OU Models

Simon Joly

BIO 6008 - Fall 2015

Contents

We previously saw the Brownian Motion model (lecture 2) to describe the evolution of traits on a phylogeny. Here, we will explore other, more complex evolutionary models.

Other models of evolution

The Ornstein-Uhelenbeck (OU) model

The Ornstein-Uhelenbeck (OU) model (Butler and King 2004) is very popular in evolutionary biology. It differs from the BM model by having an optimum trait value and a selecition pressure to maintain (or push) variation towards this optimum.

To refresh our memories, the amount of change for character X over the infinitesimal time in the interval between time t and $t + dt$ for the Brownian Motion model is:

$$dX(t) = \sigma dB(t),$$

where $dB(t)$ is the gaussian distribution.

The OU model is slightly different. First, as mentionned above, it implies that there is an optimal value for the trait. Let's call this optimal value θ. There is also a selection pressure that act to bring the variation towards this optimal value. This selection pressure is normally represented by the Greek letter α. Mathematically, the OU model looks like this:

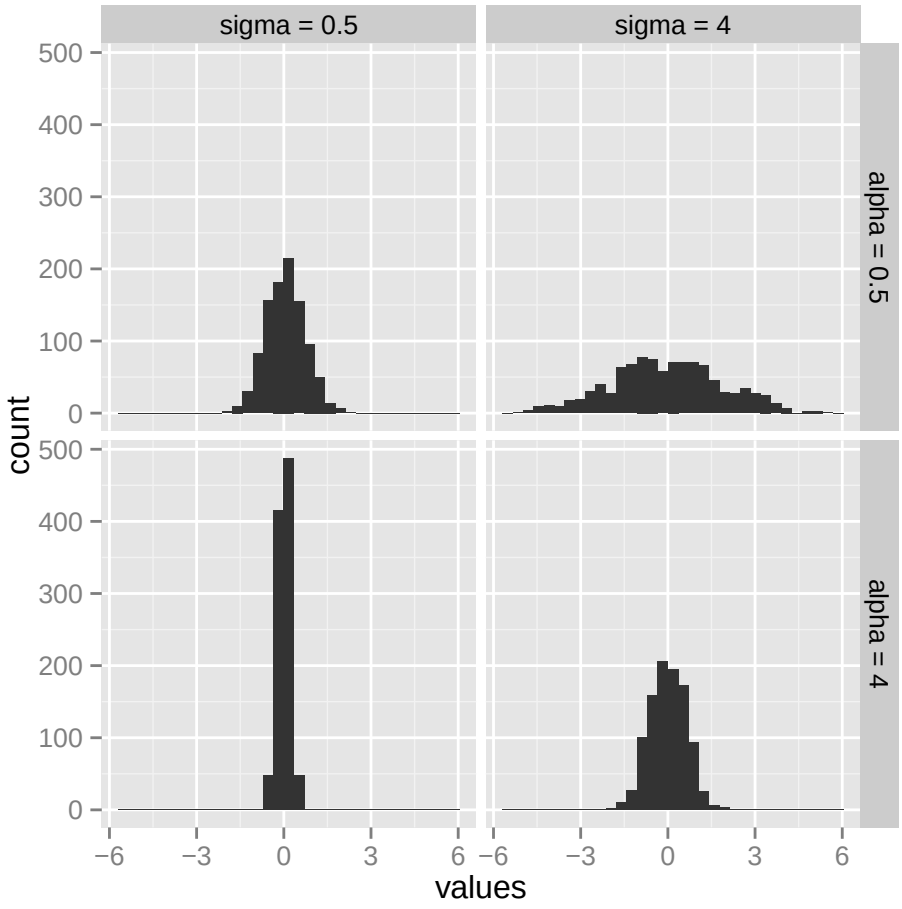
$$dX(t) = \alpha(\theta - X(t))dt + \sigma dB(t).$$

You can see that the right side of the equation for the OU model is identical to the Brownian Motion model. That is, the normal distribution is used to generate variation in the variable of interest. The left side of the formula involves selection. Actually, note that if $\alpha = 0$, which implies that there is no selection, then the OU model collapse to the simpler BM model.

Interpretation

The OU model can be interpreted in different ways. First, it can be seen as a balancing selection model where selection acts to always bring back the variation towards the optimum. However, in some other cases, it could also represent a directional selection model, in which case selection acts to bring the character to a new value. Actually, the interpretation depends largely on the ancestral value of the trait and the optimum value of the model.

If you remember well from lecture 2, the σ parameter was used to control the overall variation the trait. With the OU model, both σ and α can play this role. For instance, lets look at the distribution of trait values for characters simulated with different values of α ($\alpha = 0.5$ and $\alpha = 4$) and σ ($\sigma = 0.5$ and $\sigma = 4$).



As you can see, the distribution of the trait values are very similar for the simulations with $\alpha = 0.5$ and $\sigma = 0.5$ to that with $\alpha = 4$ and $\sigma = 4$. In other words, larger variation with greater selection gives a result similar to small variation and small selection. To be able to distinguish between the OU model and the BM model, it thus becomes important to consider the phylogenetic tree, as you can see that the distribution of the traits alone are not sufficient.

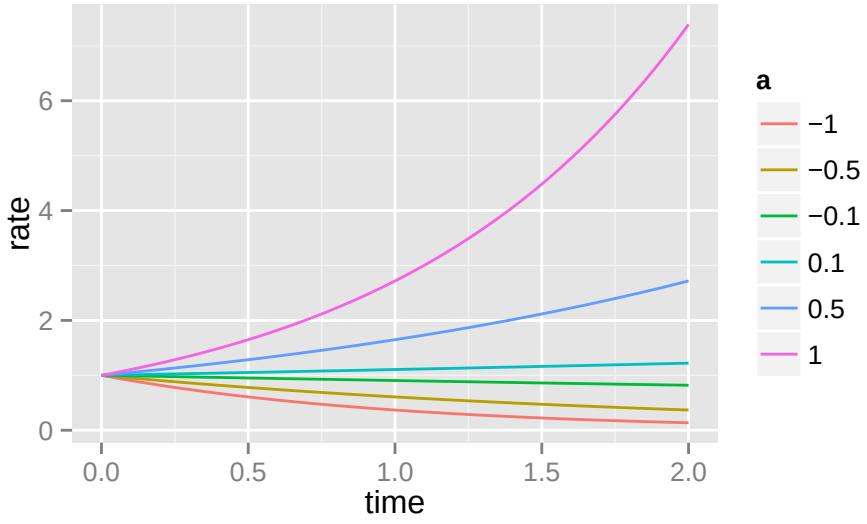
The early burst (EB) model

This Early-Burst model (Harmon et al. 2010) is also called the ACDC model (for Accelerating-decelerating: Blomberg et al. 2003). The EB model has a rate of evolution that increases or decreases exponentially with time, with the rate increase given by the parameter a. For instance, the rate at time t is given by the formula:

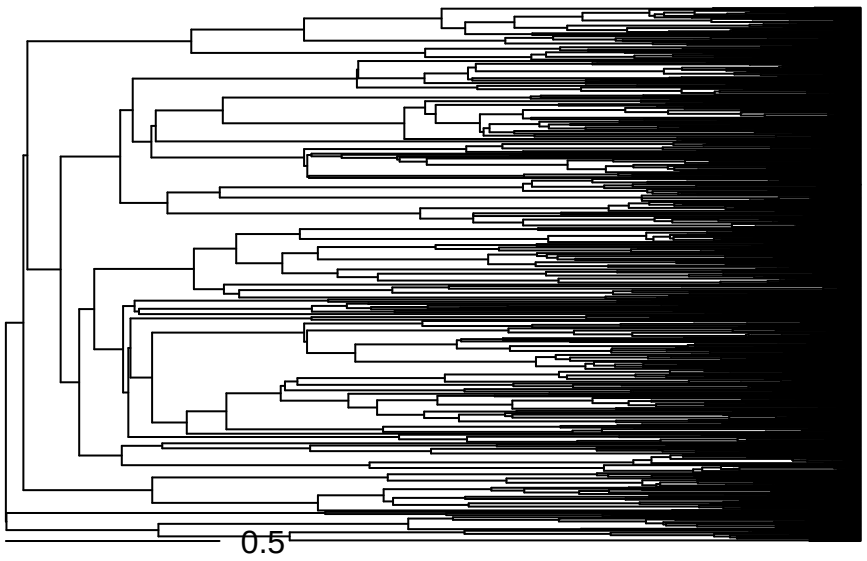
$$r(t) = r(0) \times exp(a \times t),$$

where $r(0)$ is the initial rate.

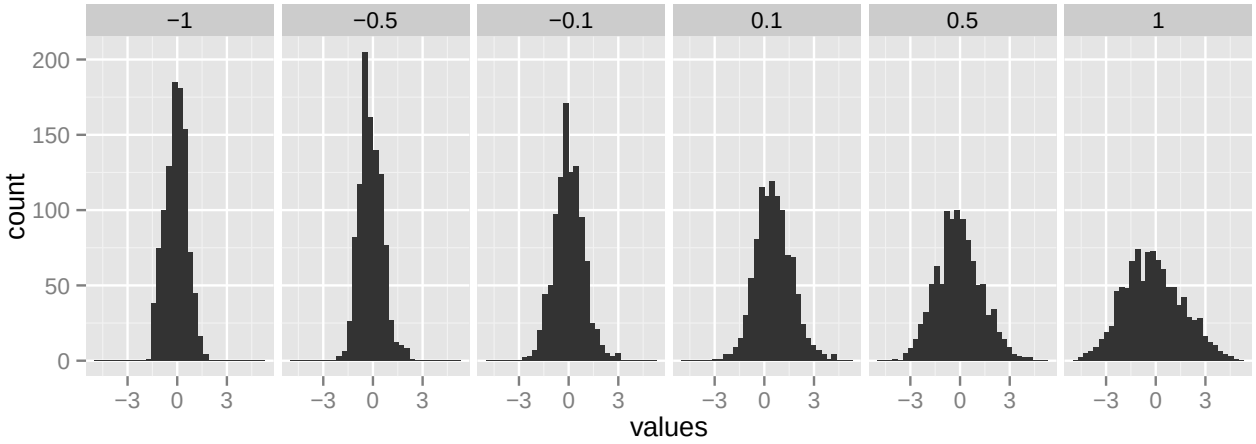
Let's look at the relationship between the rate and time for different values of the a parameter, supposing that the intial rate is 1.



Now, let's compare the expectation for a trait distribution for $a = 1$ and $a = 10$ for the following tree of 500 taxa with total length of 2.



Here are the trait distribution expectations.



As you can see, a greater burst of species early in the tree results in greater observed variation amongst the tips of the tree.

The punctuational (speciational) model

This is an interesting model in which the amount of character divergence is realted to the number of speciation events between two species. The idea is to transform the branches of the phylogeny so that all branches have the same weight. This is done using the κ transform of Pagel's trait evolution models (1999).

> Note that interpretation might be difficult if you do not have a complete taxonomic smapling with this model because some speciation events will be missing from the phylogeny.

Other models

There are several other models (or tree transformation) that are available. You can read about some of them in the help pages of the `rescale` function of the `geiger` package.

```
require(geiger)
?rescale
```

Fitting different models

The different models of evolution can be fitted using the `fitContinuous` function of the `geiger` package. We will try to fit the different models on the seed plants phylogeny of Paquette et al. (2015).

```
require(ape)
seedplantstree <- read.nexus("./data/seedplants.tre")
seedplantsdata <- read.csv2("./data/seedplants.csv")
# Remove species for which we don't have complete data
seedplantsdata <- na.omit(seedplantsdata)
# Remove species in the tree that are not in the data matrix
species.to.exclude <- seedplantstree$tip.label[!(seedplantstree$tip.label %in%
                                                   seedplantsdata$Code)]
seedplantstree <- drop.tip(seedplantstree,species.to.exclude)
rm(species.to.exclude)
# Name the rows of the data.frame with the species codes used as tree labels
rownames(seedplantsdata) <- seedplantsdata$Code
seedplantsdata <- seedplantsdata[,-1]
# Order the data in the same order as the tip.label of the tree. In the present
# example, this was already the case.
seedplantsdata <- seedplantsdata[seedplantstree$tip.label,]
# Extract trait data into vectors
Wd <- seedplantsdata$Wd
Shade <- seedplantsdata$Shade
Sm <- seedplantsdata$Sm
N <- seedplantsdata$N
# Important: Give names to your vectors
names(Wd) <- names(Shade) <- names(Sm) <- names(N) <- row.names(seedplantsdata)
```

Now, let's fit the wood density (Wd) trait under different models of evolution. Let's start with the Brownian Motion model.

```
require(geiger)
wd.bm <- fitContinuous(seedplantstree,Wd,model="BM")
```

```
## Loading required package: parallel
```

```
wd.bm
```

```
## GEIGER-fitted comparative model of continuous data
##  fitted 'BM' model parameters:
##  sigsq = 2.257138
##  z0 = 0.431869
##
##  model summary:
##  log-likelihood = 36.226944
##  AIC = -68.453888
```

```
##  AICc = -68.231665
##  free parameters = 2
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 0
##  frequency of best fit = 1.00
##
##  object summary:
##  'lik' -- likelihood function
##  'bnd' -- bounds for likelihood search
##  'res' -- optimization iteration summary
##  'opt' -- maximum likelihood parameter estimates
```

The results gives a lot of information. It first gives the ML estimates for the 2 parameters of model, σ^2 and z_0, which is the estimated value at the root of the tree. It also gives the log-likelihood and the AIC and AICc.

Let's compare with the OU model.

```
wd.ou <- fitContinuous(seedplantstree,Wd,model="OU")
wd.ou
```

```
## GEIGER-fitted comparative model of continuous data
##  fitted 'OU' model parameters:
##  alpha = 2.718282
##  sigsq = 2.269266
##  z0 = 0.432921
##
##  model summary:
##  log-likelihood = 37.688635
##  AIC = -69.377270
##  AICc = -68.924440
##  free parameters = 3
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 0
##  frequency of best fit = 0.31
##
##  object summary:
##  'lik' -- likelihood function
##  'bnd' -- bounds for likelihood search
##  'res' -- optimization iteration summary
##  'opt' -- maximum likelihood parameter estimates
```

You can see that compared to the BM model, the OU model has the α parameter. In this case, the z_0 parameter is both for the ancestral state and the optimal value of the model.

Let's also fit the early-burst and speciational models and make a table to compare them.

```
# Fit the Early-Burst model
wd.eb <- fitContinuous(seedplantstree,Wd,model="EB")
# Fit the speciational model
wd.spe <- fitContinuous(seedplantstree,Wd,model="kappa")
```

```r
# Create a table to store de results
results.evo <- data.frame(model=c("BM","OU","EB","speciational"),
                          lnL=numeric(4),AICc=numeric(4),params=numeric(4))
# Put the informtion in the table
results.evo[1,-1]<-c(wd.bm$opt$lnL,wd.bm$opt$aicc,wd.bm$opt$k)
results.evo[2,-1]<-c(wd.ou$opt$lnL,wd.ou$opt$aicc,wd.ou$opt$k)
results.evo[3,-1]<-c(wd.eb$opt$lnL,wd.eb$opt$aicc,wd.eb$opt$k)
results.evo[4,-1]<-c(wd.spe$opt$lnL,wd.spe$opt$aicc,wd.spe$opt$k)
# Order the results by AICc
results.evo <- results.evo[order(results.evo$AICc),]
results.evo
```

```
##          model      lnL       AICc params
## 4 speciational 65.92257 -125.39231      3
## 2           OU 37.68864  -68.92444      3
## 1           BM 36.22694  -68.23167      2
## 3           EB 36.22695  -66.00107      3
```

You can see that the speciational model recieved the best AICc value, which makes it the best model. But as mentionned above, the result from this model needs to be interpreted very carefully. In the present example, there are a lot of species missing from the flowering plant phylogeny. So in this specific case, this model does not make much sense.

It is also possible to test a non-phylogenetic model, which is called "white noise".

```r
wd.wn <- fitContinuous(seedplantstree,Wd,model="white")
wd.wn
```

```
## GEIGER-fitted comparative model of continuous data
##  fitted 'white' model parameters:
##  sigsq = 0.009373
##  z0 = 0.468596
##
##  model summary:
##  log-likelihood = 52.211857
##  AIC = -100.423714
##  AICc = -100.201492
##  free parameters = 2
##
## Convergence diagnostics:
##  optimization iterations = 100
##  failed iterations = 0
##  frequency of best fit = 1.00
##
##  object summary:
##  'lik' -- likelihood function
##  'bnd' -- bounds for likelihood search
##  'res' -- optimization iteration summary
##  'opt' -- maximum likelihood parameter estimates
```

You can see that this model is much better than the other models, but not as much as the speciational model. This would tend to suggest that there are little phylogenetic information in the trait. Again, that could be due to the nature of the traits studied here.

Interpretation

The results from the fit of the model can be intepreted directly. For instance, if the OU model is prefered to the BM motion model for a given trait, then one might conclude that it has evolved under balancing selection. Alternatively, if the speciation model is prefered, one might conclude that the trait has mostly evolved at the speciation events on the phylogeny (see Joly et al. 2014).

But the models can be of much more use. For instance, they can be used in simulations to predict data with which the empirical data can be compared to. This is what we will see in the next section.

Simulating data under different models of evolution

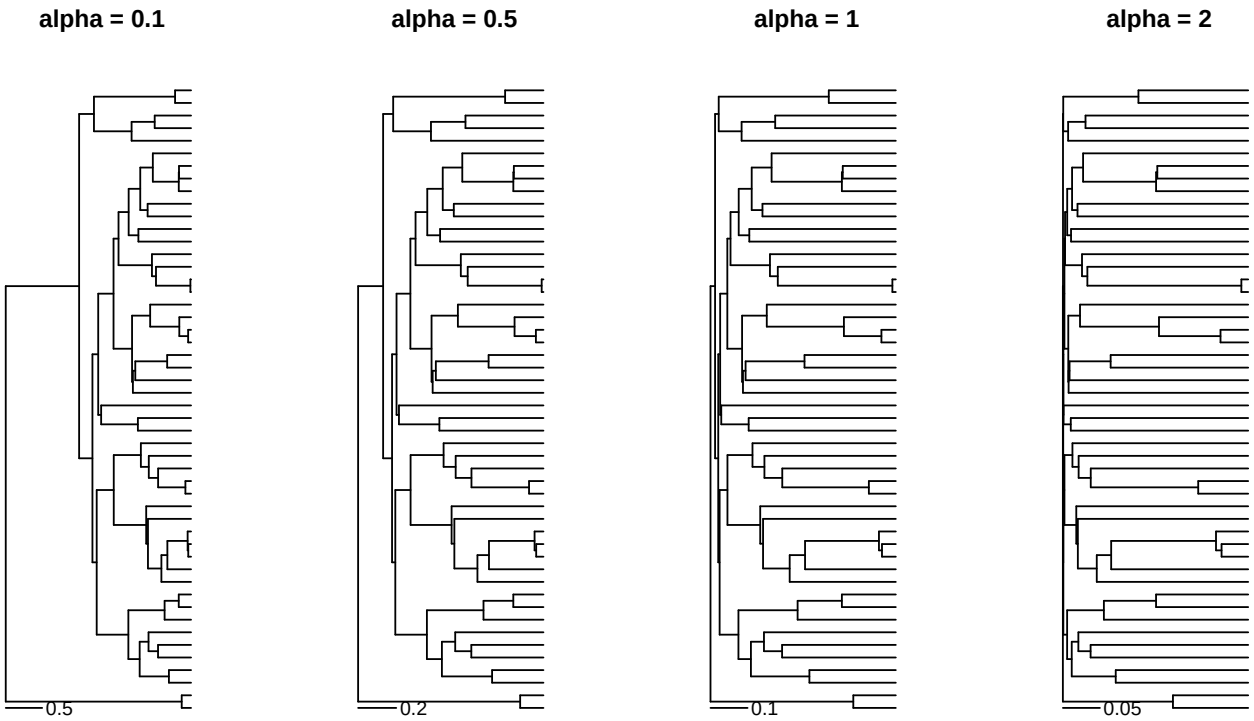
It is relatively easy to simulate data under the different evolutionary described above. To do this, we actually use a trick. In practice, the different models described above can all be seen as different ways to give weights to the branches of the phylogeny. At one end of the spectrum, you have the BM model where the branches a not modified. At the other, the white noise model, which is a non phylogenetic model, can be obtained from the BM model by giving branch lengths of 0 to all internal branches of the phylogeny. This gives a star phylogeny in which relationships are not considered.

The other models can be modelled similarly by reshaping the original phylogeny. The function `rescale` of the `geiger` package does just this. For instance, if you want to obtain a phylogeny that would correspond to a OU model with $\alpha = 4$ from an initial tree called `a_tree`, you could do the following:

```
ou_tree <- rescale(a_tree, model="OU",4)
```

To help understand how the tree topologies are affected by the models, let's look at a few examples of tree topologies reshaped to correspond to OU models with different alpha values.

```
# Number of taxa in the tree
ntaxa=50
# Simulate a tree
a_tree<-pbtree(n=ntaxa)
# A few transformations
v<-rescale(a_tree,"OU",0.1)
w<-rescale(a_tree,"OU",0.5)
x<-rescale(a_tree,"OU",1)
y<-rescale(a_tree,"OU",2)
op <- par(mfrow=c(1,4))
plot(v,show.tip.label = FALSE,no.margin=FALSE,main="alpha = 0.1");add.scale.bar()
plot(w,show.tip.label = FALSE,no.margin=FALSE,main="alpha = 0.5");add.scale.bar()
plot(x,show.tip.label = FALSE,no.margin=FALSE,main="alpha = 1");add.scale.bar()
plot(y,show.tip.label = FALSE,no.margin=FALSE,main="alpha = 2");add.scale.bar()
```




```
par(op)
rm(atree,ntaxa,v,w,x,y)
```

```
## Warning in rm(atree, ntaxa, v, w, x, y): objet 'atree' introuvable
```

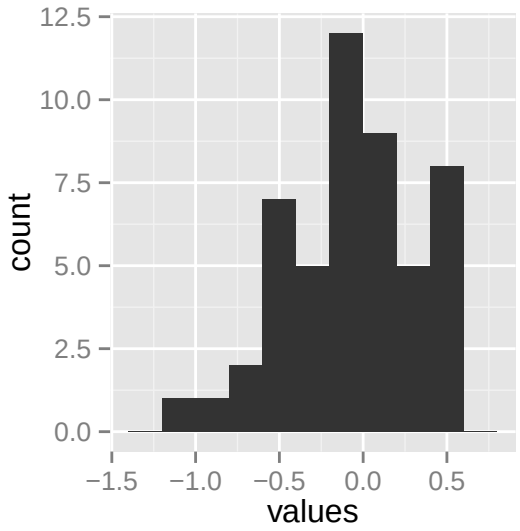
You can observe two things when increasing the value of the α parameter. First, the total tree length gets smaller, which will result in species having more similar trait values because it leaves less time to diverge from one another. Second, you can see that the nodes of the tree are pulled towards the base of the tree, which has the consequence of making all species relatively similar. Put it another way, species will not have much more similar trait values with their close relative than to distant species. This is congruent with a OU model that mimicks balancing selection; that is, there is little drift in trait values between lineages.

Simulations

To perform simulations, we could then use these transformed phylogenies to simulate traits using the Brownian Motion model. The resulting phylogeny will thus reflect traits simulated under the model used to reshape the phylogeny.

For instance, let's simulate a trait under the OU model with $\alpha = 1$ and $\sigma^2 = 0.5$.

```
# Number of taxa in the tree
ntaxa=50
# Simulate a tree
a_tree<-pbtree(n=ntaxa)
#Simulations
trait.OU <- fastBM(rescale(a_tree,"OU",1),sig2=0.5)
data<-data.frame(alpha=1,values=trait.OU)
ggplot(data,aes(x=values),y=alpha)+geom_histogram(binwidth=0.2)
```



It is easy to do the same thing with other models of evolution.

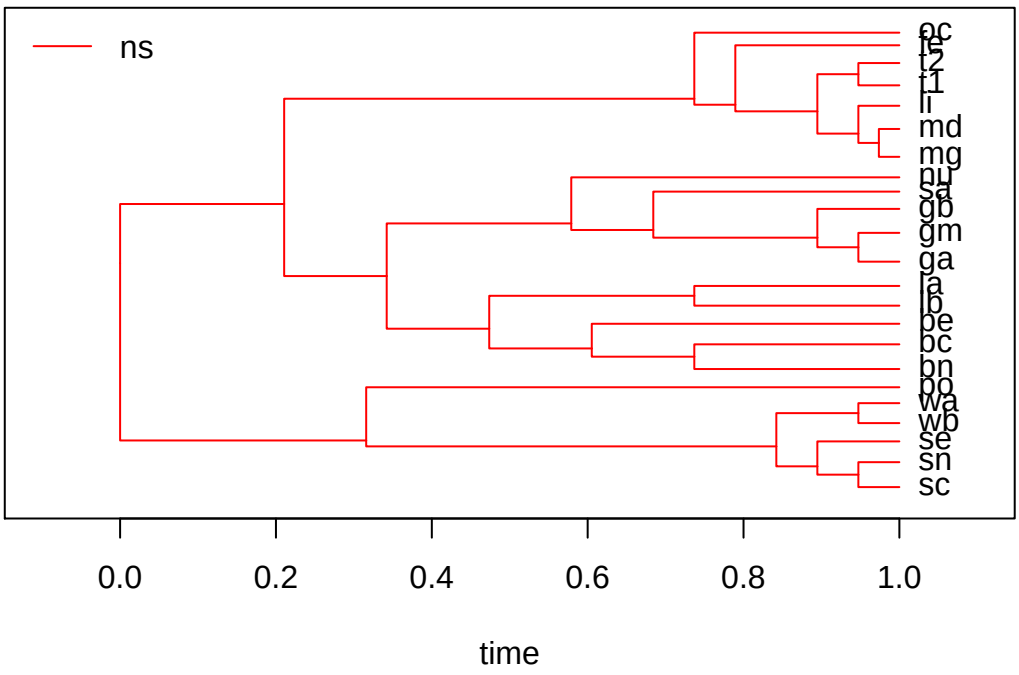
OU model with multiple regimes per tree

In some instances, we might want to evaluate models where different branches of the phylogeny evolve under different selection regimes. Butler and King (2004) have described how to do this. To see how it works, we will use the *Anolis* dataset they used in their paper to illustrate the multiple regime approach. The data describes body size in a group of *Anolis* lizards that evolved sexual dimorphisms in the Antilles. In islands where two species of this group are found, these differ in size. They thus tested whether the small, medium and tall *Anolis* evolved under different selection regimes. We will simplify their analyses here by showing only two of the scenarios tested. One scnenario has an OU model with one regime applied across the whole tree and the second scenario has one OU model but with three regimes, with the regimes painted on the internal branches according to a linear parsimony reconstruction. Let's fit the two models and see what they look like.

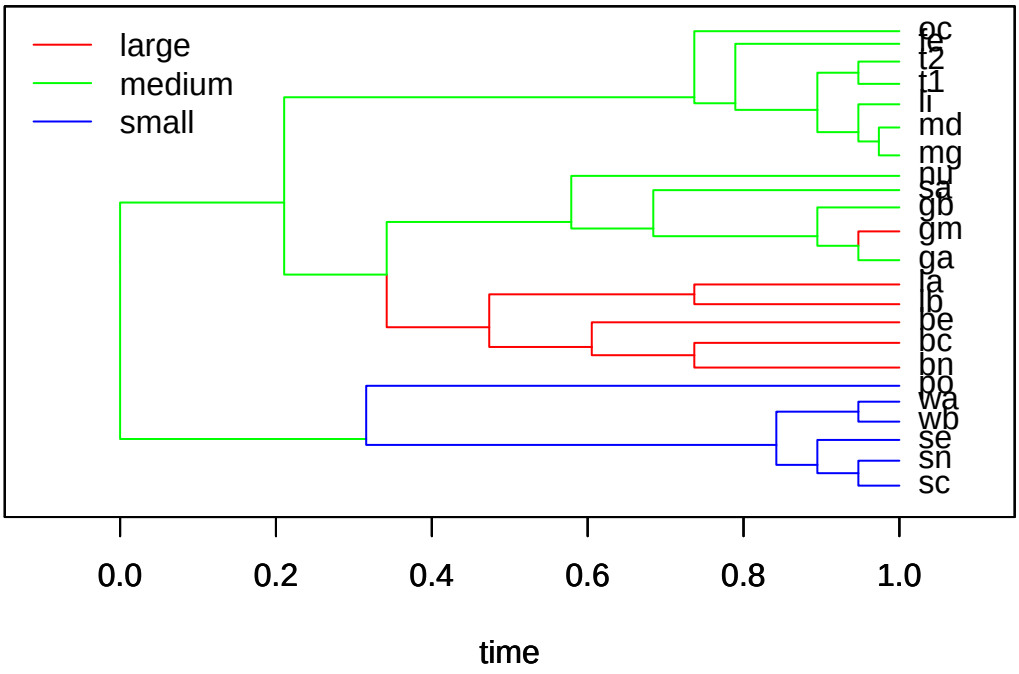
```
library(ouch)
```

```
## Loading required package: subplex
```

```
# Load the lizard data
data(bimac)
# Prepare tree in OUCH format
tree <- with(bimac,ouchtree(node,ancestor,time/max(time),species))
# Fit the OU1 model
h1 <- hansen(log(bimac['size']),tree,bimac['OU.1'],sqrt.alpha=1,sigma=1)
# Fit the OU3 model
h2 <- hansen(log(bimac['size']),tree,bimac['OU.LP'],sqrt.alpha=1,
             sigma=1,reltol=1e-5)
#Refine the fit of the OU3 model
h2 <- update(h2,method='subplex',reltol=1e-11,parscale=c(0.1,0.1),hessian=TRUE)
# Plot the two models
plot(h1)
```




```
plot(h2)
```



Now, let's make a table to compare the fit of the two models.

```
results <- data.frame(model=c("OU.1","OU.3"),
                      loglik=c(summary(h1)$loglik,summary(h2)$loglik),
                      AIC=c(summary(h1)$aic,summary(h2)$aic),
                      AICc=c(summary(h1)$aic.c,summary(h2)$aic.c),
                      params=c(summary(h1)$dof,summary(h2)$dof))
# Reorder according to AICc values
```

```
results <- results[order(results$AICc),]
results
```

```
##   model   loglik       AIC      AICc params
## 2  OU.3 24.81823 -39.63646 -36.10705      5
## 1  OU.1 15.69682 -25.39364 -24.13048      3
```

As you can see, in this example, the more complex model with three regimes (OU3) has a better fit to the data. This indicates that species with different body sizes likely evolved under different selective regimes. To see the fitted parameters values, you can type `h2`.

```
h2
```

```
##
## call:
## hansen(data = data, tree = object, regimes = regimes, sqrt.alpha = sqrt.alpha,
##     sigma = sigma, method = "subplex", hessian = TRUE, reltol = 1e-11,
##     parscale = ..3)
##    nodes ancestors     times labels  OU.LP     size
## 1      1      <NA> 0.0000000   <NA> medium       NA
## 2      2         1 0.3157895   <NA> medium       NA
## 3      3         2 0.8421053   <NA>  small       NA
## 4      4         3 0.8947368   <NA>  small       NA
## 5      5         4 0.9473684   <NA>  small       NA
## 6      6         3 0.9473684   <NA>  small       NA
## 7      7         1 0.2105263   <NA> medium       NA
## 8      8         7 0.3421053   <NA> medium       NA
## 9      9         8 0.4736842   <NA>  large       NA
## 10    10         9 0.6052632   <NA>  large       NA
## 11    11        10 0.7368421   <NA>  large       NA
## 12    12         9 0.7368421   <NA>  large       NA
## 13    13         8 0.5789474   <NA> medium       NA
## 14    14        13 0.6842105   <NA> medium       NA
## 15    15        14 0.8947368   <NA> medium       NA
## 16    16        15 0.9473684   <NA> medium       NA
## 17    17         7 0.7368421   <NA> medium       NA
## 18    18        17 0.7894737   <NA> medium       NA
## 19    19        18 0.8947368   <NA> medium       NA
## 20    20        19 0.9473684   <NA> medium       NA
## 21    21        20 0.9736842   <NA> medium       NA
## 22    22        19 0.9473684   <NA> medium       NA
## 23    23         2 1.0000000     po  small 2.602690
## 24    24         4 1.0000000     se  small 2.660260
## 25    25         5 1.0000000     sc  small 2.660260
## 26    26         5 1.0000000     sn  small 2.653242
## 27    27         6 1.0000000     wb  small 2.674149
## 28    28         6 1.0000000     wa  small 2.701361
## 29    29        10 1.0000000     be  large 3.161247
## 30    30        11 1.0000000     bn  large 3.299534
## 31    31        11 1.0000000     bc  large 3.328627
## 32    32        12 1.0000000     lb  large 3.353407
## 33    33        12 1.0000000     la  large 3.360375
## 34    34        13 1.0000000     nu medium 3.049273
```

```
## 35    35        14 1.0000000     sa medium 2.906901
## 36    36        15 1.0000000     gb medium 2.980619
## 37    37        16 1.0000000     ga medium 2.933857
## 38    38        16 1.0000000     gm  large 2.975530
## 39    39        17 1.0000000     oc medium 3.104587
## 40    40        18 1.0000000     fe medium 3.346389
## 41    41        20 1.0000000     li medium 2.928524
## 42    42        21 1.0000000     mg medium 2.939162
## 43    43        21 1.0000000     md medium 2.990720
## 44    44        22 1.0000000     t1 medium 3.058707
## 45    45        22 1.0000000     t2 medium 3.068053
##
## alpha:
##          [,1]
## [1,] 2.610141
##
## sigma squared:
##            [,1]
## [1,] 0.05054862
##
## theta:
## $size
##    large   medium    small
## 3.355242 3.040732 2.565031
##
##    loglik  deviance       aic     aic.c       sic       dof
##  24.81823 -49.63646 -39.63646 -36.10705 -33.95899   5.00000
```

You can see from the results that the α selection parameter is pretty strong. The results also gives the optimal log body sizes for each regime (θ).

An example

Let's look at another example with the seed plants data. More specifically, let's test if the wood density (Wd) of trees with high shade tolerance evolved under a different regime than the wood density of trees with low shade tolerance. To do this, we will have to first reconstruct the ancestral states for the shade tolerance character to "paint" regimes on the tree. And then we will fit the wood density data using different evolutionary models.

To do this, we will use the `ouch` library. Note that it is also possible to fit these models with the `mvMORPH` library.

Infer ancestral states using diversitree

We will start by reconstructing the ancestral states for shade tolerance on the phylogeny to be able to attribute regimes to all branches of the tree.

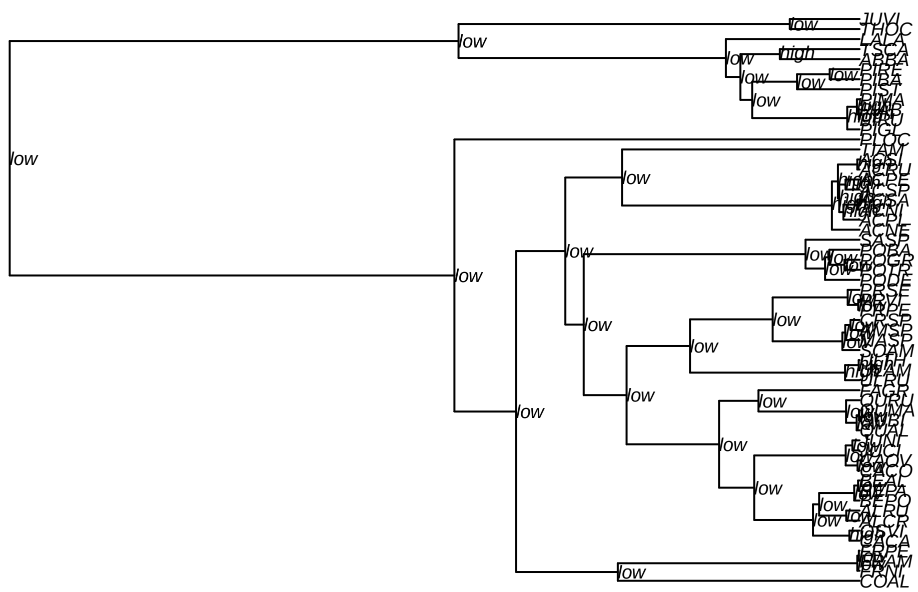
```
require(diversitree)
```

```
## Loading required package: diversitree
## Loading required package: deSolve
## Loading required package: Rcpp
```

```r
ShadeTol <- seedplantsdata$ShadeTolBin
names(ShadeTol) <- rownames(seedplantsdata)
char1 <- as.numeric(ShadeTol)-1
names(char1) <- rownames(seedplantsdata)
# Make mk2 model
lik.mk2 <- make.mk2(seedplantstree, char1)
p <- c(.1, .1)
# Fit mk2 model
fit.mk2 <- find.mle(lik.mk2, p)
coef(fit.mk2)
```

```
##       q01       q10
## 117.08498  95.72686
```

```r
# Export the marginal ancestral reconstruction at the nodes of the tree
st <- t(asr.marginal(lik.mk2,coef(fit.mk2)))
# Get ancestral nodes with maximum likelihood
anc_node<-factor(character(nrow(st)),levels=levels(ShadeTol))
for(i in 1:length(anc_node)) {
  anc_node[i] <- levels(ShadeTol)[st[i,]==max(st[i,])]
  }
# Assign ancestral states to tree
seedplantstree$node.label <- anc_node
plot(seedplantstree, show.node.label=TRUE,cex=0.6)
```



Prepare the data in OUCH format

Then, we need to convert the data into the ouch format. OUCH uses a rather peculiar format for the data and the conversion is not so simple.

```r
# Convert the ape tree into ouch format
tree.ouch <- ape2ouch(seedplantstree)
```

```
tree.ouch <- as(tree.ouch,"data.frame")
# Here is what it looks like:
head(tree.ouch,n=20)
```

```
##    nodes ancestors     times labels
## 1      1      <NA> 0.0000000      2
## 2      2        12 0.9178433      2
## 3      3        10 0.9061369      1
## 4      4         5 0.9647945      2
## 5      5         9 0.9263775      2
## 6      6         7 0.9990455      1
## 7      7         8 0.9968383      1
## 8      8         9 0.9854625      1
## 9      9        10 0.8736274      2
## 10    10        11 0.8598016      2
## 11    11        12 0.8426092      2
## 12    12         1 0.5280915      2
## 13    13        18 0.9972799      1
## 14    14        17 0.9834636      1
## 15    15        16 0.9974958      1
## 16    16        17 0.9809726      1
## 17    17        18 0.9760314      1
## 18    18        19 0.9744850      1
## 19    19        20 0.9673636      1
## 20    20        51 0.7204958      2
```

```
#
# Prepare data:
# We need to make a vector of the regimes. Need to copy the labels
# already in the ouch tree dataframe and the tip values in the same
# order as the taxa are in the ouch tree
regimes <- c(tree.ouch$labels[round(tree.ouch$times,3)!=1],
   as.numeric(ShadeTol[as.vector(tree.ouch$labels[round(tree.ouch$times,3)==1])]))
# Add the regime to the data.frame
tree.ouch$ShadeTol<-as.factor(regimes)
# Add a fake regime to the data.frame for the OU1 model
tree.ouch$ou1<-as.factor(rep(1,length.out=length(regimes)))
# Create a data.frame with the data to analyse (Wood density)
oudata <- data.frame(labels=rownames(seedplantsdata),Wd=seedplantsdata$Wd)
# Merge the data with the ouch tree
oudata <- merge(tree.ouch, oudata, by="labels",all=T)
row.names(oudata)<-oudata$nodes
# Create a new OUCH tree with the final information
outree<-ouchtree(nodes= oudata$nodes, ancestors=oudata$ancestors,
                 times=oudata$times, labels=oudata$labels)
# Here is what it should now look like:
outree
```

```
##     nodes ancestors     times labels
## 3       3        10 0.9061369      1
## 6       6         7 0.9990455      1
## 7       7         8 0.9968383      1
## 8       8         9 0.9854625      1
```

```
## 13     13        18 0.9972799      1
## 14     14        17 0.9834636      1
## 15     15        16 0.9974958      1
## 16     16        17 0.9809726      1
## 17     17        18 0.9760314      1
## 18     18        19 0.9744850      1
## 19     19        20 0.9673636      1
## 31     31        32 0.9989177      1
## 32     32        33 0.9828751      1
## 45     45        46 0.9880003      1
## 1       1      <NA> 0.0000000      2
## 2       2        12 0.9178433      2
## 4       4         5 0.9647945      2
## 5       5         9 0.9263775      2
## 9       9        10 0.8736274      2
## 10     10        11 0.8598016      2
## 11     11        12 0.8426092      2
## 12     12         1 0.5280915      2
## 20     20        51 0.7204958      2
## 21     21        22 0.9821836      2
## 22     22        23 0.9644176      2
## 23     23        24 0.9593343      2
## 24     24        50 0.9364159      2
## 25     25        26 0.9983585      2
## 26     26        30 0.9860000      2
## 27     27        28 0.9892838      2
## 28     28        29 0.9831096      2
## 29     29        30 0.9797546      2
## 30     30        33 0.8977182      2
## 33     33        49 0.8004153      2
## 34     34        35 0.9993238      2
## 35     35        36 0.9965353      2
## 36     36        37 0.9840803      2
## 37     37        48 0.8810330      2
## 38     38        40 0.9915558      2
## 39     39        40 0.9976009      2
## 40     40        47 0.9836285      2
## 41     41        42 0.9982089      2
## 42     42        44 0.9936371      2
## 43     43        44 0.9842903      2
## 44     44        46 0.9523821      2
## 46     46        47 0.9451569      2
## 47     47        48 0.8760336      2
## 48     48        49 0.8346494      2
## 49     49        50 0.7258346      2
## 50     50        51 0.6753576      2
## 51     51        55 0.6538331      2
## 52     52        53 0.9991035      2
## 53     53        54 0.9972820      2
## 54     54        55 0.7154642      2
## 55     55        56 0.5959168      2
## 56     56         1 0.5232898      2
## 113   113         3 1.0000000   ABBA
## 112   112        19 1.0000000   ACNE
```

```
## 111   111        15 1.0000000   ACNI
## 110   110        14 1.0000000   ACPE
## 109   109        16 1.0000000   ACPL
## 108   108        13 1.0000000   ACRU
## 107   107        15 1.0000000   ACSA
## 106   106        13 1.0000000   ACSI
## 105   105        14 1.0000000   ACSP
## 104   104        43 1.0000000   ALCR
## 103   103        43 1.0000000   ALRU
## 102   102        27 1.0000000   AMSP
## 101   101        41 1.0000000   BEAL
## 100   100        41 1.0000000   BEPA
## 99     99        42 1.0000000   BEPO
## 98     98        45 1.0000000   CACA
## 97     97        39 1.0000000   CACO
## 96     96        39 1.0000000   CAOV
## 95     95        54 1.0000000   COAL
## 94     94        27 1.0000000   CRSP
## 93     93        37 1.0000000   FAGR
## 92     92        52 1.0000000   FRAM
## 91     91        53 1.0000000   FRNI
## 90     90        52 1.0000000   FRPE
## 89     89        38 1.0000000   JUCI
## 88     88        38 1.0000000   JUNI
## 87     87         2 1.0000000   JUVI
## 86     86        11 1.0000000   LALA
## 85     85        28 1.0000000   MASP
## 84     84        45 1.0000000   OSVI
## 83     83         6 1.0000000   PIAB
## 82     82         4 1.0000000   PIBA
## 81     81         8 1.0000000   PIGL
## 80     80         7 1.0000000   PIMA
## 79     79         4 1.0000000   PIRE
## 78     78         6 1.0000000   PIRU
## 77     77         5 1.0000000   PIST
## 76     76        56 1.0000000   PLOC
## 75     75        22 1.0000000   POBA
## 74     74        23 1.0000000   PODE
## 73     73        21 1.0000000   POGR
## 72     72        21 1.0000000   POTR
## 71     71        25 1.0000000   PRPE
## 70     70        26 1.0000000   PRSE
## 69     69        25 1.0000000   PRVI
## 68     68        35 1.0000000   QUAL
## 67     67        34 1.0000000   QUBI
## 66     66        34 1.0000000   QUMA
## 65     65        36 1.0000000   QURU
## 64     64        24 1.0000000   SASP
## 63     63        29 1.0000000   SOAM
## 62     62         2 1.0000000   THOC
## 61     61        20 1.0000000   TIAM
## 60     60         3 1.0000000   TSCA
## 59     59        31 1.0000000   ULAM
## 58     58        32 1.0000000   ULRU
```

```
## 57     57        31 1.0000000   ULTH
```

Fit different models

Three different models will be fitted.

1. A BM model with one regime (BM.1)
2. A OU model with one regime (OU.1)
3. A OU model with two regimes (OU.2)

```
#
# Fit the models
#
# BM1
BM.1<-brown(data=oudata["Wd"], tree=outree)
#
# OU1
OU.1 <- hansen(data=oudata["Wd"], tree=outree, oudata["ou1"],
               sqrt.alpha=1,sigma=1,reltol=1e-5)
# Refine the fit
OU.1 <- update(OU.1,method='subplex',reltol=1e-11,
               parscale=c(0.1,0.1),hessian=TRUE)
#
# OU2
OU.2 <- hansen(data=oudata["Wd"], tree=outree, oudata["ShadeTol"],
               sqrt.alpha=1,sigma=1,reltol=1e-5)
# Refine the fit
OU.2 <- update(OU.2,method='subplex',reltol=1e-11,
               parscale=c(0.1,0.1),hessian=TRUE)
```

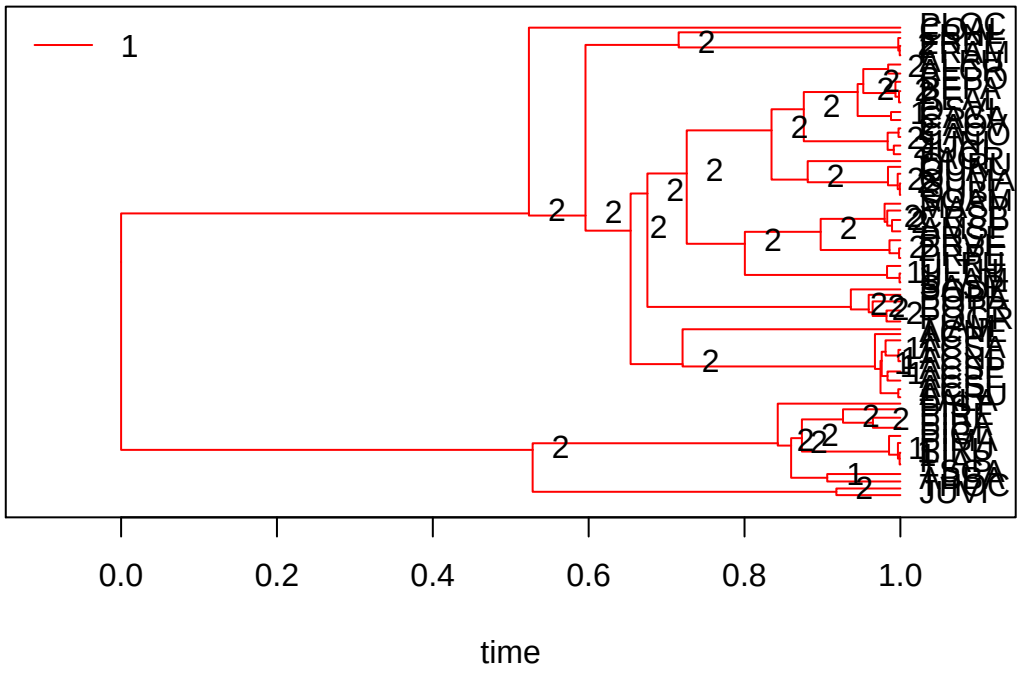
Finally, we can summarize and plot the results

```
#
#Summarize the results
results <- data.frame(model=c("BM.1","OU.1","OU.2"),
  loglik=c(summary(BM.1)$loglik,summary(OU.1)$loglik,summary(OU.2)$loglik),
  AIC=c(summary(BM.1)$aic,summary(OU.1)$aic,summary(OU.2)$aic),
  AICc=c(summary(BM.1)$aic.c,summary(OU.1)$aic.c,summary(OU.2)$aic.c),
  params=c(summary(BM.1)$dof,summary(OU.1)$dof,summary(OU.2)$dof))
results <- results[order(results$AICc),]
results
```

```
##   model   loglik        AIC       AICc params
## 2  OU.1 60.84842 -115.69685 -115.24402      3
## 3  OU.2 60.90610 -113.81220 -113.04297      4
## 1  BM.1 36.22694  -68.45389  -68.23167      2
```

You can see that the model OU.1, that is a OU model with one regimes across the whole tree, has the best fit. We could thus reject the hypothesis that the wood density of species with low shade tolerance is evolving under a different selective regime than for species with high shade tolerance. We can summarize the results and plot the model details.

```
#
# Plot the tree with the best model
plot(OU.1)
```




```
#
# Output model information
summary(OU.1)
```

```
## $call
## hansen(data = data, tree = object, regimes = regimes, sqrt.alpha = sqrt.alpha,
##     sigma = sigma, method = "subplex", hessian = TRUE, reltol = 1e-11,
##     parscale = ..3)
##
## $conv.code
## [1] 0
##
## $optimizer.message
## NULL
##
## $alpha
##          [,1]
## [1,] 42.63124
##
## $sigma.squared
##           [,1]
## [1,] 0.8169377
##
## $optima
## $optima$Wd
##         1
## 0.4528764
```

```
##
##
## $loglik
## [1] 60.84842
##
## $deviance
## [1] -121.6968
##
## $aic
## [1] -115.6968
##
## $aic.c
## [1] -115.244
##
## $sic
## [1] -109.5677
##
## $dof
## [1] 3
```

This conclusion is different from the conclusion reached with the PGLS (lecture 2) where a positive relationship was observed between shade tolerance and wood density. This might be explained by the loss on information involved with the conversion of the shade data into a binary vector.

Recent developments

Since the publication of Butler and King (2004), more developments were made on OU models. In the example above, it was possible to have multiple regimes on a tree, but the α and σ parameters were the same for all regimes. Beaulieu et al. (2012) have relaxed this assumption and have proposed more general Hansen models (another name for the OU model) that can allow either α, σ or both to vary among regimes on a tree. These models are available in the `R` package `ouwie`. Note that many species are necessary if you want to fit the most parameter rich models.

Another addition is to fit multiple traits at oncee. Bartoszek et al. (2012) have described how to do so. In short, it allows fitting a OU model on multiple parameters at once. These functions are available in the `R` packages `mvSLOUCH` and `mvMORPH`.

Incorporating phylogenetic uncertainty

It is very unfrequent to have 100% confidence in the species tree of the group under study. When the relationships are not completely certain, it is generally important to take this uncertainty into account in the statistical tests performed. For instance, you might wonder what the results would give if a given species was placed at another position on the tree for which support is considerable.

Actually, even if you have very poorly supported phylogenies, you might nevertheless be able to reach strong conclusions if you incorporate this uncertainty appropriately into the analyses. For instance, if the results are significant when incorporating phylogenetic uncertainty into account, then you can conclude that the effect are likely true across the sample of tree considered.

We saw in the previous lecture that BayesTraits allows to incorporate phylogenetic uncertainty in a Bayesian framework by giving a list of trees to the program. The same thing can be done with stochastic mapping, which we saw in lecture 3. These are interesting approaches. But in general, it is possible to incorporate phylogenetic uncertainty even when the methods do not integrate this inherently. The trick is to perform the

analyses on a list of trees that represent the phylogenetic uncertainty of the data. These could be random samples from the posterior distribution of a tree search, for example. Then, the conclusions are taken on the set of analyses that were performed.

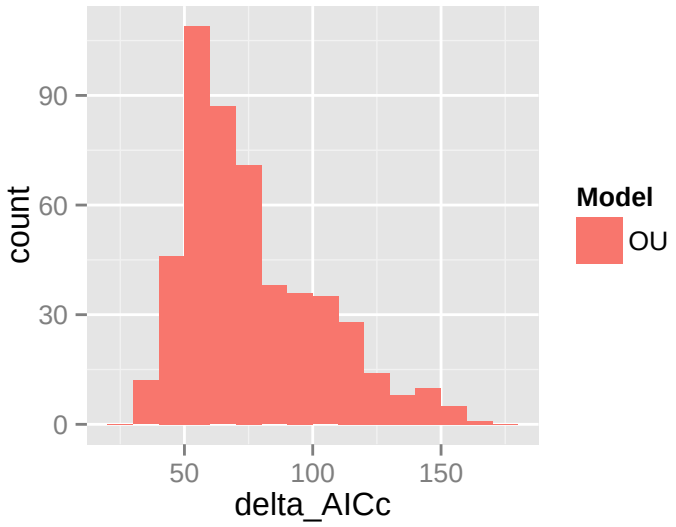
Here, we will evaluate the fit of different evolutionary models on a sample of 500 trees sampled from the posterior distribution of trees obtained from a Bayesian BEAST search on the seedplantsdata.

List of trees are generaly handle as Multiphylo objects in R. These require sligthly different approaches to manipulate them. Let's first prepare the data.

```
# Read trees
pdtrees <- read.nexus("./data/pd_500.trees")
species.to.exclude <- pdtrees[[1]]$tip.label[!(pdtrees[[1]]$tip.label %in%
                                  rownames(seedplantsdata))]
# Exclude species from all trees using the function lapply
pdtrees<-lapply(pdtrees,drop.tip,tip=species.to.exclude)
# Reattribute the list of tree a class multiphylo
class(pdtrees)<-"multiPhylo"
# Assign tip labels to the multiphylo object
attr(pdtrees,"TipLabel") <- pdtrees[[1]]$tip.label
rm(species.to.exclude)
```

Now, we will fit the models BM and OU on all 500 trees for wood density (if you try it, you might want to try with fewer replicates as 500 takes quite some time to run). The important thing is to compare the fit of the models for the same trees, as comparing the fit obtained with different trees is meaningless. The trick is to save the model fits for all trees and then compare the fit of models relative to a reference one (often the BM model). This is commonly done by substrating the AIC value of the reference model (BM) with the AIC of the alternative models for each tree. A positive difference would indicate support for the alternative model because it would mean that the AIC for the alternative model is smaller.

```
# Replicates (here max 500)
replicates = 500
# Prepare vectors to store the results
bm.post <- numeric(replicates)
ou.post <- numeric(replicates)
# Make a loop to evaluate each tree in the list
for (i in 1:replicates) {
  message(cat("\n#########################\n Processing tree",i,
              "\n#########################\n"))
  atree<-pdtrees[[i]]
  fit.bm <- fitContinuous(atree,Wd,model="BM")
  fit.ou <- fitContinuous(atree,Wd,model="OU",bounds=list(alpha=c(0,1000)))
  # Store the AICc in the vectors
  bm.post[i] <- fit.bm$opt$aicc
  ou.post[i] <- fit.ou$opt$aicc
}
# Now, compare the OU model relative to the BM model
comparisons <- data.frame(OU=bm.post-ou.post)
results <- stack(comparisons)
colnames(results) <- c("delta_AICc","Model")
# Plot the results
require(ggplot2)
ggplot(results,aes(x=delta_AICc)) + geom_histogram(aes(fill=Model),binwidth=10)
```



Because the distribution of values is positive, it supports the OU model over the BM (reference) model. In other words, the OU model has a smaller AICc value than the BM model for most of the trees (it can vary from one simulation to the next).

To make a definive conclusion, you need to confirm that the 95% credible interval excludes 0. This 95% CI can be obtained the following way:

```
# Get 95% Credible intervals
apply(comparisons,2,function(x) quantile(x,probs=c(0.025,0.5,0.975)))
```

```
##              OU
## 2.5%   40.35511
## 50%    69.21095
## 97.5% 142.14645
```

As you can see, the 95% credible interval exclude 0 and thus the results are signifcant at the $\alpha = 0.05$ level. Consequently, you can conclude from these results that it is possible to definitively accept the OU model as the best model even when taking into account phylogenetic uncertainty.

References

Bartoszek K., J. Pienaar, P. Mostad, S. Andersson, T.F. Hansen. 2012. A phylogenetic comparative method for studying multivariate adaptation. *Journal of Theoretical Biology.* 314:204–215.

Beaulieu J.M., D.-C. Jhwueng, C. Boettiger, B.C. O'Meara. 2012. Modeling stabilizing selection: expanding the Ornstein–Uhlenbeck model of adaptive evolution. *Evolution.* 66:2369–2383.

Blomberg S.P., T. Garland, A.R. Ives. 2003. Testing for phylogenetic signal in phylogenetic comparative data: behavioral traits are more labile. *Evolution.* 57:717-745.

Butler M.A., A.A. King. 2004. Phylogenetic Comparative Analysis: A Modeling Approach for Adaptive Evolution. *The American Naturalist.* 164:683–695.

Joly S., P.B. Heenan, P.J. Lockhart. 2014. Species radiation by niche shifts in New Zealand's rockcresses (Pachycladon, Brassicaceae). *Systematic Biology.* 63:192–202.