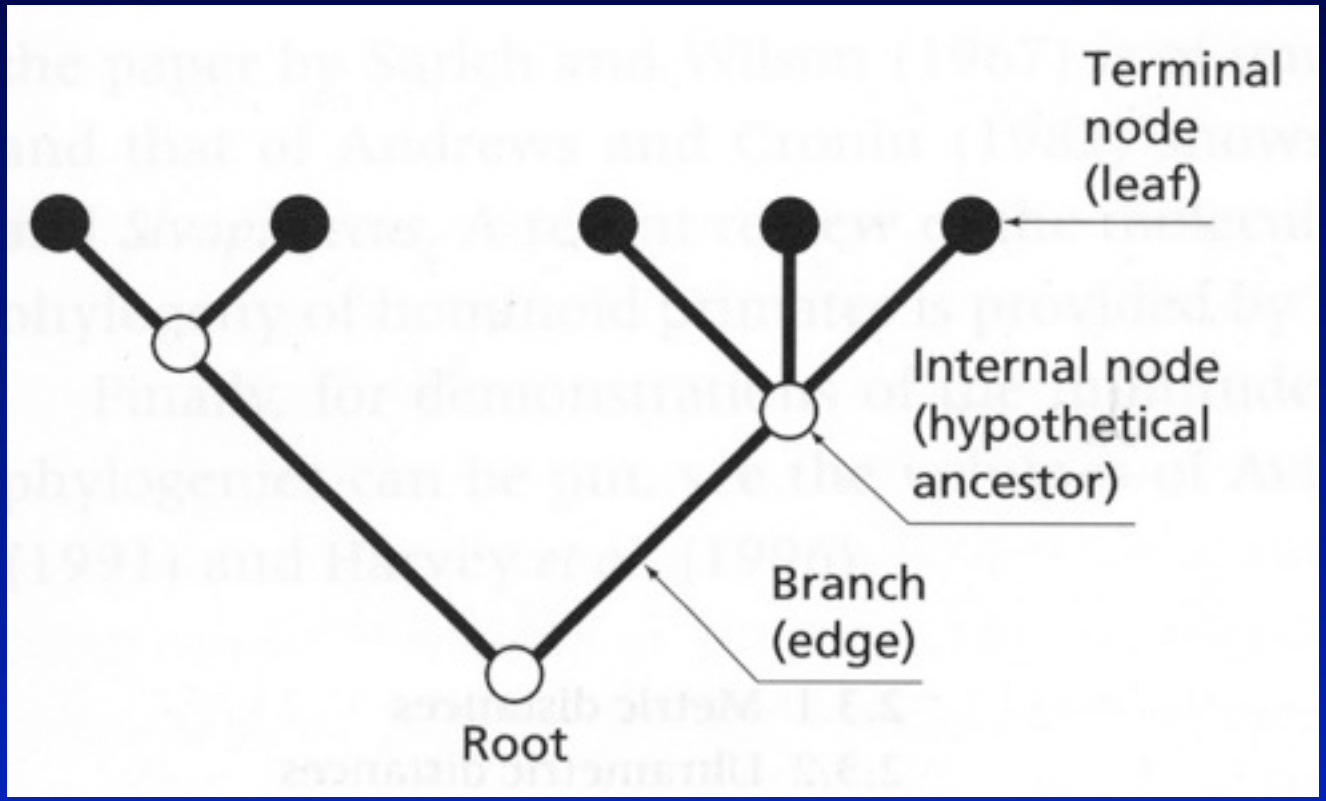
CENTER FOR BIOLOGICAL SEQUENCE ANALYSIS

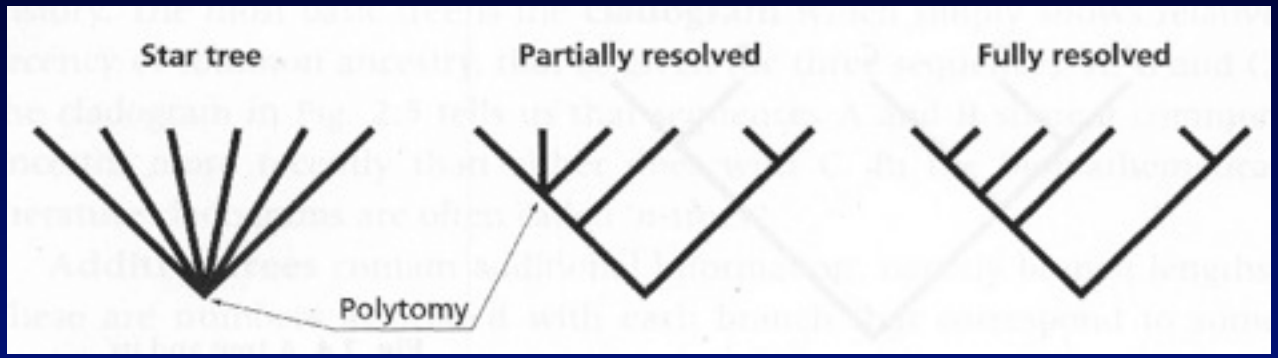
Phylogenetic Reconstruction: Distance Matrix Methods

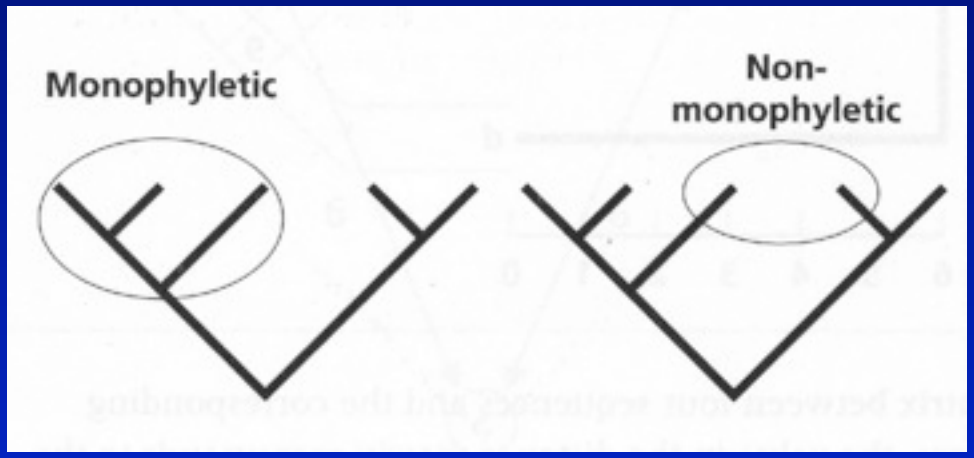
Anders Gorm Pedersen
Molecular Evolution Group
Center for Biological Sequence Analysis
Technical University of Denmark
gorm@cbs.dtu.dk

Trees: terminology

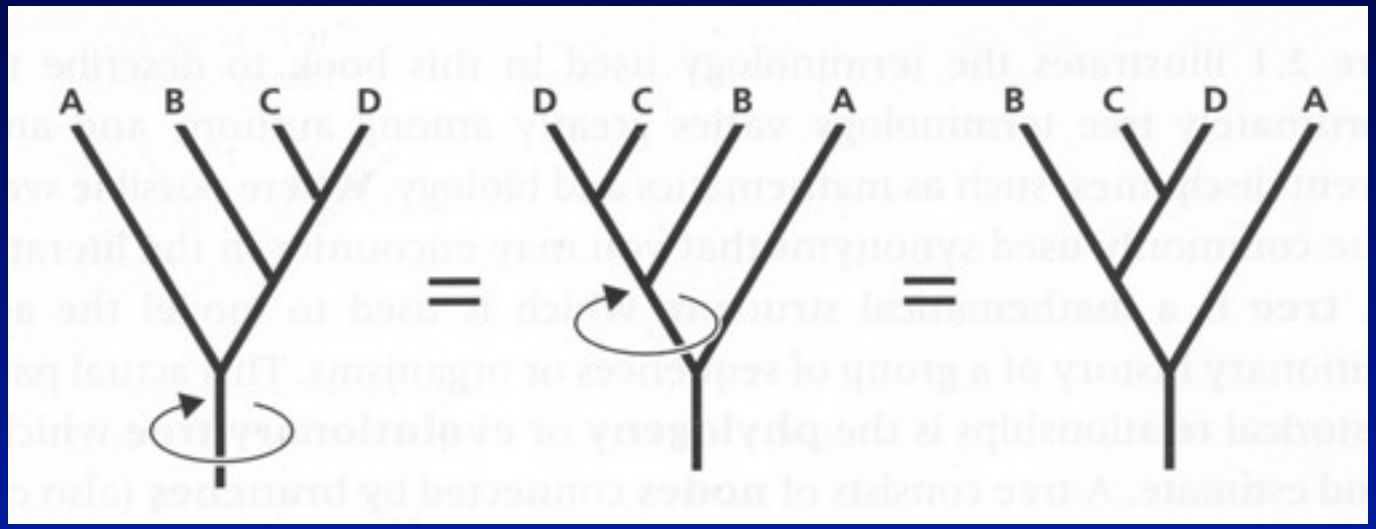


Trees: terminology



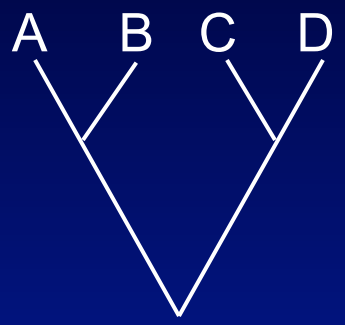


Trees: representations



Three different representations of the same tree

Trees: representation in computer files



((A , B) , (C , D));

Newick format:

- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

Newick format: named for seafood restaurant where standard was decided upon

Trees: representation in computer files

Newick format:

- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

Trees: representation in computer files

Newick format:

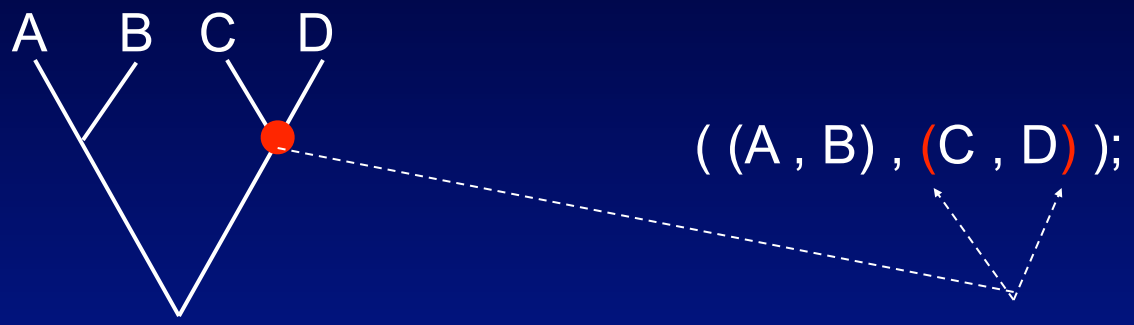
- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

Trees: representation in computer files

Newick format:

- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

Trees: representation in computer files



Newick format:

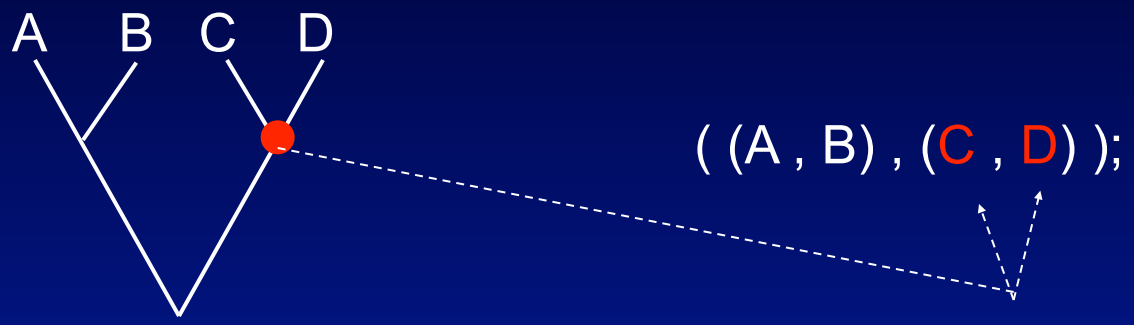
- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

Trees: representation in computer files

Newick format:

- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

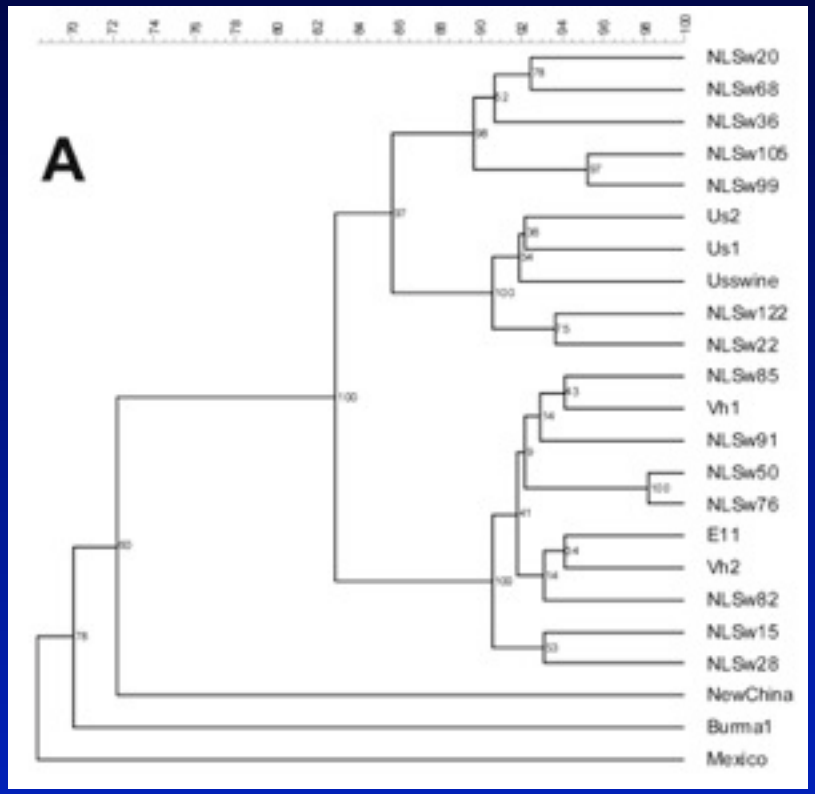
Trees: representation in computer files



Newick format:

- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

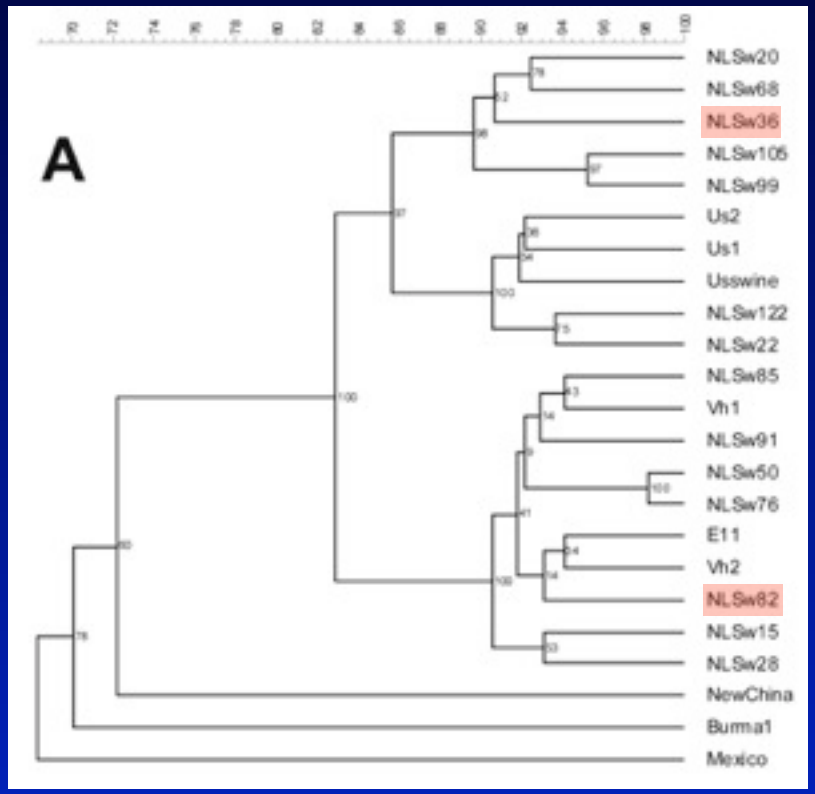
Trees: rooted vs. unrooted



Early → Late

- A rooted tree has a single node (the root) that represents a point in time that is earlier than any other node in the tree.
- A rooted tree has directionality (nodes can be ordered in terms of “earlier” or “later”).
- In the rooted tree, distance between two nodes is represented along the time-axis only (the second axis just helps spread out the leafs)

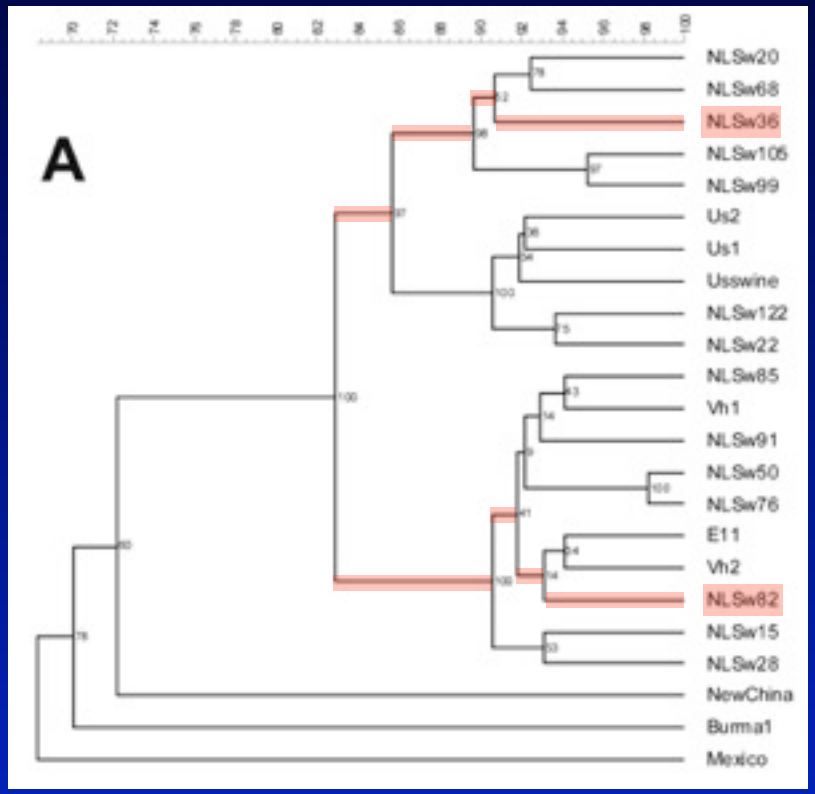
Trees: rooted vs. unrooted



Early → Late

- A rooted tree has a single node (the root) that represents a point in time that is earlier than any other node in the tree.
- A rooted tree has directionality (nodes can be ordered in terms of “earlier” or “later”).
- In the rooted tree, distance between two nodes is represented along the time-axis only (the second axis just helps spread out the leafs)

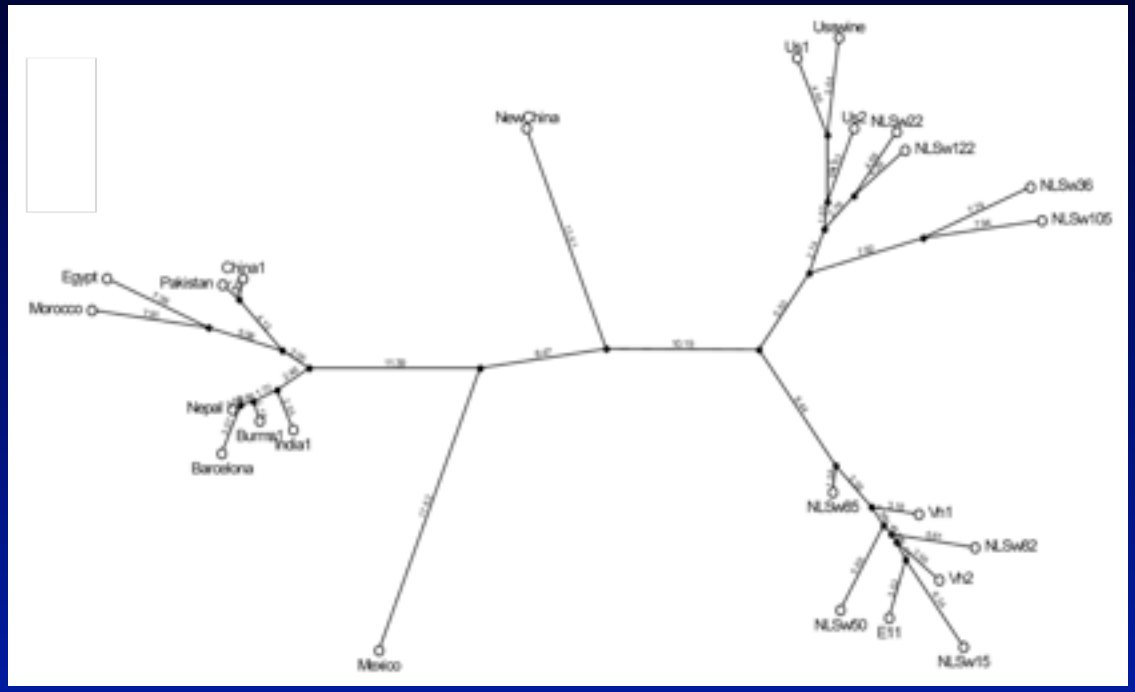
Trees: rooted vs. unrooted



Early → Late

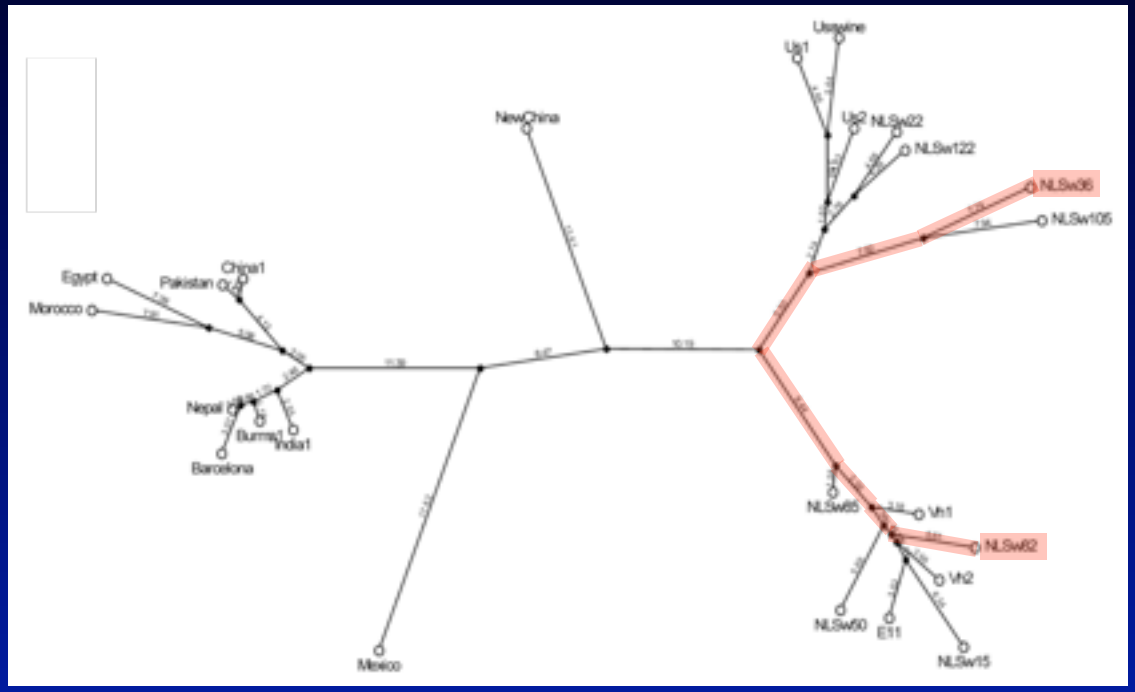
- A rooted tree has a single node (the root) that represents a point in time that is earlier than any other node in the tree.
- A rooted tree has directionality (nodes can be ordered in terms of “earlier” or “later”).
- In the rooted tree, distance between two nodes is represented along the time-axis only (the second axis just helps spread out the leafs)

Trees: rooted vs. unrooted



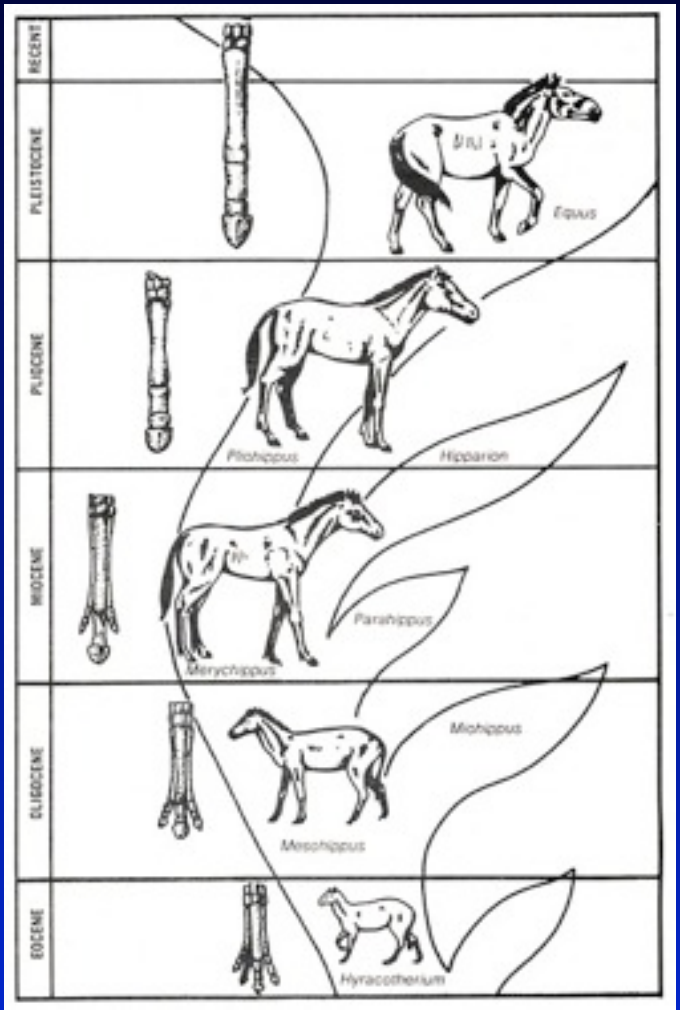
- In unrooted trees there is no directionality: we do not know if a node is earlier or later than another node
- Distance along branches directly represents node distance

Trees: rooted vs. unrooted

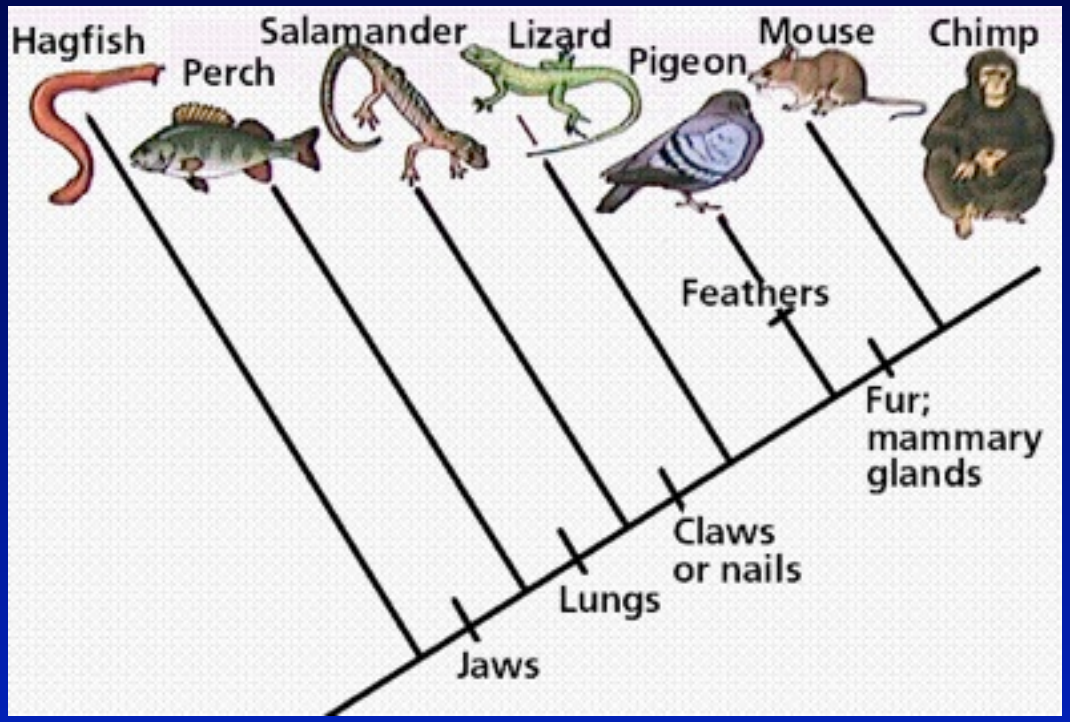


- In unrooted trees there is no directionality: we do not know if a node is earlier or later than another node
- Distance along branches directly represents node distance

Reconstructing a tree using non-contemporaneous data



Reconstructing a tree using present-day data



Data: molecular phylogeny

- DNA sequences
  - genomic DNA
  - mitochondrial DNA
  - chloroplast DNA
- Protein sequences
- Restriction site polymorphisms
- DNA/DNA hybridization
- Immunological cross-reaction

Morphology vs. molecular data

African white-backed vulture
(old world vulture)

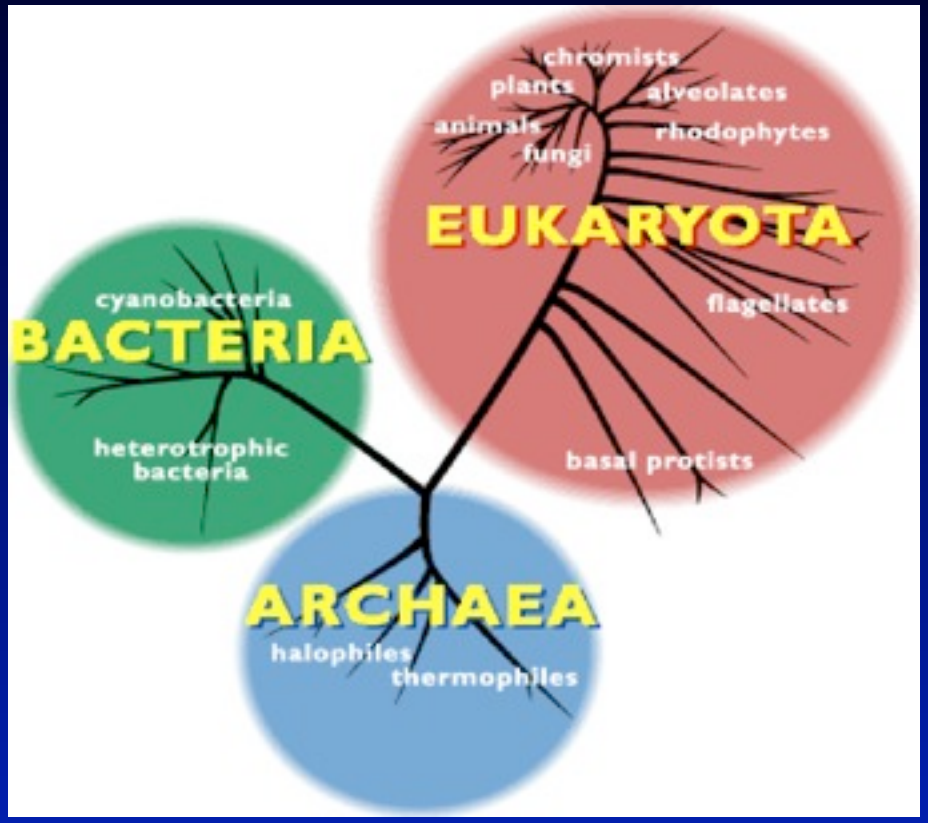
Andean condor
(new world vulture)

New and old world vultures seem to be closely related based on morphology.

Molecular data indicates that old world vultures are related to birds of prey (falcons, hawks, etc.) while new world vultures are more closely related to storks

Similar features presumably the result of convergent evolution

Molecular data: single-celled organisms



Molecular data useful for analyzing single-celled organisms (which have only few prominent morphological features).

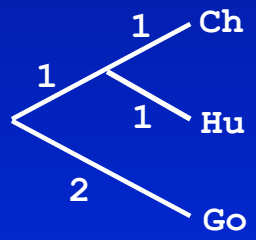
Distance Matrix Methods


```
Gorilla    :  ACGTCGTA
Human      :  ACGTTCCT
Chimpanzee :  ACGTTTCG
```

1. Construct multiple alignment of sequences

	Go	Hu	Ch
Go	-	4	4
Hu		-	2
Ch			-

2. Construct table listing all pairwise differences (distance matrix)

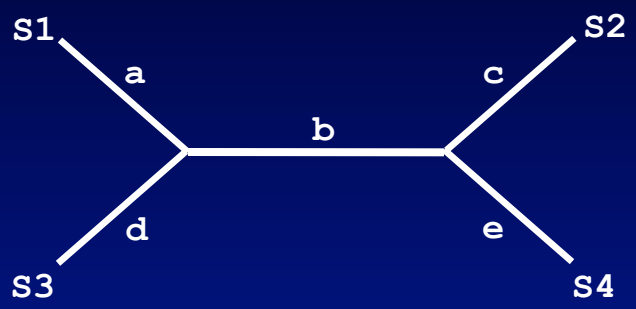


3. Construct tree from pairwise distances

Finding Optimal Branch Lengths

	S_1	S_2	S_3	S_4
S_1	-	D_{12}	D_{13}	D_{14}
S_2		-	D_{23}	D_{24}
S_3			-	D_{34}
S_4				-

Observed distance

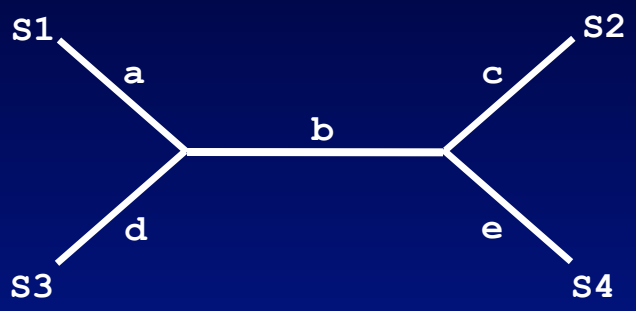


Distance along tree

Goal:

$$D_{12} \approx d_{12} = a + b + c$$
$$D_{13} \approx d_{13} = a + d$$
$$D_{14} \approx d_{14} = a + b + e$$
$$D_{23} \approx d_{23} = d + b + c$$
$$D_{24} \approx d_{24} = c + e$$
$$D_{34} \approx d_{34} = d + b + e$$

Optimal Branch Lengths: Least Squares



Distance along tree

Goal:

$$D_{12} \approx d_{12} = a + b + c$$
$$D_{13} \approx d_{13} = a + d$$
$$D_{14} \approx d_{14} = a + b + e$$
$$D_{23} \approx d_{23} = d + b + c$$
$$D_{24} \approx d_{24} = c + e$$
$$D_{34} \approx d_{34} = d + b + e$$

- Fit between given tree and observed distances can be expressed as “sum of squared differences”:

$$Q = \sum_{j>i} (D_{ij} - d_{ij})^2$$

- Find branch lengths that minimize Q - this is the optimal set of branch lengths for this tree.

Least Squares Optimality Criterion

- Search through all (or many) tree topologies
- For each investigated tree, find best branch lengths using least squares criterion
- Among all investigated trees, the best tree is the one with the smallest sum of squared errors.

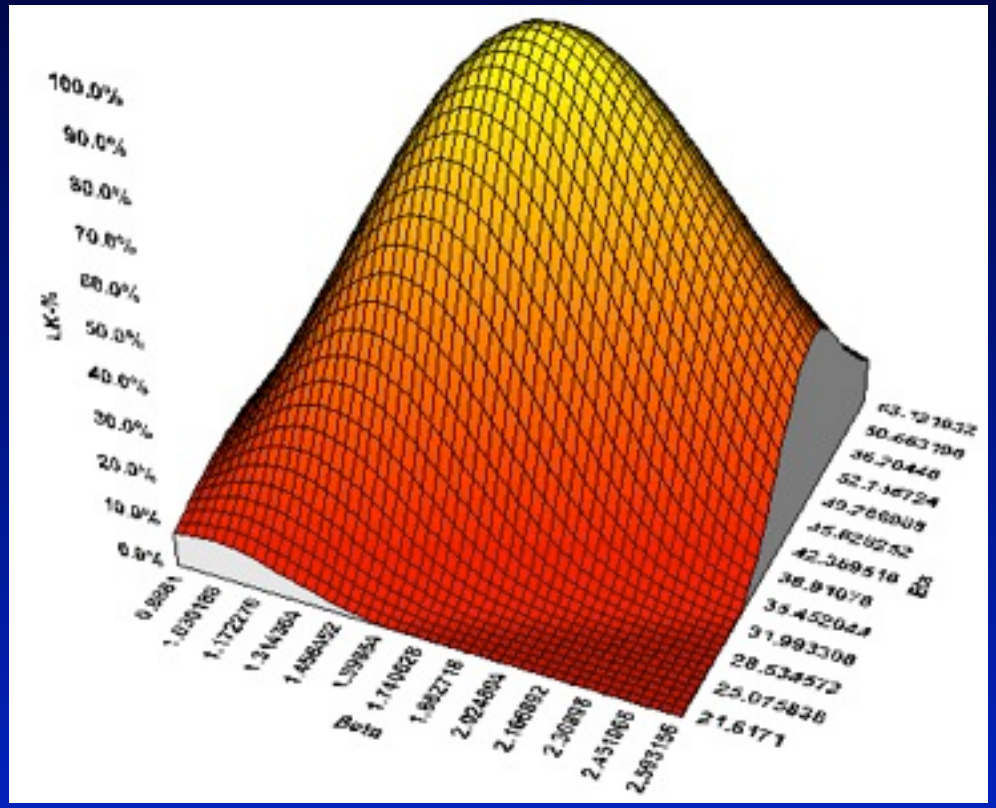
Exhaustive search impossible for large data sets

No. taxa	No. trees
3	1
4	3
5	15
6	105
7	945
8	10,395
9	135,135
10	2,027,025
11	34,459,425
12	654,729,075
13	13,749,310,575
14	316,234,143,225
15	7,905,853,580,625

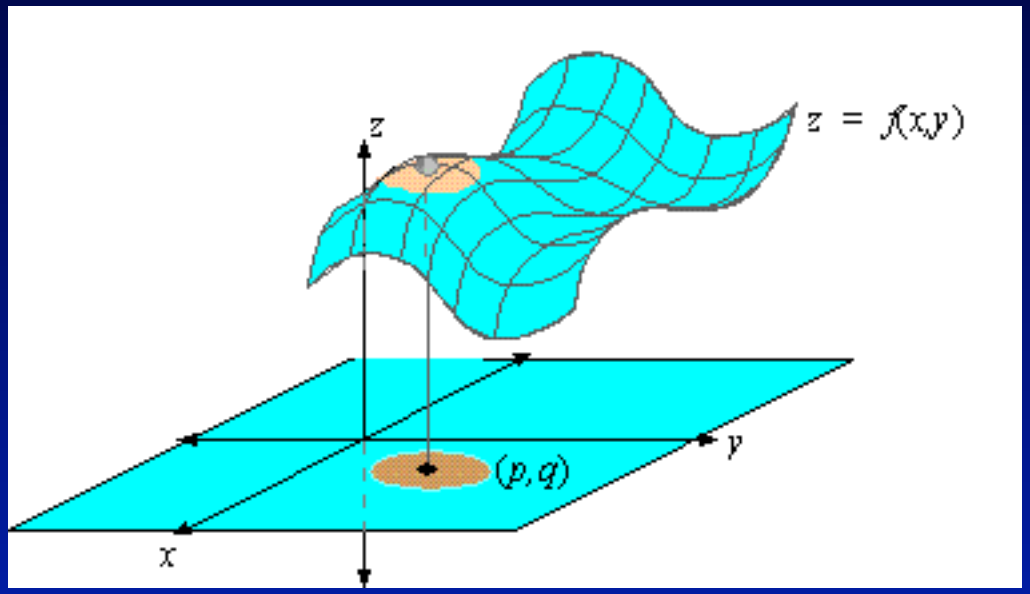
Heuristic search

1. Construct initial tree; determine sum of squares
2. Construct set of “neighboring trees” by making small rearrangements of initial tree; determine sum of squares for each neighbor
3. If any of the neighboring trees are better than the initial tree, then select it/them and use as starting point for new round of rearrangements. (Possibly several neighbors are equally good)
4. Repeat steps 2+3 until you have found a tree that is better than all of its neighbors.
5. This tree is a “local optimum” (not necessarily a global optimum!)

Heuristic search: hill-climbing

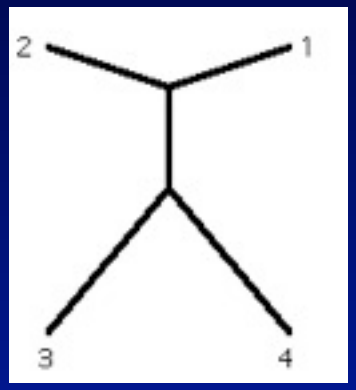


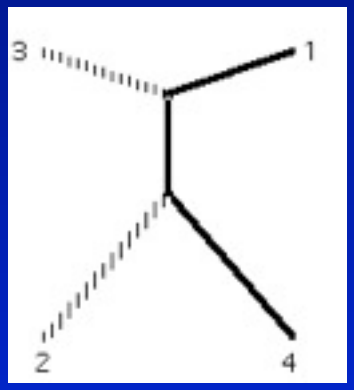
Heuristic search: local *vs.* global optimum

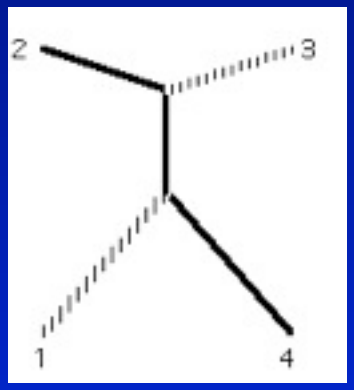


Types of rearrangement I: nearest neighbor interchange (NNI)

Original tree

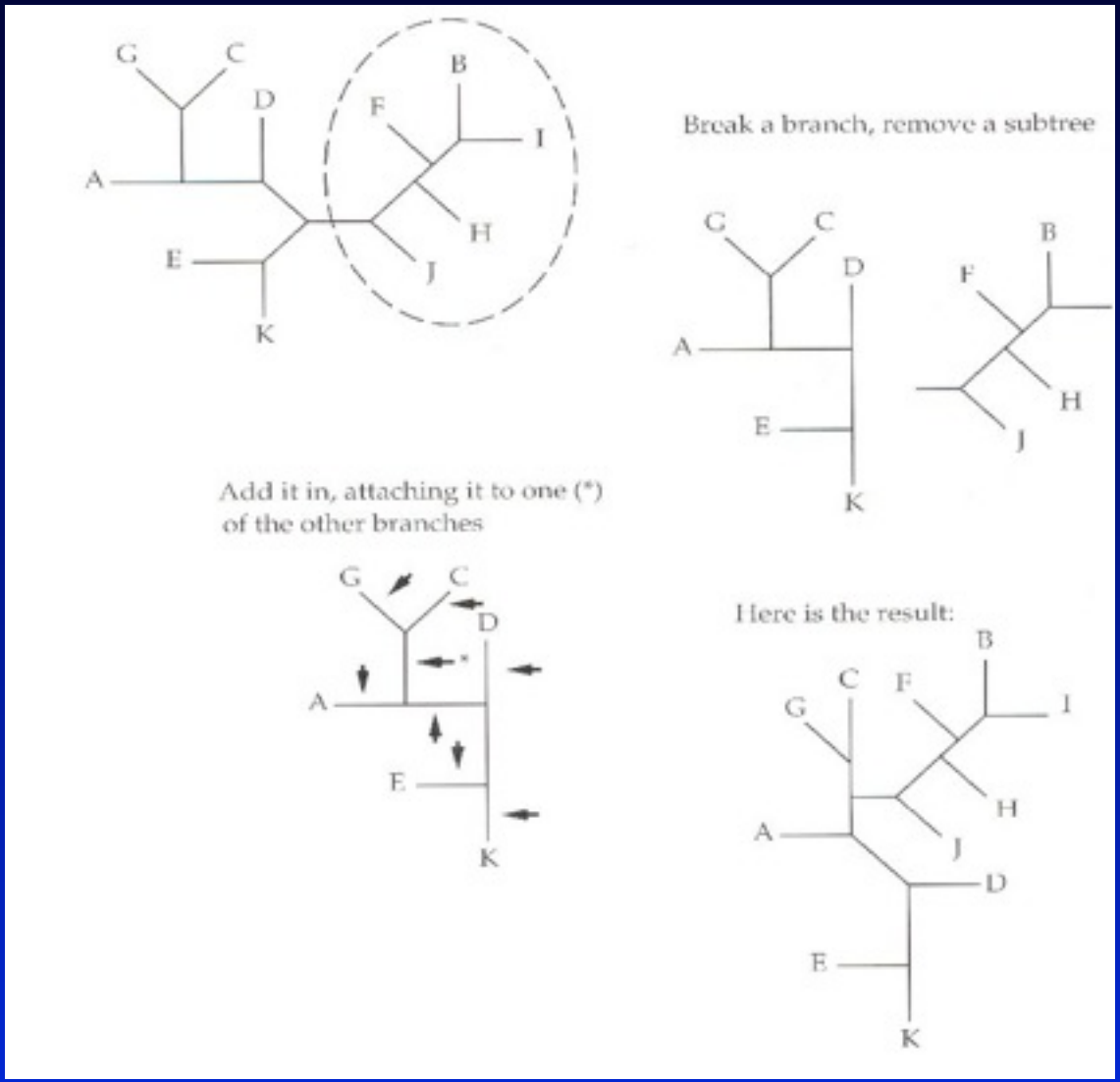






Two neighbors per internal branch: tree with n tips has 2(n-3) neighbors (For example, a tree with 20 tips has 34 neighbbors)

Types of rearrangement II: subtree pruning and regrafting (SPR)



Types of rearrangement III: tree bisection and reconnection (TBR)

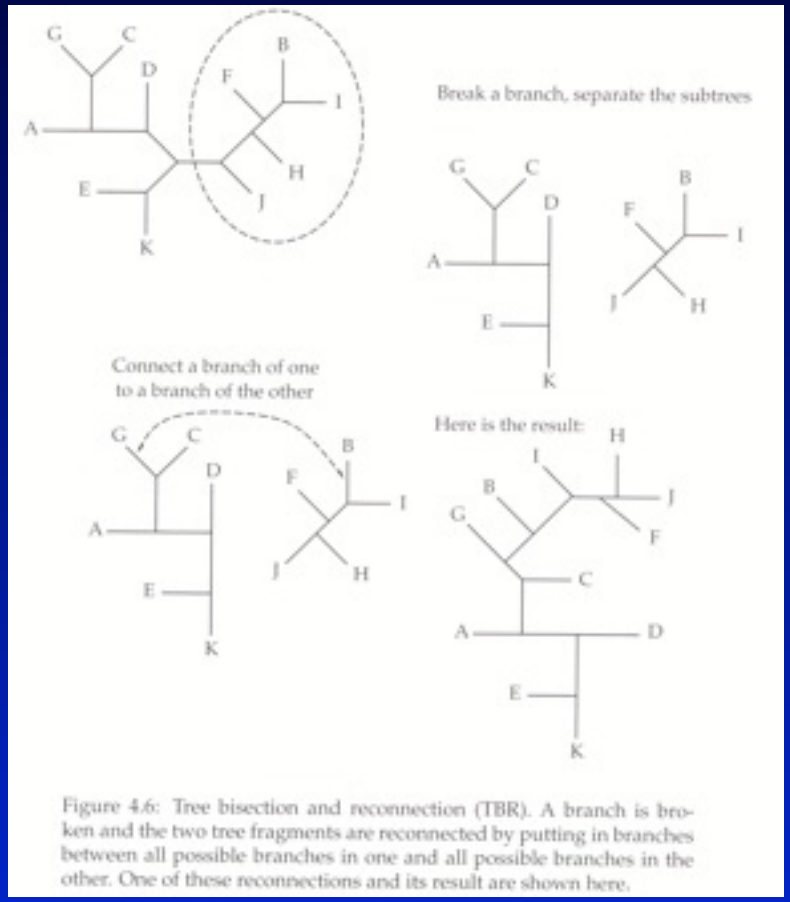


Figure 4.6: Tree bisection and reconnection (TBR). A branch is broken and the two tree fragments are reconnected by putting in branches between all possible branches in one and all possible branches in the other. One of these reconnections and its result are shown here.

Clustering Algorithms

- Starting point: Distance matrix
- Cluster least different pair of sequences:
- Repeat until all nodes are linked
- Results in only one tree, there is no measure of tree-goodness.

Neighbor Joining Algorithm

- For each tip compute $u_i = \sum_j D_{ij} / (n-2)$

  (this is essentially the average distance to all other tips, except the denominator is n-2 instead of n)

- Find the pair of tips, i and j, where $D_{ij} - u_i - u_j$ is smallest

- Connect the tips i and j, forming a new ancestral node. The branch lengths from the ancestral node to i and j are:

$$v_i = 0.5\ D_{ij} + 0.5\ (u_i - u_j)$$

$$v_j = 0.5\ D_{ij} + 0.5\ (u_j - u_i)$$

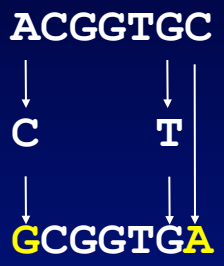
- Update the distance matrix: Compute distance between new node and each remaining tip as follows:

$$D_{ij,k} = (D_{ik} + D_{jk} - D_{ij})/2$$

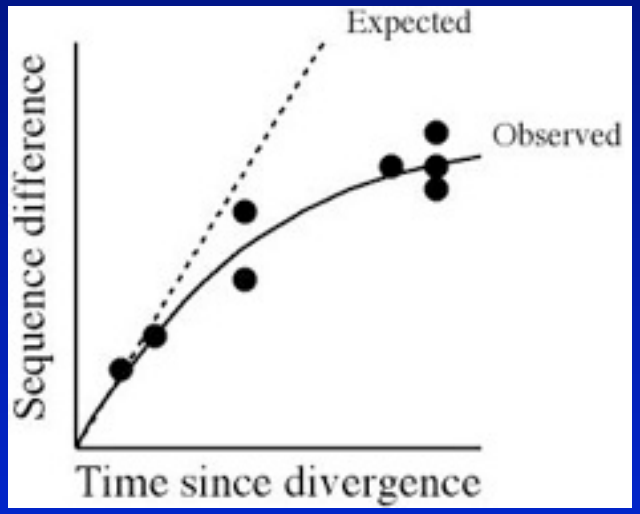
- Replace tips i and j by the new node which is now treated as a tip

- Repeat until only two nodes remain.

Superimposed Substitutions



- Actual number of evolutionary events: 5
- Observed number of differences: 2



- Distance is (almost) always underestimated
- It is possible to estimate real distance from observed distance if one makes assumptions about how evolution has occurred (substitution frequencies etc).