Phylogenetic Diversity - Traits

Katie Beidler; Z620: Quantitative Biodiversity, Indiana University

21 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).

```
clr = function() {
  ENV = globalenv()
  ll = ls(envir = ENV)
  ll = ll[ll != "clr"]
  rm(list = ll, envir = ENV)
}
getwd()
setwd("/Users/bhbeidler/GitHub/QB2017_Beidler/Week6-PhyloTraits")

package.list = c('ape', 'seqinr', 'phylobase', 'adephylo', 'geiger',
'picante', 'stats', 'RColorBrewer', 'caper', 'phylolm',
'pmc', 'ggplot2', 'tidyr', 'dplyr', 'phangorn', 'pander')
for (package in package.list) {
if (!require(package, character.only = TRUE, quietly = TRUE)) {
install.packages(package, repos='http://cran.us.r-project.org')
library(package, character.only = TRUE) }
}
```

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.

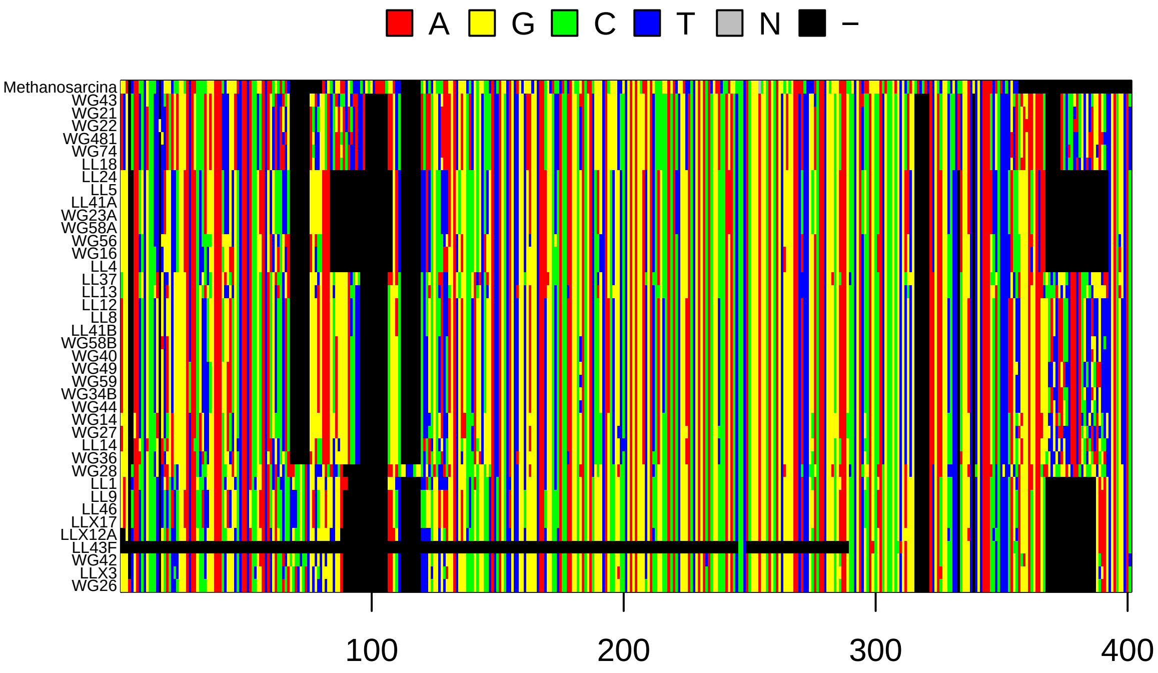
3) SEQUENCE ALIGNMENT

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.

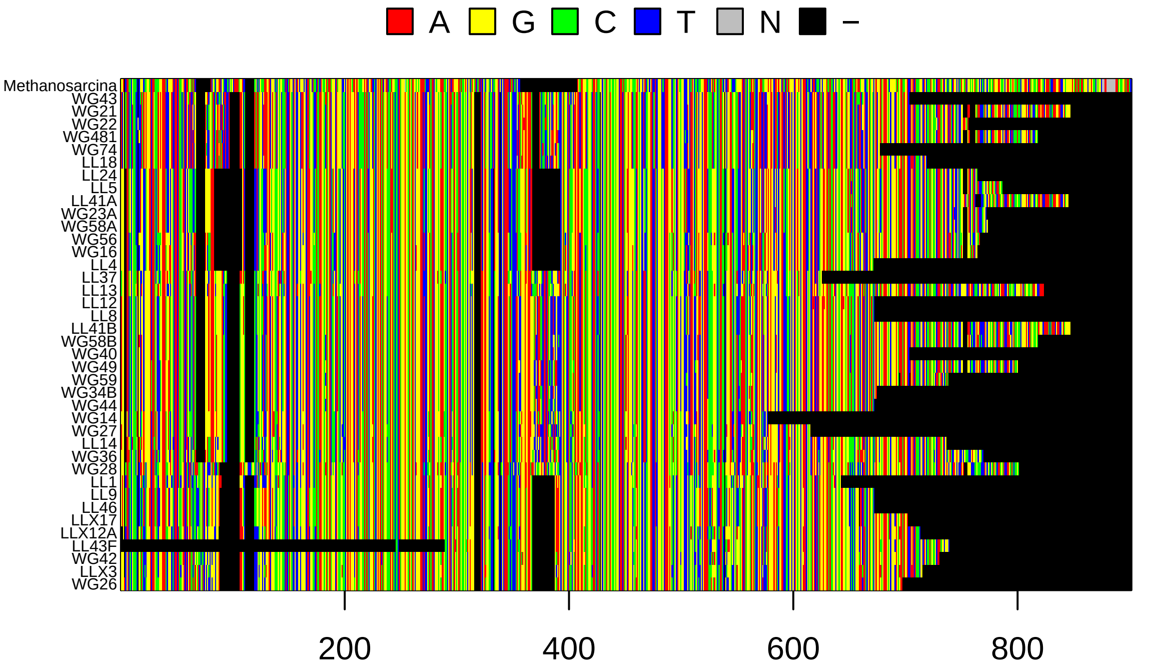
> ***Answer 1***: The `p.isolates.afa` file is the FASTA aligned output file- in which the sequences have been alligned. In the `p.isolates.fasta` file the isolates have not been alligned.

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.

```
# 1. Read in the alignment file
read.aln = read.alignment(file = "./data/p.isolates.afa", format = "fasta")

# 2. Convert the alignment to a DNAbin object
p.DNAbin = as.DNAbin(read.aln)

# 3. Identify a base pair region of 16S rRNA gene to visualize
window = p.DNAbin[, 100:500]
window.full = p.DNAbin[, 100:1000]
# 4. Plot the alignment- Command to Visualize sequence alignment {ape}
image.DNAbin(window, cex.lab = 0.50)
```



```
image.DNAbin(window.full, 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 reads?

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

Answer 2a: Approimately 700 reads. ***Answer 2b***: The regions that are variable would be the most appropriate for phylogenetic inference, because those regions with sequence variation will help us determine which species differ with respect to phosphorus metabolism.

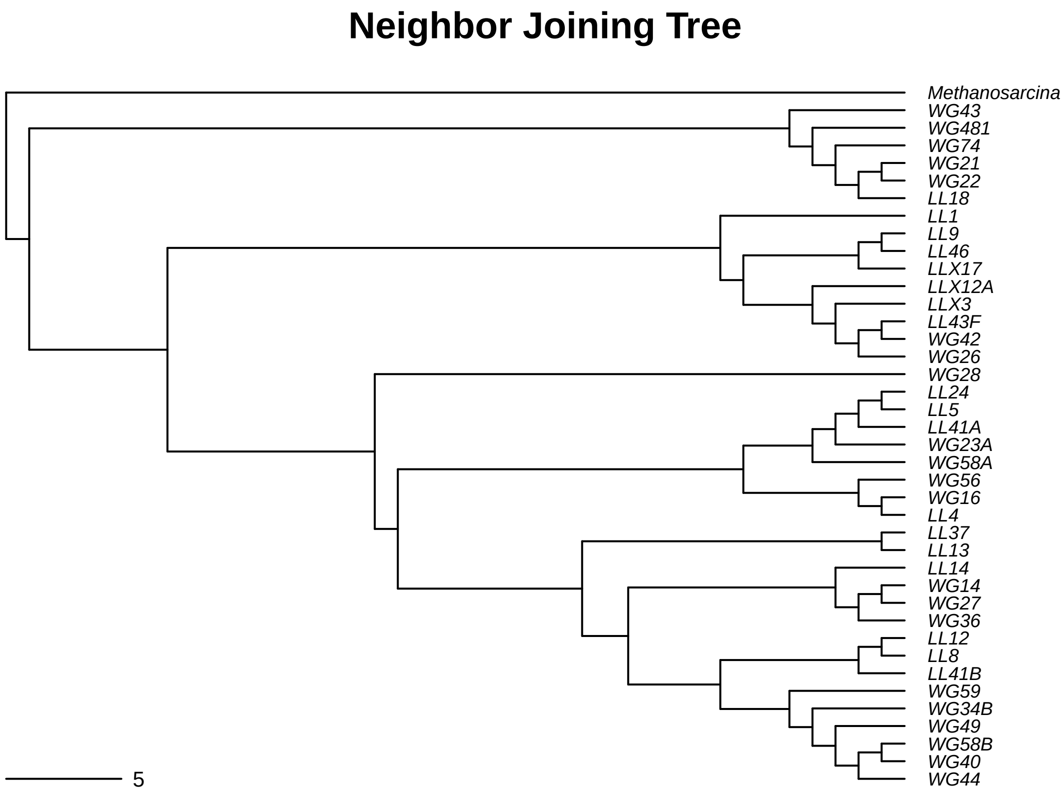
4) MAKING A PHYLOGENETIC TREE

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
# 1. calculate the distance matrix using `model = "raw"`
seq.dist.raw = dist.dna(p.DNAbin, model = "raw", pairwise.deletion = FALSE)

# 2. create a Neighbor Joining tree based on these distances
# Neighbor Joining Algorithm to Construct Tree, a 'phylo'
nj.tree = bionj(seq.dist.raw)
# ID 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)
```



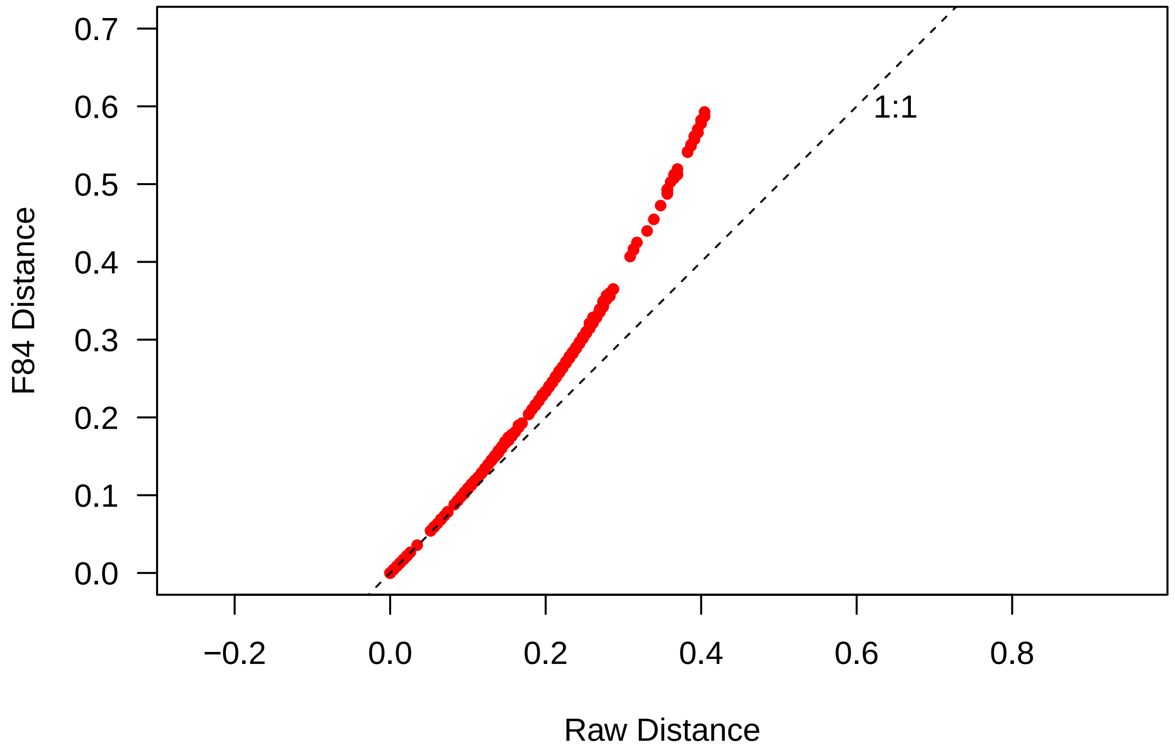
Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

> ***Answer 3***: Neighbor joined trees are advantageous in that they are fast and can serve as "guide trees" or a starting points for assessing evolutionary models with more sophisticated methods such as maximum likelihood. However, the neighbor joining method on its own - ignores models of evolution and thus does not accurately represent the real phylogeny. Additionally, neighbor joined trees only use a distance matrix, which doesn't take into account specific nucleotide states and it results in only one tree. As a result neighbor joinin is not a statistical method and is dependent on the underlying substitution model.

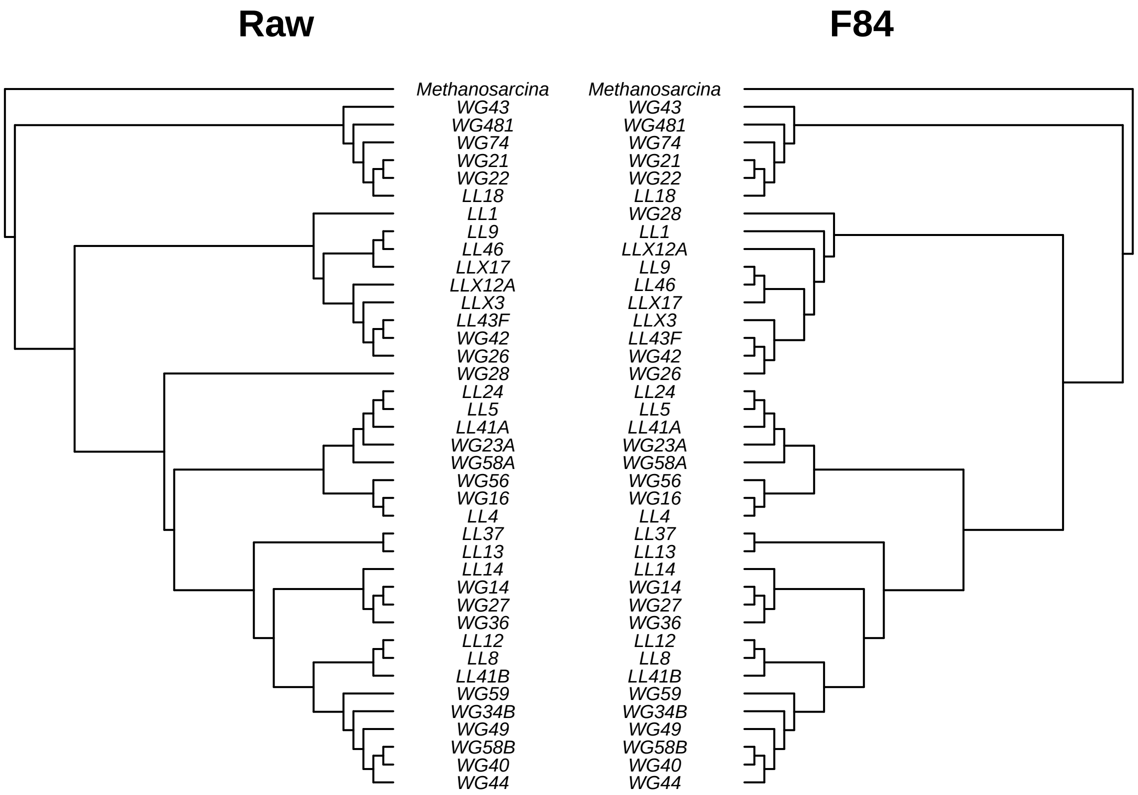
B) SUBSTITUTION MODELS OF DNA EVOLUTION

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.

```r
# 1. Create distance matrix with "F84" model {ape}
seq.dist.F84 = dist.dna(p.DNAbin, model = "F84", pairwise.deletion = FALSE)

# 2. create a cophylogenetic plot to compare the topologies of the trees
# 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,
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 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 = 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")
```

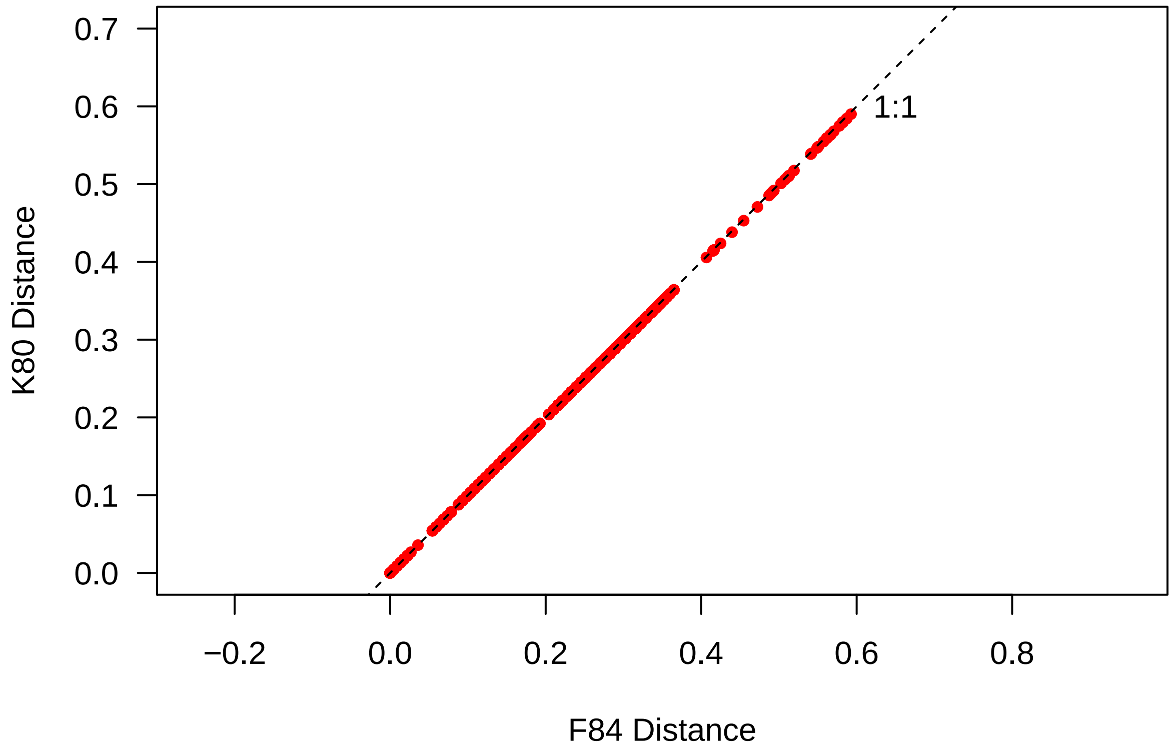


```
# Set method = 'PH85' for the symmetric difference Set method
# = 'score' for the symmetric difference This function
# automatically checks for a root and unroots rooted trees,
# so you can pass it either the rooted or unrooted tree and
# get the same answer.
dist.topo(raw.rooted, F84.rooted, method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

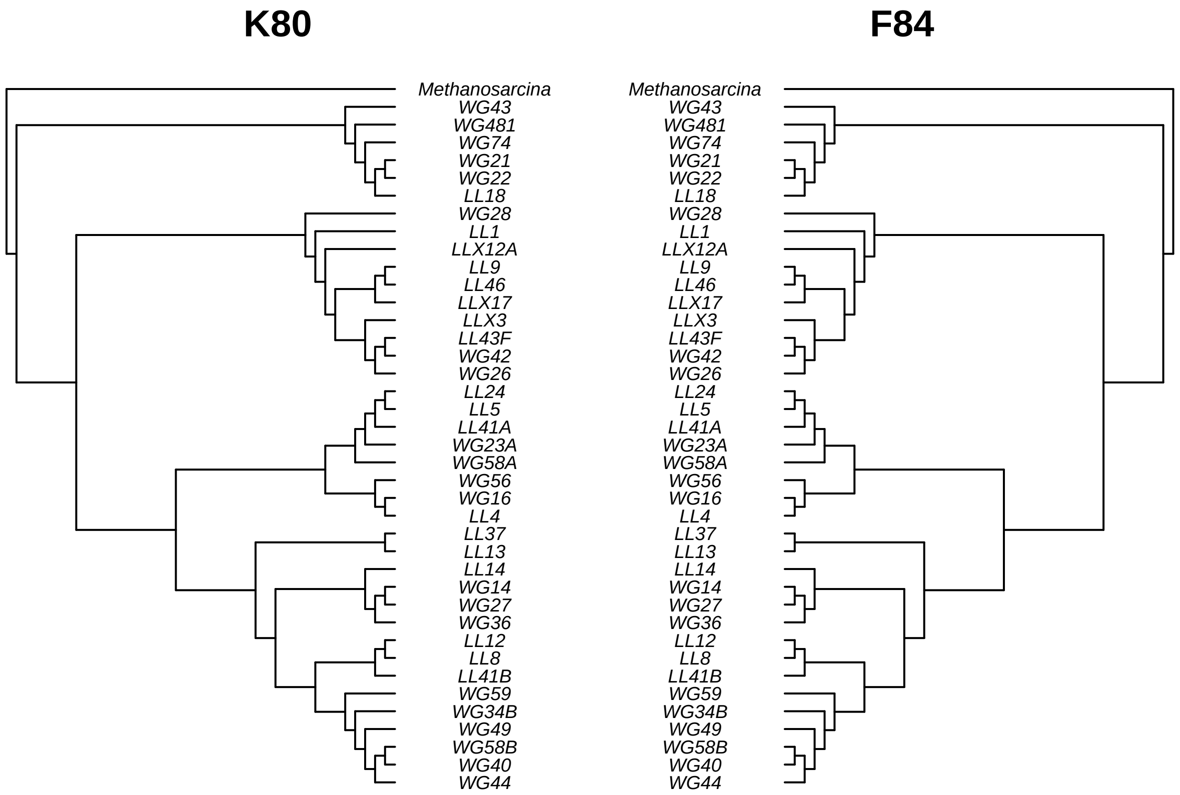
```
## [1] 0.04387426
```

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.

```
# 1. Create distance matrix with Kimura model K80 {ape}
seq.dist.K80 = dist.dna(p.DNAbin, model = "K80", pairwise.deletion = FALSE)

# 2. create a cophylogenetic plot to compare the topologies of the trees
# Plot Distances from Different DNA Substitution Models
par(mar = c(5, 5, 2, 1) + 0.1)
plot(seq.dist.F84, seq.dist.K80, pch = 20, col = "red", las = 1, asp = 1, xlim = c(0, 0.7), ylim = c(0,
abline(b = 1, a = 0, lty = 2)
text(0.65, 0.6, "1:1")
```



```
# Make Neighbor Joining Trees Using Different DNA Substitution Models {ape}
F84.tree = bionj(seq.dist.F84)
K80.tree = bionj(seq.dist.K80)
# Define Outgroups
F84.outgroup = match("Methanosarcina", F84.tree$tip.label)
K80.outgroup = match("Methanosarcina", K80.tree$tip.label)
# Root the Trees {ape}
F84.rooted = root(F84.tree, F84.outgroup, resolve.root=TRUE)
K80.rooted = root(K80.tree, K80.outgroup, resolve.root=TRUE)
# Make Cophylogenetic Plot {ape}
layout(matrix(c(1,2), 1, 2), width = c(1, 1))
par(mar = c(1, 1, 2, 0))
par(mar = c(1, 0, 2, 1))
plot.phylo(K80.rooted, type = "phylogram", direction = "right", show.tip.label=TRUE,
           use.edge.length = FALSE, adj = 0.5, cex = 0.6, label.offset = 2, main = "K80")

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")
```



```r
dist.topo(F84.rooted,K80.rooted,  method = "score")
```

```
## [1] 0.0006260388
```

```r
dev.off()
```

```
## null device
##           1
```

Question 4:

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

   Answer 4a: I chose the Kimura Model (K80) which assumes equal frequencies of nucleotides but recognizes that transition mutations occur with greater frequency than transversion mutations. Whereas, Felenstein Model (F84) assumes differences in base frequencies and different rates of base transitions and transversions.

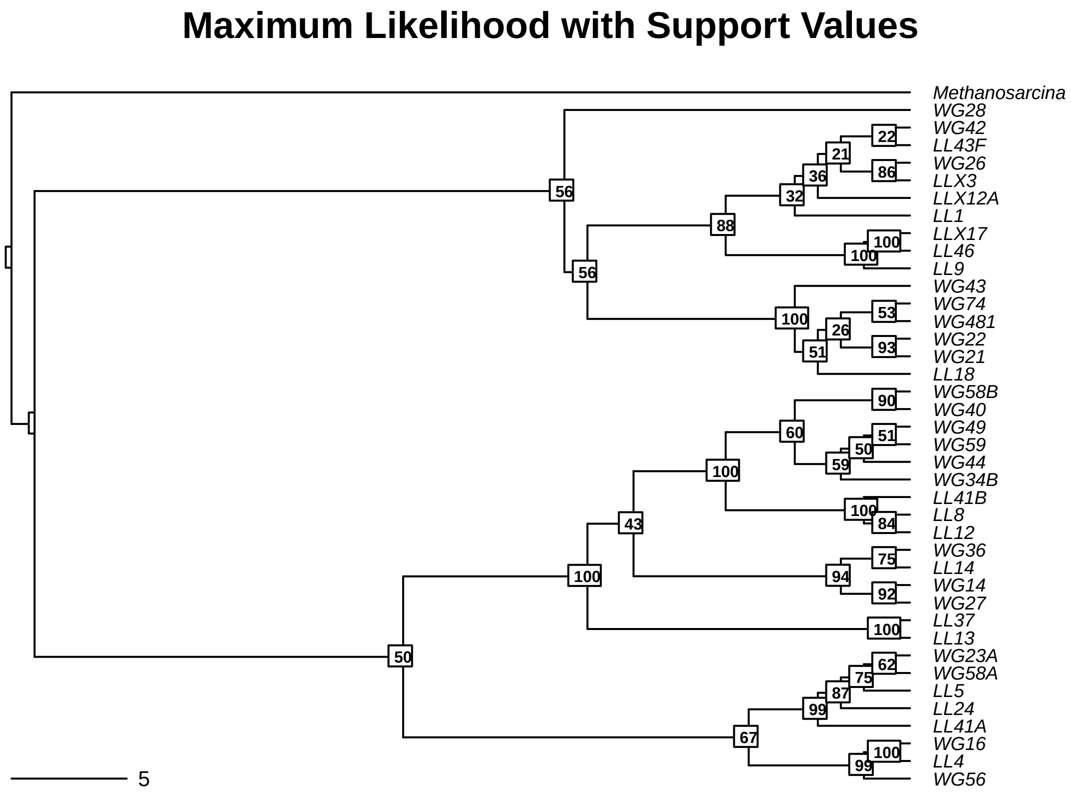
   Answer 4b: My choice of substitution model did not affect the phylogenetic reconstruction- the plots are the same. The saturation plot is essentially a 1:1 relationship between the models.

   Answer 4c: The K80 model is very similar to the F84 model, this hints that base frequencies do not differ and that the substitution rates are similar.

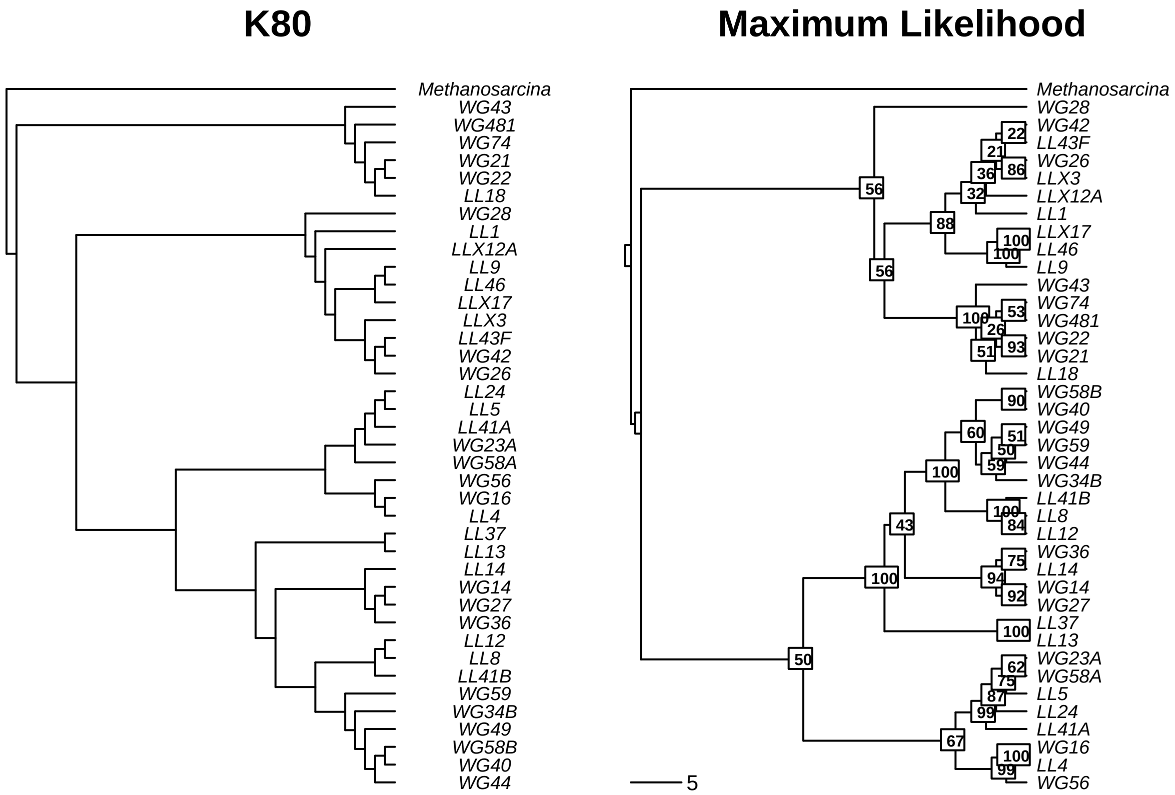
C) ANALYZING A MAXIMUM LIKELIHOOD TREE

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

```
# 1. Read in the maximum likelihood phylogenetic tree used in the handout
ml.bootstrap = read.tree("./data/ml_tree/RAxML_bipartitions.T1")
par(mar = c(1, 1, 2, 1) + 0.1)

# 2. Plot bootstrap support values onto the tree
plot.phylo(ml.bootstrap, type = "phylogram", direction = "right", show.tip.label = TRUE, use.edge.lengt
add.scale.bar(cex = 0.7)
nodelabels(ml.bootstrap$node.label, font = 2, bg = "white", frame = "r",cex = 0.5)
```



```
# Make Cophylogenetic Plot {ape}
layout(matrix(c(1,2), 1, 2), width = c(1, 1))
par(mar = c(1, 1, 2, 0))
par(mar = c(1, 0, 2, 1))
plot.phylo(K80.rooted, type = "phylogram", direction = "right", show.tip.label=TRUE,
           use.edge.length = FALSE, adj = 0.5, cex = 0.6, label.offset = 2, main = "K80")
plot.phylo(ml.bootstrap, type = "phylogram", direction = "right", show.tip.label = TRUE, use.edge.lengt
add.scale.bar(cex = 0.7)
nodelabels(ml.bootstrap$node.label, font = 2, bg = "white", frame = "r",cex = 0.5)
```



K80

Maximum Likelihood

```r
dist.topo(ml.bootstrap,K80.rooted,  method = "score")
```

```
## [1] 0.4634105
```

```r
dev.off()
```

```
## null device
##           1
```

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 Maximum Liklihood tree looks very different from the neighbor joined tree made using the K80 substitution model. With a BLS score = 0.46, the two trees differ quite a bit in their branch length.

Answer 5b: We bootstrap our tree to determine whether or not the placement of each branch in the phylogeny is reliable.

Answer 5c: Boostrap values indicate the level of support for a node. Bootstrapping is a resampling analysis that involves taking columns out of your analysis, rebuilding the tree, and testing to see if the nodes have changed. This is done many times over (100 or 1000). If the same node is recovered through 95 of 100 resampling, then support for that node is high.

Answer 5d: Any branch with a bootstrap value less than 95% is thought to have poor support. Several branches have really low support including: WG42 (22%), LL43F (21%), WG481 (26%).

Answer 5e: We should be cautious when it comes to trusting these branches.

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.

```
# 1. import the raw phosphorus growth data
p.growth = read.table("./data/p.isolates.raw.growth.txt", sep = "\t", header = TRUE, row.names = 1)

# 2. Standadize 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 ($\mu_{max}$) 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.

```
# 1. calculate the maximum growth rate ($\mu_{max}$) of each isolate across all phosphorus types
umax = (apply(p.growth, 1, max))
# 2. create a function that calculates niche breadth (*nb*), and
levins = function(p_xi = ""){ p=0
  for (i in p_xi){
    p = p + i^2
    }
nb = 1 / (length(p_xi) * p)
return(nb)
}
# 3. use this function to calculate *nb* for each isolate.
nb = as.matrix(levins(p.growth.std))
# Add Row & Column Names to Niche Breadth Matrix
rownames(nb) = row.names(p.growth)
colnames(nb) = c("NB")
```

C. Visualizing Traits on Trees

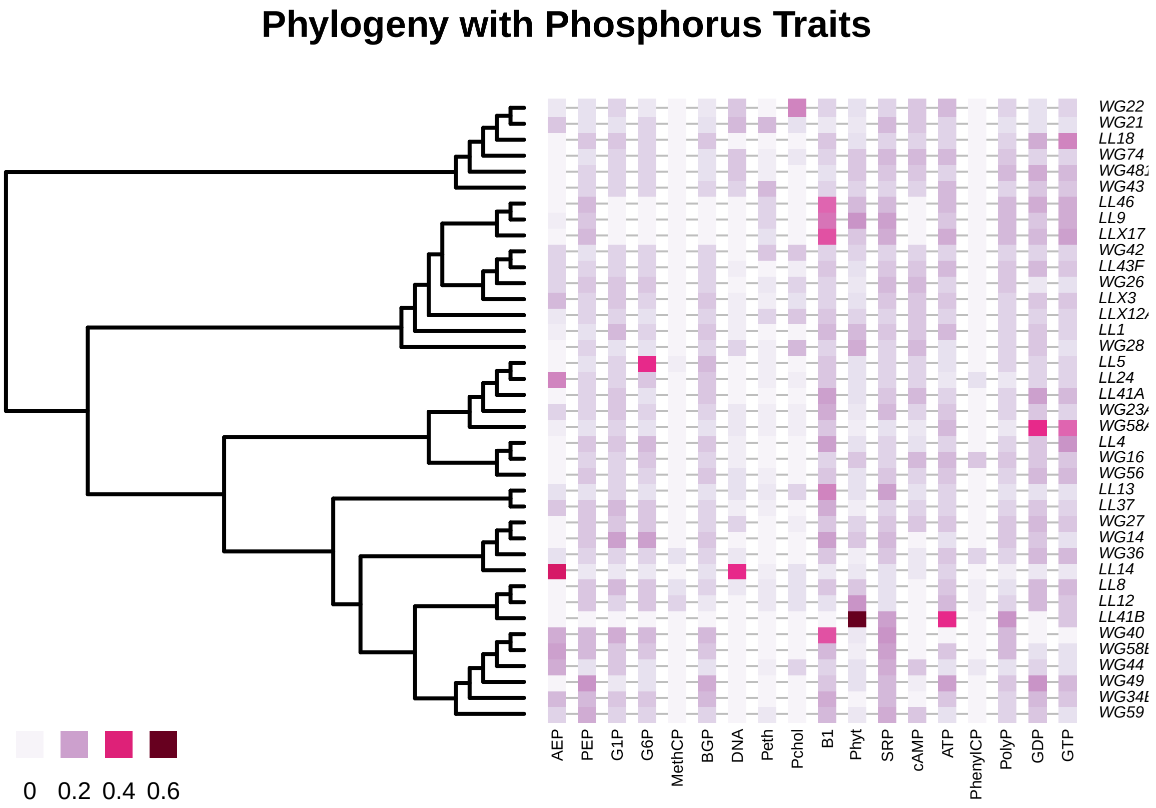
In the R code chunk below, do the following:
1. pick your favorite substitution model and make a Neighbor Joining tree,

2. define your outgroup and root the tree, and
3. remove the outgroup branch.

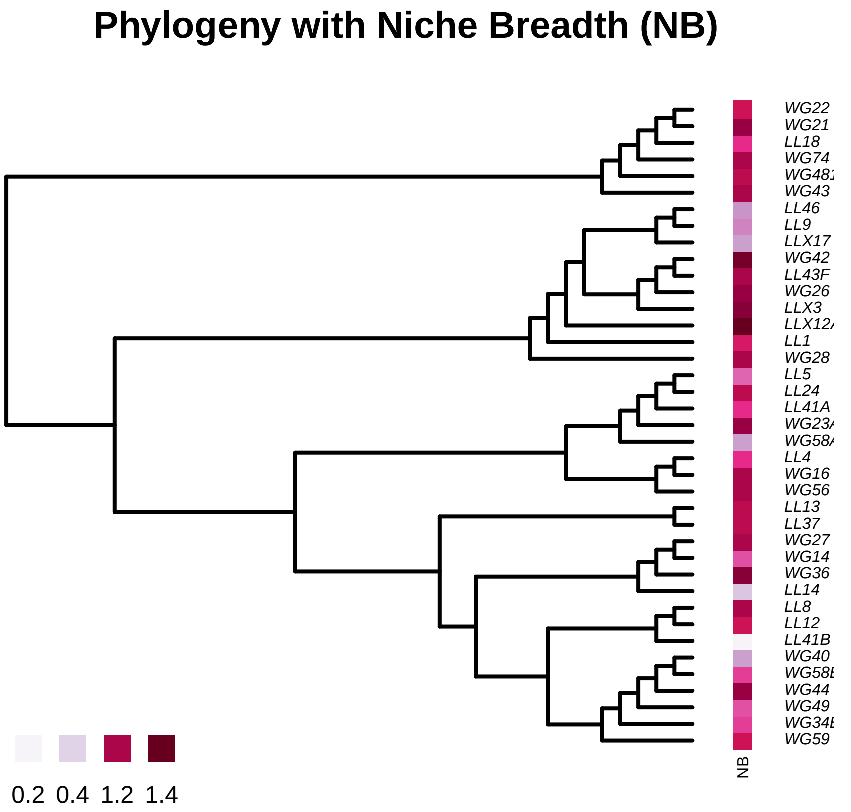
```r
# 1. pick your favorite substitution model and make a Neighbor Joining tree,
nj.tree = bionj(seq.dist.F84)
# 2. define your outgroup and root the tree
outgroup = match("Methanosarcina", nj.tree$tip.label) # Create a Rooted Tree {ape}
nj.rooted = root(nj.tree, outgroup, resolve.root = TRUE)
# 3. remove the 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).

```r
# 1. define a color palette (use something other than "YlOrRd"),
mypalette = colorRampPalette(brewer.pal(9, "PuRd"))

# 2. map the phosphorus traits onto your phylogeny
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 = FAL
title("Phylogeny with Phosphorus Traits")
```



```r
# 3. map the *nb* trait on to your phylogeny and customize the plots as desired (use `help(table.phylo4
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,show.tip.label =TRUE, cex.la
title("Phylogeny with Niche Breadth (NB)")
```



Question 6:

a) Make a hypothesis that would support a generalist-specialist trade-off. the expectation that generalists will have lower maximum growth rates because there is a cost associated with being able to use a lot of different chemical forms of phosphorus. In contrast, we expect that specialists will have a high maximum growth rate on their preferred phosphorus resource.

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

Answer 6a: Hypothesis- Specialists will have high growth rates for one or a few phosphorus substrates, whereas, generalists will have low maximum growth rates for many different phosphorus substrates.

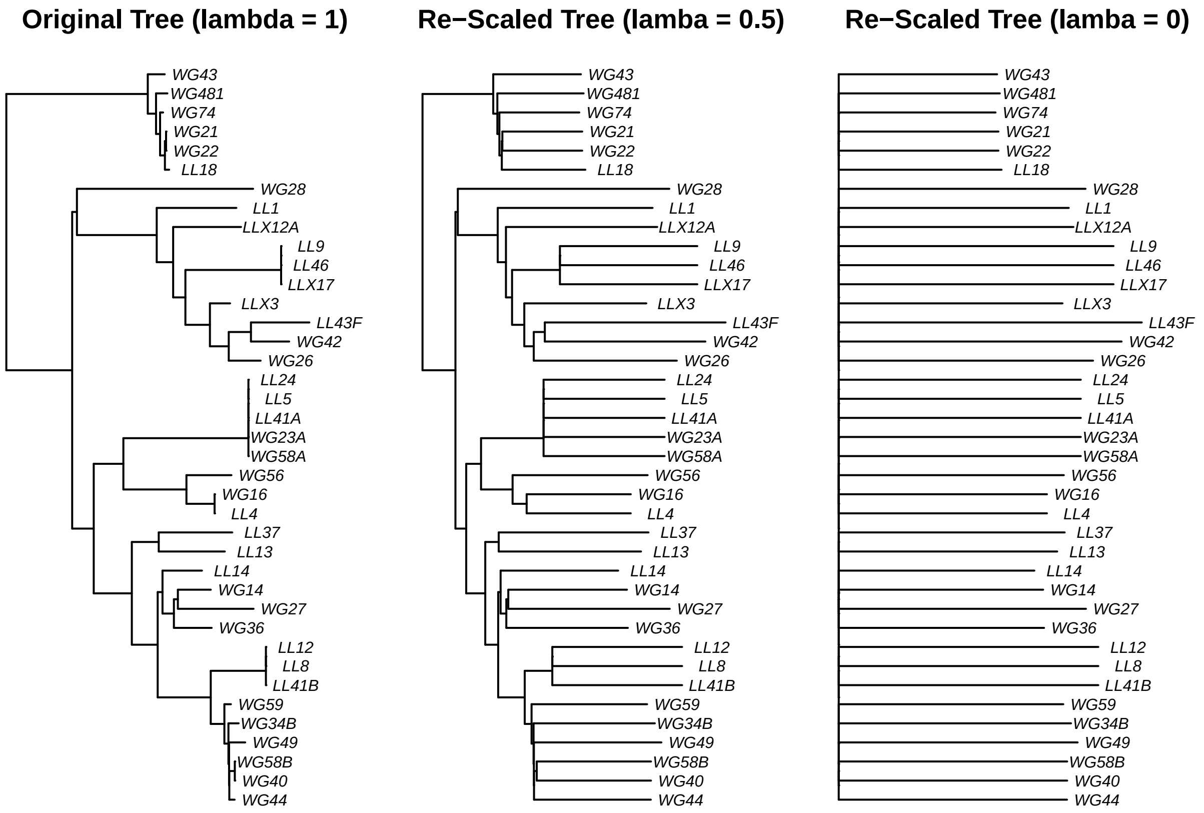
Answer 6b: Species with wide niche breadth (NB values closer to 1) would have low maximum growth rates and species with narrow niche breadth (NB values closer to 0) would have high maximum growth rates. In other words, growth rate decreases with increasing niche breadth.

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.

```
# 1. create two rescaled phylogenetic trees using lambda values of 0.5 and 0
nj.lambda.5 = rescale(nj.rooted, "lambda", 0.5)
nj.lambda.0 = rescale(nj.rooted, "lambda", 0)

# 2. plot your original tree and the two scaled trees, label and customize the trees as desired.
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 = "Original Tree (lambda = 1)", cex = 0.7, adj = 0.5)
plot(nj.lambda.5, main = "Re-Scaled Tree (lamba = 0.5)", cex = 0.7, adj = 0.5)
plot(nj.lambda.0, main = "Re-Scaled Tree (lamba = 0)", cex = 0.7, adj = 0.5)
```



```
# Lambda = 0- no phylogenetic signal
dev.off()
```

```
## null device
##           1
```

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 = 0.020848
##  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 = 46
##  frequency of best fit = NA
##
##  object summary:
##  'lik' -- likelihood function
##  'bnd' -- bounds for likelihood search
##  'res' -- optimization iteration summary
##  'opt' -- maximum likelihood parameter estimates
```

```r
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 = 0.86
##
##  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 for the untransformed tree is equal to 0.021, while the transformed tree has a lambda equal to 0. A lambda equal to 0 means that you have removed all phylogenetic signal from your tree and all branches have equal lengths. The untransformed tree's lambda is close to zero, so there may not be a strong phylogenetic signal.
>
> ***Answer 7b***: The AIC value for the non-transformed lambda is -37.32 and the AIC for the transformed lambda is -37.30. Thus, the AIC values don't really differ.
>
> ***Answer 7c***: This result suggest that there is not a 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
# 1. correct tree branch-lengths to fix any zeros
nj.rooted$edge.length = nj.rooted$edge.length + 10^-7

# Calculate Phylogenetic Signal (K) 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")

# 2. calculate Blomberg's K for each phosphorus resource using the `phylosignal()` function,
# 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)
}

# K value = 1, means that a trait is distributed on the tree according to null expectation (random gene

# 3. use the Benjamini-Hochberg method to correct for false discovery rate
p.phylosignal[6, ] = round(p.adjust(p.phylosignal[4, ], method = "BH"), 3)

# Calcualate 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              50294.09          0.526
##   PIC.variance.Z
## 1    -0.01548366
```

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***: There is a significant phylogenetic signal for standardized growth rate ($P < 0.05$) for the following phosphorus resources: BGP($P = 0.03$), DNA ($P = 0.002$) and cAMP ($P = 0.003$). There is no significant signal for niche breadth under Blomberg’s K ($P = 0.54$). ***Answer 8b***: Where the phylogenetic signal is significant, the results are suggestive of 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.

```
# 1. turn the continuous growth data into categorical data
p.growth.pa = as.data.frame((p.growth > 0.01) * 1)
# Look at Phosphorus Use for Each Resource

# 2. add a column to the data with the isolate name
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)

# 3. combine the tree and trait data using the `comparative.data()` function
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 :  0.4732658
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.007
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.012
```

```r
# 4. use `phylo.d()` to calculate *D* on at least three phosphorus traits
phylo.d(p.traits, binvar = BGP)
```

```
##
## Calculation of D statistic for the phylogenetic structure of a binary variable
##
##   Data :  p.growth.pa
##   Binary variable :  BGP
##   Counts of states:  0 = 4
##                      1 = 35
##   Phylogeny :  nj.rooted
##   Number of permutations :  1000
##
## Estimated D :  -0.3984878
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.002
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.654
```

```r
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 :  0.5992558
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.044
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.005
```

```r
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 :  0.148568
## Probability of E(D) resulting from no (random) phylogenetic structure :  0
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.305
```

```
# When D is close to 0, traits are randomly clumped. When D is close to 1, traits are dispersed in a wa
```

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***: For BGP-D = -0.34, for DNA-D = 0.602, for cAMP-D = 0.125. BGP is significantly clustered, while, DNA and cAMP are significantly overdispersed.
>
> ***Answer 9b***: The results differ from Blomberg's K analysis, in that, BGP is no longer overdispersed.
>
> ***Answer 9c***: Blomberg's K value is calculated as the mean squared error of the trait data measured from the phylogenetic corrected mean and the mean squared error based on the variance-covariance matrix derived from the phylogeny under the assumption of Brownian motion. For the dispersion of a trait (D), a permutation test calculates the probability of D being different from 1 (no phylogenetic structure) or 0 (Brownian phylogenetic structure). This difference in statistical approach may be the reason for the discrepancy in results.

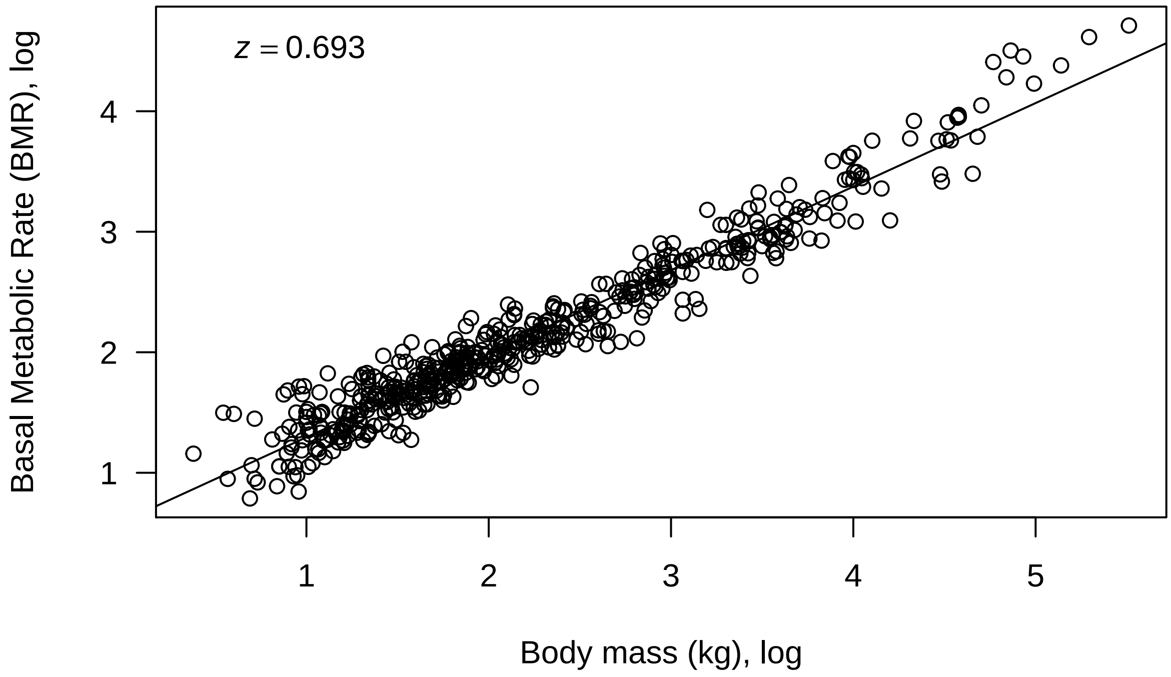
7) PHYLOGENETIC REGRESSION

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, 3. Fit a phylogenetic regression to the trait dataset, taking into account the mammal supertree

```
# 1. Load and clean the mammal phylogeny and trait 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$tip.label[-na.omit(match(mammal.species,
    mammal.Tree$tip.label))])

# Select the species from the dataset that are in our prunned 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

# 2. Fit a linear model to the trait dataset, examining the relationship between mass and BMR
fit = lm(log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.), data = pruned.mammal.data)
summary(fit)
```

```
##
## Call:
## lm(formula = log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.),
##     data = pruned.mammal.data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43832 -0.10172 -0.00950  0.09284  0.53039
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   0.601224   0.018229   32.98   <2e-16 ***
## log10(Body_mass_for_BMR_.gr.) 0.693300   0.007443   93.15   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1694 on 516 degrees of freedom
## Multiple R-squared:  0.9439, Adjusted R-squared:  0.9438
## F-statistic:  8676 on 1 and 516 DF,  p-value: < 2.2e-16
```

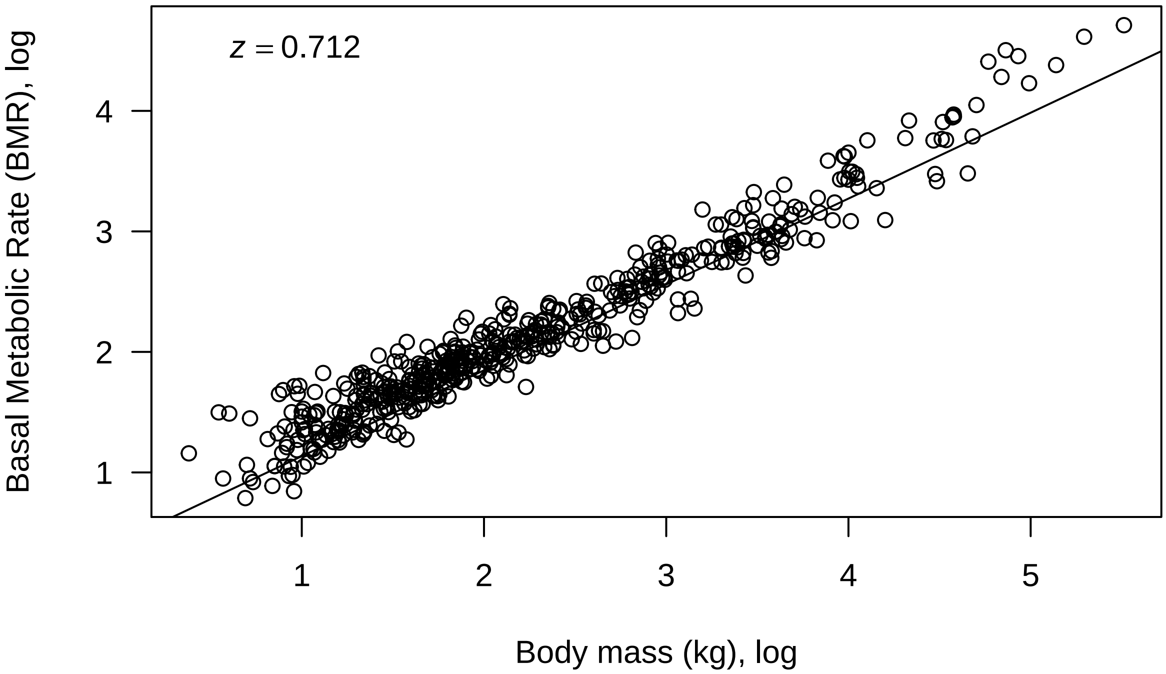
```r
plot(log10(pruned.mammal.data$Body_mass_for_BMR_.gr.), log10(pruned.mammal.data$BMR_.mlO2.hour.),
     las = 1, xlab = "Body mass (kg), log", ylab = "Basal Metabolic Rate (BMR), log")
abline(a = fit$coefficients[1], b = fit$coefficients[2])
b1 = round(fit$coefficients[2], 3)
eqn = bquote(italic(z) == .(b1))
# plot the slope
text(0.5, 4.5, eqn, pos = 4)
```



```r
# 3. Fit a phylogenetic regression to the trait dataset, taking into account the mammal supertree
# 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)
summary(fit.phy)
```

```
##
## Call:
## phylolm(formula = log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.),
##     data = pruned.mammal.data, phy = pruned.mammal.tree, model = "lambda",
##     boot = 0)
##
##    AIC logLik
## -646.9  327.5
##
## Raw residuals:
##      Min       1Q   Median       3Q      Max
## -0.32221  0.03159  0.12863  0.23411  0.68828
##
## Mean tip height: 166.2
## Parameter estimate(s) using ML:
## lambda : 0.8566919
## sigma2: 0.0003072979
##
## Coefficients:
##                               Estimate   StdErr t.value   p.value
## (Intercept)                   0.422397 0.104414  4.0454 6.023e-05 ***
## log10(Body_mass_for_BMR_.gr.) 0.712474 0.010663 66.8182 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Note: p-values are conditional on lambda=0.8566919.
```

```
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)
```



```r
AIC(fit, fit.phy)
```

```
##         df       AIC
## fit      3 -365.5042
## fit.phy  4 -646.9165
```

Question 10: 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***: Evolution plays a role in the distribution of traits among species and we need to correct for shared evolutionary history, if not, our samples will violate the assumption of independence for regression analysis.
>
> ***Answer 10b***: In a phylogenetic regression, the residual errors are described by a variance-covariance matrix that takes into account the branch lengths of the underlying phylogeny. Standard linear regression does not take into account phylogeny.
>
> ***Answer 10c***: The slope of the standard linear regression is 0.69 and the slope of the phylogenetic regression is 0.71- both regression models were statistically significant ($P < 0.01$). Accounting for evolutionary history improved the fit and results in a more positive slope (stronger relationship between mass and basal metabolic rate).
>
> ***Answer 10d***: one possible scenario where phylogenetic signal is strong/ influences the realtionship between traits could be seed mass and plant size or seed dispersal syndrome (air vs animal) for different plant genera.

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 nich breadth vs. μ_{max} and the slope of the regression for each lake. Be sure to color the data from each lake differently.

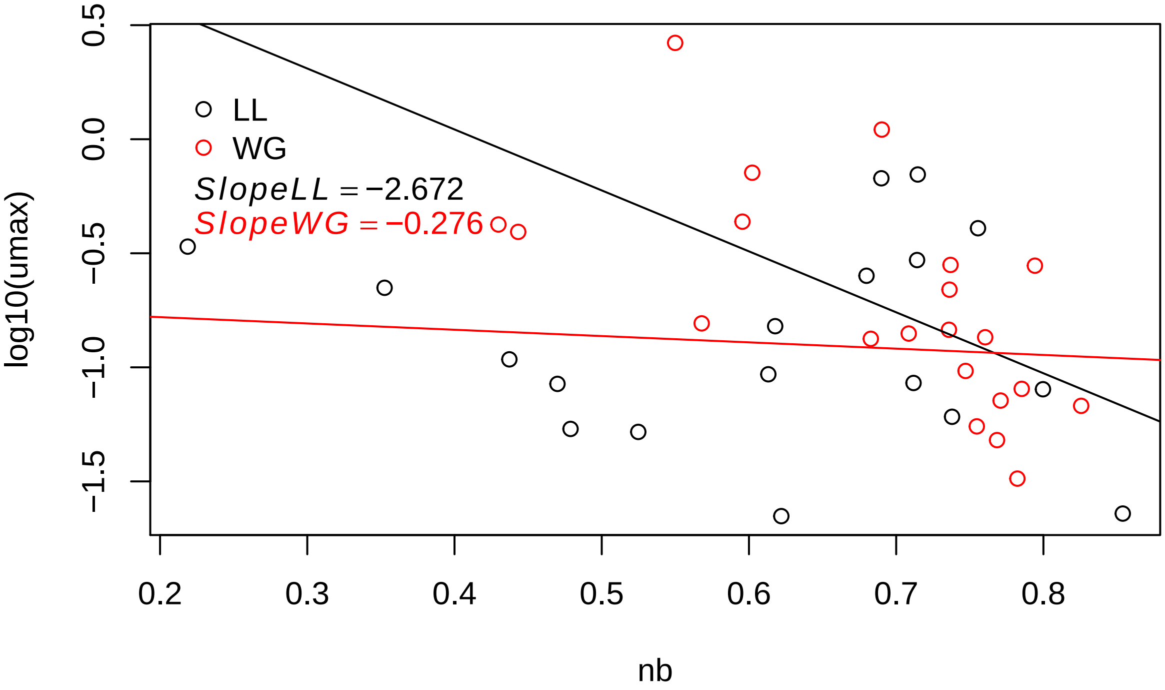
```
## 
## Call:
## lm(formula = log10(umax) ~ nb + D + nb * D)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -0.7557 -0.3108 -0.1077  0.3102  0.7800 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)   
## (Intercept)   1.1117     0.5736   1.938  0.06074 . 
## nb           -2.6721     0.8220  -3.251  0.00255 **
## D            -1.8364     0.6909  -2.658  0.01177 * 
## nb:D          2.3958     1.0234   2.341  0.02506 * 
## ---
## 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
```

```
## 
## Call:
## lm(formula = log10(umax) ~ nb + E + nb * E)
## 
## 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  
## E             1.8364     0.6909   2.658   0.0118 *
## nb:E         -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
```

```
##
## Call:
## lm(formula = log10(umax) ~ NB * lake, data = tradeoff)
##
## 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
```

```
##
## Call:
## lm(formula = log10(umax) ~ NB, data = LL)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7557 -0.2706 -0.1280  0.3144  0.7675
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.7247     0.4253  -1.704    0.108
## NB           -0.2763     0.6732  -0.410    0.687
##
## Residual standard error: 0.4615 on 16 degrees of freedom
## Multiple R-squared:  0.01042,    Adjusted R-squared:  -0.05143
## F-statistic: 0.1684 on 1 and 16 DF,  p-value: 0.687
```

```
##
## Call:
## lm(formula = log10(umax) ~ NB, data = WG)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50903 -0.33338 -0.07422  0.19614  0.78000
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.1117     0.5180   2.146  0.04500 *
## NB           -2.6721     0.7423  -3.600  0.00191 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3774 on 19 degrees of freedom
## Multiple R-squared:  0.4055, Adjusted R-squared:  0.3742
## F-statistic: 12.96 on 1 and 19 DF,  p-value: 0.001909
```



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***: The existence of a a generalist-specialist tradeoff means that growth rate should decrease with increasing niche breadth (wider breadth is characteristic of a generalist). When we plot niche breadth vs. max growth rate, the relationship is negaitve for both lakes, supporting the predictions generated for a generalist-specialist. However, the relationship is only significant (P = 0.002, R2 = 0.37) for Wintergreen Lake, the lake with high phosphorus concentrations. When P is abundant or there is an abundance of different forms of P, there may be sufficient niche differentiation that leads to trade-offs between generalist and specialist species with respect to maximum growth rate and substrate usage.