Phylogenetic Diversity - Traits

Venus Kuo; Z620: Quantitative Biodiversity, Indiana University

18 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).

```r
# Set working directory #
rm(list = ls())
getwd()
```

```
## [1] "C:/Users/Venus/Github/QB2017_Kuo/Week6-PhyloTraits"
```

```
setwd("C:/Users/Venus/Github/QB2017_Kuo/Week6-PhyloTraits/")
#setwd("/Users/vkuo/GitHub/QB2017_Kuo/Week6-PhyloTraits/")

# Require or install packages #
package.list <- c('ape', 'seqinr', 'phylobase', 'adephylo', 'geiger', 'picante', 'stats', 'RColorBrewer
for (package in package.list) {
  if (!require(package, character.only=T, quietly=T)) {
    install.packages(package)
    library(package, character.only=T)
} }
```

```
##
## 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 2.4-1

##
## 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

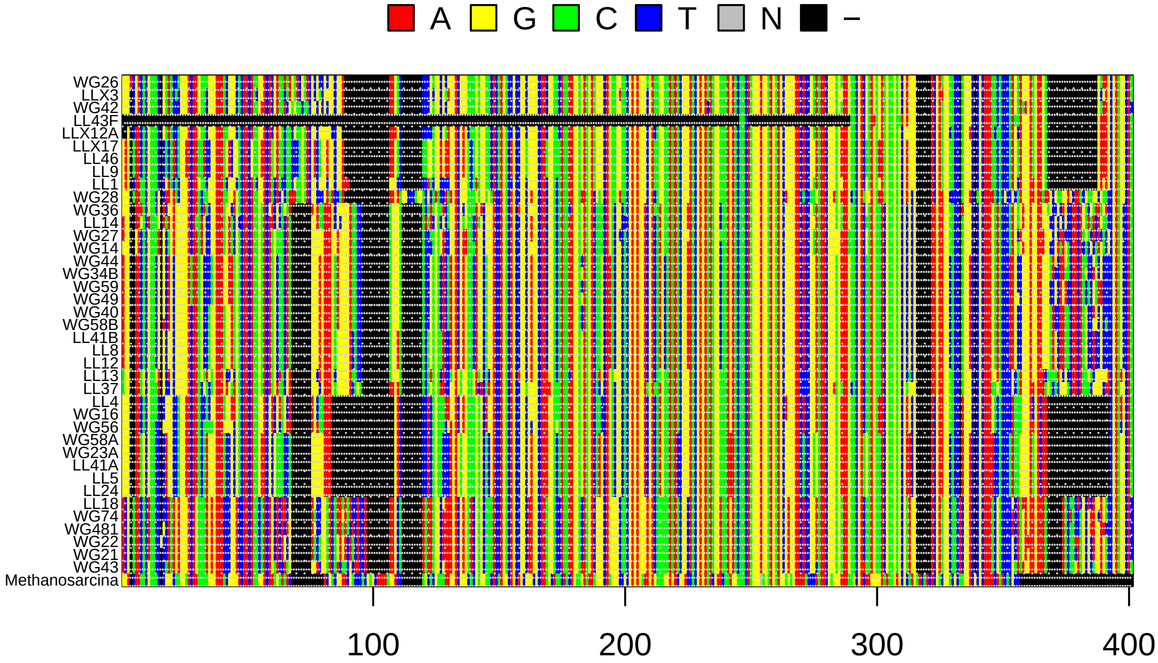
Question 1: Using less or your favorite text editor, compare the `p.isolates.fasta` file and the `p.isolates.afa` file. Describe the differences that you observe between the files.

> ***Answer 1***: The p.isolates.fasta shows the raw base pair sequences of each isolate, while the p.isolates.afa shows the base pair sequence insertations and deletion differences between each isolate sequence region

In the R code chunk below, do the following: 1. read your alignment file, 2. convert the alignment to a DNAbin object, 3. select a region of the gene to visualize (try various regions), and 4. plot the alignment using a grid to visualize rows of sequences.

```r
# Performing Alignment #
# muscle -in ./data/p.isolates.fasta -out ./data/p.isolates.afa # In bash terminal

# Read Alignment File #
read.aln <- read.alignment(file = "./data/p.isolates.afa", format = "fasta")

# Convert Alignment file to DNAbin Object #
p.DNAbin <- as.DNAbin(read.aln)

# Identify Base Pair Region of 16S rRNA Gene to Visualize #
window <- p.DNAbin[, 100:500]
```

```r
# Command to visualize sequence alignment #
image.DNAbin(window, cex.lab = 0.50)

# 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 are appropriate for phylogenetic inference and why?

   Answer 2a: Our reads are approximately 400 bp in length. ***Answer 2b***: The variable region is probably the most appropiate for phylogenetic inference becuase it would show the strongest difference in phylogentic signal.

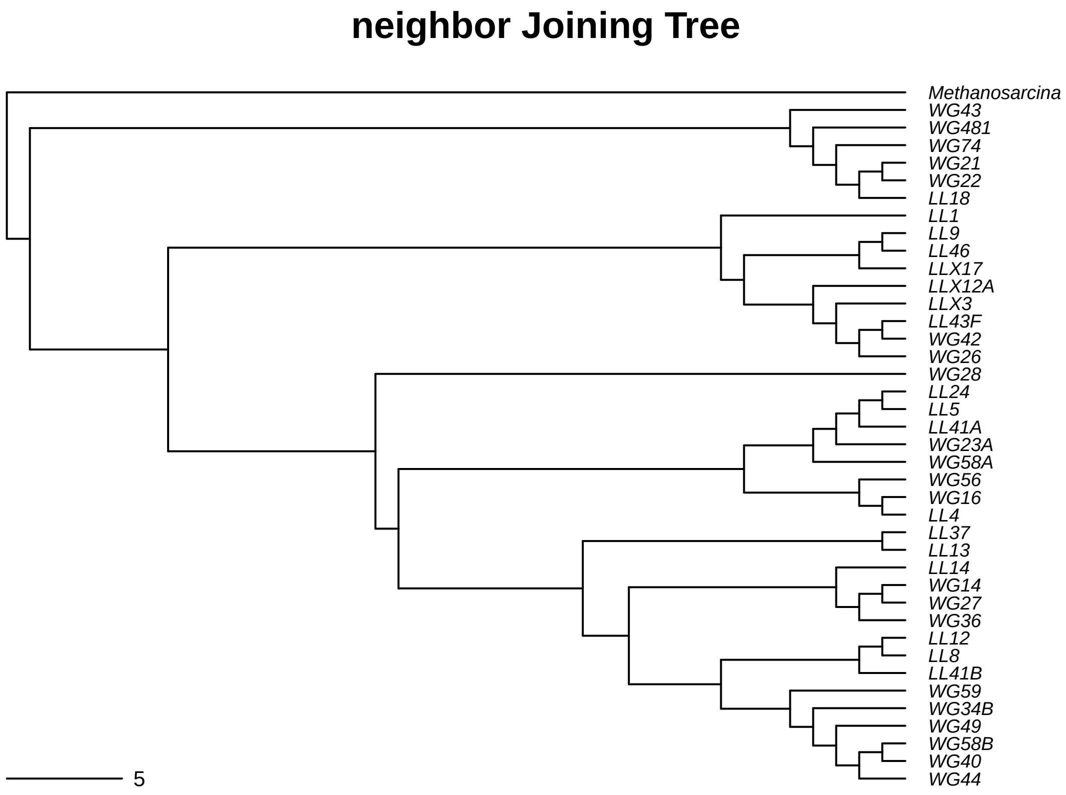
4) MAKING A PHYLOGENETIC TREE

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 Matri with "raw" model #
seq.dist.raw <- dist.dna(p.DNAbin, model = "raw", pairwise.deletion = FALSE)

# Neighbor Joining Algorithm to construct 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 with 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)
```



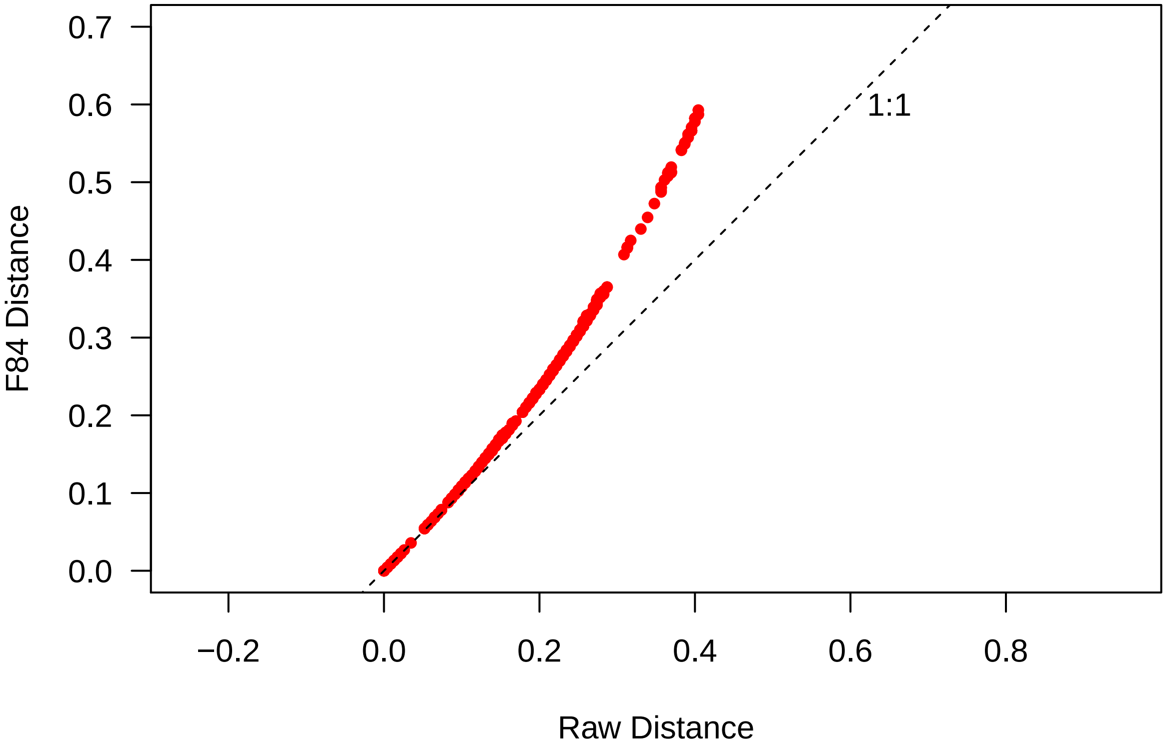
Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

Answer 3: Neighbor joining tree uses a distance matrix that specifies the distance between each pair of taxa as the input. The algorthium pairs up more similar (lowest distance) taxa with each other and repeats the process until all taxa are accounted for. The neighbor joining tree is a good way to start assessing evolutionary relatedness among groups of organisms becuase it requires little information and uses a relatively simple algorithm to make a fairly accurate phylogenetic tree. The NJ tree is probably not as comprehensive as for example the Maximum liklihood tree method that incorporates more than just distance matrix into its algorthium.

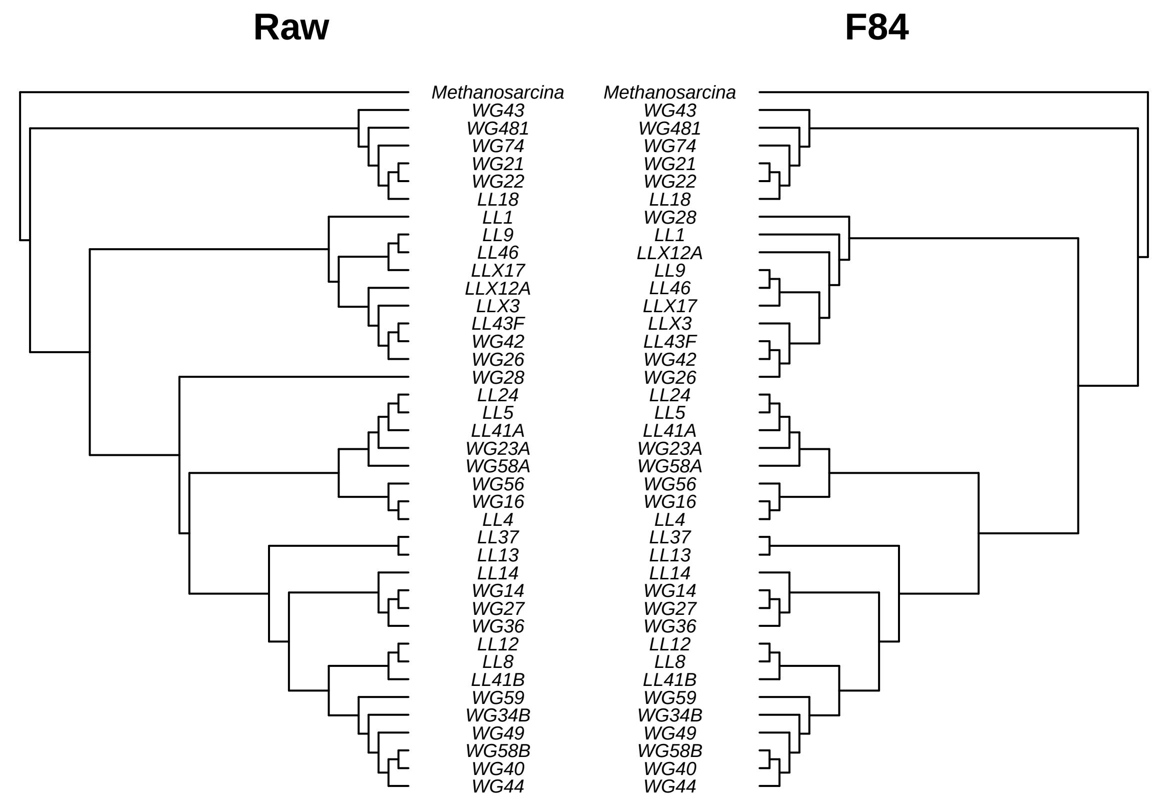
B) SUBSTITUTION MODELS OF DNA EVOLUTION

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
# Create distance matrix with F84 model #
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", ylab = "F84 Distance")
abline(b=1, a=0, lty=2)
text(0.65, 0.6, "1:1")
```



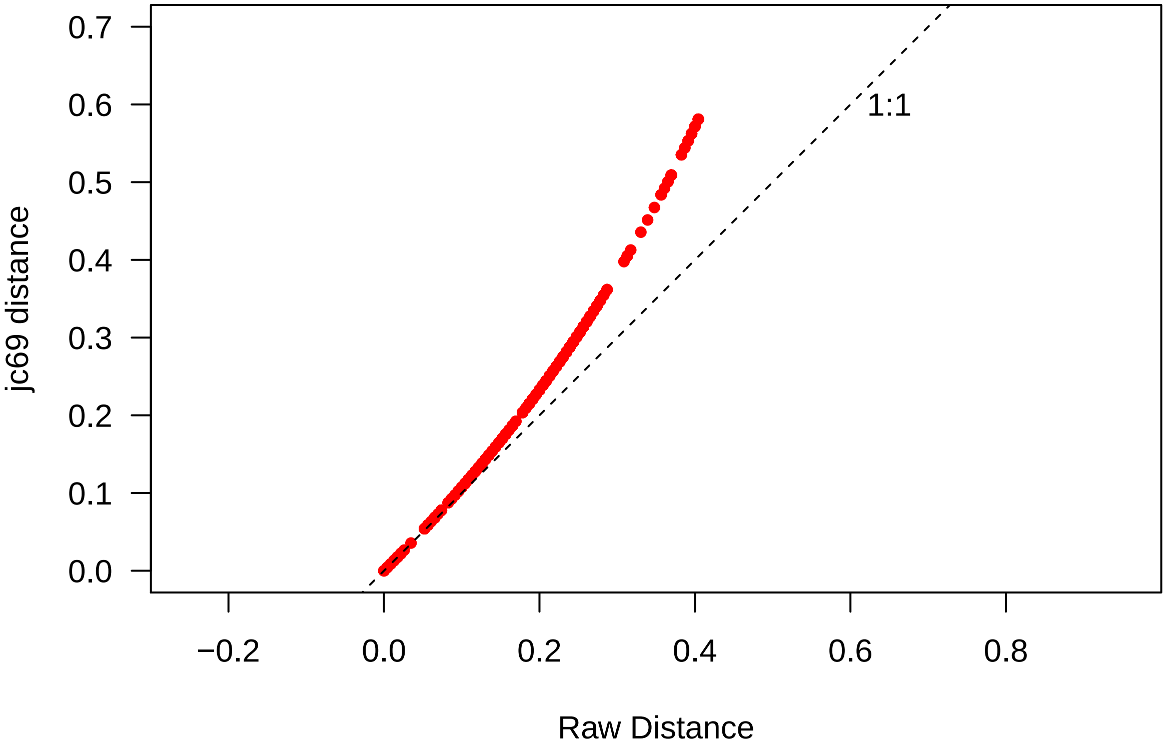
```r
# Make Neighbor Joining Trees Usng Different DNA Substitution Models #
raw.tree <- bionj(seq.dist.raw)
F84.tree <- bionj(seq.dist.F84)

# Define Outgroups #
raw.outgroup <- match("Methanosarcina", raw.tree$tip.label)
F84.outgroup <- match("Methanosarcina", F84.tree$tip.label)

# Root the tree #
raw.rooted <- root(raw.tree, raw.outgroup, resolve.root=TRUE)
F84.rooted <- root(F84.tree, F84.outgroup, resolve.root=TRUE)

# Make Cophylogenetic Plot #
layout(matrix(c(1,2), 1,2), width=c(1,1))
par(mar = c(1,1,2,0))
plot.phylo(raw.rooted, type="phylogram", direction = "right", show.tip.label=TRUE,
           use.edge.length=FALSE, adj=0.5, cex = 0.6, label.offset = 2, main="Raw")

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



In the R code chunk below, do the following:
1. pick another substitution model,
2. create and distance matrix and tree for this model,
3. make a saturation plot that compares that model to the *Felsenstein (F84)* model,

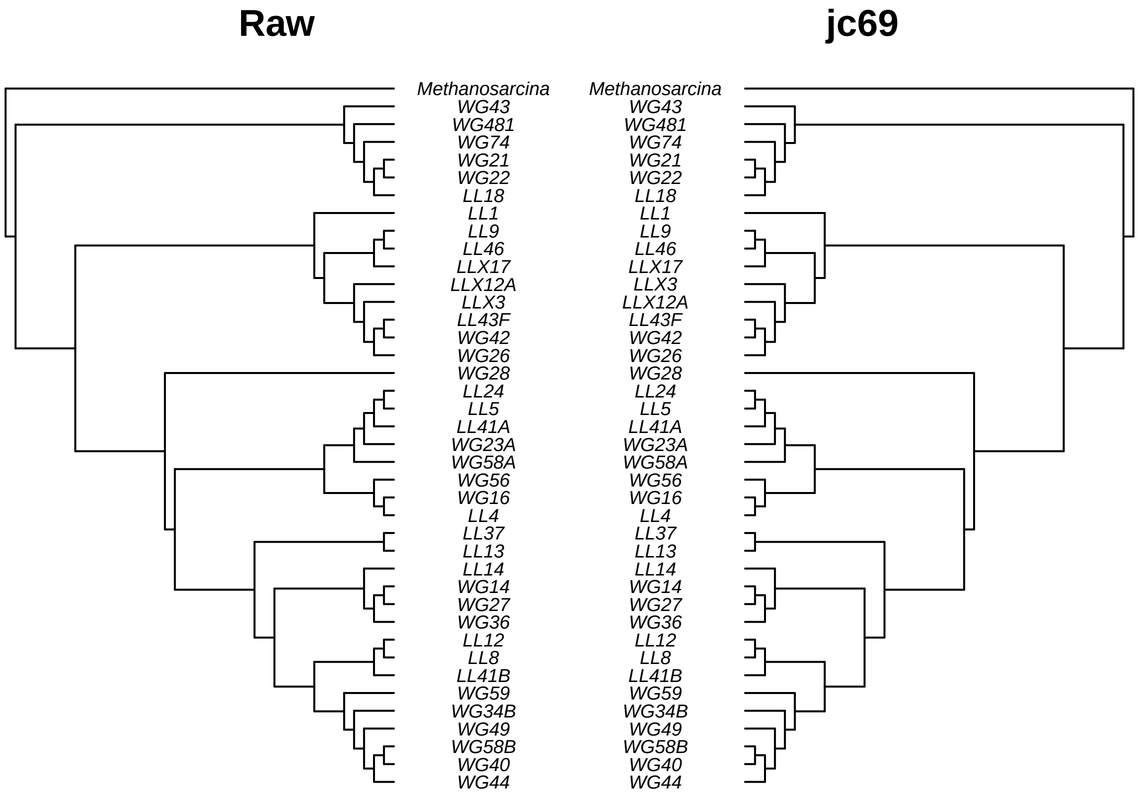
4. make a cophylogenetic plot that compares the topologies of both models, and
5. be sure to format, add appropriate labels, and customize each plot.

```
# Choose Jukes-Cantor model #
# Create distance matrix #
seq.dist.jc69 <- dist.dna(p.DNAbin, model = "JC69", 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.jc69,
     pch=20, col="red", las = 1, asp = 1, xlim=c(0,0.7), ylim=c(0,0.7),
     xlab = "Raw Distance", ylab = "jc69 distance")
abline(b=1, a=0, lty=2)
text(0.65, 0.6, "1:1")
```



```
# Make Neighbor Joining Trees Usng Different DNA Substitution Models #
raw.tree <- bionj(seq.dist.raw)
jc69.tree <- bionj(seq.dist.jc69)

# Define Outgroups #
raw.outgroup <- match("Methanosarcina", raw.tree$tip.label)
jc69.outgroup <- match("Methanosarcina", jc69.tree$tip.label)

# Root the tree #
raw.rooted <- root(raw.tree, raw.outgroup, resolve.root=TRUE)
jc69.rooted <- root(jc69.tree, F84.outgroup, resolve.root=TRUE)
```

```r
# Make Cophylogenetic Plot #
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(jc69.rooted, type="phylogram", direction = "left", show.tip.label = TRUE,
           use.edge.length = FALSE, adj = 0.5, cex = 0.6, label.offset = 2, main="jc69")
```



Question 4:

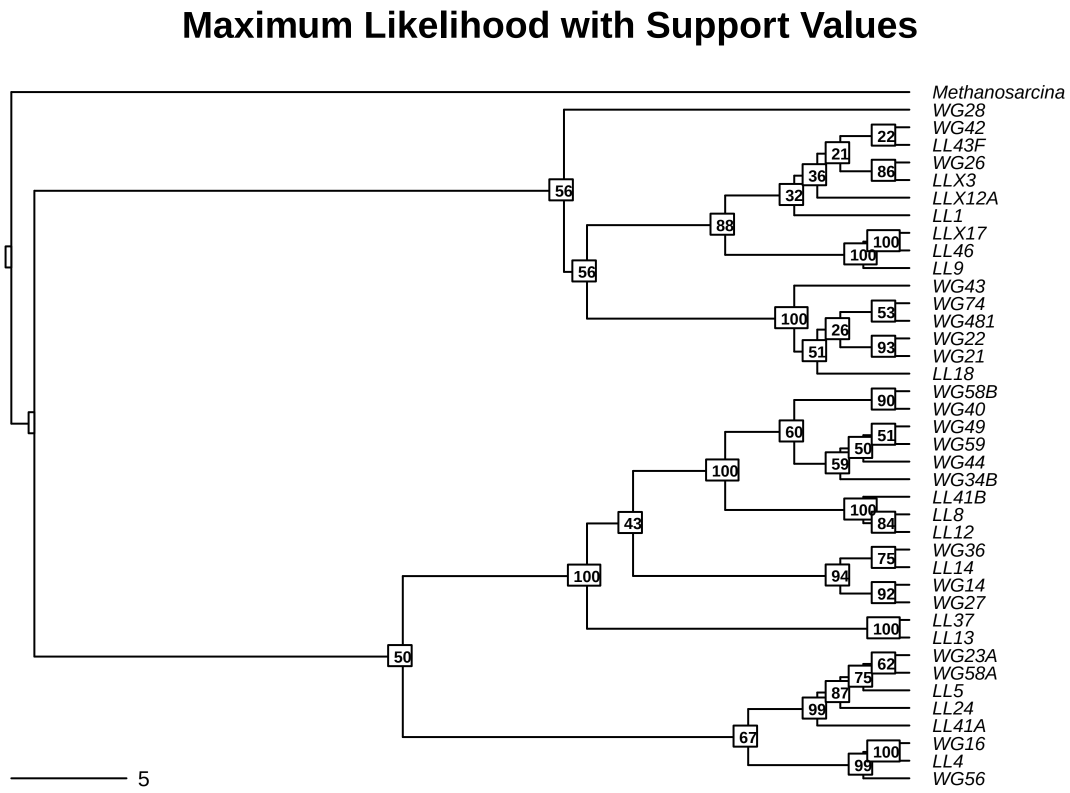
a. Describe the substitution model that you chose. What assumptions does it make and how does it compare to the F84 model?
b. Using the saturation plot and cophylogenetic plots from above, describe how your choice of substitution model affects your phylogenetic reconstruction. If the plots are inconsistent 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 that I chose was the Jukes-Cantor model (jc69). The JC69 model is the simplest model that assumes equal frequencies of nucleotides and same probabilities of nucleotide substitution/mutation. The F84 model assumes different rates of base frequencies as well as transitions and transversions. ***Answer 4b***: The saturation plots between the Jukes-Cantor and the Felsenstein model were similar in that both the jc69 and F84 distance tended to b greater than the raw distance/ correcting for multiple substitutions. The phylogenetic reconstruction between the two models looked similar for the most part, however, it was clear that some taxa phylogeny were arranged differently once assuming different rates of base transitions/transversions and different base frequencies. So perhaps adding a level of complexity (unequal nucleotide frequency and substitutions/mutuation) created a tree that resolved the relatedness of some taxa (e.g. WG28) to all the other taxa. ***Answer 4c***: The JC69 model is the simplest evolutionary model to start with, but it seems clear from the F84 model tree that the substitution rates of nucleotide transisiton may be unequal between the taxa in this dataset.

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(fitGTR, model="GTR", optInv=TRUE, optGamma=TRUE)
# Model selection with either an ANOVA test or by AIC value #
#anova(fitJC, fitGTR)
#AIC(fitJC)
#AIC(fitGTR)

# Perform bootstrap test to see how well-supported the edges are #
#bs = bootstrap.pml(fitJC, bs=100, optNni=TRUE,
#                  control=pml.control(trace=0))

# Read in Maximum likelihood tree #
ml.bootstrap <- read.tree("./data/ml_tree/RAxML_bipartitions.T1")
par(mar = c(1,1,2,1)+0.1)
plot.phylo(ml.bootstrap, type="phylogram", direction="right",
           show.tip.label=TRUE, use.edge.length=FALSE, cex=0.6,
           label.offset=1, main="Maximum Likelihood with Support Values")
add.scale.bar(cex=0.7)
nodelabels(ml.bootstrap$node.label, font=2, bg="white", frame="r", cex=0.5)
```



Question 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 likelihood and neighbor-joining tree appears to be very different from each other. There is this clear 2 major group splitting in the Maximum likelihood tree that was not seen in the neighbor-joining tree. There is also clear disagreement between the two models at the scale of which two taxa are most related. These differences in tree phylogeny may be a reflection of the tree method properities. The neighbor-joining tree identifies relatedness between taxa iteratively starting from 2 closely related taxa, while maximum likelihood tree identifies the tree relationship that best predicts the taxa sequences at hand. For instance, the tree topology of the NJ tree may change if another taxa (one closely related to a taxa in the dataset) was introduced into the dataset, but it would not change for the ML tree. ***Answer 5b***: We use bootstrapping because constructing trees using any model can give us different topologies, none of them are more correct but generating 100-10,000 simulations and comparing those outputs to each other can give us a tree that is more often than not the "correct" tree. ***Answer 5c***: The bootstrap values tell us the number of simulations where that particular node branching agreed. ***Answer 5d***: The lower the bootstrap values are in the nodes, the lower the support is for those branches. For example, WG42 and LL43F has low bootstrap values, so that relationship is probably not as well supported. ***Answer 5e***: Probably not.

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 #
# Importing growth rate data #
p.growth <- read.table("./data/p.isolates.raw.growth.txt", sep="\t", header=TRUE,
                       row.names=1)
# Standerdize 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.

```
# Trait manipulations #
# Calculate Max Growth Rate #
umax <- (apply(p.growth, 1, max))
# Niche Breadth #
levins <- function(p_xi = ""){
  p=0
  for(i in p_xi){
    p=p+i^2
  }
  nb = 1/ (length(p_xi)*p)
  return(nb)
}

# Calculate the nb of 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

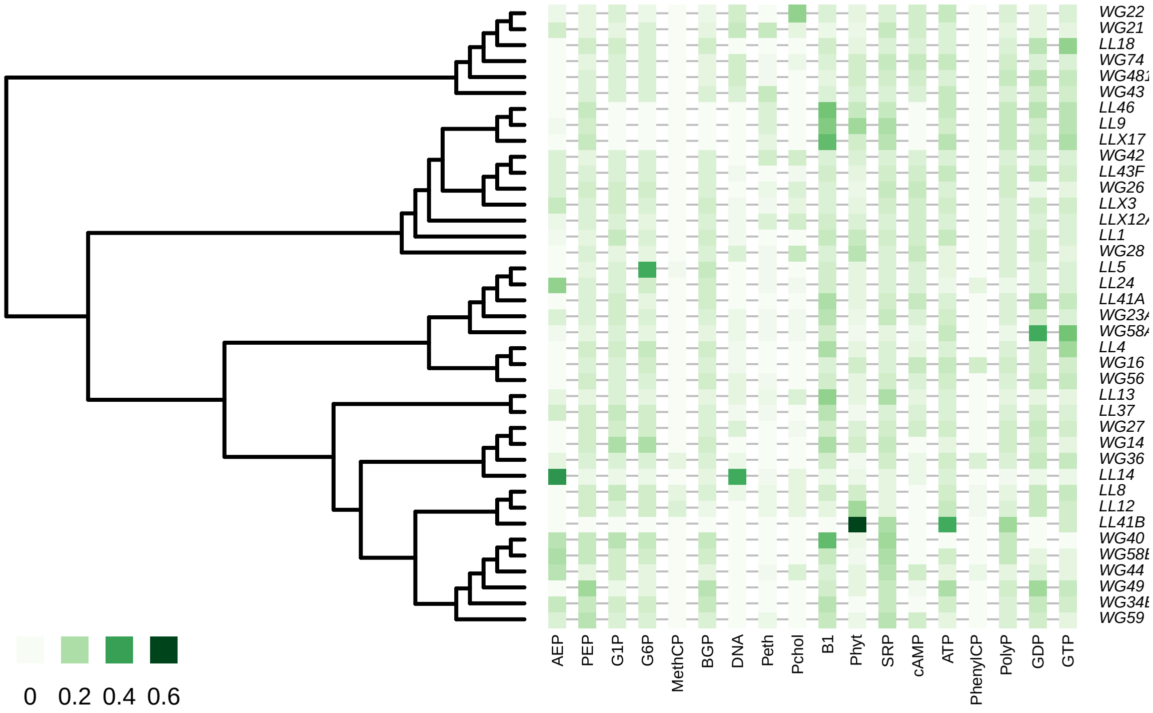
In the R code chunk below, do the following:
1. pick your favorite substitution model and make a Neighbor Joining tree,
2. define your outgroup and root the tree, and
3. remove the outgroup branch.

```
# May favoriate substiution model is the F84 model, Generate NJ tree #
nj.tree <- bionj(seq.dist.F84)

# Define the outgroup #
outgroup <- match("Methanosarcina", nj.tree$tip.label)
```

```r
# create a rooted tree #
nj.rooted <- root(nj.tree, outgroup, resolve.root = TRUE)

# Keep rooted but drop outgroup branch #
nj.rooted <- drop.tip(nj.rooted, "Methanosarcina")
```

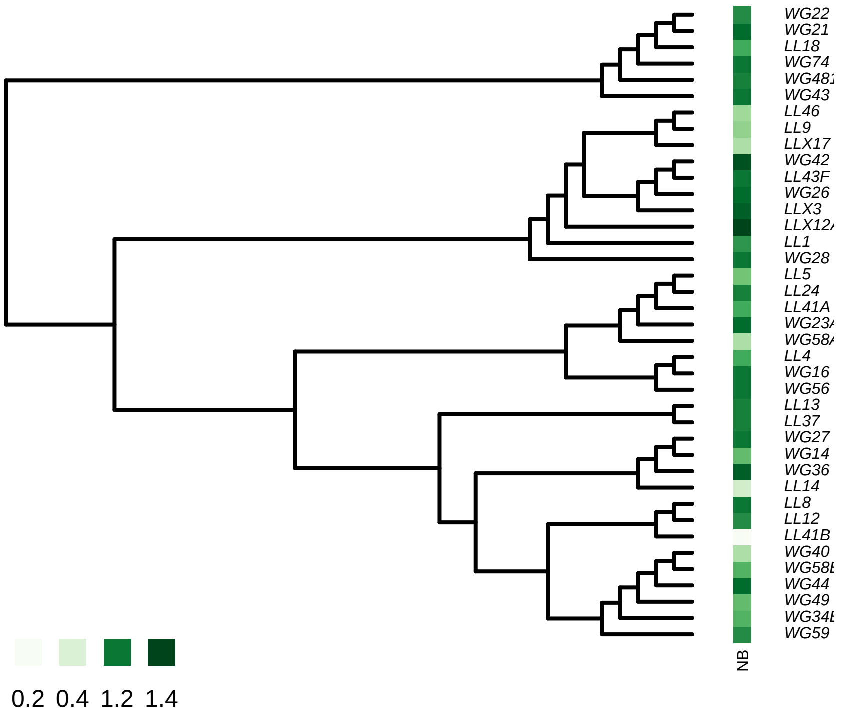
In the R code chunk below, do the following:
1. define a color palette (use something other than "YlOrRd"),
2. map the phosphorus traits onto your phylogeny,
3. map the *nb* trait on to your phylogeny, and
4. customize the plots as desired (use `help(table.phylo4d)` to learn about the options).

```r
# Define color palette #
mypalette <- colorRampPalette(brewer.pal(9, "Greens")) #My favoriate color is Green, suck it colorblink

# Map phosphorus traits onto 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=FALSE,
              col=mypalette(25), pch=15, cex.symbol = 1.25,
              ratio.tree = 0.5, cex.legend = 1.5, center = FALSE)
```

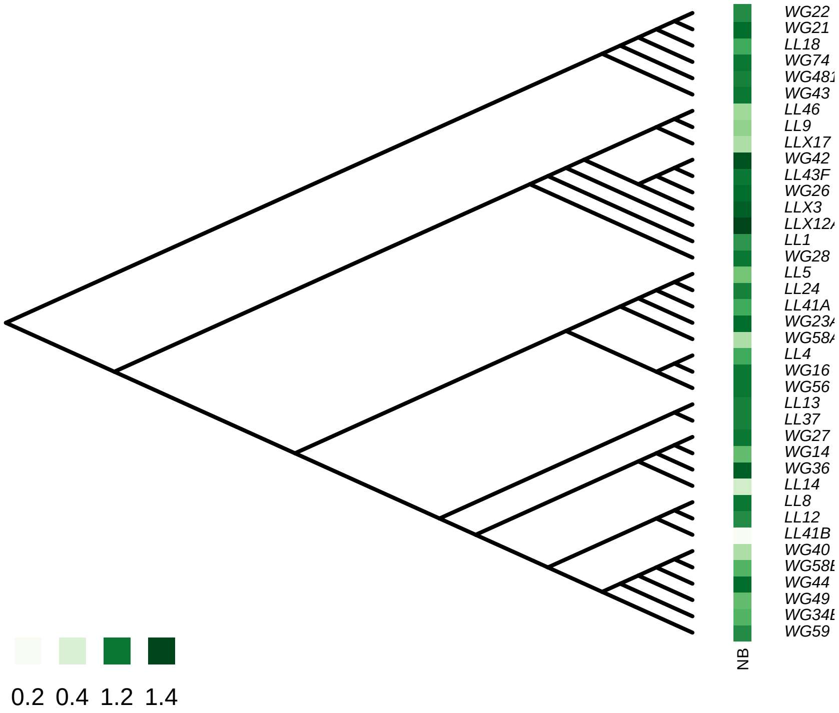


```r
# Mapping niche breadth #
par(mar=c(1,5,1,5)+0.1)
```

```
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)
```



```
# Using arguements in phylo4d #
par(mar=c(1,5,1,5)+0.1)
x.nb <- phylo4d(nj.rooted, nb)
table.phylo4d(x.nb, treetype = "clado", symbol = "colors", show.node = TRUE,
              cex.label=0.5, scale = FALSE, use.edge.length=FALSE,
              edge.color="black", edge.width=2, box=FALSE,
              col=mypalette(25), pch=15, cex.symbol = 1.25, var.label = ("   NB"),
              ratio.tree = 0.90, cex.legend = 1.5, center = FALSE)
```



Question 6:

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

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

   Answer 6a: If some taxa are capable of efficently using particular types of substrates then it may be competitively inferior at competiting for a broad range of substrate types. ***Answer 6b***: For specialists taxa that exhibit a strong utilization for one type of substrate, it is probably less likely to use other types of substrates. Generalists taxa would have a even use of many different substrate types.

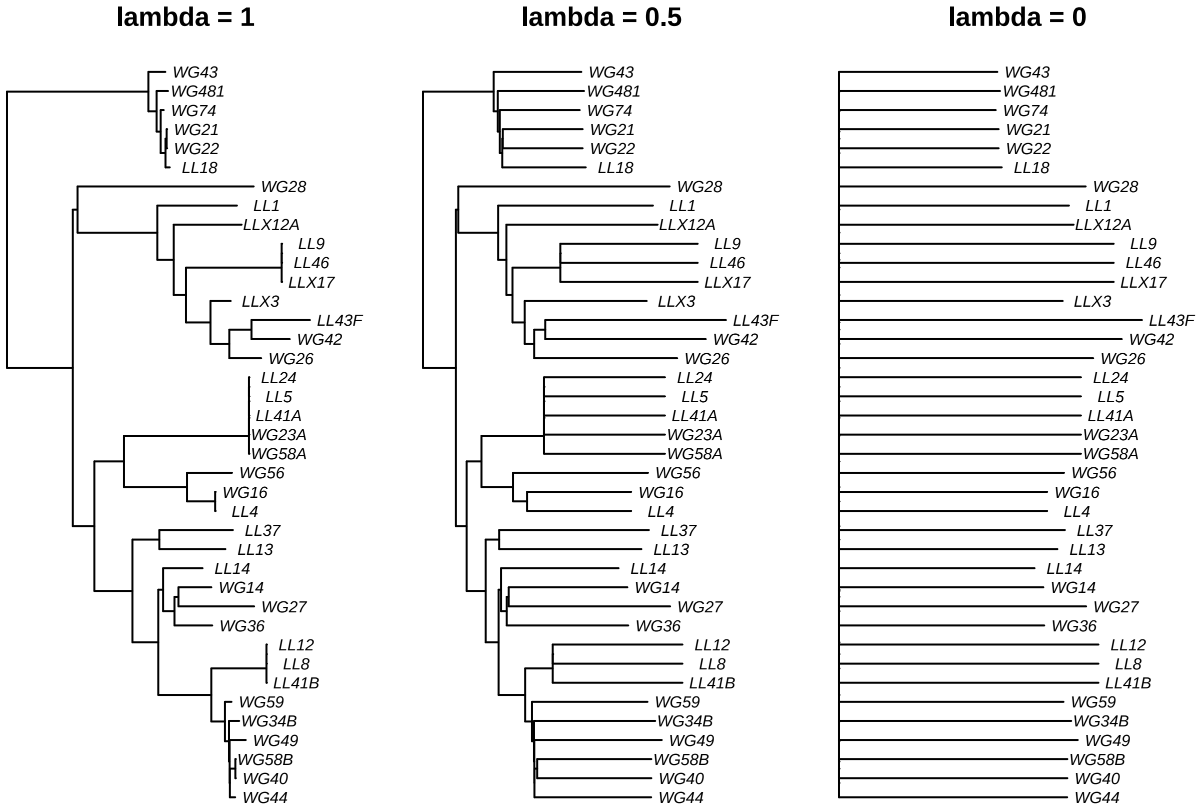
6) HYPOTHESIS TESTING

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
# Visualize Trees with Different Levels of Phylogenetic Signal #
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))
```

```r
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.

```r
# Generate Test Statistics for Comparing Phylogenetic signal #
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 = 42
##  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 = 0.88
##
##  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 and transformed tree is 0.02 and 0, respectively. ***Answer 7b***: The AIC scores for both the transformed and untransformed trees were ~-37. I would choose either one since the difference in AIC values were negliable. ***Answer 7c***: The results suggest that there's 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.

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

# Calculate phylogenetic signal for growth on all phosphorus resources #
# 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 bloomberg'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 dorrection on p-value #
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 0.3795289     1.119536e-09           1.26052e-09          0.185
##   PIC.variance.Z
## 1     -0.9081864
```

```
signal.p <- phylosignal(x, nj.rooted)
signal.p
```

```
##           K PIC.variance.obs PIC.variance.rnd.mean PIC.variance.P
## 1 0.3368096     6.205486e-11           7.48263e-11          0.078
##   PIC.variance.Z
## 1      -1.480335
```

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 not phylogenetic signal for niche breadth (~0.1) or for phosphorus resource use (0.077). ***Answer 8b***: If there is a phlogenetic signal then the results would suggest clustering of either niche breadth or phosphorus use because the traits are conserved based on phylogeny.

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

```r
# 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
```

```r
# Add names column to data #
p.growth.pa$name <- rownames(p.growth.pa)

# Merge Traits and Phylogenetic Data #
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 :  0.2026917
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.002
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.309
```

```r
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 :  0.6746742
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.133
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.138
```

```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 :  0.5107976
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.05
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.098
```

```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 :  -0.1395095
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.001
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.619
```

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.

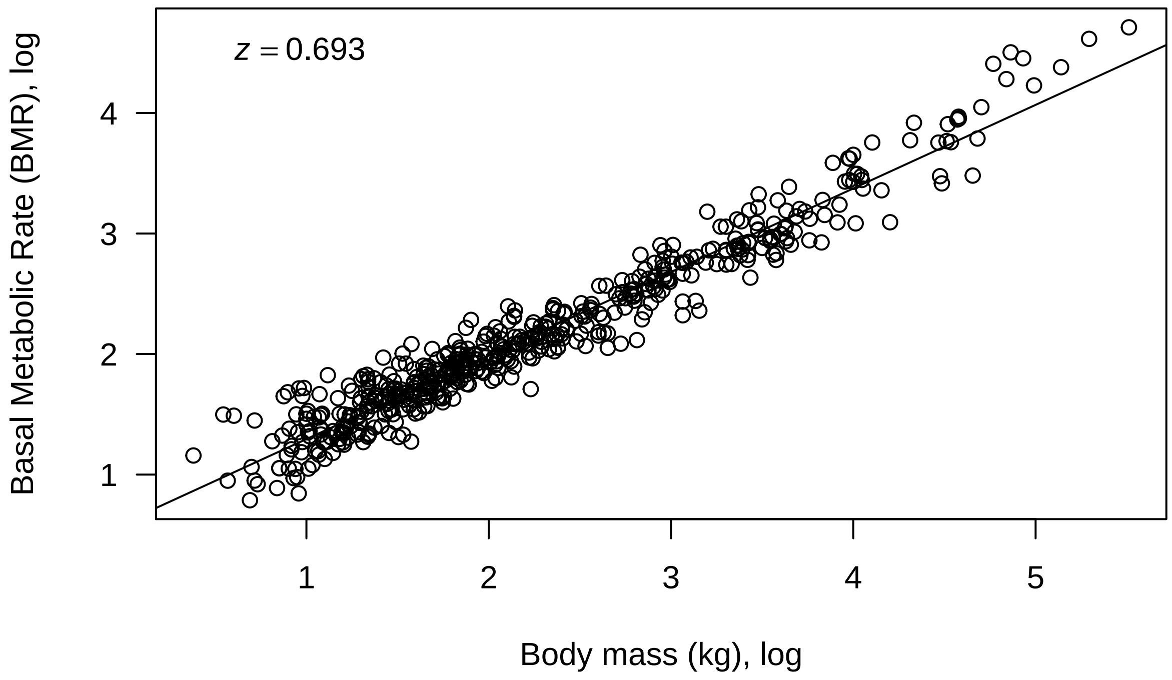
    Answer 9a: I chose AEP, PhenylCP and cAMP growth traits and the D values were 0.19, 0.67, and -0.11 respectively. It appears the cAMP and AEP phosphorus growth traits are clustered, while PhenylCP trait is overdispersed. ***Answer 9b***: These results suggests that there is a phylogenetic signal with taxa phosphorus traits, but the Blomberg's K analysis suggests that there isn't a phylogenetic signal.
    Answer 9c: Blomberg's K quantifies phylogentic singal by compairing observed trait distribution on a tree to what would evolve randomly (Brownian's motion). While dispersion (D) is a measure of how traits are actually dispersed in the tree. Perhaps Blomberg's K is a bit more rigorous of a phylogeny signal test than dispersion measurement.

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

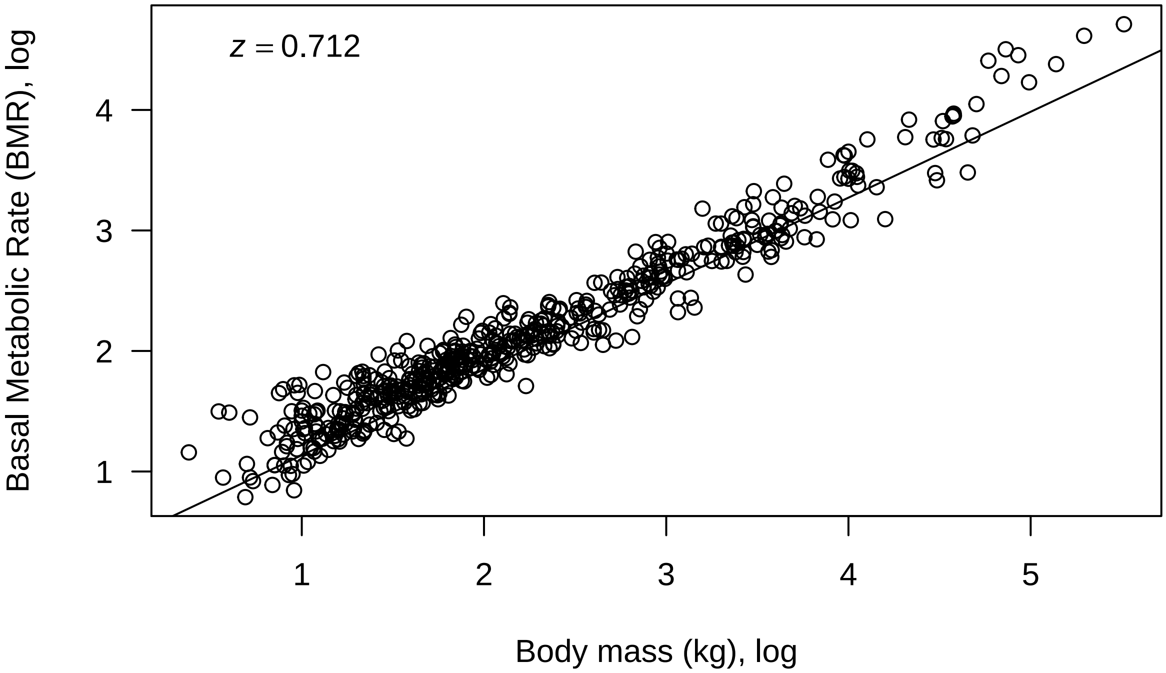
```r
# Input the tree and dataset #
mammal.Tree <- read.tree("./data/mammal_best_super_tree_fritz2009.tre")
mammal.data <- read.table("./data/mammal_BMR.txt",
                          header = TRUE)
# Select the variables we want to analyse #
mammal.data <- mammal.data[, c("Species", "BMR_.mlO2.hour.",
                               "Body_mass_for_BMR_.gr.")]
```

```r
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 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

# Run 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)
```



```r
# Run 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)
```

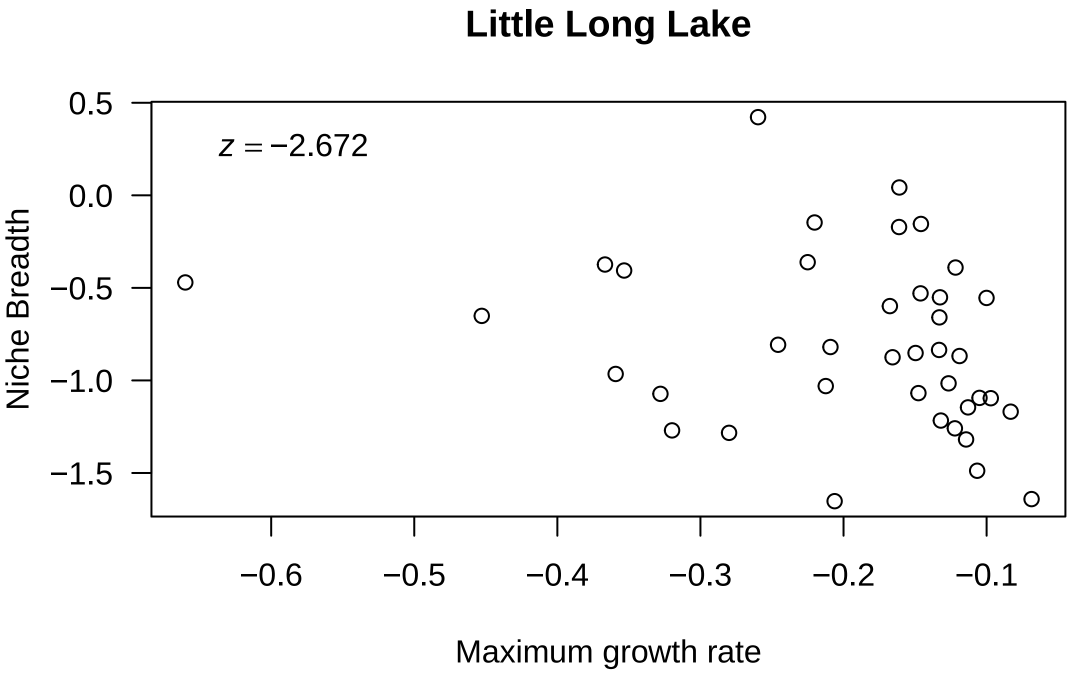


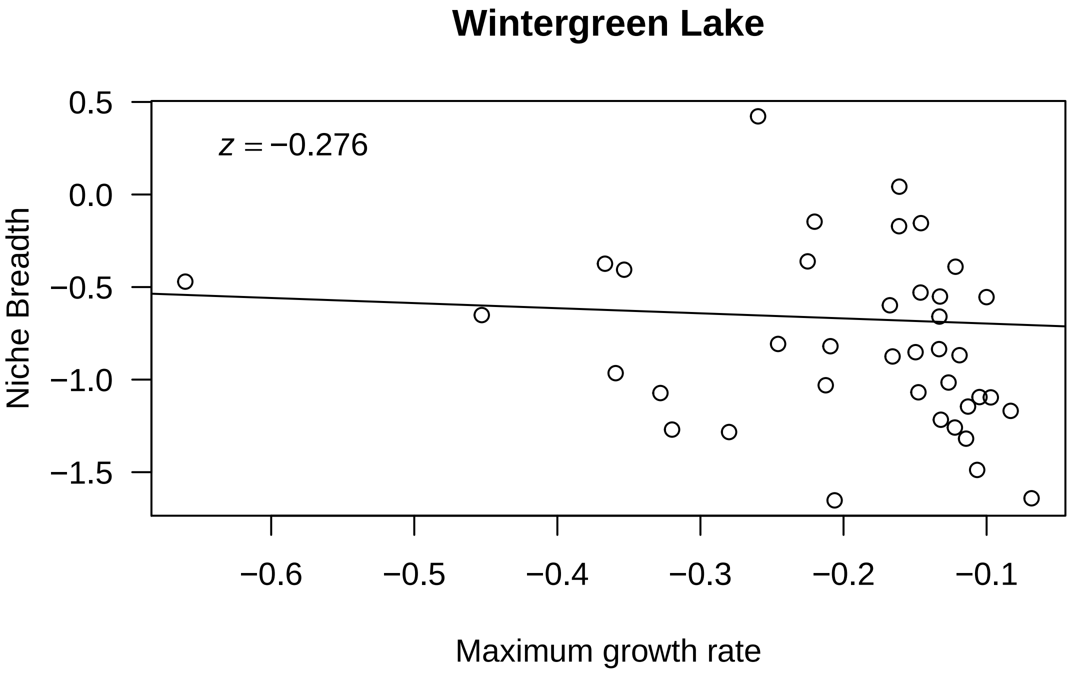
a. Why do we need to correct for shared evolutionary history?
b. How does a phylogenetic regression differ from a standard linear regression?
c. Interpret the slope and fit of each model. Did accounting for shared evolutionary history improve or worsten the fit?
d. Try to come up with a scenario where 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 the variables (species) are not independent to each other due to shared evolutionary history. ***Answer 10b***: Phylogenetic regression describes the variance of residual errors as a covariance matrix that takes in account the branch lengths of the phylogenetic tree (distance of relatedness). While the standard linear regression, the residual errors of the variables are assumed to be independent and identically distributed random variables that follow a normal distribution. ***Answer 10c***: Incorporating the shared evolutionary history improved the fit by a z-score of ~0.02. ***Answer 10d***: Perhaps if we were looking at the relationship between body mass of many animals (mammals, lizards, and birds) and latitude.

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.





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***: There are some evidence to support both the hypthesis that phosphorus use trait is phylogenetically clustered and overdispersed. I would say that based on the phylogeny tests that phosphorus traits are more overdispersed than clustered and that there isn't a strong generalists-specialists tradeoff based off of the niche breadth and maximum growth rate linear regression.