8. Worksheet: Phylogenetic Diversity - Traits

Lauren Albert; Z620: Quantitative Biodiversity, Indiana University

22 February, 2023

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 today, it is *imperative* that you **push** this file to your GitHub repo, at whatever stage you are. This will enable you to pull your work onto your own computer.
6. When you have completed the worksheet, **Knit** the text and code into a single PDF file by pressing the `Knit` button in the RStudio scripting panel. This will save the PDF output in your '8.BetaDiversity' folder.
7. After Knitting, please submit the worksheet by making a **push** to your GitHub repo and then create a **pull request** via GitHub. Your pull request should include this file (**11.PhyloTraits_Worksheet.Rmd**) with all code blocks filled out and questions answered) and the PDF output of `Knitr` (**11.PhyloTraits_Worksheet.pd**

The completed exercise is due on **Wednesday, February 22nd, 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 "*/8.PhyloTraits*" folder, and
4. load all of the required R packages (be sure to install if needed).

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

```
## [1] "/Users/laurenalbert/GitHub/QB2023_Albert/2.Worksheets/8.PhyloTraits"
```

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 .fasta file contains sequences in lowercase, without place holders for gaps whereas the .afa file contains sequences in uppercase and with '-' where gaps exist.

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.

```
package.list <- c('ape', 'seqinr', 'phylobase', 'adephylo', 'geiger', 'picante', 'stats', 'RColorBrewer
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: 'permute'

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

## This is vegan 2.6-4

##
## 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 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: 'phytools'

## The following object is masked from 'package:vegan':
##
##     scores

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

##
## 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 1.16.0
## 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:phytools':
##
##     untangle

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

## The following object is masked from 'package:geiger':
##
##     is.phylo
```

```
## 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: 'scales'

## The following object is masked from 'package:geiger':
##
##     rescale
```

```r
if (!require("BiocManager", quietly = TRUE)){
  install.packages("BiocManager")
}
if(!require("msa", quietly = TRUE)){
  BiocManager::install("msa")
}
```

```
##
## 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, setdiff, sort,
##     table, tapply, union, unique, unsplit, which.max, which.min

##
## Attaching package: 'S4Vectors'

## The following objects are masked from 'package:dplyr':
##
##     first, rename

## The following object is masked from 'package:tidyr':
##
##     expand

## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname

##
## Attaching package: 'IRanges'

## The following objects are masked from 'package:dplyr':
##
##     collapse, desc, slice

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

## 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'

##
## 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

##
## Attaching package: 'msa'

## The following object is masked from 'package:BiocManager':
##
##     version
```

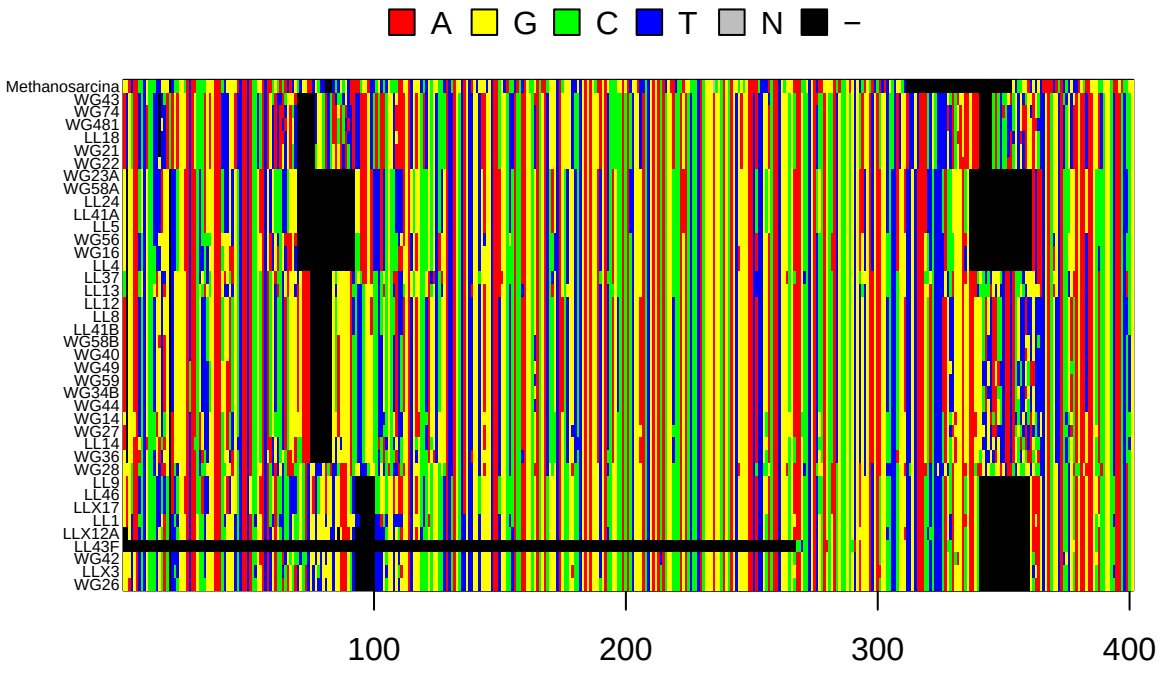
```
library(msa)

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

```
## DNAStringSet object of length 40:
##      width seq                                               names
##  [1]   619 ACACGTGAGCAATCTGCCCTTCT...TTCTCTGGGAATACCTGACGCT LL9
##  [2]   597 CGGCAGCGGGAAGTAGCTTGCTA...AACTGTTCAGCTAGAGTCTTGT WG14
##  [3]   794 CAGCGGCGGACGGGTGAGTAACA...GCTAACGCATTAAGCACTCCGC WG28
##  [4]   716 CTTCAGAGTTAGTGGCGGACGGG...TGCTAGTTGTCGGGATGCATGC LL24
##  [5]   803 ACGAACTCTTCGGAGTTAGTGGC...TAAAACTCAAAGGAATTGACGG LL41A
##  ...   ... ...
## [36]   652 TTCGGGAGTACACGAGCGGCGAA...TTCTCTGGGAATACCTGACGCT LL46
## [37]   661 GCGAACGGGTGAGTAACACGTGG...GAGCGAAAGCGTGGGTAGCGAA WG26
## [38]   694 GGCGAACGGGTGAGTAACACGTG...ACCCTGGTAGTCCACGCCGTAA WG42
## [39]   699 TACAGGTACCAGGCTCCTTCGGG...AAAGCATGGGTAGCGAACAGGA LLX17
## [40]  1426 TTCTGGTTGATCCTGCCAGAGGT...AACCTNAATTTTGCAAGGGGGG Methanosarcina
```

```
read.aln <- msaClustalOmega(seqs)
```

```
## using Gonnet
```

```
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 are appropriate for phylogenetic inference and why?

Answer 2a: The sequences are approximately 500bps ***Answer 2b***: I think the region between 100 to 350 would be appropriate because there is little missing information there.

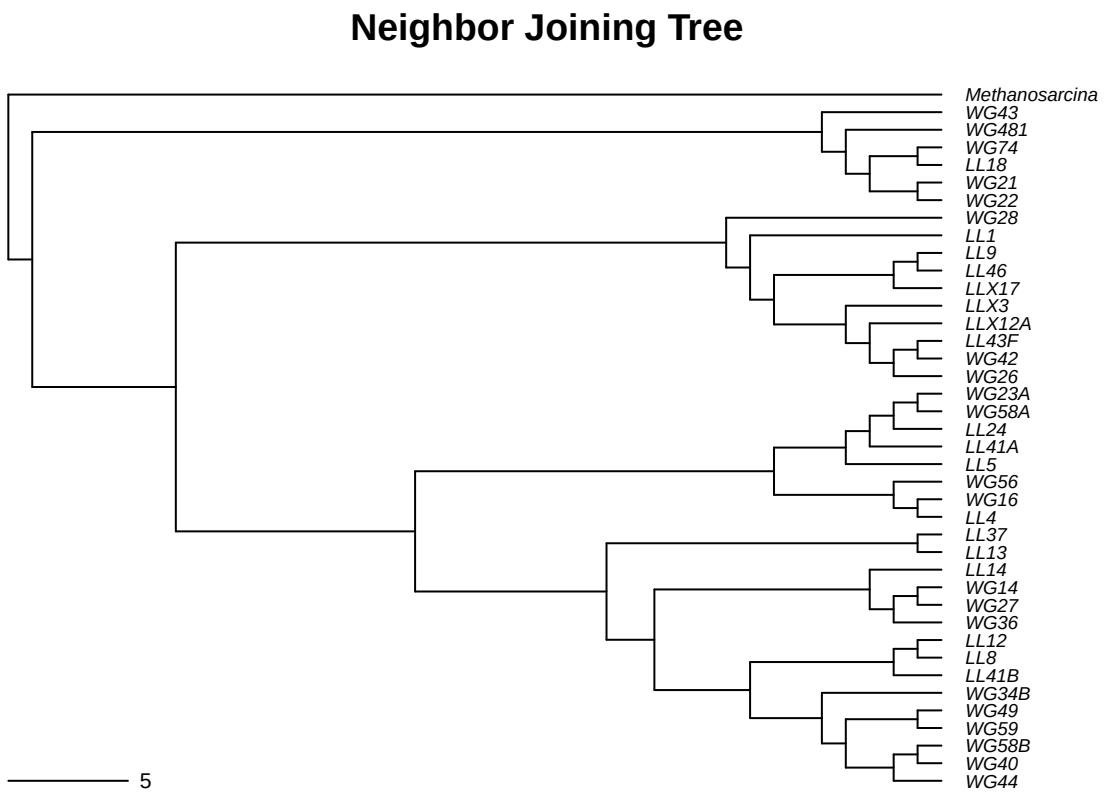
4) MAKING A PHYLOGENETIC TREE

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 =
           label.offset = 1)
add.scale.bar(cex = 0.7)
```



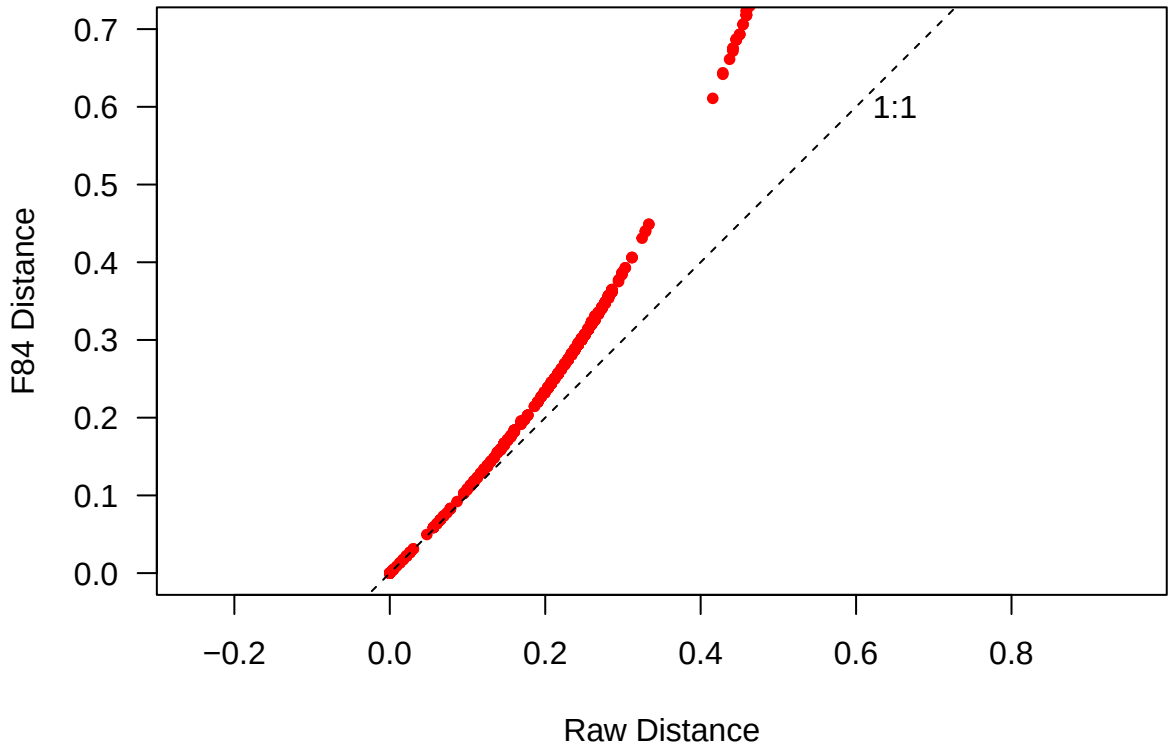
Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

Answer 3: Neighbor joining trees are useful to visualize the distances between taxa as a guide because they are built from distance matrices. The downside of neighbor joining trees is that entire sites may be deleted if there are multiple missing observations in the sequence.

B) SUBSTITUTION MODELS OF DNA EVOLUTION

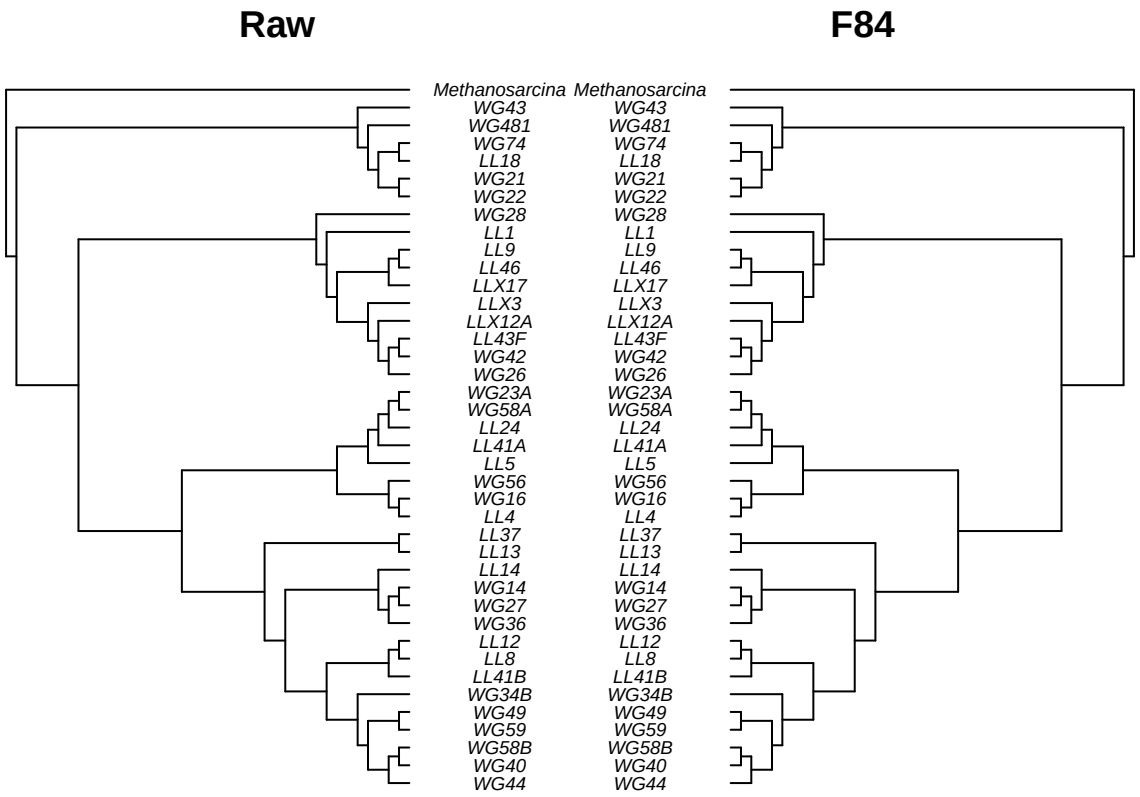
In the R code chunk below, do the following:
1. make a second distance matrix based on the Felsenstein 84 substitution model,
2. create a saturation plot to compare the *raw* and *Felsenstein (F84)* substitution models,
3. make Neighbor Joining trees for both, and
4. create a cophylogenetic plot to compare 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, xlim = c(0,0.7),
     ylim = c(0, 0.7), xlab = "Raw Distance", ylab = "F84 Distance")
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)
```

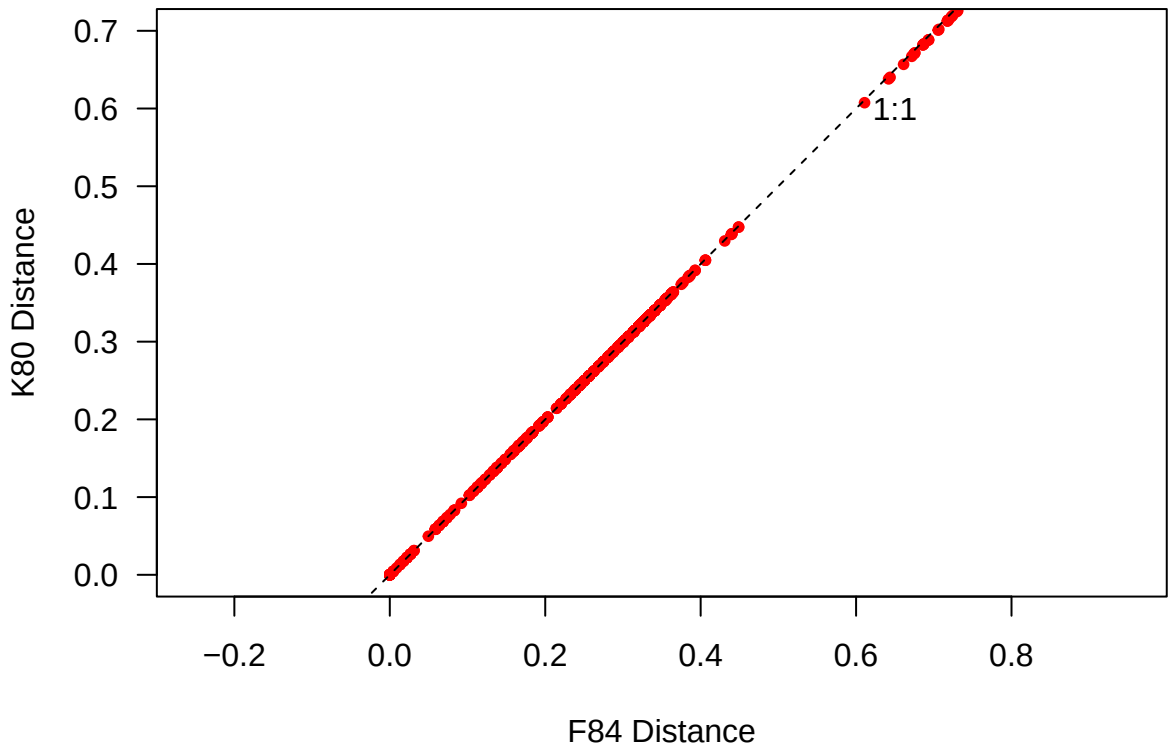
```
F84.outgroup <- match("Methanosarcina",  F84.tree$tip.label)

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")
```



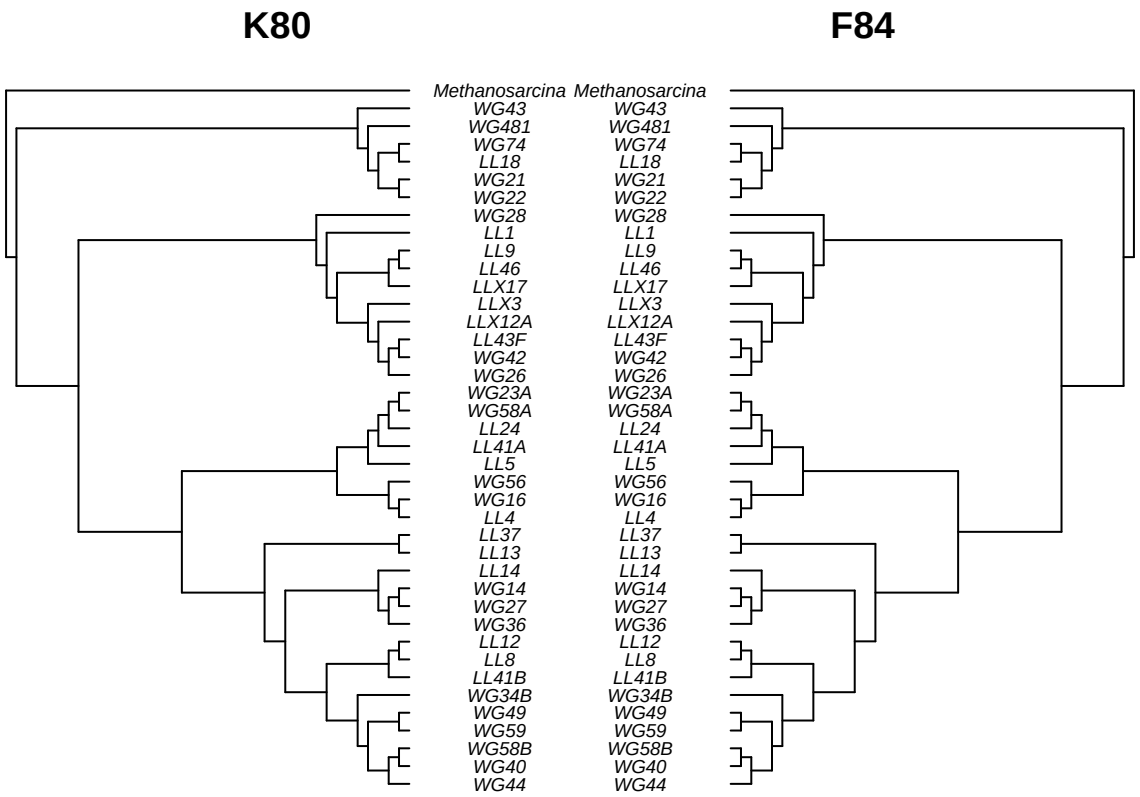
In the R code chunk below, do the following:
1. pick another substitution model,
2. create a distance matrix and tree for this model,
3. make a saturation plot that compares that model to the *Felsenstein (F84)* model,
4. make a cophylogenetic plot that compares the topologies of both models, and
5. be sure to format, add appropriate labels, and customize each plot.

```
seq.dist.K80<- dist.dna(p.DNAbin, model = "K80", pairwise.deletion = FALSE)

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

```
plot(seq.dist.F84, seq.dist.K80,
     pch = 20, col = "red", las = 1, asp = 1, xlim = c(0,0.7),
     ylim = c(0, 0.7), xlab = "F84 Distance", ylab = "K80 Distance")
abline(b = 1, a = 0, lty = 2)
text(0.65, 0.6, "1:1")
```




```
K80.tree <- bionj(seq.dist.K80)
F84.tree <- bionj(seq.dist.F84)

K80.outgroup <- match("Methanosarcina", K80.tree$tip.label)
F84.outgroup <- match("Methanosarcina",  F84.tree$tip.label)

K80.rooted <- root(K80.tree, K80.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(K80.rooted, type = "phylogram", direction = "right",
           show.tip.label = TRUE, use.edge.length = FALSE, adj = 0.5,
           cex = 0.6, label.offset = 2, main = "K80")

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")
```



Question 4:

a. Describe the substitution model that you chose. What assumptions does it make and how does it compare to the F84 model?
b. Using the saturation plot and cophylogenetic plots from above, describe how your choice of substitution model affects your phylogenetic reconstruction. If the plots are inconsistent with one another, explain why.
c. How does your model compare to the *F84* model and what does this tell you about the substitution rates of nucleotide transitions?

Answer 4a: I chose the Kimura model (K80) whihc assumes equal frequencies of nucleotides, but recognizes that transition mutations occur wirh higher probability than transversion mutations.
Answer 4b: The saturation plot follows the 1:1 line when comparing K80 and F84. The F84 distance is greater than the raw, suggesting that it is correcting for multiple substitutions. ***Answer 4c***: Switching the substitution model did not affect the phylogenetic reconstruction. The first model used, the Felsenstein model, assumes different rates of transitions and transversions while allowing for differences in base frequencies. I would have assummed the contructions would differ based on the properties of these two models, but perhaps the allowance of base frequency differences is stronger than any difference in rates of transition vs transversion or that substitution rates are slow/inconsequential.

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

```
phyDat.aln <- msaConvert(read.aln, type = "phangorn::phyDat")

aln.dist <- dist.ml(phyDat.aln)
aln.NJ <- NJ(aln.dist)

fit <- pml(tree = aln.NJ, data = phyDat.aln)

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

```
## optimize edge weights:  -10650.42 --> -10501.89
## optimize edge weights:  -10501.89 --> -10501.89
## optimize topology:  -10501.89 --> -10441.07  NNI moves:  10
## optimize edge weights:  -10441.07 --> -10441.07
## optimize topology:  -10441.07 --> -10441.07  NNI moves:  0
```

```
fitGTR <- optim.pml(fit, model = "GTR", optInv = TRUE, optGamma = TRUE,
                    rearrangement = "NNI", control = pml.control(trace = 0))
```

```
## only one rate class, ignored optGamma
```

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

```
## Likelihood Ratio Test Table
##   Log lik. Df Df change Diff log lik. Pr(>|Chi|)
## 1 -10441.1 77
## 2  -9976.9 86         9        928.34  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

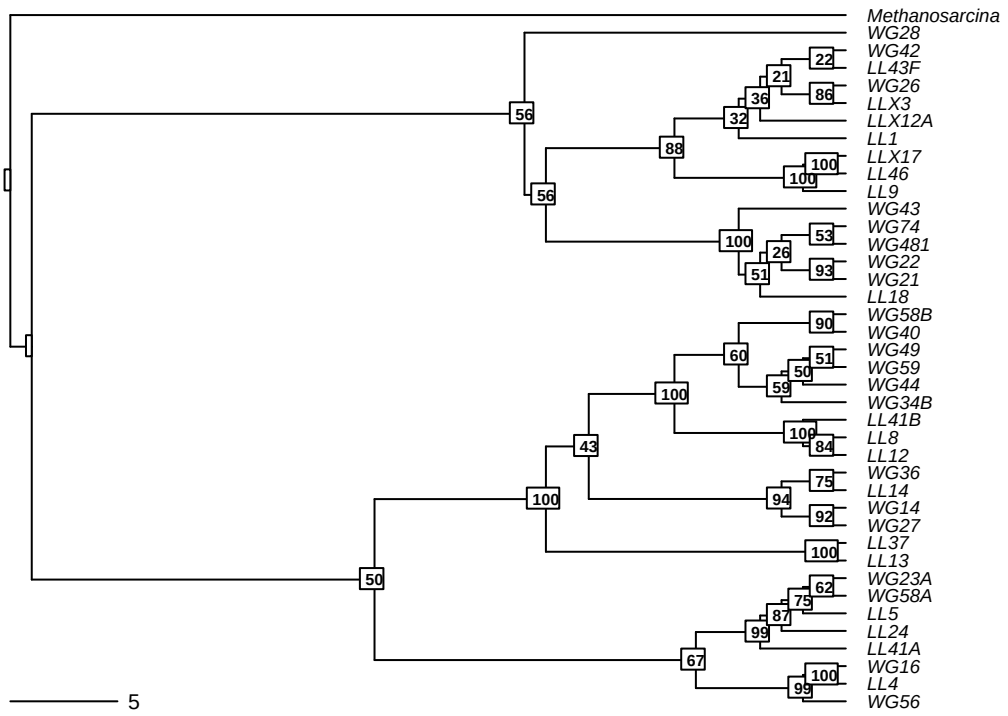
```
AIC(fitJC)
```

```
## [1] 21036.15
```

```
AIC(fitGTR)
```

```
## [1] 20125.8
```

```
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)
```



Question 5:

a) 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.

b) Why do we bootstrap our tree?

c) What do the bootstrap values tell you?

d) Which branches have very low support?

e) Should we trust these branches?

Answer 5a: Maximum likelihood trees are built on robust statistical procedures – the process of finding the parameter value that maximizes the likelihood of the data. This method is least affected by sampling error and takes into account nucleotide states (unlike NJ trees). The maximum likelihood tree and NJ tree demonstrate completely different phylogeny constructions.
Answer 5b: Bootstrapping helps provide more confidence in the placement of each brank in the phylogeny. The process re-samples the data to determine how reliable the placements are.
Answer 5c: Any interior branch that is different from the original tree get a score of 0, while matching branches get a 1.This is done repeatedly, additive numbers.
Answer 5d: Any nodes with a value less than 50% has low support. ***Answer 5e***: Our branches has mostly low support, however the lower half very interior branches have generally high support. The two main branches has around 50% support. Does this simply mean that the arrangement on the tree could just be rotated? (50% that the correct placement is opposite what is constructued here?)

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

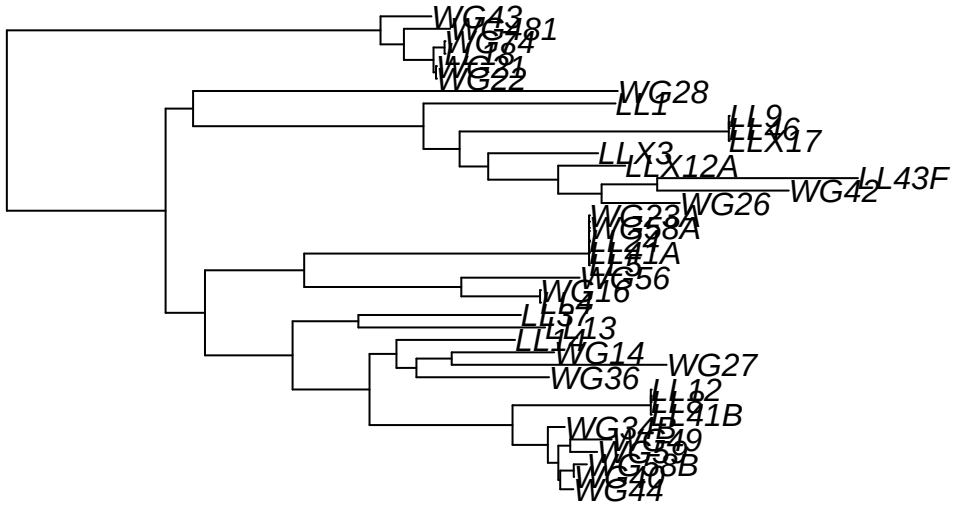
In the R code chunk below, do the following:
1. calculate the maximum growth rate (μ_{max}) of each isolate across all phosphorus types,
2. create a function that calculates niche breadth (nb), and
3. use this function to calculate nb for each isolate.

```
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

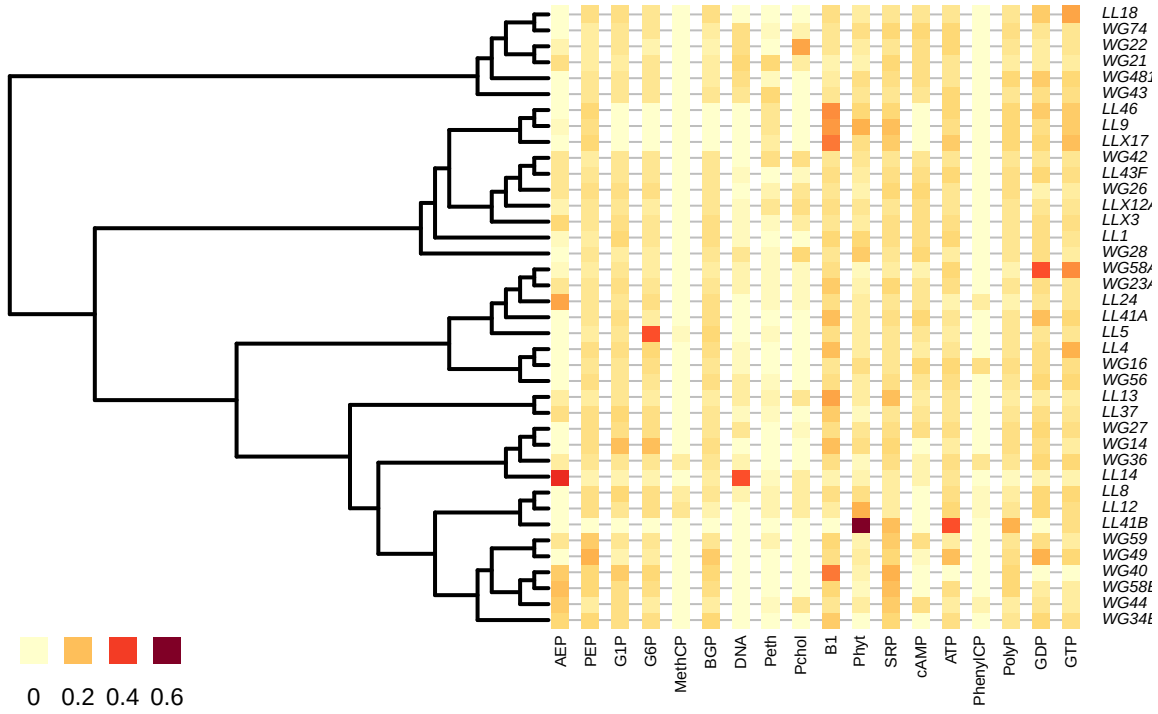
In the R code chunk below, do the following:
1. pick your favorite substitution model and make a Neighbor Joining tree,
2. define your outgroup and root the tree, and
3. remove the outgroup branch.

```
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)
```

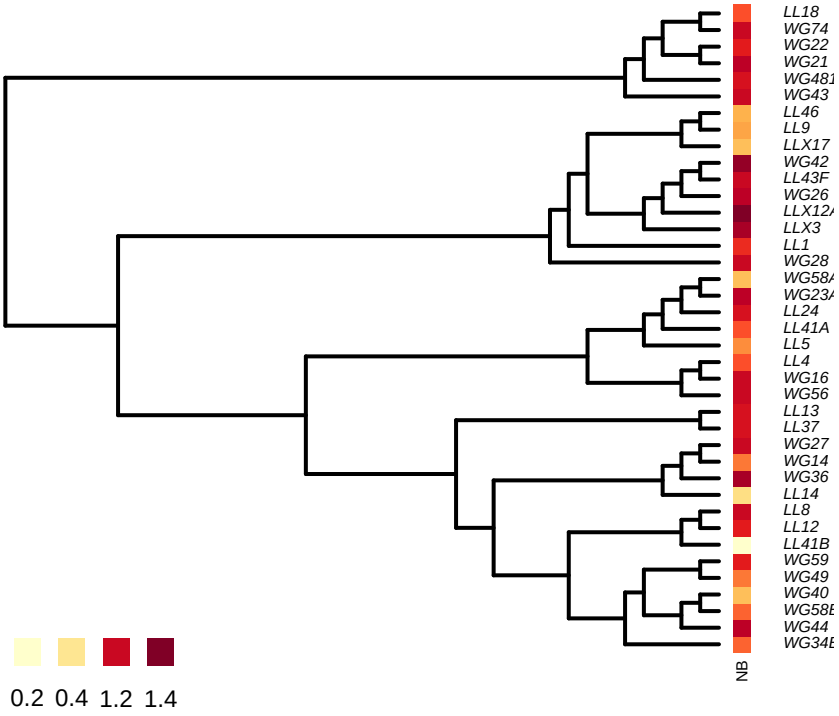


In the R code chunk below, do the following:
1. define a color palette (use something other than "YlOrRd"),
2. map the phosphorus traits onto your phylogeny,
3. map the *nb* trait on to your phylogeny, and
4. customize the plots as desired (use `help(table.phylo4d)` to learn about the options).

```
mypalette <- colorRampPalette(brewer.pal(9, "YlOrRd"))
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.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,
              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 6:

a) Make a hypothesis that would support a generalist-specialist trade-off.

b) What kind of patterns would you expect to see from growth rate and niche breadth values that would support this hypothesis?

Answer 6a: If taxa are spatially distant, then there should be more generalists but if taxa overlap in niche space then there will be more specialists on resources. ***Answer 6b***: If there are mostly generalists, where taxa don't compete for the same resources then there should be low growth rates on most resources. If most are specialists and are competing, each taxa will have a single or few resources for which it has high growth rates.

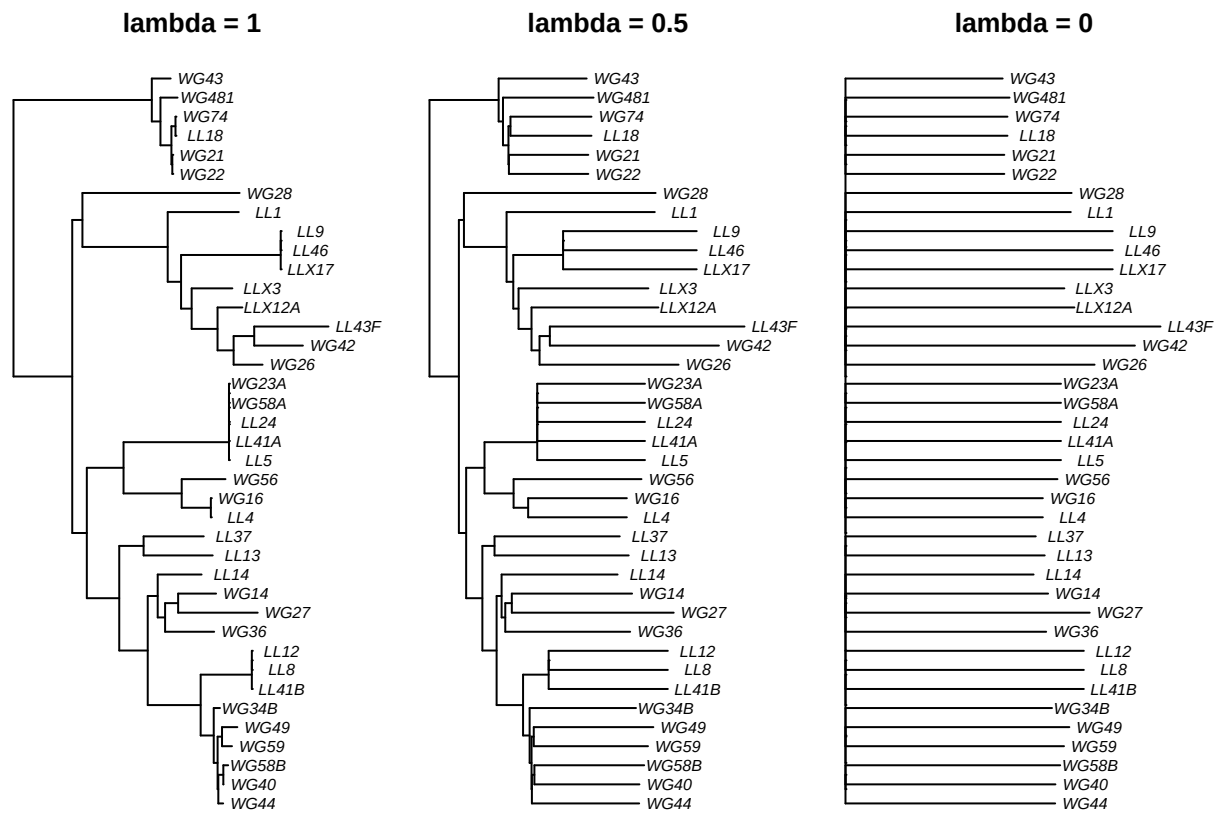
6) HYPOTHESIS TESTING

A) Phylogenetic Signal: Pagel's Lambda

In the R code chunk below, do the following:
1. create two rescaled phylogenetic trees using lambda values of 0.5 and 0,
2. plot your original tree and the two scaled trees, and
3. label and customize the trees as desired.

```
nj.lambda.5 <- geiger::rescale(nj.rooted, "lambda", 0.5)
nj.lambda.0 <- geiger::rescale(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 = 0.023830
##  sigsq = 0.104513
##  z0 = 0.659864
##
##  model summary:
##  log-likelihood = 21.569597
##  AIC = -37.139194
##  AICc = -36.453480
##  free parameters = 3
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 42
```

```
##  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 = 0.104395
##  z0 = 0.656188
##
##  model summary:
##  log-likelihood = 21.559484
##  AIC = -37.118968
##  AICc = -36.433254
##  free parameters = 3
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 0
##  number of iterations with same best fit = 89
##  frequency of best fit = 0.89
##
##  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 : 0.0238322
## logL(lambda) : 21.5696
## LR(lambda=0) : 0.0202263
## P-value (based on LR test) : 0.886907
```

Question 7: 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 7a***: Completely untransformed trees have a lambda of 1, where completely transformed trees have a lambda of 0. ***Answer 7b***: The difference of the AIC scores is less than 2 suggesting they are equivalent models. There is one model with a slighlty lower score. ***Answer 7c***: Suggests there is no phylogenetic signal.

B) Phylogenetic Signal: Blomberg's K

In the R code chunk below, do the following:
1. correct tree branch-lengths to fix any zeros,
2. calculate Blomberg's K for each phosphorus resource using the `phylosignal()` function,
3. use the Benjamini-Hochberg method to correct for false discovery rate, and
4. calculate Blomberg's K for niche breadth using the `phylosignal()` function.

```
nj.rooted$edge.length <- nj.rooted$edge.length + 10^-7
p.phylosignal <- matrix(NA, 6, 18)
colnames(p.phylosignal) <- colnames(p.growth.std)
rownames(p.phylosignal) <- c("K", "PIC.var.obs", "PIC.var.mean",
                             "PIC.var.P", "PIC.var.z", "PIC.P.BH")

for (i in 1:18){
  x <- setNames(as.vector(p.growth.std[,i]), row.names(p.growth))
                out <- phylosignal(x, nj.rooted)
                p.phylosignal[1:5, i] <- round(t(out), 3)
}

p.phylosignal[6, ] <- round(p.adjust(p.phylosignal[4, ], method = "BH"), 3)

# Use a For Loop to Calculate Blomberg's K for Each Resource
for (i in 1:18){
  x <- setNames(as.vector(p.growth.std[,i]),row.names(p.growth))
  out <- phylosignal(x, nj.rooted)
  p.phylosignal[1:5, i] <- round(t(out), 6)
}

print(p.phylosignal)
```

```
##                      AEP         PEP         G1P         G6P     MethCP
## K               0.000007    0.000007    0.000006    0.000003   0.000004
## PIC.var.obs  3444.343496  769.795083  925.439731 4356.162535 344.672980
## PIC.var.mean 8323.505128 1532.609295 1892.295150 3811.842273 507.883034
## PIC.var.P       0.140000    0.122000    0.102000    0.585000   0.347000
## PIC.var.z      -1.052714   -1.123565   -1.211211    0.214559  -0.475910
## PIC.P.BH        0.319000    0.319000    0.319000    0.706000   0.704000
##                      BGP         DNA        Peth       Pchol          B1
## K               0.000006    0.000016    0.000004    0.000004    0.000009
## PIC.var.obs   813.246026 1220.112379 1526.890222 2417.266612 1895.029807
## PIC.var.mean 1797.601444 5191.126841 1829.284538 3282.801432 5451.647764
## PIC.var.P       0.079000    0.071000    0.391000    0.416000    0.049000
## PIC.var.z      -1.348029   -1.063056   -0.363756   -0.495531   -1.603208
## PIC.P.BH        0.319000    0.319000    0.704000    0.704000    0.319000
##                     Phyt         SRP        cAMP         ATP    PhenylCP
## K               0.000003    0.000004    0.000012    0.000003    0.000002
## PIC.var.obs  9349.914736 1410.423535  926.600518 3475.808931 1240.302904
## PIC.var.mean 9236.302750 1637.808360 3065.001770 3079.423328  748.199508
## PIC.var.P       0.565000    0.366000    0.008000    0.578000    0.842000
## PIC.var.z       0.014479   -0.407047   -2.389455    0.179670    1.057185
## PIC.P.BH        0.706000    0.704000    0.198000    0.706000    0.901000
##                    PolyP         GDP         GTP
## K               0.000003    0.000002    0.000002
```

```
## PIC.var.obs  1169.765824 6314.948708 4589.816606
## PIC.var.mean 1235.695975 3695.449201 2963.916855
## PIC.var.P       0.501000    0.858000    0.871000
## PIC.var.z      -0.116596    1.208289    1.180076
## PIC.P.BH        0.706000    0.901000    0.901000
```

```
signal.nb <- phylosignal(nb, nj.rooted)
signal.nb
```

```
##              K PIC.variance.obs PIC.variance.rnd.mean PIC.variance.P
## 1 3.171942e-06         51226.32              49964.89          0.565
##   PIC.variance.Z
## 1     0.06150494
```

Question 8: Using the K-values and associated p-values (i.e., "PIC.var.P"") from the `phylosignal` output, answer the following questions:

a. Is there significant phylogenetic signal for niche breadth or standardized growth on any of the phosphorus resources?

b. If there is significant phylogenetic signal, are the results suggestive of clustering or overdispersion?

Answer 8a: I do not see any significant phylogenetic signals (PIC.var.P > 0.05) ***Answer 8b***: Most of the K values are very small, or close to 0.

C. Calculate Dispersion of a Trait

In the R code chunk below, do the following:
1. turn the continuous growth data into categorical data,
2. add a column to the data with the isolate name,
3. combine the tree and trait data using the `comparative.data()` function in `caper`, and
4. use `phylo.d()` to calculate D on at least three phosphorus traits.

```
p.growth.pa <- as.data.frame((p.growth > 0.01) * 1)

apply(p.growth.pa, 2, sum)
```

```
##      AEP      PEP      G1P      G6P   MethCP      BGP      DNA     Peth
##       20       38       35       34        3       35       19       21
##    Pchol       B1     Phyt      SRP     cAMP      ATP PhenylCP    PolyP
##       18       38       36       39       29       38        6       39
##      GDP      GTP
##       37       38
```

```
p.growth.pa$name <- rownames(p.growth.pa)

p.traits <- comparative.data(nj.rooted, p.growth.pa, "name")
phylo.d(p.traits, binvar = AEP, permut = 10000)
```

```
##
## Calculation of D statistic for the phylogenetic structure of a binary variable
##
##   Data :  p.growth.pa
##   Binary variable :  AEP
##   Counts of states:  0 = 19
##                      1 = 20
##   Phylogeny :  nj.rooted
##   Number of permutations :  10000
##
## Estimated D :  0.3298405
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.0013
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.0837
```

```
phylo.d(p.traits, binvar = PhenylCP, permut = 10000)
```

```
##
## Calculation of D statistic for the phylogenetic structure of a binary variable
##
##   Data :  p.growth.pa
##   Binary variable :  PhenylCP
##   Counts of states:  0 = 33
##                      1 = 6
##   Phylogeny :  nj.rooted
##   Number of permutations :  10000
##
## Estimated D :  0.8004877
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.2021
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.0319
```

```
phylo.d(p.traits, binvar = DNA, permut = 10000)
```

```
##
## Calculation of D statistic for the phylogenetic structure of a binary variable
##
##   Data :  p.growth.pa
##   Binary variable :  DNA
##   Counts of states:  0 = 20
##                      1 = 19
##   Phylogeny :  nj.rooted
##   Number of permutations :  10000
##
## Estimated D :  0.7171274
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.0817
## Probability of E(D) resulting from Brownian phylogenetic structure    :  9e-04
```

```
phylo.d(p.traits, binvar = cAMP, permut = 10000)
```

```
##
## Calculation of D statistic for the phylogenetic structure of a binary variable
##
##   Data :  p.growth.pa
```

```
##   Binary variable :  cAMP
##   Counts of states:  0 = 10
##                      1 = 29
##   Phylogeny :  nj.rooted
##   Number of permutations :  10000
##
## Estimated D :  0.05101528
## Probability of E(D) resulting from no (random) phylogenetic structure :  0
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.4358
```

Question 9: Using the estimates for D and the probabilities of each phylogenetic model, answer the following questions:

a. Choose three phosphorus growth traits and test whether they are significantly clustered or overdispersed?

b. How do these results compare the results from the Blomberg's K analysis?

c. Discuss what factors might give rise to differences between the metrics.

Answer 9a: The D for cAMP is close to zero, meaining randomly clumped. Where the D for PhenylCP and DNA are closer to 1, consistent with Brownian motion. ***Answer 9b***: Confused on how to interpret the output across these two models. ***Answer 9c***:

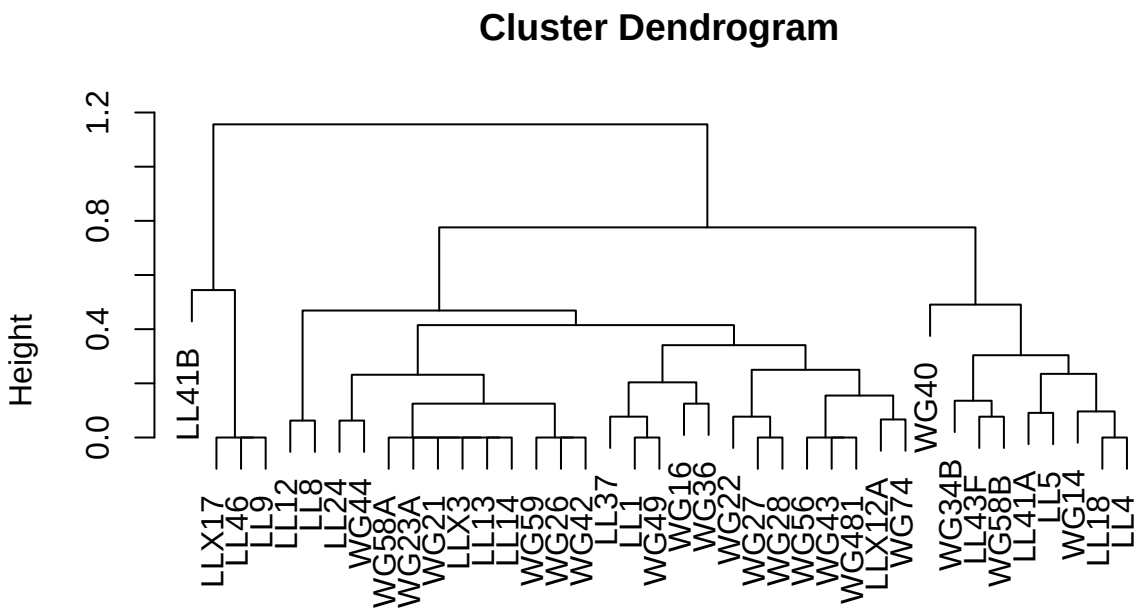
D) CORRESPONDENCE BETWEEN TRAIT CLUSTERS AND PHYLOGENY

In the R code chunk below, do the following: 1. calculate Jaccard Index on resource use incidence matrix 2. create a hierarchical cluster of resource use 3. map the resource use cluster onto the phylogeny for each environment, and 4. use RF.dist and mantel to measure the degree of correspondence between each dendrogram.

```
no <- vegdist(p.growth.pa[,1:18], method = 'jaccard', binary = TRUE)
m <- c("average", "single", "complete", "ward")
names(m) <- c("average", "single", "complete", "ward")

ac <- function(x) {
  agnes(no, method = x)$ac
}

sapply(m, ac)
```

```
##   average    single  complete      ward
## 0.9064731 0.8881997 0.9207206 0.9470011
```

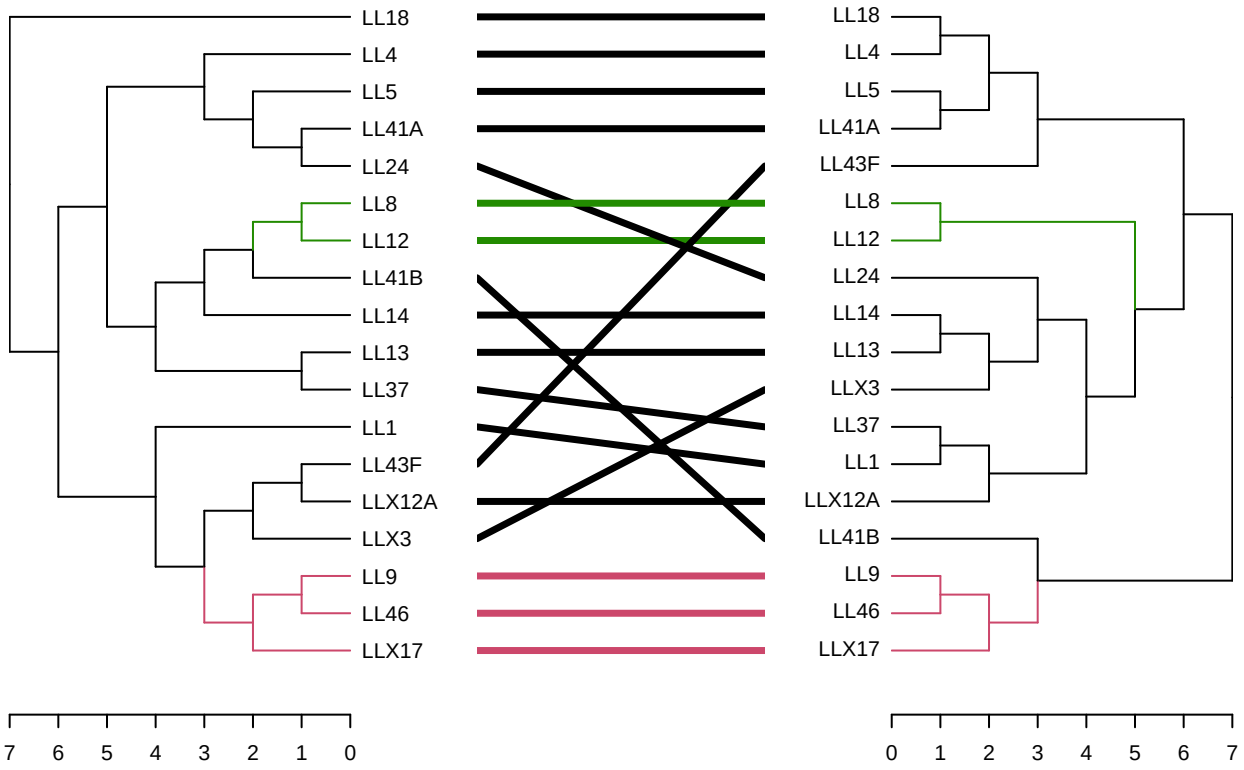
```
no.tree <- hclust(no, method = "ward.D2")
plot(no.tree)
```




```
is.ultrametric(nj.rooted)
```

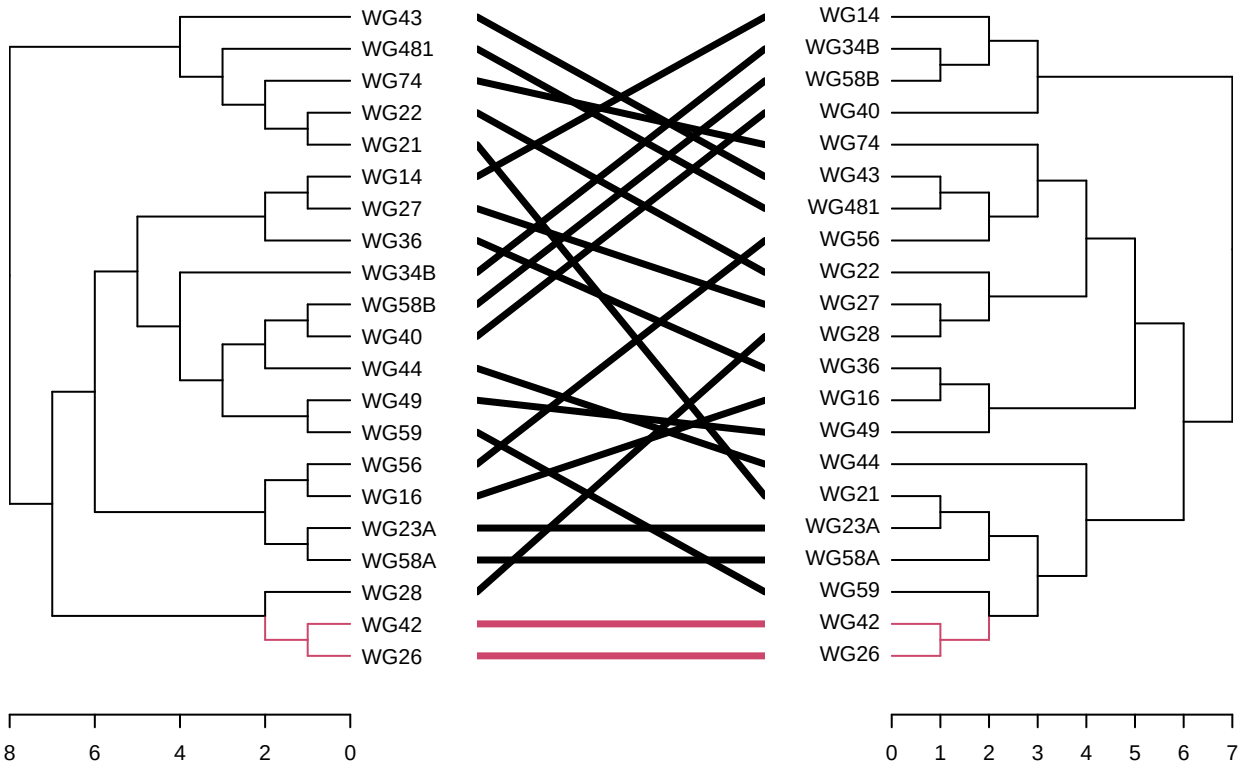
```
## [1] FALSE
```

```
LL.tree <- drop.tip(nj.rooted, c(nj.rooted$tip.label[grepl("WG",
                                                           nj.rooted$tip.label)]))

LL.function <- drop.tip(as.phylo(no.tree),
                        c(no.tree$labels[grepl("WG", no.tree$labels)]))

WG.tree <- drop.tip(nj.rooted, c(nj.rooted$tip.label[grepl("LL",
                                                         nj.rooted$tip.label)]))

WG.function <- drop.tip(as.phylo(no.tree),
                        c(no.tree$labels[grepl("LL", no.tree$labels)]))

par(mar = c(1,5,1,5) + 0.1)
dendlist(as.cladogram(as.dendrogram.phylo(LL.tree)),
         as.cladogram(as.dendrogram(LL.function)))%>%
  untangle(method = "step2side")%>%
  tanglegram(common_subtrees_color_branches = TRUE,
             highlight_distinct_edges = FALSE, highlight_branches_lwd = FALSE,
             margin_inner = 5)%>%
  entanglement()
```




```
## [1] 0.1026526
```

```
par(mar = c(1,5,1,5) + 0.1)
dendlist(as.cladogram(as.dendrogram.phylo(WG.tree)),
         as.cladogram(as.dendrogram(WG.function)))%>%
  untangle(method = "step2side")%>%
  tanglegram(common_subtrees_color_branches = TRUE,
             highlight_distinct_edges = FALSE,
             highlight_branches_lwd = FALSE,
             margin_inner = 5)%>%
  entanglement()
```




```
## [1] 0.2682644
```

```
RF.dist(LL.tree, as.phylo(as.dendrogram(LL.function)), normalize = TRUE,
        check.labels = TRUE, rooted = FALSE)
```

```
## [1] 0.8
```

```
RF.dist(WG.tree, as.phylo(as.dendrogram(WG.function)), normalize = TRUE,
        check.labels = TRUE, rooted = FALSE)
```

```
## [1] 0.9444444
```

```
mantel(cophenetic.phylo(LL.tree), cophenetic.phylo(LL.function),
       method = "spearman", permutations = 999)
```

```
##
## Mantel statistic based on Spearman's rank correlation rho
##
## Call:
## mantel(xdis = cophenetic.phylo(LL.tree), ydis = cophenetic.phylo(LL.function),      method = "spearma
##
## Mantel statistic r: 0.1064
##       Significance: 0.133
```

```
##
## Upper quantiles of permutations (null model):
##   90%   95% 97.5%   99%
## 0.132 0.179 0.213 0.288
## Permutation: free
## Number of permutations: 999
```

```
mantel(cophenetic.phylo(WG.tree), cophenetic.phylo(WG.function),
       method = "spearman", permutations = 999)
```

```
##
## Mantel statistic based on Spearman's rank correlation rho
##
## Call:
## mantel(xdis = cophenetic.phylo(WG.tree), ydis = cophenetic.phylo(WG.function),      method = "spearma
##
## Mantel statistic r: -0.0755
##       Significance: 0.746
##
## Upper quantiles of permutations (null model):
##   90%   95% 97.5%   99%
## 0.153 0.208 0.247 0.293
## Permutation: free
## Number of permutations: 999
```

Question 10: Using a hierarchical clustering algorithm, map similarity in resource use map onto the phylogeny and answer the following questions: a. Compare the patterns between resource use and phylogeny between each lake. How do the two sets of tangelgrams differ between the taxa isolated from each lake? b. Interpret the Robinson-Fould index and Mantel correlation test results. How does each analysis differ and shape our interpretation of correlating niche overlap with phylogeny.

> ***Answer 10a***: Resource use and phylogeny are consistent between each lake, both scores are close to 0. The taxa isolated from WG seem to be more tangled than those from LL. ***Answer 10b***:

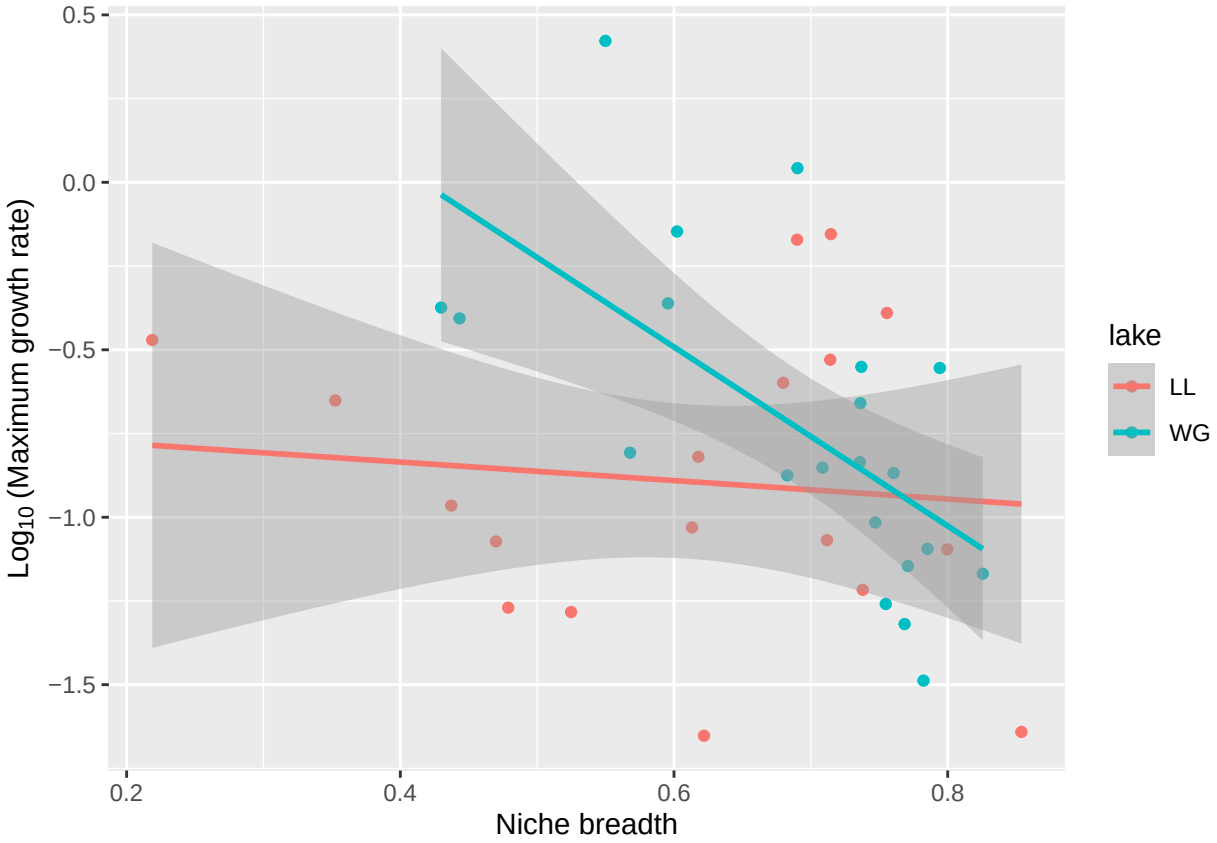
7) PHYLOGENETIC REGRESSION

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

```
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
##  40.63 -14.32
##
## Raw residuals:
##      Min       1Q   Median       3Q      Max
## -0.76854 -0.20196 -0.09182  0.31168  0.94216
##
## Mean tip height: 0.1881134
## Parameter estimate(s) using ML:
## lambda : 0.5186814
## sigma2: 0.9115308
##
## Coefficients:
##              Estimate    StdErr t.value p.value
## (Intercept) -0.896644  0.371387 -2.4143 0.02113 *
## NB           0.020719  0.519025  0.0399 0.96838
## lakeWG       1.466073  0.566374  2.5885 0.01395 *
## NB:lakeWG   -2.001663  0.833823 -2.4006 0.02182 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.2037    Adjusted R-squared: 0.1354
##
## Note: p-values and R-squared are conditional on lambda=0.5186814.
```

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

```
## [1] 40.63487
```

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

> ***Answer 11a***: We expect traits to be non-independent due to evolutionary history, where there are shared ancestors affecting the relationship among traits. ***Answer 11b***: A phylogenetic regression residual errors are described by a covariance matric that takes into account the branch length underlying phylogeny. The phylogenetic signal is also taken into account for statistical performance. ***Answer 11c***: Accounting for non-independence demonstrated that phylogeny is impacting the results, where there is phylogenetic signal in the residuals. ***Answer 11d***: A scenario where a relationship between two variables would disappear once shared evolutionary history is accounted for might be examples of convergent evolution (i.e. echolocation in aquatic and terrestrial environments)

7) SYNTHESIS

Work with members of your Team Project to obtain reference sequences for taxa in your study. Sequences for plants, animals, and microbes can 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 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 course 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 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?

```
zoopseqs <- readDNAStringSet(c("cerioBIO.fasta", "puliBIO.fasta", "daphBIO.fasta", "platyseq.fasta"))
zoopseqs
```

```
## DNAStringSet object of length 4:
##     width seq                                                names
## [1]   658 GACATTGTATTTTATTTTTGGAG...CTATTTTATATCAACATCTCTTT MG450106.1 Ceriod...
## [2]   653 TACTCTCTATTTTATCTTTGGTA...GGATCCAATCTTATACCAACATT JN233925.1 Daphni...
## [3]   658 AACCCTTTACTTCATTTTCGGGA...CAATCTTATACCAGCATCTATTC MG449742.1 Daphni...
## [4]   633 GTGATCCTGTGCTGTTTCAGCAT...GCATTATTTTGGTATGTGTGGTT OQ423079.1 Digram...
```

```
zoop.aln <- msaClustalOmega(zoopseqs)
```

```
## using Gonnet
```

```
z.save.aln <- msaConvert(zoop.aln, type = "bios2mds::align")
export.fasta(z.save.aln, "./zoopseq.afa")

z.DNAbin <- as.DNAbin(zoop.aln)
window <- z.DNAbin[, 100:500]
image.DNAbin(window, cex.lab = 0.50)
```