Using transformations of the tree or VCV to infer trait evolutionary history

Göttingen R Workshop 2023, Week 3

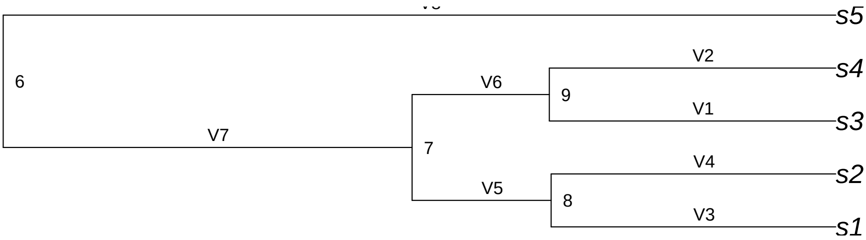
ahipp@mortonarb.org

15 June 2023

Why stretch a tree?

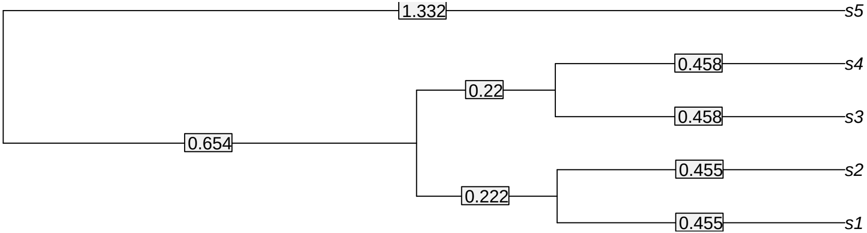
To date, we have focused on the phylogenetic covariance matrix primarily as a statistical tool to address non-independence among tips of a phylogeny, and secondarily as a tool for estimating ancestral states under Brownian motion model.

```
## Loading required package: phytools
## Loading required package: maps
```



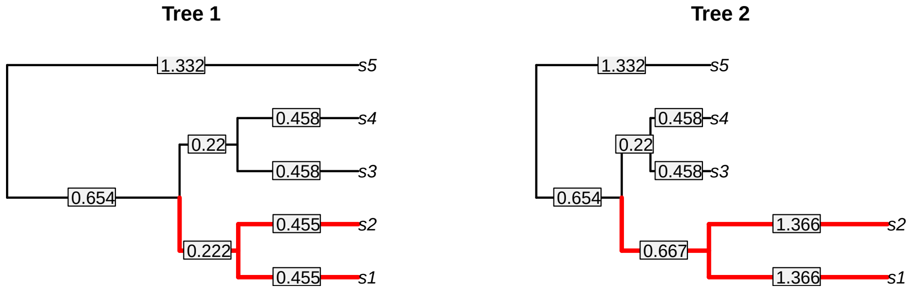
Recall what the covariance matrix is based on: expected variance accumulated along each branch, where branch length predicts the amount of accumulated variance.

$$C = \begin{bmatrix} 1.332 & 0.877 & 0.654 & 0.654 & 0.000 \\ 0.877 & 1.332 & 0.654 & 0.654 & 0.000 \\ 0.654 & 0.654 & 1.332 & 0.874 & 0.000 \\ 0.654 & 0.654 & 0.874 & 1.332 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.332 \end{bmatrix}$$



This means that stretching any single branch implies a model in which traits have had more opportunity to accumulate variance along that branch. A longer branch, in other words, implies two things: (1) a larger expected amount of trait evolution along that branch, and (2) lower covariance between the descendents of that branch an the remainder of the tree.

Let's imagine a scenario in which the stem group of tips S1 and S2 below has undergone a tripling in the rate of trait evolution. This would imply a tree stretching that looks like this:



. . . and compare the covariance matrices. Note that the variances and covariances for S1 and S2 (the first two rows / columns) change, but nothing else does:

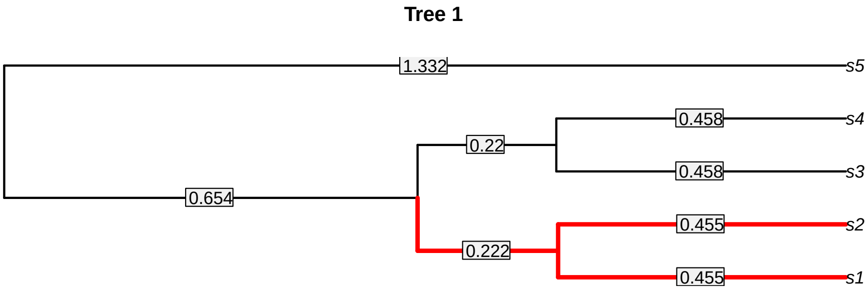
$$C_{Tree1} = \begin{bmatrix} 1.332 & 0.877 & 0.654 & 0.654 & 0.000 \\ 0.877 & 1.332 & 0.654 & 0.654 & 0.000 \\ 0.654 & 0.654 & 1.332 & 0.874 & 0.000 \\ 0.654 & 0.654 & 0.874 & 1.332 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.332 \end{bmatrix}$$

$$C_{Tree2} = \begin{bmatrix} 2.687 & 1.321 & 0.654 & 0.654 & 0.000 \\ 1.321 & 2.687 & 0.654 & 0.654 & 0.000 \\ 0.654 & 0.654 & 1.332 & 0.874 & 0.000 \\ 0.654 & 0.654 & 0.874 & 1.332 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.332 \end{bmatrix}$$

Stretching the tree in this way, it turns out, is equivalent to making the direct transformation of C that Brian O'Meara introduces in his 2006 paper. Let's calculate it as he has done.

In this tree, the rate on the red branches we will denote as σ_A^2 and the rate on the black branches as σ_B^2, echoing O'Meara's figure 3. We'll transform the original C thus:

$$C_{Tree2} = \begin{bmatrix} 0.654\sigma_B^2 + 0.678\sigma_A^2 & 0.654\sigma_B^2 + 0.222\sigma_A^2 & 0.654\sigma_B^2 & 0.654\sigma_B^2 & 0.000\sigma_B^2 \\ 0.654\sigma_B^2 + 0.222\sigma_A^2 & 0.654\sigma_B^2 + 0.678\sigma_A^2 & 0.654\sigma_B^2 & 0.654\sigma_B^2 & 0.000\sigma_B^2 \\ 0.654\sigma_B^2 & 0.654\sigma_B^2 & 1.332\sigma_B^2 & 0.874\sigma_B^2 & 0.000\sigma_B^2 \\ 0.654\sigma_B^2 & 0.654\sigma_B^2 & 0.874\sigma_B^2 & 1.332\sigma_B^2 & 0.000\sigma_B^2 \\ 0.000\sigma_B^2 & 0.000\sigma_B^2 & 0.000\sigma_B^2 & 0.000\sigma_B^2 & 1.332\sigma_B^2 \end{bmatrix}$$



Because we used a scalar of 3 for σ_A^2, we get the following if we multiply out:

$$C_{Tree2} = \begin{bmatrix} 0.654+2.034 & 0.654+0.666 & 0.654 & 0.654 & 0.000 \\ 0.654+0.666 & 0.654+2.034 & 0.654 & 0.654 & 0.000 \\ 0.654 & 0.654 & 1.332 & 0.874 & 0.000 \\ 0.654 & 0.654 & 0.874 & 1.332 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.332 \end{bmatrix}$$

$$= \begin{bmatrix} 2.688 & 1.320 & 0.654 & 0.654 & 0.000 \\ 1.320 & 2.688 & 0.654 & 0.654 & 0.000 \\ 0.654 & 0.654 & 1.332 & 0.874 & 0.000 \\ 0.654 & 0.654 & 0.874 & 1.332 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.332 \end{bmatrix}$$

... which is exactly (aside from rounding error) what we got when we calculated C on tree 2 after stretching the branches. Note that not all transformations of C for a given tree can be recovered by calculating C' on a modified tree, but rate transformations of the type described in O'Meara et al. 2006 can.

As you may recall from last week, these are actually regressions with no slope. Thus, model one has two free parameters:

- $\hat{a}$, the phylogenetic mean
- σ^2, the rate

which we estimated last week using generalized least squares:

$$\hat{a} = (1'C^{-1}1)^{-1}(1'C^{-1}X)$$

$$\sigma^2 = \frac{(x - \hat{a})'C^{-1}(x - \hat{a})}{N}$$

Model two differs only in having only an additional rate parameter. The two models–model 1 with a single rate and phylogenetic mean, model 2 with two rates but a single phylogenetic mean–are based on the same tip data and are thus directly comparable. They are also based on the same tree and thus a transformation of the same C, but this does not enter in as a parameter in the model: only the parameter that we use to scale C enters into model 2 as a parameter.

The relative support for these two models can be estimated as their log-likelihoods:

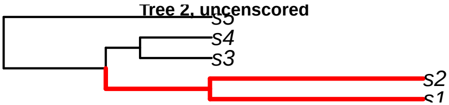
- $ln(\mathcal{L}_{Tree1})$: the log-likelihood of the Brownian motion model for tip data + Tree 1
- $ln(\mathcal{L}_{Tree2})$: the log-likelihood of the Brownian motion model for tip data + Tree 2

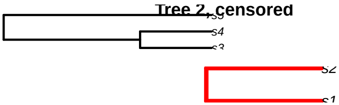
which we calculated two weeks using a function in the `mvtnorm` package:

```
lnL <- dmvnorm(x, rep(a.hat, N), V, log = TRUE) # from mvtnorm package
```

Here:
* `x` are your tip data, arranged in the same order as the rows and columns of C
* `a.hat` is the phylogenetic mean
* `N` is the number of tips on the tree
* `V` $= \sigma^2 C \ldots$ but only for model 1!
* For model 2, `V` is calculated using both σ_A^2 and σ_B^2, as shown a few slides ago.

What we've done here to investigate a model of change in the rate of evolution of a trait is what O'Meara et al. refer to as the "noncensored" test. It requires you to assume that you model which clade(s) are affected by a change in rate as well as where on the branch that rate change occurs. In most cases, I suspect that your model likelihoods will be minimally affected by the latter, but it is an assumption. You can dispense with this assumption by cleaving the tree in two, eliminating the intervening branch, and summing the log likelihoods. This costs you however an additional parameter: model 2 will have four parameters instead of three, as you will now have to calculate $\hat{a}$ as well as σ^2 separately for each tree. You'll implement the censored test in today's (optional) tutorial.


Tree 2, uncenscored


Tree 2, censored

You can compare these models using either a likelihood ratio test:

$$D = -2(ln(\mathcal{L}_{Tree1}) - ln(\mathcal{L}_{Tree2}))$$

where Tree 1 is the simpler of the two models. With large enough sample sizes, D is distributed approximately as χ^2 with d.f. equal to the difference in free parameters between model 1 and model 2. You can also simulate D under a Brownian motion model, which you will also do in today's tutorial.

You can also compare models using AICc weights, as we discussed last week, and use model-averaging to estimate the rates on different branches. Another approach altogether is to use reversible-jump Markov chain Monte Carlo (rjMCMC) to obtain model-averaged rates for every branch on the tree, as introduced in two papers you had in the supplements for this week (Eastman et al. and Venditti et al., both 2011, and apparently independent, as neither references the other).

There are other tree-scaling models out there

There are a few common tree scaling parameters that you may consider. Three were introduced by Pagel in his 1997 and 1999 publications:

- λ, a scalar multiplied by the off-diagonal elements of the covariance matrix. This may be interpreted approximately as the proportion of trait variance that is phylogenetically heritable.

There are other tree-scaling models out there

There are a few common tree scaling parameters that you may consider. Three were introduced by Pagel in his 1997 and 1999 publications:

- λ, a scalar multiplied by the off-diagonal elements of the covariance matrix. This may be interpreted approximately as the proportion of trait variance that is phylogenetically heritable.
- κ, an exponent on individual branches. This may be interpreted as the importance of branch length relative to number of nodes.

There are other tree-scaling models out there

There are a few common tree scaling parameters that you may consider. Three were introduced by Pagel in his 1997 and 1999 publications:

- λ, a scalar multiplied by the off-diagonal elements of the covariance matrix. This may be interpreted approximately as the proportion of trait variance that is phylogenetically heritable.
- κ, an exponent on individual branches. This may be interpreted as the importance of branch length relative to number of nodes.
- δ, an exponent on node depths. This is similar to the ACDC model of Blomberg et al. 2003 or the early-burst model of Harmon et al. 2010; $\delta > 1$ indicates that evolution has accelerated.

There are other tree-scaling models out there

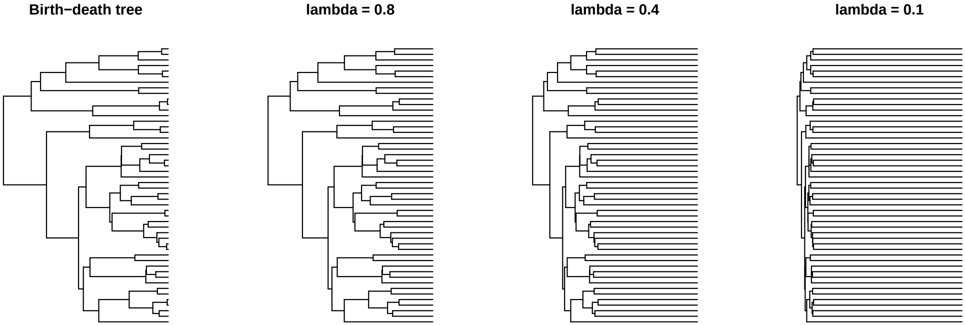
There are a few common tree scaling parameters that you may consider. Three were introduced by Pagel in his 1997 and 1999 publications:

- λ, a scalar multiplied by the off-diagonal elements of the covariance matrix. This may be interpreted approximately as the proportion of trait variance that is phylogenetically heritable.
- κ, an exponent on individual branches. This may be interpreted as the importance of branch length relative to number of nodes.
- δ, an exponent on node depths. This is similar to the ACDC model of Blomberg et al. 2003 or the early-burst model of Harmon et al. 2010; $\delta > 1$ indicates that evolution has accelerated.

- Let's look at each of these in turn on a simulated birth-death tree of 50 tips.

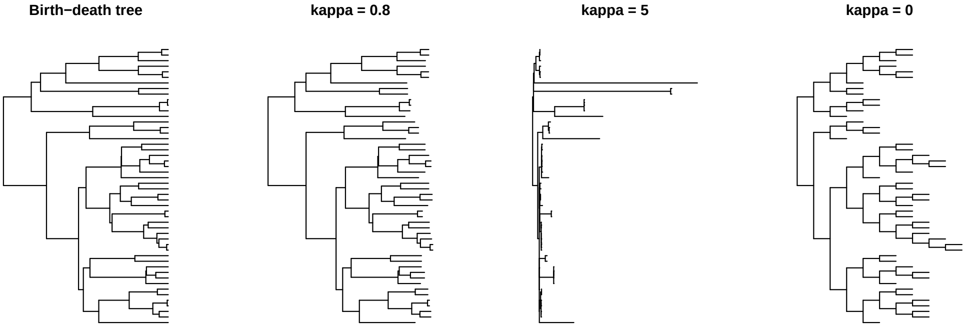
Pagel's λ

Pagel's λ is perhaps the most commonly implemented tree scaling parameter... this is the scalar we have estimated jointly with GLS regression coefficients to relax the assumption of strict Brownian motion. Formulated as a multiplier of the off-diagonal elements of C (modifying covariance while leaving variance untouched), λ can also be implemented as a tree scalar by multiplying all branches by λ and then stretching the tree back out to its original unscaled length:



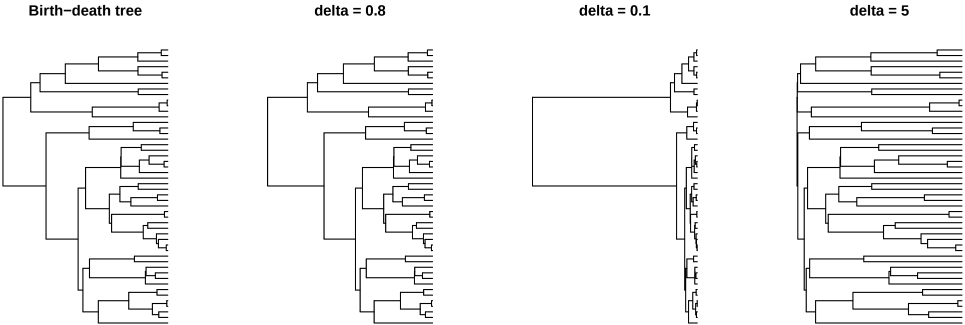
Pagel's κ

Pagel's κ has been utilized in a number of studies to test a speciational model (one in which the amount of character change is associated with cladogenesis rather than time). This makes a dubious assumption about sampling. More generally, κ tests the relationship between time and character evolution, allowing us to ask for example whether trait evolution occurs disproportionately on shorter or longer branches, but again, I don't know how exactly to interpret it if any branches are missing. I don't use it.



Pagel's δ

Think of this as a model of increasing or decreasing rates of trait evolution. δ » 1, for example, may indicate a burst of diversification near the crown of the tree; δ « 1 will approximate λ « 1, high species-specific evolution.



Birth−death tree

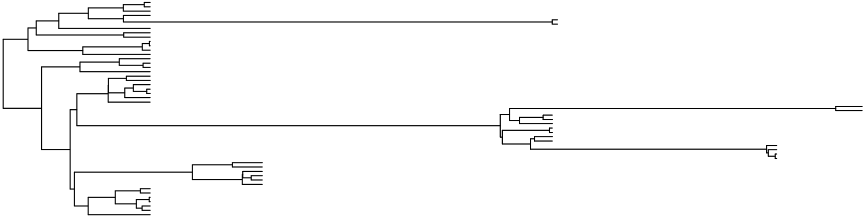
delta = 0.8

delta = 0.1

delta = 5

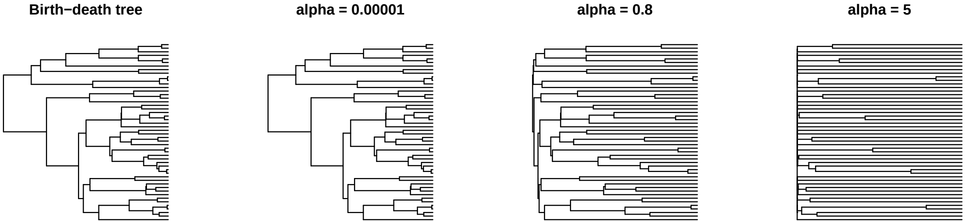
Jumps in phenotypic space

Let's look at two more interesting types of tree scaling. The first is a model in which we allow evolutionary jumps in trait space. This model is illustrated by Duchen et al. (2017), in which they write, "Under the assumption that jump effects are normally distributed, a jump configuration can be seen as simply stretching the branches of the trevolutionee on which they occur, and the likelihood is then given by a multivariate normal distribution with the variance–covariance matrix resulting from the stretched tree" (p. 951). Handy! Such a rescaling might look like this, if we imagine 5 branches in which the rate of evolution has increased 20-fold:



Ornstein-Uhlenbeck models

A second model, which we will dive into in some greater detail in the third lecture, is the Ornstein-Uhlenbeck model of stabilizing selection. Under this model, trait evolution is modeled as according to a Brownian motion process with a pull toward an optimum. The effect of natural selection is to erode phylogenetic structure, similar to what we saw with Pagel's λ... only in this case, the tree scaling has an explicit evolutionary interpretation. It turns out that the O-U model can be expressed in terms of a tree scaling if and only if your phylogeny is ultrametric; if not, the model must be expressed as a transformation of the covariance structure. Here the scaling parameter is α, sometimes referred to as the rate of adaptation:



Birth−death tree

alpha = 0.00001

alpha = 0.8

alpha = 5

Tree-stretching methods apply GLS to the question of how trait evolution has proceeded

Importantly, this is the first time we're selecting among models of evolution for a single trait in which the target of our inquiry is the evolutionary history of that trait. By stretching the tree, modeling a modified expected covariance structure in the data, we evaluate the support for alternative histories of trait evolution. As Blomberg et al. (2003, p. 726) write:

> *Most commonly, the type of branch-length transformation used has been chosen arbitrarily and from a purely statistical perspective. . . . Alternatively, if branch lengths are viewed as entities that may contain important biological information in their own right, then transformations of branch lengths according to some explicit model of evolution and estimation of the parameters in the model may provide useful information about the underlying evolutionary processes.*

This is a step beyond a lot of the PIC literature of the 90s, in which the rationale for tree stretching was strictly statistical.

Testing for phylogenetic signal: Pagel's λ

Everything we have talked about in this class to date is predicated on the assumption that the tree is a meaningful predictor of trait variance. Often in evolutionary biology or ecology you want to test that assumption, asking (1) whether there is significant phylogenetic signal in your data and (2) what the particular direction of phylogenetic signal is (i.e., are close relatives more similar or less similar than expected if they were unrelated).

We have already seen how λ model can be used to test for phylogenetic signal. Let's do a worked example. First, let's generate two datasets, one with high phylogenetic signal (Brownian motion) and one with low phylogenetic signal ($\lambda = 0.2$). We'll also create a matrix to hold the output.

```
options(warn = -1)
dat3 <- list(low = sim.char(lambdaTree(tr3, 0.2), par = 1, model = "BM")[,,1],
              high = sim.char(tr3, par = 1, model = "BM")[,,1])
out <- matrix(NA, nrow = 4, ncol = 4,
              dimnames = list(c('BM.high', 'lambda.high', 'BM.low', 'lambda.low')
                              c('lambda', 'sigsq', 'lnL', 'aicc')))
```

The λ model is clearly favored on the low-phylogenetic signal dataset:

```
for(model in c('lambda', 'BM')) {
  for(signal in c('low', 'high')) {
    temp <- fitContinuous(tr3, dat3[[signal]], model = model)
    out[paste(model, signal, sep = '.'), c('sigsq', 'lnL', 'aicc')] <-
      unlist(temp$opt[c('sigsq', 'lnL', 'aicc')])
    if(model == 'lambda') out[paste(model, signal, sep = '.'), 'lambda'] <- temp$
  }}
print(out)
```

```
##                   lambda      sigsq        lnL     aicc
## BM.high               NA  1.0796904  -70.49108 145.2375
## lambda.high  9.912393e-01  0.9352119  -69.51856 145.5589
## BM.low                NA 14.8487040 -136.02204 276.2994
## lambda.low  2.928153e-217  0.9913395 -104.48337 215.4885
```

It always worth asking whether the signal is significantly different from 0:

```
fitContinuous(lambdaTree(tr3, 0), dat3[['low']], model = 'BM')
```

I leave this to you to try out.