] -3.970402e-02 -6.212443e-02
##
## $x
## NULL
##
## $ac
## [1] 0
##
## $GOF
## [1] 0.4633313 0.4761311
```

```r
#Calculate explpained variation
xplainvar1 <- round(tax.pcoa$eig[1] / sum(tax.pcoa$eig), 3) * 100
xplainvar2 <- round(tax.pcoa$eig[2] / sum(tax.pcoa$eig), 3) * 100
xplainvar3 <- round(tax.pcoa$eig[3] / sum(tax.pcoa$eig), 3) * 100
sum.eig2 <- sum(xplainvar1, xplainvar2, xplainvar3)


# Define Plot Parameters
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 (", xplainvar1, "%)", sep = ""),
     ylab = paste("PCoA 2 (", xplainvar2, "%)", 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***: More varibility was explained by the taxonomic ordination (PCoA 1 = 28.4%, PCoA 2 = 12%) than for the phylogenetic ordination (PCoA 1 = 9.5%, PCoA 2 = 6%). Also, the groupings of the sites vary between the taxonomic and phylogenetic ordination (e.g., BC 105 and BC 108 are clumped alone together for the taxonomic but not the phylogenetic ordination). This shows that just because in this system two sites are similar taxonomically, they may not be phylogenetically similar.

C. Hypothesis Testing

i. Categorical Approach

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

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

# Run PERMANOVA with `adonis()` Function
adonis(dist.uf ~ watershed, permutations = 999)
```

```
##
## Call:
## adonis(formula = dist.uf ~ watershed, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model     R2 Pr(>F)
## watershed  2   0.13316 0.066579  1.2679 0.0492  0.032 *
## Residuals 49   2.57305 0.052511         0.9508
## Total     51   2.70621                  1.0000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# We can compare to PERMANOVA results based on taxonomy
adonis(
  vegdist(                              # create a distance matrix on
    decostand(comm, method = "log"),    # log-transformed relative abundances
    method = "bray") ~ watershed,       # using Bray-Curtis dissimilarity metric
  permutations = 999)
```

```
##
## Call:
## adonis(formula = vegdist(decostand(comm, method = "log"), method = "bray") ~      watershed, permuta
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## watershed  2   0.16601 0.083003  1.5689 0.06018  0.006 **
## Residuals 49   2.59229 0.052904         0.93982
## Total     51   2.75829                  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()`).

```r
# Define environmental variables related to physical and chemical properties
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
# Conduct 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: 0.056
##
## Upper quantiles of permutations (null model):
##   90%   95% 97.5%   99%
## 0.123 0.167 0.198 0.232
## Permutation: free
## Number of permutations: 999
```

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

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

```r
# Conduct dbRDA (`vegan`)
ponds.dbrda <- vegan::dbrda(dist.uf ~ ., data = as.data.frame(scale(envs)))
# Permutation tests: axes and environmental variables
anova(ponds.dbrda, by = "axis")
```

```
## Permutation test for dbrda under reduced model
## Forward tests for axes
## Permutation: free
## Number of permutations: 999
##
## Model: vegan::dbrda(formula = dist.uf ~ Elevation + Diameter + Depth + ORP + Temp + SpC + DO + pH + C
##          Df SumOfSqs      F Pr(>F)
## dbRDA1    1  0.10566 2.0152  0.460
## dbRDA2    1  0.09258 1.7658  0.625
```

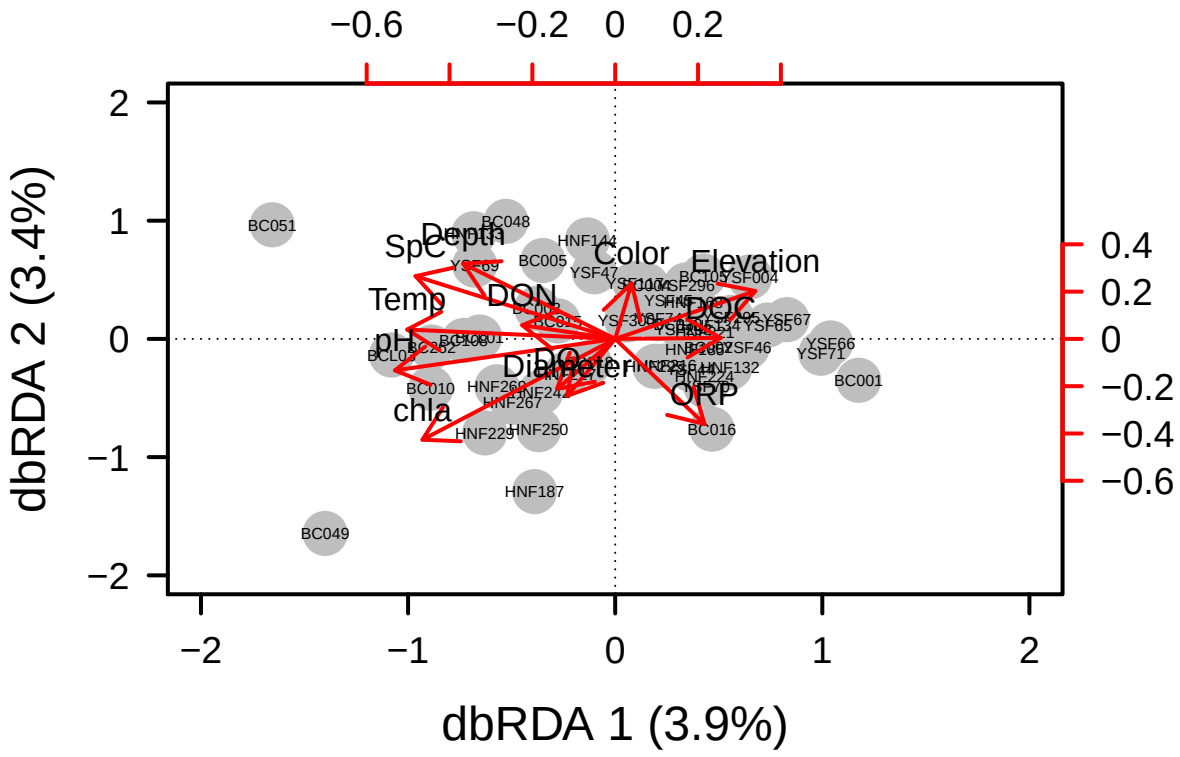
```
## dbRDA3    1  0.07555 1.4409  0.981
## dbRDA4    1  0.06677 1.2735  0.995
## dbRDA5    1  0.05666 1.0807  1.000
## dbRDA6    1  0.05293 1.0095  1.000
## dbRDA7    1  0.04750 0.9059  1.000
## dbRDA8    1  0.03941 0.7517  1.000
## dbRDA9    1  0.03775 0.7201  1.000
## dbRDA10   1  0.03280 0.6256  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  0.079 .
## Diameter  -0.27972 -0.96008 0.0541  0.252
## Depth     -0.63137  0.77548 0.1756  0.008 **
## ORP        0.41879 -0.90808 0.1437  0.025 *
## Temp      -0.98250  0.18628 0.1523  0.019 *
## SpC       -0.77101  0.63682 0.2087  0.006 **
## DO        -0.39318 -0.91946 0.0464  0.343
## pH        -0.96210 -0.27270 0.1756  0.010 **
## Color      0.06353  0.99798 0.0464  0.295
## chla      -0.60392 -0.79704 0.2626  0.012 *
## DOC        0.99847 -0.05526 0.0382  0.369
## DON       -0.91633  0.40042 0.0339  0.456
## ---
## 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 the permanova, it appears that there is a significant difference between the watersheds with respect to phylogenetic patterns. The Mantel's test suggests that there is tentative evidence (p = .054) to reject the null hypothesis and conclude that UniFrac distance is related to environmental conditions. However, not much variation is explained by the dbRDA, and none of the analyzed environmental conditions appear to have an especially big impact on the communities.

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

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

# Phylogenetic similarity among ponds (UniFrac)
unifrac.dist <- 1 - dist.uf

# Transform all distances into list format:
unifrac.dist.ls <- liste(unifrac.dist, entry = "unifrac")
bray.curtis.dist.ls <- liste(bray.curtis.dist, entry = "bray.curtis")
coord.dist.ls <- liste(coord.dist, entry = "geo.dist")
env.dist.ls <- liste(env.dist, entry = "env.dist")


#Create data frame
df <- data.frame(coord.dist.ls, bray.curtis.dist.ls[, 3], unifrac.dist.ls[, 3],
                 env.dist.ls[, 3])
names(df)[4:6] <- c("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)
```

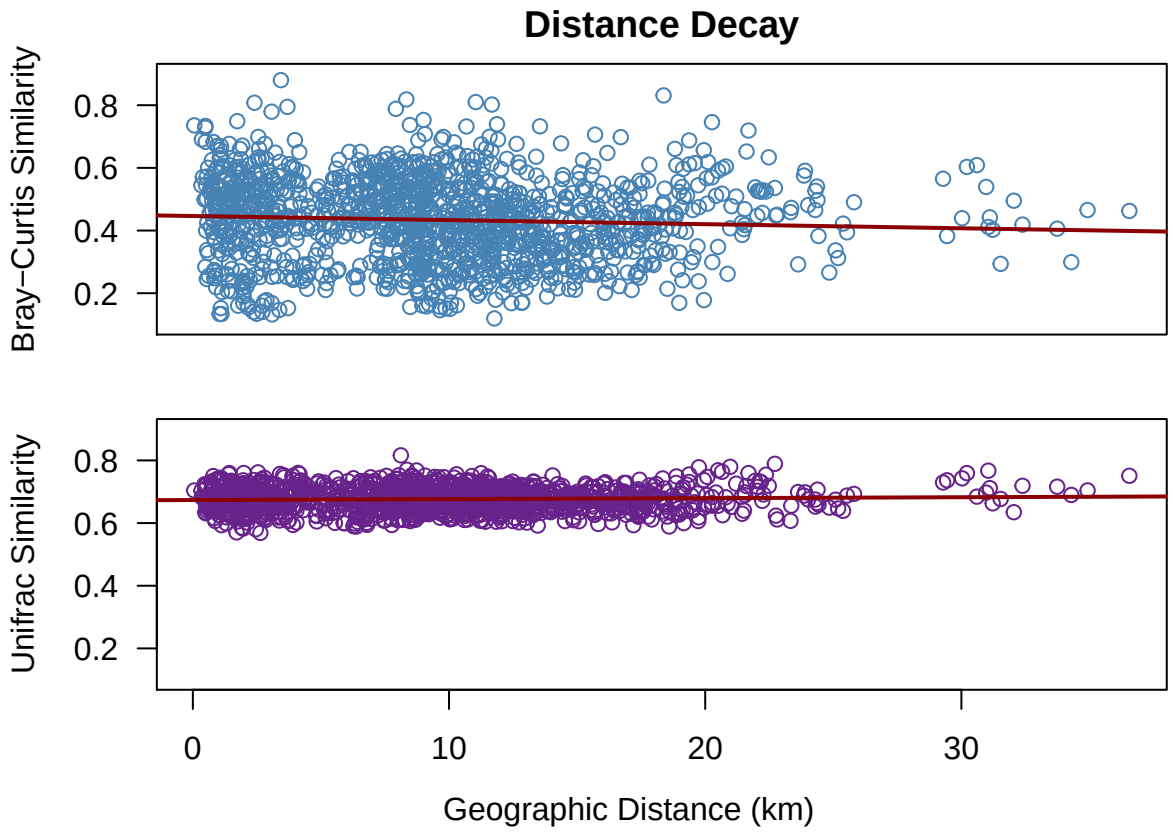
```
##
## Call:
## lm(formula = df$bray.curtis ~ df$geo.dist)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31151 -0.08843  0.00315  0.09121  0.43817
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.4463453  0.0066883  66.735   <2e-16 ***
## 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
```

```
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
```

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

```
##
```

```
## Is difference in slope significant?
## Significance is based on 1000 permutations
##
## Call:
## diffslope(x1 = df$geo.dist, y1 = df$unifrac, x2 = df$geo.dist,      y2 = df$bray.curtis)
##
## Difference in Slope: 0.001603
## Significance: 0.005
##
## Empirical upper confidence limits of r:
##     90%     95%   97.5%     99%
## 0.00076 0.00101 0.00114 0.00145
```

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 distance between the two slopes. While the distance decay for Bray-Curtis is noticeable on the plot, the distance decay for Unifrac is imperceptible. I hypothesize that evolutionary history has not significantly affected differences in distribution of OTUs at the sites, but rather that a changing environment has extirpated from some sites species that at one time had been at all of the sites.

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.

> I'm most interested in conservation biology, so incorporating the idea of phylogenetic diversity in investigations of biodiversity could be very helpful. I now understand that, when thinking about how ecosystems have been and will be impacted by human disurbance, it is important to consider phylogenetic diversity given that a more phyolgenetically diverse community likely will be more resilient. I have written reports for entities that contracted me to describe avifaunal communities in the past without reporting on phylogenetic diversity (just taxonomic diversity), but in the future I'll know the importance of considering phylogenetic diversity when reporting. In order to do this, I'll only need access to GenBank so that I can acquuire sequences to construct a phylogenetic tree of the species that I observed at a site, and the knowledge that I've gained from this course.This knowledge could ultimately help, say, a conservation organization decide which property to acquire.