Phylogenetic Diversity - Traits

Matt Gibson; 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).

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

```
## [1] "C:/Users/matth/Documents/bin/QB2017_Gibson/Week6-PhyloTraits"
```

```
setwd("c:/Users/matth/Documents/bin/QB2017_Gibson/Week6-PhyloTraits")
package.list <- c("ape", "seqinr", "phylobase", "adephylo", "geiger", "picante", "stats", "RColorBrewer
                  "pmc", "ggplot2", "tidyr", "dplyr", "phangorn", "pander")

library(alr4)
```

```
## Warning: package 'alr4' was built under R version 3.2.5

## Loading required package: car

## Warning: package 'car' was built under R version 3.2.5

## Loading required package: effects

## Warning: package 'effects' was built under R version 3.2.5

##
## Attaching package: 'effects'

## The following object is masked from 'package:car':
##
##     Prestige
```

```
for (package in package.list){
  if (!require(package, character.only=T, quietly=T)){
    install.packages(package, repos="http://cran.us.r-project.org")
    library(package, character.only=T)
  }
}
```

```
## Warning: package 'ape' was built under R version 3.2.5

## Warning: package 'seqinr' was built under R version 3.2.5

##
## Attaching package: 'seqinr'

## The following objects are masked from 'package:ape':
##
##     as.alignment, consensus

## Warning: package 'phylobase' was built under R version 3.2.5

##
## Attaching package: 'phylobase'

## The following object is masked from 'package:ape':
##
##     edges
## Warning: package 'adephylo' was built under R version 3.2.5
```

```
## Warning: package 'ade4' was built under R version 3.2.5
```

```
## Warning in FUN(X[[i]], ...): failed to assign RegisteredNativeSymbol for
## twins to twins since twins is already defined in the 'cluster' namespace
```

```
## 
## Attaching package: 'adephylo'
```

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

```
## Warning: package 'geiger' was built under R version 3.2.5
```

```
## Warning: package 'picante' was built under R version 3.2.5
```

```
## Warning: package 'vegan' was built under R version 3.2.5
```

```
## Warning: package 'permute' was built under R version 3.2.5
```

```
## 
## 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
```

```
## Warning: package 'nlme' was built under R version 3.2.4
```

```
## 
## Attaching package: 'nlme'
```

```
## The following object is masked from 'package:seqinr':
## 
##     gls
```

```
## Warning: package 'caper' was built under R version 3.2.5
## Warning: package 'mvtnorm' was built under R version 3.2.5

## Warning: package 'phylolm' was built under R version 3.2.5

## Warning: package 'pmc' was built under R version 3.2.5

## Warning: package 'ggplot2' was built under R version 3.2.5

## Warning: package 'tidyr' was built under R version 3.2.5

## Warning: package 'dplyr' was built under R version 3.2.5

##
## 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 object is masked from 'package:car':
##
##     recode

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## Warning: package 'phangorn' was built under R version 3.2.5

##
## Attaching package: 'phangorn'

## The following objects are masked from 'package:vegan':
##
##     diversity, treedist

## Warning: package 'pander' was built under R version 3.2.5
```

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.

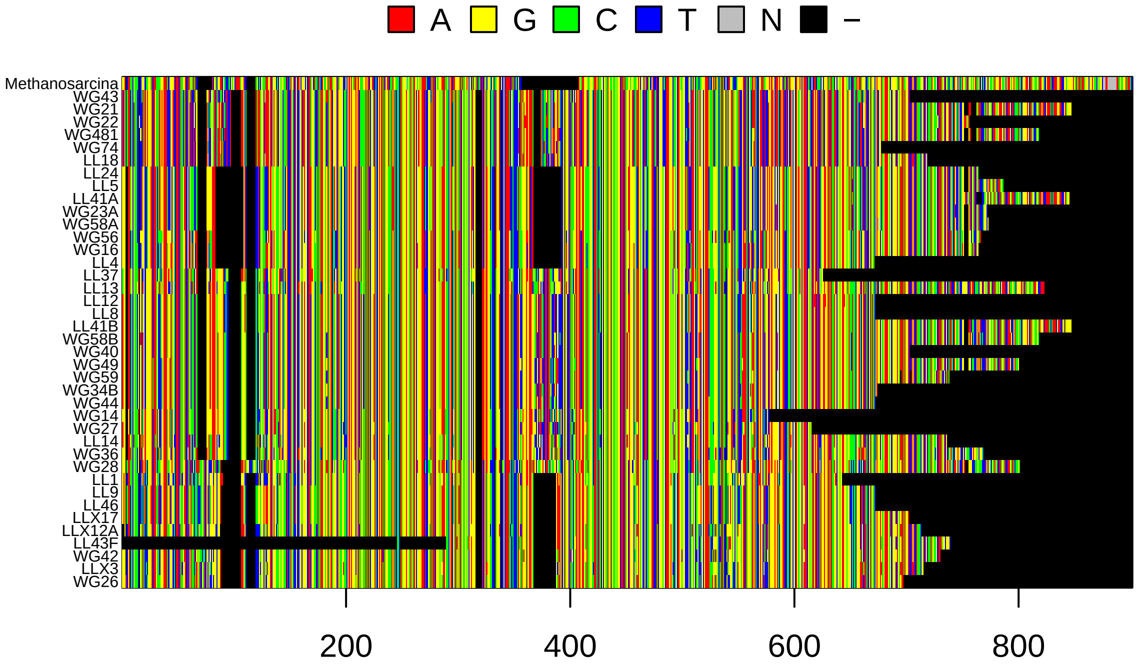
3) SEQUENCE ALIGNMENT

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 contains sites with the `-` character to indicate insertions. The `p.isolates.fasta` file does not have this information. The fasta file just contains continuous sequences as they were read by the sequencer (i.e. it is blind to the location of insertions or deletions). The `-` characters are added by aligning the sequences against a reference.

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
read.aln <- read.alignment(file = "./data/p.isolates.afa", format = "fasta")
p.DNAbin <- as.DNAbin(read.aln)
window <- p.DNAbin[, 100:1000]
image.DNAbin(window, cex.lab = 0.5)
```



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***: ~700 bp ***Answer 2b***: The positions that are polymorhpic.

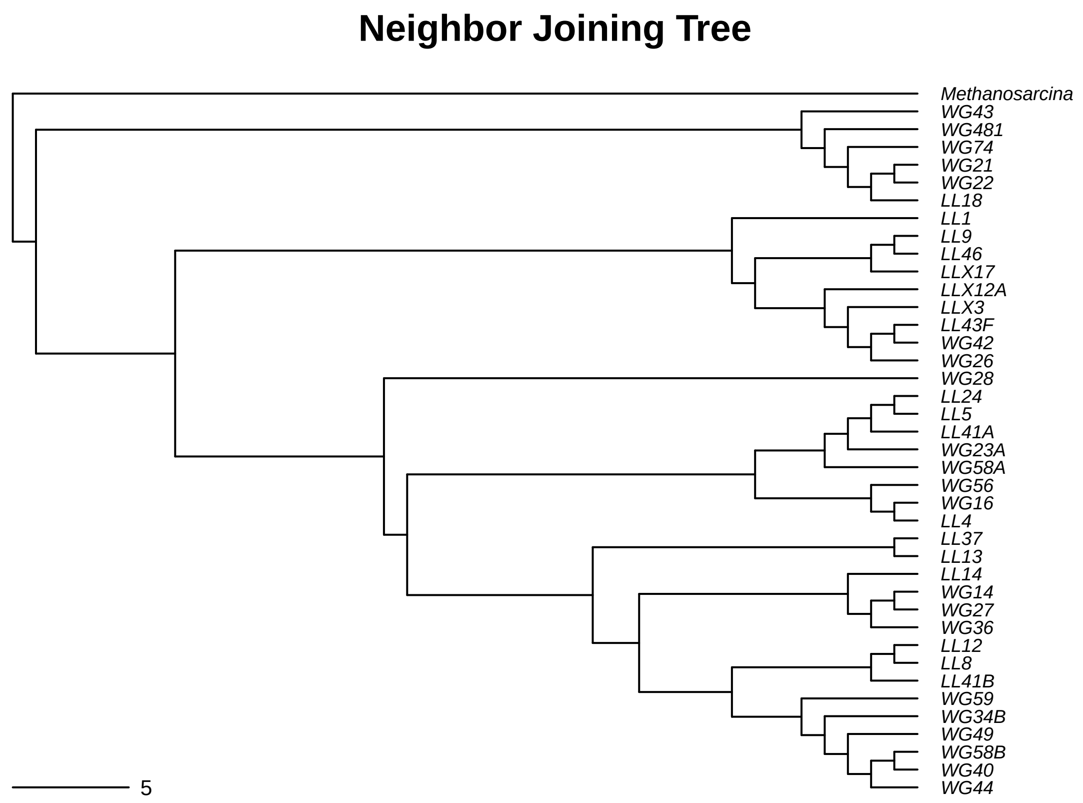
4) MAKING A PHYLOGENETIC TREE

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
seq.dist.raw <- dist.dna(p.DNAbin, model = "raw", pairwise.deletion=F)

nj.tree <- bionj(seq.dist.raw)

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

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

par(mar = c(1,1,2,1), 0.1)
plot.phylo(nj.rooted, main = "Neighbor Joining Tree", "phylogram",
           use.edge.length = F, direction = "right", cex = 0.6,
           label.offset = 1)

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



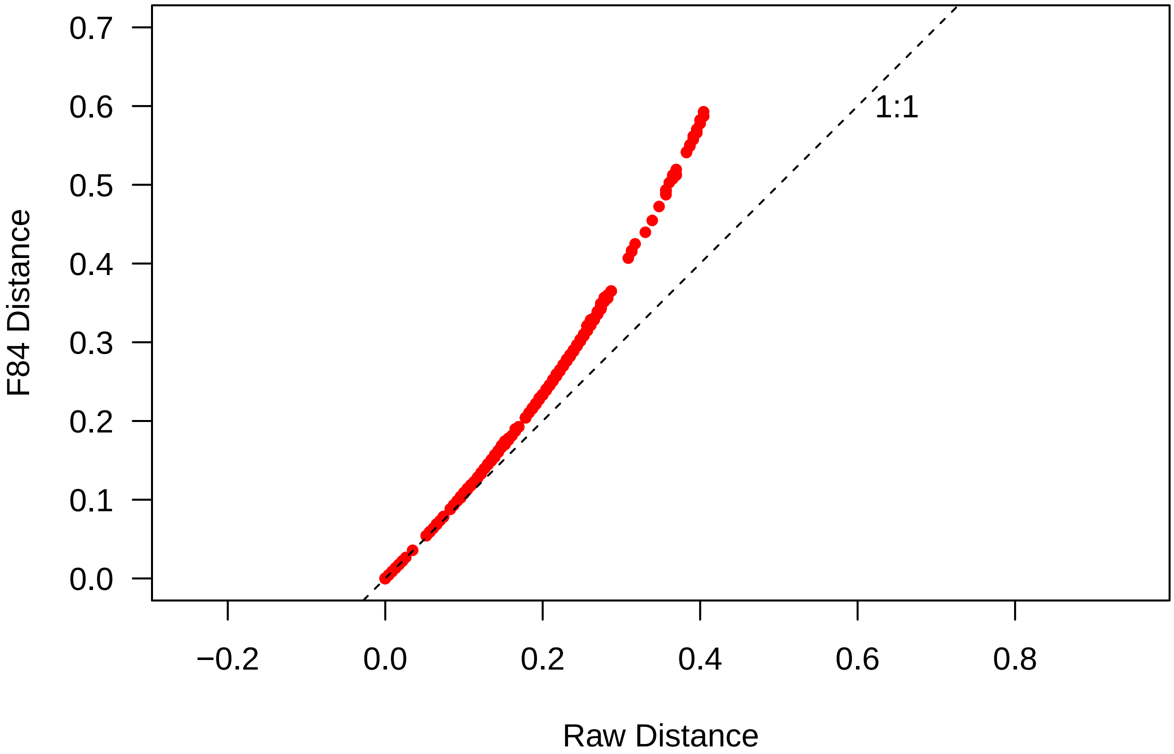
Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

> ***Answer 3***: Neighbor joining trees simply find the most similar taxa and groups them to create a node. There is no inference based on maximum liklihood, bayesian statistics, or bootstrapping. It is great for getting a quick idea of what the phylogeny might be, but it lacks any measures of support for the nodes.

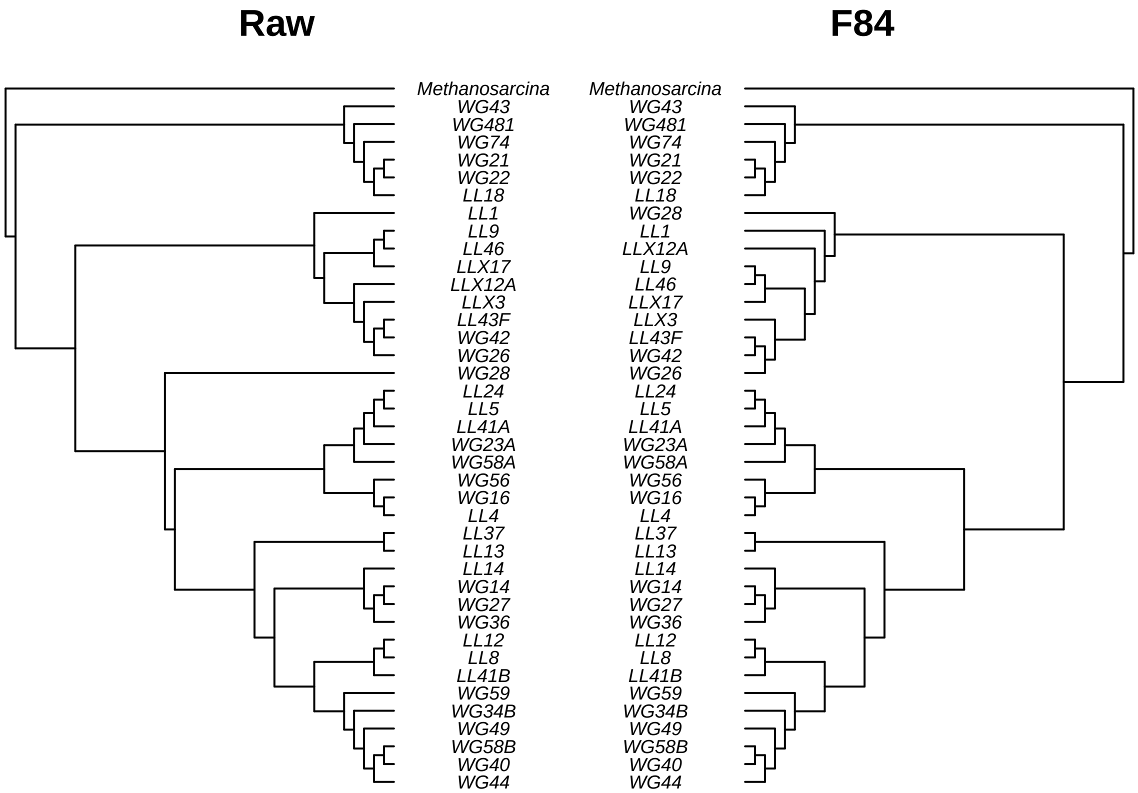
B) SUBSTITUTION MODELS OF DNA EVOLUTION

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

```
seq.dist.F84 <- dist.dna(p.DNAbin, model = "F84", pairwise.deletion = F)

par(mar = c(5,5,2,1), 0.1)
plot(seq.dist.raw, seq.dist.F84,
     pch = 20, col="red", las = 1, asp = 1, xlim = c(0,0.7), ylim = c(0,0.7),
     xlab = "Raw Distance", ylab="F84 Distance")
abline(b = 1, a = 0, lty = 2)
text(0.65, 0.6, "1:1")
```



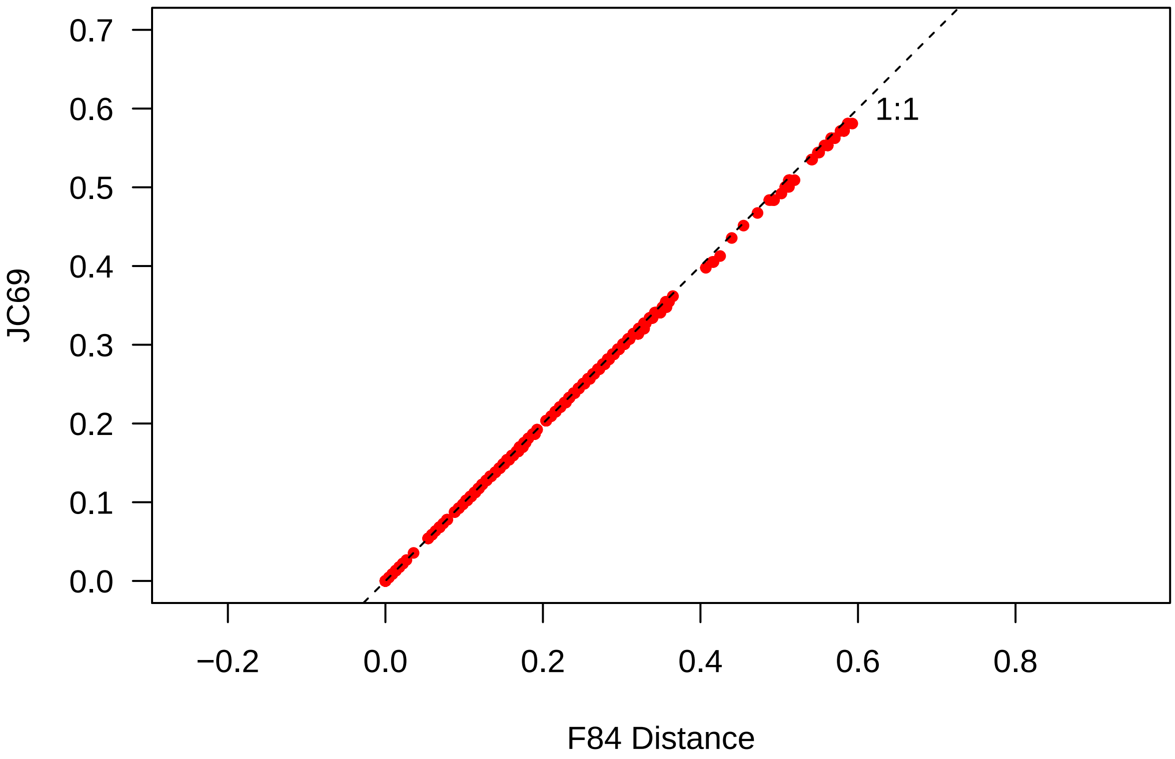
```
raw.tree <- bionj(seq.dist.raw)
F84.tree <- bionj(seq.dist.F84)

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

raw.rooted <- root(raw.tree, raw.outgroup, resolve.root = T)
F84.rooted <- root(F84.tree, F84.outgroup, resolve.root=T)

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=T,
           use.edge.length = F, 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=T,
           use.edge.length = F, adj = 0.5, cex = 0.6, label.offset = 2, main = "F84")
```



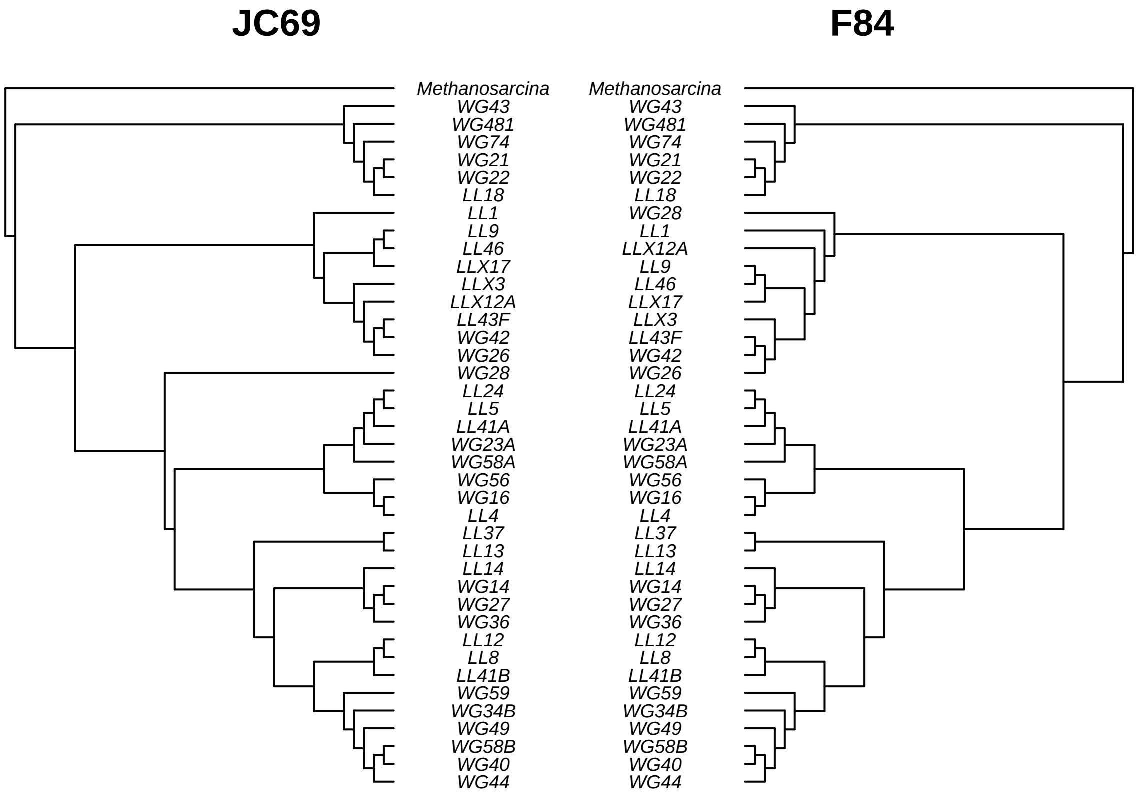
In the R code chunk below, do the following:

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.

```r
seq.dist.K80 <- dist.dna(p.DNAbin, model = "JC69", pairwise.deletion = F)

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(0,0.7)
     xlab = "F84 Distance", ylab="JC69")
abline(b = 1, a = 0, lty = 2)
text(0.65, 0.6, "1:1")
```



```r
#IS actually JC69...not K80
K80.tree <- bionj(seq.dist.K80)
F84.tree <- bionj(seq.dist.F84)

K80.outgroup <- match("Methanosarcina", K80.tree$tip.label)
K80.rooted <- root(K80.tree, K80.outgroup, resolve.root=T)


layout(matrix(c(1,2), 1, 2), width = c(1,1))
par(mar = c(1,1,2,0))
plot.phylo(K80.rooted, type = "phylogram", direction = "right", show.tip.label=T,
           use.edge.length = F, adj = 0.5, cex = 0.6, label.offset = 2, main = "JC69")

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



Question 4:

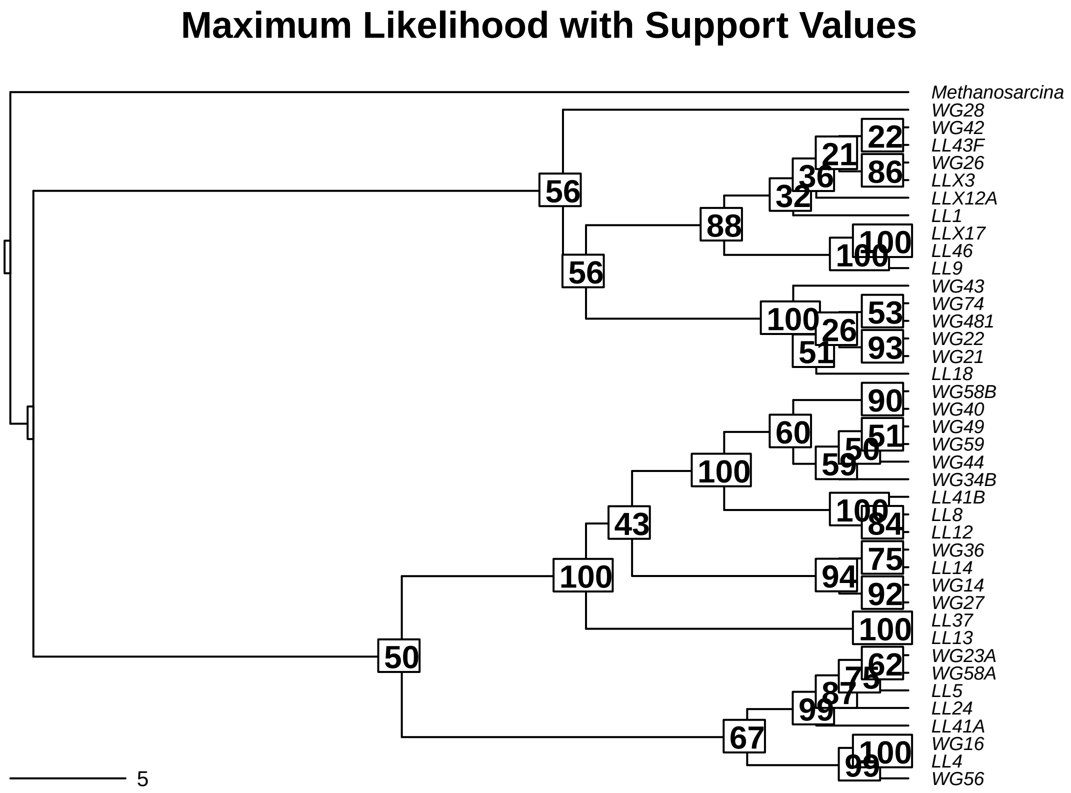
a. Describe the substitution model that you chose. What assumptions does it make and how does it compare to the F84 model?
b. Using the saturation plot and cophylogenetic plots from above, describe how your choice of substitution model affects your phylogenetic reconstruction. If the plots are inconsistent 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 JC69 model. It assumes that all nucleotides occur at equal frequencies and that all mutation types occur with equal probability. The F84 model, however, takes into account the different probabilities of transitions and transversions and allows for differences in nucleotide frequencies. The F84 model is likely more accurate! ***Answer 4b***: Based on the saturation plot, it appears that F84 distance may tend to be greater than the JC69 distance, but this is a very slight difference. The models are very similar. The topologies are different based on the cophylogenetic trees. Interestingly, the JC69 tree looks very similar to the raw tree plotted above. The JC69 and F84 trees differ because we are using two different models of nucleotide evolution. One that assumes equal frequency of all nucleotides and equal probability of any mutation type and one that allows nucleotides to occur at different frequencies and mutations to be biased. ***Answer 4c***: Based on the saturation plot, I would think that the `F84` model's assumption of different transition and transversion rates is more accurate. Also because we know that there is actually a bias for transition mutations.

C) ANALYZING A MAXIMUM LIKELIHOOD TREE

In the R code chunk below, do the following:
1. Read in the maximum likelihood phylogenetic tree used in the handout. 2. Plot bootstrap support values onto the tree

```
ml.bootstrap <- read.tree("./data/ml_tree/RAxML_bipartitions.T1")
par(mar = c(1,1,2,1) + 0.1)
plot.phylo(ml.bootstrap, type = "phylogram", direction = "right", show.tip.label = T, use.edge.length =
           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")
```



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 topologies are different but the major groupings are roughly the same (some clades are in flipped orientations, though). There are more deep bifurcations in the neighbor joining tree. The differences likely arise since the NJ tree does not take into account nucleotide states as in a ML tree and is not a statistical estimation of the most probable tree. We don't know the "true distance" between each taxa, so we must resort to maximization procedures such as ML. ***Answer 5b***: So that we can estimate our confidence in the branches of our tree. We generate a sample of trees from sampling data randomly with replacement from our original data. ***Answer 5c***: The values tell us how often we see that branch in our sample of bootstrapped trees. Each time a new bootstrapped tree is generated, it is comapred to the original. If it shares a particular branch, the branch is given a score of 1. If it doesn't, it gets a score of 0. This is repeated over hundreds to thousands of iterations. ***Answer 5d***: Three of the deep interior branches have low support (56, 56, and 50) as well as several exterior branches. ***Answer 5e***: We should not trust these branches as they have bootstrap values below 95%. Though I think it would depend on our a priori expectations. If these are all very closely related individuals and we are trying to visualize a phylogeny, we would probably expect branch support to be low overall.

5) INTEGRATING TRAITS AND PHYLOGENY

A. Loading Trait Database

In the R code chunk below, do the following:
1. import the raw phosphorus growth data, and
2. standardize the data for each strain by the sum of growth rates.

```
p.growth <- read.table("./data/p.isolates.raw.growth.txt", sep = "\t", header = T, row.names = 1)
p.growth.std <- p.growth / (apply(p.growth, 1, sum))
```

B. Trait Manipulations

In the R code chunk below, do the following:
1. calculate the maximum growth rate (μ_{max}) of each isolate across all phosphorus types,
2. create a function that calculates niche breadth (*nb*), and
3. use this function to calculate *nb* for each isolate.

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

```
##        LL1       LL12       LL13       LL14       LL18       LL24 
## 0.25206654 0.67409943 0.70041864 0.22320936 0.15150357 0.08539280 
##       LL37        LL4      LL41A      LL41B      LL43F       LL46 
## 0.29531665 0.09322218 0.02226927 0.33829852 0.06070650 0.08461478 
##        LL5        LL8        LL9     LLX12A      LLX17       LLX3 
## 0.05210208 0.40725197 0.05370428 0.02285066 0.10837476 0.08015598 
##       WG14       WG16       WG21       WG22      WG23A       WG26 
## 0.15585259 0.14610988 0.08046866 0.13337010 0.07151097 0.03252064 
##       WG27       WG28      WG34B       WG36       WG40       WG42 
## 0.21906230 0.05509375 0.71314834 0.27902816 0.42281774 0.06779520 
##       WG43       WG44      WG481       WG49       WG56      WG58A 
## 0.13545974 0.04797530 0.14064695 2.64416686 0.28117640 0.39246096 
##      WG58B       WG59       WG74 
## 0.43516869 1.10251743 0.09642441
```

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

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

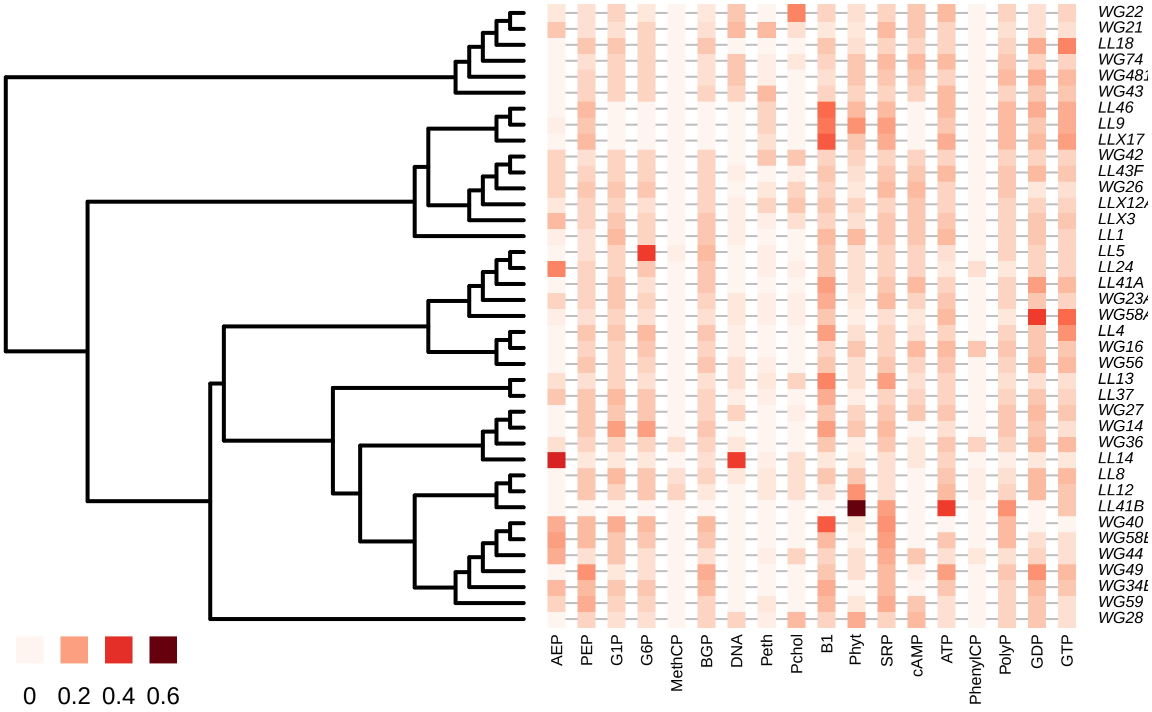
C. Visualizing Traits on Trees

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.

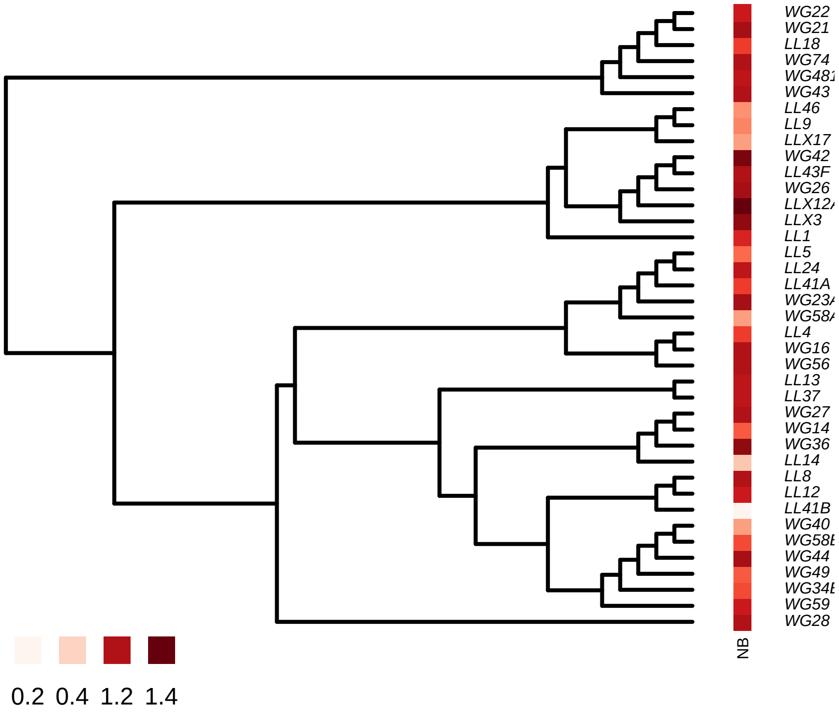
```
seq.dist.JC69 <- dist.dna(p.DNAbin, model = "JC69", pairwise.deletion = F)
JC69.tree <- bionj(seq.dist.JC69)

JC69.outgroup <- match("Methanosarcina", JC69.tree$tip.label)
JC69.rooted <- root(JC69.tree, JC69.outgroup, resolve.root=T)
JC69.rooted <- drop.tip(JC69.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).

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

par(mar = c(1,1,1,1) + 0.1)
x <- phylo4d(JC69.rooted, p.growth.std)
table.phylo4d(x, treetype = "phylo", symbol = "colors", show.node = T,
              cex.label = 0.5, scale = F, use.edge.length = F,
              edge.color = "black", edge.width = 2, box = F,
              col=mypalette(25), pch=15, cex.symbol = 1.25,
              ratio.tree = 0.5, cex.legend = 1.5, center = F)
```



```
par(mar = c(1,5,1,5) + 0.1)
x.nb <- phylo4d(JC69.rooted, nb)
table.phylo4d(x.nb, treetype = "phylo", symbol = "colors",show.node = T,
              cex.label = 0.5, scale = F, use.edge.length =F, edge.color = "black",
              edge.width = 2, box = F, col = mypalette(25), pch = 15, cex.symbol = 1.25, var.label=("
              ratio.tree = 0.9, cex.legend = 1.5, center = F)
```



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?

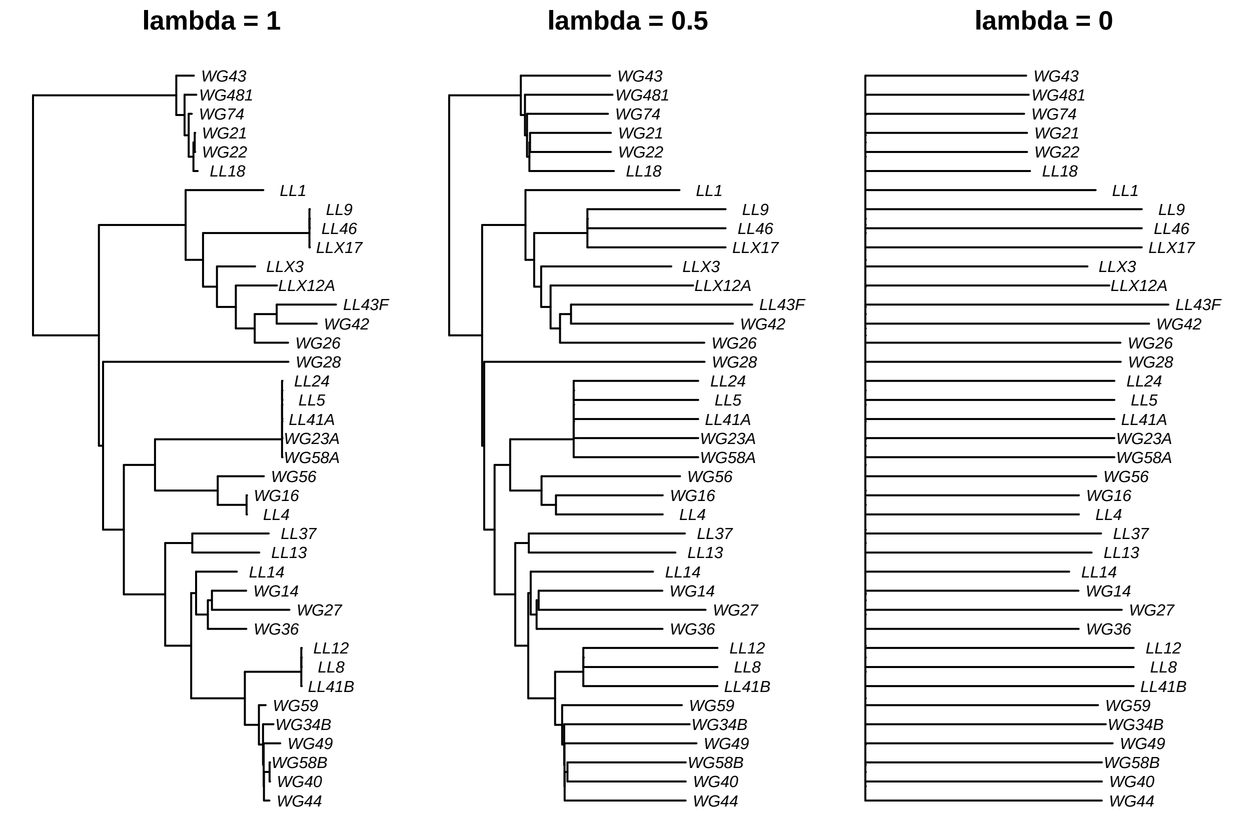
Answer 6a: Being a generalist (as quantified using Levin's niche breadth) leads to lower maximum growth rate and being a specialist, higher. ***Answer 6b***: A generalist species should grow moderately well on many or all substrates and a specialist should grow very well on few substrates and poor on the rest. This is generally what we see in our results. The taxa WG49 (with the highest max growth rate and smallest niche breadth) grows well on really only one substrate (Phyt). WG26 however, has a low max growth rate, higher niche breadth, and can grow moderately well on many more substrates.

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.

```
nj.lambda.5 <- rescale(JC69.rooted, "lambda", 0.5)
nj.lambda.0 <- rescale(JC69.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(JC69.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.

```
fitContinuous(JC69.rooted, nb, model = "lambda")
```

```
## GEIGER-fitted comparative model of continuous data
##  fitted 'lambda' model parameters:
##  lambda = 0.046243
##  sigsq = 0.107359
##  z0 = 0.665129
##
##  model summary:
##  log-likelihood = 21.745015
##  AIC = -37.490031
##  AICc = -36.804316
##  free parameters = 3
```

```
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 22
##  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 = 0.106940
##  z0 = 0.658124
##
##  model summary:
##  log-likelihood = 21.706952
##  AIC = -37.413904
##  AICc = -36.728190
##  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 value in the untransformed tree is higher (0.0462) than in the transformed tree (0.0) ***Answer 7b***: The AIC of the untransformed tree (-37.49) is slightly lower than that for the transformed tree (-37.41). Although, the models must be considered equivalent since the difference in AIC is not greater than 2. ***Answer 7c***: This result suggests there is NO phylogenetic signal for niche breadth. The transformed tree removed all existing phylogenetic signal.... So when we tested the two models and they came out equal, that suggests there is no phylogenetic signal for niche breadth.

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.

```
JC69.rooted$edge.length <- JC69.rooted$edge.length + 10^-7

p.phylosignal <- matrix(NA, 6, 18)
colnames(p.phylosignal) <- colnames(p.growth.std)
rownames(p.phylosignal) <- c("K", "PIC.var.obs", "PIC.var.mean",
                             "PIC.var.P", "PIC.var.z", "PIC.P.BH")

for (i in 1:18){
  x <- as.matrix(p.growth.std[ ,i, drop = FALSE])
  out <- phylosignal(x, JC69.rooted)
  p.phylosignal[1:5, i] <- round(t(out), 3)

}


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

p.phylosignal
```

```
##                   AEP      PEP      G1P      G6P  MethCP      BGP      DNA
## K               0.000    0.000    0.000    0.000   0.000    0.000    0.000
## PIC.var.obs  4373.157  664.095  948.941 5924.730 350.894  536.104  259.084
## PIC.var.mean 8148.844 1483.115 1878.479 3558.768 495.384 1764.417 5239.024
## PIC.var.P       0.259    0.080    0.114    0.771   0.386    0.023    0.001
## PIC.var.z      -0.816   -1.266   -1.186    0.948  -0.412   -1.702   -1.314
## PIC.P.BH        0.632    0.360    0.410    0.808   0.632    0.138    0.018
##                  Peth    Pchol       B1     Phyt      SRP     cAMP
## K               0.000    0.000    0.000    0.000    0.000    0.000
## PIC.var.obs  1446.463 2368.390 3517.018 9240.368 1307.025  690.723
## PIC.var.mean 1851.854 3298.098 5342.411 9025.559 1600.541 3000.653
## PIC.var.P       0.328    0.382    0.229    0.585    0.322    0.006
## PIC.var.z      -0.501   -0.541   -0.801    0.027   -0.528   -2.472
## PIC.P.BH        0.632    0.632    0.632    0.738    0.632    0.054
##                   ATP PhenylCP    PolyP      GDP      GTP
## K               0.000    0.000    0.000    0.000    0.000
## PIC.var.obs  4040.137 1224.017 1126.345 4473.878 2721.766
## PIC.var.mean 3055.181  764.426 1231.414 3627.266 2936.937
## PIC.var.P       0.630    0.808    0.490    0.656    0.491
## PIC.var.z       0.443    0.977   -0.186    0.386   -0.153
## PIC.P.BH        0.738    0.808    0.680    0.738    0.680
```

```
signal.nb <- phylosignal(nb, JC69.rooted)
signal.nb
```

```
##               K PIC.variance.obs PIC.variance.rnd.mean PIC.variance.P
## 1 3.411681e-06         49966.78              49720.95          0.558
##   PIC.variance.Z
## 1      0.0122452
```

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 no significant phylogenetic signal for niche breadth (alpha = 0.05). There is a significant phylogenetic signal for two phosphorous resources (DNA and cAMP). ***Answer 8b***: The results are suggestive of overdispersion (K < 1) for all resources.

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.

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

apply(p.growth.pa, 2, sum)
```

```
##      AEP      PEP      G1P      G6P   MethCP      BGP      DNA     Peth
##       20       38       35       34        3       35       19       21
##    Pchol       B1     Phyt      SRP     cAMP      ATP PhenylCP    PolyP
##       18       38       36       39       29       38        6       39
##      GDP      GTP
##       37       38
```

```r
p.growth.pa$name <- rownames(p.growth.pa)

p.traits <- comparative.data(JC69.rooted, p.growth.pa, "name")
print(d1 <- 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 :  JC69.rooted
##   Number of permutations :  1000
##
## Estimated D :  -0.4687683
## Probability of E(D) resulting from no (random) phylogenetic structure :  0
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.762
```

```
d1$Pval1
```

```
## [1] 0
```

```
d1$Pval0
```

```
## [1] 0.762
```

```
print(d2 <- 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 :  JC69.rooted
##   Number of permutations :  1000
##
## Estimated D :  0.5280636
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.008
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.01
```

```
d2$Pval1
```

```
## [1] 0.008
```

```
d2$Pval0
```

```
## [1] 0.01
```

```
print(d3 <- 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 :  JC69.rooted
##   Number of permutations :  1000
##
## Estimated D :  0.1194971
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.002
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.349
```

```
d3$Pval1
```

```
## [1] 0.002
```

```
d3$Pval0
```

```
## [1] 0.349
```

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.

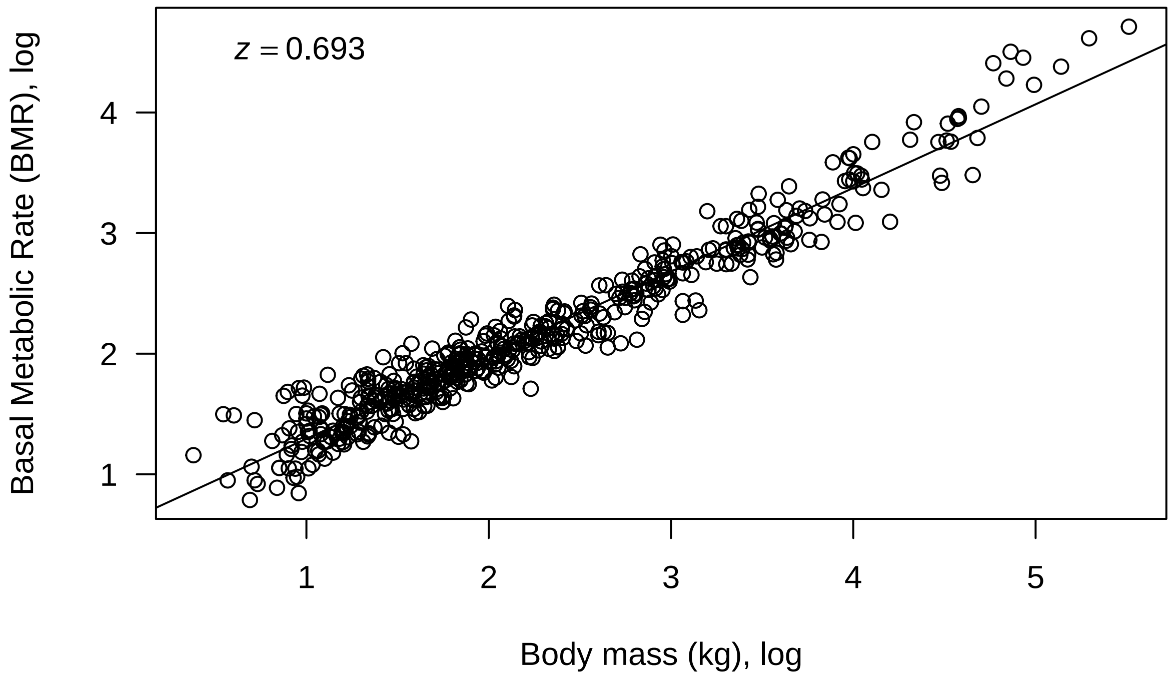
> ***Answer 9a***: BGP had a negative D (-.45; clustered) but this is not significantly different from 0 so we can't rule out whether this is due to random clumping. DNA had a positive D (0.5259; overdispersed) and this value is significantly different from 0 and 1. cAMP had a positive D (0.12107; overdispersed). This value is significantly different from 1, ruling out Brownian dispersal of traits but is not significantly different from 0, so this trait may be randomly clumped into a phylogenetic structure. ***Answer 9b***: Bloomberg's K revealed that growth on two substrates (DNA, and cAMP) were overdispersed. Based on these analyses, we can only conclude that growth on DNA is overdispersed. ***Answer 9c***: The two methods attempt to address the same things but do so in different ways. Bloomberg's K is calculated by comparing the MSE of the trait data and the MSE of the variance-covariance matrix derived from the phylogeny under the assumption of Brownian motion. The estimate of D, however, is a formal test of ecological dispersion used to measure how traits or species are spatially dispersed. It is being applied to phylogenetics. One thing to note is that in order to calculate an estimate of D, we had to convert our continuous data to categorical and we may have lost some power to detect dispersion by doing so. tl;dr The two estimates use fairly different methods to calculate dispersion and to estimate D we had to convert continuous data to categorical.

7) PHYLOGENETIC REGRESSION

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

```
mammal.Tree <- read.tree("./data/mammal_best_super_tree_fritz2009.tre")
mammal.data <- read.table("./data/mammal_BMR.txt", sep = "\t", header=T)

mammal.data <- mammal.data[, c("Species", "BMR_.mlO2.hour.","Body_mass_for_BMR_.gr.")]
mammal.species <- array(mammal.data$Species)

pruned.mammal.tree <- drop.tip(mammal.Tree, mammal.Tree$tip.label[-na.omit(match(mammal.species,
                                                                                  mammal.Tree$tip.label))
```

```
pruned.mammal.data <- mammal.data[mammal.data$Species %in% pruned.mammal.tree$tip.label,]

rownames(pruned.mammal.data) <- pruned.mammal.data$Species

fit <- lm(log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.),
          data = pruned.mammal.data)

plot(log10(pruned.mammal.data$Body_mass_for_BMR_.gr.), log10(pruned.mammal.data$BMR_.mlO2.hour.),
     las = 1, 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))

text(0.5, 4.5, eqn, pos = 4)
```

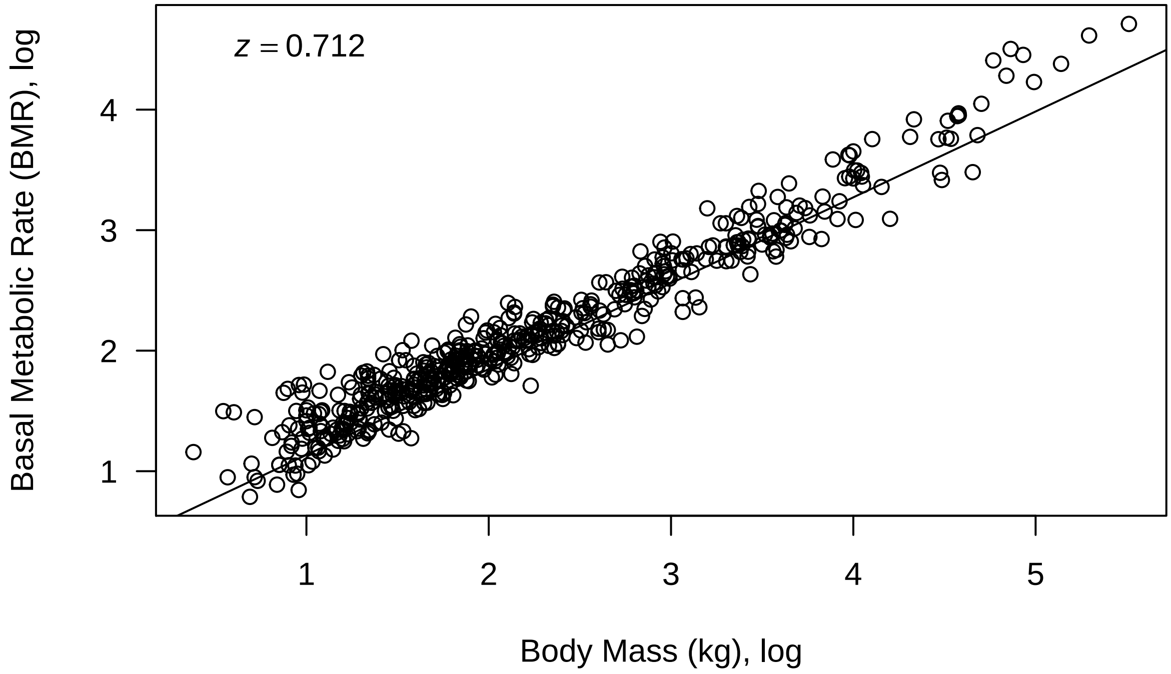


```
summary(fit)
```

```
##
## 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
```

```
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)
```



```
summary(fit.phy)
```

```
##
```

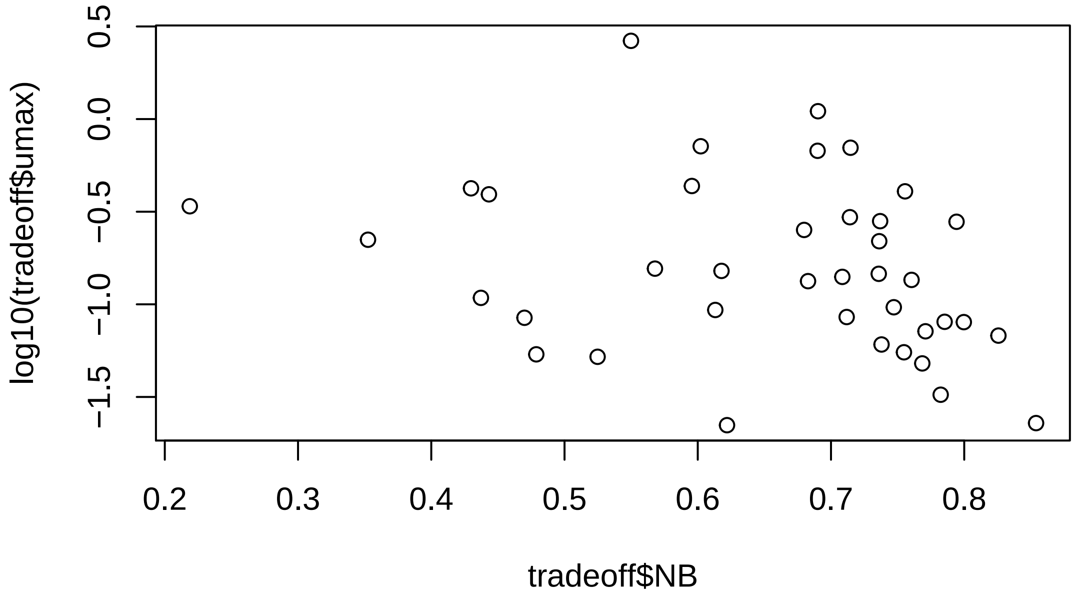
```
## Call:
## phylolm(formula = log10(BMR_.mlO2.hour.) ~ log10(Body_mass_for_BMR_.gr.),
##     data = pruned.mammal.data, phy = pruned.mammal.tree, model = "lambda",
##     boot = 0)
##
##    AIC logLik
## -646.9  327.5
##
## Raw residuals:
##      Min       1Q   Median       3Q      Max
## -0.32221  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.
```

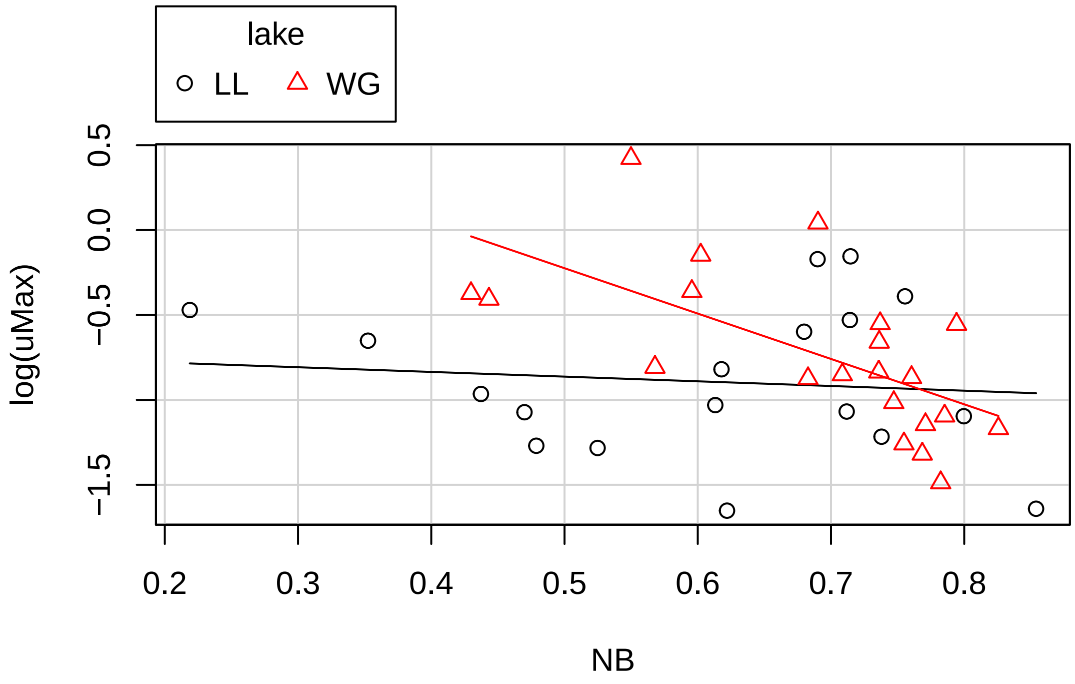
a. Why do we need to correct for shared evolutionary history?
b. How does a phylogenetic regression differ from a standard linear regression?
c. Interpret the slope and fit of each model. Did accounting for shared evolutionary history improve or worsten the fit?
d. Try to come up with a scenario where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

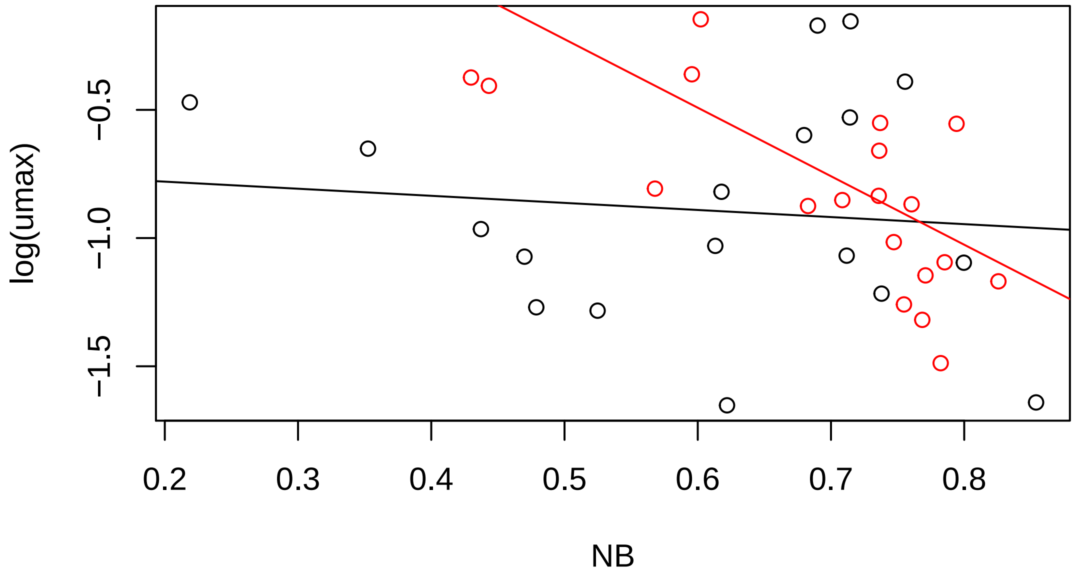
> ***Answer 10a***: Because observations of a trait value in multiple taxa may not be independent if they evolved in a common ancestor! ***Answer 10b***: In a phylogenetic regression, instead of assuming that resisduals are iid. normal, we describe them by a covariance matrix taking into account the underlying phylogeny. ***Answer 10c***: Accounting for shared evolutionary history improved the fit of the model. We got a higher slope (a stronger relationship). ***Answer 10d***: Flower size vs. latitude in plants. There may exist a pattern of smaller flower size as you move further north. This relationship may be explained mostly by phylogenetic relatedness and evolution of smaller or larger flower sizes in common ancestors.

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.


log10(tradeoff$umax)
0.5
0.0
−0.5
−1.0
−1.5
0.2
0.3
0.4
0.5
0.6
0.7
0.8
tradeoff$NB


lake
○ LL
△ WG
log(uMax)
0.5
0.0
−0.5
−1.5
0.2
0.3
0.4
0.5
0.6
0.7
0.8
NB



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***: When there are more available phosphorous sources (WG lake), the relationship between max growth rate and niche breadth is stronger (i.e. there is a tradeoff). When there are fewer available phosphorous sources (as in the LL lake), the weaker the relationship (i.e. there is no tradeoff).