11. Worksheet: Phylogenetic Diversity - Traits

Herbert Sizek; Z620: Quantitative Biodiversity, Indiana University

30 April, 2021

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, April 28$^{\text{th}}$, 2021 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 "*/11.PhyloTraits*" folder, and
4. load all of the required R packages (be sure to install if needed).

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

```
## [1] "D:/GitHub/QB2021_Sizek/2.Worksheets/11.PhyloTraits"
```

```
setwd("D:/GitHub/QB2021_Sizek/2.Worksheets/11.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.

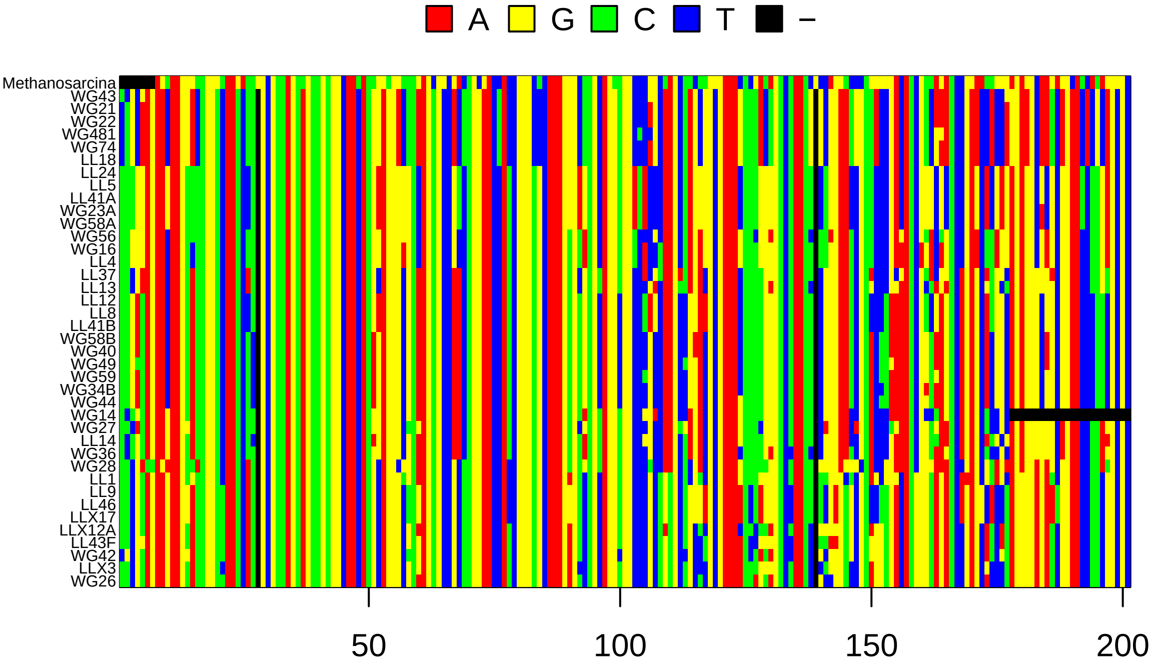
> ***Answer 1***: The AFA file has some alignment spacing in it, while fasta is just the reads/processed reads.

```
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
## Registered S3 method overwritten by 'spdep':
##   method   from
##   plot.mst ape

##
## Attaching package: 'permute'

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

## This is vegan 2.5-7

##
## 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
```

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.

```
read.aln <-read.alignment(file= './data/p.isolates.afa',format='fasta')
p.DNAbin <- as.DNAbin(read.aln)
window <- p.DNAbin[,500:700]
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***: About 600 to 700 bp. ***Answer 2b***: probably between 200 and 700 because it is present in most of the groups for the majority of that region.

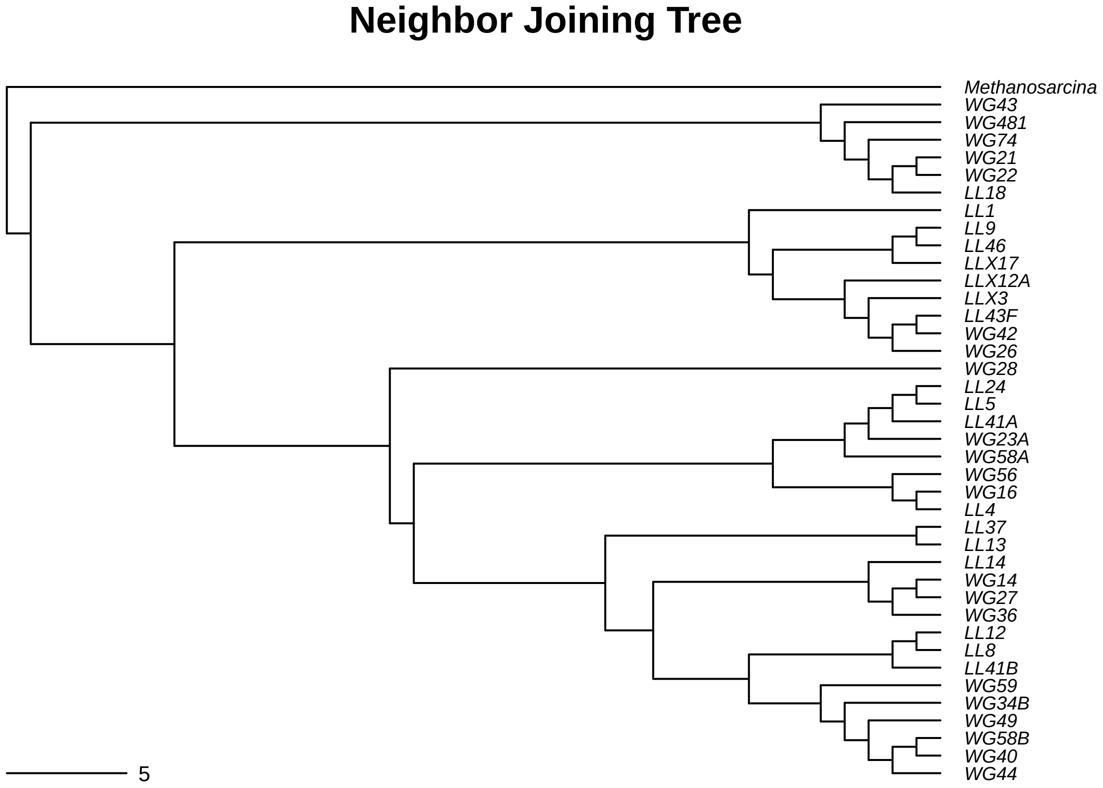
4) MAKING A PHYLOGENETIC TREE

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 = "ri
add.scale.bar(cex=0.7)
```



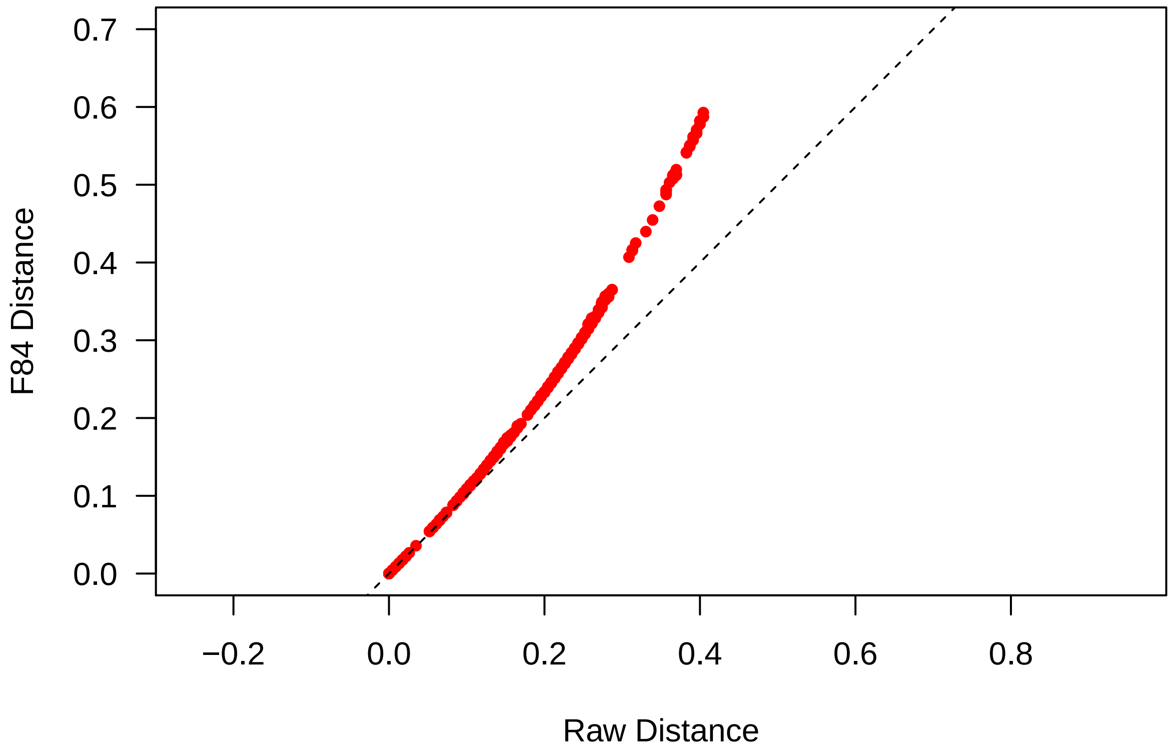
Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

> ***Answer 3***: Biologically the assumed transitions might not have equal evolutionary weight or might have switching behaviors, so depending on the consistency of the sequence ourtcomes would vary. The simplicity makes if fairly fast untill you are comparing distances between lots of sequences. Because it is distance based you could get odd/unclear groupings because of structures: "AAB" "AAA" and "ABA" could be clustered in multiple different ways depending on the implementation choices.

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,

```
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,xlab="Raw Distance",ylab = "F84 Distance
abline(b=1,a=0,lty=2)
```

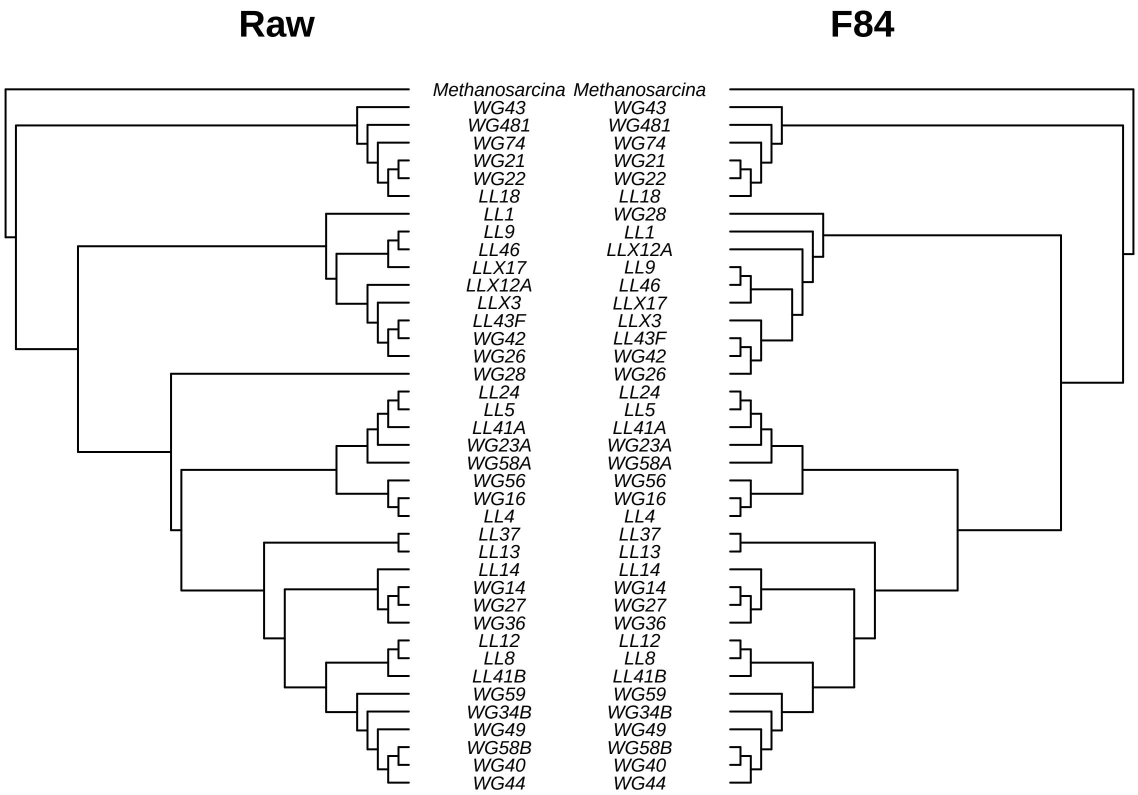


3. make Neighbor Joining trees for both, and

```
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)
```

4. create a cophylogenetic plot to compare the topologies of the trees.

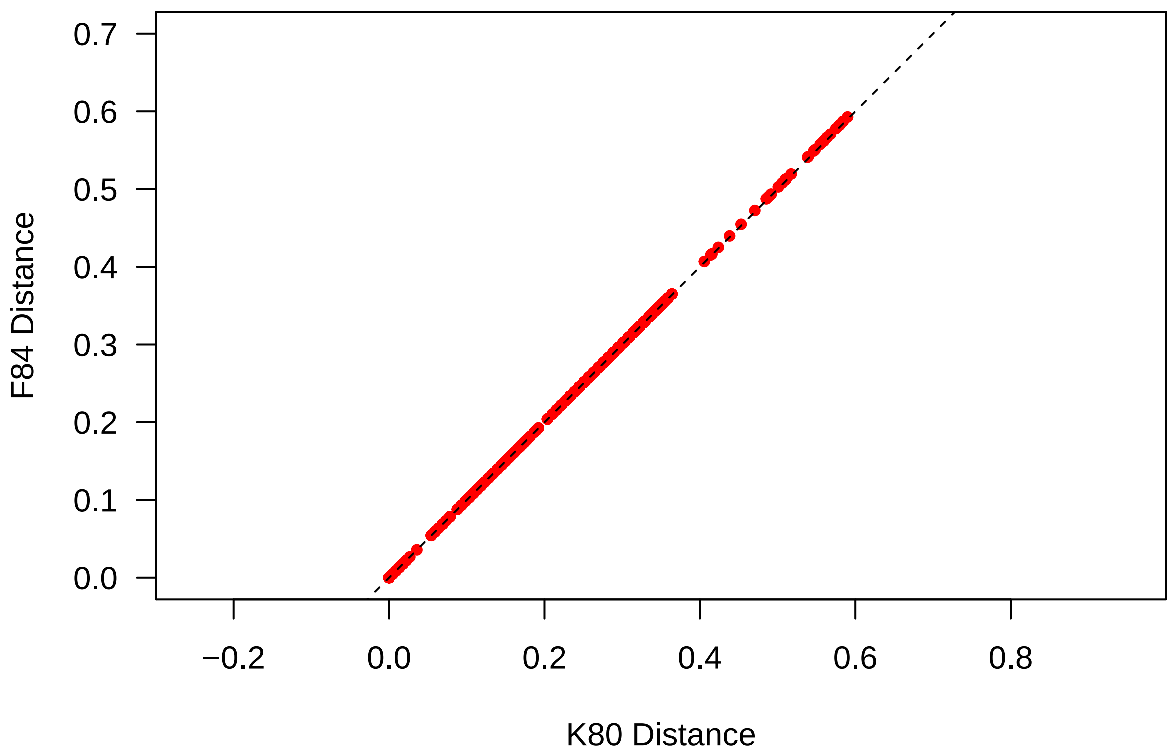
```
layout(matrix(c(1,2),1,2),width = c(1,1))
par(mar = c(1,1,2,0))
plot.phylo(raw.rooted,type="phylogram",direction="right",show.tip.label=TRUE,use.edge.length=FALSE,adj=
par(mar = c(1,0,2,1))
plot.phylo(F84.rooted,type="phylogram",direction="left",show.tip.label=TRUE,use.edge.length=FALSE,adj= 0
```



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,

```
seq.dist.K80 <- dist.dna(p.DNAbin,model="K80",pairwise.deletion = FALSE)
par(mar = c(5,5,2,1)+0.1)
plot(seq.dist.K80,seq.dist.F84,pch =20,col ="red",las =1,asp=1,xlab="K80 Distance",ylab = "F84 Distance
abline(b=1,a=0,lty=2)
```



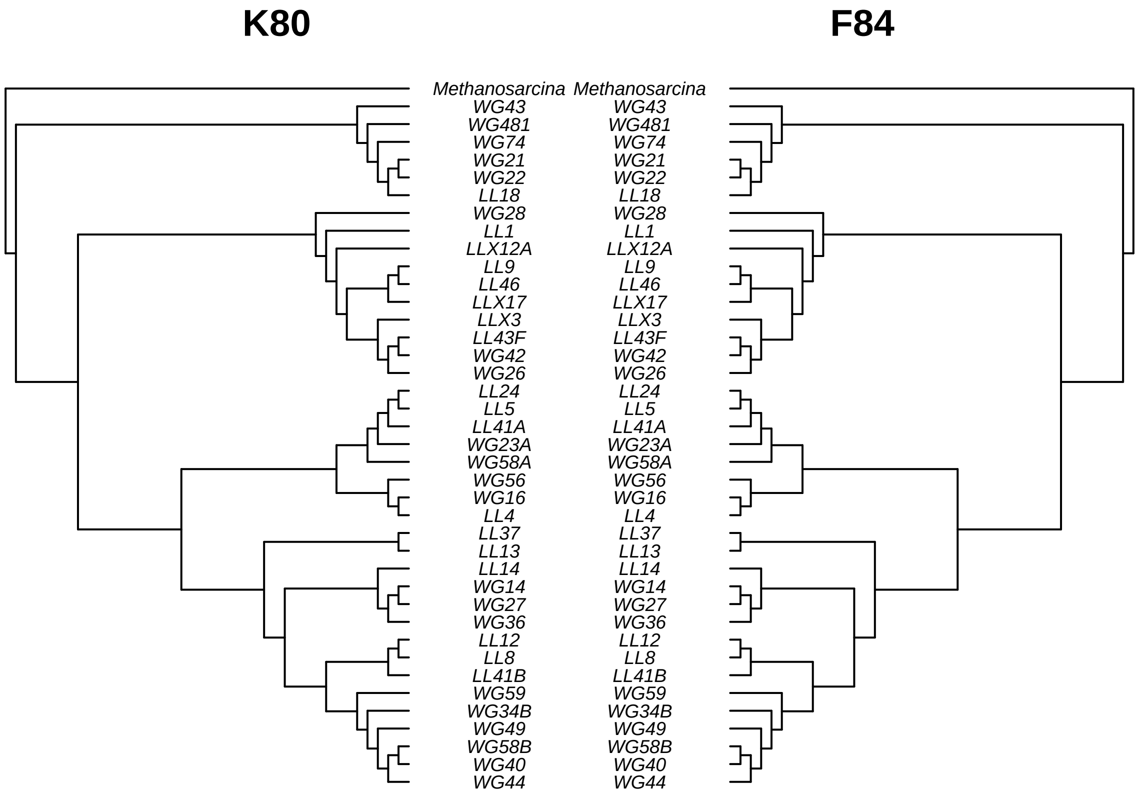
4. make a cophylogenetic plot that compares the topologies of both models, and

5. be sure to format, add appropriate labels, and customize each plot.

```
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)
```

4. create a cophylogenetic plot to compare the topologies of the trees.

```
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=
par(mar = c(1,0,2,1))
plot.phylo(F84.rooted,type="phylogram",direction="left",show.tip.label=TRUE,use.edge.length=FALSE,adj= (
```



Question 4:

a. Describe the substitution model that you chose. What assumptions does it make and how does it compare to the F84 model?

   Answer 4a: The primary difference is on the assumption of frequencies of nucleotides, with the K80 model not accounting for differences in frequencies while F84 does. K80 only accounts for differences in transition mutation frequencies

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.

   Answer 4b: My choice didn't affect the substitution model I used. In this situation it doesn't change the model much, probably indicating that this gene has stayed active. If this were looking at non-coding regions, I would expect that these two models would operate differently.

c. How does your model compare to the *F84* model and what does this tell you about the substitution rates of nucleotide transitions?

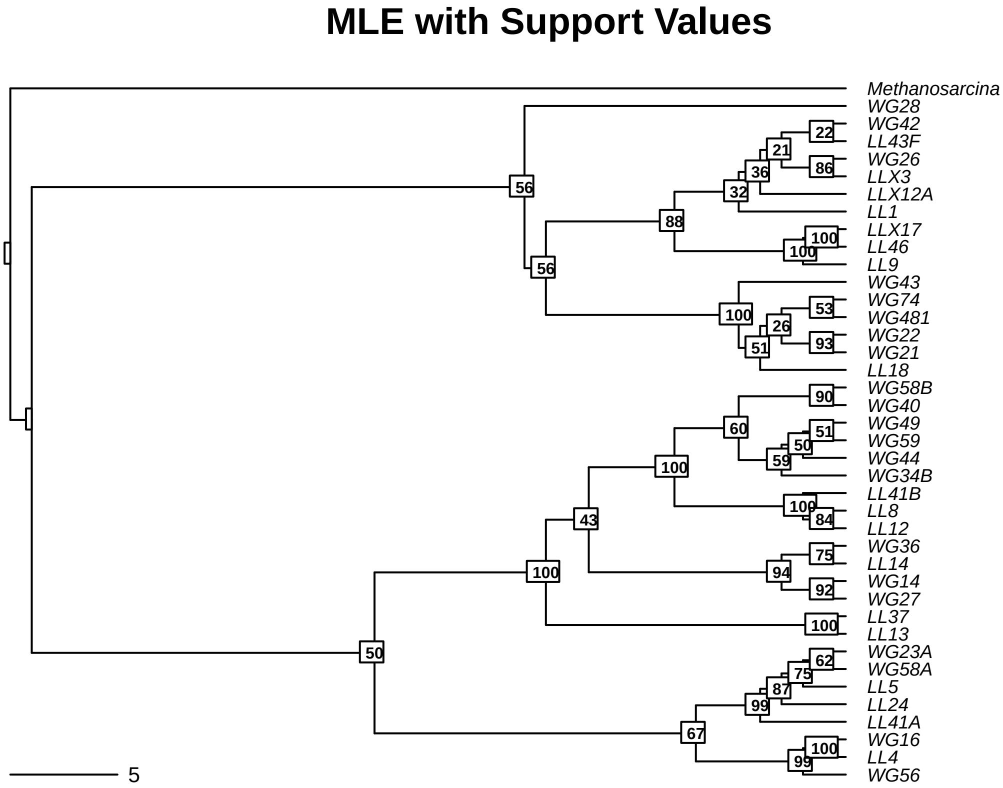
   Answer 4c: The model comparison doesn't inform us on the nucleotide transition. In comparing with the unbiased model, we see that the the distance produced by F84 is larger which suggests that there are more unlikely transformations (G->A or C->T) than likely transitions (the inverse).

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")
```

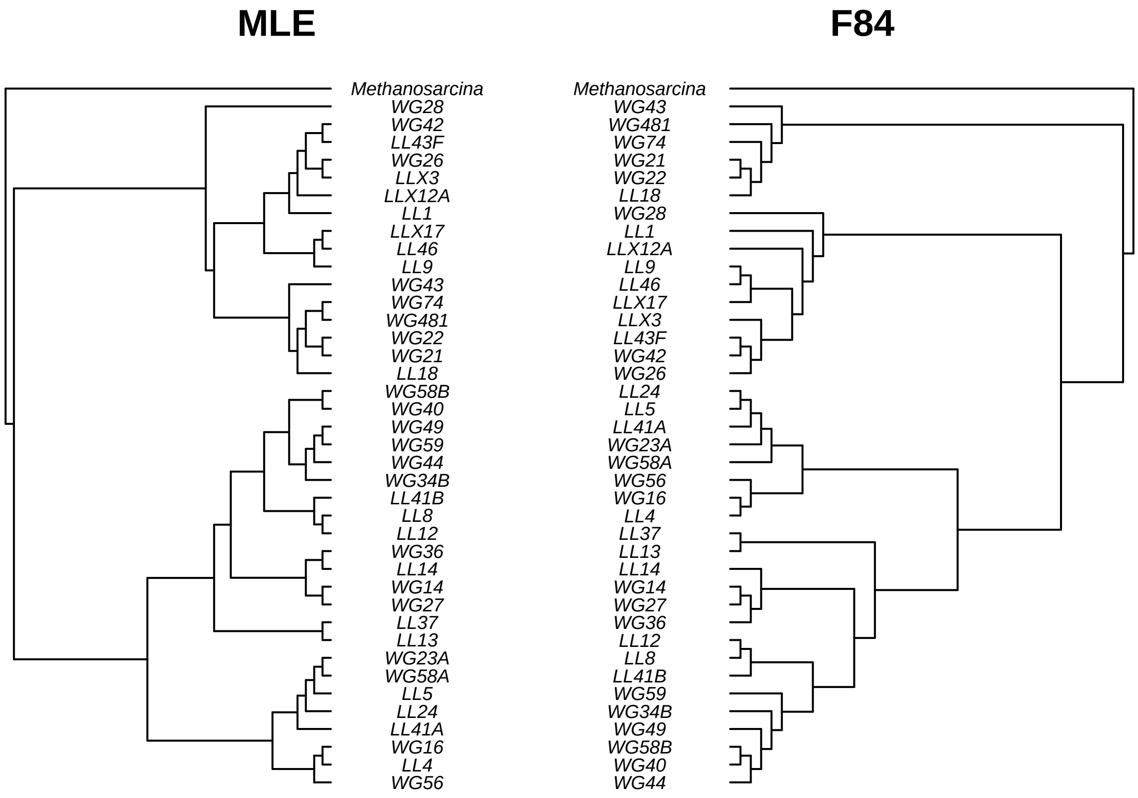
```
par(mar = c(1,1,2,1)+0.1)
plot.phylo(ml.bootstrap,type = "phylogram",direction="right",show.tip.label=TRUE,use.edge.length = FALS
add.scale.bar(cex = 0.7)
nodelabels(ml.bootstrap$node.label,font=2,bg = "white",cex=0.5)
```



Question 5:

a) How does the maximum likelihood tree compare the to the neighbor-joining tree in the handout?

```
layout(matrix(c(1,2),1,2),width = c(1,1))
par(mar = c(1,1,2,0))
plot.phylo(ml.bootstrap,type="phylogram",direction="right",show.tip.label=TRUE,use.edge.length=FALSE,ad
par(mar = c(1,0,2,1))
plot.phylo(F84.rooted,type="phylogram",direction="left",show.tip.label=TRUE,use.edge.length=FALSE,adj= 
```



If the plots seem to be inconsistent with one another, explain what gives rise to the differences.

Answer 5a: It is hard to tell what components are the same versus different because of the alignments, but they do appear to be different in both grouping and in distances between groups. This makes sense as njt are just calculating the correlations between groups, while the MLE is accounting for the actual structure of sequences and the relationships between structures.

b) Why do we bootstrap our tree?

Answer 5b: MLE calculates local maxima, by changing the data composition, we are likely to determine how stable those local maxima are, reducing the likelihood that we have a tree that is stuck due to the generation process.

c) What do the bootstrap values tell you?

Answer 5c: They tell us how many times the underlying tree algorithm agrees or disaggrees with the structure of the reference tree.

d) Which branches have very low support?

Answer 5d: The group near the top, with WG42 to LL1 have low support.

e) Should we trust these branches?

Answer 5e: I would trust that they are associated with each other in some way, but it would be worth looking at how other estimations handle their relationships if we need more certain information on the relationships between sample, perhaps we could focus on the group between WG28 and LL9 with other methods to determine if there is an issue with the ML estimation.

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,

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

2. create a function that calculates niche breadth (nb), and

```
nb.levins <- function(p_xi=""){
  # p = sum(p_xi^2)
  # return( 1/(length(p_xi)*p))
  p = 0
  for (i in p_xi){
    p=p+i^2
  }
  nb = 1/(length(p_xi)*p)
}
```

3. use this function to calculate nb for each isolate.

```
nb <- as.matrix(nb.levins(p.growth.std))
rownames(nb) <- row.names(p.growth)
colnames(nb) <- c("NB")
```

C. Visualizing Traits on Trees

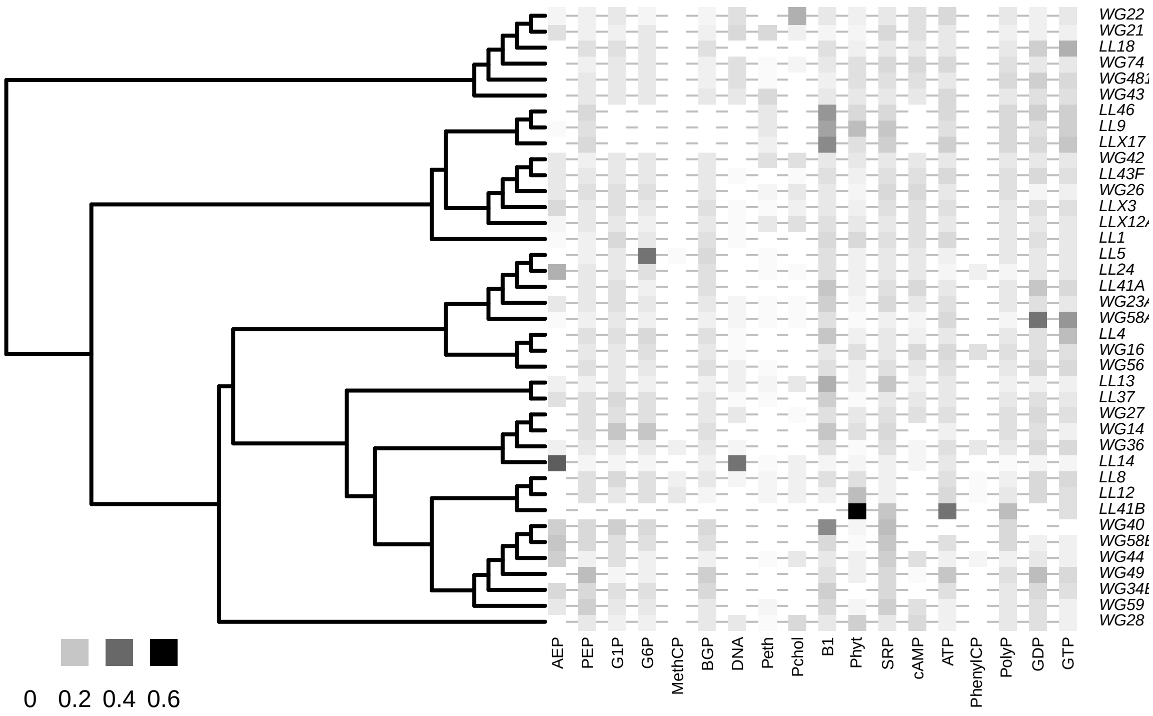
In the R code chunk below, do the following:
1. pick your favorite substitution model and make a Neighbor Joining tree,
2. define your outgroup and root the tree, and
3. remove the outgroup branch.

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

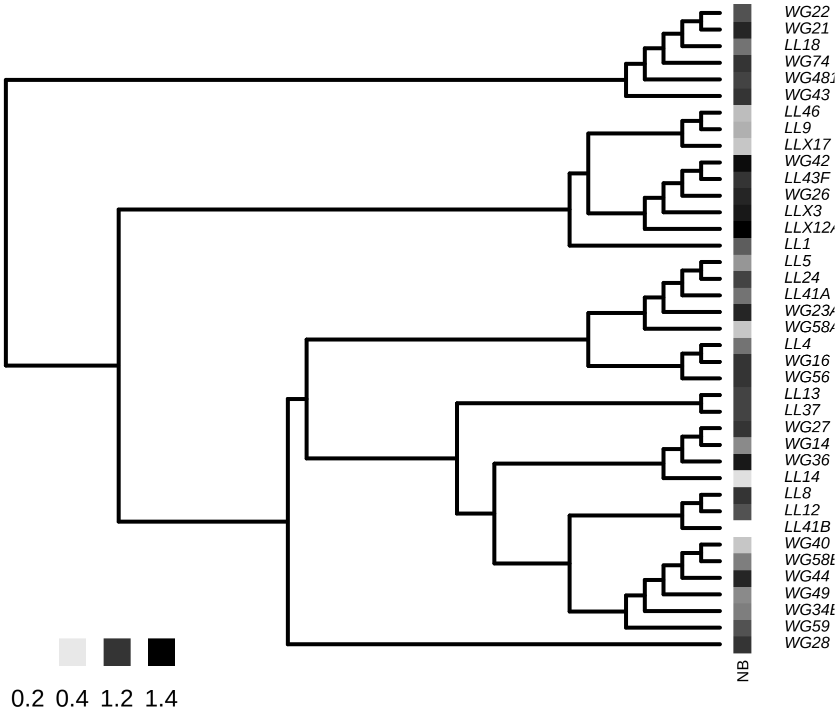
In the R code chunk below, do the following:
1. define a color palette (use something other than "YlOrRd"),

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).

```
mypallette <-colorRampPalette(brewer.pal(9,"Greys"))
par(mar = c(1,1,1,1)+0.1)
x <- phylo4d(nj.rooted,p.growth.std)
table.phylo4d(x,treetype="phylo",symbol="colors",show.node=TRUE,cex.label=0.5,scale=FALSE,use.edge.leng
```



```
mypallette <-colorRampPalette(brewer.pal(9,"Greys"))
par(mar = c(1,5,1,5)+0.1)
x.nb <- phylo4d(nj.rooted,nb)
table.phylo4d(x.nb,treetype="phylo",symbol="colors",show.node=TRUE,cex.label=0.5,scale=FALSE,use.edge.le
```



Question 6:

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

   Answer 6a: Generalists can change resource supplies, so they have less variable growth rates given changing conditions. Specialists cannot change resource supplies, so their growth rate is dependent on resource availibility.

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

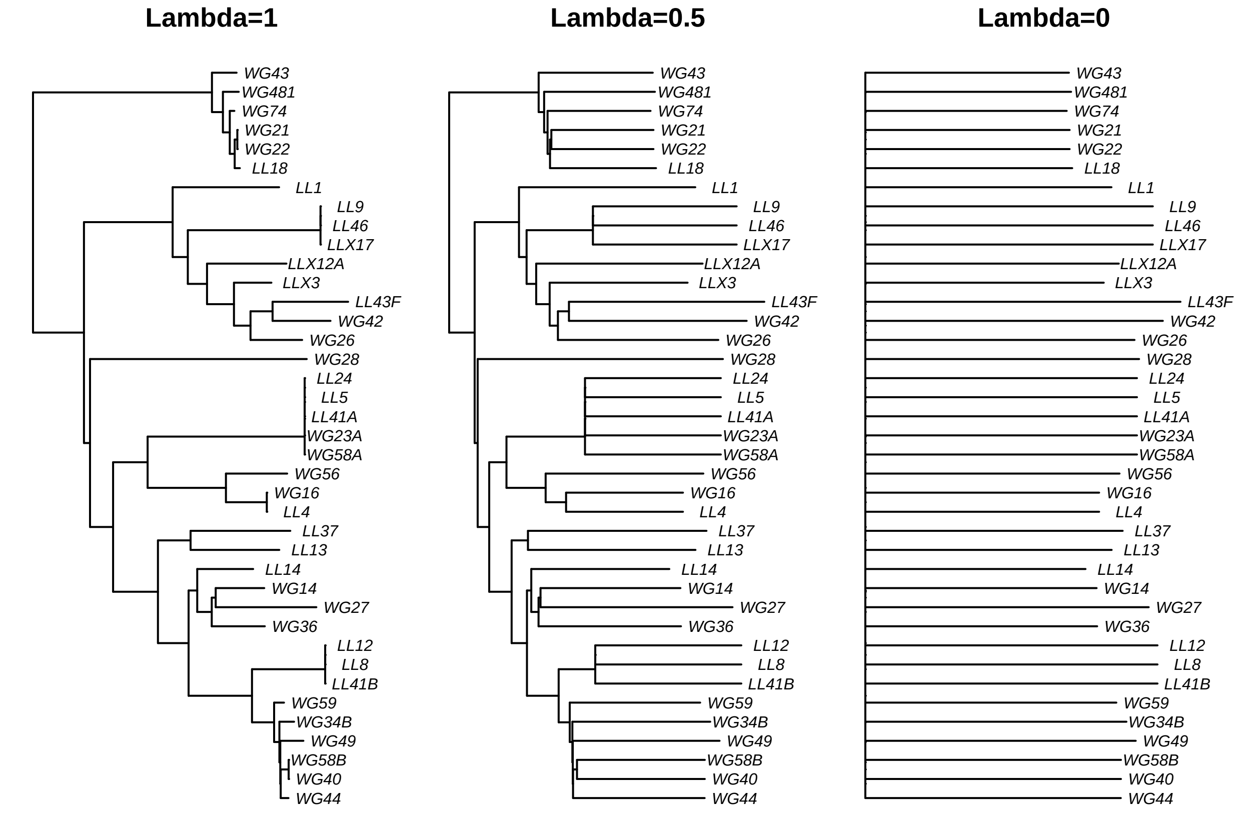
   Answer 6b: Generalists should have a low nb and have less variable growth rates across environments.

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.

```r
nj.lambda.5 <- rescale(nj.rooted,"lambda", 0.5)
nj.lambda.0 <- rescale(nj.rooted,"lambda", 0)
layout(matrix(c(1,2,3), 1,3),width=c(1,1,1))
par(mar=c(1,.5,2,.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.

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

```
## GEIGER-fitted comparative model of continuous data
##  fitted 'lambda' model parameters:
##  lambda = 0.061976
##  sigsq = 0.140021
##  z0 = 0.664039
##
##  model summary:
##  log-likelihood = 21.456738
##  AIC = -36.913476
##  AICc = -36.227761
##  free parameters = 3
##
```

```
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 50
##  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 = 0.139249
##  z0 = 0.656203
##
##  model summary:
##  log-likelihood = 21.399126
##  AIC = -36.798252
##  AICc = -36.112537
##  free parameters = 3
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 0
##  number of iterations with same best fit = 84
##  frequency of best fit = 0.84
##
##  object summary:
##  'lik' -- likelihood function
##  'bnd' -- bounds for likelihood search
##  'res' -- optimization iteration summary
##  'opt' -- maximum likelihood parameter estimates
```

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).

> ***Answer 7a***: 0.06 to 0.00

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)?

> ***Answer 7b***: Basically the same

c. Does this result suggest that there's phylogenetic signal?

> ***Answer 7c***: Nope does not suggest that ther is 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.

```r
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:dim(p.phylosignal)[2]){
  x<-as.matrix(p.growth.std[,i,drop=FALSE])
  out <- phylosignal(x,nj.rooted)
  p.phylosignal[1:5,i]<-round(t(out),3)
}
```

```
## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used
## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used

## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used
```

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

```
##                   AEP      PEP      G1P      G6P   MethCP      BGP      DNA
## K               0.000    0.000    0.000    0.000    0.000    0.000    0.000
## PIC.var.obs  4373.159  664.095  948.942 5924.730  350.894  536.105  259.088
## PIC.var.mean 8148.688 1588.081 1859.872 3745.124  515.416 1713.547 5177.150
## PIC.var.P       0.245    0.063    0.111    0.744    0.330    0.032    0.002
## PIC.var.z      -0.818   -1.359   -1.192    0.884   -0.485   -1.660   -1.323
## PIC.P.BH        0.628    0.284    0.400    0.788    0.628    0.192    0.027
##                  Peth    Pchol       B1     Phyt      SRP     cAMP      ATP
## K               0.000    0.000    0.000    0.000    0.000    0.000    0.000
## PIC.var.obs  1446.463 2368.387 3517.019 9240.370 1307.026  690.724 4040.138
## PIC.var.mean 1806.087 3368.353 5340.604 9152.763 1608.799 3053.090 3143.385
## PIC.var.P       0.372    0.384    0.248    0.574    0.329    0.003    0.610
## PIC.var.z      -0.437   -0.558   -0.781    0.011   -0.545   -2.624    0.391
## PIC.P.BH        0.628    0.628    0.628    0.720    0.628    0.027    0.720
##              PhenylCP    PolyP      GDP      GTP
## K               0.000    0.000    0.000    0.000
## PIC.var.obs  1224.018 1126.345 4473.879 2721.768
## PIC.var.mean  751.017 1217.261 3631.614 2973.652
## PIC.var.P       0.825    0.488    0.640    0.462
## PIC.var.z       1.008   -0.164    0.388   -0.184
## PIC.P.BH        0.825    0.676    0.720    0.676
```

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

```
## Warning in if (dataclass == "data.frame") {: the condition has length > 1 and
## only the first element will be used
```

```
signal.nb
```

```
##              K PIC.variance.obs PIC.variance.rnd.mean PIC.variance.P
## 1 4.187392e-06         49966.79              50608.06          0.521
##   PIC.variance.Z
## 1    -0.03271668
```

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***: No, all values of K are around zero suggesting that the phylogeny and environmental conditions are randomly dispersed. ***Answer 8b***: Nope (clustering is greater than 1, less than one is overdispersed.)

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)
```

```
##
## 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 :  1000
##
## Estimated D :  0.4191971
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.003
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.045
```

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

```
##
## 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 :  1000
##
## Estimated D :  0.8866962
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.306
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.009
```

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

```
##
## 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 :  1000
##
## Estimated D :  0.605707
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.025
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.006
```

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

```
##
## 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 :  1000
##
## Estimated D :  0.1106232
## Probability of E(D) resulting from no (random) phylogenetic structure :  0
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.356
```

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?
   > The issue with this framing is that the p-value can only tell you if something is different from a distribution (0.05 means that only 5% of the distribution of the null is farther out than the real measure). We would have to establish some metric in this space to say that these are the same. I don't know how to do this.

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

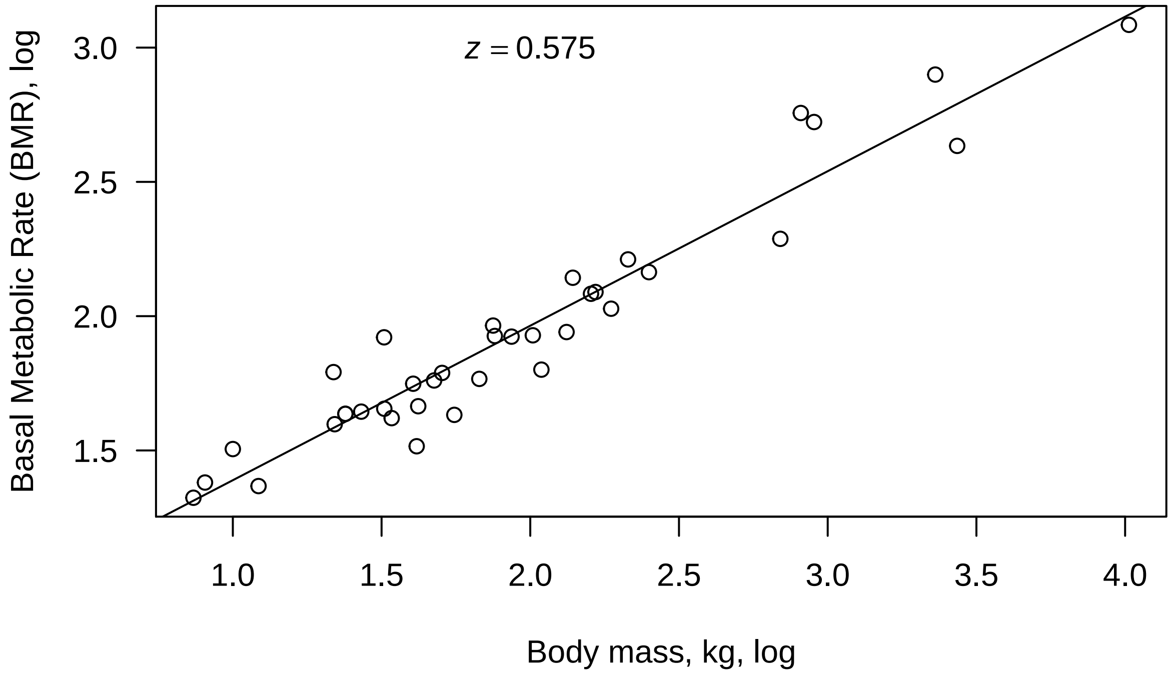
c. Discuss what factors might give rise to differences between the metrics.
   > ***Answer 9a***:
   > PhenylCP overlydispersed most likely > AEP ? kinda overdispersed but not really > cAMP not really over dispersed, randomly clumped maybe. > ***Answer 9b***:
   > Similar in that they are inconclusive that there is sructure. > ***Answer 9c***:
   > Both look at brownian motion on a tree, however K is operating in a continous space, with covarience, while D is operating in a discrete space, with random shuffling.

7) PHYLOGENETIC REGRESSION

In the R code chunk below, do the following:
1. Load and clean the mammal phylogeny and trait dataset, 2. Fit a linear model to the trait dataset, examining the relationship between mass and BMR, 2. Fit a phylogenetic regression to the trait dataset, taking into account the mammal supertree

```
mammal.Tree <- read.tree("./data/mammal_best_super_tree_fritz2009.tre")
mammal.data <- read.table("./data/mammal_BMR.txt",sep="\t",header=TRUE)
mammal.data <- mammal.data[,c("Species","BMR_.mlO2.hour.","Body_mass_for_BMR_.gr.")]
mammal.species <-array(mammal.data$Species)
pruned.mammal.tree <- drop.tip(mammal.Tree,mammal.Tree$tip.label[-na.omit(match(mammal.species,mammal.da
pruned.mammal.data <- mammal.data[mammal.data$Species %in% pruned.mammal.tree$tip.label,]
rownames(pruned.mammal.data) <- pruned.mammal.data$Species
```

```
fit <- lm(log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.), data=pruned.mammal.data)
plot(log10(pruned.mammal.data$Body_mass_for_BMR_.gr.),log10(pruned.mammal.data$BMR_.mlO2.hour.), las =1
abline(a=fit$coefficients[1],b=fit$coefficients[2])
b1 <- round(fit$coefficients[2],3)
eqn <- bquote(italic(z)==.(b1))
text(2,3,eqn)
```

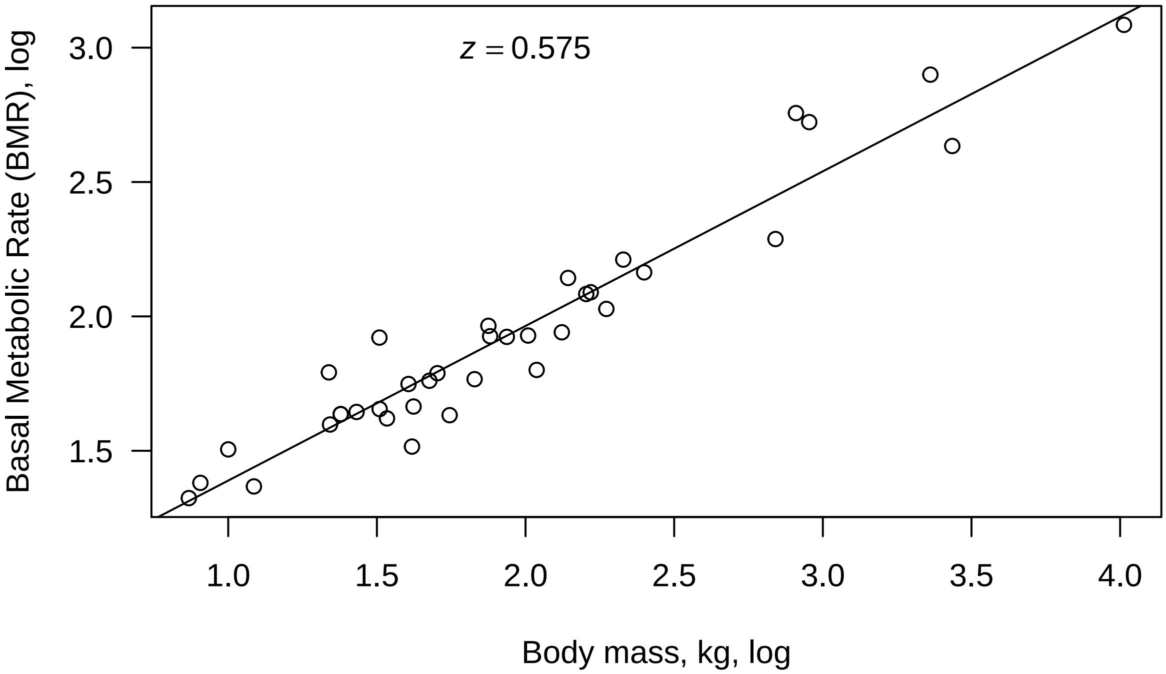


```
fit.phy <- phylolm(log10(BMR_.mlO2.hour.)~ log10(Body_mass_for_BMR_.gr.),data=pruned.mammal.data, prune
```

```
## Warning in phylolm(log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.), :
## will drop from the tree 540 taxa with missing data
```

```
## Warning in phylolm(log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.), : the estimation of lambd
##                            You may change the bounds using options "upper.bound" and "lower.bound".
```

```
plot(log10(pruned.mammal.data$Body_mass_for_BMR_.gr.),log10(pruned.mammal.data$BMR_.mlO2.hour.),las = 1
abline(a=fit.phy$coefficients[1],b=fit.phy$coefficients[2])
b1.phy <- round(fit.phy$coefficients[2],3)
eqn <- bquote(italic(z)==.(b1.phy))
text(2,3,eqn)
```



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 10a: Evolutionary history causes correlation between variables of specific species, so we could overestimate the impact of two closely related species. ***Answer 10b***: The residuals are assumed to be dependent on the tree structure and not randomly distributed ***Answer 10c***: No change, but I think there is an issue with how I'm coding it. ***Answer 10d***: If there are three clusters of points distributed along the two variables of interest, and the phylogeny was split into the same three groups, the correlation could fall away.

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.

```
read.GenBank("NC_051960.1")
```

```
## 1 DNA sequence in binary format stored in a list.
##
## Sequence length: 155684
##
## Label:
## NC_051960.1
##
## Base composition:
##     a     c     g     t
## 0.307 0.193 0.186 0.314
## (Total: 155.68 kb)
```

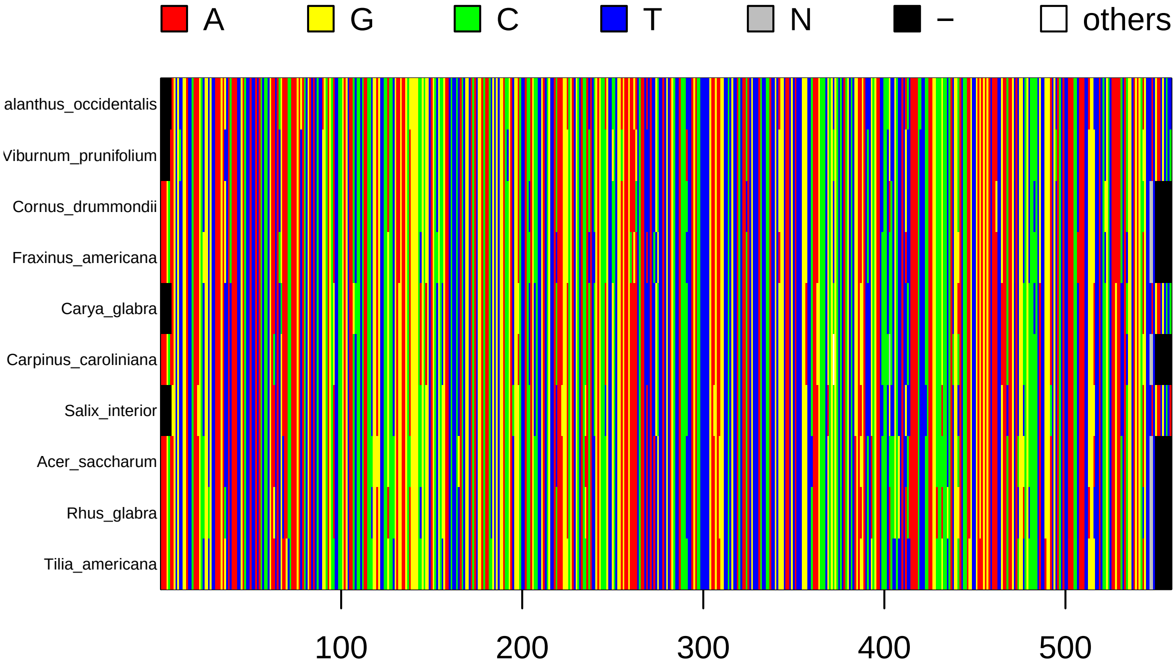
But before your team proceeds, you need to give some thought to which gene you want to focus on. For microorganisms like the bacteria we worked with above, many people use the ribosomal gene (i.e., 16S rRNA). This has many desirable features, including it is relatively long, highly conserved, and identifies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

> We're going to use a chorloplast gene, ribulose which is important in the chloroplast and fairly well conserved.

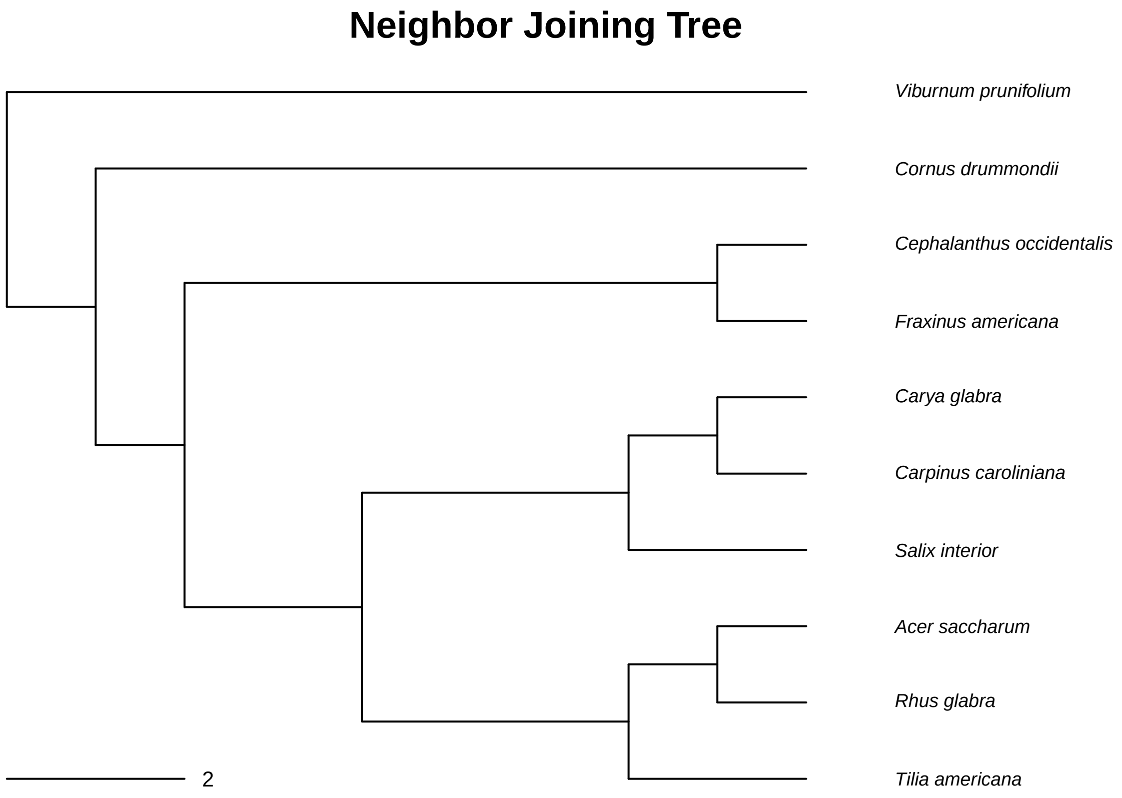
After you identify an appropriate gene, download sequences and create a properly formatted fasta file. Next, align the sequences and confirm that you have a good alignment. Choose a substitution model and make a tree of your choice.

> can't figure out how to do sequence alignment in r. Did online

```
read.aln <-read.alignment(file= './data/woodytreephylaln.fasta',format='fasta')
p.DNAbin <- as.DNAbin(read.aln)
window <- p.DNAbin
image.DNAbin(window,cex.lab=0.50)
```



```
seq.dist.raw <- dist.dna(p.DNAbin,model="raw",pairwise.deletion = FALSE)
nj.tree <- bionj(seq.dist.raw)
outgroup <- match("Viburnum_prunifolium", 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 = "ri
add.scale.bar(cex=0.7)
```



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?

> It makes some sense, but there are different expectations for where they don't group properly. To resolve this I would probably use multiple genes to produce different trees. This helps ensure that conservation within a gene or selection due to locality or past events doesn't effect the tree structure and increase the number of base comparisons. For this project I think we are just going to use previously constructed trees instead of generating our own trees.