

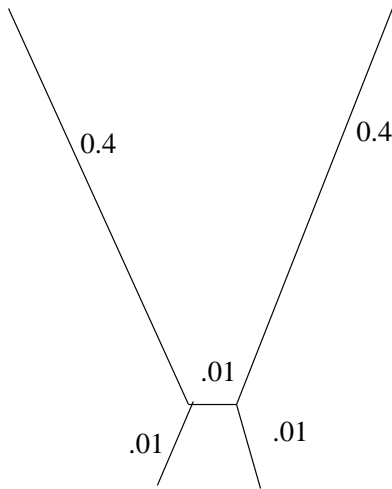
ALGORITHMS FOR COMPUTATIONAL BIOLOGY - CLASS NOTES

SRINIVAS NEDUNURI

25th April 2007

Maximum Parsimony and Maximum Compatibility are not Statistically Consistent

Both have problems in the Felsenstein Zone, shown below:



The basic approach to demonstrating this is as follows:

- Pick a model and a set of sequences, and demonstrate which tree MP is going to produce
- For the same model and set of sequences, demonstrate which tree is expected the most often as the sequence length increases, given the above model tree
- Show they are not the same

Parsimony assumes that nature is conservative, and that a tree with the least number of changes is a tree that best explains the given collection of taxa.

We did a “handwaving” proof in class. First recall the definition of statistical consistency:

A method is statistically consistent under a model M if for all model trees (T, λ) in M , given random sequences S generated at the leaves of (T, λ) , the probability that $M(S) = T$ goes to 1 as $|S|$ increases.

Consider now the simplest Cavender-Farris model and 4 taxa, A, B, C, and D. We can think of the assignment of states to a given character in all the taxons as a bit pattern. The 16 possible patterns are shown below, classified according to whether they are all 0s, no 0s, one 0, three 0s, or two 0s. (We could also consider the completely symmetric case of all 1s, but it doesn't change the argument)

											p	q	r	p'	q'	r'
A	0	1	0	0	0	1	0	1	1	1	0	0	0	1	1	1
B	0	1	0	0	1	0	1	0	1	1	0	1	1	1	0	0
C	0	1	0	1	0	0	1	1	0	1	1	0	1	0	1	0
D	0	1	1	0	0	0	1	1	1	0	1	1	0	0	0	1

Now suppose we are trying to construct the unrooted binary (quartet) tree that matches these 4 taxa. There is only one tree topology, but 3 possible configurations (because each node can be a sibling of one other, e.g. AB, AC, and AD)

T1 = AB | CD,

T2 = AC | BD,

T3 = AD | BC

Now consider a model tree shown above. It is easy to see that the case of all 0s, no 0s, one 0, and 3 0s are all going to lead to equally parsimonious trees (e.g. in the case of all 0s or 3 0s, both internal nodes are also 0, and 1 in the other two cases). Such patterns are called *parsimony uninformative*. So we can focus on the case of 2 0s: Consider pattern p. It would have the following labeling on each of the 3 trees:

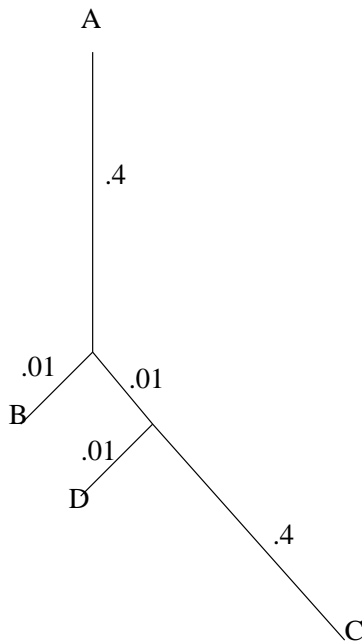
p on T1 is 00|11,

p on T2 is 01|01,

p on T3 is 01|10

It is easy to see that p on T1 has the lowest cost (similar arguments apply to q on T2 and r on T3). This is also the pattern in which A and B have the same state.

Now consider which tree is expected to occur the most often given the model tree above. We can determine this by calculating the likelihood of each tree, ie. the probability of a given leaf assignment of the taxa. By the independence assumption in C-F we can consider each character separately. Consider again p on T1. By rooting the tree at one of the leaves, e.g. A, as shown below



we can calculate its probability. It is:

$$\begin{aligned} \frac{1}{2} \times (.6 \times .99 \times .99 \times .01 \times .4 \\ + .6 \times .99 \times .01 \times .99 \times .6 \\ + .4 \times .01 \times .01 \times .01 \times .4 \\ + .4 \times .01 \times .99 \times .99 \times .6) \end{aligned}$$

Similarly, we can calculate the probability of each of the other trees T2 and T3. When we do this, we find that p and p' are the most likely patterns on tree T2, not T1. As the sequence length increases, more and more sites are likely to be p/p' on T2 (or equivalently q/q' on T1 or r/r' on T3) Assuming the more likely tree is the correct tree (My comment: is this some well known statistical result?), then MP will not pick the correct tree.

Intuitively, given the model tree above, as the sequence length increases, A and B have a very small probability of differing from each other. Therefore, the tree configuration in which A and B are siblings is also the most parsimonious and is therefore the configuration to be returned by Max Parsimony. It is not however, the correct model tree, which is AD | BC. Therefore Max Parsimony is not Statistically Consistent

Maximum Compatibility. A character encoding is called *compatible* if there are no back mutations or parallel evolution of that character. (The presence of either case is called homoplasy). Mathematically, this is true if we can label a tree with states of that character and all subtrees are indentially labelled (all nodes in that subtree have the same state). Another way of stating this is that there is at most one subtree labeled with a given state. But how can we tell if a character is compatible or not without having to go and label all the internal nodes? Because there is a theorem that comes to our rescue:

Theorem. *A site c is compatible on tree T iff its length on T is $r_c - 1$ where r_c is the number of states of c*

(The length of a character on a tree is the number of times it changes state)

Two characters are said to be compatible iff there exists a tree on which each is compatible.

Maximum Compatibility is a method that returns the tree on which the maximum number of characters are compatible.

It turns out Maximum Compatibility is not statistically consistent either. The explanation is very similar to that for Maximum Parsimony above

□