

8. Worksheet: Phylogenetic Diversity - Traits

El Park; Z620: Quantitative Biodiversity, Indiana University

05 March, 2025

OVERVIEW

Up to this point, we have been focusing on patterns taxonomic diversity in Quantitative Biodiversity. Although taxonomic diversity is an important dimension of biodiversity, it is often necessary to consider the evolutionary history or relatedness of species. The goal of this exercise is to introduce basic concepts of phylogenetic diversity.

After completing this exercise you will be able to:

1. create phylogenetic trees to view evolutionary relationships from sequence data
2. map functional traits onto phylogenetic trees to visualize the distribution of traits with respect to evolutionary history
3. test for phylogenetic signal within trait distributions and trait-based 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, **push** this file to your GitHub repo.
6. For the assignment portion of the worksheet, follow the directions at the bottom of this file.
7. When you are done, **Knit** the text and code into a PDF file.
8. After Knitting, submit the completed exercise by creating a **pull request** via GitHub. Your pull request should include this file `PhyloTraits_Worskheet.Rmd` and the PDF output of Knitr (`PhyloTraits_Worskheet.pdf`).

The completed exercise is due on **Wednesday, February 26th, 2025 before 12:00 PM (noon)**.

1) SETUP

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 `Week6-PhyloTraits/` folder, and
4. load all of the required R packages (be sure to install if needed).

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

```
## [1] "/cloud/project/QB2025_Park/Week6-PhyloTraits"
setwd("/cloud/project/QB2025_Park/Week6-PhyloTraits")
package.list<-c('ape', 'seqinr', 'phylobase', 'adephylo', 'geiger',
  'picante', 'stats', 'RColorBrewer', 'caper', 'phylolm', 'pmc',
  'ggplot2', 'tidyr', 'dplyr', 'phangorn', 'pander', 'phytools', 'vegan', 'cluster', 'dendexts')
for (package in package.list) {
  if (!require(package, character.only=TRUE, quietly=TRUE)) {
    install.packages(package)
    library(package, character.only=TRUE)
  }
}
```

```
##
## Attaching package: 'seqinr'

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

## The following object is masked from 'package:ape':
##
##   edges
##
## Attaching package: 'phytools'

## The following object is masked from 'package:phylobase':
##
##   readNexus
##
## Attaching package: 'permute'

## The following object is masked from 'package:seqinr':
##
##   getType
##
## Attaching package: 'vegan'

## The following object is masked from 'package:phytools':
##
##   scores
##
## Attaching package: 'nlme'

## The following object is masked from 'package:seqinr':
##
##   gls
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:MASS':
##
##   select
```

```

## The following object is masked from 'package:nlme':
##
## collapse
## The following object is masked from 'package:seqinr':
##
## count
## The following object is masked from 'package:ape':
##
## where
## The following objects are masked from 'package:stats':
##
## filter, lag
## The following objects are masked from 'package:base':
##
## intersect, setdiff, setequal, union
##
## Attaching package: 'phangorn'
## The following objects are masked from 'package:vegan':
##
## diversity, treedist
##
## Attaching package: 'cluster'
## The following object is masked from 'package:maps':
##
## votes.repub
## Registered S3 method overwritten by 'dendextend':
## method from
## rev.hclust vegan
##
## -----
## Welcome to dendextend version 1.19.0
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
## https://stackoverflow.com/questions/tagged/dendextend
##
## To suppress this message use: suppressPackageStartupMessages(library(dendextend))
## -----
##
## Attaching package: 'dendextend'
## The following object is masked from 'package:permute':
##
## shuffle

```

```

## The following object is masked from 'package:geiger':
##
##   is.phylo
## The following object is masked from 'package:phytools':
##
##   untangle
## The following objects are masked from 'package:phylobase':
##
##   labels<-, prune
## The following objects are masked from 'package:ape':
##
##   ladderize, rotate
## The following object is masked from 'package:stats':
##
##   cutree
##
## Attaching package: 'phylogram'
## The following object is masked from 'package:dendextend':
##
##   prune
## The following object is masked from 'package:phylobase':
##
##   prune
##
## Attaching package: 'amap'
## The following object is masked from 'package:vegan':
##
##   pca
##
## Attaching package: 'scales'
## The following object is masked from 'package:phytools':
##
##   rescale
## Warning in rgl.init(initValue, onlyNULL): RGL: unable to open X11 display
## Warning: 'rgl.init' failed, will use the null device.
## See '?rgl.useNULL' for ways to avoid this warning.
#BiocManager::install("Biostrings")
library(Biostrings)

## Loading required package: BiocGenerics
##
## Attaching package: 'BiocGenerics'
## The following objects are masked from 'package:dplyr':
##
##   combine, intersect, setdiff, union

```

```

## The following object is masked from 'package:ade4':
##
##     score
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
## The following objects are masked from 'package:base':
##
##     anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##     colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##     get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
##     match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##     Position, rank, rbind, Reduce, rownames, sapply, saveRDS, setdiff,
##     table, tapply, union, unique, unsplit, which.max, which.min
## Loading required package: S4Vectors
## Loading required package: stats4
##
## Attaching package: 'S4Vectors'
## The following objects are masked from 'package:dplyr':
##
##     first, rename
## The following object is masked from 'package:tidyr':
##
##     expand
## The following object is masked from 'package:utils':
##
##     findMatches
## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname
## Loading required package: IRanges
##
## Attaching package: 'IRanges'
## The following objects are masked from 'package:dplyr':
##
##     collapse, desc, slice
## The following object is masked from 'package:nlme':
##
##     collapse
## Loading required package: XVector
## Found more than one class "Annotated" in cache; using the first, from namespace 'RNeXML'
## Also defined by 'S4Vectors'
## Found more than one class "Annotated" in cache; using the first, from namespace 'RNeXML'
## Also defined by 'S4Vectors'

```

```

## Found more than one class "Annotated" in cache; using the first, from namespace 'RNeXML'
## Also defined by 'S4Vectors'
## Found more than one class "Annotated" in cache; using the first, from namespace 'RNeXML'
## Also defined by 'S4Vectors'
## Found more than one class "Annotated" in cache; using the first, from namespace 'RNeXML'
## Also defined by 'S4Vectors'
## Found more than one class "Annotated" in cache; using the first, from namespace 'RNeXML'
## Also defined by 'S4Vectors'
## Loading required package: GenomeInfoDb
##
## Attaching package: 'Biostrings'
## The following object is masked from 'package:dendextend':
##
##     nnodes
## The following object is masked from 'package:seqinr':
##
##     translate
## The following object is masked from 'package:ape':
##
##     complement
## The following object is masked from 'package:base':
##
##     strsplit
library(msa)

##
## Attaching package: 'msa'
## The following object is masked from 'package:BiocManager':
##
##     version
#install.packages("pak")
#pak::pkg_install("msa")
#install.packages("libcurl4-openssl-dev - curl")
library(seqinr)

```

2) DESCRIPTION OF DATA

The maintenance of biodiversity is thought to be influenced by **trade-offs** among species in certain functional traits. One such trade-off involves the ability of a highly specialized species to perform exceptionally well on a particular resource compared to the performance of a generalist. In this exercise, we will take a phylogenetic approach to mapping phosphorus resource use onto a phylogenetic tree while testing for specialist-generalist trade-offs.

3) SEQUENCE ALIGNMENT

Question 1: Using your favorite text editor, compare the `p.isolates.fasta` file and the `p.isolates.afa` file. Describe the differences that you observe between the two files.

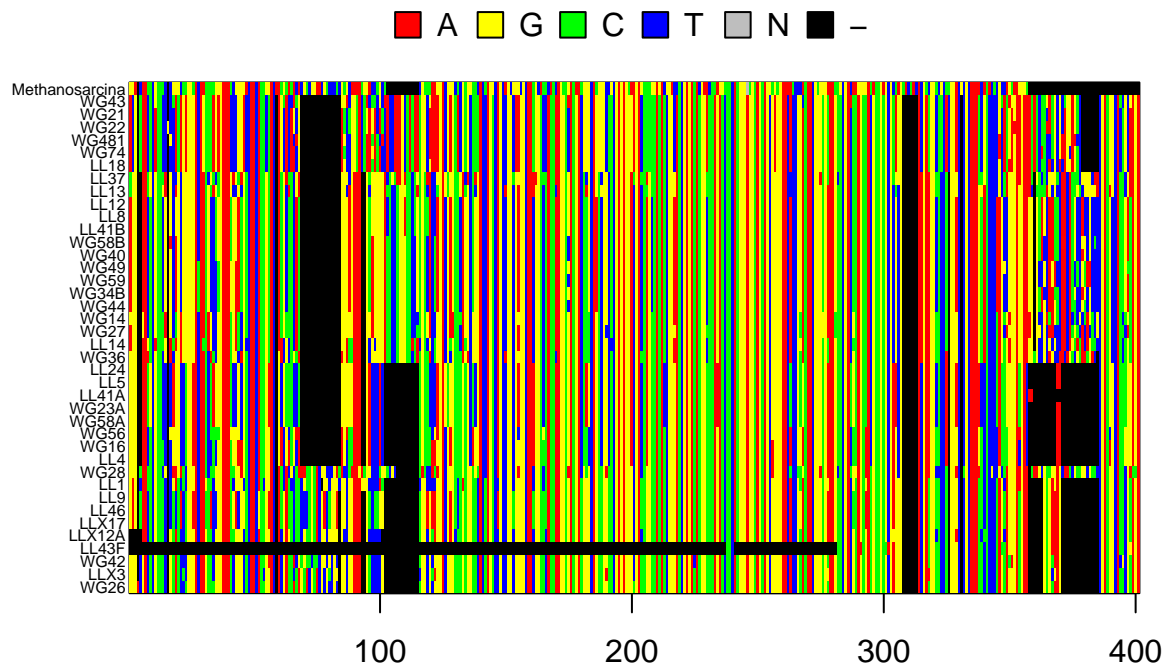
Answer 1: the afa file has a lot of missing information and seems to be general sequencing data and the fasta file seems cleaner with isolated genes?

In the R code chunk below, do the following: 1. read your alignment file, 2. convert the alignment to a DNABin object, 3. select a region of the gene to visualize (try various regions), and 4. plot the alignment using a grid to visualize rows of sequences.

```
seqs<-readDNAStringSet("data/p.isolates.fasta", format='fasta')
seqs
```

```
## DNABin object of length 40:
##      width seq                                     names
## [1]   619 ACACGTGAGCAATCTGCCCTTCT...TTCTCTGGGAATACCTGACGCT LL9
## [2]   597 CGGCAGCGGGAAGTAGCTTGCTA...AACTGTTTCAGCTAGAGTCTTGT WG14
## [3]   794 CAGCGCGGACGGGTGAGTAACA...GCTAACGCATTAAGCACTCCGC WG28
## [4]   716 CTTTCAGAGTTAGTGGCGGACGGG...TGCTAGTTGTCTGGGATGCATGC LL24
## [5]   803 ACGAACTCTTCGAGTTAGTGGC...TAAACTCAAAGGAATTGACGG LL41A
## ...   ...
## [36]  652 TTCGGGAGTACACGAGCGGCGAA...TTCTCTGGGAATACCTGACGCT LL46
## [37]  661 GCGAACGGGTGAGTAACACGTGG...GAGCGAAAGCGTGGGTAGCGAA WG26
## [38]  694 GCGAACGGGTGAGTAACACGTG...ACCCTGGTAGTCCACGCCGTAA WG42
## [39]  699 TACAGGTACCAGGCTCCTTCGGG...AAAGCATGGGTAGCGAACAGGA LLX17
## [40] 1426 TTCTGTTGATCCTGCCAGAGGT...AACCTNAATTTTGCAAGGGGGG Methanosarcina
```

```
read.aln<-msaMuscle(seqs)
save.aln<-msaConvert(read.aln, type="bios2mds::align")
export.fasta(save.aln, "./data/p.isolates.afa")
p.DNABin<-as.DNABin(read.aln)
window<-p.DNABin[, 100:500]
image.DNABin(window, cex.lab=0.50)
```



Question 2: Make some observations about the `muscle` alignment of the 16S rRNA gene sequences for our bacterial isolates and the outgroup, *Methanosarcina*, a member of the domain Archaea. Move along the alignment by changing the values in the `window` object.

- a. Approximately how long are our sequence reads?
- b. What regions do you think would be appropriate for phylogenetic inference and why?

Answer 2a: There are a little over 400 base pairs (bases 100 through 500)

Answer 2b: Between 150 and 300 seems ideal because most of it is conserved with specific areas of variation.

4) MAKING A PHYLOGENETIC TREE

Once you have aligned your sequences, the next step is to construct a phylogenetic tree. Not only is a phylogenetic tree effective for visualizing the evolutionary relationship among taxa, but as you will see later, the information that goes into a phylogenetic tree is needed for downstream analysis.

A. Neighbor Joining Trees

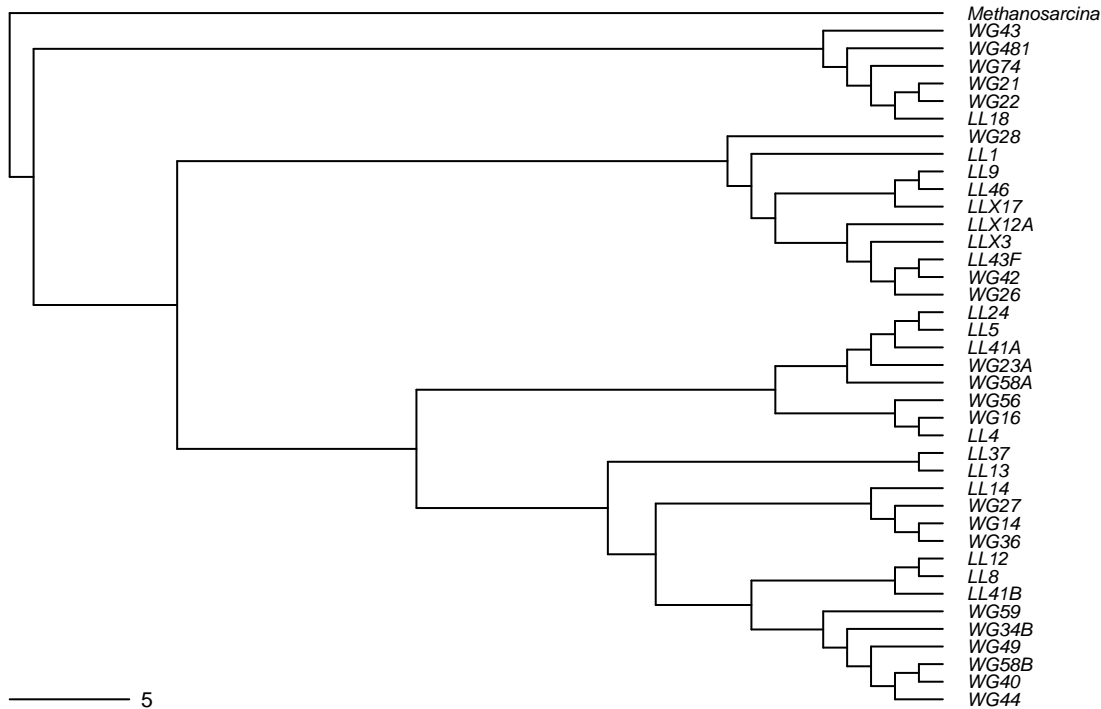
In the R code chunk below, do the following:

1. calculate the distance matrix using `model = "raw"`,
2. create a Neighbor Joining tree based on these distances,
3. define “*Methanosarcina*” as the outgroup and root the tree, and
4. plot the rooted tree.

```
seq.dist.raw<-dist.dna(p.DNABin, model="raw", pairwise.deletion=FALSE)
nj.tree<-bionj(seq.dist.raw)
outgroup<-match("Methanosarcina", nj.tree$tip.label)
nj.rooted<-root(nj.tree, outgroup, resolve.root=TRUE)

par(mar=c(1, 1, 2, 1) + 0.1)
plot.phylo(nj.rooted, main="Neighbor Joining Tree", "phylogram",
           use.edge.length=FALSE, direction="right", cex=0.6,
           label.offset=1)
add.scale.bar(cex=0.7)
```


Neighbor Joining Tree



Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

Answer 3: Advantages include ease of use/simplicity and accuracy, meaning it is relatively straightforward, does not assume a constant rate of evolution, and maintains better accuracy compared to more complex methods.

Disadvantages include loss of data since this method takes sequence alignments and interprets them as lengths of the branches. Another disadvantage is that only one tree can be made using this method, as outputting multiple trees is useful for hypothesis testing.

B) SUBSTITUTION MODELS OF DNA EVOLUTION

In the R code chunk below, do the following:

1. make a second distance matrix based on the Felsenstein 84 substitution model,
2. create a saturation plot to compare the *raw* and *Felsenstein (F84)* substitution models,
3. make Neighbor Joining trees for both, and
4. create a cophylogenetic plot to compare the topologies of the trees.

```
seq.dist.F84<-dist.dna(p.DNAbin, model="F84", pairwise.deletion=FALSE)
par(mar=c(5, 5, 2, 1)+0.1)
plot(seq.dist.raw, seq.dist.F84,
     pch=20, col="red", las=1, asp=1, clim=c(0, 0.7),
     ylim=c(0, 0.7), xlab="Raw Distance", ylab="F84 Distance")
```

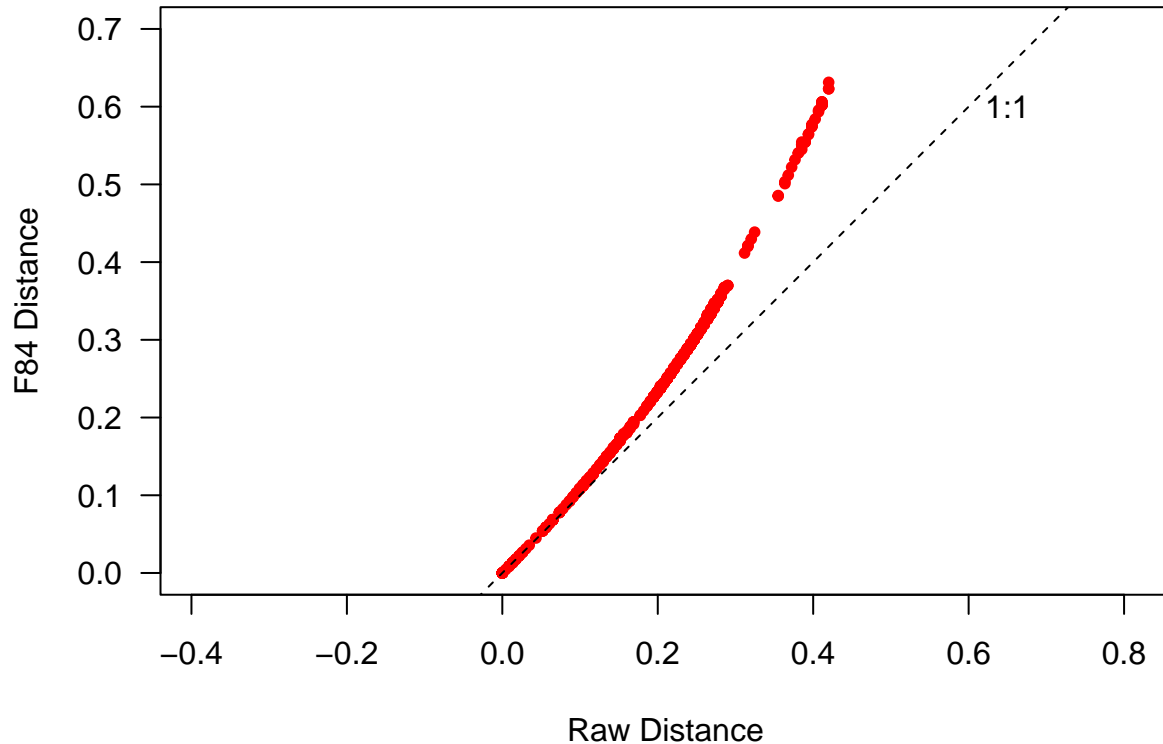
```
## Warning in plot.window(...): "clim" is not a graphical parameter
```

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

```
## Warning in axis(side = side, at = at, labels = labels, ...): "clim" is not a
## graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "clim" is not a
## graphical parameter
```

```
## Warning in box(...): "clim" is not a graphical parameter
## Warning in title(...): "clim" is not a graphical parameter
abline(b=1, a=0, lty=2)
text(0.65, 0.6, "1:1")
```

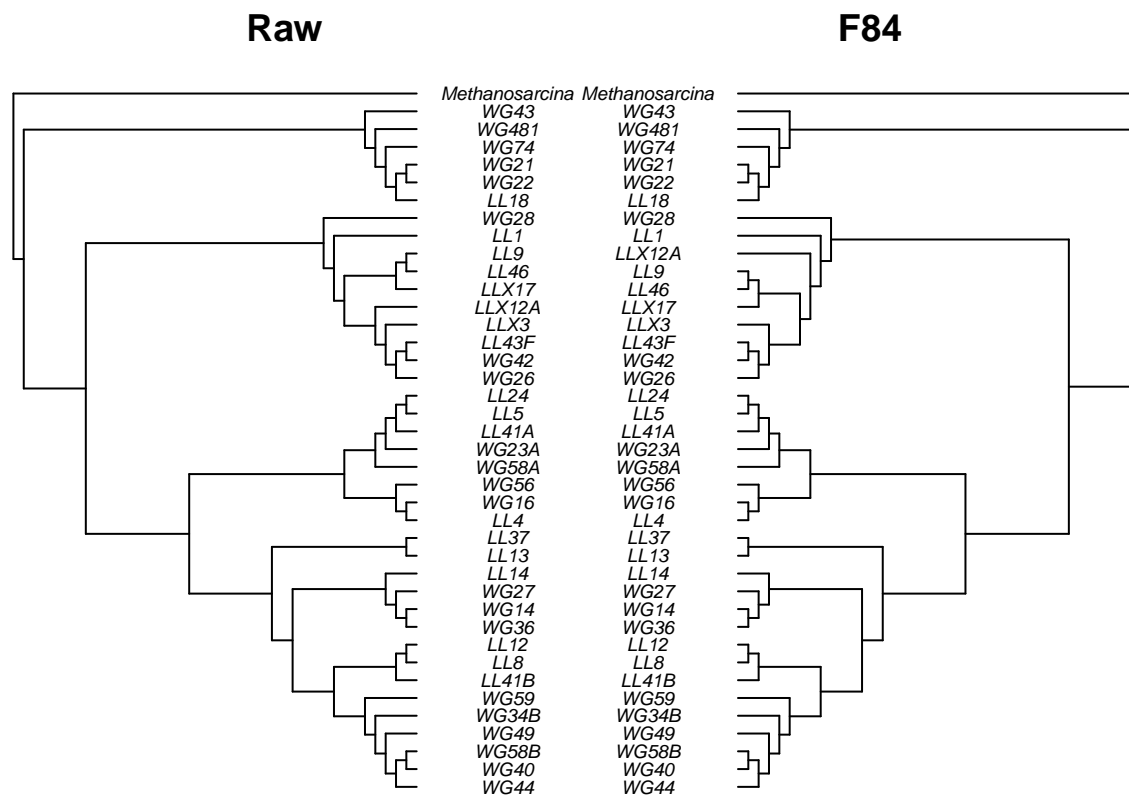


```
raw.tree<-bionj(seq.dist.raw)
F84.tree<-bionj(seq.dist.F84)

raw.outgroup<-match("Methanosarcina", raw.tree$tip.label)
F84.outgroup<-match("Methanosarcina", F84.tree$tip.label)

raw.rooted<-root(raw.tree, raw.outgroup, resolve.root=TRUE)
F84.rooted<-root(F84.tree, F84.outgroup, resolve.root=TRUE)

layout(matrix(c(1,2), 1, 2), width=c(1,1))
par(mar=c(1, 1, 2, 0))
plot.phylo(raw.rooted, type="phylogram", direction="right",
  show.tip.label=TRUE, use.edge.length=FALSE, adj=0.5,
  cex=0.6, label.offset=2, main="Raw")
par(mar=c(1, 0, 2, 1))
plot.phylo(F84.rooted, type="phylogram", direction="left",
  show.tip.label=TRUE, use.edge.length=FALSE, adj=0.5,
  cex=0.6, label.offset=2, main="F84")
```



```
?plot.phylo
```

C) ANALYZING A MAXIMUM LIKELIHOOD TREE

In the R code chunk below, do the following:

1. Read in the maximum likelihood phylogenetic tree used in the handout.
2. Plot bootstrap support values onto the tree

```
phytDat.aln<-msaConvert(read.aln, type="phangorn::phyDat")
aln.dist<-dist.ml(phytDat.aln)
aln.NJ<-NJ(aln.dist)
fit<-pml(tree=aln.NJ, data=phytDat.aln)
fitJC<-optim.pml(fit, TRUE)
```

```
## optimize edge weights: -10571.04 --> -10396.64
## optimize edge weights: -10396.64 --> -10396.64
## optimize topology: -10396.64 --> -10341.45 NNI moves: 10
## optimize edge weights: -10341.45 --> -10341.45
## optimize topology: -10341.45 --> -10341.45 NNI moves: 0
```

```
fitGTR<-optim.pml(fit, TRUE)
```

```
## optimize edge weights: -10571.04 --> -10396.64
## optimize edge weights: -10396.64 --> -10396.64
## optimize topology: -10396.64 --> -10341.45 NNI moves: 10
## optimize edge weights: -10341.45 --> -10341.45
## optimize topology: -10341.45 --> -10341.45 NNI moves: 0
```

```
anova(fitJC, fitGTR)
```

```
## Likelihood Ratio Test Table
```

```
## Log lik. Df Df change Diff log lik. Pr(>|Chi|)
## 1 -10342 77
## 2 -10342 77 0 0 1

AIC(fitJC)

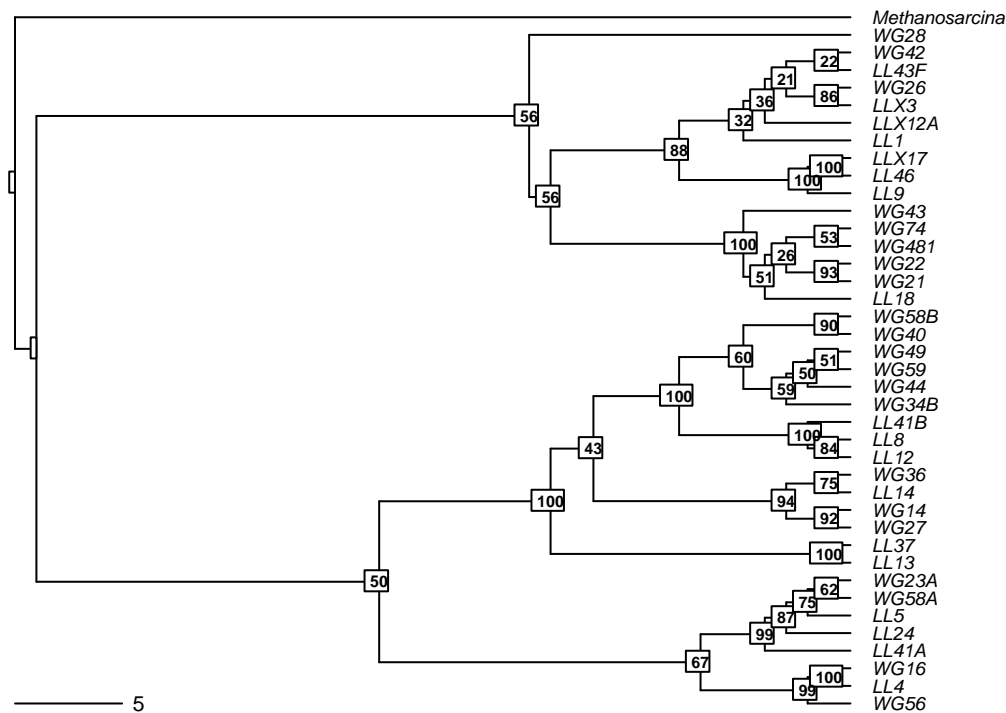
## [1] 20836.9

AIC(fitGTR)

## [1] 20836.9

ml.bootstrap<-read.tree("./data/ml_tree/RAxML_bipartitions.T1")
par(mar=c(1, 1, 2, 1)+0.1)
plot.phylo(ml.bootstrap, type="phylogram", direction="right",
  show.tip.label=TRUE, use.edge.length=FALSE, cex=0.6,
  label.offset=1, main="Maximum Likelihood with Support Values")
add.scale.bar(cex=0.7)
nodelabels(ml.bootstrap$node.label, font=2, bg = "white", frame="r", cex=0.5)
```

Maximum Likelihood with Support Values



Question 4:

- How does the maximum likelihood tree compare the to the neighbor-joining tree in the handout? If the plots seem to be inconsistent with one another, explain what gives rise to the differences.
- Why do we bootstrap our tree?
- What do the bootstrap values tell you?
- Which branches have very low support?
- Should we trust these branches? Why or why not?

Answer 4a: These two methods seem relatively similar, with the addition of confidence values

in the ML tree. Some inconsistencies include variation in relationships between certain organisms, which may be due to the way the ML tree assumes a specific model of evolution whereas the NJ tree uses a distance-based analysis.

Answer 4b: Bootstrapping allows us to make better inferences of the tree based on confidence values; ie. it allows us to see which branches are well supported vs branches that might have been based on data with a lot of background noise.

Answer 4c: The values tell you how confident the branches are. Anything lower than 50 is considered unreliable and the higher the number, the more confident.

Answer 4d: WG42 and LL43F and branches preceding these two sequences are very low.

Answer 4e: We should definitely take these with a grain of salt when interpreting these branches. It's not to say they are completely untrustworthy, but the data is most likely missing or messy.

5) INTEGRATING TRAITS AND PHYLOGENY

A. Loading Trait Database

In the R code chunk below, do the following:

1. import the raw phosphorus growth data, and
2. standardize the data for each strain by the sum of growth rates.

```
p.growth<-read.table("./data/p.isolates.raw.growth.txt", sep="\t",
                    header=TRUE, row.names=1)
p.growth.std<-p.growth/(apply(p.growth, 1, sum))
```

B. Trait Manipulations

In the R code chunk below, do the following:

1. calculate the maximum growth rate (μ_{max}) of each isolate across all phosphorus types,
2. create a function that calculates niche breadth (nb), and
3. use this function to calculate nb for each isolate.

```
umax<-apply(p.growth, 1, max)
levins<-function(p_xi=""){
  p=0
  for (i in p_xi){
    p=p+i^2
  }
  nb=1/(length(p_xi)*p)
  return(nb)
}

nb<-as.matrix(levins(p.growth.std))
nb<-setNames(as.vector(nb), as.matrix(row.names(p.growth)))
```

C. Visualizing Traits on Trees

In the R code chunk below, do the following:

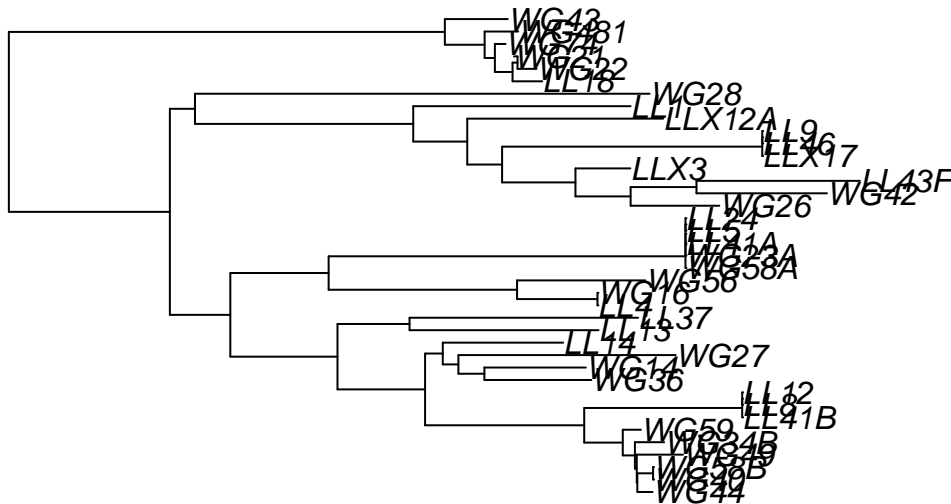
1. pick your favorite substitution model and make a Neighbor Joining tree,
2. define your outgroup and root the tree, and
3. remove the outgroup branch.

```
nj.tree<-bionj(seq.dist.F84)
outgroup<-match("Methanosarcina", nj.tree$tip.label)
nj.rooted<-root(nj.tree, outgroup, resolve.root = TRUE)
```

```

nj.rooted<-drop.tip(nj.rooted, "Methanosarcina")
plot(nj.rooted)

```



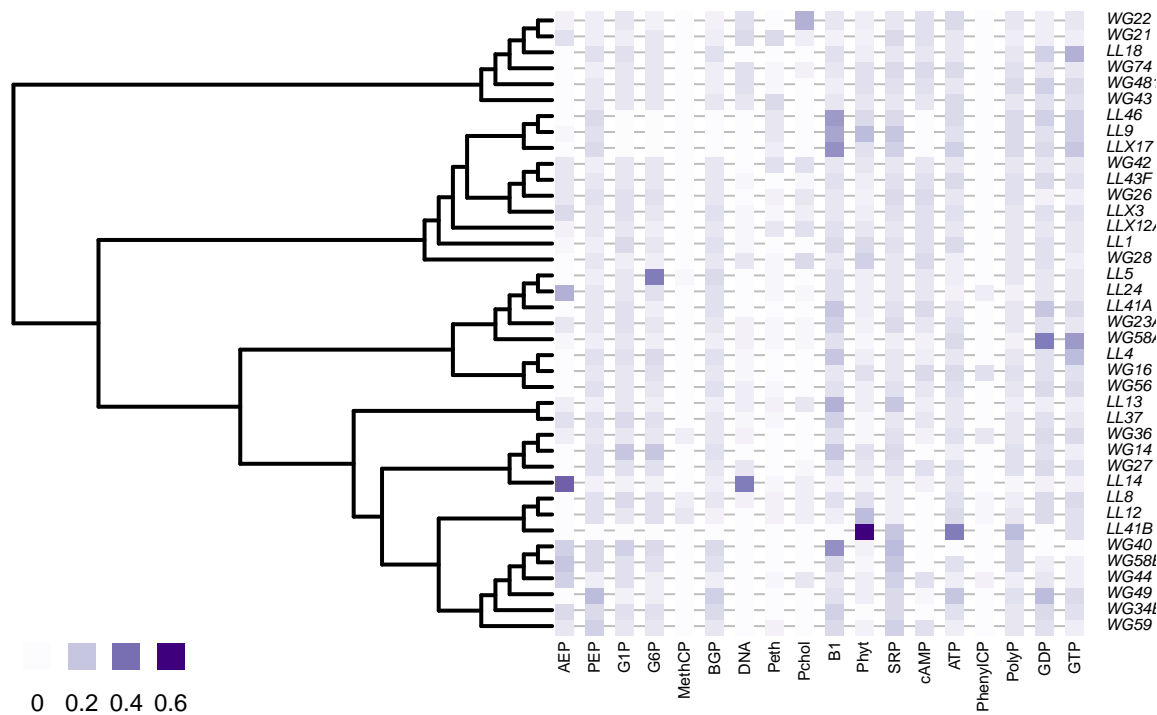
In the R code chunk below, do the following:

1. define a color palette (use something other than “YlOrRd”),
2. map the phosphorus traits onto your phylogeny,
3. map the *nb* trait on to your phylogeny, and
4. customize the plots as desired (use `help(table.phylo4d)` to learn about the options).

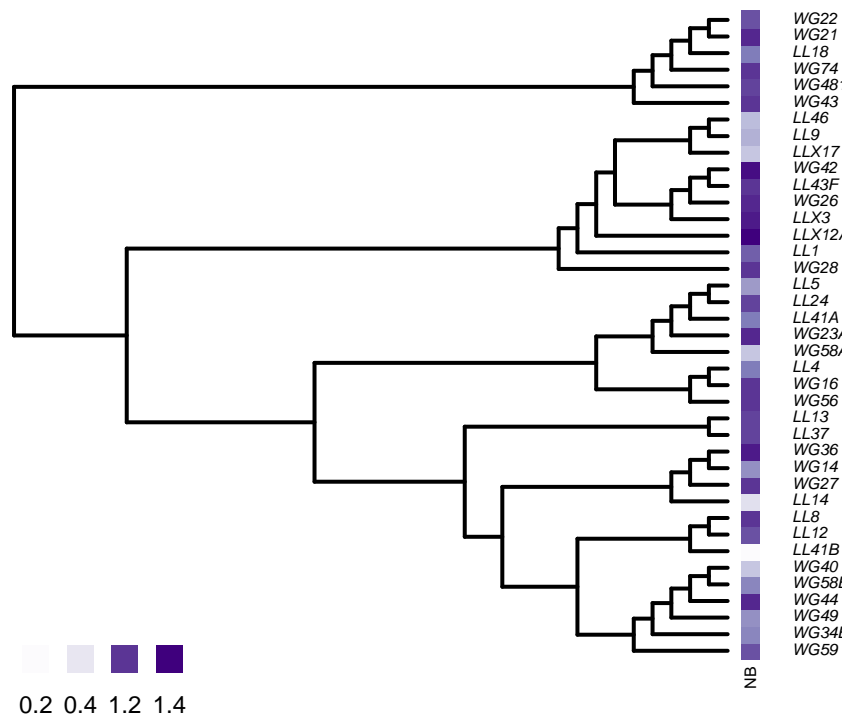
```

mypalette<-colorRampPalette(brewer.pal(9,"Purples"))
nj.plot<-nj.rooted
nj.plot$edge.length<-nj.plot$edge.length+10^-1
par(mar=c(1, 1, 1, 1)+0.1)
x<-phylo4d(nj.plot, p.growth.std)
table.phylo4d(x, treetype="phylo", symbol="colors", show.node=TRUE,
  cex.labe=0.5, scale=FALSE, use.edge.length=FALSE,
  edge.color="black", edge.width=2, box=FALSE,
  col=mypalette(25), pch=15, cex.symbol=1.25,
  ratio.tree=0.5, cex.legend=1.5, center=FALSE)

```



```
par(mar=c(1, 5, 1, 5) + 0.1)
x.nb<-phylo4d(nj.plot, nb)
table.phylo4d(x.nb, treetype="phylo", symbol="colors", show.node=TRUE,
  cex.label=0.5, scale=FALSE, use.edge.length=FALSE,
  edge.color="black", edge.width=2, box=FALSE, col=mypalette(25),
  pch= 15, cex.symbol = 1.25, var.label = ("NB"),ratio.tree = 0.90,
  cex.legend = 1.5, center = FALSE)
```



Question 5:

- Develop a hypothesis that would support a generalist-specialist trade-off.
- What kind of patterns would you expect to see from growth rate and niche breadth values that would support this hypothesis?

Answer 5a: Organisms that are generalists will have overall lower growth rates due to the cost of maintaining flexibility as opposed to specializing in specific environments.

Answer 5b: We would see low to medium growth rates across all of the niches (phosphorous sources) for generalists and little to no growth except for a few medium to high growth rates for specialists.

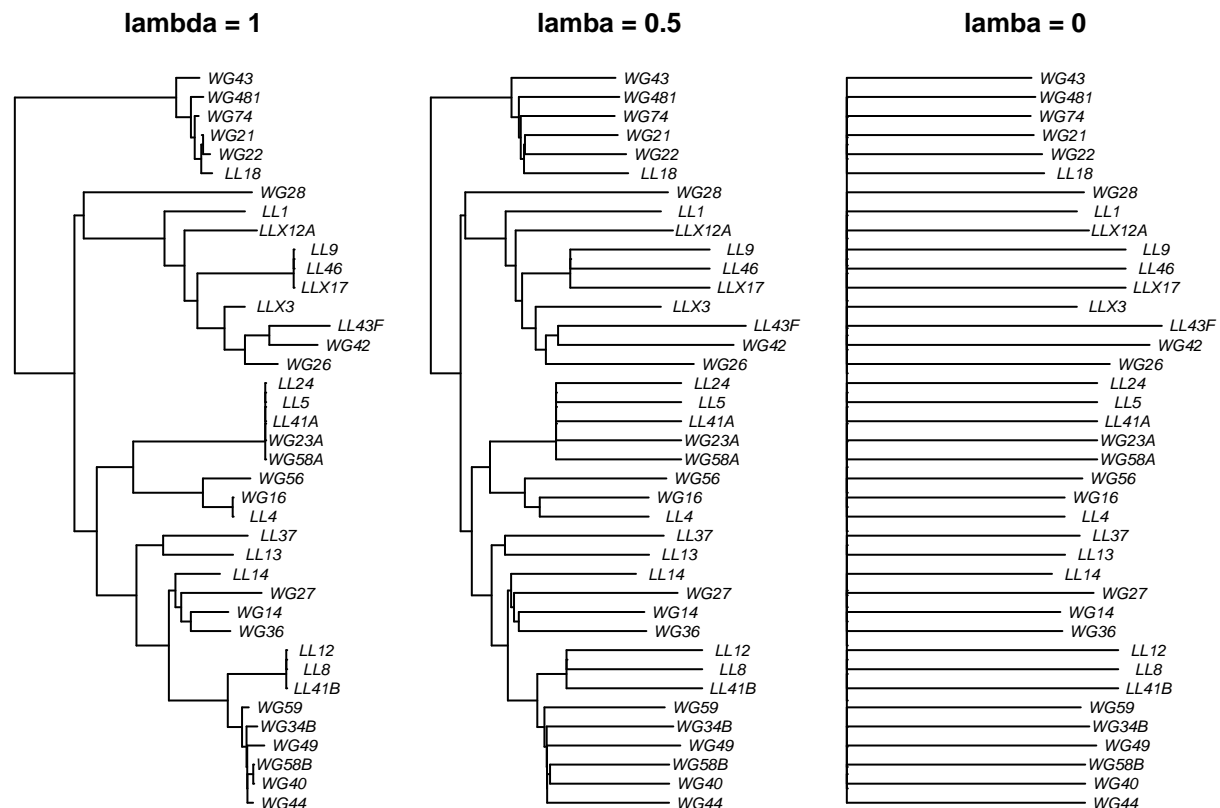
6) HYPOTHESIS TESTING

Phylogenetic Signal: Pagel's Lambda

In the R code chunk below, do the following:

- create two rescaled phylogenetic trees using lambda values of 0.5 and 0,
- plot your original tree and the two scaled trees, and
- label and customize the trees as desired.

```
library(geiger)
nj.lambda.5 <- geiger::rescale.phylo(nj.rooted, "lambda", 0.5)
nj.lambda.0 <- geiger::rescale.phylo(nj.rooted, "lambda", 0)
layout(matrix(c(1, 2, 3), 1, 3), width = c(1, 1, 1))
par(mar=c(1, 0.5, 2, 0.5) + 0.1)
plot(nj.rooted, main = "lambda = 1", cex = 0.7, adj = 0.5)
plot(nj.lambda.5, main = "lambda = 0.5", cex = 0.7, adj = 0.5)
plot(nj.lambda.0, main = "lambda = 0", cex = 0.7, adj = 0.5)
```



In the R code chunk below, do the following:

1. use the `fitContinuous()` function to compare your original tree to the transformed trees.

```
fitContinuous(nj.rooted, nb, model = "lambda")
```

```
## GEIGER-fitted comparative model of continuous data
## fitted 'lambda' model parameters:
## lambda = 0.000976
## sigsq = 0.108047
## z0 = 0.656654
##
## model summary:
## log-likelihood = 21.502742
## AIC = -37.005485
## AICc = -36.319770
## free parameters = 3
##
## Convergence diagnostics:
## optimization iterations = 100
## failed iterations = 55
## number of iterations with same best fit = NA
## frequency of best fit = NA
##
## object summary:
## 'lik' -- likelihood function
## 'bnd' -- bounds for likelihood search
## 'res' -- optimization iteration summary
## 'opt' -- maximum likelihood parameter estimates
```

```
fitContinuous(nj.lambda.0, nb, model = "lambda")
```

```
## GEIGER-fitted comparative model of continuous data
## fitted 'lambda' model parameters:
## lambda = 0.000000
## sigsq = 0.108048
## z0 = 0.656477
##
## model summary:
## log-likelihood = 21.502505
## AIC = -37.005010
## AICc = -36.319295
## free parameters = 3
##
## Convergence diagnostics:
## optimization iterations = 100
## failed iterations = 0
## number of iterations with same best fit = 92
## frequency of best fit = 0.920
##
## object summary:
## 'lik' -- likelihood function
## 'bnd' -- bounds for likelihood search
## 'res' -- optimization iteration summary
## 'opt' -- maximum likelihood parameter estimates
```

```
phylosig(nj.rooted, nb, method = "lambda", test = TRUE)
```

```
##
## Phylogenetic signal lambda : 0.00699105
## logL(lambda) : 21.5034
## LR(lambda=0) : 0.00181763
## P-value (based on LR test) : 0.965994
```

Question 6: There are two important outputs from the `fitContinuous()` function that can help you interpret the phylogenetic signal in trait data sets. a. Compare the lambda values of the untransformed tree to the transformed (lambda = 0). b. Compare the Akaike information criterion (AIC) scores of the two models. Which model would you choose based off of AIC score (remember the criteria that the difference in AIC values has to be at least 2)? c. Does this result suggest that there's phylogenetic signal?

Answer 6a: The untransformed tree is 0.006974, which is very close to the transformed tree (0)

Answer 6b: The rooted tree has an AIC of -37.006827 and the transformed tree is -37.005010. There is almost no difference between these, meaning the data is more similar to the transformed tree?

Answer 6c: No, these results suggest there is no phylogenetic signal as lambda and AIC values suggest a strong similarity between the untransformed and transformed tree and the p value is 0.965994 (so we fail to reject the null)

7) PHYLOGENETIC REGRESSION

Question 7: In the R code chunk below, do the following:

1. Clean the resource use dataset to perform a linear regression to test for differences in maximum growth rate by niche breadth and lake environment.
2. Fit a linear model to the trait dataset, examining the relationship between maximum growth rate by niche breadth and lake environment.
2. Fit a phylogenetic regression to the trait dataset, taking into account the bacterial phylogeny

```
nb.lake = as.data.frame(as.matrix(nb))
nb.lake$lake = rep("A")

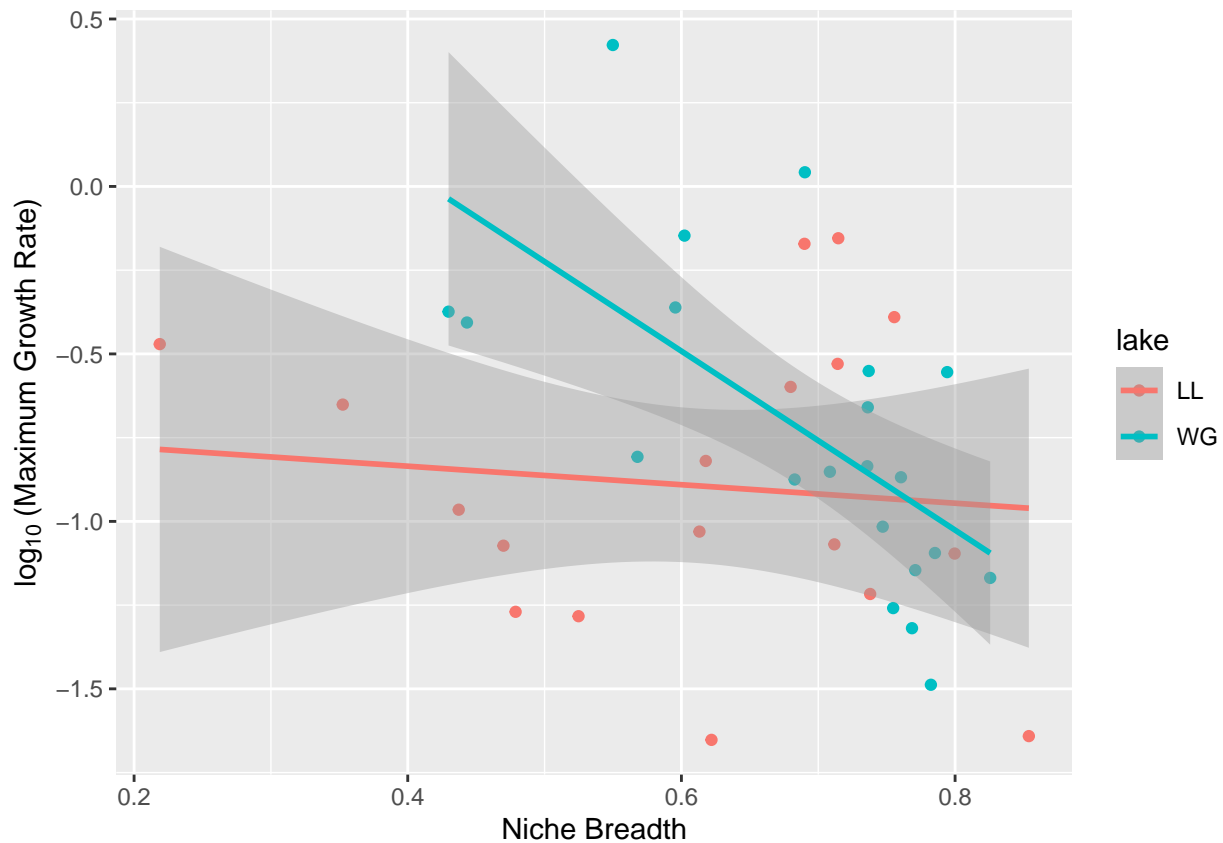
for (i in 1:nrow(nb.lake)) {
  ifelse(grepl("WG", row.names(nb.lake)[i]), nb.lake[i, 2] <- "WG",
    nb.lake[i, 2] <- "LL")
}

colnames(nb.lake)[1] <- "NB"

umax <- as.matrix(apply(p.growth, 1, max))
nb.lake = cbind(nb.lake, umax)

ggplot(data = nb.lake, aes(x = NB, y = log10(umax), color = lake)) +
  geom_point() + geom_smooth(method = "lm") + xlab("Niche Breadth") +
  ylab(expression(log[10] ~ "(Maximum Growth Rate)"))

## `geom_smooth()` using formula = 'y ~ x'
```



```
fit.lm <- lm(log10(umax) ~ NB * lake, data = nb.lake)
summary(fit.lm)
```

```
##
## Call:
## lm(formula = log10(umax) ~ NB * lake, data = nb.lake)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7557 -0.3108 -0.1077  0.3102  0.7800
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.7247     0.3852  -1.882   0.0682 .
## NB           -0.2763     0.6097  -0.453   0.6533
## lakeWG        1.8364     0.6909   2.658   0.0118 *
## NB:lakeWG    -2.3958     1.0234  -2.341   0.0251 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.418 on 35 degrees of freedom
## Multiple R-squared:  0.2595, Adjusted R-squared:  0.196
## F-statistic: 4.089 on 3 and 35 DF,  p-value: 0.01371
```

```
AIC(fit.lm)
```

```
## [1] 48.413
```

```

fit.plm <- phylolm(log10(umax) ~ NB * lake, data = nb.lake, nj.rooted,
  model = "lambda", boot = 0)
summary(fit.plm)

##
## Call:
## phylolm(formula = log10(umax) ~ NB * lake, data = nb.lake, phy = nj.rooted,
##   model = "lambda", boot = 0)
##
##   AIC logLik
## 41.08 -14.54
##
## Raw residuals:
##      Min      1Q   Median      3Q      Max
## -0.75804 -0.18999 -0.07425  0.32496  0.95857
##
## Mean tip height: 0.1814501
## Parameter estimate(s) using ML:
## lambda : 0.4861372
## sigma2: 0.9184437
##
## Coefficients:
##              Estimate      StdErr t.value p.value
## (Intercept) -0.891268   0.370036 -2.4086 0.02142 *
## NB          -0.004805   0.521303 -0.0092 0.99270
## lakeWG       1.438930   0.577231  2.4928 0.01755 *
## NB:lakeWG    -1.966388   0.848702 -2.3169 0.02648 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.1935    Adjusted R-squared: 0.1243
##
## Note: p-values and R-squared are conditional on lambda=0.4861372.

```

```
AIC(fit.plm)
```

```
## [1] 41.07574
```

- Why do we need to correct for shared evolutionary history?
- How does a phylogenetic regression differ from a standard linear regression?
- Interpret the slope and fit of each model. Did accounting for shared evolutionary history improve or worsen the fit?
- Try to come up with a scenario where the relationship between two variables would completely disappear when the underlying phylogeny is accounted for.

Answer 7a: We have to correct for this because it violates statistical assumptions that any two samples are independent of each other entirely. Continuing without correction can lead to misleading interpretations of the relationships between organisms and traits.

Answer 7b: the phylogenetic analysis accounts for phylogenetic information in addition to the regression

Answer 7c: The AIC for the standard linear regression is 48.413 and the phylogenetic regression is 41.076. Since there is a difference above 2, the addition of phylogeny did change the analysis. However, when looking at the p values, they don't seem to be very different, so I'm unsure whether it's better or worse.

Answer 7d: Maybe if we were studying the relationship between fur thickness and cold tolerance in mammals, we might see that thicker furs and cold tolerance are strongly correlated. However, if we include phylogenetic information to the analysis, we may see that both of these traits were developed independently by mammals that evolved in colder climates.

7) SYNTHESIS

Work with members of your Team Project to obtain reference sequences for 10 or more taxa in your study. Sequences for plants, animals, and microbes can be found in a number of public repositories, but perhaps the most commonly visited site is the National Center for Biotechnology Information (NCBI) <https://www.ncbi.nlm.nih.gov/>. In almost all cases, researchers must deposit their sequences in places like NCBI before a paper is published. Those sequences are checked by NCBI employees for aspects of quality and given an **accession number**. For example, here is an accession number for a fungal isolate that our lab has worked with: JQ797657. You can use the NCBI program nucleotide **BLAST** to find out more about information associated with the isolate, in addition to getting its DNA sequence: <https://blast.ncbi.nlm.nih.gov/>. Alternatively, you can use the `read.GenBank()` function in the `ape` package to connect to NCBI and directly get the sequence. This is pretty cool. Give it a try.

But before your team proceeds, you need to give some thought to which gene you want to focus on. For microorganisms like the bacteria we worked with above, many people use the ribosomal gene (i.e., 16S rRNA). This has many desirable features, including it is relatively long, highly conserved, and identifies taxa with reasonable resolution. In eukaryotes, ribosomal genes (i.e., 18S) are good for distinguishing coarse taxonomic resolution (i.e. class level), but it is not so good at resolving genera or species. Therefore, you may need to find another gene to work with, which might include protein-coding gene like cytochrome oxidase (COI) which is on mitochondria and is commonly used in molecular systematics. In plants, the ribulose-bisphosphate carboxylase gene (*rbcL*), which on the chloroplast, is commonly used. Also, non-protein-encoding sequences like those found in **Internal Transcribed Spacer (ITS)** regions between the small and large subunits of the ribosomal RNA are good for molecular phylogenies. With your team members, do some research and identify a good candidate gene.

After you identify an appropriate gene, download sequences and create a properly formatted fasta file. Next, align the sequences and confirm that you have a good alignment. Choose a substitution model and make a tree of your choice. Based on the decisions above and the output, does your tree jibe with what is known about the evolutionary history of your organisms? If not, why? Is there anything you could do differently that would improve your tree, especially with regard to future analyses done by your team?

SUBMITTING YOUR ASSIGNMENT

Use Knitr to create a PDF of your completed `8.PhyloTraits_Worksheet.Rmd` document, push it to GitHub, and create a pull request. Please make sure your updated repo include both the pdf and RMarkdown files. Unless otherwise noted, this assignment is due on **Wednesday, February 26th, 2025 at 12:00 PM (noon)**.