Phylogenetic Diversity - Traits

Alison Partee; Z620: Quantitative Biodiversity, Indiana University

22 February, 2017

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. Change "Student Name" on line 3 (above) with your name.
2. Complete as much of the exercise as possible during class; what you do not complete in class will need to be done on your own outside of class.
3. Use the handout as a guide; it contains a more complete description of data sets along with the proper scripting needed to carry out the exercise.
4. Be sure to **answer the questions** in this exercise document; they also correspond to the handout. Space for your answer is provided in this document and indicated by the ">" character. If you need a second paragraph be sure to start the first line with ">".
5. Before you leave the classroom, **push** this file to your GitHub repo.
6. For homework, 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, please submit the completed exercise by creating a **pull request** via GitHub. Your pull request should include this file *PhyloTraits_exercise.Rmd* and the PDF output of `Knitr` (*PhyloTraits_exercise.pdf*).

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

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

```
## [1] "/Users/flopsei/GitHub/QB2017_Partee/Week6-PhyloTraits"
```

```
setwd("/Users/flopsei/GitHub/QB2017_Partee/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: 'adephylo'

## The following object is masked from 'package:ade4':
##
##     orthogram

##
## Attaching package: 'permute'

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

## This is vegan 2.4-2

##
## Attaching package: 'vegan'

## The following object is masked from 'package:ade4':
##
##     cca

##
## 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 objects are masked from 'package:seqinr':
##
##     count, query

## 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
```

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

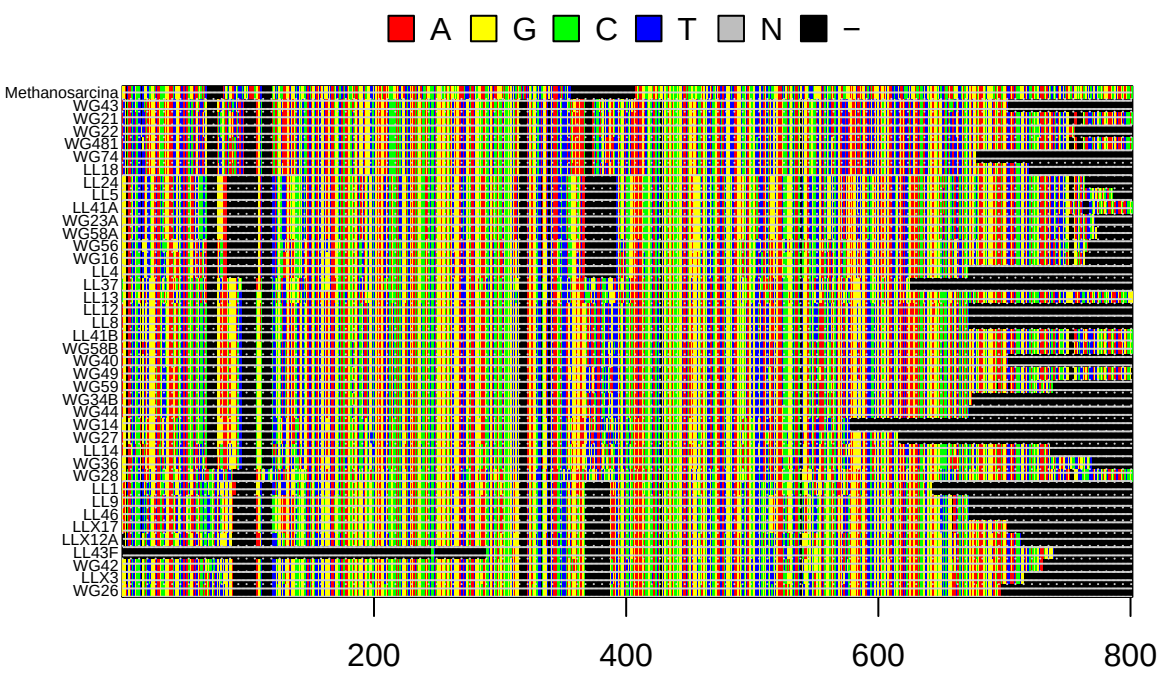
Question 1: Using less or your favorite text editor, compare the `p.isolates.fasta` file and the `p.isolates.afa` file. Describe the differences that you observe between the files.

> ***Answer 1***: p.isolates.fasta lists lower-case continuous nucleotide sequences with names above them, while p.isolates.afa lists upper-case nucleotide sequences with the same names as p.isolates.fasta but with slightly different sequences corresponding to each name. p.isolates.afa contains dashes ('-') but p.isolates.fasta does not.

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.

```
# visualizing the alignment

# read alignment file {sequinr}
read.aln <- read.alignment(file = './data/p.isolates.afa', format = 'fasta')

# convert alignment file to DNAbin Object {ape}
p.DNAbin <- as.DNAbin(read.aln)

# identify base pair region of 16S rRNA gene to visualize
window <- p.DNAbin[,100:900]
```

```
# command to visualize sequence alignment {ape}
image.DNAbin(window, cex.lab = 0.5)

#optional code adds grid to help visualize rows of sequences
grid(ncol(window), nrow(window), col = 'lightgrey')
```



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 reads?

b. What regions do you think would are appropriate for phylogenetic inference and why?

Answer 2a: Our reads are about 1500 units long. ***Answer 2b***: I think the regions between 100 and 700 would be the most appropriate for phylogenetic inference because it has fewer alignment gaps.

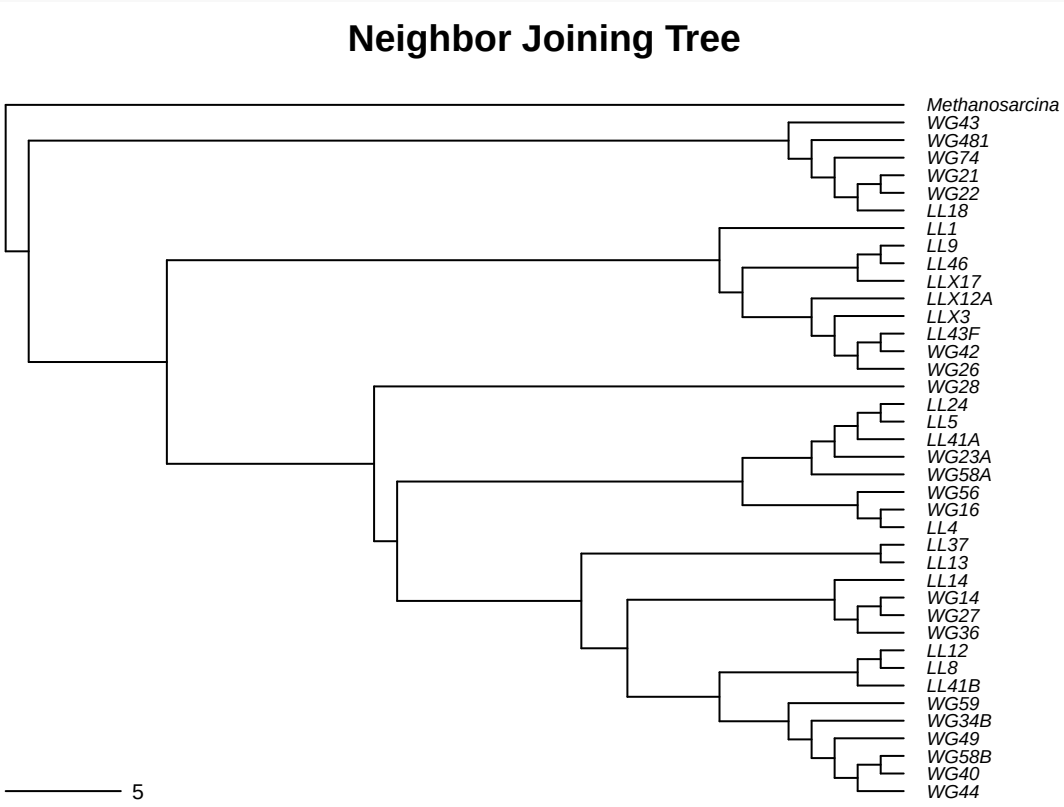
4) MAKING A PHYLOGENETIC TREE

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.

```r
# create distance matrix with 'raw' model {ape}
seq.dist.raw <- dist.dna(p.DNAbin, model = 'raw', pairwise.deletion = FALSE)

# neighbor joining algorithm to constuct tree, a 'phylo'
# object {ape}
nj.tree <- bionj(seq.dist.raw)

# identify outgroup sequence
outgroup <- match('Methanosarcina', nj.tree$tip.label)

# root the tree {ape}
nj.rooted <- root(nj.tree, outgroup, resolve.root = TRUE)

# plot the rooted tree {ape}
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)
```



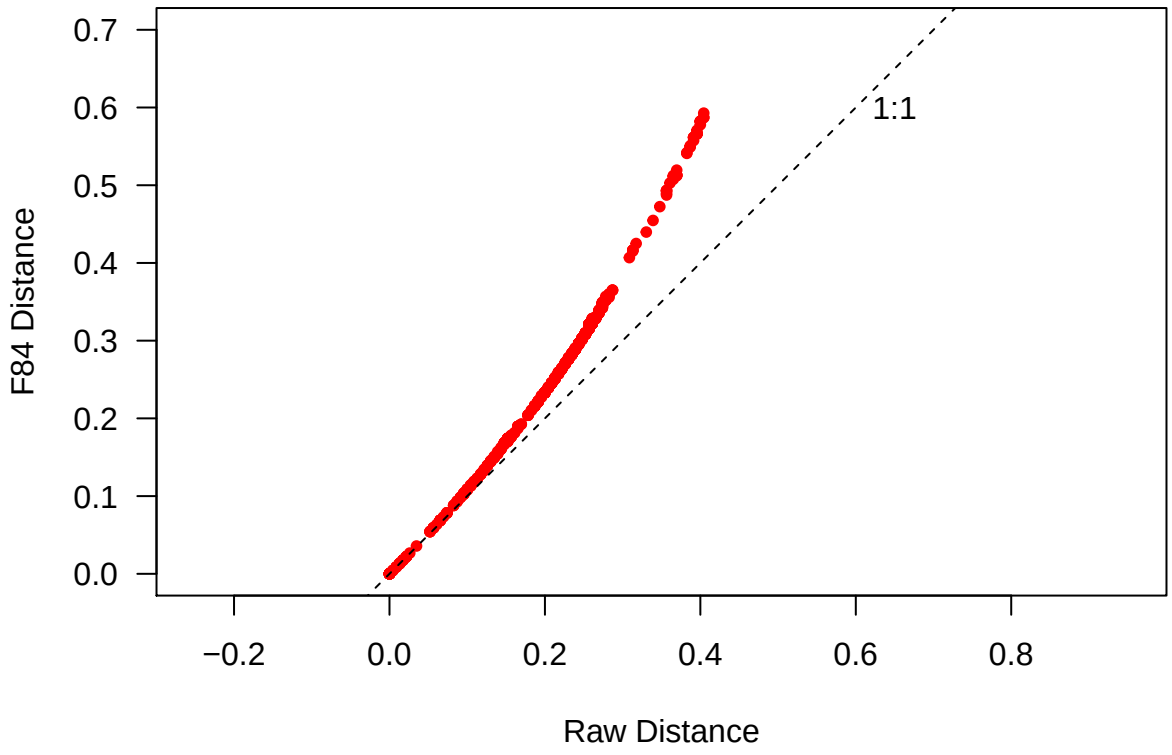
Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

> ***Answer 3***: Neighbor joining trees are often a good starting point for building more sophisticated models because they are simple, but they are not perfect for interpreting phylogenetic data. Neighbor joining trees cannot account for multiple substitutions that occur at a particular site, and they do not account for nucleotide substitution biases.

B) SUBSTITUTION MODELS OF DNA EVOLUTION

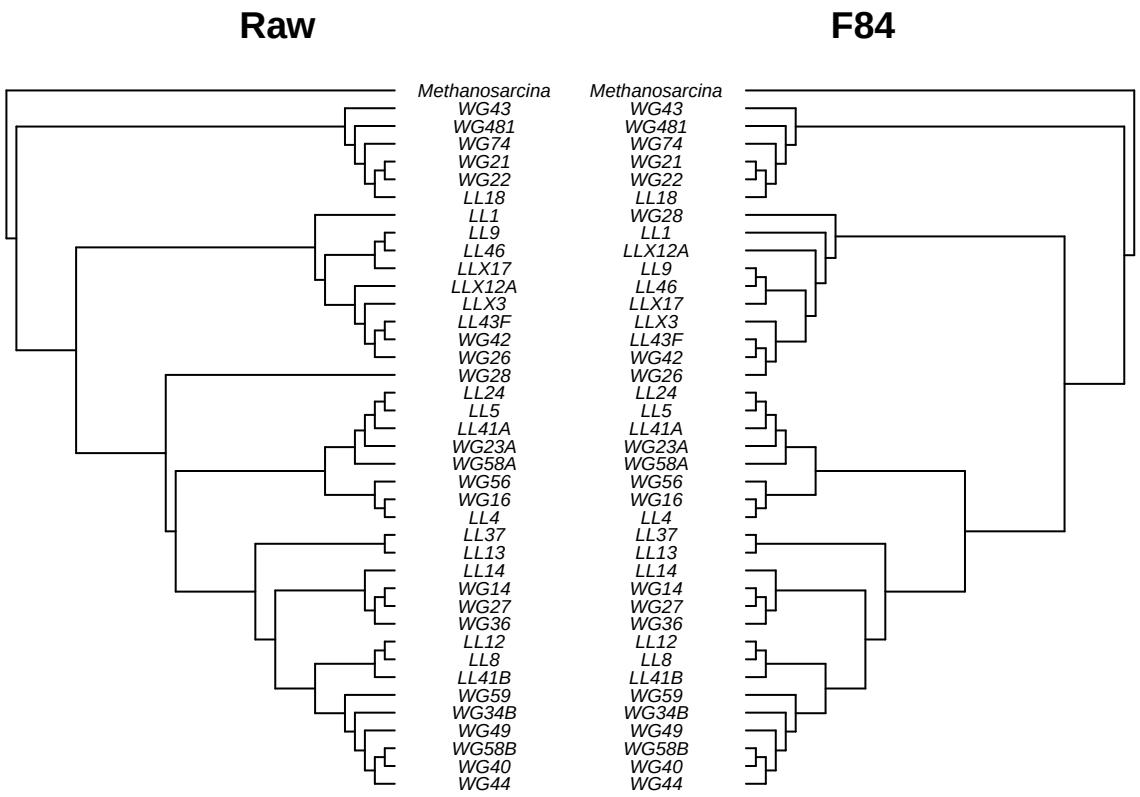
In the R code chunk below, do the following:
1. make a second distance matrix based on the Felsenstein 84 substitution model,
2. create a saturation plot to compare the *raw* and *Felsenstein (F84)* substitution models,
3. make Neighbor Joining trees for both, and
4. create a cophylogenetic plot to compare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

```
# create distance matrix with 'F84' model {ape}
seq.dist.F84 <- dist.dna(p.DNAbin, model = 'F84', pairwise.deletion = FALSE)

#plot distances from different dna substitution models
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', y
abline(b = 1, a = 0, lty = 2)
text(0.65, 0.6, '1:1')
```




```
#make neighbor joining trees using different dna substitution models {ape}
raw.tree <- bionj(seq.dist.raw)
F84.tree <- bionj(seq.dist.F84)

#define outgroups
raw.outgroup <- match('Methanosarcina', raw.tree$tip.label)
F84.outgroup <- match('Methanosarcina', F84.tree$tip.label)

#root the trees {ape}
raw.rooted <- root(raw.tree, raw.outgroup, resolve.root=TRUE)
F84.rooted <- root(F84.tree, F84.outgroup, resolve.root=TRUE)

#make the cophylogenetic plot {ape}
```

```
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 =

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



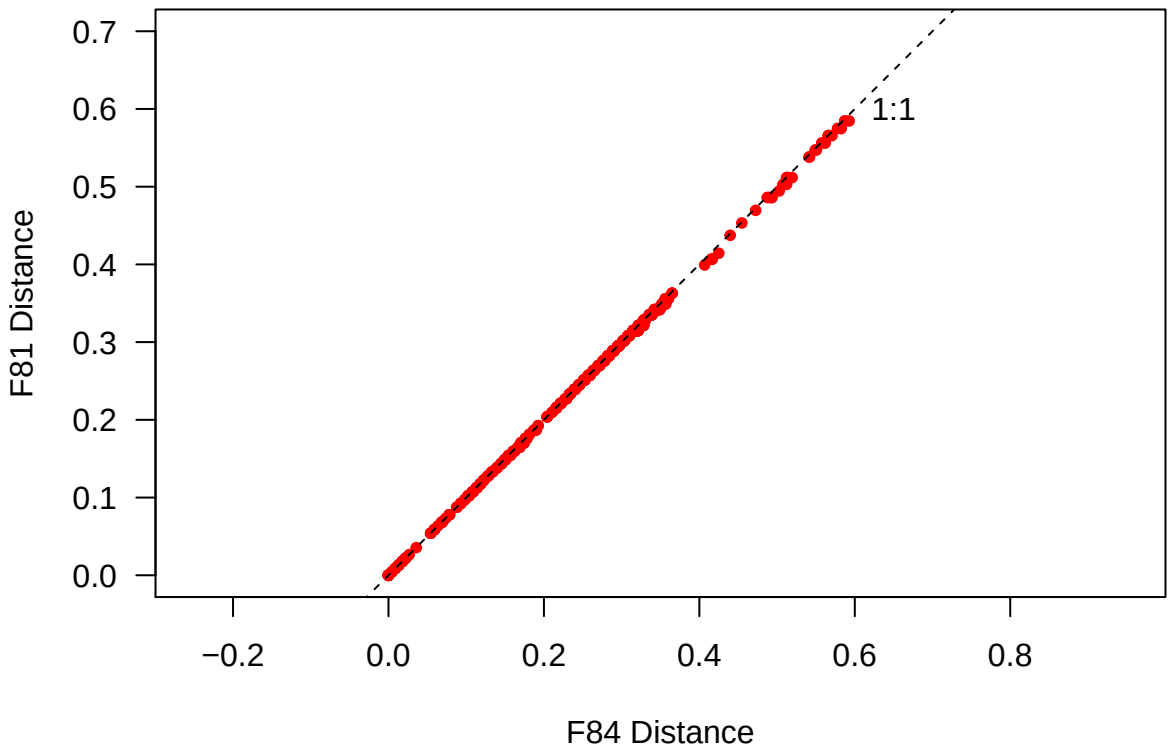

```
# Set method = 'PH85' for the symmetric difference Set method
# = 'score' for the symmetric difference This function
# automatically checks for a root and unroots rooted trees,
# so you can pass it either the rooted or unrooted tree and
# get the same answer.
dist.topo(raw.rooted, F84.rooted, method = 'score')
```

```
## [1] 0.04387426
```

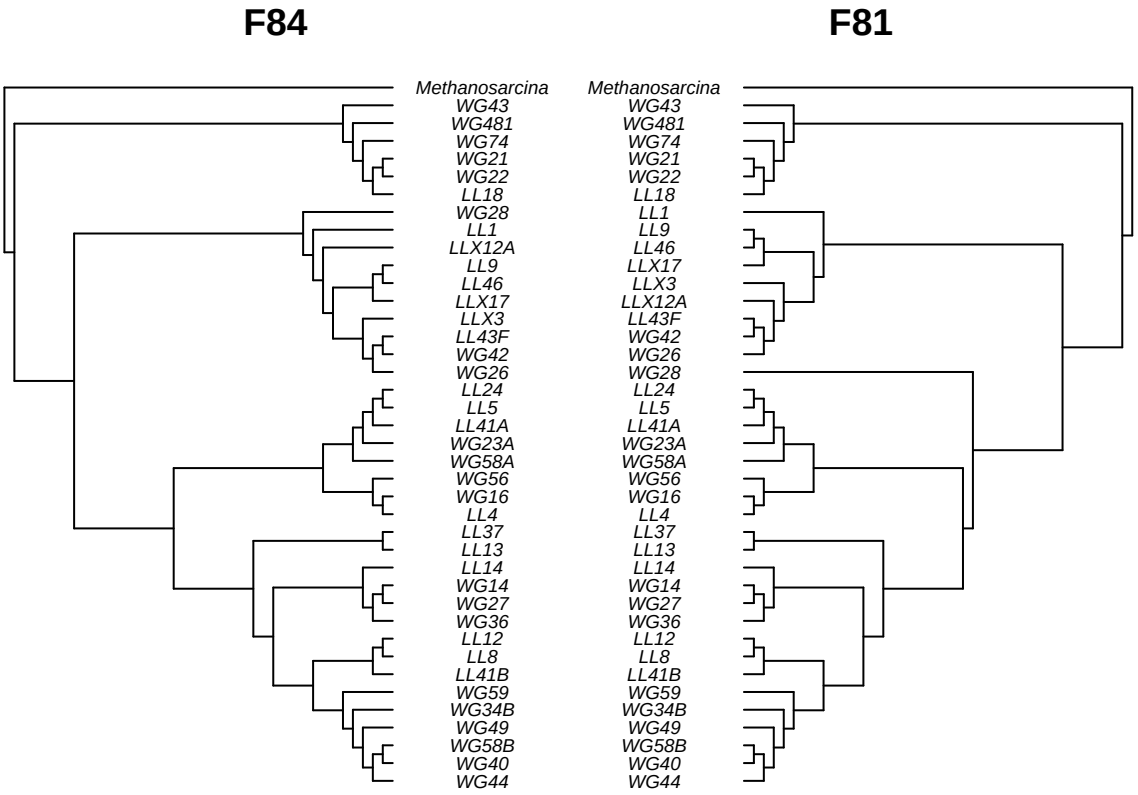
In the R code chunk below, do the following:
1. pick another substitution model,
2. create and distance matrix and tree for this model,
3. make a saturation plot that compares that model to the *Felsenstein (F84)* model,
4. make a cophylogenetic plot that compares the topologies of both models, and
5.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

```
# create distance matrix with 'JC69' model {ape}
seq.dist.F81 <- dist.dna(p.DNAbin, model = 'F81', pairwise.deletion = FALSE)

# plot distances from different dna substitution models
par(mar = c(5, 5, 2, 1) + 0.1)
plot(seq.dist.F84, seq.dist.F81,
     pch = 20, col = 'red', las = 1, asp = 1, xlim = c(0,0.7), ylim = c(0,0.7), xlab = 'F84 Distance', y
```

```
abline(b = 1, a = 0, lty = 2)
text(0.65, 0.6, '1:1')
```




```
# make neighbor joining trees using different dna substitution models {ape}
F81.tree <- bionj(seq.dist.F81)

# define outgroups
F81.outgroup <- match('Methanosarcina', F81.tree$tip.label)

# root the trees {ape}
F81.rooted <- root(F81.tree, F81.outgroup, resolve.root=TRUE)

# make the cophylogenetic plot {ape}
layout(matrix(c(1,2), 1, 2), width = c(1,1))
par(mar = c(1, 1, 2, 0))
plot.phylo(F84.rooted, type = 'phylogram', direction = 'right', show.tip.label = TRUE, use.edge.length =

par(mar = c(1,0,2,1))
plot.phylo(F81.rooted, type = 'phylogram', direction = 'left', show.tip.label = TRUE, use.edge.length =
```



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?

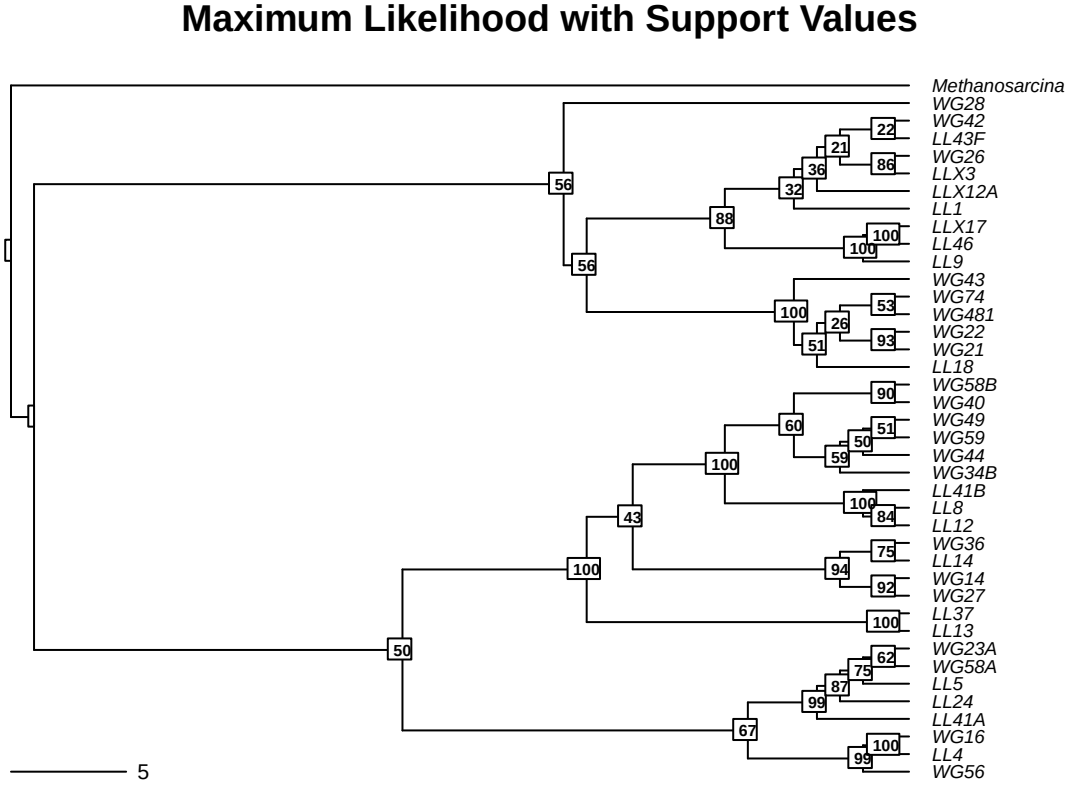
Answer 4a: The substitution model I chose was the F81 Felsenstein model, which differs from the F84 model in that it is a simpler version of the model that does not account for different rates of base transitions and transversions. ***Answer 4b***: The F81 substitution model I chose returns a different result than that of the F84 model, differing by only a couple changes in the placement of WG28, LL1, LLX12A, LL9, LL46, LLX17, and LLX3. These changes are likely because of the different assumptions of base transitions and transversions. ***Answer 4c***: The differences between the F81 and F84 models tell me that the corrections in more complex models likely do contribute to the accuracy of the tree. The groups that differ in placement between the F81 model and the F84 model may have a large amount of changes in bases that are more likely to mutate.

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

```
# # Requires alignment to be read in with as phyDat object
# p.DNAbin.phyDat <- read.phyDat('./data/p.isolates.afa', format = 'fasta', type = 'DNA')
# fit <- pml(nj.rooted, data=p.DNAbin.phyDat)
# # Fit tree using a JC69 substitution model
```

```
# fitJC <- optim.pml(fit, TRUE)
# # Fit tree using a GTR model with gamma distributed rates
# fitGTR <- optim.pml(fitGTR, model = 'GTR', optInv = TRUE, optGamma = TRUE,
# rearrangement = 'NNI', control = pml.control(trace = 0))
# # You can perform model selection with either an ANOVA test or by AIC value
# anova(fitJC, fitGTR)
# AIC(fitJC)
# AIC(gitGTR)
#
# # And you can perform a bootstrap test to see how well-supported the edges are
# bs = bootstrap.pml(fitJC, bs=100, optNni = TRUE, control = pml.control(trace = 0))

# bootstrap support

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
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: The MLE tree is different from the neighbor-joining tree in the handout because it uses a robust statistical technique to build this tree, whereas the neighbor-joining tree from before only used a distance matrix to construct the phylogeny. ***Answer 5b***: We bootstrap to generate confidence intervals for our tree. This is to see the likelyhood of accuracy in our tree. ***Answer 5c***: The bootstrap values tell us that we have some very strong, likely relationships within our tree. But we also have some very weak relationships with low bootstrap values, meaning that we have less confidence that these phylogenetic relationships are accurate. ***Answer 5d***: The branches that have the lowest support are the branches between WG42, LL43F, WG26, LLX3, LLX12A, and LL1. ***Answer 5e***: I don't think we should trust these branches because they are much lower in bootstrap value than the standard 95% threshold, having values as low as 21.

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.

```
#loading trait database

#import growth rate data
p.growth <- read.table('./data/p.isolates.raw.growth.txt', sep = '\t', header = TRUE, row.names = 1)

#standardize growth rates across strains
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.

```
#calculate maximum growth rate
umax <- (apply(p.growth, 1, max))

#quantify whether strains are generalists or specialists
levins <- function(p_xi = '') {
  p = 0
  for (i in p_xi) {
    p = p + i^2
  }
  nb = 1 / (length(p_xi) * p)
  return(nb)
}

#calculate niche breadth for each isolate
nb <- as.matrix(levins(p.growth.std))

# add row and column names to niche breadth matrix
```

```
rownames(nb) <- row.names(p.growth)
colnames(nb) <- c('NB')
```

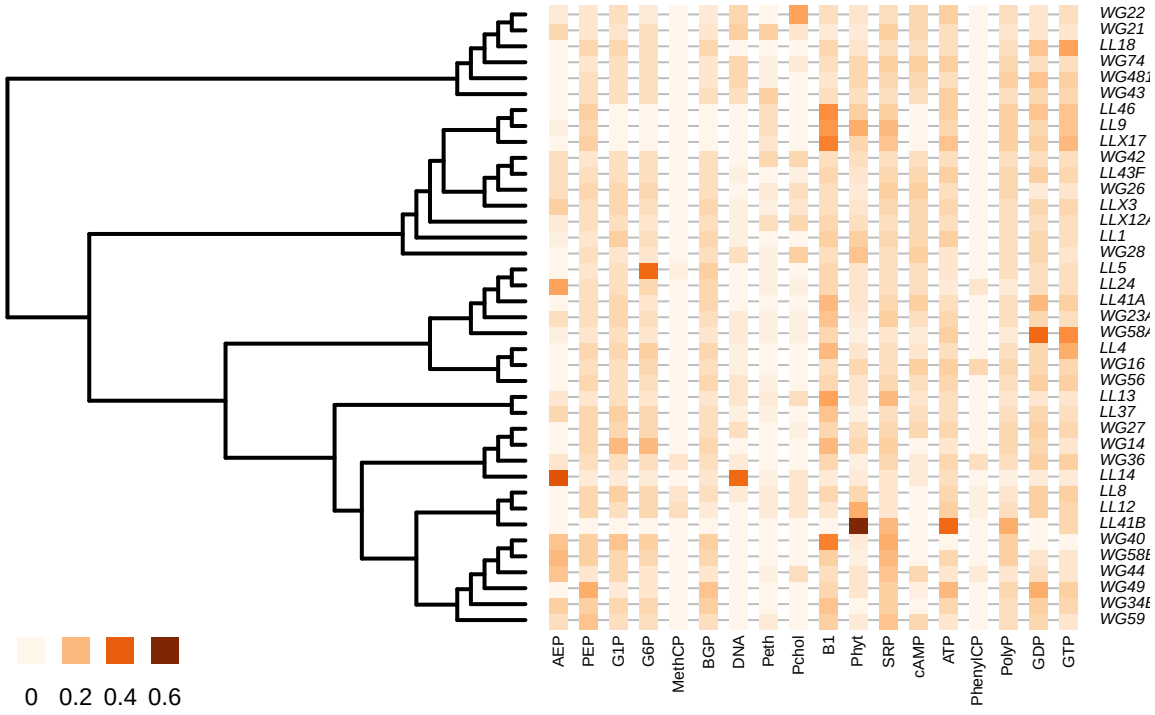
C. Visualizing Traits on Trees

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.

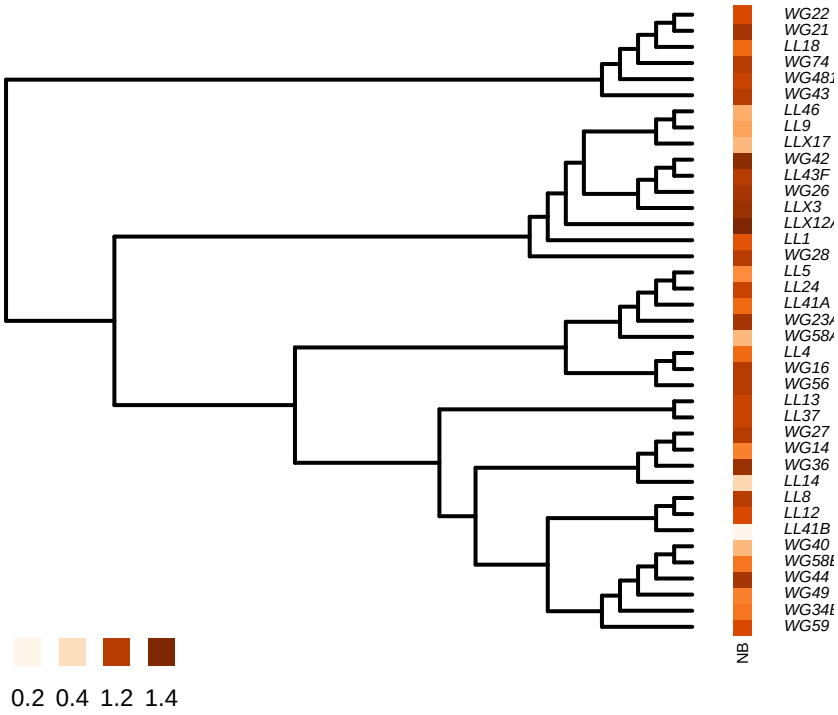
```
# generate neighbor joining tree using F84 DNA substitution model {ape}
nj.tree <- bionj(seq.dist.F84)

# define the outgroup
outgroup <- match('Methanosarcina', nj.tree$tip.label)

# create a rooted tree {ape}
nj.rooted <- root(nj.tree, outgroup, resolve.root = TRUE)

# keep rooted but drop outgroup branch
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).

```
#define color palette
mypalette <- colorRampPalette(brewer.pal(9, 'Oranges'))

# Map phosphorus traits {adephylo}
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.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)
```




```
# Niche breadth
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.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: Organisms that come from oligotrophic lakes will likely have greater niche breadth because they may need to use whatever resources they can get in their habitat, since oligotrophic lakes have low phosphorus and other nutrient contents. Organisms from eutrophic lakes will likely be specialist since they typically have an abundance of nutrients available to them. ***Answer 6b***: This answer depends on how organisms got into the different lakes in the first place. If the lakes started with similar populations and were allowed to diverge phylogenetically, then I would expect to see pairs of species where one has high niche breadth values and low growth rate and the other has low niche breadth values and higher growth rates because the organisms would be genetically closely related but phenotypically different due to divergence from environmental differences. If the lake populations started as different groups more closely related to each other than to organisms in the other lake, then I would expect to see two large groups in our tree where one group has individuals that have high niche breadth values and low growth rate and the other group has low niche breadth values and high growth rate.

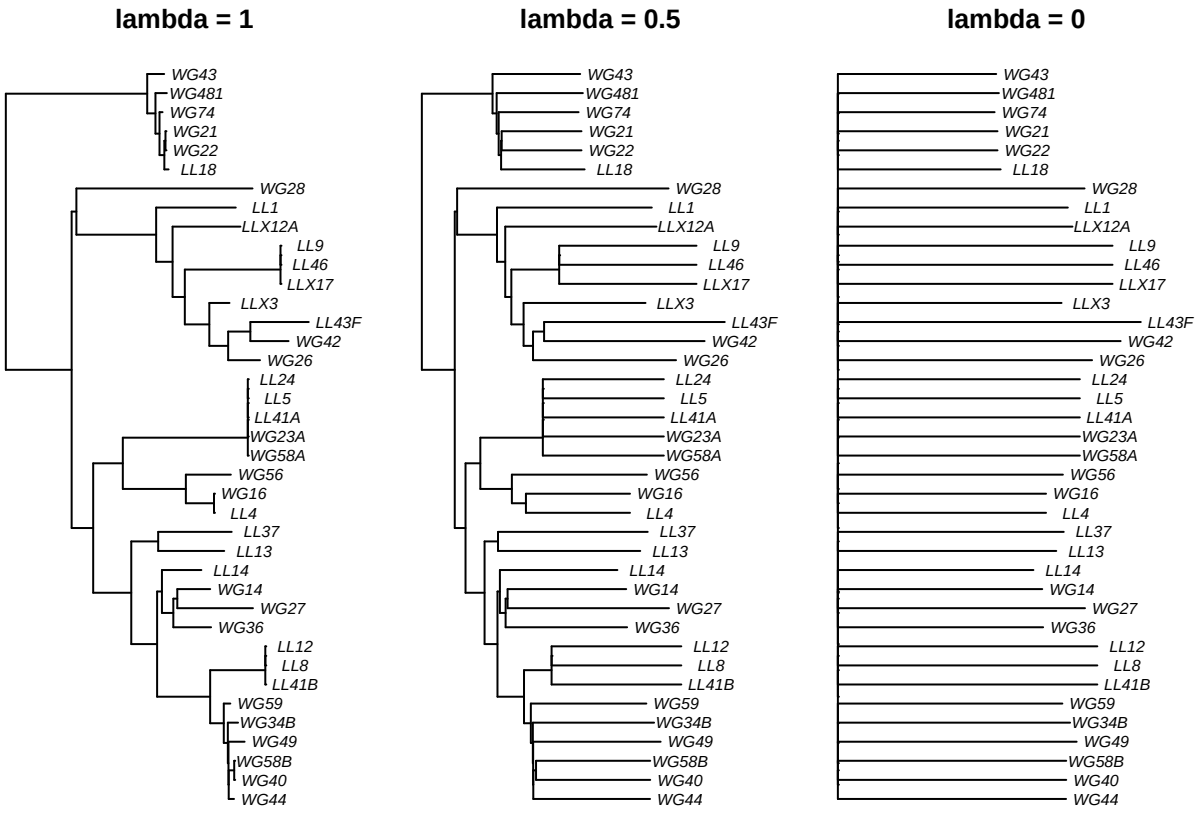
6) HYPOTHESIS TESTING

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.

```
# Visualize Trees with different levels of phylogenetic signal {geiger}
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(1,0.5,2,0.5) + 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.

```
# Generate test statistics for comparing phylogenetic signal {geiger}
fitContinuous(nj.rooted, nb, model = 'lambda')
```

```
## GEIGER-fitted comparative model of continuous data
##  fitted 'lambda' model parameters:
##  lambda = 0.020848
##  sigsq = 0.106492
##  z0 = 0.661368
##
##  model summary:
##  log-likelihood = 21.661104
##  AIC = -37.322208
##  AICc = -36.636494
##  free parameters = 3
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 43
```

```
##  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 = 0.106395
##  z0 = 0.657777
##
##  model summary:
##  log-likelihood = 21.652293
##  AIC = -37.304587
##  AICc = -36.618872
##  free parameters = 3
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 0
##  frequency of best fit = 0.91
##
##  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). 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***: The lambda values of the untransformed tree is 1 indicating the model accounts for all phylogenetic signal and the transformed tree's lambda value is 0 indicating no phylogenetic signal is used to create the model. ***Answer 7b***: The AIC value when lambda = 1 is -37.322 and the AIC value when lambda = 0 is -37.305. Since the values are so close to each other and their difference is less than 2, I cannot determine which of these models is better. ***Answer 7c***: Since the AIC values indicated that the models were equivalent, this suggests that there is little 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.

```
# first, correct for zero branch-lengths on out tree
nj.rooted$edge.length <- nj.rooted$edge.length + 10^-7

# Calculate plylogenetic signal for growth on all phosphorus resources
# first, create a blank output matrix
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')

# use a for loop to calculate Blomberg's K for each resource
for (i in 1:18) {
  x <- as.matrix(p.growth.std[ ,i, drop = FALSE])
  out <- phylosignal(x,nj.rooted)
  p.phylosignal[1:5, i] <- round(t(out), 3)
}

# use the BH correction on p-values:
p.phylosignal[6, ] <- round(p.adjust(p.phylosignal[4,], method = 'BH'), 3)

#calculate phylogenetic signal for Niche breadth
signal.nb <- phylosignal(nb, nj.rooted)
signal.nb
```

```
##              K PIC.variance.obs PIC.variance.rnd.mean PIC.variance.P
## 1 3.427719e-06         49966.78              49463.41           0.56
##   PIC.variance.Z
## 1     0.02440362
```

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: The K values were all near 0, indicating lack of phylogenetic signal for the phosphorus resources. ***Answer 8b***: Since the K values were near 0, this indicated overdispersion.

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.

```
#turn continuous data into categorical data
p.growth.pa <- as.data.frame((p.growth > 0.01) * 1)

# look at phosphorus use for each resource
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
```

```
# add names column to data
p.growth.pa$name <- row.names(p.growth.pa)

#merge trait and phylogenetic data; run 'phylo.d'
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 :  0.4592424
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.004
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.017
```

```
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 :  0.6074918
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.028
## 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 :  0.1582507
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.002
```

```
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.288
```

```
# # compare evolutionary models
#
# # brownian motion = BM
# # brownian motion + directional selection = trend
# # ornstein-uhlenbeck = OU
# # pagel's lambda = lambda
# is.ultrametric(nj.rooted)
# nj.rooted.um <- chronos(nj.rooted)
#
# out <- pmc(nj.rooted.um, nb, 'BM', 'OU', nboot = 100)
#
# # After running the above (took a long time) I got the following error:
# # Warning message:
# # In fitContinuous (phy = tree, dat = data, model = model, ..., ncores = 1) :
# #     Parameter estimates appear at bounds:
# #     alpha
#
# dists <- data.frame(null = out$null, test = out$test)
# colnames(dists) <- c('BM', 'OU')
#
# # dev.off()
#
# png('./figs/H_test.png', width = 480, height = 240, res = 120)
#
# dists %>%
#   gather(model, value) %>%
#   ggplot(aes(value, fill = model)) +
#   geom_density(alpha = 0.5) +
#   geom_vline(xintercept = out$lr) +
#   xlab(expression(delta)) +
#   theme(panel.background = element_blank())
# dev.off()
#
# # determine approximate p value
# length(dists$BM[dists$BM > out$lr]) length(dists$BM)
```

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?

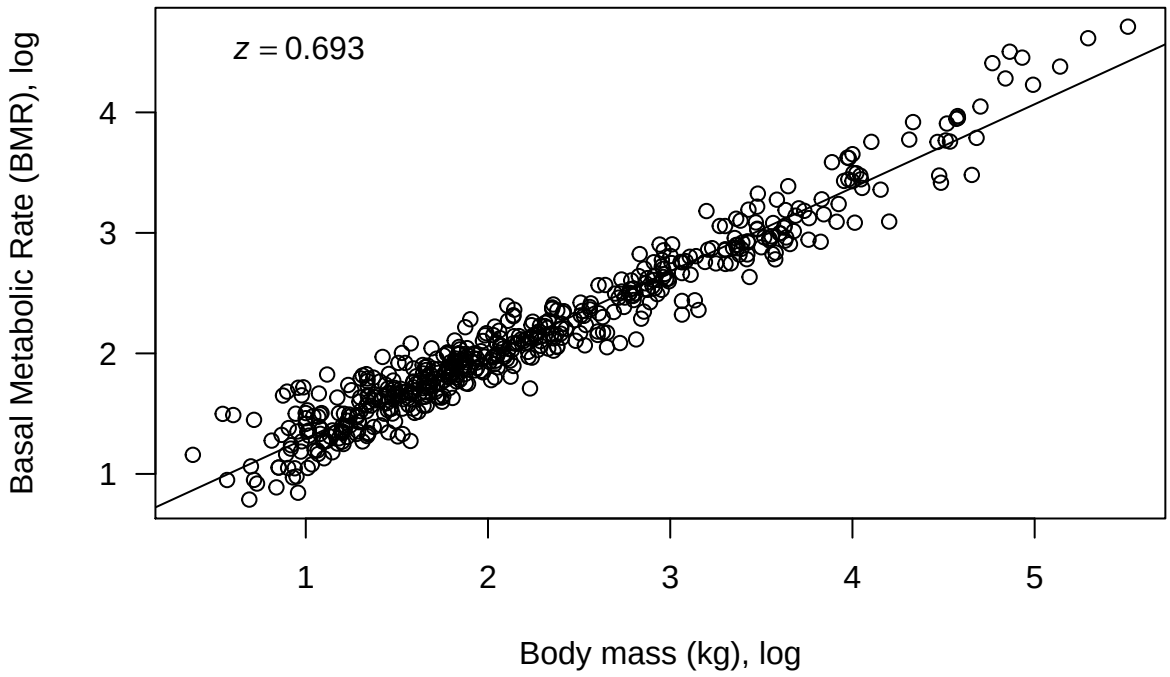
c. Discuss what factors might give rise to differences between the metrics.

Answer 9a: Okay ***Answer 9b***: The results show that the traits for AEP, DNA, and cAMP likely stem from some kind of phylogentic structure, because their probabilities of stemming from no phylogenetic structures are close to zero, differing from the Blomberg's K analysis. However, all three of their D values are positive, especially DNA and AEP, meaning the traits are overdispersed, which is more consistent with the Blomberg's K analysis. ***Answer 9c***: Grouping the growth data categorically likely changed the analysis. There may be information gained or lost when choosing to calculate dispersion categorically vs calculating it using Blomberg's K.

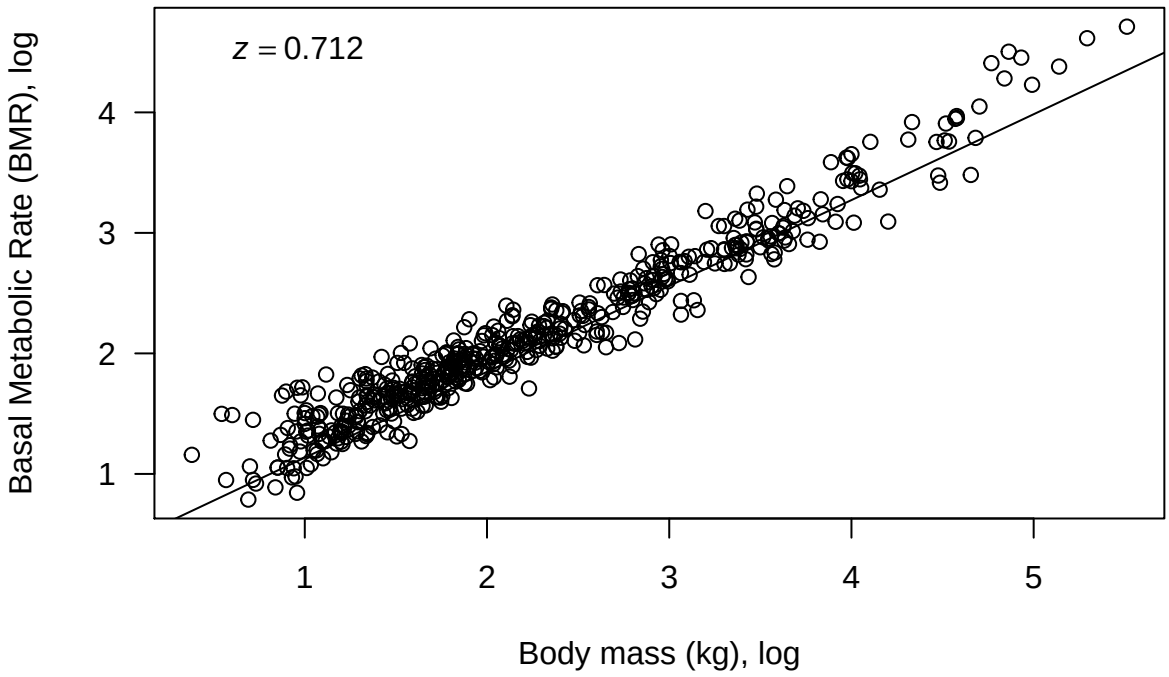
7) PHYLOGENETIC REGRESSION

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

```
#input the tree and dataset
mammal.Tree <- read.tree('./data/mammal_best_super_tree_fritz2009.tre')
mammal.data <- read.table('./data/mammal_BMR.txt', sep = '\t', header = TRUE)

# select the variables we want to analyze
mammal.data <- mammal.data[ , c('Species', 'BMR_.mlO2.hour.', 'Body_mass_for_BMR_.gr.')]
mammal.species <- array(mammal.data$Species)

# select the tips in the mammal tree that are also in the dataset
pruned.mammal.tree <- drop.tip(mammal.Tree, mammal.Tree$tip.label[-na.omit(match(mammal.species, mammal

# select the species from the dataset that are in our pruned tree
pruned.mammal.data <- mammal.data[mammal.data$Species %in% pruned.mammal.tree$tip.label,]

# turn column of species names into rownames
rownames(pruned.mammal.data) <- pruned.mammal.data$Species

# look at the relationship between mass and BMR

# run a simple linear regression
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, xlab = 'Body mass (kg), log',
     ylab = 'Basal Metabolic Rate (BMR), log')
abline(a = fit$coefficients[1], b = fit$coefficients[2])
b1 <- round(fit$coefficients[2], 3)
eqn <- bquote(italic(z) == .(b1))

# plot the slope
text(0.5, 4.5, eqn, pos = 4)
```




```
# correct for phylogeny using 'phylolm()' function from the package 'phylolm'

# run a phylogeny-corrected regression with no bootstrap replicates
fit.phy <- phylolm(log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.),
                   data = pruned.mammal.data, pruned.mammal.tree, model = 'lambda',
                   boot = 0)
plot(log10(pruned.mammal.data$Body_mass_for_BMR_.gr.),
     log10(pruned.mammal.data$BMR_.mlO2.hour.),
     las = 1, xlab = 'Body mass (kg), log', ylab = 'Basal Metabolic Rate (BMR), log')
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(0.5, 4.5, eqn, pos = 4)
```



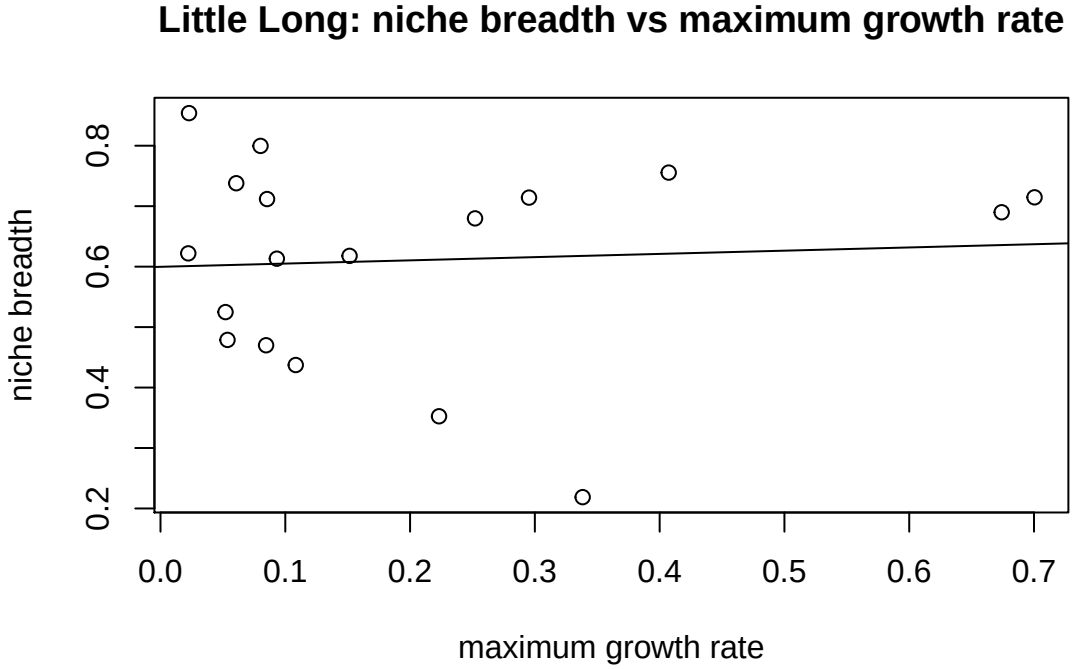
a. Why do we need to correct for shared evolutionary history?
b. How does a phylogenetic regression differ from a standard linear regression?
c. Interpret the slope and fit of each model. Did accounting for shared evolutionary history improve or worsten the fit?
d. Try to come up with a scenario where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

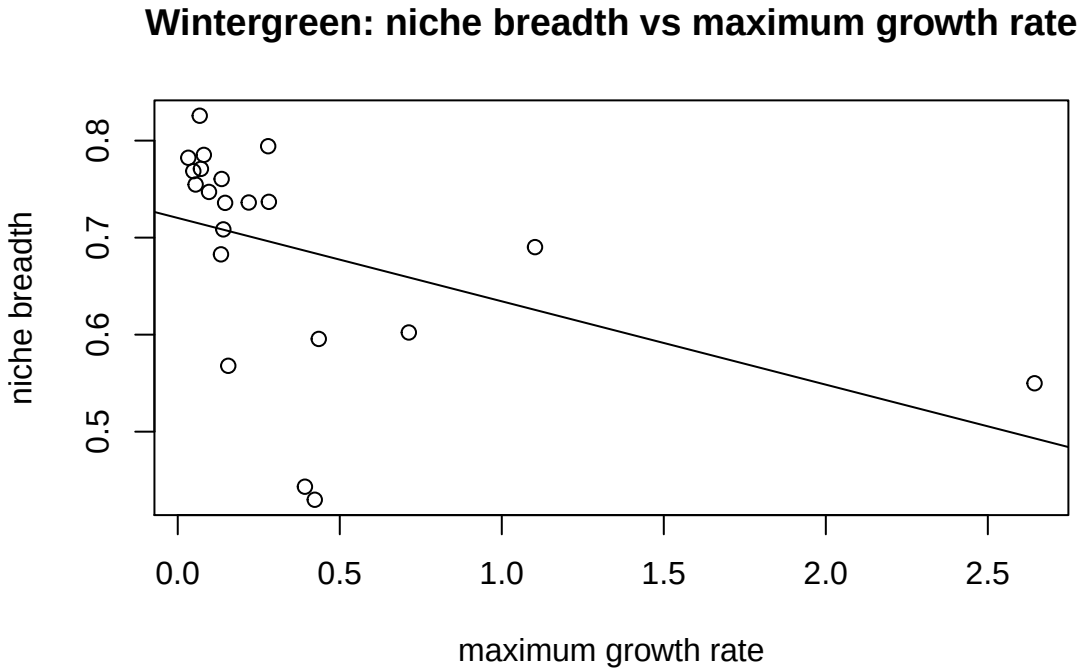
Answer 10a: We need to correct for shared evolutionary history because shared evolutionary history implies dependence and simple linear regression assumes independence. Hence, this needs to be corrected. ***Answer 10b***: A standard linear regression assumes errors are normally distributed random variables, while a phylogenetic regression the errors' variance is found by using a covariance matrix that accounts for underlying phylogeny. ***Answer 10c***: Accounting for shared evolutionary history does not seem to make much of a difference in this case, because the slopes of the uncorrected and corrected fits are very similar to each other. ***Answer 10d***: The relationship between two variables may completely disappear when accounting for underlying phylogeny if we were analyzing two groups of organisms, each with its own lengthy evolutionary history where the two groups diverged a long time ago. In this situation, we measure BMR and Body mass and look for a relationship. One of the groups of animals has a high BMR with low body mass and the other has a low BMR with relatively high body mass. This creates a negative looking relationship until phylogeny is accounted for, and after phylogeny is accounted for we see an absence of any relationship based on our data.

7) SYNTHESIS

Below is the output of a multiple regression model depicting the relationship between the maximum growth rate (μ_{max}) of each bacterial isolate and the niche breadth of that isolate on the 18 different sources of phosphorus. One feature of the study which we did not take into account in the handout is that the isolates came from two different lakes. One of the lakes is an very oligotrophic (i.e., low phosphorus) ecosystem named Little Long (LL) Lake. The other lake is an extremely eutrophic (i.e., high phosphorus) ecosystem named Wintergreen (WG) Lake. We included a "dummy variable" (D) in the multiple regression model (0 = WG, 1 = LL) to account for the environment from which the bacteria were obtained. For the last part of the assignment, plot niche breadth vs. μ_{max} and the slope of the regression for each lake. Be sure to color the

data from each lake differently.





Question 11: Based on your knowledge of the traits and their phylogenetic distributions, what conclusions would you draw about our data and the evidence for a generalist-specialist tradeoff?

> ***Answer 11***: Little Long is the oligotrophic lake and it seems to have little correlation between maximum growth rate and niche breadth and Wintergreen lake seems to have a negative correlation between maximum growth rate and niche breadth. The oligotrophic lake may have little correlation between the two variables because its organisms should tend to have a higher niche breadth while trying to maximize their growth rate. This could cause organisms with high niche breadth and maximum growth rate due to competition between the organisms. These organisms that have both large niche breadth and high max growth may have acquired these favorable traits by

trading off other traits that we do not have data for. The eutrophic lake, Wintergreen, has more of a relationship that we would expect. It has a negative correlation, which may mean that its organisms have less pressure to have both high niche breadth and high max growth rate, likely due to the abundance of nutrients available in the lake.