8. Worksheet: Phylogenetic Diversity - Traits

Madison Brown; Z620: Quantitative Biodiversity, Indiana University

25 February, 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).

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

```
## [1] "/cloud/project/QB2025_Brown/Week6-PhyloTraits"
```

```r
setwd("/cloud/project/QB2025_Brown/Week6-PhyloTraits")

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: '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.
```

```r
library(ape)
library(seqinr)
library(phylobase)
library(adephylo)
library(geiger)
library(picante)
library(stats)
library(RColorBrewer)
library(caper)
library(phylolm)
library(pmc)
library(ggplot2)
library(tidyr)
library(dplyr)
library(phangorn)
library(pander)
library(vegan)
library(phytools)
library(cluster)
library(dendextend)
library(phylogram)
library(bios2mds)
library(formatR)

# comment out

#install.packages("pak")

# comment out

# reinstall/update
library("pak")
#pak::pkg_install("msa")

library(msa)
```

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

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***: You can see that both of these files include DNA sequences; yet, they are in slightly different forms. The first obvious distinction between the two files is that the sequences in the afa file are capitalized, while the sequences in the fasta file are lowercase. The sequences in the afa file also contain gaps, dictated by dashes, that the fasta file does not have. The sequences in the fasta file are one long, conitnuous sequence. This is likely because the sequences in the afa file have been aligned, hence why the gaps are present. The fasta file appears to be unaligned, raw sequences.

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.

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

```
##
## Attaching package: 'BiocManager'
```

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

```r
BiocManager::install("Biostrings")
```

```
## 'getOption("repos")' replaces Bioconductor standard repositories, see
## 'help("repositories", package = "BiocManager")' for details.
## Replacement repositories:
##     CRAN: http://rspm/default/__linux__/focal/latest
```

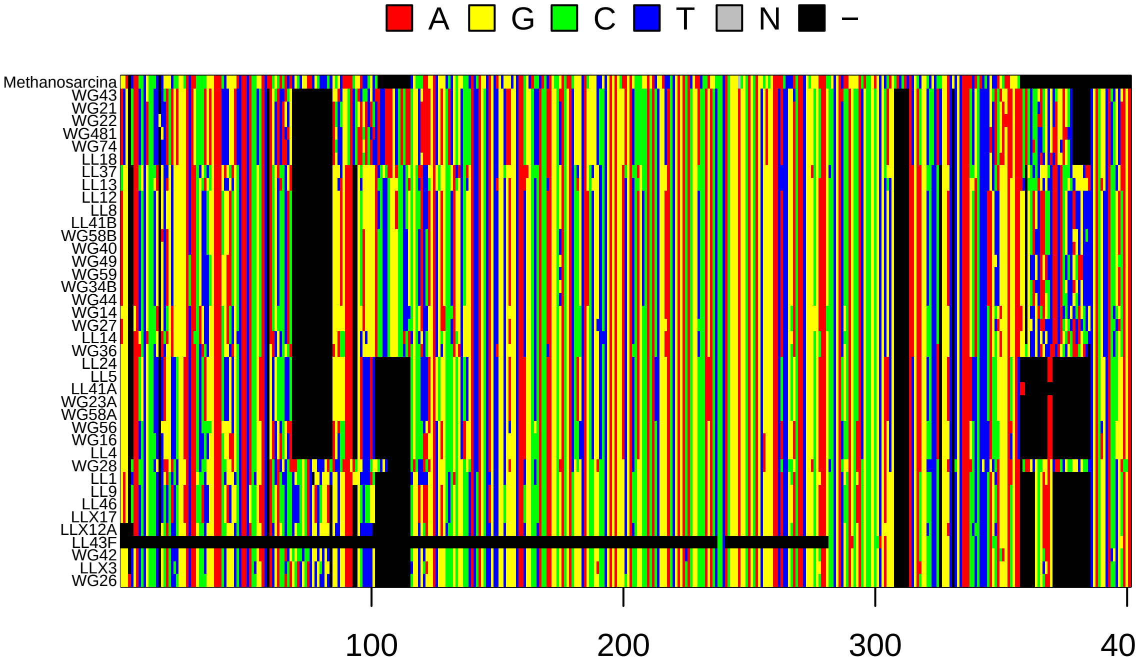
```
## Bioconductor version 3.20 (BiocManager 1.30.25), R 4.4.2 (2024-10-31)
```

```
## Warning: package(s) not installed when version(s) same as or greater than current; use
##   `force = TRUE` to re-install: 'Biostrings'
## Installation paths not writeable, unable to update packages
##   path: /opt/R/4.4.2/lib/R/library
##   packages:
##     class, cluster, foreign, KernSmooth, MASS, Matrix, nlme, nnet, rpart,
##     spatial, survival

## Old packages: 'data.table', 'indicspecies', 'processx'
```

```r
library(Biostrings)

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
```

```r
read.aln <- msaMuscle(seqs)
save.aln <- msaConvert(read.aln, type = "bios2mds::align")
library(bios2mds)
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 sequence is about 400 basepairs long.
   Answer 2b: To determine regions best suited for a phylogenetic inference, you would want to identify the areas that are best aligned. This would correspond to the vertical areas that are all the same color. These are ideal areas because if they are the same color, then that means the same nucleotide is present across that region. Based on the visualization above, the region located between 125-150 base pairs appear to have very good alignments along with regions ~ 230-310.

4) MAKING A PHYLOGENETIC TREE

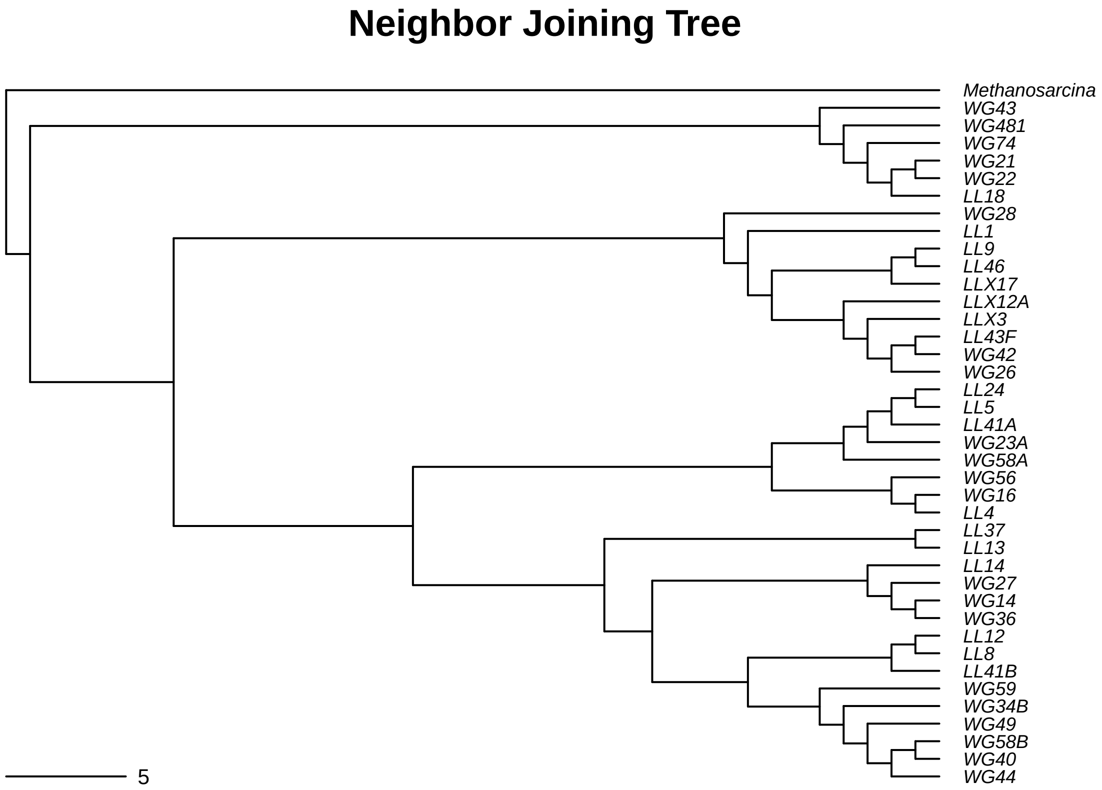
Once you have aligned your sequences, the next step is to construct a phylogenetic tree. Not only is a phylogenetic tree effective for visualizing the evolutionary relationship among taxa, but as you will see later, the information that goes into a phylogenetic tree is needed for downstream analysis.

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)
```



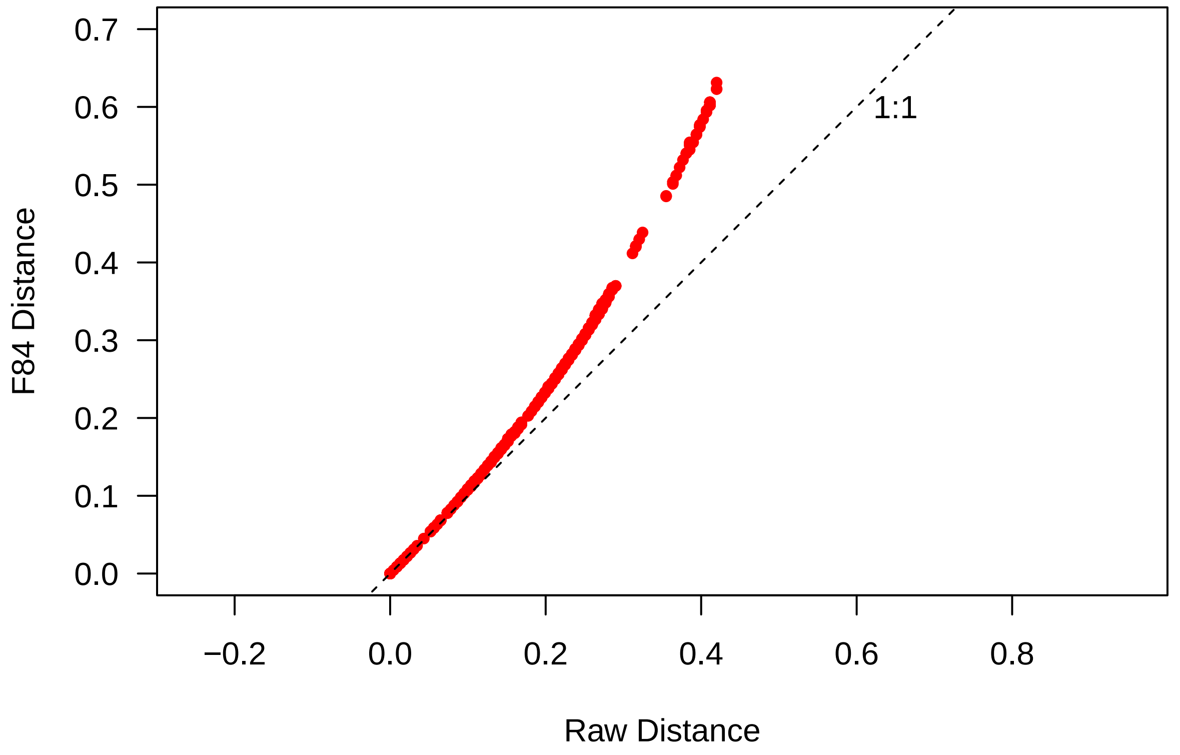
Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

> ***Answer 3***: Neighbor joining tree are relatively simple to make and serve as a good first step when making phylogenetic trees. They provide a good preliminary basis of the taxonomic relationships present and are very useful when wanting to visualize large data sets. This type of tree is also good for looking at raw data. However, it does not correct for multiple substitutions or nucleotide transitions overtime and only accounts for distance. Neighbor joining trees also do not provide any statistical information and does not allow you to quantify any differences.

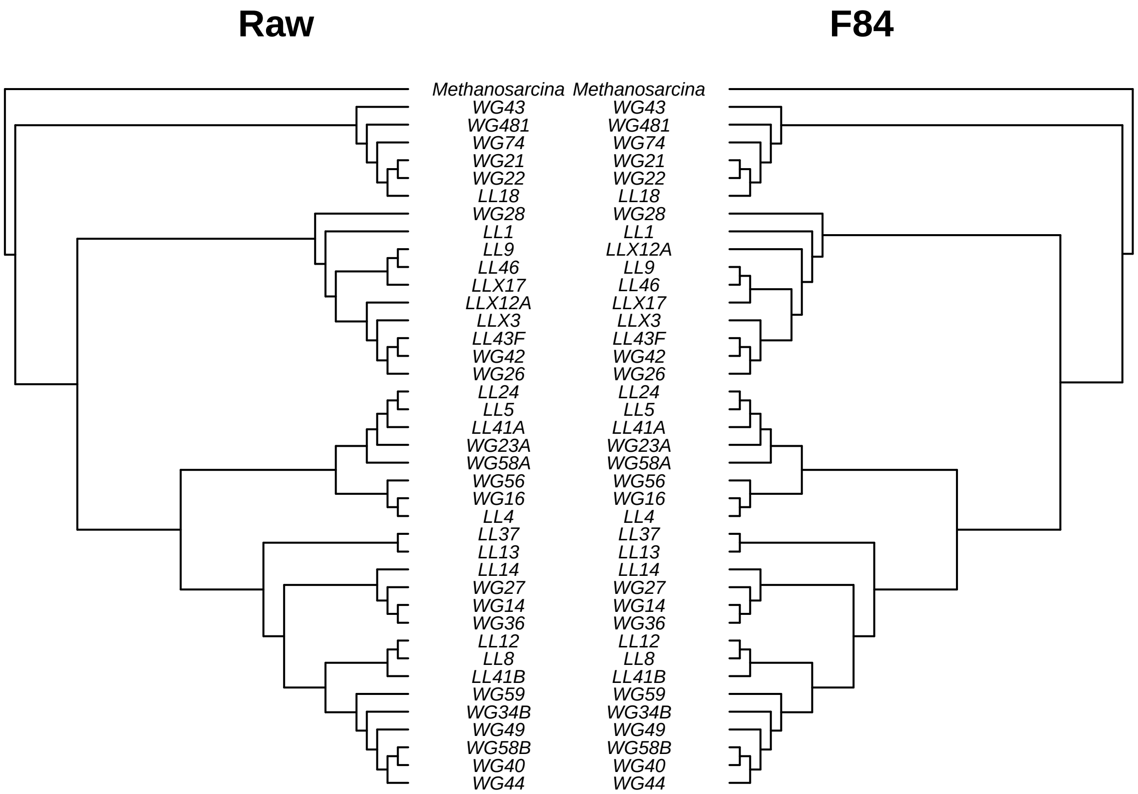
B) SUBSTITUTION MODELS OF DNA EVOLUTION

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")
```



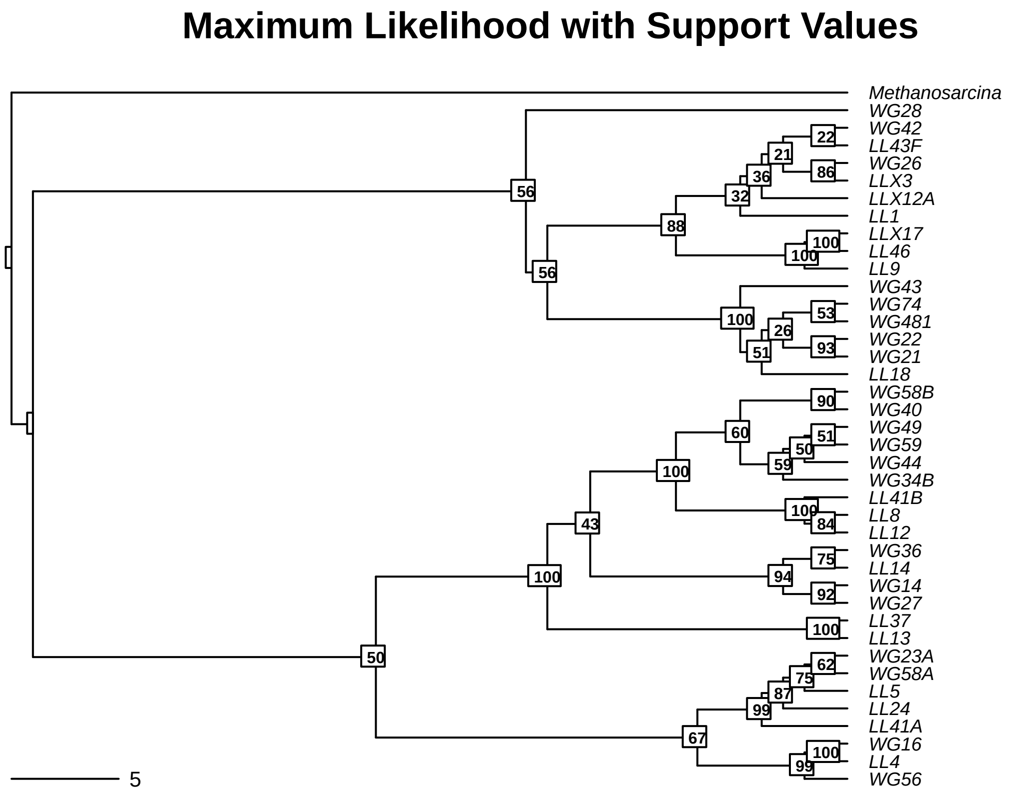
```
dist.topo(raw.rooted, F84.rooted, method = "score")
```

```
##            tree1
## tree2 0.04219896
```

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

```
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 4:

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? Why or why not?

Answer 4a: The maximum likelihood tree, at first glance, appears to look like the neighbor joining tree in terms of the layout of the two plots. However, upon closer examination, you can see that many of the phylogenetic relationships between taxa are different. Taxa are located in different areas of the ML plot and many of them appear to have a closer relationship with a different taxa than what is showed in the neighbor joining tree. These differences are because the ML tree has a statistical method associated with it that neighbor joining trees do not have. ML trees have a parameter value that informs you of the accuracy of the data and also takes into consideration the states of the nucleotides. ***Answer 4b***: We bootstrap our tree to get an idea of how accurate each placement on the tree is. It lets us know how reliable the tree is and also which nodes and placements are more certain than others. ***Answer 4c***: Bootstrap values give us an idea of how "correct" the placement is. Values above 95% tells you that that node/placement is essentially correct. If the value is higher than 70%, it has moderate support while values less than 50% implies very weak support and tells you that the placement is not certain. ***Answer 4d***: There are several branches at the top of the ML plot with very low support values in the 20s and 30s. Some of these branches are the branches connecting WG42 and LL43F. Several of the nodes before that branch also have very low support values. Additionally, all of the deep nodes furthest to the left have moderate support with values in the 50s. ***Answer 4e***: Beginning with the far left of the plot, the moderate bootstrap values imply that we should not fully trust the early evolutionary divergences. While there may be some accuracy, it is likely that there is more information that needs be uncovered to fully understand the early divergences. Moving forward to the top right of the plot, many of these values are extremely low, aside from the bootstrap value representing the relationship between WG26 and LLX3. This region of the graph should not be trusted for the most part because the values indicate little to no statistical support for the placement of those taxa. Therefore, we should not infer that those taxa are closely related.

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.

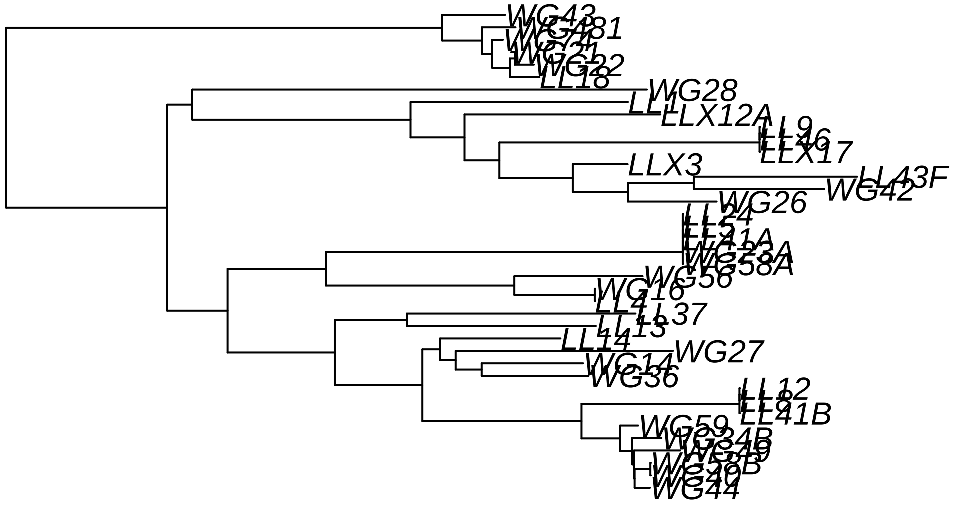
```
umax <- (apply(p.growth, 1, max))

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.

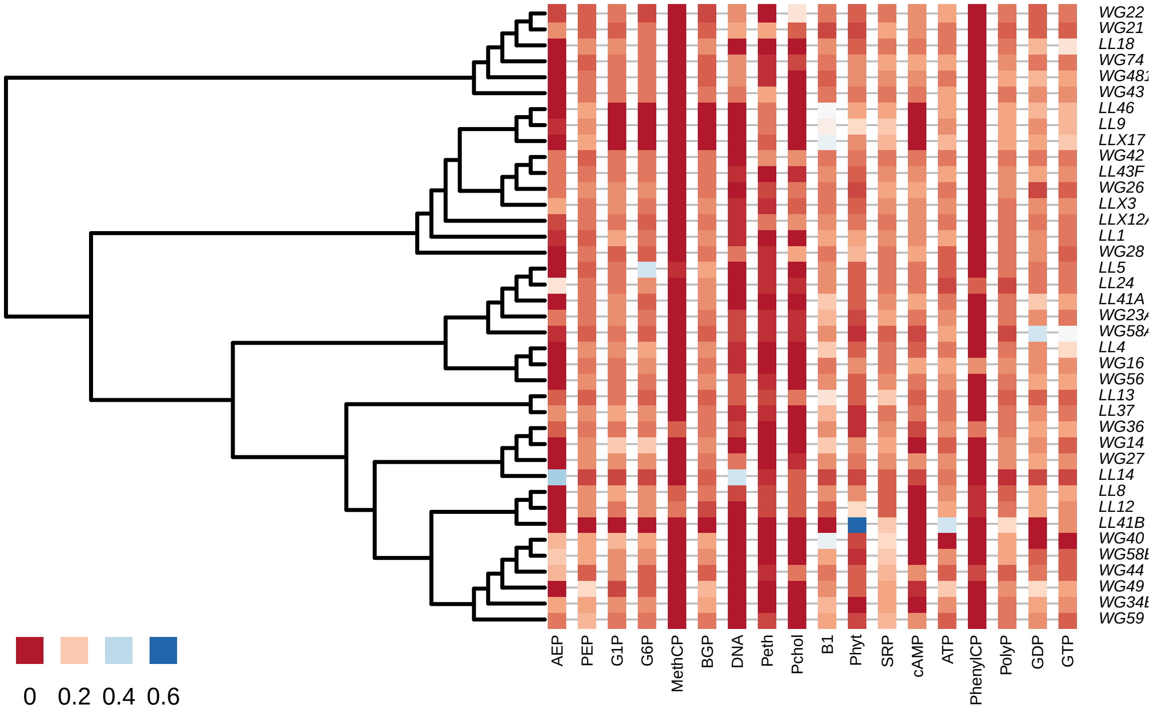
```
nj.tree <- bionj(seq.dist.F84)

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

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

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

```r
plot(nj.rooted)
```

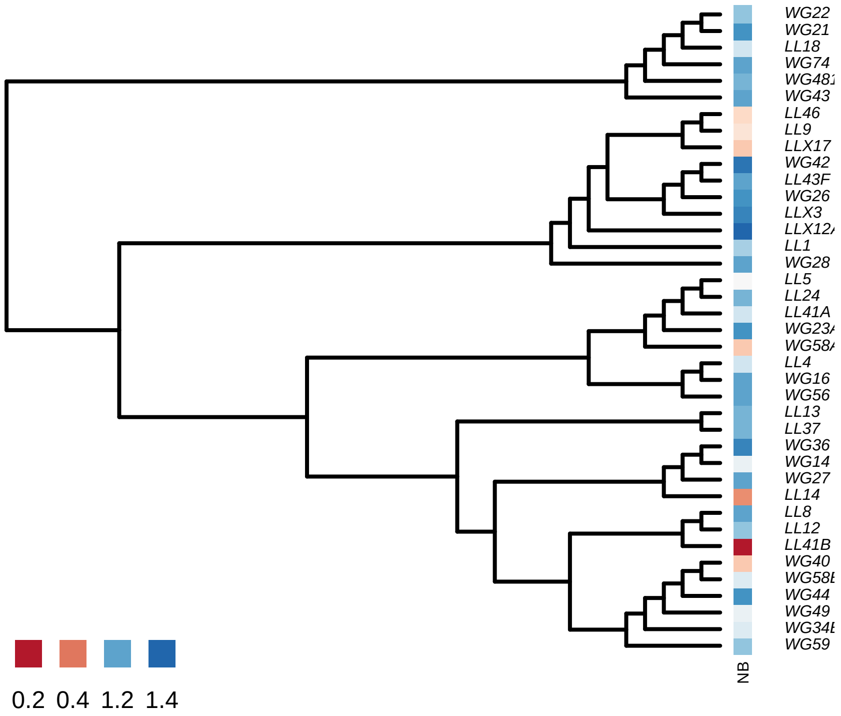


In the R code chunk below, do the following:
1. define a color palette (use something other than "YlOrRd"),
2. map the phosphorus traits onto your phylogeny,
3. map the *nb* trait on to your phylogeny, and
4. customize the plots as desired (use `help(table.phylo4d)` to learn about the options).

```r
mypalette <- colorRampPalette(brewer.pal(9, "RdBu"))

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)
```



```r
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:

a) Develop 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 5a: An organism will experience maximal growth when its preferred phosphorus source is in great abundance as opposed to moderate growth an organism receives when it survives off many different phosphorus sources.
Answer 5b: If an organism is a specialist, it will have a very small niche breadth because it only relies on one or a few resources. On the other hand, if an organism is a generalist, it will have a much larger niche breadth because it relies on numerous resources. When looking at growth rate, specialists would likely have a much larger growth rate if its preferred resource(s) is/are avaiable, than a generalist who also has many of their resources present. However, if a specialists preferred resource is not present, it will have a much smaller growth rate than generalists who does have resources.

6) HYPOTHESIS TESTING

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.

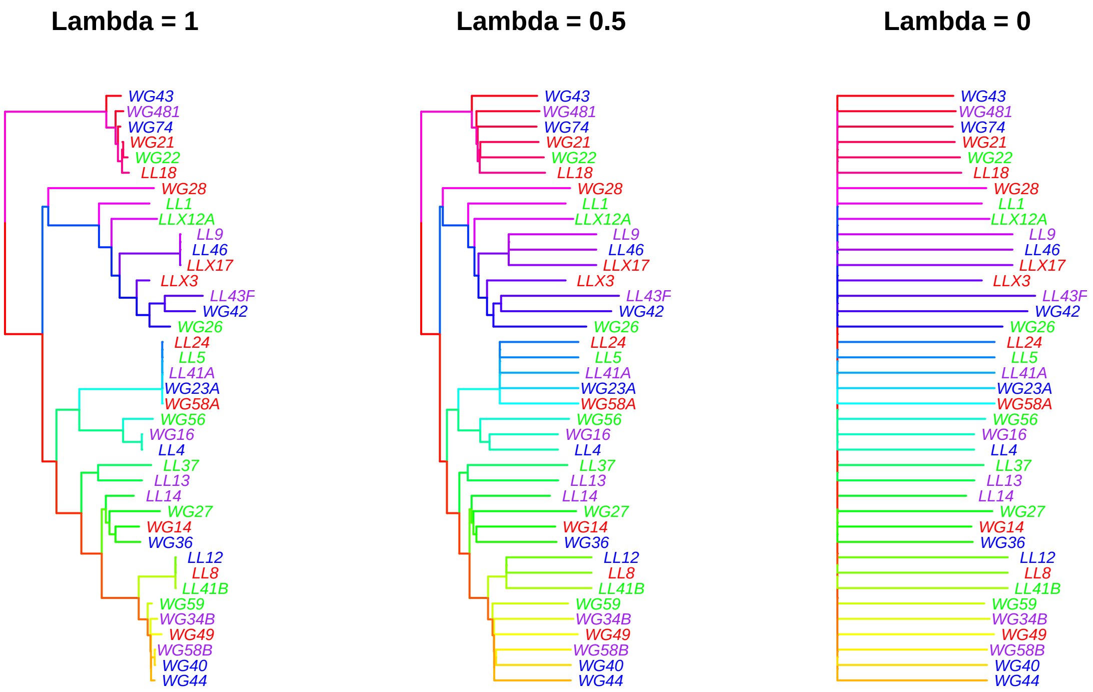
```
library(phytools)

nj.lambda.5 <- lambdaTree(nj.rooted, lambda = 0.5)
```

```
## Warning in .deprecate("lambdaTree", "rescale.phylo"): 'lambdaTree' is being
## deprecated: use 'rescale.phylo' instead
```

```
nj.lambda.0 <- lambdaTree(nj.rooted, lambda = 0)
```

```
## Warning in .deprecate("lambdaTree", "rescale.phylo"): 'lambdaTree' is being
## deprecated: use 'rescale.phylo' instead
```

```
edge_colors <- rainbow(nrow(nj.rooted$edge))
tip_colors <- rep(c("blue", "red", "green", "purple"), length.out = length(nj.rooted$tip.label))

layout(matrix(c(1, 2, 3), 1, 3), width = c(1, 1, 1))

plot(nj.rooted, main = "Lambda = 1", cex = 0.7, adj = 0.5, edge.color = edge_colors,
     tip.color = tip_colors)
plot(nj.lambda.5, main = "Lambda = 0.5", cex = 0.7, adj = 0.5, edge.color = edge_colors,
     tip.color = tip_colors)
plot(nj.lambda.0, main = "Lambda = 0", cex = 0.7, adj = 0.5, edge.color = edge_colors,
     tip.color = tip_colors)
```



Lambda = 1

Lambda = 0.5

Lambda = 0

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 = 0.006975
##  sigsq = 0.108060
##  z0 = 0.657697
##
##  model summary:
##  log-likelihood = 21.503414
##  AIC = -37.006827
##  AICc = -36.321113
##  free parameters = 3
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 44
##  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 = 0.965171
##  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 = 85
##  frequency of best fit = 0.850
##
##  object summary:
##  'lik' -- likelihood function
##  'bnd' -- bounds for likelihood search
##  'res' -- optimization iteration summary
##  'opt' -- maximum likelihood parameter estimates
```

```r
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 transformed tree does not show any phylogenetic relationships, unlike the untransformed tree that does. The transformed tree has no branches, which means there is no phylogenetic signal in the tree. The transformed tree indicates that the trait you are analyzing has no historic or evolutionary history. The untransformed tree has branching and nodes, and shows an evolutionary history. ***Answer 6b***: The AIC values for the untransformed and transformed model are both approximately -37. AIC values are calculated to determine which model best fits a data set; however, to deem one model better than the other, the difference in AIC values must be at least two. Given this information and the fact that the AIC difference is less than 1, the models are considered equivalent. Therefore, it would not matter which model you choose to visualize the data.
> ***Answer 6c***: The AIC scores only allow you to determine if one model is better than the other and does not give an indication if there is a phylogenetic signal. However, by performing a likelihood ratio test, you can determine if there is a phylogenetic signal. The likelihood ratio test resulted in a p-value of 0.965994, which indicates that there is a not a phylogenetic signal present.

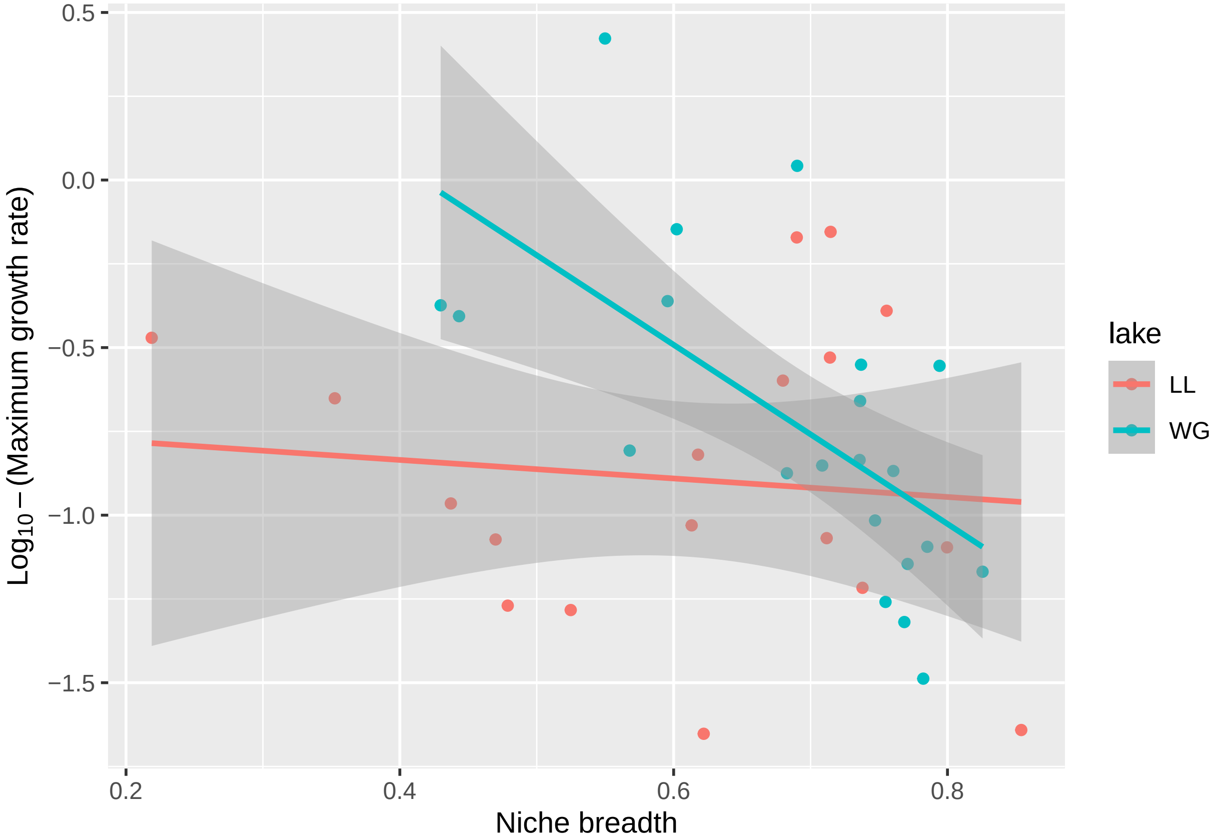
7) PHYLOGENETIC REGRESSION

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
```

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

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

```r
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
```

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 7a***: Shared evolutionary history must be accounted for in phylogenetic regressions because if not, you are assuming that each trait is independent of one another and we know that is not typically the case. If you do not take it into account, your statistical analysis will be untrue and not an accurate explanation of your data. ***Answer 7b***: A standard linear regression assumes that observations are made independent of one another and come from a normal distribution and follows a bell curve. This is different than a phylogenetic regression because it takes into account the evolutionary history that traits share, and are therefore not independent of one another, along with the phylogenetic signal present in the residuals. Residuals in a phylogenetic regression are also explained by a covariance matrix, as opposed to the bell curve.
> ***Answer 7c***: For the simple linear regression, the slope for NB:lakeWG was -2.3958 while the phylogeny regression resulted in a slope of -1.966. While both slopes indicate a negative correlation, the change in slope is important. Going from -2.3958 to -1.966 indicates that evolutionary hisotry explains some of the variance. If evolutionary history did not have an impact on the relationship between growth rate and niche breadth, the slopes would not have changed. Additionally, the AIC value for the linear regression was 48.413 while the phylogeny regression AIC value was much lower at 41.07572. A lower AIC value indicates a better fit; therefore, accounting for evolutionary history improved the fit. ***Answer 7d***: While I am not sure if these variables can be 100% explained by underlying phylogeny, I think social structure and infant care in primates have a strong relationship due to underlying phylogeny.

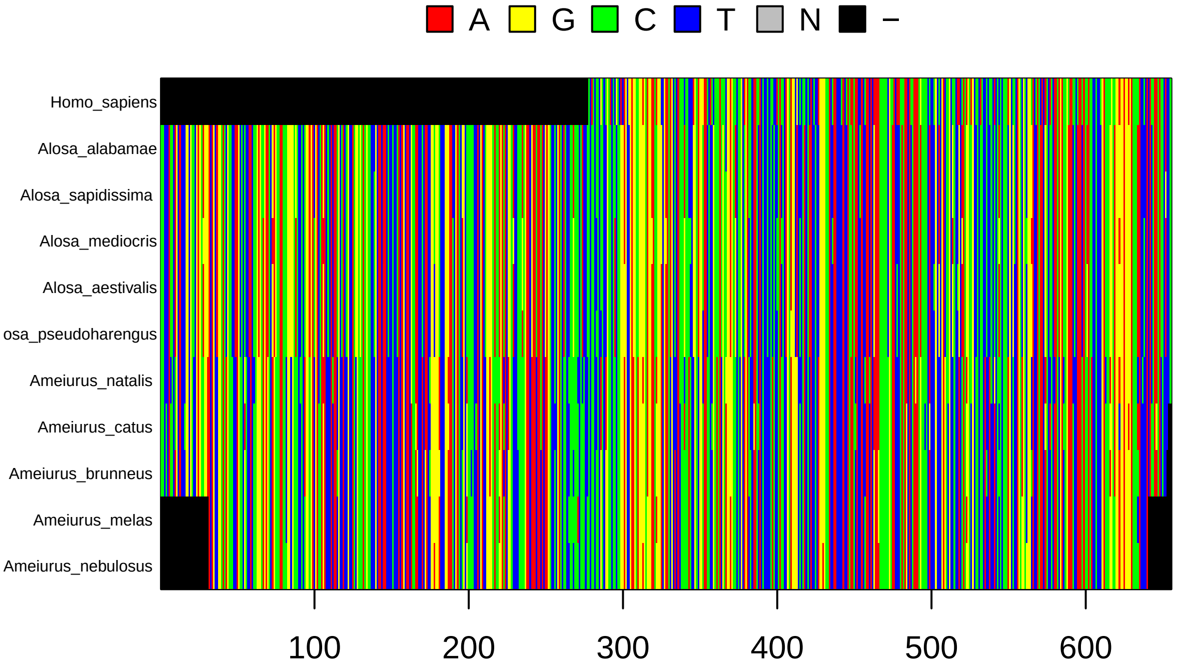
7) SYNTHESIS

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 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.

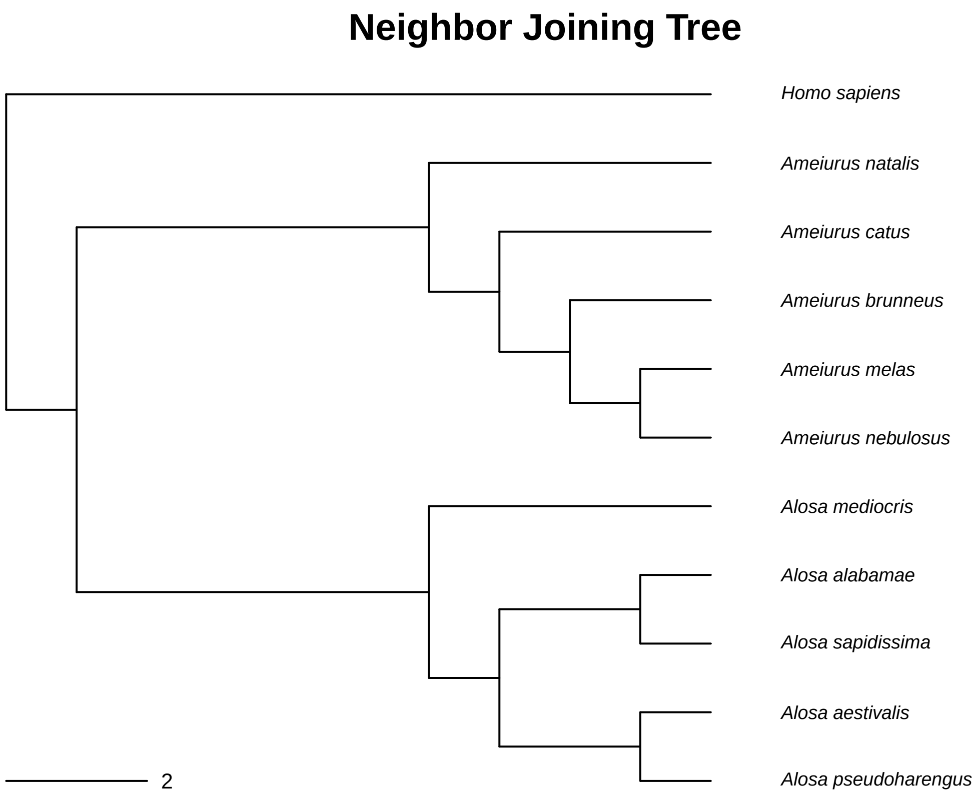
But before your team proceeds, you need to give some thought to which gene you want to focus on. For microorganisms like the bacteria we worked with above, many people use the ribosomal gene (i.e., 16S rRNA). This has many desirable features, including it is relatively long, highly conserved, and identifies 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?

```
fish_data <- read.csv("/cloud/project/QB2025_Brown/Fish_Dataset.csv")

fish_datanew <- cbind(fish_data[, 2:3], fish_data[, 5], fish_data[, 7:9], fish_data[, 23:658])

colnames(fish_datanew)[3] = "Water_Temp"
#fish_datanew

water.mod <- model.matrix(~ Water_Temp + pH + Nitrate_ug_L + TotalPhosphorus_ug_L, as.data.frame(fish_da

only.species <- fish_datanew[, 7:642]

qbfish <- readDNAStringSet("./qbfish.fasta", format = "fasta")

fish.aln <- msaMuscle(qbfish)

savefish.aln <- msaConvert(fish.aln, type = "bios2mds::align")
export.fasta(savefish.aln, "./qbfish.afa")

#visualize alignment

fish.DNAbin <- as.DNAbin(fish.aln)

window_fish <- fish.DNAbin[, 0:655]
image.DNAbin(window_fish, cex.lab = 0.50)
```

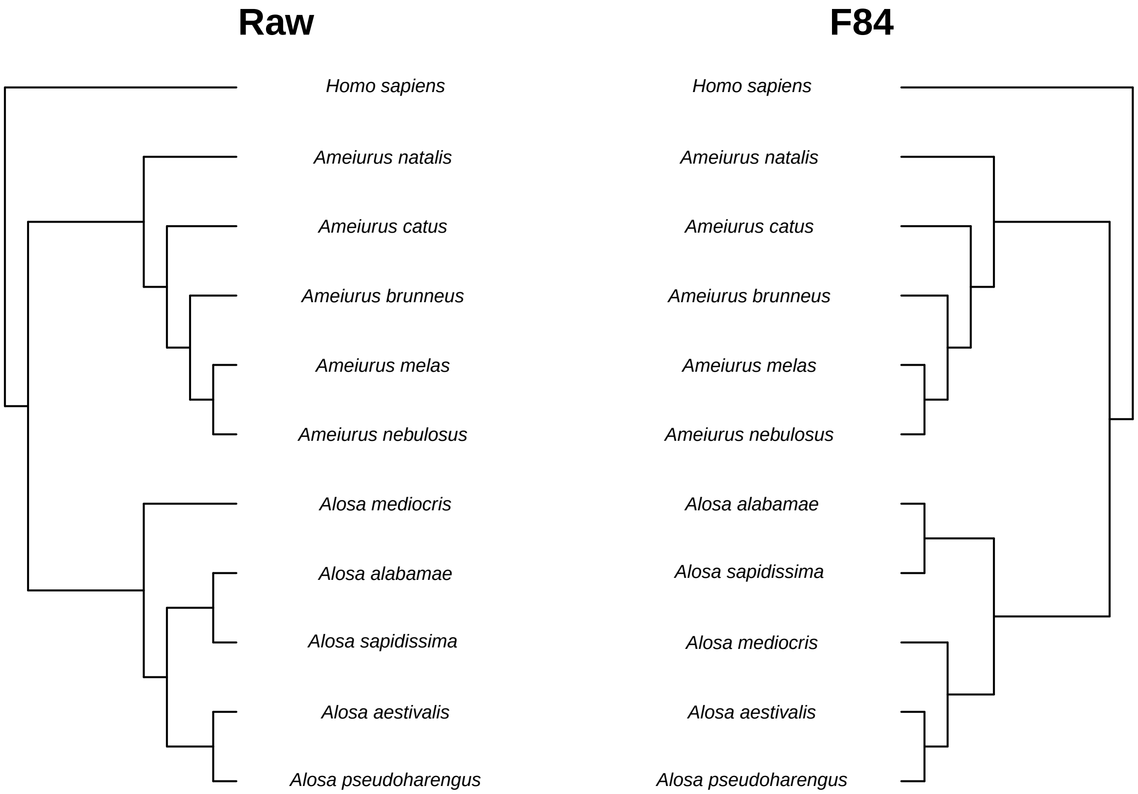


```r
# Neighbor joining tree

fish.dist.raw <- dist.dna(fish.DNAbin, model = "raw", pairwise.deletion = FALSE)

fish.tree <- bionj(fish.dist.raw)

fishgroup <- match("Homo_sapiens", fish.tree$tip.label)

fish.rooted <- root(fish.tree, fishgroup, resolve.root = TRUE)

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



```r
# NJTs with Different Models

fish.dist.F84 <- dist.dna(fish.DNAbin, model = "F84", pairwise.deletion = FALSE)

raw.fish <- bionj(fish.dist.raw)
F84.fish <- bionj(fish.dist.F84)

rawfish.outgroup <- match("Homo_sapiens", raw.fish$tip.label)
F84fish.outgroup <- match("Homo_sapiens", F84.fish$tip.label)

rawfish.rooted <- root(raw.fish, rawfish.outgroup, resolve.root = TRUE)
F84fish.rooted <- root(F84.fish, F84fish.outgroup, resolve.root = TRUE)
```

```
layout(matrix(c(1, 2), 1, 2), width = c(1, 1))
par(mar = c(1, 1, 2, 0))
plot.phylo(rawfish.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(F84fish.rooted, type = "phylogram", direction = "left",
           show.tip.label = TRUE, use.edge.length = FALSE, adj = 0.5,
           cex = 0.6, label.offset = 2, main = "F84")
```



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?

> ***Answer Synthesis***: My partner and I chose to analyze the COI gene in these fish species. Based on current evolutionary history knowledge of the organisms, the charts above seem to align. There are three clades, one belonging to Homo sapiens which was our outlier group, and the other two belonging to the Alosa and Ameiurus genera. The trees correctly graphed these clades, given that we know the Alosa genus are more closely related to one another than they are to the Ameiurus genus (and vice versa for Ameiurus). The F84 tree did place a few taxa differently, but this is expected given that the F84 model accounts for various transition and transversion rates. The placement of the different Alosa and Ameriurus species also aligned with phylogenetic trees created by the Florida Museum of Natural History (https://www.floridamuseum.ufl.edu/fish/catfish/ictaluridae/phylogeny/) and the USDA (chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.srs.fs.usda.gov/pubs/ja/ja_bowen001.pdf). If we wanted to improve our tree, the most beneficial thing we could do would be to bootstrap our tree to allow us to have a statistical understanding of how reliable our tree is.

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)**.