9.Phylogenetic Diversity - Communities

Thomas Zambiasi; 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.

```
#clearing environment and setting wd
rm(list=ls())
getwd()
```

```
## [1] "C:/Users/tmzam/GitHub/QB2023_Zambiasi/2.Worksheets/9.PhyloCom"
```

```
setwd("C:/Users/tmzam/GitHub/QB2023_Zambiasi/2.Worksheets/9.PhyloCom")
#loading packages
library(picante)
```

```
## Warning: package 'picante' was built under R version 4.2.2

## Loading required package: ape

## Warning: package 'ape' was built under R version 4.2.2

## Loading required package: vegan

## Warning: package 'vegan' was built under R version 4.2.2

## Loading required package: permute

## Loading required package: lattice

## This is vegan 2.6-4

## Loading required package: nlme
```

```
library(ape)
library(seqinr)
```

```
## Warning: package 'seqinr' was built under R version 4.2.2

##
## Attaching package: 'seqinr'

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

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

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

```r
library(vegan)
library(fossil)
```

```
## Warning: package ’fossil’ was built under R version 4.2.2

## Loading required package: sp

## Warning: package ’sp’ was built under R version 4.2.2

## Loading required package: maps

## Loading required package: shapefiles

## Warning: package ’shapefiles’ was built under R version 4.2.2

## Loading required package: foreign

##
## Attaching package: ’shapefiles’

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

```r
library(reshape)
```

```
## Warning: package ’reshape’ was built under R version 4.2.2
```

```r
library(devtools)
```

```
## Warning: package ’devtools’ was built under R version 4.2.2

## Loading required package: usethis

##
## Attaching package: ’devtools’

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

```r
library(BiocManager)
```

```
## Warning: package ’BiocManager’ was built under R version 4.2.2

##
## Attaching package: ’BiocManager’

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

```
library(ineq)
library(labdsv)
```

```
## Warning: package 'labdsv' was built under R version 4.2.2

## Loading required package: mgcv

## This is mgcv 1.8-40. 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
```

```
library(matrixStats)
```

```
## Warning: package 'matrixStats' was built under R version 4.2.2

##
## Attaching package: 'matrixStats'

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

```
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 4.2.2

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

##
## Attaching package: 'pROC'

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

```
#loading source code
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

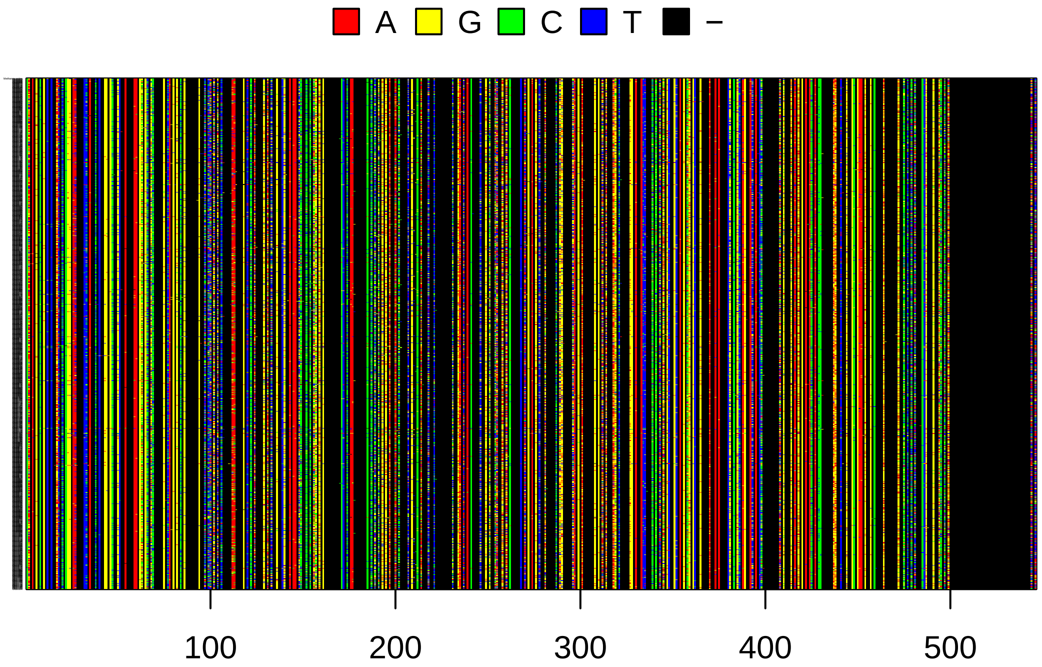
In the R code chunk below, do the following:
1. load the environmental data for the Brown County ponds (*20130801_PondDataMod.csv*),
2. load the site-by-species matrix using the `read.otu()` function,
3. subset the data to include only DNA-based identifications of bacteria,
4. rename the sites by removing 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.

```
#loading environmental pond data ad removing NAs
pond.env<-read.table("data/20130801_PondDataMod.csv",sep=",",header=TRUE)
pond.env<-na.omit(pond.env)
#loading siteXspp matrix
otucomm<-read.otu(shared="./data/INPonds.final.rdp.shared",cutoff="1")
#subsetting data so only has dna-based bacteria identification
otucomm<-otucomm[grep("*-DNA",rownames(otucomm)),]
#removing extra characters in site names
rownames(otucomm)<-gsub("\\-DNA","",rownames(otucomm))
rownames(otucomm)<-gsub("\\_","",rownames(otucomm))
#removing unnecessary sites not in environmental set
otucomm<-otucomm[rownames(otucomm)%in%pond.env$Sample_ID,]
#taking out OTUs with no abundance
otucomm<-otucomm[,colSums(otucomm)>0]
#loading taxonomic data
tax<-read.tax(taxonomy="./data/INPonds.final.rdp.1.cons.taxonomy")
```

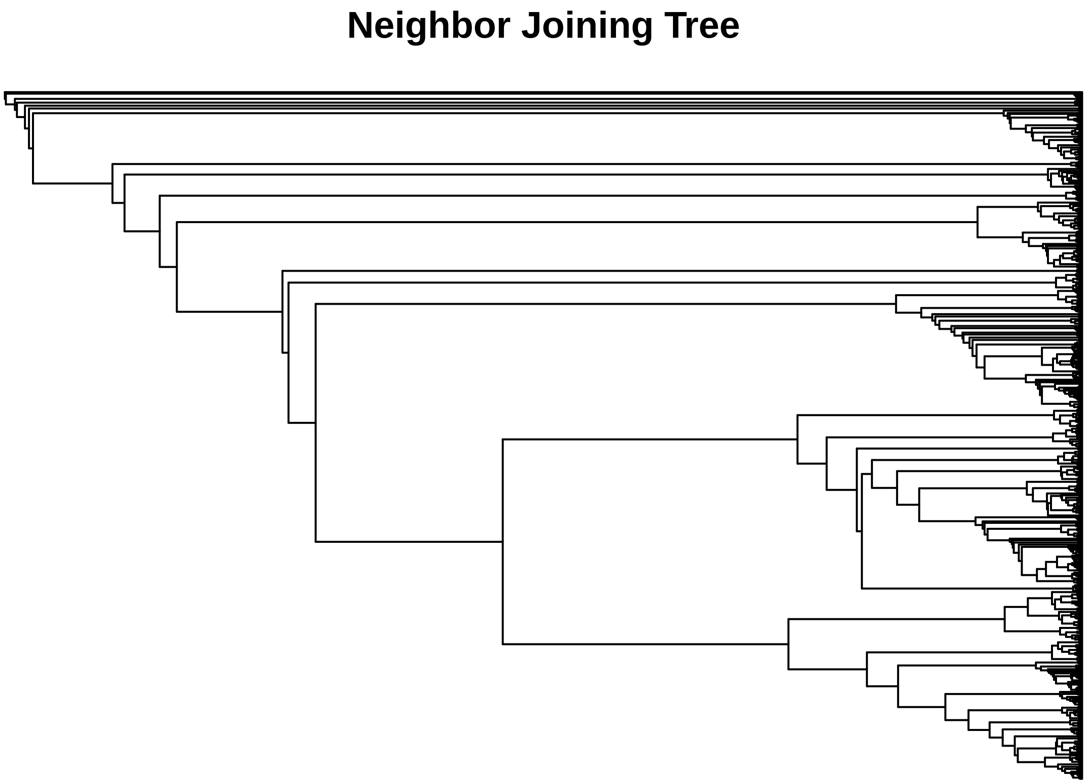
```
## Warning in type.convert.default(as.character(x)): 'as.is' should be specified
## by the caller; using TRUE

## 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.

```
#loading alignment file for OTUs
otu.align<-read.alignment(file="./data/INPonds.final.rdp.1.rep.fasta",
                          format="fasta")
#renaming OTUs
otu.align$nam<-gsub(".*\t","",otu.align$nam)
otu.align$nam<-gsub("\\|.*","",otu.align$nam)
#importing Methanosarcina sequence
outgroup<-read.alignment(file="./data/methanosarcina.fasta",format="fasta")
#converting fasta files to combined DNAbin file
combo.DNAbin<-rbind(as.DNAbin(outgroup),as.DNAbin(otu.align))
#plotting alignment
image.DNAbin(combo.DNAbin,show.labels=T,cex.lab=0.05,las=1)
```



```
#making distance matrix (will use jukes cantor model)
dist.jc<-dist.dna(combo.DNAbin,model="JC",pairwise.deletion=FALSE)
#making neighbor joining tree
njotu.all<-bionj(dist.jc)
#dropping tips of zero-occurrence OTUs
njotu<-drop.tip(njotu.all,njotu.all$tip.label[!njotu.all$tip.label%in%
                                                c(colnames(otucomm),"Methanosarcina")])
#specifying outgroup
outgroup<-match("Methanosarcina",njotu$tip.label)
#rooting tree
njotu<-root(njotu,outgroup,resolve.root=TRUE)
#plotting tree
par(mar=c(1,1,2,1)+0.1)
plot.phylo(njotu,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)

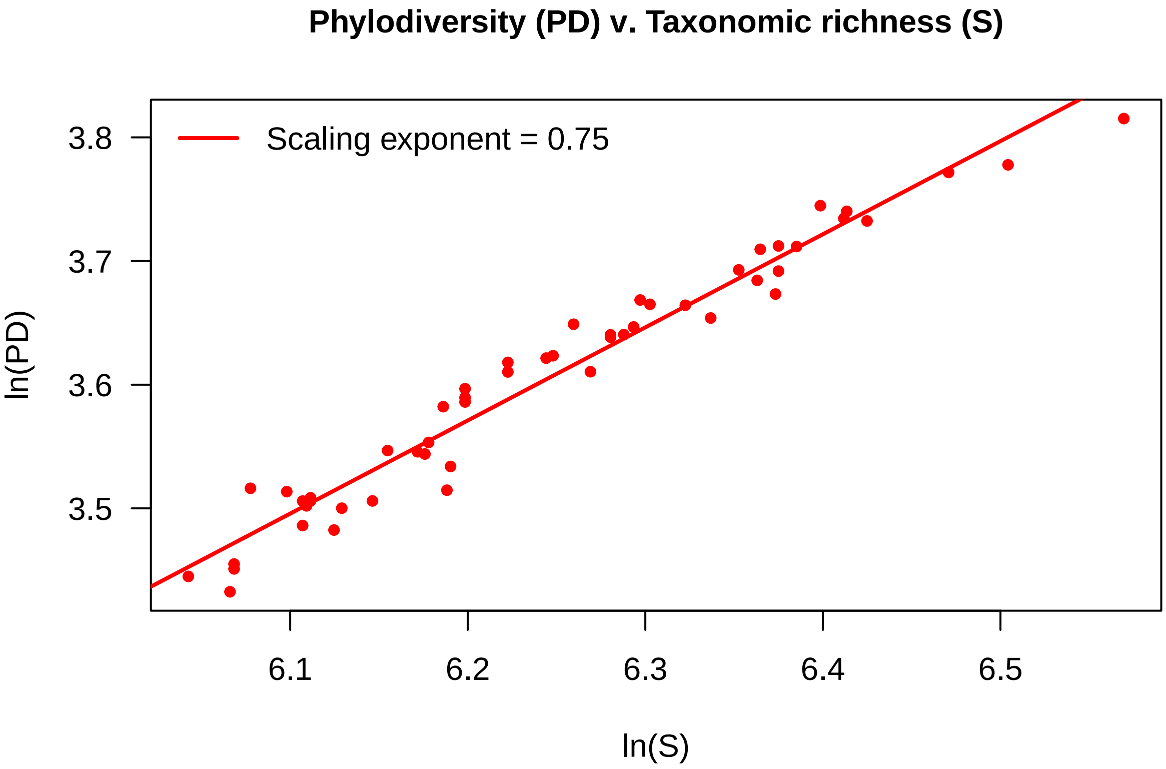
In the R code chunk below, do the following:
1. calculate Faith's D using the `pd()` function.

```
#calculating Faith D w/ pd()
pd<-pd(otucomm,njotu,include.root=FALSE)
```

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

```
#plotting sp. richness against diversity
#making biplot
par(mar=c(5,5,4,1)+0.1)
plot(log(pd$SR),log(pd$PD),pch=20,col="red",las=1,xlab="ln(S)",ylab="ln(PD)",
     cex.main=1,main="Phylodiversity (PD) v. Taxonomic richness (S)")
#test of power-law relationship
fit<-lm('log(pd$PD)~log(pd$SR)')
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***: Phylodiversity (PD) is calculated by adding together the branch lengths of the species in the phylogeny of interest. Having a higher species richness in the community will generally result in higher values for PD since more species present allow for more branches to be added together. Although communities with the same number of species may have different levels of phylogenetic diversity, greater species richness will increase the likelihood of having species in the community that aren't as closely related.
> ***Answer 1b***: In the lake dataset, we see a positive linear relationship between the log of species richness and the log of phylodiversity. Essentially, increases in species richness are associated with increased phylodiversity.
> ***Answer 1c***: I would expect the species richness and phylodiversity to diverge when looking at relatively homogenous communities. These could describe many different situations, such as biological invasions/anthropogenic homogenization or specialist habitats that only contain a set of closely related species with similar adaptations (hot springs with extremophile bacteria, tundra or apline areas with cold-adapted plants). These may be relatively scale-dependent, but overall I would expect these examples to have communities where organisms are more closely related (likely for ecological reasons) than similar environments that may not be as specialized and richness may not exactly translate to phylodiversity.

Answer 1d: The PD-S scaling exponent appears to be the slope of the linear regression of the log of species richness and the log of phylodiversity. For every increase in 1 of the log of species richness, the log of phylodiversity will increase by 0.75.

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.pd using richness as null model
ses.rich<-ses.pd(otucomm[1:2,],njotu,null.model="richness",runs=25,
               include.root=FALSE)
ses.rich
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank   pd.obs.z  pd.obs.p
## BC001   668 43.71912     43.89881   0.891419          13 -0.2015780 0.5000000
## BC002   587 40.94334     39.93534   1.023774          22  0.9845944 0.8461538
##       runs
## BC001   25
## BC002   25
```

```
#ses.pd using sample.pool as null model
ses.sample<-ses.pd(otucomm[1:2,],njotu,null.model="sample.pool",runs=25,
                   include.root=FALSE)
ses.sample
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank   pd.obs.z  pd.obs.p
## BC001   668 43.71912     44.12204  0.9398537          10 -0.4286966 0.3846154
## BC002   587 40.94334     39.79188  0.7886433          25  1.4600470 0.9615385
##       runs
## BC001   25
## BC002   25
```

```
#ses.pd using taxa.labels as null model
ses.taxa<-ses.pd(otucomm[1:2,],njotu,null.model="taxa.labels",runs=25,
                 include.root=FALSE)
ses.taxa
```

```
##       ntaxa   pd.obs pd.rand.mean pd.rand.sd pd.obs.rank   pd.obs.z  pd.obs.p
## BC001   668 43.71912     44.29020  0.8043978           9 -0.7099494 0.3461538
## BC002   587 40.94334     39.87397  0.9101605          24  1.1749214 0.9230769
##       runs
## BC001   25
## BC002   25
```

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

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

Answer 2a: When we calculate the standardized effect size using ses.pd and randomization, the null hypothesis is that observed phylodiversity is no different from what would be expected in a randomly assembled community. Our alternate hypothesis is that the observed phylodiversity in our community is gerater than what would be expected of a randomly assembled community. ***Answer 2b***: The different null models had slight influences on the calculated ses.pd values. All three null models I used - richness, frequency, and taxa labels - had fairly close ses.pd values between 1.3 and 1.65 for the second pond. These are all greater than 0 (with the richness model producing the highest value), indicating that pond 2 has greater phylodiversity than expected under random assembly. The three models all gave fairly close values (from -0.4 to 0.1) for pond 1, but only the richness null model resulted in a positive ses.pd value. Overall, the richness null model resulted in the highest ses.pd values and indicated the greatest difference between the observed and null communities in terms of phylodiversity. It seems that this happens because the richness null model holds the number of species in the null community the same as in the observed community; with richness being controlled for, the ses.pd value reflects the difference in phylogenetic relatedness between the observed and null communities. The other models do not keep richness as a constant, which could mean that some randomized null communities may have higher species richness. We established earlier in this worksheet that communities with higher species richness tend to express more diversity, so this could result in the null community diversity for the other models appearing closer to what was observed.

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.

```
#making phylogenetic distance matrix
phydist<-cophenetic.phylo(njotu)
```

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.

```
#finding standardized effect size of NRI using randomization
ses.mpd<-ses.mpd(otucomm,phydist,null.model="taxa.labels",abundance.weighted=FALSE,
                 runs=25)
#calculating 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  -2.4296532
## BC002  -2.5802693
## BC003  -1.6896864
## BC004  -3.2555057
## BC005  -3.4881340
## BC010  -2.8566724
## BC015  -2.1989979
## BC016  -1.9444615
## BC018  -1.5282289
## BC020  -1.3271275
## BC048  -1.9802229
## BC049  -0.4820865
## BC051  -2.6173651
## BC105  -3.4020491
## BC108  -2.3273909
## BC262  -1.8298813
## BCL01  -4.1190347
## BCL03  -1.0990254
## HNF132 -2.5429372
## HNF133 -2.6584719
## HNF134 -2.6258020
## HNF144 -4.3414488
## HNF168 -1.7067175
## HNF185 -3.5243094
## HNF187  0.3943001
## HNF216 -1.9026761
## HNF217 -2.0848178
## HNF221 -2.4491792
## HNF224 -3.0443043
## HNF225 -2.0091628
## HNF229 -1.1264550
## HNF242 -1.2853293
## HNF250 -1.9845990
## HNF267 -1.1920166
## HNF269 -2.6750618
## YSF004 -3.9651777
## YSF117 -2.2523451
## YSF295 -2.3675797
## YSF296 -2.9893293
## YSF298 -4.1105973
## YSF300 -2.8154160
## YSF44  -1.3268697
## YSF45  -2.3171019
## YSF46  -1.5254238
## YSF47  -2.7650444
## YSF65  -0.3003620
## YSF66  -1.4805678
## YSF67  -2.3949293
## YSF69  -1.4992460
## YSF70  -1.8261692
## YSF71  -1.4137810
## YSF74  -2.8357182
```

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
#finding standardized effect size of NRI using randomization
ses.mntd<-ses.mntd(otucomm,phydist,null.model="taxa.labels",abundance.weighted=FALSE,
                 runs=25)
```

```
#calculating 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   0.82473936
## BC002  -1.55865004
## BC003   0.09235420
## BC004  -1.46153421
## BC005  -1.84527816
## BC010  -1.19352502
## BC015  -0.88664231
## BC016  -0.07514170
## BC018  -0.67268782
## BC020  -0.73582366
## BC048  -1.95267520
## BC049   0.03692127
## BC051  -0.66057848
## BC105  -1.17207780
## BC108  -1.57602510
## BC262   0.37313876
## BCL01  -1.57864846
## BCL03   0.35035016
## HNF132 -0.30510247
## HNF133 -0.59846366
## HNF134 -0.73087816
## HNF144 -2.24106202
## HNF168 -1.82733949
## HNF185 -0.71381803
## HNF187  0.26537038
## HNF216 -2.66247366
## HNF217 -1.28975557
## HNF221 -0.77664808
## HNF224 -1.86673329
## HNF225 -1.46608057
## HNF229  0.21306080
## HNF242 -1.43593582
## HNF250 -0.71369265
## HNF267  1.16940368
## HNF269  0.49208684
## YSF004 -1.98032139
## YSF117 -1.16231106
## YSF295 -1.33995557
## YSF296  0.04206708
## YSF298 -1.05089618
## YSF300 -0.68124095
## YSF44  -0.99004841
## YSF45  -1.00083600
## YSF46  -0.62099434
## YSF47  -0.54586325
## YSF65   0.35992813
## YSF66   1.60745612
## YSF67  -0.60369770
## YSF69   0.19278495
## YSF70  -0.09641717
## YSF71  -0.44469117
## YSF74  -1.05348784
```

Question 3:

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

> ***Answer 3a***: NRI is a metric used to determine whether community members are more or less closely related than expected under a null model. This is calculated by finding the average length of the branches between every pair of species in the community and comparing this value to the one calculated from a randomly-assembled community.
> ***Answer 3b***: NTI is very similar to NRI in that it is used to determine whether species in a comunity are more or less closely related than expected under a null model. However, this metric is calculated by finding the average branch distance to each species' nearest neighbor on the tree and comparing this value to the one calculated from a randomly-assembled community.
> ***Answer 3c***: Only one NRI value is positive, indicating that only pond HNF187 (0.6801283) is underdispersed (spp. more closely related than expected by null). All other NRI values are negative, meaning that all other ponds in the dataset have overdispersed communities. Several pond sites have positive NTI values, including BC001, BC016, BCL03, HNF267, HNF269, and YSF66. The communities in these ponds are underdispersed, meaning they are more closely related than would be expected under a null model. The remaining ponds all had negative NTI values, meaning they were all overdispersed (less closely related than expected under null).

```
#Rerunning NRI and NTI code using abundance data
#NRI
#finding standardized effect size of NRI using randomization
ses.mpd2<-ses.mpd(otucomm,phydist,null.model="taxa.labels",abundance.weighted=TRUE,
                 runs=25)
#calculating NRI
NRI2<-as.matrix(-1*((ses.mpd2[,2]-ses.mpd2[,3])/ses.mpd2[,4]))
rownames(NRI2)<-row.names(ses.mpd2)
colnames(NRI2)<-"NRI"
NRI2
```

```
##                   NRI
## BC001   0.0382665425
## BC002   0.3985867918
## BC003   0.4571007823
## BC004  -0.0190941592
## BC005   0.3476544821
## BC010   0.3650725822
## BC015   0.0125531079
## BC016   0.4535449323
## BC018   0.2174913599
## BC020   0.2313853399
## BC048  -0.1487869844
## BC049   0.2225553941
## BC051  -0.4154939147
## BC105  -0.3275432695
## BC108  -0.1200587151
## BC262   0.0674161231
## BCL01  -0.0027201694
## BCL03  -0.3525020937
## HNF132 -0.0951420822
## HNF133  0.0722313446
## HNF134  0.2236271306
## HNF144 -0.1246846989
## HNF168 -0.0074097132
## HNF185  0.1191141868
## HNF187  0.5533278031
## HNF216  0.5340801771
## HNF217 -0.1128106207
## HNF221 -0.5193230312
## HNF224  0.0262624095
## HNF225  0.7433224469
## HNF229  0.0786737404
## HNF242  0.2222118553
## HNF250  0.0767651797
## HNF267  0.0005886521
## HNF269  0.0601737406
## YSF004 -0.1149409046
## YSF117  0.6265659995
## YSF295 -1.0158825370
## YSF296  0.9044881247
## YSF298  0.7489034602
## YSF300  0.3698755132
## YSF44   0.3274002728
## YSF45   0.3998849308
## YSF46   0.9761000969
## YSF47   0.2931341499
## YSF65   0.4625528875
## YSF66  -0.3006570309
## YSF67  -0.1112919014
## YSF69  -0.1087496067
## YSF70  -0.7172117887
## YSF71   0.3373986299
## YSF74   0.5789538733
```

```
#NTI
#finding standardized effect size of NRI using randomization
ses.mntd2<-ses.mntd(otucomm,phydist,null.model="taxa.labels",abundance.weighted=TRUE,
                 runs=25)
#calculating NTI
NTI2<-as.matrix(-1*((ses.mntd2[,2]-ses.mntd2[,3])/ses.mntd2[,4]))
rownames(NTI2)<-row.names(ses.mntd2)
colnames(NTI2)<-"NTI"
NTI2
```

```
##                NTI
## BC001   0.8102922
## BC002   1.4876320
## BC003   0.9817286
## BC004   1.0066225
## BC005   1.9228151
## BC010   0.3971855
## BC015   0.8326201
## BC016   1.4346767
## BC018   1.2699599
## BC020   1.2186612
## BC048   1.0536732
## BC049   1.5326748
## BC051   1.6251927
## BC105   1.5297162
## BC108   1.3321956
## BC262   1.2882613
## BCL01   1.5048053
## BCL03   0.8856729
## HNF132  1.1198160
## HNF133  0.9925027
## HNF134  1.4880220
## HNF144  0.5986334
## HNF168  0.4778946
## HNF185  1.0973424
## HNF187  0.4341880
## HNF216  0.1461784
## HNF217  0.3146742
## HNF221  0.3312050
## HNF224  1.2400205
## HNF225  0.2575386
## HNF229  1.5928440
## HNF242  1.5035447
## HNF250  1.3086807
## HNF267  0.9504425
## HNF269  0.9361538
## YSF004  0.3057234
## YSF117  1.5995971
## YSF295 -1.0619710
## YSF296  1.8979964
## YSF298  1.5232887
## YSF300  1.3650432
## YSF44   0.9170660
## YSF45   0.8298281
## YSF46   1.3129135
## YSF47   0.8711960
## YSF65   1.1078528
## YSF66   1.3115224
## YSF67   0.7413691
## YSF69   1.1781011
## YSF70   0.9862220
## YSF71   1.1708787
## YSF74   1.4710827
```

Answer 3d: Changing the NRI and NTI calculations so that they're based on abundance data changes the results quite drastically. For NRI, only 18 of the 52 sites have a negative value indicating that the species there are less closely related than expected under a null model (overdispersed). Unlike with the presence/absence data, most sites for NRI would be classified as underdispersed. The change is even more drastic for NTI. Only one of the sites, YSF295, has a negative value indicating that its taxa are overdispersed. The vast majority of ponds with the NTI calculation using abundance data are underdispersed. The different data type has flipped the conclusions about what form of phylogenetic dispersion is most common among the ponds.

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.

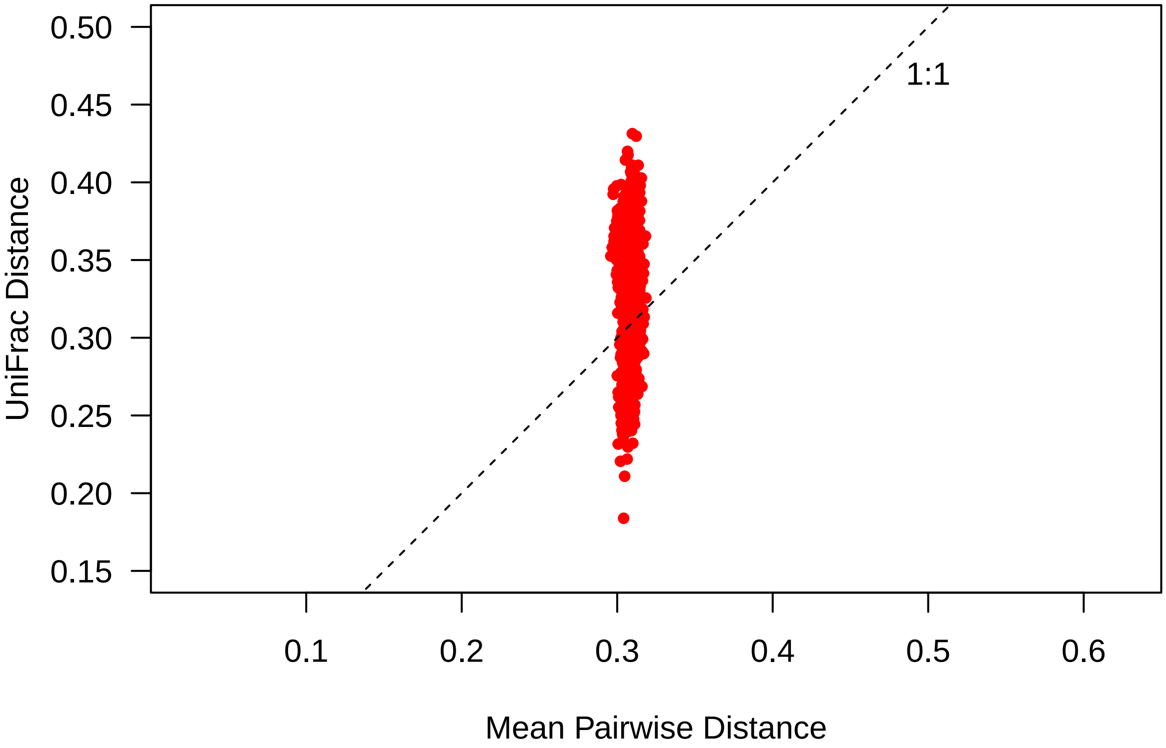
```
#calculating resemblance matrix with Mean Pairwise Distance
dist.mp<-comdist(otucomm,phydist)
```

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

```
#calculating resemblance matrix with unifrac distance
dist.uf<-unifrac(otucomm,njotu)
```

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 Pairwise 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 and UniFrac distance are both methods of determining phylogenetic distance between community members. Mean Pairwise Distance simply finds the average distance on the tree (by branch length) between every possible species pairing. The UniFrac distance calculates phylogenetic distance between species by finding the ratio of lengths of branches not shared by the species to the total length of branches involved in the pairing.
Answer 4b: The Mean Pairwise Distance and UniFrac Distance metrics do not vary that directly with one another. For any small range of distances measured via MPD (essentially a single value), UniFrac will calculate a much wider range of distance values. This is very clear from the plot above; MPD stays right around 0.3, but UniFrac ranges from about 0.17 to 0.45.
Answer 4c: MPD might have less variation than UniFrac because it only really assesses the total branch lengths between species pairs. Many different pairings on a complex phylogeny with plenty of branching may have very similar branch lengths, which would result in a fairly low variance for the distance metric. UniFrac factors in the distance of branch lengths that aren't shared by a pair of species. Different pairs of species may be very closely related and share many branches or be distantly related and share few branches, so UniFrac will result in a much more varied distance calculation.

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.

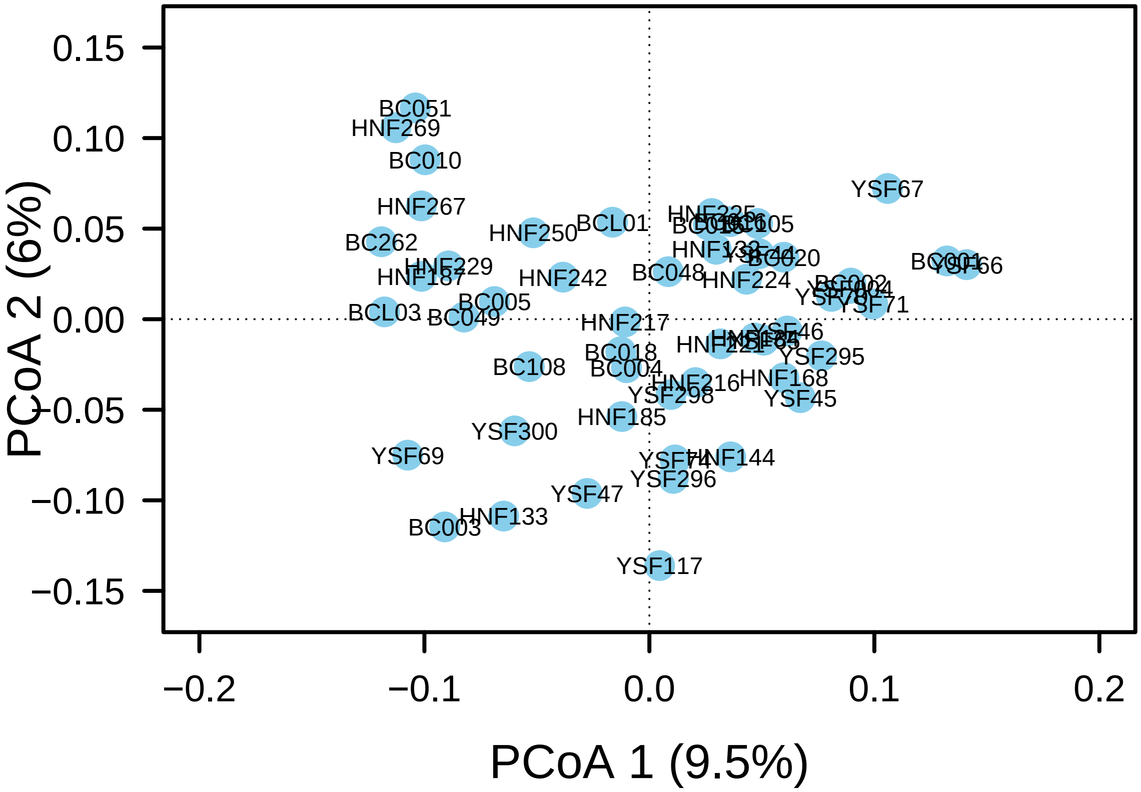
In the R code chunk below, do the following:
1. perform a PCoA based on the UniFrac distances, and
2. calculate the explained variation for the first three PCoA axes.

```
#pcoa for unifrac
pond.pcoa<-cmdscale(dist.uf,eig=T,k=3)
#calculating variation explained by first 3 axes
explvar1<-round(pond.pcoa$eig[1]/sum(pond.pcoa$eig),3)*100
explvar2<-round(pond.pcoa$eig[2]/sum(pond.pcoa$eig),3)*100
explvar3<-round(pond.pcoa$eig[3]/sum(pond.pcoa$eig),3)*100
sum.eig<-sum(explvar1,explvar2,explvar3)
```

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.

```
#plot parameters
par(mar=c(5,5,1,2)+0.1)
#making plot
plot(pond.pcoa$points[,1],pond.pcoa$points[,2],xlim=c(-0.2,0.2),ylim=c(-0.16,0.16),
     xlab=paste("PCoA 1 (",explvar1,"%)",sep=""),
     ylab=paste("PCoA 2 (",explvar2,"%)",sep=""),pch=16,cex=2.0,type="n",
     cex.lab=1.5,cex.axis=1.2,axes=FALSE)
#adding 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)
#adding points and labels
points(pond.pcoa$points[,1],pond.pcoa$points[,2],pch=19,cex=2,bg="skyblue",col="skyblue")
text(pond.pcoa$points[,1],pond.pcoa$points[,2],labels=row.names(pond.pcoa$points),
     cex=0.75)
```



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 with taxonomic data
#creating resemblance matrix using bray-curtis distance
comm.db<-vegdist(otucomm,method="bray")
#doing pcoa
comm.pcoa<-cmdscale(comm.db,eig=TRUE,k=3)
comm.ev1<-round(comm.pcoa$eig[1]/sum(comm.pcoa$eig),3)*100
comm.ev1
```

```
## [1] 28.4
```

```
comm.ev2<-round(comm.pcoa$eig[2]/sum(comm.pcoa$eig),3)*100
comm.ev2
```

```
## [1] 12
```

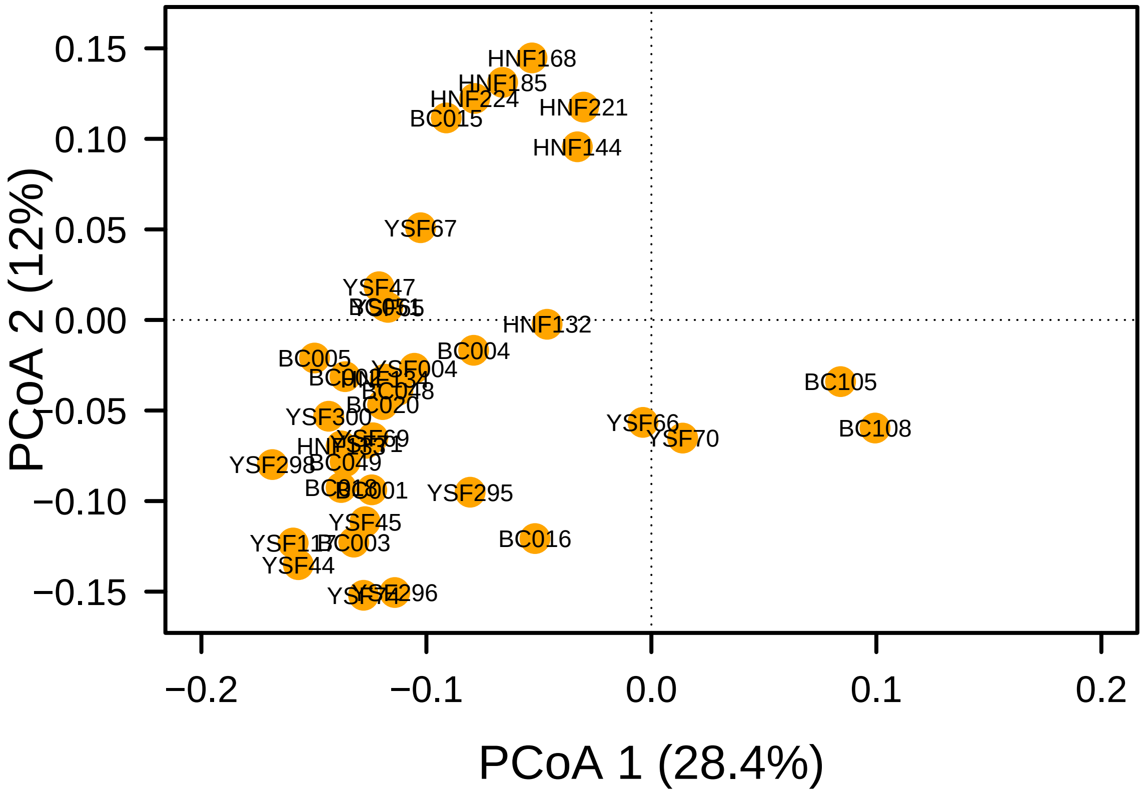
```
comm.ev3<-round(comm.pcoa$eig[3]/sum(comm.pcoa$eig),3)*100
comm.ev3
```

```
## [1] 8.6
```

```
comm.sumeig<-sum(comm.ev1,comm.ev2,comm.ev3)
comm.sumeig
```

```
## [1] 49
```

```
#making plot to visualize
#plot parameters
par(mar=c(5,5,1,2)+0.1)
#making plot
plot(comm.pcoa$points[,1],comm.pcoa$points[,2],xlim=c(-0.2,0.2),ylim=c(-0.16,0.16),
     xlab=paste("PCoA 1 (",comm.ev1,"%)",sep=""),
     ylab=paste("PCoA 2 (",comm.ev2,"%)",sep=""),pch=16,cex=2.0,type="n",
     cex.lab=1.5,cex.axis=1.2,axes=FALSE)
#adding 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)
#adding points and labels
points(comm.pcoa$points[,1],comm.pcoa$points[,2],pch=19,cex=2,bg="orange",col="orange")
text(comm.pcoa$points[,1],comm.pcoa$points[,2],labels=row.names(comm.pcoa$points),
     cex=0.75)
```



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 two principal coordinate analyses (based in phylogeny and taxonomy) seem to have relatively large distances in percentage of variation explained by the different axes. The first two axes under the phylogenetic model explain 9.5% and 6% of the variation, while in the taxonomic PCoA the first two axes explain 28.4% and 12% of the variation. This seems to conflict slightly with the patterns seen within the ordinations. The taxonomic ordination - which seems to explain more sequence variation within the bacterial community - has more clustering of the taxa along the horizontal first axis and a wider spread of the taxa along the vertical second axis. In the phylogenetic ordination, we see the taxa fairly evenly spread across both axes. Although the taxonomic ordination appears to explain more of the variation, the lack of clustering in the phylogenetic PCoA appears to allow for a more thorough comparison of individual taxa along the two axes. Because of this, I would conclude that phylogenetic information is important for determining actual differences between individual bacterial taxa. The lack of clustering in the phylogenetic PCoA would also make it easier to determine which bacterial taxa are more closely related to one another.

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.

```
#specifying environmental categories
watershed<-pond.env$Location
#doing PERMANOVA analysis
phylo.adonis<-adonis2(dist.uf~watershed,permutations=999)
phylo.adonis
```

```
## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 999
##
## adonis2(formula = dist.uf ~ watershed, permutations = 999)
##           Df SumOfSqs     R2      F Pr(>F)
## watershed  2  0.13316 0.0492 1.2679  0.032 *
## Residual  49  2.57305 0.9508
## Total     51  2.70621 1.0000
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
#comparing to PERMANOVA based on taxonomy; will involve making a distance
#matrix for log-transformed relative abundances using a Bray-Curtis metric
tax.adonis<-adonis2(vegdist(decostand(otucomm,method="log"),method="bray")
                    ~watershed,permutations=999)
tax.adonis
```

```
## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 999
##
## adonis2(formula = vegdist(decostand(otucomm, method = "log"), method = "bray") ~ watershed, permutati
##           Df SumOfSqs      R2      F Pr(>F)   
## watershed  2  0.16601 0.06018 1.5689  0.006 **
## Residual  49  2.59229 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()`).

```
#subsetting variables for physical/chemical properties of ponds
envs<-pond.env[,5:19]
#getting rid of redundant variables
envs<-envs[,-which(names(envs)%in%c("TDS","Salinity","Cal_Volume"))]
#making distance matrix to find environmental distance between ponds
env.dist<-vegdist(scale(envs),method="euclid")
env.dist
```

```
##           1        2        3        4        5        6        7
## 2  3.926087                                                      
## 3  3.682537 4.523855                                             
## 4  3.761515 2.219279 4.257997                                    
## 5  3.060147 2.746652 4.123132 1.655233                           
## 6  5.055940 5.387317 5.323267 4.127621 4.146190                  
## 7  4.781864 4.860117 4.853669 5.231987 5.202675 5.851210         
## 8  2.980519 2.277859 4.817545 2.809496 2.764444 4.983184 4.171228
## 9  5.405292 4.781893 4.262966 4.987421 5.024253 5.997176 3.109246
## 10 4.576676 4.271373 2.737799 3.591474 3.973858 4.709103 5.346219
## 11 5.463851 4.751248 3.613136 4.109551 4.504486 4.691747 5.538011
## 12 8.750984 7.947683 8.182127 7.843458 8.141829 8.296324 8.623668
## 13 5.891495 4.749821 4.816683 4.597797 5.218806 6.228800 4.944954
## 14 4.610817 4.015423 2.427467 3.952770 4.042204 5.620621 5.129826
## 15 4.632544 3.837039 4.886468 3.232454 3.907974 4.372837 3.506292
## 16 4.018698 3.027075 3.884779 3.155069 3.532800 3.999194 3.229807
## 17 6.086370 5.979765 3.836575 6.129762 5.915754 5.710004 5.386449
## 18 6.307248 4.417289 6.322835 4.950370 5.376387 6.314805 4.181309
## 19 3.218067 4.336003 3.766275 3.921982 4.058730 6.083983 5.195095
## 20 5.726498 3.269296 5.088590 3.939235 4.444865 5.930275 3.858868
## 21 4.472862 2.645191 5.621437 2.684463 3.497497 6.067567 5.727955
## 22 5.725530 4.635340 6.031088 4.382700 5.378640 6.295476 5.753711
## 23 5.887140 4.853863 6.241556 4.636550 5.574360 6.621876 5.170263
## 24 4.960005 3.754815 6.353339 3.949102 4.711234 6.463883 5.890235
## 25 5.083642 4.183639 6.410684 4.599233 5.470094 6.881958 5.035502
## 26 4.716278 3.676519 6.222880 3.984227 4.521703 5.904433 6.279303
## 27 4.870017 3.347209 5.010329 3.033084 4.212669 5.045749 4.261598
## 28 4.742908 4.897960 6.612459 4.698099 5.235014 6.761189 6.453802
## 29 5.820222 3.869077 6.060998 4.252877 5.166338 6.498438 5.850895
## 30 5.386506 4.043792 6.905190 4.606630 5.055816 7.577624 6.158784
## 31 5.141874 4.360308 4.970081 4.143762 4.735094 4.026986 4.720056
## 32 5.242132 3.783979 4.908992 3.508530 4.323396 4.847881 5.166268
## 33  7.778144  6.289534  9.207465  7.638448  7.760637  9.713698  8.343351
## 35  5.387267  3.578651  4.755867  4.326806  4.895139  6.307441  4.779848
## 36  4.538342  2.715397  4.542719  4.057802  4.067033  5.938975  3.512394
## 37  6.021651  4.717369  6.096870  5.522975  5.552899  7.000445  4.819180
## 38  4.186834  4.655385  3.694927  4.839409  4.370112  5.738133  5.640424
## 39  3.479810  3.200614  4.117360  3.204025  3.529082  4.961671  2.827634
## 40  3.007935  3.956036  3.649481  3.760500  3.855182  5.039997  3.607948
## 41  4.481813  3.517851  3.505156  4.331994  4.566159  6.327068  3.683178
## 42  4.408320  3.384063  4.392676  3.773345  4.256983  6.149541  4.487584
## 43  4.544940  2.674845  5.393479  3.031824  3.635287  5.971535  4.582629
## 44  3.739417  3.051307  4.755246  2.756843  2.947909  5.289276  3.814796
## 45  3.641648  1.768200  4.658178  2.275006  2.954150  5.494298  5.034674
## 46  4.849152  2.890426  4.160600  3.440594  3.834575  5.275827  3.708536
## 47  4.713463  2.485427  5.391636  2.735354  2.792944  5.604219  6.314981
## 48  3.748947  2.905054  4.715050  3.647683  3.908871  5.947293  3.299940
## 49  5.170902  3.186804  4.748967  4.468346  4.637927  6.145399  4.711553
## 50  4.098498  4.323837  5.789591  3.619556  3.949342  6.093455  5.793416
## 51  5.140261  4.611372  6.334092  4.674338  5.235688  5.449603  4.886168
## 52  4.522508  3.186258  5.213482  3.811031  4.105237  6.617212  4.035127
## 53  4.794195  2.862106  5.759322  3.488081  3.923257  6.039112  4.642011
##            8         9        10        11        12        13        14
## 2
## 3
## 4
## 5
## 6
## 7
## 8
## 9   4.709936
## 10  5.010138  4.800845
## 11  5.588788  5.418712  1.829252
## 12  8.694740  8.739030  7.835478  7.554958
## 13  5.792543  5.255483  3.833104  3.424600  6.791708
## 14  4.897898  4.515354  1.869047  2.675596  7.783863  4.231309
## 15  3.827716  4.033921  3.902672  4.180985  7.553803  3.339425  4.579790
## 16  3.235716  3.841731  3.040421  3.189545  7.469317  3.638612  3.350414
## 17  6.421370  4.592572  4.454888  4.669018  7.714675  6.031839  3.708217
## 18  5.080738  4.781731  5.799733  5.452489  7.557800  3.684467  6.022092
## 19  4.156081  5.409945  3.706351  4.708150  7.982054  4.761234  4.137512
## 20  4.212752  4.191716  4.349870  3.946710  7.491215  3.479723  4.030394
## 21  3.315720  6.043004  4.776264  5.507774  8.392374  5.207154  4.780406
## 22  5.421152  6.421603  5.170183  5.282932  7.502310  3.072559  5.855528
## 23  5.194184  6.154853  5.345484  5.426835  7.607131  3.369309  5.740972
## 24  4.035560  6.738342  5.653710  6.226107  8.768824  5.423574  5.703135
## 25  4.167876  6.208248  5.990903  6.586279  8.615876  5.121588  6.245720
## 26  4.428130  6.928650  5.216251  5.734502  8.413009  5.005830  5.717484
## 27  3.677879  4.723684  4.614793  4.819362  7.647423  3.745169  5.066767
## 28  4.860879  7.517777  6.171157  6.867612  9.089907  5.877272  6.474832
## 29  5.107487  6.131021  5.209719  5.273376  7.398206  3.162740  5.684993
## 30  4.703930  6.676405  6.610176  7.157153  8.466471  5.205039  6.651555
## 31  4.717540  4.351195  4.440985  4.932838  8.391861  4.760628  5.267595
## 32  4.787079  4.541405  4.041070  4.541611  7.372300  3.562738  4.804565
## 33  7.288958  9.144773  8.664983  8.916699 10.046045  7.477229  8.648327
## 35  4.257453  5.120755  4.479417  4.246535  6.620062  3.977104  4.060739
## 36  2.940345  3.666263  4.546100  4.881374  8.546988  5.138821  3.791837
## 37  5.209786  5.529695  5.147452  4.918916  7.533555  4.214169  4.871550
## 38  4.869782  5.879473  3.448647  3.509646  8.397055  5.243410  3.226000
## 39  2.239045  3.930208  4.262372  4.666781  8.211248  4.718453  4.202173
## 40  3.200978  4.923953  4.091736  4.329085  7.755489  4.475251  4.118537
## 41  3.825574  3.613120  3.634188  4.265253  8.318784  4.559751  2.711885
## 42  3.482672  4.856175  4.253784  4.717021  7.785325  4.547984  3.693697
## 43  3.232235  5.019377  4.857845  5.163661  7.175530  3.771645  4.805383
## 44  2.889537  4.475319  4.239077  4.649956  7.355227  3.931546  4.366597
## 45  2.118847  5.350787  4.302994  4.941633  8.370201  5.121731  4.103850
## 46  3.621490  3.983833  3.540150  3.350938  6.974622  3.553168  3.070830
## 47  3.662395  5.952825  4.809201  5.074729  7.247013  5.135993  4.621049
## 48  2.766260  4.685726  4.590876  4.957670  7.936876  4.178065  4.232915
## 49  4.175668  4.417109  3.991940  4.486172  8.482931  4.997640  3.446074
## 50  4.396062  6.535217  5.438428  6.034160  8.277735  4.768379  5.851666
## 51  4.644666  6.041256  5.707404  5.802634  7.833240  4.214535  6.400776
## 52  2.894895  4.256903  5.473092  5.866647  8.239924  4.862044  5.137902
## 53  3.661992  5.426545  4.938863  5.025294  7.540344  3.673299  5.122502
##            15        16        17        18        19        20        21
## 2
## 3
## 4
## 5
## 6
## 7
## 8
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16  2.234863
## 17  5.982366  4.580433
## 18  3.514516  3.794729  6.907872
## 19  4.235814  4.125177  6.681587  5.983062
## 20  3.417267  2.685404  5.578329  2.942384  5.228350
## 21  3.931337  4.032905  7.326447  5.590494  4.304557  4.463944
## 22  3.347877  4.390662  7.723802  4.364358  5.006855  4.901051  4.325573
## 23  3.057842  4.247059  7.700903  4.711004  4.961974  4.516785  4.180726
## 24  4.170983  4.487384  7.799326  6.029920  5.276266  5.254439  2.248811
## 25  3.759237  4.449511  8.106315  5.138753  4.771989  5.134543  3.263478
## 26  4.196941  4.188887  7.617122  5.351403  5.113540  5.377626  3.229766
## 27  2.357872  3.195065  6.622358  3.852846  4.559504  3.708482  3.668255
## 28  4.757893  5.351651  8.502745  6.766754  5.189567  6.523472  3.361598
## 29  3.693418  4.192573  7.470368  3.645923  5.262414  4.308444  4.328127
## 30  4.648042  5.394827  8.606098  4.953320  5.436914  5.486880  3.542364
## 31  3.160290  3.254589  5.553490  4.441132  5.648533  4.930296  5.309333
## 32  2.811631  3.379090  5.903335  3.835889  5.026755  4.347275  4.503736
## 33  7.624647  7.304970 10.102097  6.355009  8.049327  7.239359  6.584390
## 35  4.301514  3.364455  5.666744  4.633855  4.641169  2.940620  4.838989
## 36  4.132339  2.699082  4.978120  4.547480  5.007542  2.748840  4.327920
## 37  4.436928  3.654206  6.269528  4.278690  5.447119  3.306740  5.520259
## 38  5.447870  3.821349  5.069688  6.381371  4.217837  4.866969  5.825795
## 39  2.895164  2.531988  5.808869  4.644610  3.568658  3.433998  3.731875
## 40  3.569230  3.020171  5.769281  5.308954  3.075805  4.263139  4.478436
## 41  4.151176  2.943900  4.577360  5.330005  4.116325  3.324528  4.492803
## 42  3.884611  3.386091  5.760613  5.633309  4.109893  3.826399  3.673618
## 43  2.871869  3.371623  6.805743  3.855413  4.214211  3.329879  2.811385
## 44  2.413685  2.885890  6.441182  3.894717  3.156034  3.272842  3.029136
## 45  3.857159  3.316534  6.475958  5.443089  3.921987  3.961724  1.860942
## 46  3.242254  1.989810  4.690267  3.828834  4.479043  1.488838  4.136255
## 47  4.645354  4.170630  6.529140  5.280063  5.018477  4.306098  3.595260
## 48  3.133568  2.731863  6.098850  4.362741  3.912212  3.223233  3.090582
## 49  4.482443  2.876979  4.787959  4.927683  5.267218  3.313808  4.514870
## 50  3.833096  4.828637  8.023552  5.551785  4.164628  5.666704  3.242884
## 51  3.019325  3.789540  7.480596  3.834879  5.347591  4.985766  4.904864
## 52  3.798502  3.973080  6.915642  4.416747  4.274866  3.670651  4.060623
## 53  3.119070  3.247407  7.125260  2.859861  4.398096  3.030035  3.266459
##           22        23        24        25        26        27        28
## 2
## 3
## 4
## 5
## 6
## 7
## 8
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23  2.213532
## 24  3.840309  3.434284
## 25  3.265719  3.029727  2.572389
## 26  3.477927  4.343777  2.898563  3.544047
## 27  2.761157  2.891200  3.726783  3.068082  4.256676
## 28  3.824356  3.921810  2.036189  2.828656  3.073167  4.384136
## 29  1.876920  3.177001  4.158972  3.793601  3.204659  3.139035  4.660738
## 30  3.549381  4.082386  3.396926  3.166282  3.317997  4.173834  3.522730
## 31  4.429060  5.290250  5.436355  5.056892  4.484733  3.535611  5.808585
## 32  3.243954  4.347021  4.871046  4.594119  3.849974  3.030636  5.304259
## 33  6.713998  7.649943  6.736793  6.392562  4.983393  7.632172  6.941484
## 35  4.981485  4.349167  5.197349  5.163644  5.571358  3.968000  6.348674
## 36  5.978966  5.513829  4.995193  5.147524  5.281046  4.465312  6.294925
## 37  5.668595  4.989222  5.826787  5.861012  5.441090  5.567132  6.867186
## 38  6.684851  6.599363  6.550137  7.000806  5.897480  6.162454  7.101553
## 39  4.887129  4.207605  4.265110  3.871614  5.088509  2.940358  5.014178
## 40  4.790980  4.219668  4.657287  4.421865  5.187836  3.592708  4.952261
## 41  5.698043  5.032953  5.039889  5.064892  5.632652  4.430666  6.113586
## 42  4.862421  3.776286  3.696363  4.257815  5.025598  3.721796  4.881539
## 43  3.328533  2.831287  3.093855  3.244577  3.650007  2.764629  4.116482
## 44  4.105990  3.669117  4.020501  3.766052  4.249358  3.174056  4.602964
## 45  4.646412  4.357465  2.777803  3.637178  3.721950  3.335185  3.992300
## 46  4.917419  4.404826  4.870346  5.011968  5.102547  3.623768  6.090869
## 47  5.194369  5.363720  4.540625  5.594605  4.185452  4.589197  5.565937
## 48  4.202646  3.441619  3.147613  3.073414  4.022299  3.274694  4.071194
## 49  5.974759  5.868735  5.248412  5.557586  4.808325  5.114955  6.497018
## 50  3.211227  3.666673  3.256127  3.400951  3.403207  3.714110  2.496225
## 51  2.680010  3.333181  4.229523  3.461016  3.281244  3.240645  4.262088
## 52  4.900162  4.166969  4.516791  4.149120  5.380635  3.425343  5.466690
## 53  3.503009  3.737348  3.969159  3.606267  3.268037  3.423945  4.708054
##           29        30        31        32        33        35        36
## 2
## 3
## 4
## 5
## 6
## 7
## 8
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24
## 25
## 26
## 27
## 28
## 29
## 30  3.240490
## 31  4.274040  5.651256
## 32  2.865831  4.477484  2.054112
## 33  5.712233  5.063068  7.653591  6.951973
## 35  4.411720  5.806261  5.693797  5.034530  7.981665
## 36  5.290117  5.643692  4.941994  4.939953  7.261181  3.388697
## 37  4.924529  5.933317  6.133053  5.503199  6.840306  3.585227  3.686069
## 38  6.270773  7.288224  6.196243  6.030517  8.377506  4.300189  4.214772
## 39  5.057225  5.237391  4.635798  4.687650  8.131362  3.555036  2.970301
## 40  5.142801  5.627900  5.194067  5.130611  8.428034  3.429134  3.949753
## 41  5.380618  5.989343  5.239961  5.031645  8.147545  3.041553  2.126208
## 42  4.793536  5.169500  5.569704  5.057351  8.304259  2.592077  3.208208
## 43  2.951820  2.910071  4.820500  3.730812  6.510569  3.345907  3.727746
## 44  4.148112  3.973582  4.784047  4.034166  7.016051  3.889348  3.653798
## 45  4.430089  4.239483  5.006537  4.549625  7.128331  3.729677  3.235370
## 46  4.453585  5.572590  4.737585  4.263924  7.597745  2.116420  2.241453
## 47  4.291878  4.547146  5.450318  4.423164  6.926529  4.059671  4.185155
## 48  4.184633  4.002224  4.976448  4.604791  6.686254  3.342030  2.766110
## 49  5.114772  5.921548  4.718881  4.579388  6.667033  4.040270  2.096682
## 50  3.993331  2.845961  5.285079  4.365251  6.846859  5.884587  5.934348
## 51  2.801949  3.966191  3.674111  3.431082  6.291767  5.130929  5.421562
## 52  4.496075  4.254487  5.521144  4.886525  7.687342  3.355942  3.329408
## 53  2.857558  3.163268  4.706541  3.736524  5.100050  4.059726  3.893843
##           37        38        39        40        41        42        43
## 2
## 3
## 4
## 5
## 6
## 7
## 8
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24
## 25
## 26
## 27
## 28
## 29
## 30
## 31
## 32
## 33
## 35
## 36
## 37
## 38  4.327240
## 39  4.788386  4.698545
## 40  4.865996  3.955637  1.919643
## 41  3.993093  3.874772  2.927654  3.377843
## 42  4.311497  4.563486  2.768366  2.851892  2.356911
## 43  3.974120  5.437497  3.215252  3.689420  3.972693  2.862407
## 44  4.057349  4.810692  2.390371  3.020476  3.824368  3.414656  2.072334
## 45  5.095943  4.772750  2.715511  3.360643  3.428573  2.549371  2.693686
## 46  3.119478  3.874337  2.804970  3.331285  2.421532  2.801493  3.095745
## 47  4.818392  4.810341  4.674480  4.876292  4.831538  4.050480  3.008130
## 48  3.781473  4.686570  2.114276  2.615490  2.823120  2.351232  2.291610
## 49  3.577906  4.071849  4.300724  4.981230  2.751510  4.071256  4.375568
## 50  6.290095  6.472786  4.439437  4.391403  5.843308  4.837233  3.244766
## 51  5.111251  6.385676  4.534561  4.539723  5.743228  5.116927  3.521971
## 52  4.579498  5.557284  2.718905  3.473651  3.529051  2.843620  2.301461
## 53  3.657320  5.269916  3.758298  4.279060  4.552198  4.328177  2.179450
##           44        45        46        47        48        49        50
## 2
## 3
## 4
## 5
## 6
## 7
## 8
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24
## 25
## 26
## 27
## 28
## 29
## 30
## 31
## 32
## 33
## 35
## 36
## 37
## 38
## 39
## 40
## 41
## 42
## 43
## 44
## 45  2.975407
## 46  2.968307  3.293781
## 47  3.701504  3.119652  3.750977
## 48  2.322807  2.481138  2.726631  4.222648
## 49  4.400758  3.892023  2.872590  4.371491  3.733993
## 50  3.236699  3.898268  5.405761  4.673489  3.843152  6.311159
## 51  4.025155  4.816583  4.896783  5.231469  3.984466  5.597918  3.892627
## 52  2.756647  3.172846  3.407754  4.028024  2.702675  4.756026  4.573799
## 53  2.291076  3.477142  3.332273  3.639227  2.825323  4.105006  3.651382
##           51        52
## 2
## 3
## 4
## 5
## 6
## 7
## 8
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24
## 25
## 26
## 27
## 28
## 29
## 30
## 31
## 32
## 33
## 35
## 36
## 37
## 38
## 39
## 40
## 41
## 42
## 43
## 44
## 45
## 46
## 47
## 48
## 49
## 50
## 51
## 52  4.762600
## 53  3.348797  3.550557
```

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 test code
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: 0.055
##
## Upper quantiles of permutations (null model):
##   90%   95% 97.5%   99%
## 0.123 0.163 0.200 0.226
## 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

```
#dbRDA code
ponds.dbrda<-vegan::dbrda(dist.uf~.,data=as.data.frame(scale(envs)))
#permutation test
anova(ponds.dbrda,by="axis")
```

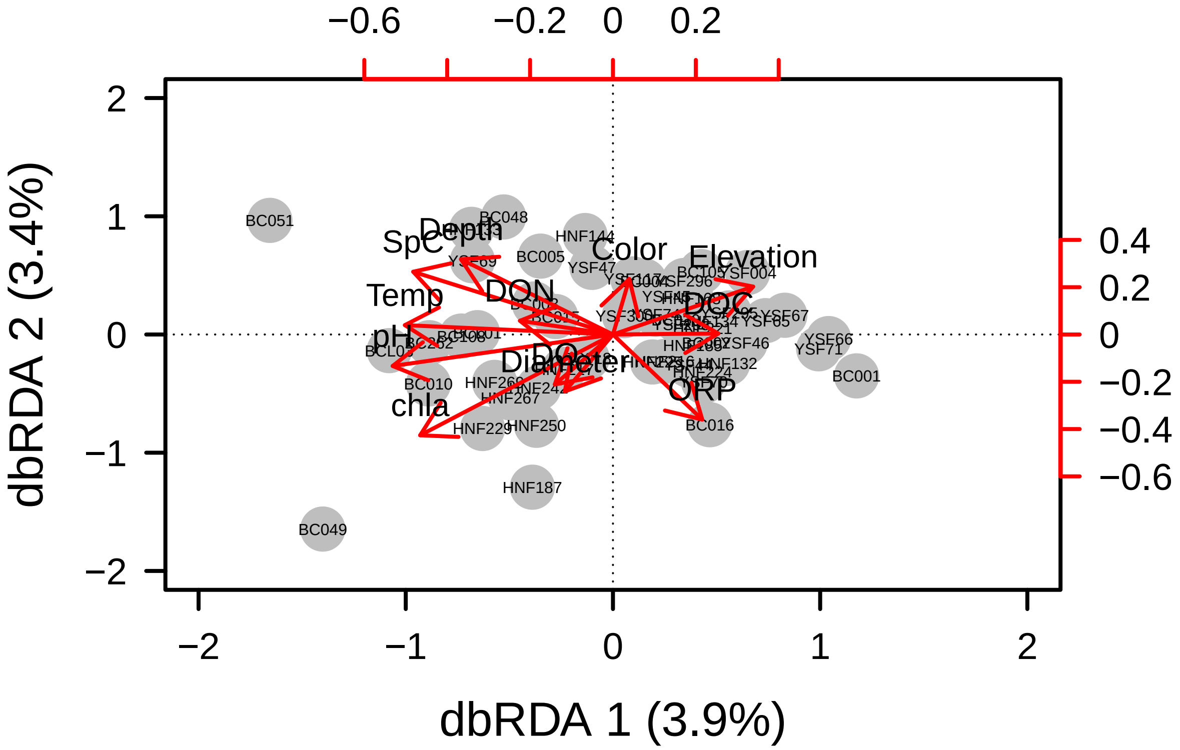
```
## Permutation test for dbrda under reduced model
## Forward tests for axes
## Permutation: free
## Number of permutations: 999
##
## Model: vegan::dbrda(formula = dist.uf ~ Elevation + Diameter + Depth + ORP + Temp + SpC + DO + pH +
##          Df SumOfSqs      F Pr(>F)
## dbRDA1    1  0.10566 2.0152  0.445
## dbRDA2    1  0.09258 1.7658  0.640
## dbRDA3    1  0.07555 1.4409  0.961
## dbRDA4    1  0.06677 1.2735  0.996
## 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  0.999
## 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  0.094 .
## Diameter  -0.27972 -0.96008 0.0541  0.260
## Depth     -0.63137  0.77548 0.1756  0.003 **
## ORP        0.41879 -0.90808 0.1437  0.015 *
## Temp      -0.98250  0.18628 0.1523  0.019 *
## SpC       -0.77101  0.63682 0.2087  0.004 **
## DO        -0.39318 -0.91946 0.0464  0.299
## pH        -0.96210 -0.27270 0.1756  0.009 **
## Color      0.06353  0.99798 0.0464  0.275
## chla      -0.60392 -0.79704 0.2626  0.013 *
## DOC        0.99847 -0.05526 0.0382  0.372
## DON       -0.91633  0.40042 0.0339  0.434
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## Permutation: free
## Number of permutations: 999
```

```
#finding explained variation
dbrda.ev1<-round(ponds.dbrda$CCA$eig[1]/sum(c(ponds.dbrda$CCA$eig,
                                              ponds.dbrda$CA$eig)),3)*100
dbrda.ev2<-round(ponds.dbrda$CCA$eig[2]/sum(c(ponds.dbrda$CCA$eig,
                                              ponds.dbrda$CA$eig)),3)*100
#plotting dbRDA
#plot parameters
par(mar=c(5,5,4,4)+0.1)
#starting plot
plot(scores(ponds.dbrda,display="wa"),xlim=c(-2,2),ylim=c(-2,2),
     xlab=paste("dbRDA 1 (",dbrda.ev1,"%)",sep=""),
     ylab=paste("dbRDA 2 (",dbrda.ev2,"%)",sep=""),
     pch=16,cex=2.0,type="n",cex.lab=1.5,cex.axis=1.2,axes=FALSE)
#adding axes back in
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)
#adding points and 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)
#adding environmental vectors
vectors<-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,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***: The hypothesis testing procedures above broadly allow us to conclude that bacterial beta diversity is influenced by specific characteristics of the pond environment. Via the PERMANOVA analysis, we found that specific watersheds are associated with the occurrance of phylogenetically similar bacterial communities. Watershed alone explained about 4.9% of the variation in community structure ($R^2 = 0.0492$, $F_{2,49} = 1.2679$, $p = 0.024$). Further analysis using a Mantel test allowed us to see how bacterial communities with high phylogenetic similarity were broadly correlated with environmental variables. We found that these communities with higher phylogenetic similarity were somewhat correlated with the environmental variables at each pond site, but that this relationship was not the strongest ($r = 0.1604$, sig $= 0.06$). However, conducting the distance-based redundancy analysis allowed for a better look at what specific environmental variables were associated with bacterial community assembly in these ponds. Bacterial

community assembly was specifically driven by six different environmental variables: chla (R^2 = 0.2626, p = 0.010), SpC (R^2 = 0.2087, p = 0.002), pond pH (R^2 = 0.1756, p = 0.008), pond depth (R^2 = 0.1756, p = 0.010), pond temperature (R^2 = 0.1523, p = 0.018), and ORP (R^2 = 0.1437, p = 0.26). Based on the results of all of these multivariate procedures, I would conclude that bacteria of higher phylogenetic similarity tend to exist together in communities within the same watershed *and* that their ability to live in a certain location is dependent on that pond's depth, temperature, pH, ORP, SpC, and chla.

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.

```
#finding geographic distances between ponds (in km)
long.lat<-as.matrix(cbind(pond.env$long,pond.env$lat))
cord.dist<-earth.dist(long.lat,dist=TRUE)
#finding taxonomic similarity of pond communities (Bray-Curtis)
bc.dist<-1-vegdist(otucomm)
#finding phylogenetic similarity of pond communities (UniFrac)
uf.dist<-1-dist.uf
#making dataframe with all info created above
#changing distances to pairwise long format using melt()
uf.dist.mlt<-melt(as.matrix(uf.dist))[melt(upper.tri(as.matrix(uf.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
```

```
bc.dist.mlt<-melt(as.matrix(bc.dist))[melt(upper.tri(as.matrix(bc.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
```

```
coord.dist.mlt<-melt(as.matrix(cord.dist))[melt(upper.tri(as.matrix(cord.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
```

```
#making data frame from distance lists
df<-data.frame(coord.dist.mlt,bc.dist.mlt[,3],uf.dist.mlt[,3],env.dist.mlt[,3])
names(df)[3:6]<-c("geo.dist","bray.curtis","unifrac","envv.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.

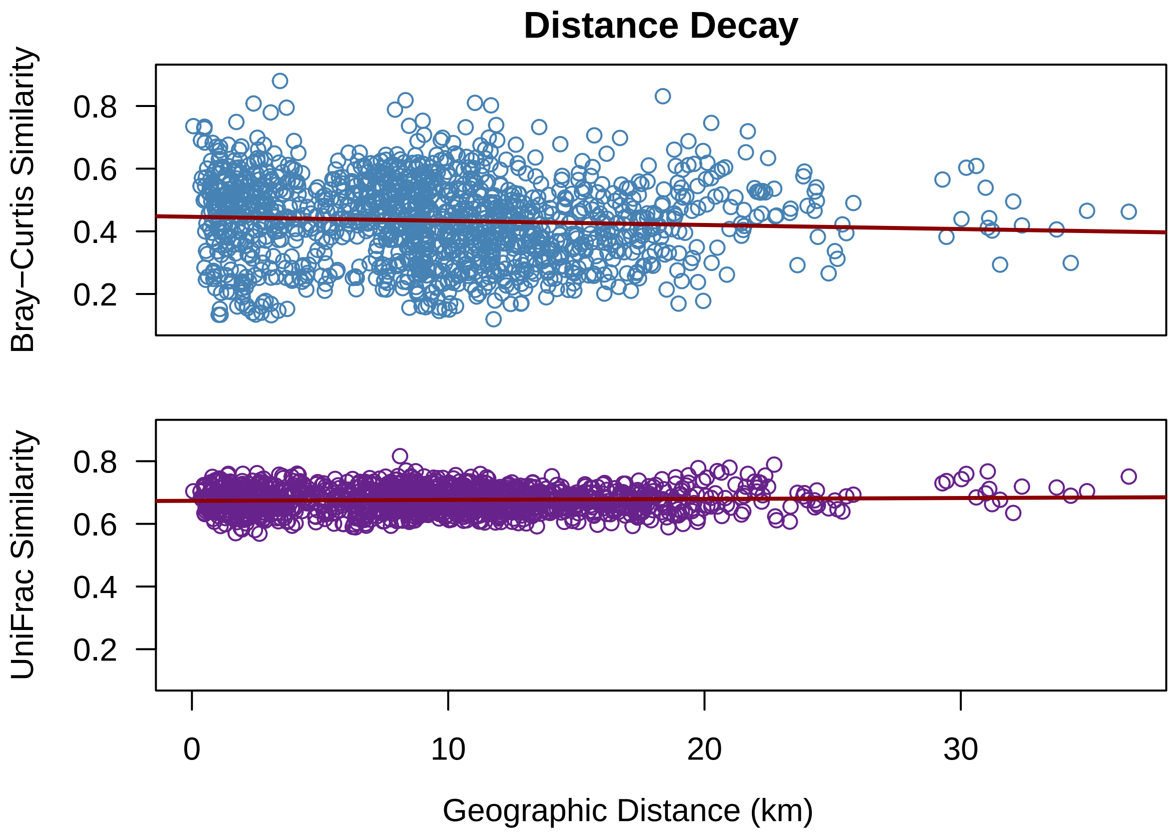
```
#plot parameters
par(mfrow=c(2,1),mar=c(1,5,2,1)+0.1,oma=c(2,0,0,0))
#making plot for taxonomic DD
plot(df$geo.dist,df$bray.curtis,xlab="",xaxt="n",las=1,ylim=c(0.1,0.9),
     ylab="Bray-Curtis Similarity",main="Distance Decay",col="steelblue")
#regression for taxonomic DD
dd.reg.bc<-lm(df$bray.curtis~df$geo.dist)
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)
#making 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.uf<-lm(df$unifrac~df$geo.dist)
summary(dd.reg.uf)
```

```
##
## 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.uf,col="red4",lwd=2)
#adding 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.

```r
#sourcing in diffslope code
source("./bin/diffslope.R")
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] 0.004
##
## $permutations
## [1] 1000
##
## $perms
##    [1] -4.576944e-04  2.869000e-04 -6.898065e-04 -5.175626e-04 -1.130491e-03
##    [6] -3.744020e-04  3.458314e-04  8.635090e-04 -6.276038e-04  4.020076e-04
##   [11]  7.862293e-04  9.926354e-04 -4.987979e-05 -1.533424e-04 -6.284171e-04
##   [16] -1.322716e-04 -6.448164e-04 -4.111585e-04 -4.047506e-04  3.915257e-04
##   [21] -1.055308e-03  3.061068e-04  5.237115e-04 -2.402006e-04 -1.349673e-03
##   [26]  3.413948e-04 -2.033297e-04  4.590321e-04 -6.661080e-04 -3.691407e-04
##   [31]  2.510460e-04  1.121897e-03  6.223892e-04 -3.257022e-04  2.059666e-04
##   [36]  6.908473e-04 -9.629286e-05  1.300065e-04 -2.620853e-04  3.166730e-04
##   [41]  5.412542e-04  3.089089e-04  1.221627e-03 -1.033555e-04  2.716733e-04
##   [46] -2.922048e-04 -2.456743e-04  5.110916e-04  4.274724e-04  1.636233e-04
##   [51] -2.127622e-04  7.850659e-05 -1.099013e-03 -8.166714e-05  2.386017e-04
##   [56] -1.082617e-04 -7.204098e-04  6.406057e-05 -4.647032e-04  8.072217e-04
##   [61]  2.532905e-04 -1.820534e-04 -1.317418e-04  1.679715e-04  5.713458e-04
##   [66]  1.147785e-03  3.338237e-04  5.121299e-04  2.395120e-04  1.065132e-03
##   [71] -2.175259e-04 -1.010075e-03 -6.207641e-04 -9.585401e-04  1.147445e-03
##   [76]  4.422145e-04 -1.121569e-04  9.085755e-04 -6.948609e-04  3.489944e-04
##   [81]  8.816086e-05 -2.354331e-04  2.379017e-04  9.123922e-04  3.347459e-04
##   [86]  2.869197e-05  3.276785e-04  2.782058e-04  3.644103e-04  6.616394e-05
##   [91]  4.056393e-04 -9.023652e-06  4.722934e-04  6.442234e-05 -3.474713e-05
##   [96] -1.008061e-03  5.960936e-04  1.809937e-04  7.953721e-05  1.267631e-03
##  [101] -7.534455e-04  8.883747e-06  4.359468e-04  8.393279e-05  4.351089e-04
##  [106] -1.066514e-03 -4.428362e-04  5.134935e-04 -4.045220e-04 -2.503801e-05
##  [111]  1.548120e-04 -6.292392e-04  4.959800e-04  1.931098e-04  1.944483e-04
##  [116] -5.964032e-05  5.361274e-04 -7.099350e-05 -3.266056e-04  5.877151e-04
##  [121] -4.934978e-04  4.425702e-05  2.313513e-04 -1.654010e-05  8.514630e-04
##  [126] -8.781104e-04 -1.198120e-03  2.139438e-04 -9.594664e-05  5.416308e-04
##  [131]  5.363025e-04  5.021253e-04  3.925073e-04  6.667802e-04  1.187291e-04
##  [136]  8.978965e-04 -5.554453e-04  5.482319e-05 -1.052242e-04  9.858413e-05
##  [141] -1.754134e-04  1.967045e-04 -1.889576e-04 -1.916889e-04  9.621309e-04
##  [146]  2.471427e-04  4.731151e-04 -3.674334e-05 -6.647459e-06 -1.294897e-04
##  [151] -3.389774e-04 -4.750959e-04  3.092485e-04  6.126060e-04 -2.077434e-05
##  [156] -4.853864e-04 -3.732943e-04  7.946502e-04  5.957734e-05 -4.400546e-05
##  [161]  4.354892e-04 -5.882939e-05  7.794745e-04  1.181775e-04  3.834782e-04
##  [166] -7.618057e-05 -2.877085e-04  6.225492e-05  1.030577e-03 -3.330414e-04
##  [171]  1.155115e-03 -8.632963e-04  2.475983e-04 -1.915057e-04  6.477941e-04
##  [176] -2.224538e-04  3.927628e-04  2.421318e-04 -8.999436e-04  2.255312e-04
##  [181]  1.194731e-03 -4.940609e-04  1.214342e-04  4.551210e-04  1.246224e-03
##  [186] -4.753591e-05  4.140925e-04 -5.532936e-04  3.988371e-04 -6.754232e-04
##  [191] -3.564533e-04 -6.155473e-04 -3.654676e-04  4.342698e-04  2.947614e-04
##  [196] -2.495019e-04  2.036161e-04 -1.137204e-03  9.682561e-04 -6.785196e-04
##  [201]  2.418259e-04 -1.577537e-04  3.378615e-04  1.200629e-03  7.283635e-04
##  [206] -1.069964e-04 -4.872195e-04 -4.302461e-05  9.343598e-04  1.348068e-04
##  [211] -1.607071e-04  4.790585e-04  3.414613e-04  2.079723e-04 -4.953162e-04
##  [216] -9.292658e-05  8.513356e-04  4.478227e-04 -3.216390e-04 -4.565768e-04
##  [221] -8.373630e-04 -2.986687e-04  5.783720e-04 -7.719912e-04  7.090346e-04
##  [226]  8.195330e-04 -1.864766e-04  8.545967e-04  5.688193e-04 -4.030285e-04
##  [231] -2.506672e-04 -1.089054e-03 -6.886951e-04 -3.035138e-04  1.609638e-04
##  [236]  4.570557e-04  6.880503e-04 -1.191030e-04 -3.340949e-04 -6.741154e-04
##  [241] -1.377473e-05  1.201113e-03  1.111059e-03  8.165936e-04 -1.237315e-04
##  [246] -3.849926e-04  8.078602e-04  3.355094e-04 -8.560775e-04  6.286298e-04
##  [251] -7.609533e-04 -1.764162e-04 -4.944114e-04  1.303403e-04  2.611930e-04
##  [256]  3.099258e-04  9.143999e-05 -3.059347e-04  7.949696e-04  4.740376e-04
##  [261] -6.203669e-04 -1.524970e-03 -6.205269e-05  2.480699e-04  5.701915e-05
##  [266]  2.897047e-04  9.901741e-04 -5.627621e-04 -4.273124e-04  3.532483e-04
##  [271]  3.557313e-04  3.369530e-05  9.864684e-04  9.769711e-04  2.438661e-04
##  [276] -8.452043e-04  2.561342e-05 -2.112096e-04  2.130850e-04  4.679283e-05
##  [281]  7.783010e-04  3.666247e-04 -6.105640e-04 -4.379408e-04  7.620107e-04
##  [286]  6.066554e-04  1.009474e-04 -7.385635e-04  2.307421e-05  8.366040e-05
##  [291]  2.834594e-04  4.738750e-04  2.085555e-04 -1.025310e-05 -9.713622e-04
## [296] -2.003633e-04 -6.120296e-05  3.077433e-05 -2.332290e-04  1.931241e-05
## [301] -1.343852e-04 -6.112010e-05 -1.041630e-03  4.454700e-04 -1.696116e-04
## [306] -1.010640e-04 -7.877458e-05 -2.816389e-04 -2.862007e-04  1.225504e-04
## [311] -7.873672e-04  4.512834e-04 -9.420005e-05 -7.994739e-04  3.651593e-04
## [316] -1.575580e-04  2.647165e-04 -5.885354e-04 -1.085661e-03 -6.861784e-04
## [321] -9.219605e-04  3.365173e-04 -3.443926e-04  8.140609e-04 -1.347770e-04
## [326] -2.848099e-04 -4.076855e-04  3.061352e-04  8.059211e-04  2.212683e-04
## [331]  6.610850e-04 -3.970742e-04 -8.260157e-04  1.585926e-04 -6.818221e-04
## [336]  7.367316e-04 -4.477521e-04  2.081138e-04  5.281805e-04  1.535378e-04
## [341] -9.767846e-04 -3.733266e-04  2.899909e-04  8.201924e-05 -1.817472e-04
## [346]  3.794094e-04 -3.503536e-04  1.260986e-03 -3.811756e-04 -7.116318e-04
## [351]  6.794261e-04  1.181766e-04 -7.774100e-04  1.887997e-04 -3.087411e-04
## [356]  4.420049e-04 -1.485379e-05 -5.379894e-06 -5.785042e-05 -7.633490e-04
## [361]  2.696230e-04 -1.886373e-04  8.505324e-04  2.010515e-04 -4.245136e-04
## [366] -1.191994e-04  7.233413e-04 -3.384697e-05  7.338954e-04 -1.683131e-04
## [371] -4.445498e-04 -7.167980e-04  3.773135e-04 -7.961694e-04 -3.241146e-04
## [376]  1.331102e-04 -1.001250e-04 -2.639237e-04 -6.952319e-04  1.946268e-04
## [381]  8.125547e-04  4.539008e-04 -1.940055e-04  7.459457e-04 -2.417742e-04
## [386]  7.595749e-04  7.752548e-04 -1.031374e-04  6.577449e-04  8.734291e-05
## [391]  6.906949e-04 -2.267278e-04 -2.270384e-04  1.890789e-04 -9.309880e-04
## [396] -2.045830e-04  2.515387e-04  5.563766e-05  2.812566e-04  9.546010e-04
## [401]  8.857217e-04  6.533362e-04 -3.735041e-04 -3.308534e-04  8.190647e-05
## [406] -6.667933e-04 -5.399519e-05 -7.838255e-04 -8.629858e-04  2.086162e-05
## [411]  9.742957e-05 -8.153625e-05 -4.450405e-05 -1.259218e-04 -7.292638e-04
## [416]  3.492304e-04  2.443079e-04 -6.807436e-04 -2.401882e-04  1.662934e-04
## [421] -3.600401e-04 -2.120871e-05  3.274535e-04 -5.758162e-04 -5.629842e-04
## [426] -3.561103e-04 -9.977149e-05 -5.320268e-04 -4.506253e-04 -6.132559e-04
## [431]  2.208039e-04  4.235952e-05  7.332900e-05  9.414017e-04 -3.152188e-04
## [436] -4.331002e-04  3.257645e-04  3.210094e-04 -1.070171e-04  7.591565e-04
## [441]  4.189983e-04  9.458742e-05 -1.001158e-03 -7.495507e-05  5.548785e-04
## [446]  5.292333e-04  4.036637e-04 -1.334284e-03  4.227126e-07 -8.284002e-04
## [451]  9.419206e-04 -3.783339e-04 -2.577740e-04 -5.274320e-05 -4.801097e-04
## [456]  4.935841e-04 -1.104722e-03 -6.075404e-05 -5.310209e-05 -3.340026e-04
## [461]  4.697881e-04 -1.914165e-04 -7.620982e-04  2.969275e-04  5.044932e-06
## [466]  7.685998e-04 -7.890644e-04  5.909463e-04  8.020911e-04 -3.211734e-05
## [471] -1.739230e-03  3.815426e-04 -2.112748e-04  1.350042e-03  9.425557e-04
## [476]  1.156354e-04  8.649212e-05 -1.750061e-04 -6.321502e-05  6.917267e-05
## [481]  9.503815e-04  4.514925e-04  4.062030e-04  1.620146e-03  5.931340e-05
## [486]  2.882627e-04 -1.526903e-03 -2.023166e-03 -3.782071e-04  2.114121e-04
## [491] -9.317337e-04 -5.612988e-04  9.763998e-04 -3.475908e-04 -2.303333e-05
## [496]  1.485383e-05 -2.759649e-04  1.144082e-03  6.701577e-04  5.700815e-05
## [501]  3.557233e-04 -1.010330e-03 -2.503220e-05  3.338149e-04 -7.055860e-04
## [506]  5.149586e-04  1.082567e-04  6.298711e-04  3.249853e-04  1.456070e-03
## [511] -1.253032e-04  2.160438e-04  9.718061e-05 -2.517112e-04  1.078894e-03
## [516]  2.354550e-04 -1.826925e-04  1.230280e-03  1.131021e-03  4.266272e-04
## [521]  4.925943e-04 -2.132514e-04  4.386303e-05  2.436864e-04  4.187141e-04
## [526] -4.723873e-04  1.189659e-04  1.194016e-03 -8.125747e-04 -2.549285e-04
## [531] -7.570228e-04  5.091921e-04 -1.429360e-03  1.247266e-04  6.861222e-04
## [536] -1.322949e-04 -6.636247e-07  3.020814e-04 -4.587058e-04 -9.183764e-05
## [541]  3.573231e-04 -2.458580e-04  8.248976e-04  2.715333e-04  6.639983e-04
## [546]  3.885149e-05 -3.865837e-05  1.346052e-03  4.996604e-06  8.115046e-04
## [551] -4.116197e-04  4.539500e-04  1.870657e-04  1.993627e-04  5.903680e-05
## [556] -4.897385e-05 -2.502876e-07  4.372779e-04 -1.455341e-04  2.471363e-04
## [561]  4.383310e-04 -8.827769e-04 -1.181048e-04  8.998276e-04 -8.054333e-05
## [566] -1.633353e-04  8.602696e-04  1.658747e-03 -4.519615e-04 -4.011395e-04
## [571] -6.528864e-04  1.145721e-04 -2.864700e-05  9.238007e-04 -6.285637e-04
## [576] -7.254663e-04  1.331796e-04  2.173091e-03 -1.975812e-04 -2.108793e-05
## [581] -1.277299e-05 -2.888871e-04 -1.295004e-04  9.932262e-04  1.839049e-04
## [586]  3.323900e-04 -9.795743e-04  3.676668e-05  8.187701e-04  1.085970e-04
## [591] -4.896064e-04  8.355765e-04 -8.736064e-04 -8.218639e-04  9.829778e-04
## [596] -4.243342e-04 -1.284905e-03  4.758823e-04 -8.599508e-04 -1.132633e-03
## [601]  2.695116e-04 -1.127325e-05  2.236891e-04  3.328470e-04  2.133855e-04
## [606]  6.807683e-04  1.378110e-04  3.037120e-04 -7.342798e-04 -7.301634e-04
## [611]  3.749792e-05 -1.196055e-04 -2.010181e-04 -2.962669e-04 -1.039357e-03
## [616] -9.905433e-05  6.306842e-04 -2.498339e-04 -6.101996e-04  1.608670e-05
## [621] -6.184044e-04  3.074013e-04  1.944747e-03 -6.952201e-04  4.788690e-04
## [626] -2.912411e-04 -3.814275e-04 -6.366756e-04 -1.033708e-03 -7.316434e-04
## [631] -6.608431e-04  2.919801e-05 -8.700332e-04 -1.161502e-03  3.267409e-05
## [636]  4.094089e-04  1.566640e-04 -1.134172e-03  1.055188e-03 -5.941046e-04
## [641] -1.210119e-03  1.114277e-04 -9.399523e-04  3.626700e-04  2.119289e-05
## [646] -4.443461e-05 -1.450821e-04  2.990499e-04 -2.360087e-04 -1.307258e-05
## [651] -5.260666e-05  1.924339e-04  1.110259e-03 -3.755490e-04  3.109738e-04
## [656] -1.133387e-04 -1.810299e-04  7.231906e-04 -6.187725e-04 -1.748221e-04
## [661]  3.362855e-04  8.557797e-04  8.381829e-04 -9.683058e-05  7.730120e-04
## [666] -6.540408e-04  5.475768e-04 -4.645910e-05 -8.311939e-04 -1.611831e-03
## [671] -3.287972e-04  8.071223e-04  8.489299e-04 -4.337813e-04 -8.927685e-04
## [676]  5.320281e-04  3.990067e-04  6.492164e-04  3.737133e-04 -5.209631e-06
## [681]  9.223801e-04 -4.972634e-04 -5.359702e-05 -1.063689e-03  5.624694e-04
## [686]  4.755569e-04  1.234376e-03 -8.932165e-04 -2.538225e-04 -4.561811e-04
## [691]  6.654662e-05  9.349139e-04  2.109251e-04  5.142613e-04 -2.626217e-04
## [696] -3.853601e-04  3.284940e-05 -4.394499e-05 -2.413483e-04 -2.410605e-04
## [701] -3.036805e-04  4.997389e-04 -4.585601e-04  1.507761e-04  1.154693e-04
## [706] -1.647784e-04  4.359922e-04 -7.088721e-04 -1.000171e-03  2.382600e-04
## [711]  1.992008e-04  6.178690e-04 -1.376518e-04  8.403611e-04 -8.507558e-04
## [716] -1.539924e-04 -3.581133e-05 -6.237065e-04  1.528103e-04  4.575740e-05
## [721]  3.528114e-04  9.378262e-05 -2.484682e-04 -2.369493e-04 -9.654628e-05
## [726] -9.273663e-04 -5.657874e-04 -1.130915e-03 -8.820836e-04  8.792892e-04
## [731]  3.295914e-04  4.960494e-04  7.296001e-04 -7.628663e-04 -5.114828e-04
## [736]  6.041507e-04 -1.146518e-03 -1.236277e-03 -6.263404e-04  1.065633e-04
## [741]  9.087693e-05 -1.136034e-03 -1.073999e-03 -2.166303e-03 -2.096917e-04
## [746] -6.377070e-04 -2.163966e-04  8.293998e-04  6.449354e-04  6.582047e-04
## [751]  7.484107e-05 -7.553743e-05  7.938866e-04  8.540370e-04  5.335663e-07
## [756] -1.060794e-04  9.073764e-04 -5.743127e-04 -4.005608e-04 -4.757886e-04
## [761]  3.611138e-04 -5.494567e-04  4.783850e-04 -1.441650e-04  3.095333e-04
## [766]  6.018932e-04 -3.933676e-04  1.332812e-04  1.731262e-04  5.921389e-04
## [771]  5.262263e-04  5.514228e-04  4.557231e-04 -4.255842e-04 -8.053997e-04
## [776] -8.440309e-04 -6.279551e-04  3.480711e-04  6.426449e-05  1.361861e-04
## [781] -5.022969e-04  5.521372e-04  7.502694e-04  6.440098e-04 -1.246307e-04
## [786] -5.565294e-04  7.055232e-04  1.282534e-05  2.949298e-04  5.383421e-04
## [791] -8.857719e-04  4.547987e-04 -1.750224e-04 -5.872232e-04  2.724450e-04
## [796]  1.166073e-03 -1.875138e-04  5.287033e-04  6.131874e-04  2.192232e-04
## [801]  7.278606e-04  3.286016e-04  8.426340e-04  4.595844e-04 -3.399849e-05
## [806]  4.768933e-04 -7.378977e-04  2.204080e-04 -1.091918e-03 -4.481637e-04
## [811]  3.391889e-05 -2.628254e-04  3.213196e-04 -2.923721e-04  8.829289e-04
## [816]  8.553882e-04  9.446209e-05 -4.443969e-04  4.323669e-04 -4.258558e-04
## [821]  3.779084e-04  5.320264e-04  2.698294e-04  1.013352e-05  9.067271e-04
## [826]  1.628545e-04 -1.348825e-04  4.310434e-04 -6.980240e-04  2.738712e-04
## [831]  7.674779e-04  2.504795e-04  1.077324e-03  9.826164e-05  6.146587e-04
## [836]  6.776124e-04  5.803883e-05 -1.191911e-03  4.598952e-04  4.285334e-04
## [841]  1.087175e-04 -1.129920e-03  4.505286e-04  1.006035e-03 -2.176004e-04
## [846]  4.337831e-04 -2.692604e-04  8.973057e-04 -3.992376e-04  3.537979e-04
## [851] -2.072166e-04  5.510729e-04 -1.687350e-03  4.955986e-04 -8.109105e-04
## [856]  4.407539e-05 -1.678189e-03  5.478939e-04  9.637112e-04 -1.001822e-03
## [861]  4.491759e-04  8.368411e-04 -2.640291e-04 -1.053189e-05 -6.610065e-04
## [866] -4.873817e-04  8.199825e-04  1.144782e-03 -3.234081e-04 -2.495635e-04
## [871] -5.478676e-04 -2.308036e-04 -8.274263e-04  1.690536e-04  2.304754e-04
## [876] -4.496091e-04  5.342966e-04 -2.308168e-04 -8.908438e-04 -3.552914e-04
## [881] -5.182260e-04  1.310053e-03  1.813433e-04  1.670539e-04  3.593571e-04
## [886] -3.979593e-04  6.081434e-04 -6.990032e-06  2.009539e-04  5.157666e-05
## [891]  6.696588e-05  7.772024e-04 -1.391365e-04  3.877869e-04 -8.894740e-05
## [896] -5.056065e-04 -2.374392e-04 -9.711971e-04  4.734410e-06  5.634908e-07
## [901]  1.504239e-03 -2.844912e-04  2.263214e-04 -7.817123e-04 -3.447785e-04
## [906]  3.966885e-04  3.762855e-04 -9.750060e-04 -7.149601e-04 -4.591860e-04
## [911]  2.362001e-04  3.310327e-04  1.184293e-04 -1.672540e-04  8.555206e-05
## [916] -2.048832e-04  2.724397e-04 -5.341562e-04 -1.246930e-04 -3.754983e-04
## [921] -2.313906e-04 -3.464810e-04  4.148927e-04  7.995408e-04  2.882852e-04
## [926] -5.155157e-04 -1.227152e-04  1.835633e-04  4.822368e-05 -3.988270e-04
## [931] -1.691389e-04  9.618275e-04 -2.445078e-04 -5.632828e-04  4.365825e-04
## [936]  1.111318e-03  1.802209e-05  1.336760e-04  1.679491e-04 -7.280408e-05
## [941] -2.967393e-04  6.813805e-04 -1.510734e-04  5.600410e-04  4.866397e-04
## [946]  8.147913e-04 -1.258897e-03 -4.777117e-04  9.095225e-04  1.199471e-04
## [951] -4.746722e-04  4.452332e-04 -8.636057e-04 -6.184673e-04  7.363218e-05
## [956] -3.741646e-04 -1.258474e-03 -3.174967e-04  1.228943e-04  4.343581e-04
## [961]  1.927245e-04 -1.981062e-04  3.482474e-04  6.506720e-05 -4.058772e-04
## [966] -7.049729e-04  3.926130e-05  2.506913e-04  8.147760e-04 -1.966115e-03
## [971] -2.640331e-04  2.961837e-04 -2.773672e-04  7.815550e-04  6.731162e-04
## [976]  3.268986e-04 -4.164639e-04  1.289935e-03 -2.418968e-04  6.509478e-06
## [981]  7.880846e-05 -9.287546e-05  7.326892e-04  5.832658e-04 -4.762876e-04
## [986]  1.333371e-04 -1.814616e-04  2.727296e-05  4.293104e-05  1.824712e-04
## [991]  2.256790e-04 -7.025683e-04  1.363880e-03 -5.397688e-04  9.685497e-04
## [996]  1.174728e-03 -6.091583e-04 -1.870464e-04 -6.025773e-04  4.285487e-04
##
## 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***: The taxonomic distance-decay has a slope of -0.0013, indicating a very slight negative relationship between geographic distance and Bray-Curtis dissimilarity. This means that as the physical distance between two bacterial taxa increases, their taxonomic relatedness increases very slightly (given that in Bray-Curtis, 0 is a measure of identical communities). The phylogenetic distance-decay has a slope of 0.0003, which shows a very slight positive relationship between geographic distance and UniFrac similarity. Since UniFrac represents the proportion of branch length between two taxa that isn't shared, this slope indicates that increasing distance between bacterial taxa results in a slight decrease in phylogenetic relatedness. As a result of permutation testing with diffslope(), we are able to conclude that these slopes are not equivalent despite their apparent similarity (slope diff. = 0.0016, signif. = 0.005). The difference between these slopes might be explained by local introgression in bacteria that are slightly less taxonomically similar. Bacteria living in the same environment have more opportunities to share genetic information with one another via methods like conjugation, so as a result OTUs exisiting within the same pond may have very similar genetic information and would have higher phylogenetic similarity

regardless of their taxonomies. Bacteria that live further apart in different ponds are not able to share genetic information in this way, so even taxonomically similar OTUs that do not occur in the same environments might not be as closely related in a phylogenetic sense.

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.

> Although I am still working out the details of my first research project, I plan on studying the stability-diversity relationship in plant communities and how it relates to environmental change. Specifically, I hope to look at how homogenization of beta diversity impacts the stability of plant communities responding to disturbance. This homogenization can take several different forms, but phylogenetic information could be useful for assessing genetic and functional homogenization (these are reviewed very nicely in Olden et al. 2004, "Ecological and evolutionary consequences of biotic homogenization"). The phylogenetic data needed to address how genetic and functional homogenization occur with anthropogenic change would ideally be collected from similar species across several locations and over several years (genetic info could potentially be collected from herbarium samples as well to achieve this). I think it would be most useful to collect genetic information related to functional traits (such as aboveground biomass and surface leaf area) to evaluate both genetic and functional homogenization. It would be ideal to collect this information for several plant species (since I will likely be working with Midwestern prairie communities, this could include several grasses and forbs) as well to see whether homogenization occurs similarly across species. With this data, performing distance decay analyses for different traits within each species across multiple years could be used to see the rate at which traits are homogenizing. In an analysis like this, I would expect homogenization to cause species to be more phylogenetically similar across distances with time (due to increasing human impacts). Ordinations like PCoA or dbRDA could also potentially be used to see how different environmental variables (temperature, precipitation, soil nutrient content, etc.) correlate with genetic or functional homogenization in the prairie communities. This could provide valuable insight into how anthropogenic environmental change is affecting biodiversity in important communities such as prairies.