Introduction to Trees

Tandy Warnow

December 28, 2016

Introduction to Trees

Tandy Warnow

Clades of a rooted tree

- Every node v in a leaf-labelled rooted tree defines a subset of the leafset that is below v, and is referred to as a **clade** or **cluster**.
- The set of all clades for the rooted tree T is denoted by $Clades(T)$.
- Two rooted trees T and T' are considered identical if $Clades(T) = Clades(T')$.

Different ways of drawing rooted trees

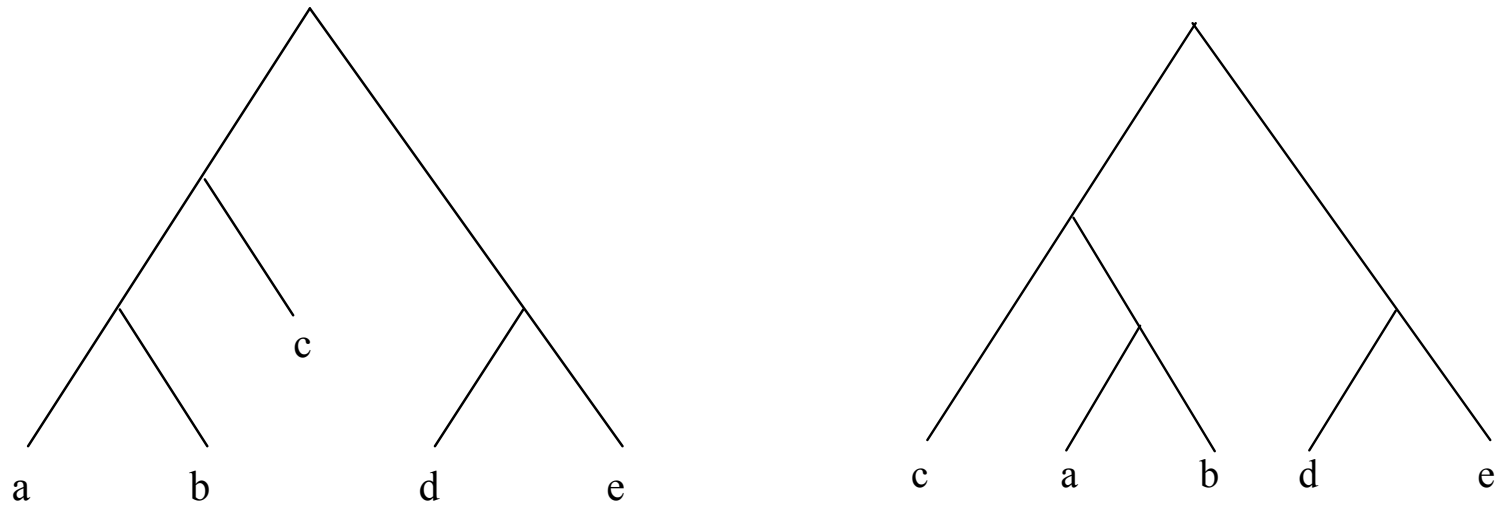


Figure: Two drawings of the same rooted tree, given by (((a,b),c),(d,e)). The trees are considered identical because they have the same set of clades; note that branch length does not matter.

A rooted tree

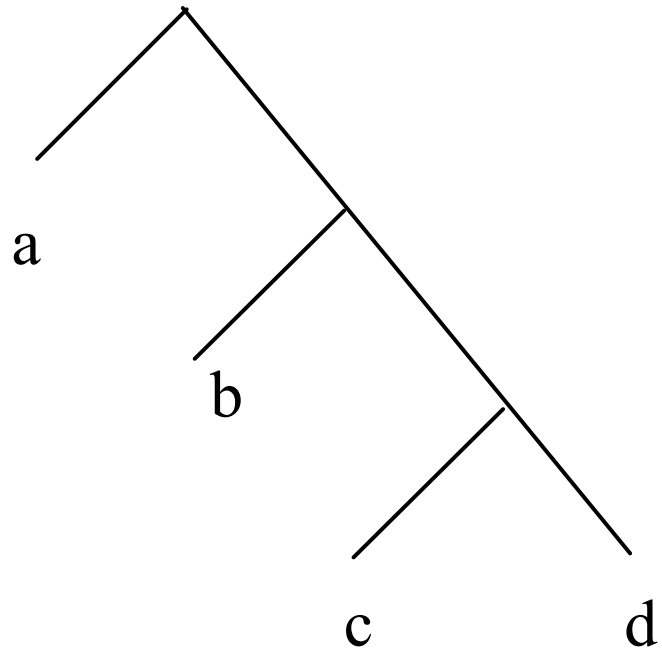


Figure: Rooted tree (a,(b,(c,d))).

Unrooted version of the previous tree

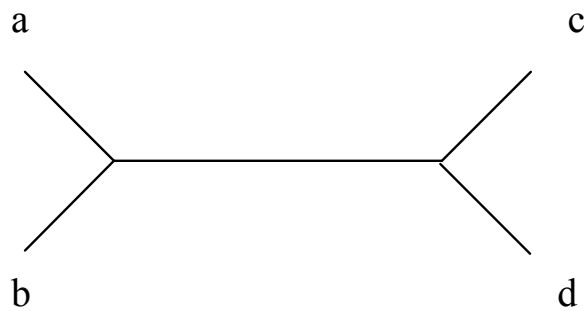


Figure: Unrooted tree ((a,b), (c,d))

Why unrooted trees?

- Estimated trees are almost always unrooted.
- The reason is that the statistical models for sequence evolution are *time-reversible*.
- You need to be able to switch between rooted and unrooted versions of a tree.
- To root a tree, you can:
  - Pick up the tree from the "outgroup",
  - Use the midpoint of the longest path, or
  - Find the root that best fits some statistical model of sequence evolution that is not time-reversible

Rooting using an outgroup taxon

Figure: Tree on some mammals with fly as the outgroup

Newick Notation

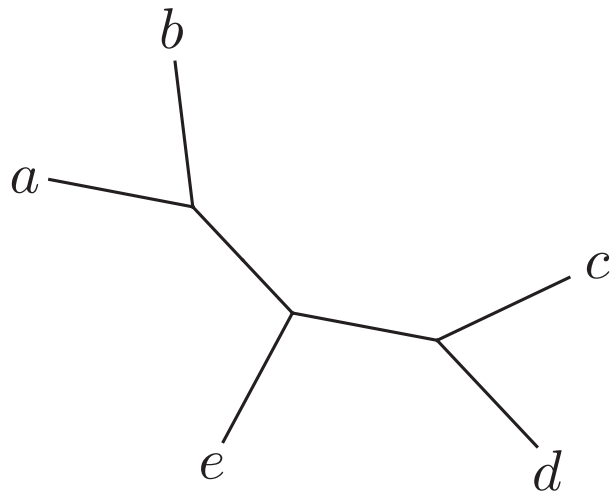
- Newick notation can be used for rooted trees, and for their unrooted equivalent.
- You can't tell whether the tree is intended to be rooted or unrooted unless you are told.
- Read sections 2.2.1 and 2.3.1 from Computational Phylogenetics for more about Newick Notation.

Exercises for Newick Notation

1. Draw the rooted and unrooted trees for each Newick string:
   - (u,(v,(x,y)))
   - (u,v,w,((x,y),(a,b)))
2. Draw the unrooted trees given by
   - (a,(b,(c,d)))
   - ((a,b),(c,d))
   - (d,(c,(a,b)))
   - (b,(a,(c,d)))
   - (b,(d,(a,c)))
   - (b,((d,a),c))
3. Draw any unrooted tree on five leaves and give at least five of its Newick strings
4. How many Newick strings are there for a binary tree on four leaves?

Bipartitions (also called Splits) of a tree

- Every edge e in a leaf-labelled tree defines a bipartition π_e of the leafset.
- The set of all bipartitions for the tree T is denoted by $C(T)$, and is called the *bipartition encoding* or *split encoding* of T.
- Two unrooted trees T and T' are considered identical if $C(T) = C(T')$.
- The trivial bipartitions are those that appear in every tree on the same leafset - these are for the edges incident with leaves.
- To describe a tree, it suffices to write down the non-trivial bipartitions $C_I(T)$, associated with the *internal edges*.



(a) Unrooted tree T

$\{a\}|\{b,c,d,e\}$
$\{b\}|\{a,c,d,e\}$
$\{c\}|\{a,b,d,e\}$
$\{d\}|\{a,b,c,e\}$
$\{e\}|\{a,b,c,d\}$
$\{a,b\}|\{c,d,e\}$
$\{a,b,e\}|\{c,d\}$

(b) Split encoding of T

Figure: Figure 5.2 from Huson et al. (2010). Unrooted tree T and its set $C(T)$ of bipartitions.

Inferring trees from clocklike distances

Inferring Clocklike Evolution

If |S|=2, make the taxa in S siblings

If |S| > 2 then

find pair x,y of closest taxa;

Recurse on S\{y}

Insert y as sibling to x

Return tree

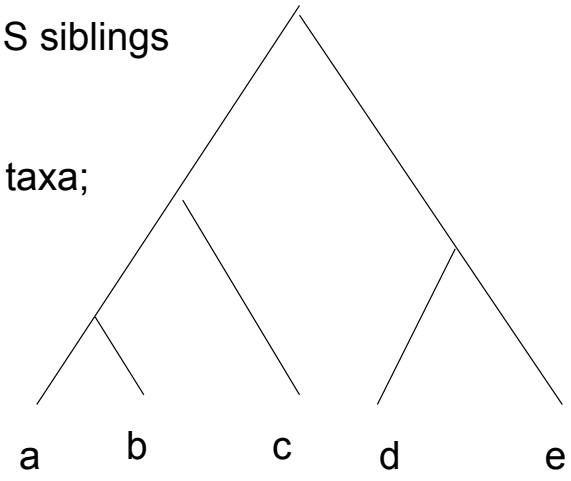


Figure: **Constructing trees when evolution is clocklike.** Branch lengths are drawn proportional to the expected number of changes. When evolution is clocklike, simple techniques will reconstruct the model tree with probability that converges to 1 as the sequence length increases.

Inferring trees from non-clocklike distances

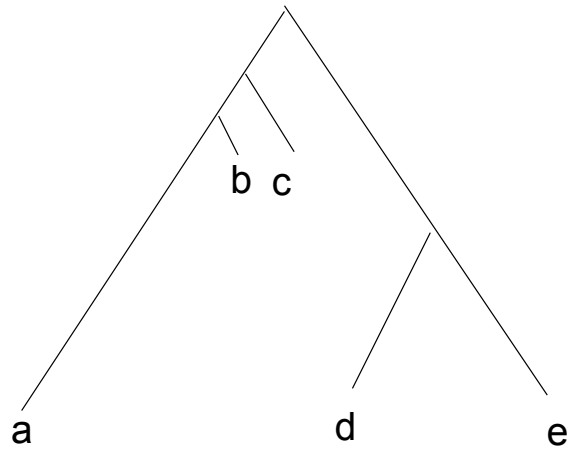
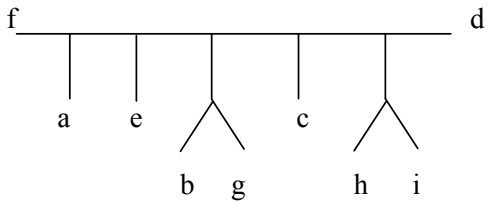


Figure: **Constructing evolutionary trees when evolution is not clocklike.** Branch lengths are drawn proportionally to the expected number of changes of a random site (i.e., position in the sequence alignment).

Homeomorphic subtrees



T

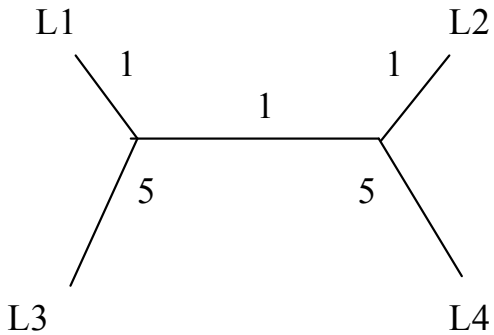
T | {a, b, c, d}

Figure: Tree T and its homeomorphic subtree on a, b, c, d denoted by $T|\{a, b, c, d\}$. We also write this quartet tree as $ab|cd$.

Inferring T from its set $Q(T)$ of quartet trees

- A quartet tree in a tree T is a four-leaf homeomorhic subtree, and can be written as $ab|cd$.
- The set of all quartet trees in a tree T is denoted $Q(T)$
- T can be constructed from $Q(T)$ in polynomial time using a recursive algorithm.
- How can we construct $Q(T)$?

Additive distances and edge-weighted trees



	L2	L3	L4
L1	3	6	7
L2		7	6
L3			11
L4			

Figure: Additive matrix and its edge-weighted tree

The Four Point Condition

Theorem: Let A be an additive matrix corresponding to an binary tree T with positive branch lengths $w : E \longrightarrow R^+$. Then for all four indices $ijkl$, the smallest of the three pairwise sums given below is strictly smaller than the other two pairwise sums, and the other two are equal. Furthermore, if $A_{ij} + A_{kl}$ is the smallest of the three, then T induces quartet tree $ij|kl$.

- $A_{ij} + A_{kl}$
- $A_{ik} + A_{jl}$
- $A_{il} + A_{jk}$

The Four Point Method

- A dissimilarity matrix is symmetric and zero on the diagonal, but need not satisfy the triangle inequality.
- Given a dissimilarity matrix A, to compute a tree on $ijkl$, return $ij|kl$ if $A_{ij} + A_{kl}$ is the unique smallest pairwise sum, and otherwise return *Fail*.

Theorem: The Four Point Method returns $T|\{i, j, k, l\}$ for all sets $\{i, j, k, l\}$ of four leaves and binary trees T when the input matrix A is additive and corresponds to a positive edge-weighting of T.

Naive Quartets Method

We are given a dissimilarity matrix A, and we wish to contruct a tree T.

- Use the Four Point Method to construct each quartet tree. If the Four Point Method ever returns *Fail*, then also return *Fail*.
- Assemble a tree T from the set of quartet trees, if they are compatible; else return *Fail.*

Theorem: The Naive Quartets Method returns the binary tree T whenever the matrix A corresponds to T with positive edge weights.

Analysis of the Naive Quartets Method

We know the Naive Quartets Method is correct if the input is an additive matrix A corresponding to a binary tree with positive edge weights.

1. What happens if some edge weights can be zero?
2. What happens if A corresponds to a non-binary tree (i.e., a tree that can have nodes with more than three neighbors) with positive edge weights?
3. What happens if A is a noisy version of an additive matrix, so that each entry is at most δ from its true distance?
4. Does A need to be *ultrametric* (i.e., correspond to clocklike distances)?
5. What happens if A has some missing entries?
6. What is the running time of the Naive Quartets Method?
7. Would you ever want to use this method in practice? Why or why not?