Continuous Data

Testing competing models is a common task in analyzing data. Today I am going to give you a data set and you are going to write your own code to test whether a Brownian motion or an Ornstein–Uhlenbeck model provides a better fit for the observed data. The data that we will use is the number of chromosomes that are present in taxa from the beetle suborder Adephaga.

To performs today's analysis you will need to install the R package Geiger.

Once you have Geiger installed:
1) Download the data from the website:
Adephaga tree
Adephaga chromosome numbers

2) Write your own code to test between our two models. Below is pseudocode to help guide you through the analysis.

```
Require geiger
Read in the tree
Read in the data
Fit a BM model
Fit an OU model
Calculate the likelihood ratio
Calculate the p value
```

These are the only functions that you will need:
require( )
read.nexus( )
read.csv( )
names( )
fitContinuous( )
pchisq( )

Remember if you need to figure out how to use a function just type help(function), where function is the name of the function.

For the really ambitious people:

If you found the OU model to be a better fit; see if you can figure out how to calculate Z bar which is the “adaptive peak” in the OU model. This is a bit harder than you might expect.

1) Convert your phylogeny into vcv matrix using the ape function “vcv.phylo”.
2) Use the function “match” on the vcv matrix and your tip data.
3) Use that result to pull out the tip data of interest.
4) Then use the internal Geiger function phylogMean. This is a function that takes a vcv matrix and your subsetted data and returns the phylogenetic mean or Z bar. You access these internal function like this:

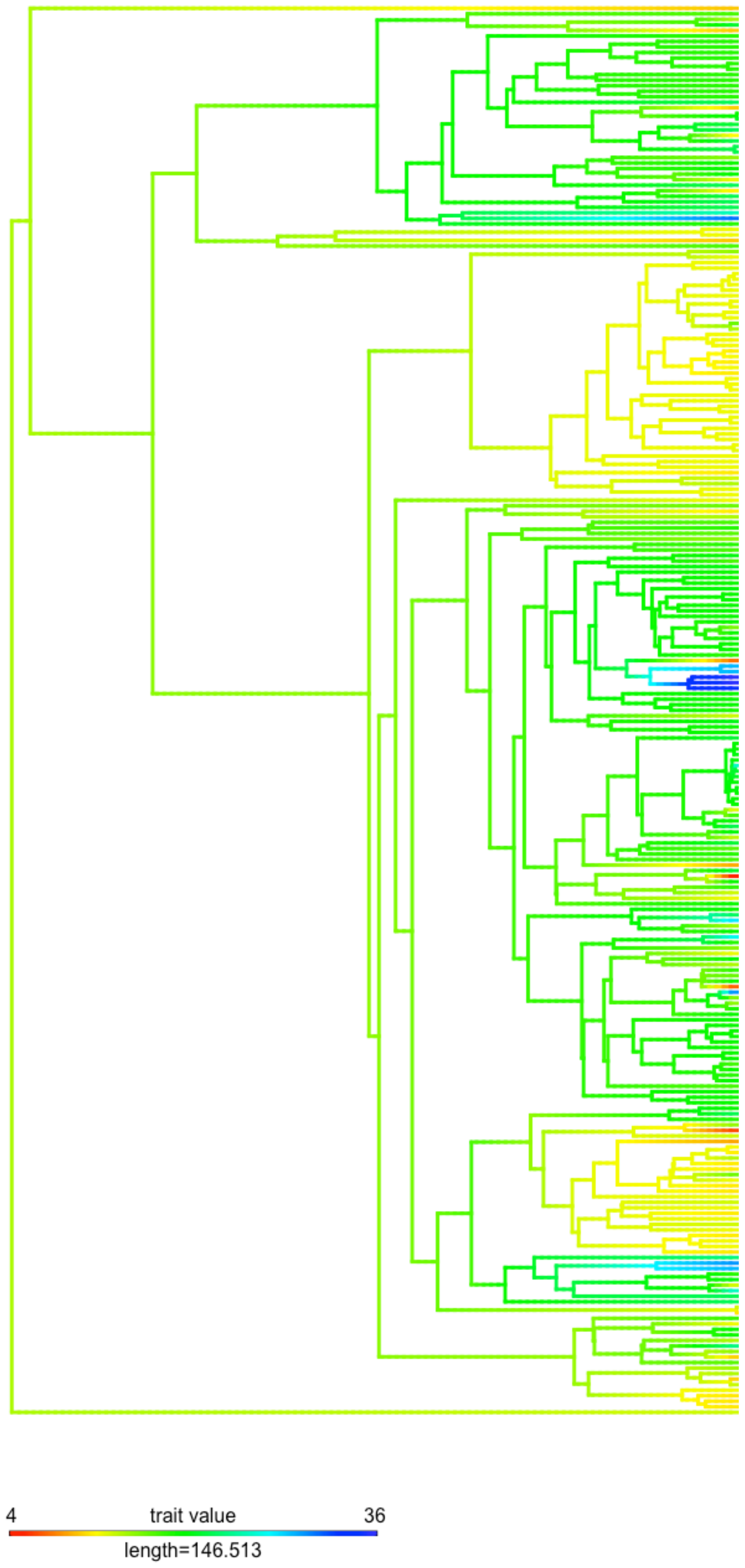
   ```
   geiger:::phylogMean(vcvmatrix,subsetted_tip_data)
   ```

   If you get this to work you should get the answer 14.

or

Try considering how you could display this type of data. I think that my favorite way is a new function called “contMap” which is in the package Phytools

This is the kind of output you can produce with contMap


trait value
length=146.513