

February 8, 2007.

## **I. Continuous characters – ancestral states**

Many traits of interest are measured on **continuous or metric scales** – size and shape, physiological rates, etc. Continuous traits are often useful for species identification and taxonomic descriptions; historically, they were also used in phylogenetic analysis through the use of clustering algorithms that can group taxa based on multivariate phenetic similarity. With the advent of cladistics, reconstruction has shifted entirely to discrete trait analysis.

In principal, ordinal traits take on discrete states while continuous traits are real numbers. In reality, there is a continuum from ordered discrete traits with only a few states (1,2,3,4,5 petals) to those with enough states that we may treat them as continuous (dozens to hundreds). It is also possible to 'discretize' traits if breaks are observed in the distributions for different species.

Distributions of continuous traits may take on any arbitrary shape, but certain distributions occur repeatedly, possibly reflecting underlying 'natural' processes.

Normal distribution: sum of many small additive effects

Exponential distribution: product of many small multiplicative effects

Poisson distribution: frequencies of rare events in discrete intervals

etc.

This becomes important when we consider whether ancestral reconstruction of continuous traits should reflect an underlying evolutionary model of the process that describes or dictates trait evolution. Traits may be transformed to better meet an appropriate distribution (e.g. log-transform).

## **I.B. Parsimony methods for ancestral states**

1. Linear or Wagner Parsimony: minimize the sum of absolute or linear changes along each branch (analogous to normal parsimony for discrete traits). The ancestral value at each node will be the *median* of the three values around it (two child nodes, one parent node). The root is a special case, where it will be the median of the two child nodes, as there is no parent node.

2. Squared Change Parsimony: minimize the sum of squared changes along each branch. The ancestral value at each node will be the *mean* of the three values around it (two child nodes, one parent node). Weighted SCP can also be calculated, where the change along each branch is divided by its branch length before summing – a given change on a long branch is penalized less.

### I.C. Brownian motion and maximum likelihood ancestral states

Brownian motion (BM) is the term for a random walk in a continuous valued variable. If a trait was determined by multiple, independent additive factors of small effect, and if each factor was mutating or changing at random (e.g., by drift), then the character change would constitute BM. Brownian motion is the starting point for discussions of continuous character evolution, for its simplicity and its close ties to parametric statistics based on normal distributions.

In Brownian motion the size of each step is drawn from a normal distribution with mean = 0 (no trend) and variance  $s^2$  (= standard deviation  $s$ ), where each step is a unit of time. When we consider Brownian motion as a process, this variance is viewed as a rate parameter,  $\beta$ . One of the fundamental principles of probability theory is that the variance of the sum of two random processes is the sum of their variances. In other words, if the variance of a brownian motion process is  $\beta$  after one time step, it will be  $\beta + \beta = 2\beta$  after two time steps. So the variance increases linearly with time.

If you apply BM to a large number of independent random walks, with time =  $t$  along each walk, then you can probably see that the variance of the resulting values at the tips of the walks will be  $t\beta$ . What is less intuitive for most of us (if you are not used to statistical thinking) is that a single value resulting from a random walk also has a variance that refers to the underlying (and unobserved) distribution from which that value has been drawn.

Whether you know it or not, we all solve a maximum likelihood (ML) problem on a daily basis when we calculate the mean for a set of numbers. The mean of  $X$  (a set of numbers) is the sum of  $X$  divided by  $N$ , the number of values in  $X$ , right? Yes. Alternatively, the mean of  $X$  is the ML solution for the starting point of  $N$  random walks that end with values  $X$ . This can be solved from the following steps:

- 1) From the central limit theorem, we know that random walks generate values drawn from a normal distribution, with mean  $u$  and variance  $s^2$ .
- 2) The probability of each value of  $X$ , under a normal distribution is:

$$P(x) = \frac{\exp\left[-(x-u)^2/2s^2\right]}{s\sqrt{2\pi}}$$

- 3) The ML solution for  $u$  and  $s^2$  are the values that maximize their cumulative probability over all values of  $x$ , and the cumulative probability is the product of the individual probabilities. A product of a series of values for  $P(x)$  looks pretty nasty, so instead let's take the sum of the log of  $P(x)$  (because the log of a product is the sum of the logs):

$$\log(P(x)) = \frac{-(x-u)^2}{2s^2} - \log(s\sqrt{2\pi})$$

- 4) To maximize this, we can ignore the denominator ( $2s^2$ ) and the second term, since they will be constants. And if we are maximizing the sum of negative terms, we can instead minimize the sum of the positive terms. So the mean of  $X$  is that value which minimizes:

$$\sum (x-u)^2$$

Look familiar!? It's the sum of squares of  $X$ . And now there's some magic, and the sum of squares is also how we calculate  $s^2$ , but we won't try and derive that as a ML problem here.

5) Try this R script to solve for the mean of a set of numbers by finding the minimum of the sum of squares:

```
## enter a set of numbers in xx
xx = c(1,2,4,5)

## create a sequence of candidate values for the mean of xx
xu = seq(1,5,by=0.1)

## create a variable to hold the sum of squares
lxu = rep(NA,length(xu))

## loop through xu and calculate the likelihood score for each candidate value
as sum of squared deviations of xx from xu
for (i in 1:length(xu)) lxu[i] = sum((xx-xu[i])^2)

## plot the likelihood score vs. xu
plot(xu,lxu)

## find the minimum; print it out and compare it to the mean of xx
minxu = xu[which(lxu==min(lxu))]
print(minxu)print(mean(xx))
```

**Ancestral states:** Now we apply the same principles to solve for ancestral states under ML and Brownian motion, treating each ancestral state as the ML solution for a local BM process derived from that node, and finding the set of ancestral states that maximizes the likelihood over the entire tree. The branch lengths are key now, as the overall  $s^2$  value at each node is proportional to the BM rate parameter times the branch lengths.

**Local likelihood solution:** The ML reconstruction of ancestral states can be calculated as a local ML solution, based only on the trait data of tips descended from a node. This amounts to a recursive averaging process down the tree, except that at each node you calculate the weighted average of the two daughter nodes, weighted by the inverse of the square root of the branch length (more on that later)! (The everyday mean that we calculate for data sets is also a maximum likelihood solution for that value that minimizes the squared changes between the mean and the data points, i.e. minimizes the variance around the mean).

**Global likelihood solution:** The global ML solution uses information from the entire tree, including descendent taxa and all sister clades at and above (towards the root) a given node. Since ML is minimizing the sum of squared changes, the ancestral states found under the global likelihood solution are equivalent to the results under squared change parsimony. When a solution is found, a BM rate parameter is also calculated, based on the variance of the normal distribution per unit branch length (see Schluter et al. 1997, top right of p. 1701). *The big difference between parsimony and ML is that ML techniques can provide confidence intervals on the ancestors!* Given the BM rate parameter, we can calculate a distribution of ancestral states (i.e. support limits) that are consistent with the observed data. The rather troubling result of much work in this area is that sometimes the error bars exceed the range of values observed in the terminal taxa (in other words the ancestor could be anywhere in the range of present-day trait values, or even outside it!). See Schluter et al. 1997 Fig. 7 and Fig. 8 (error bars in Fig. 8 don't seem to show up in the pdf - check the printed journal).