11. Worksheet: Phylogenetic Diversity - Traits

Mark Hibbins; Z620: Quantitative Biodiversity, Indiana University

19 February, 2019

OVERVIEW

Up to this point, we have been focusing on patterns taxonomic diversity in Quantitative Biodiversity. Although taxonomic diversity is an important dimension of biodiversity, it is often necessary to consider the evolutionary history or relatedness of species. The goal of this exercise is to introduce basic concepts of phylogenetic diversity.

After completing this exercise you will be able to:

1. create phylogenetic trees to view evolutionary relationships from sequence data
2. map functional traits onto phylogenetic trees to visualize the distribution of traits with respect to evolutionary history
3. test for phylogenetic signal within trait distributions and trait-based patterns of biodiversity

Directions:

1. In the Markdown version of this document in your cloned repo, change "Student Name" on line 3 (above) with your name.
2. Complete as much of the worksheet as possible during class.
3. Use the handout as a guide; it contains a more complete description of data sets along with examples of proper scripting needed to carry out the exercises.
4. Answer questions in the worksheet. Space for your answers is provided in this document and is indicated by the ">" character. If you need a second paragraph be sure to start the first line with ">". You should notice that the answer is highlighted in green by RStudio (color may vary if you changed the editor theme).
5. Before you leave the classroom today, it is *imperative* that you **push** this file to your GitHub repo, at whatever stage you are. This will enable you to pull your work onto your own computer.
6. When you have completed the worksheet, **Knit** the text and code into a single PDF file by pressing the `Knit` button in the RStudio scripting panel. This will save the PDF output in your '8.BetaDiversity' folder.
7. After Knitting, please submit the worksheet by making a **push** to your GitHub repo and then create a **pull request** via GitHub. Your pull request should include this file (**11.PhyloTraits_Worksheet.Rmd**) with all code blocks filled out and questions answered) and the PDF output of `Knitr` (**11.PhyloTraits_Worksheet.pdf**).

The completed exercise is due on **Wednesday, February 20th, 2019 before 12:00 PM (noon)**.

1) SETUP

Typically, the first thing you will do in either an R script or an RMarkdown file is setup your environment. This includes things such as setting the working directory and loading any packages that you will need.

In the R code chunk below, provide the code to:
1. clear your R environment,
2. print your current working directory,
3. set your working directory to your "*/11.PhyloTraits*" folder, and
4. load all of the required R packages (be sure to install if needed).

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

```
## [1] "/Users/mark/Box Sync/Courses/Quantitative Biodiversity/QB2019_Hibbins/2.Worksheets/11.PhyloTrai
```

```
setwd("/Users/mark/Box Sync/Courses/Quantitative Biodiversity/QB2019_Hibbins/2.Worksheets/11.PhyloTraits

package.list <- c('ape', 'seqinr', 'phylobase', 'adephylo', 'geiger', 'picante', 'stats', 'RColorBrewer
for (package in package.list) {
  if (!require(package, character.only=TRUE, quietly=TRUE)) {
    install.packages(package)
    library(package, character.only=TRUE)
  }
}
```

```
##
## Attaching package: 'seqinr'
```

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

```
##
## Attaching package: 'phylobase'
```

```
## The following object is masked from 'package:ape':
##
##     edges
```

```
##
## Attaching package: 'permute'
```

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

```
## This is vegan 2.5-3
```

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

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

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following object is masked from 'package:nlme':
##
##     collapse
```

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

```
## The following objects are masked from 'package:stats':
```

```
##
##     filter, lag

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

##
## Attaching package: 'phangorn'

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

2) DESCRIPTION OF DATA

The maintenance of biodiversity is thought to be influenced by **trade-offs** among species in certain functional traits. One such trade-off involves the ability of a highly specialized species to perform exceptionally well on a particular resource compared to the performance of a generalist. In this exercise, we will take a phylogenetic approach to mapping phosphorus resource use onto a phylogenetic tree while testing for specialist-generalist trade-offs.

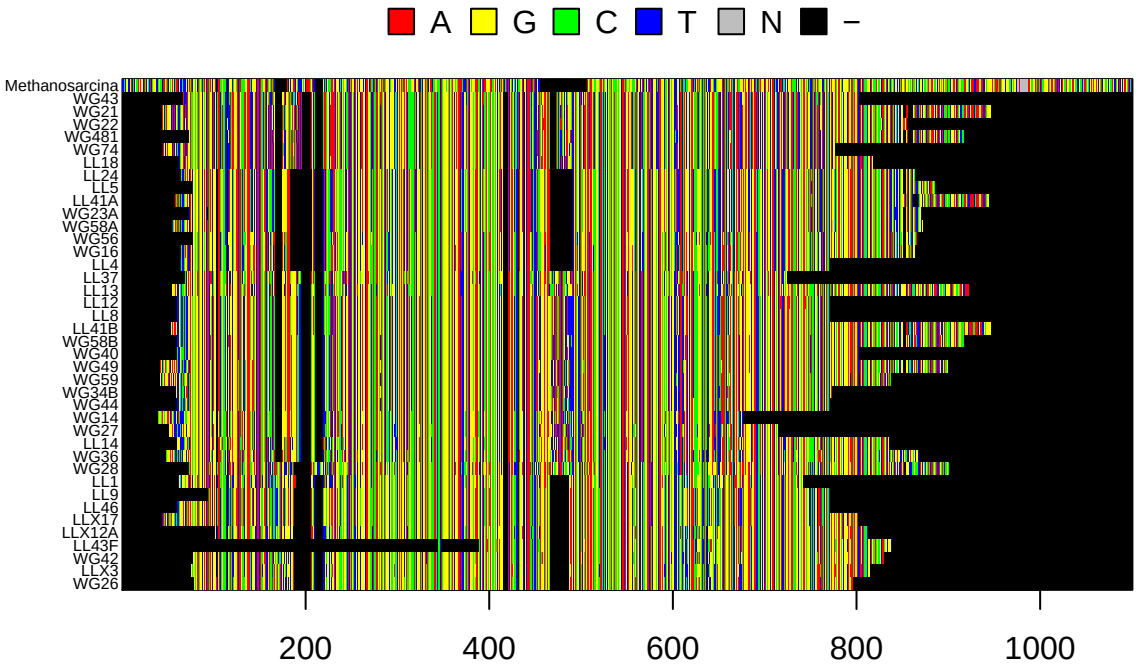
3) SEQUENCE ALIGNMENT

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

> ***Answer 1***: In p.isolates.afa, there are gaps inserted in the sequences, whereas in p.isolates.fasta they are all at the ends. p.isolates.afa is the alignment file, while p.isolates.fasta is the raw fasta file.

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

```
read.aln <- read.alignment(file = './data/p.isolates.afa', format = 'fasta')
p.DNAbin <- as.DNAbin(read.aln)
window <- p.DNAbin[, 0:1100]
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 sequence reads?

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

Answer 2a: The reads are roughly 600-700 base pairs long, although it does vary quite a bit by strain. ***Answer 2b***: In this alignment, there are two regions which seems to be fairly well-aligned (ie. few gaps), but still contain enough variation as to be phylogenetically informative. These are the blocks which are approximately between positions 250 and 450, and between 500 and 700.

4) MAKING A PHYLOGENETIC TREE

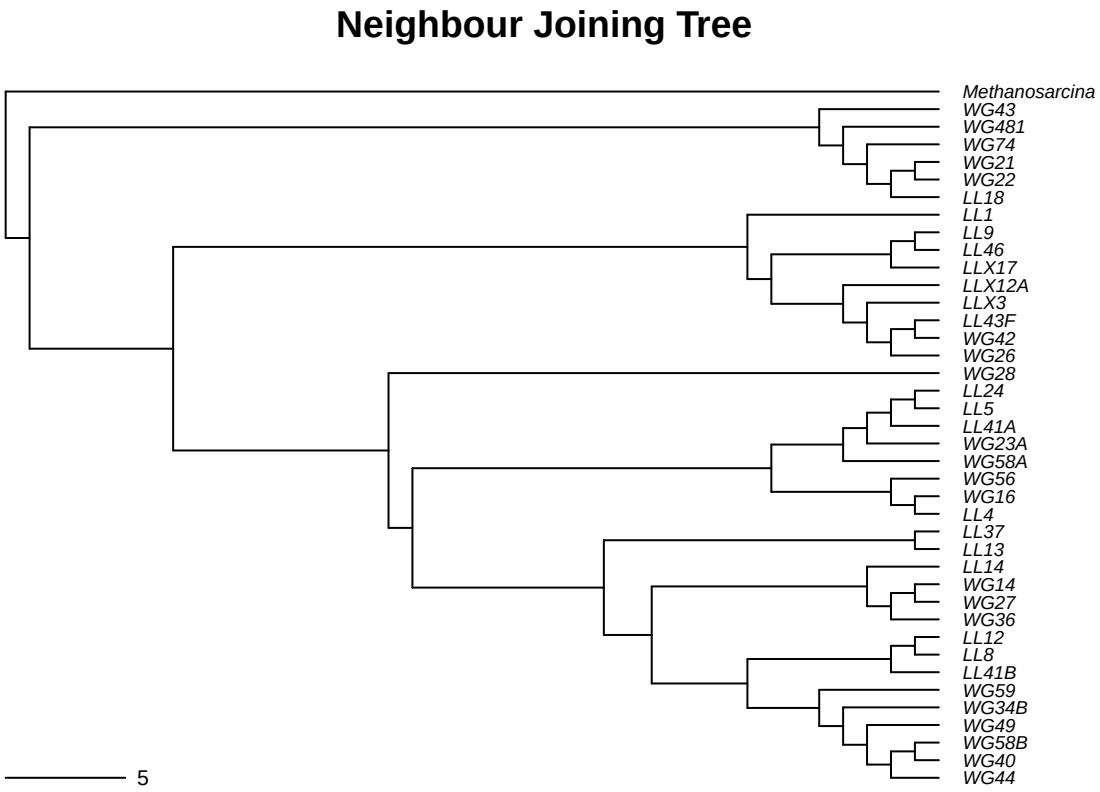
Once you have aligned your sequences, the next step is to construct a phylogenetic tree. Not only is a phylogenetic tree effective for visualizing the evolutionary relationship among taxa, but as you will see later, the information that goes into a phylogenetic tree is needed for downstream analysis.

A. Neighbor Joining Trees

In the R code chunk below, do the following:
1. calculate the distance matrix using `model = "raw"`,
2. create a Neighbor Joining tree based on these distances,
3. define "Methanosarcina" as the outgroup and root the tree, and
4. plot the rooted tree.

```
seq.dist.raw <- dist.dna(p.DNAbin, model = 'raw', pairwise.deletion = FALSE)
nj.tree <- bionj(seq.dist.raw)
outgroup <- match('Methanosarcina', nj.tree$tip.label)
nj.rooted <- root(nj.tree, outgroup, resolve.root = TRUE)

par(mar = c(1,1,2,1) + 0.1)
```

```
plot.phylo(nj.rooted, main = 'Neighbour Joining Tree', 'phylogram',
           use.edge.length = FALSE, direction = 'right',
           cex = 0.6, label.offset = 1)
add.scale.bar(cex = 0.7)
```



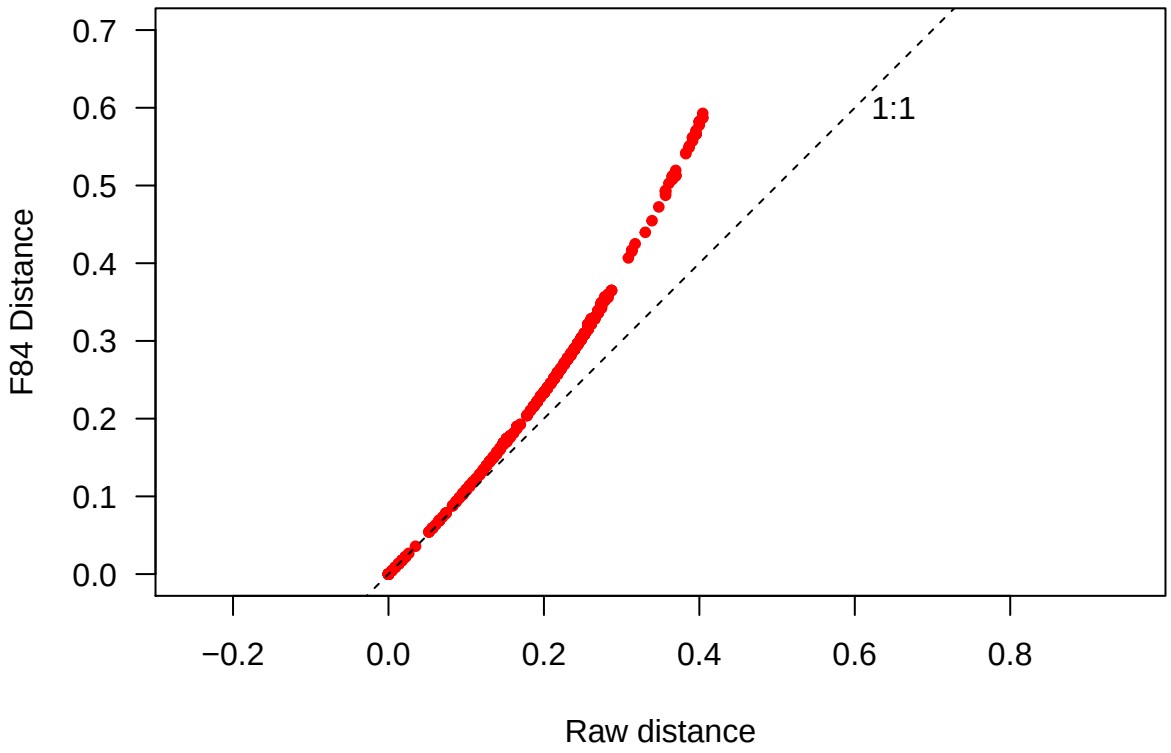
Question 3: What are the advantages and disadvantages of making a neighbor joining tree?

> ***Answer 3***: The disadvantage of neighbour-joining is that it makes a number of oversimplifications about how DNA sequences evolve over time. It assumes that all bases occur with equal frequency, all mutations occur at an equal rate, and all lineages evolve at an equal rate. The main advantage of neighbour-joining is that it is much faster than more complex methods. It can also be used as a guide for constructing phylogenies with more complex methods.

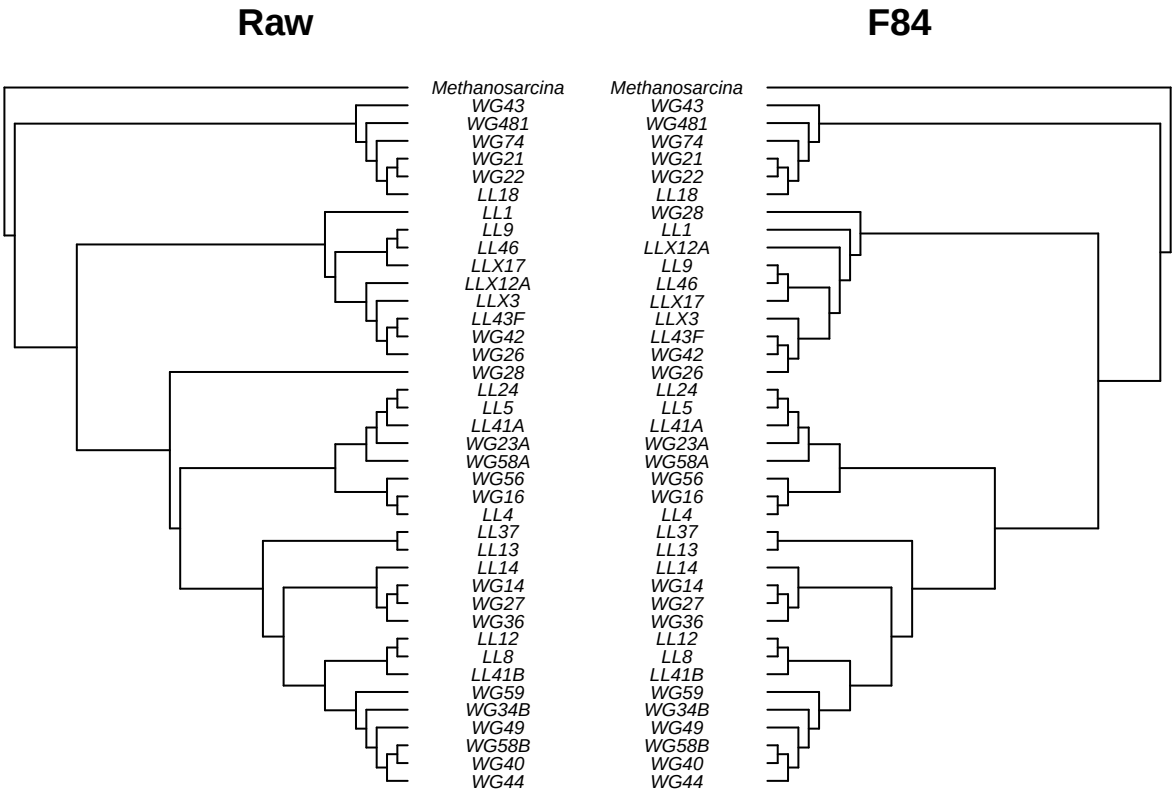
B) SUBSTITUTION MODELS OF DNA EVOLUTION

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 = FALSE)

par(mar = c(5, 5, 2, 1) + 0.1)
plot(seq.dist.raw, seq.dist.F84, pch = 20, col = 'red', las = 1, asp = 1,
     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')
```

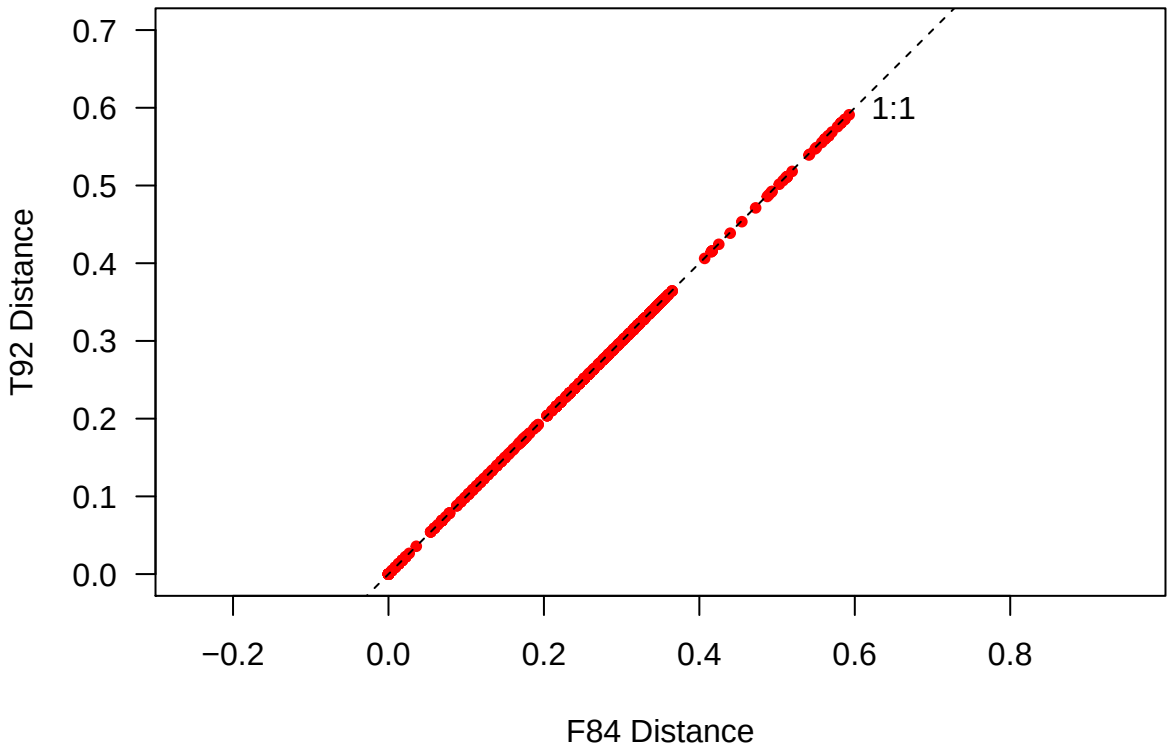



```
raw.tree <- bionj(seq.dist.raw)
F84.tree <- bionj(seq.dist.F84)
raw.outgroup <- match('Methanosarcina', raw.tree$tip.label)
F84.outgroup <- match('Methanosarcina', F84.tree$tip.label)
raw.rooted <- root(raw.tree, raw.outgroup, resolve.root = TRUE)
F84.rooted <- root(F84.tree, F84.outgroup, resolve.root = TRUE)

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')
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')
```

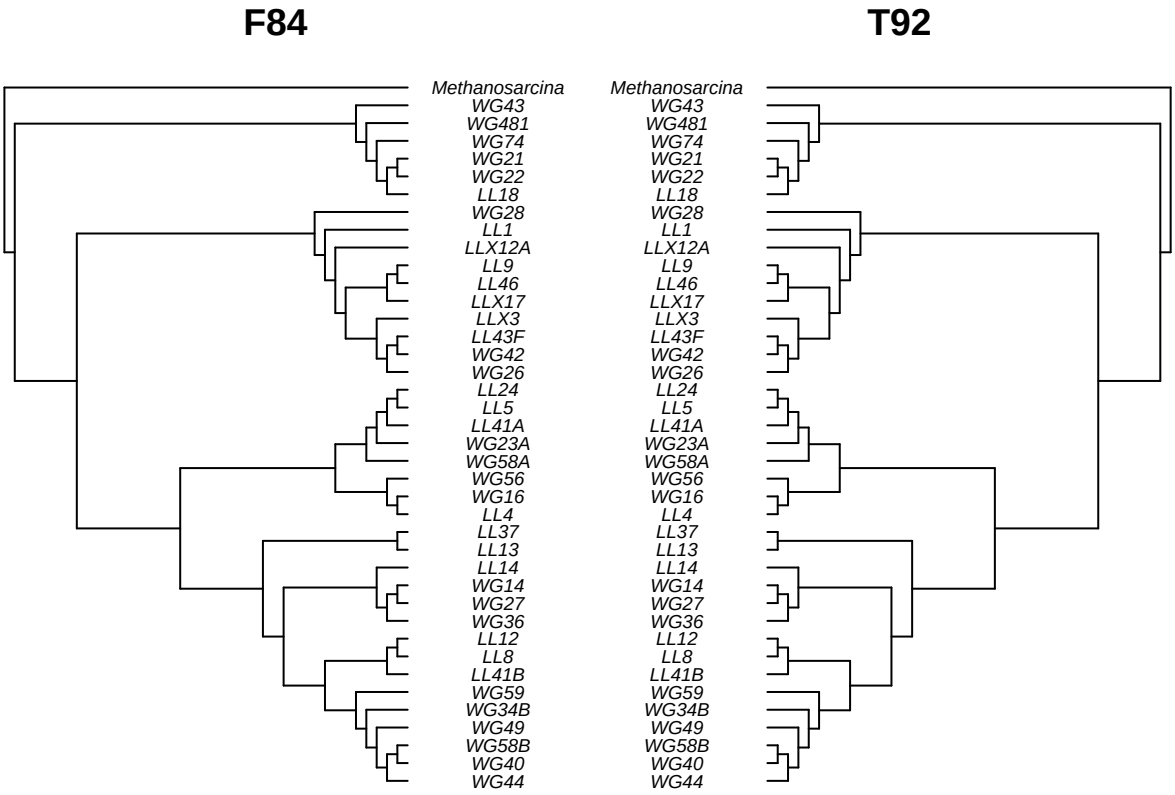


In the R code chunk below, do the following:
1. pick another substitution model,
2. create a distance matrix and tree for this model,
3. make a saturation plot that compares that model to the *Felsenstein (F84)* model,
4. make a cophylogenetic plot that compares the topologies of both models, and
5.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

```
seq.dist.F84 <- dist.dna(p.DNAbin, model = 'F84', pairwise.deletion = FALSE)
seq.dist.T92 <- dist.dna(p.DNAbin, model = 'T92', pairwise.deletion = FALSE)

par(mar = c(5, 5, 2, 1) + 0.1)
plot(seq.dist.F84, seq.dist.T92, pch = 20, col = 'red', las = 1, asp = 1,
     xlim = c(0, 0.7), ylim = c(0, 0.7), xlab = 'F84 Distance',
     ylab = 'T92 Distance')
abline(b = 1, a = 0, lty = 2)
text(0.65, 0.6, '1:1')
```




```
F84.tree <- bionj(seq.dist.F84)
T92.tree <- bionj(seq.dist.T92)

F84.outgroup <- match('Methanosarcina', F84.tree$tip.label)
T92.outgroup <- match('Methanosarcina', T92.tree$tip.label)
F84.rooted <- root(F84.tree, F84.outgroup, resolve.root = TRUE)
T92.rooted <- root(T92.tree, T92.outgroup, resolve.root = TRUE)

layout(matrix(c(1,2), 1, 2), width = c(1,1))
par(mar = c(1, 1, 2, 0))
plot.phylo(F84.rooted, type = 'phylogram', direction = 'right',
           show.tip.label = TRUE, use.edge.length = FALSE,
           adj = 0.5, cex = 0.6, label.offset = 2, main = 'F84')
plot.phylo(T92.rooted, type = 'phylogram', direction = 'left',
           show.tip.label = TRUE, use.edge.length = FALSE,
           adj = 0.5, cex = 0.6, label.offset = 2, main = 'T92')
```



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?

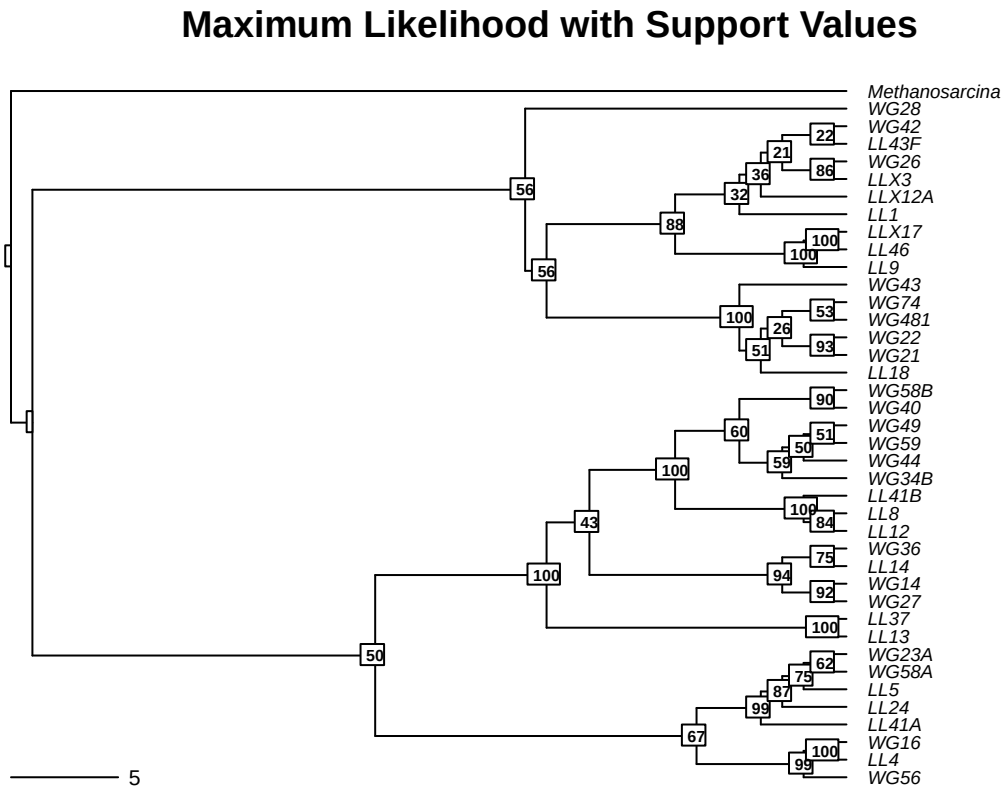
Answer 4a: The Tamura model accounts for transition/transversion mutation bias, and for G/C nucleotide content. The F84 model is the same except that it allows all four nucleotides to have different frequencies instead of incorporating G and C content ***Answer 4b***: There do not appear to be any practical differences whatsoever between the F84 and T92 models. This suggests to me that modelling G/C content is sufficient, rather than separate frequencies for each base. ***Answer 4c***: Both models have separate rates for transitions and transversions, so there isn't much information to be gained from this comparison.

C) ANALYZING A MAXIMUM LIKELIHOOD TREE

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

```
ml.bootstrap <- read.tree('./data/ml_tree/RAxML_bipartitions.T1')
par(mar = c(1,1,2,1) + 0.1)
plot.phylo(ml.bootstrap, type = 'phylogram', direction = 'right',
           show.tip.label = TRUE, use.edge.length = 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 tree is quite different from the neighbour-joining tree. This is because the maximum-likelihood method for tree construction is much more statistically rigorous and uses a fundamentally different algorithm than the neighbour-joining approach. ***Answer 5b***: Bootstrapping is essentially a statistical measure of the confidence in the phylogeny in the face of sampling error. By resampling the idea repeatedly, we get an idea of how variance in the dataset affects our conclusions about the tree. If there is weak phylogenetic signal at a particular node, it will have poor bootstrap support. ***Answer 5c***: The bootstrap value is the number of bootstrap replicates that supported the internal node on which it is labelled. ***Answer 5d***: There are many branches that have low support values, in the range of 20 - 60. ***Answer 5e***: Our trust in the branches with low support values should definitely be low. However, (and I know this is beyond the scope of the course), even 100% bootstrap support is not a guarantee that the node is reflective of true species relationships, because of biological causes of discordance (ie. incomplete lineage sorting and introgression). Another metric, the concordance factor, provides a summary of these forces.

5) INTEGRATING TRAITS AND PHYLOGENY

A. Loading Trait Database

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

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

p.growth.std <- p.growth / (apply(p.growth, 1, sum))
```

B. Trait Manipulations

In the R code chunk below, do the following:
1. calculate the maximum growth rate (μ_{max}) of each isolate across all phosphorus types,
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))

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')
```

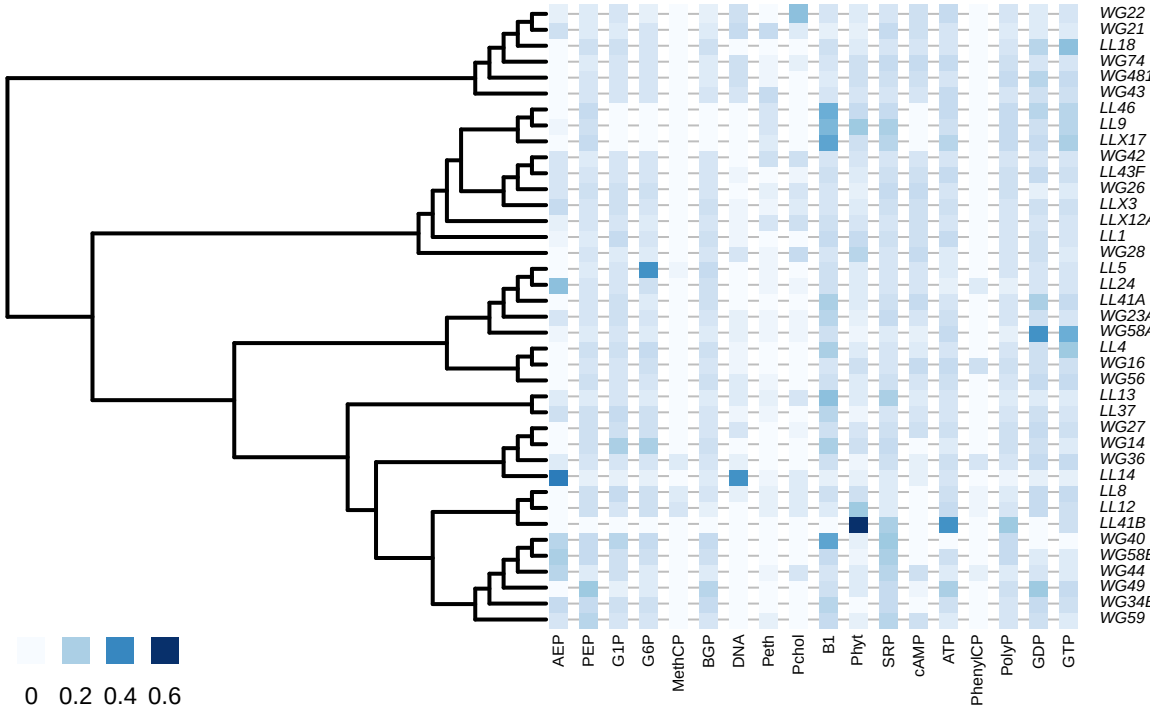
C. Visualizing Traits on Trees

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.

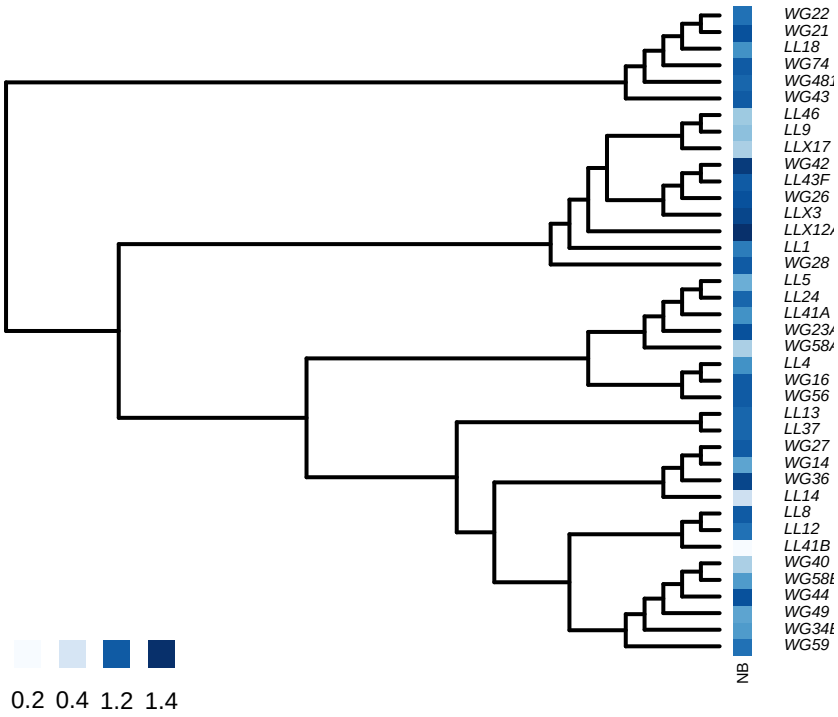
```
nj.tree <- bionj(seq.dist.F84)
outgroup <- match('Methanosarcina', nj.tree$tip.label)
nj.rooted <- root(nj.tree, outgroup, resolve.root = TRUE)
nj.rooted <- drop.tip(nj.rooted, 'Methanosarcina')
```

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

```
mypalette <- colorRampPalette(brewer.pal(9, 'Blues'))

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)
```




```
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,
              ratio.tree = 0.9, 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?

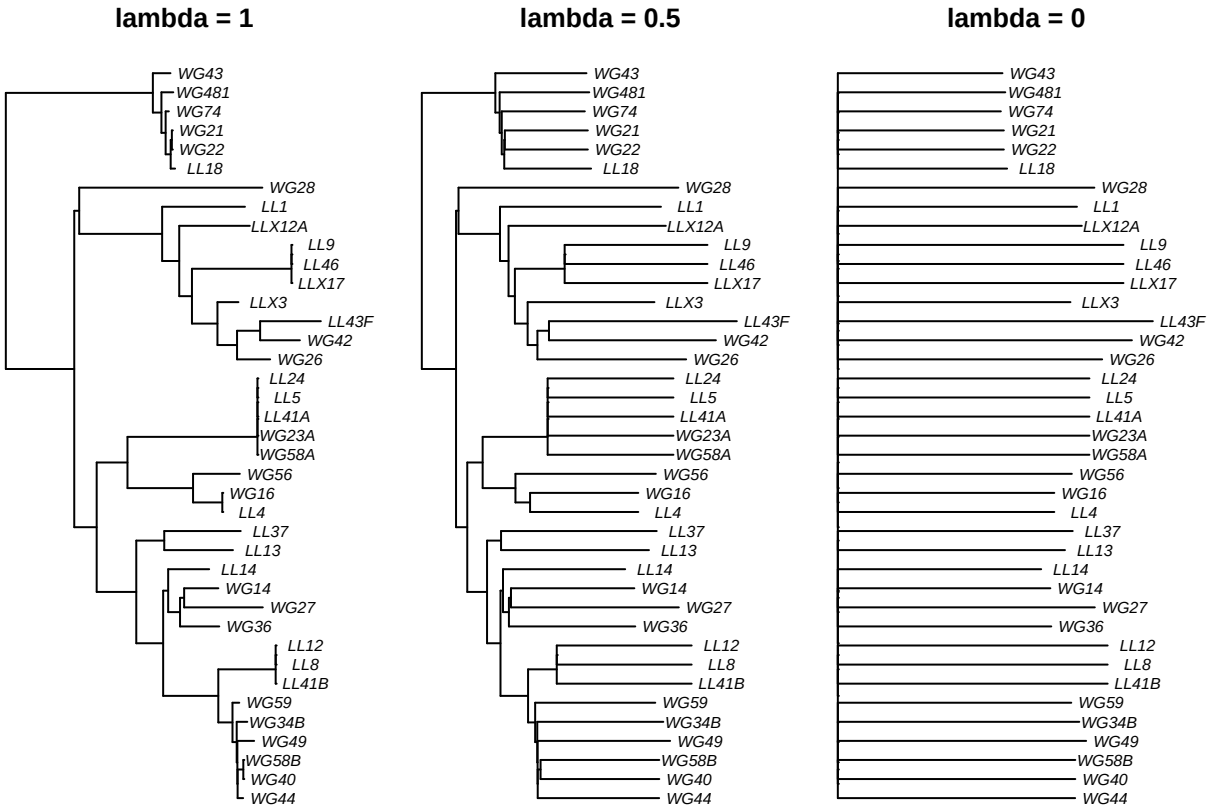
Answer 6a: If there is a generalist-specialist trade-off, then specialists will have a higher max growth rate (across all resources) and lower niche breadth, and vice versa. ***Answer 6b***: See above. There should be a negative correlation between niche breadth and max growth rate.

6) HYPOTHESIS TESTING

A) Phylogenetic Signal: Pagel's Lambda

In the R code chunk below, do the following:
1. create two rescaled phylogenetic trees using lambda values of 0.5 and 0,
2. plot your original tree and the two scaled trees, and
3. label and customize the trees as desired.

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

layout(matrix(c(1,2,3), 1, 3), width = c(1, 1, 1))
par(mar = c(1, 0.5, 2, 0.5) + 0.1)
plot(nj.rooted, main = 'lambda = 1', cex = 0.7, adj = 0.5)
plot(nj.lambda.5, main = 'lambda = 0.5', cex = 0.7, adj = 0.5)
plot(nj.lambda.0, main = 'lambda = 0', cex = 0.7, adj = 0.5)
```



In the R code chunk below, do the following:
1. use the `fitContinuous()` function to compare your original tree to the transformed trees.

```
fitContinuous(nj.rooted, 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 = 51
##  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 = 0.87
##
##  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 fitted lambda value for the untransformed tree is 0.02, while it is 0 for the transformed tree. ***Answer 7b***: The untransformed tree has an AIC score of -37.32, whereas the transformed tree has a score of -37.3. These scores are virtually identical, so either model would work. ***Answer 7c***: Since the untransformed tree is statistically indistinguishable from the transformed one, there does not appear to be any phylogenetic signal in this data.

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.

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

p.phylosignal <- matrix(NA, 6, 18)
colnames(p.phylosignal) <- colnames(p.growth.std)
rownames(p.phylosignal) <- 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, nj.rooted)
  p.phylosignal[1:5, i] <- round(t(out), 3)
}

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

signal.nb <- phylosignal(nb, nj.rooted)
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 8106.546 1560.168 1885.728 3743.065 492.833 1762.733 5280.064
## PIC.var.P       0.257    0.063    0.096    0.763   0.354    0.033    0.003
## PIC.var.z      -0.812   -1.360   -1.241    0.860  -0.426   -1.672   -1.329
## PIC.P.BH        0.635    0.284    0.346    0.808   0.635    0.198    0.036
##                   Peth    Pchol       B1     Phyt      SRP     cAMP
## K               0.000    0.000    0.000    0.000    0.000    0.000
## PIC.var.obs  1446.463 2368.391 3517.018 9240.368 1307.025  690.723
## PIC.var.mean 1845.385 3323.249 5451.090 9131.618 1589.190 3035.250
## PIC.var.P       0.333    0.388    0.203    0.575    0.330    0.004
## PIC.var.z      -0.485   -0.533   -0.872    0.014   -0.492   -2.391
## PIC.P.BH        0.635    0.635    0.609    0.724    0.635    0.036
##                   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 3064.224  760.633 1214.973 3633.000 3026.962
## PIC.var.P       0.626    0.823    0.505    0.644    0.453
## PIC.var.z       0.434    1.007   -0.161    0.390   -0.225
## PIC.P.BH        0.724    0.823    0.699    0.724    0.680
```

```
signal.nb
```

```
##              K PIC.variance.obs PIC.variance.rnd.mean PIC.variance.P
## 1 3.427719e-06         49966.78              49040.69          0.549
##   PIC.variance.Z
## 1     0.04621705
```

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: Based on the adjusted p-values, there is a weak phylogenetic signal for growth on DNA ($p = 0.036$) and cAMP ($p = 0.045$). There is no signal for niche breadth. ***Answer 8b***: For both significant cases, K is much less than 1, which is indicative 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.

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

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

p.traits <- comparative.data(nj.rooted, p.growth.pa, 'name')

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 :  0.6064025
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.036
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.004
```

```
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 :  0.1632925
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.002
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.29
```

```
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 :  -0.3329355
## Probability of E(D) resulting from no (random) phylogenetic structure :  0.003
## Probability of E(D) resulting from Brownian phylogenetic structure    :  0.602
```

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.

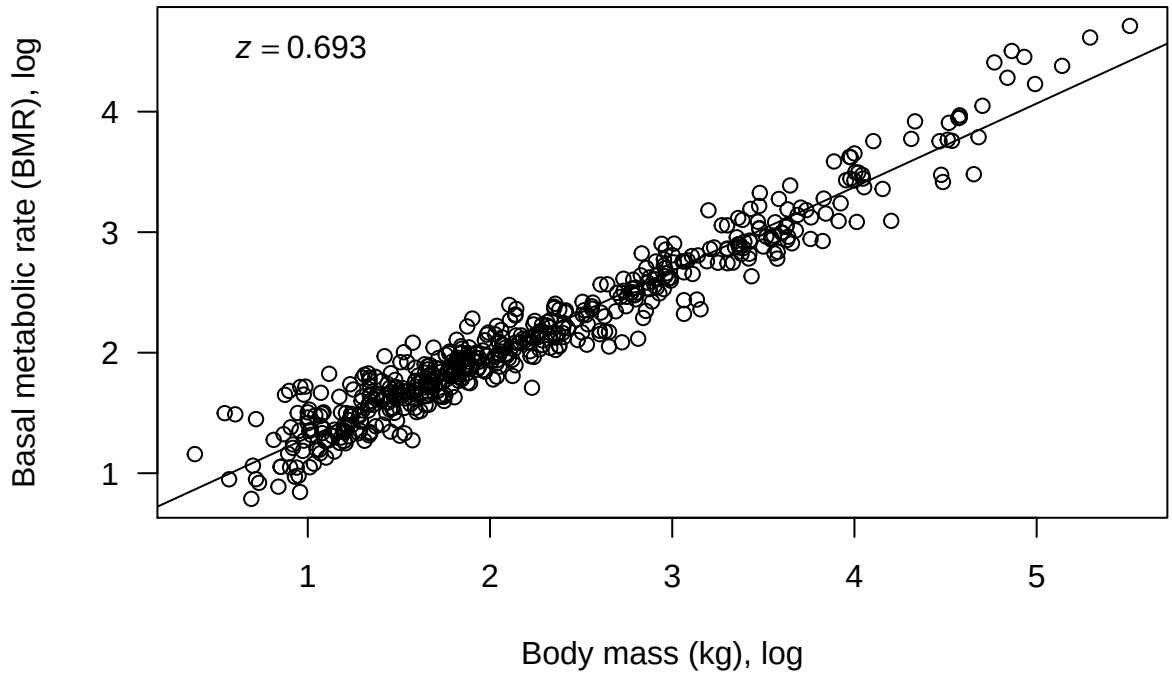
Answer 9a: BGP and cAMP are much more likely to have arisen under a Brownian phylogenetic structure than from no structure. This suggests that neither clustering or overdispersion are likely. For DNA, both models are highly unlikely but no structure is slightly more likely than Brownian structure. The positive value of D for DNA suggests growth on DNA is overdispersed. ***Answer 9b***: The result from Bloomberg's K suggested significant overdispersion for growth on cAMP, whereas the D statistic does not. The other two results are consistent across methods. ***Answer 9c***: I would expect transforming a continuous variable into a categorical variable to result in some loss of statistical power. Also, the D statistic uses permutations to calculcate significance, which have less statistical power than significance tests that make stronger assumptions (if those assumptions are true).

7) PHYLOGENETIC REGRESSION

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

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

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

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

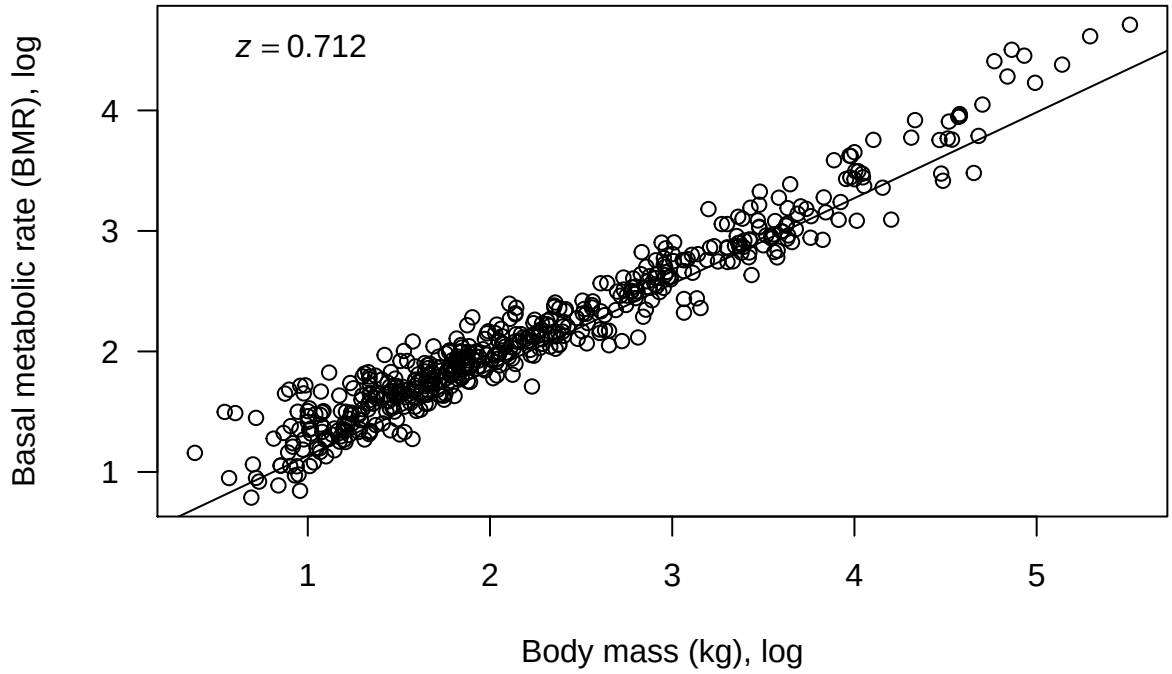



```
summary(fit)
```

```
##
## 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 worsen the fit?
d. Try to come up with a scenario where the relationship between two variables would completely disappear when the underlying phylogeny is accounted for.

> ***Answer 10a***: Shared evolutionary history violates the assumption of linear regression that each observation is independent. ***Answer 10b***: It explicitly accounts for the covariance between observations using a phylogenetic covariance matrix. ***Answer 10c***: The model fits are very similar, but it appears to be slightly worse after accounting for phylogeny. ***Answer 10d***: This would happen if the relationship between body mass and BMR was driven entirely by phylogenetic relatedness. In other words, it would be entirely an artefact of history and not due to any biological mechanism.

7) SYNTHESIS

Work with members of your Team Project to obtain reference sequences for 10 or more taxa in your study. Sequences for plants, animals, and microbes can found in a number of public repositories, but perhaps the most commonly visited site is the National Center for Biotechnology Information (NCBI) https://www.ncbi.nlm.nih.gov/. In almost all cases, researchers must deposit their sequences in places like NCBI before a paper is published. Those sequences are checked by NCBI employees for aspects of quality and given an **accession number**. For example, here an accession number for a fungal isolate that our lab has worked with: JQ797657. You can use the NCBI program nucleotide **BLAST** to find out more about information associated with the isolate, in addition to getting its DNA sequence: https://blast.ncbi.nlm.nih.gov/. Alternatively, you can use the `read.GenBank()` function in the `ape` package to connect to NCBI and directly get the sequence. This is pretty cool. Give it a try.

But before your team proceeds, you need to give some thought to which gene you want to focus on. For microorganisms like the bacteria we worked with above, many people use the ribosomal gene (i.e., 16S rRNA). This has many desirable features, including it is relatively long, highly conserved, and identifies taxa with reasonable resolution. In eukaryotes, ribosomal genes (i.e., 18S) are good for distinguishing course taxonomic resolution (i.e. class level), but it is not so good at resolving genera or species. Therefore, you may need to find another gene to work with, which might include protein-coding gene like cytochrome oxidase (COI) which is on mitochondria and is commonly used in molecular systematics. In plants, the ribulose-bisphosphate carboxylase gene (*rbcL*), which on the chloroplast, is commonly used. Also, non-protein-encoding sequences like those found in **Internal Transcribed Spacer (ITS)** regions between the small and large subunits of of the ribosomal RNA are good for molecular phylogenies. With your team members, do some research and identify a good candidate gene.

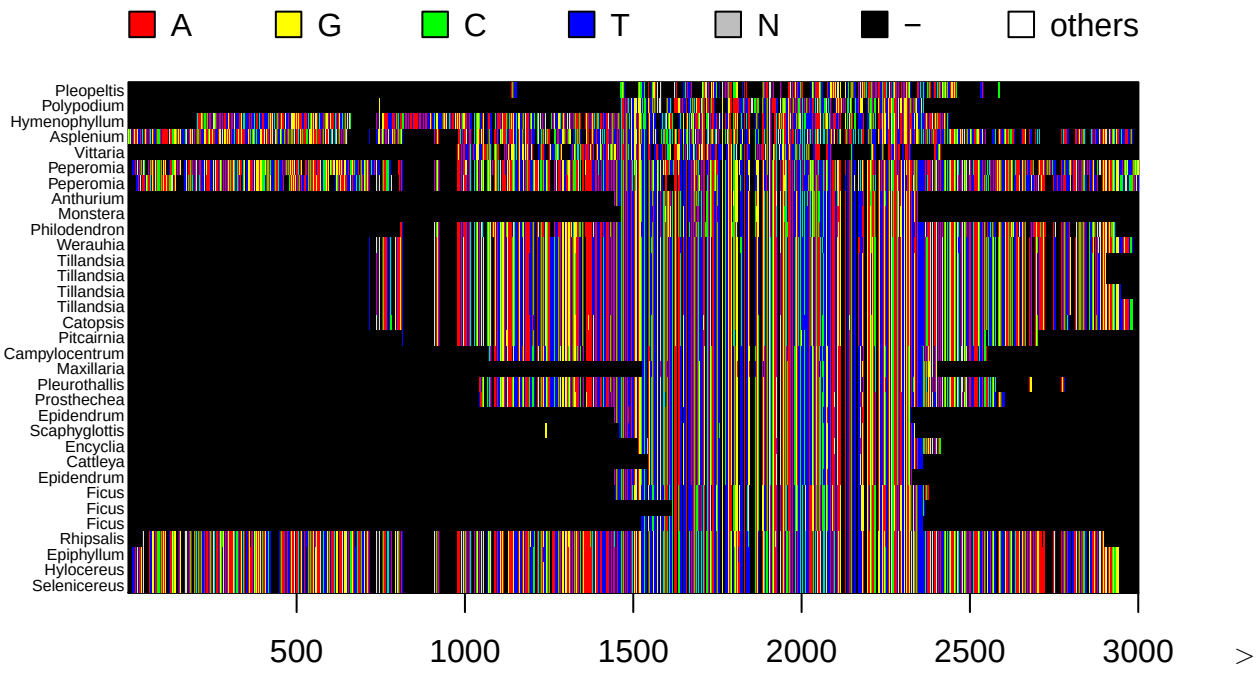
After you identify an appropriate gene, download sequences and create a properly formatted fasta file. Next, align the sequences and confirm that you have a good alignment. Choose a substitution model and make a tree of your choice. Based on the decisions above and the output, does your tree jibe with what is known about the evolutionary history of your organisms? If not, why? Is there anything you could do differently that would improve your tree, especially with regard to future analyses done by your team?

> Diego, who is the one familiar with our study system, decided to pick the chloroplast gene matK as our marker. The sequence record for our dataset is quite messy; some species do not have the matK sequence available, and some records have the matK gene sequence concatenated with some other sequence. In cases where it was reasonable, we picked a genus-level representative sequence in place of the specific species if the species-specific sequence was not available. I aligned our sequences with the following line of code:

```
#./muscle -in epiphytesCR.fasta -out epiphytesCR_align.fasta
```

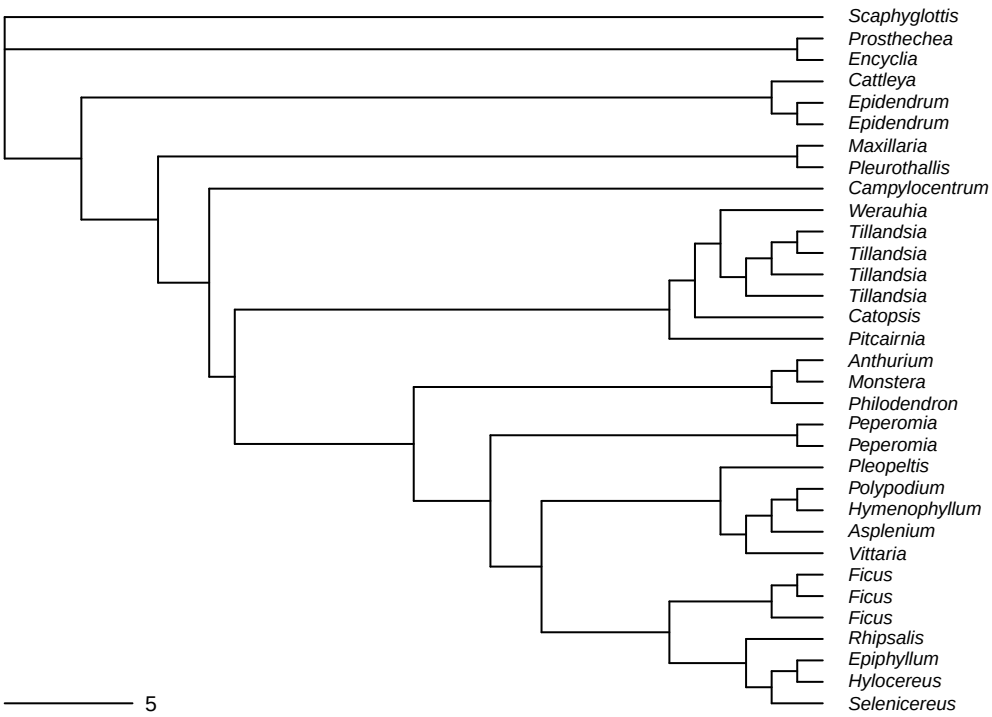
Here is a visualization of our alignment:

```
ep.aln <- read.alignment(file = './data/epiphytesCR_align.fasta', format = 'fasta')
ep.DNAbin <- as.DNAbin(ep.aln)
window <- ep.DNAbin[, 0:3000]
image.DNAbin(window, cex.lab = 0.5)
```



Most of our alignment looks terrible, but there appears to be a useful stretch from roughly positions 1500 to 2400 (which probably corresponds to the matK coding sequence). Technically you are supposed to cut out the unwanted sequence and re-align what's left, but for now we can use this window of the alignment to build our phylogeny:

```
ep.dist <- dist.dna(ep.DNAbin[ ,1500:2400], model = 'K80', pairwise.deletion = FALSE, as.matrix = TRUE)
ep.nj.tree <- bionj(ep.dist)

par(mar = c(1,1,2,1) + 0.1)
plot.phylo(ep.nj.tree, main = 'Neighbour Joining Tree', 'phylogram',
           use.edge.length = FALSE, direction = 'right',
           cex = 0.6, label.offset = 1)
add.scale.bar(cex = 0.7)
```



For some reason the alignment parser can only read in the text in the header before the first whitespace, so there are only genus-level taxa labels in this dataset. Based on these labels it seems like the neighbour-joining tree groups things together fairly well (there are no polyphyletic groupings of genera). This might be good enough for our purposes, but it would be nice to know if it is well-resolved to the species level for future study (although Diego will be the one to know this).