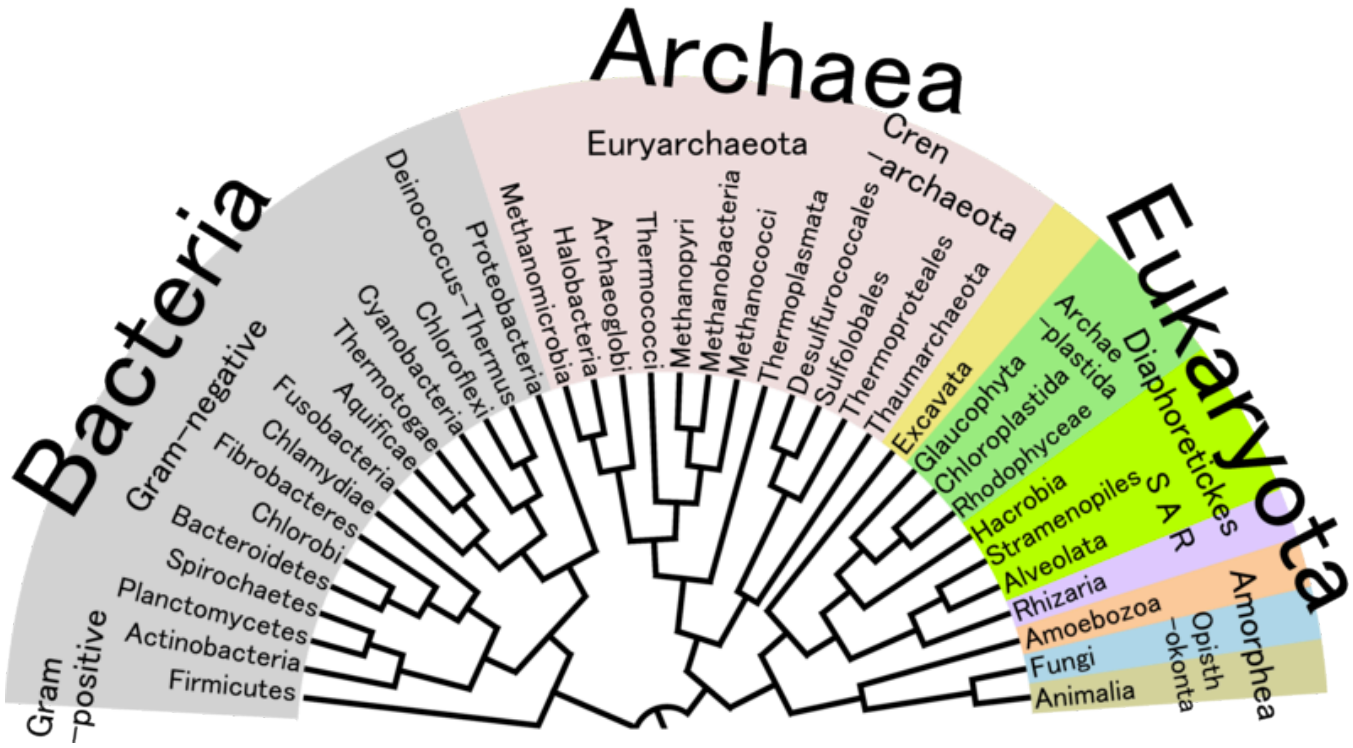
Phylogenetics



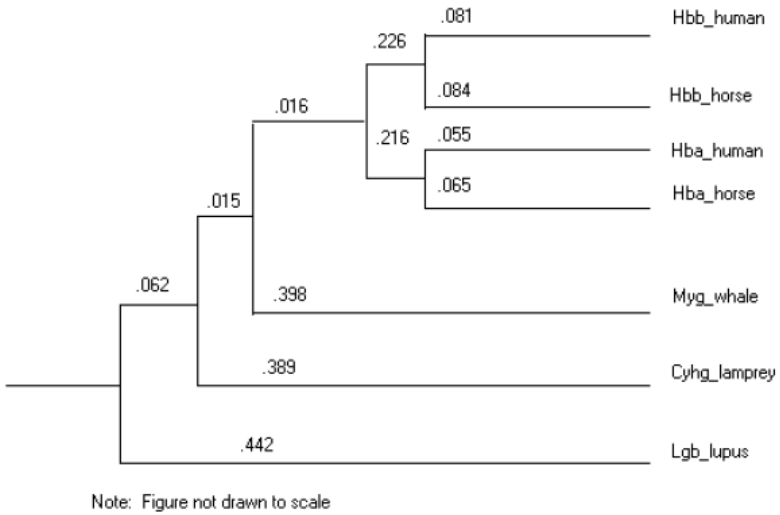
ECES T480/680

Rosen

Constructing Phylogenetic Trees

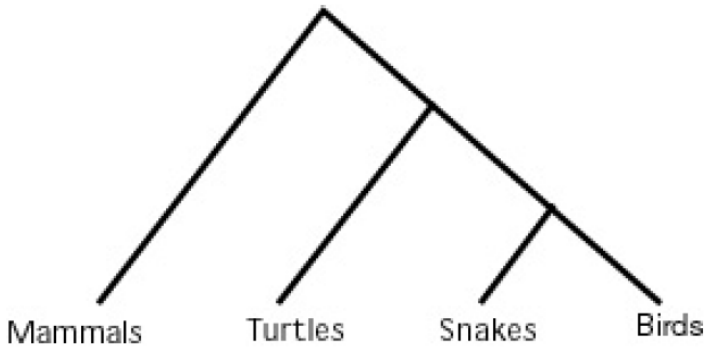
- Phylogenetic trees illustrate the evolutionary relationships among groups of organisms, or among a family of related nucleic acid or protein sequences
- E.g., how might have this family been derived during evolution

Globin Sequences



Note: Figure not drawn to scale

Hypothetical Tree Relating Organisms



Phylogenetic Relationships Among Organisms

- Entrez: www.ncbi.nlm.nih.gov/Taxonomy
- Ribosomal database project:
  rdp.cme.msu.edu/html/
- Tree of Life:
  phylogeny.arizona.edu/tree/phylogeny.html

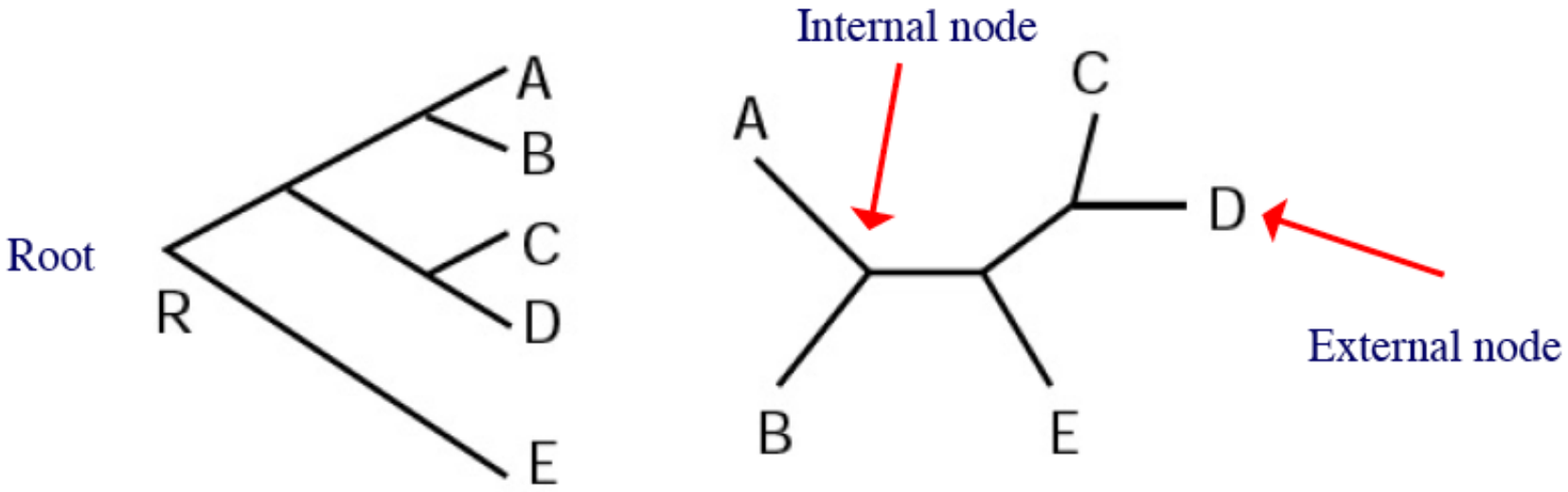
Phylogeny Applications

- Tree of life: Analyzing changes that have occurred in evolution of different organisms
- Phylogenetic relationships among genes can help predict which ones might have similar functions (e.g., ortholog detection)
- Follow changes occuring in rapidly changing species (e.g., HIV virus)

Traditional Methods

- Traditionally: morphological features (e.g.,number of legs, beak shape, etc.)
- Today: Mostly molecular data (e.g., DNA and protein sequences)

Rooted vs. Unrooted Trees



Rooted tree

Unrooted tree

Note: Here, each node has three neighboring nodes

Terminology

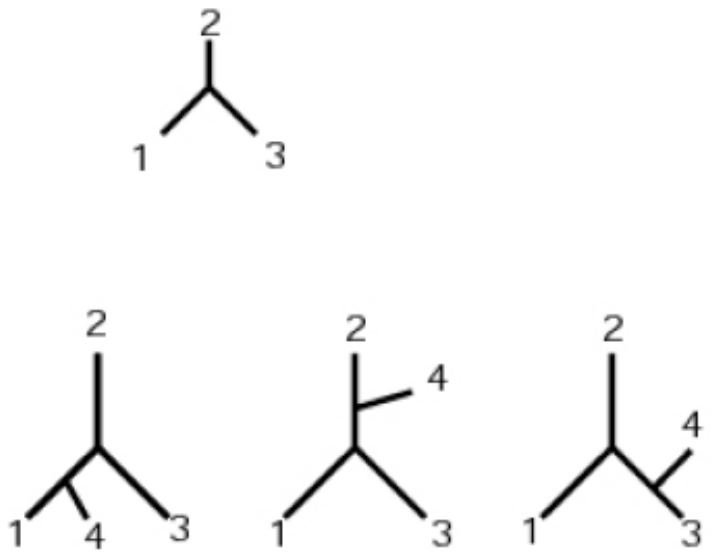
- External nodes: things under comparison; operational taxonomic units (OTUs)
- Internal nodes: ancestral units; hypothetical; goal is to group current day units
- Root: common ancestor of all OTUs under study. Path from root to node defines evolutionary path
- Unrooted: specify relationship but not evolutionary path
  - If have an outgroup (external reason to believe certain OTU branched off first), then can root
- Topology: branching pattern of a tree
- Branch length: amount of difference that occurred along a branch

Tree construction methods

- Distance methods: evolutionary distances are computed for all OTUs and build tree where distance between OTUs "matches" these distances
- Maximum parsimony (MP): choose tree that minimizes number of changes required to explain data
- Maximum likelihood (ML): under a model of sequence evolution, find the tree which gives the highest likelihood of the observed data

Number of Possible Trees

Given n OTUs, there are $\prod_{i=3}^{n}(2i-5)$ unrooted trees



OTUs	unrooted trees
3	1
4	3
5	15
10	2,027,025

Number of possible trees

Given n OTUs, there are $\prod_{i=3}^{n}(2i-3)$ rooted trees

Bottom Line: an enumeration strategy over all possible trees to find the best one under some criteria is not feasible!

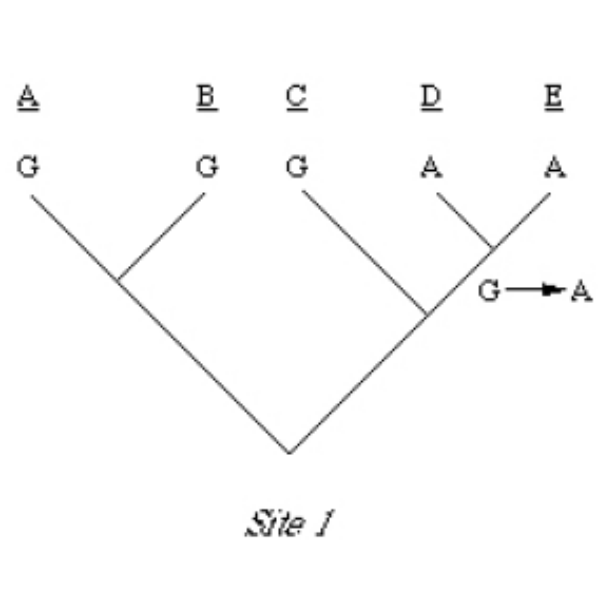
OTUs	Rooted trees
3	3
4	15
5	105
10	34,459,425

Parsimony

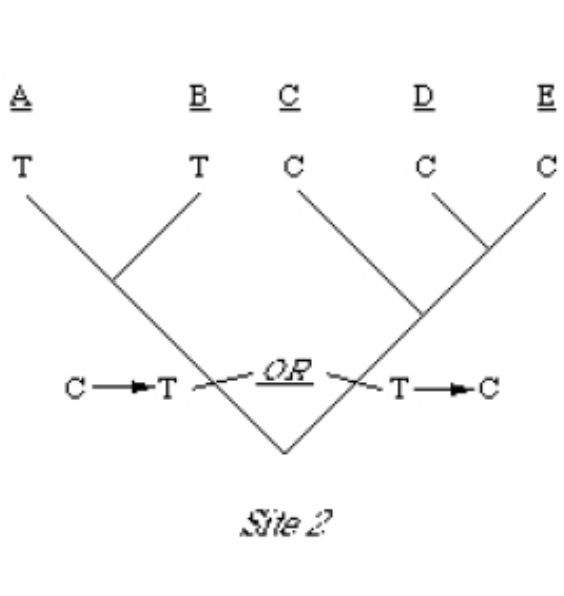
Find tree which minimizes number of changes needed to explain data

Ex:

	123456
A	GTCGTA
B	GTCACT
C	GCGGTA
D	ACGACA
E	ACGGAA



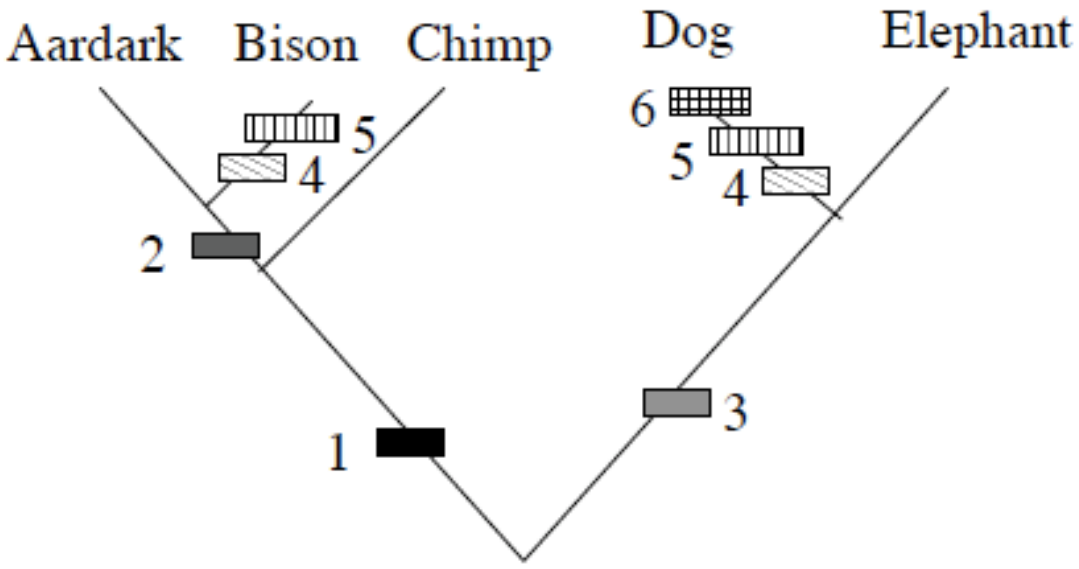
Site 1



Site 2

Example for all sites

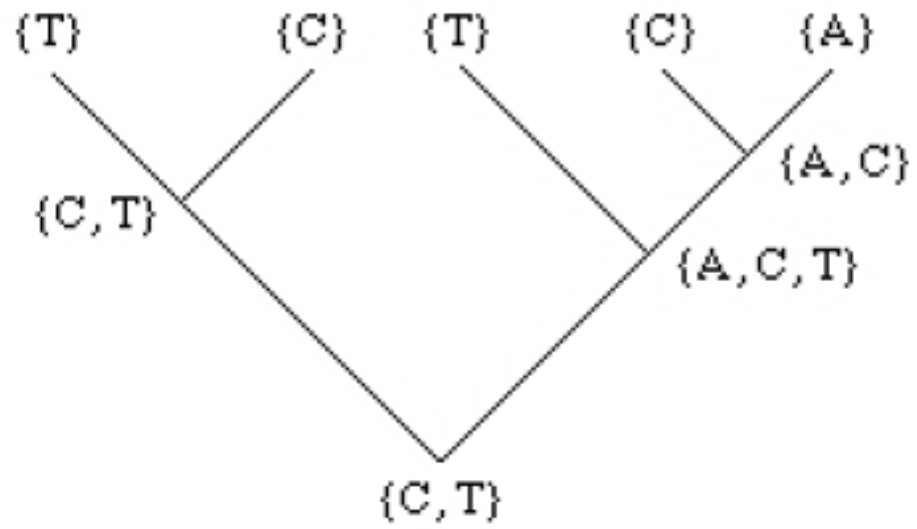
Species	site 1	site 2	site 3	site 4	site 5	site 6
Aardvark	C	A	G	G	T	A
Bison	C	A	G	A	C	A
Chimp	C	G	G	G	T	A
Dog	T	G	C	A	C	T
Elephant	T	G	C	G	T	A



Parsimony

- For given example tree and alignment, can do this for all sites, and get away with as few as 8 changes
- Changing the tree (either the topology or labeling of leaves) changes the minimum number of changes need
- Two computational problems
  - (Easy) Given a particular tree, how do you find minimum number of changes need to explain data? (Fitch)
  - (Hard) How do you search through all trees?

Parsimony: Fitch's Algorithm



Idea: construct set of possible nucleotides for internal nodes, based on possible assignments of children

Parsimony: Fitch's Algorithm

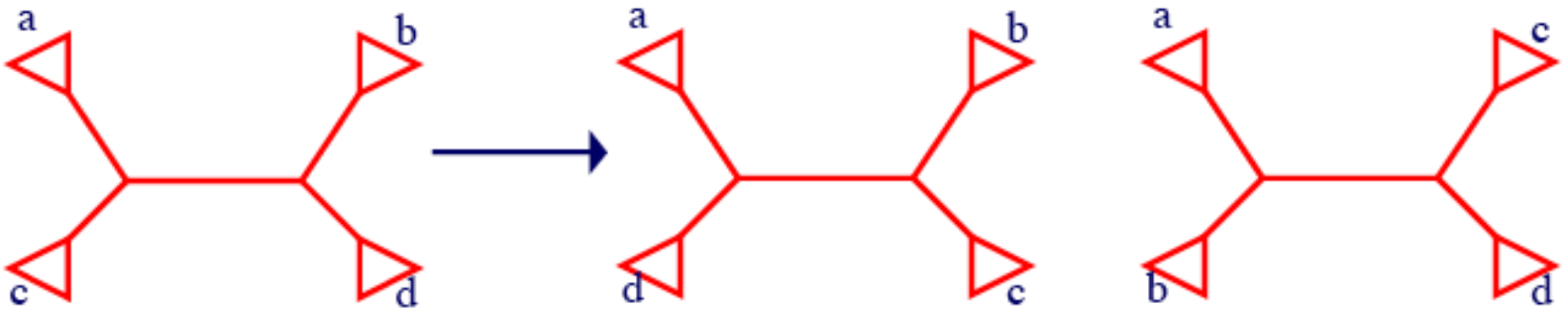
- For each site:
  - Each leaf is labeled with set containing observed nucleotide at that position
  - For each internal node i with children j and k with labels S_j and S_k

$$S_i = \begin{cases} S_j \cup S_k \text{ if } S_j \cap S_k \text{ is empty} \\ S_j \cap S_k \text{ otherwise} \end{cases}$$

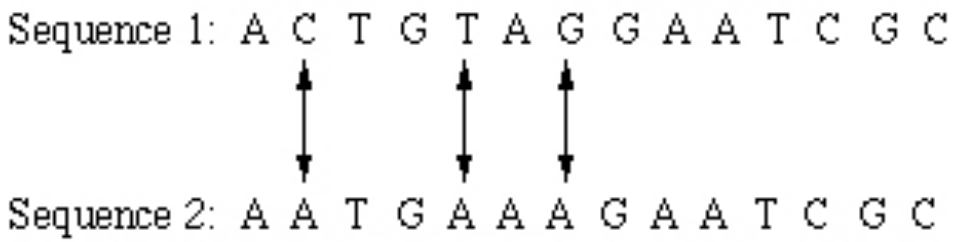
- Total # changes necessary for a site is # of union operations

Parsimony

- How do you search through all trees?
  - Enumerate all trees (too many…)
  - Can use techniques to try to limit the search space (e.g., branch and bound)
  - or use heuristics (many possibilities)
    - E.g., nearest neighbor interchange. Start with a tree and consider neighboring trees. If any neighboring tree has fewer changes, take it as current tree. Stop when no improvements



Computing Distances between two sequences



Could compute fraction of mismatches between two sequences; however, this is an underestimate of actual distance

A simple clustering method for building a ROOTED tree

UPGMA (Unweighted Pair Group Method using Arithmetic averages)
Or the **Average Linkage Method**

Given two disjoint clusters S_i, S_j of sequences.

$$d_{ij} = \frac{1}{|S_i| \times |S_j|} \sum_{\{p \in Si,\, q \in Sj\}} d_{pq}$$

Claim that if $S_k = S_i \cup S_j$, then distance to another cluster S_l is:

$$d_{kl} = \frac{d_{il}\,|S_i| + d_{jl}\,|S_j|}{|S_i| + |S_j|}$$

Algorithm: Average Linkage

Initialization:

Assign each x_i into its own cluster S_i

Define one leaf per sequence, height 0

Iteration:

Find two clusters S_i, S_j s.t. d_{ij} is min

Let $S_k = S_i \cup S_j$

Define node connecting S_i, S_j,

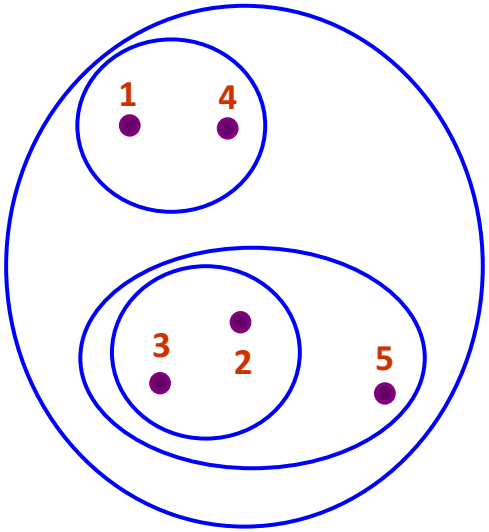
& place it at height $d_{ij}/2$

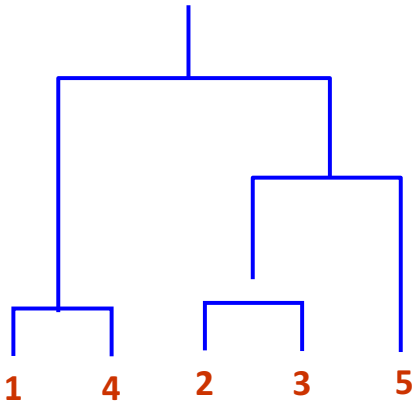
Delete S_i, S_j

Termination:

When two clusters i, j remain,

place root at height $d_{ij}/2$





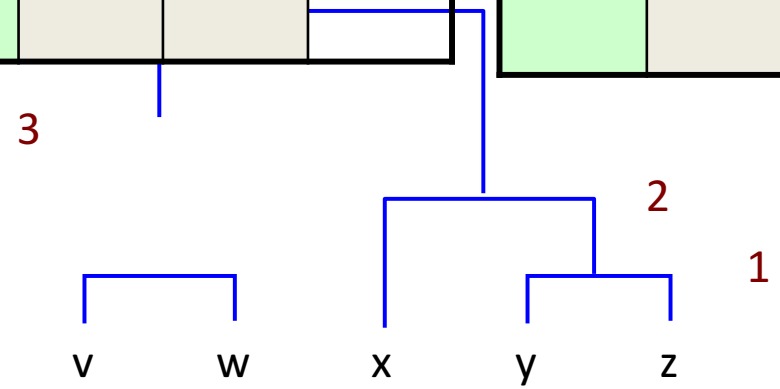
Example

	v	w	x	y	z
v	0	6	8	8	8
w		0	8	8	8
x			0	4	4
y				0	2
z					0

	v	w	yz
v	0	6	8
w		0	8
x			0

	v	w	xyz
v	0	6	8
w		0	8
xyz			0

	vw	xyz
vw	0	8
xyz		0



Ultrametric Distances and Molecular Clock

Definition:

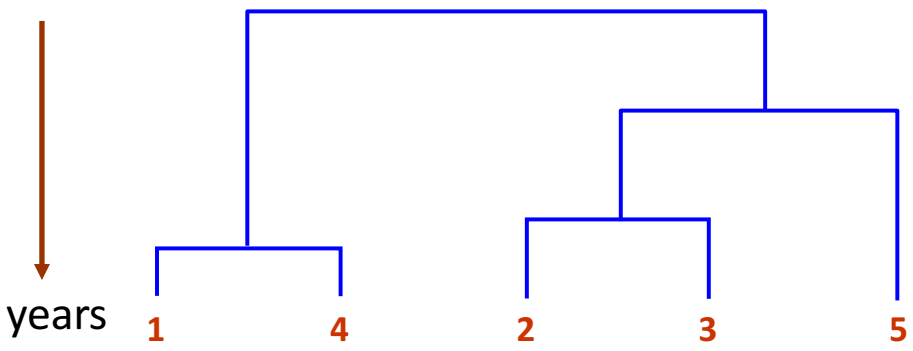
A distance function d(.,.) is ultrametric if for any three distances $d_{ij} \leq d_{ik} \leq d_{jk}$, it is true that

$$d_{ij} \leq d_{ik} = d_{jk}$$

The Molecular Clock:

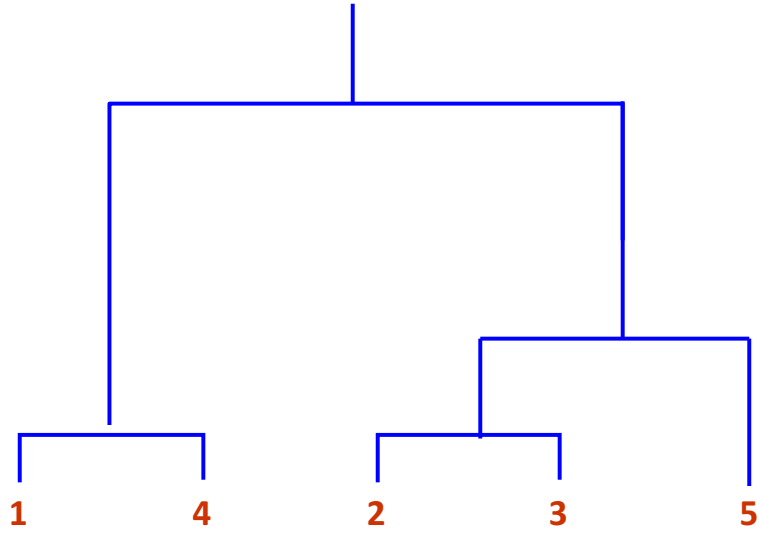
The evolutionary distance between species x and y is 2× the Earth time to reach the nearest common ancestor

That is, the molecular clock has constant rate in all species



The molecular clock results in ultrametric distances

Ultrametric Distances & Average Linkage



Average Linkage is guaranteed to reconstruct correctly a binary tree with ultrametric distances

Maximum Likelihood

- Given a probabilistic model for nucleotide (or protein) substitution (e.g., Jukes & Cantor), pick the tree that has highest probability of generating observed data
  - I.e., Given data D and model M, find tree T such that $Pr(D|T, M)$ is maximized
- Models gives values $p_{ij}(t)$, the probability of going from nucleotide i to j in time t

Maximum Likelihood

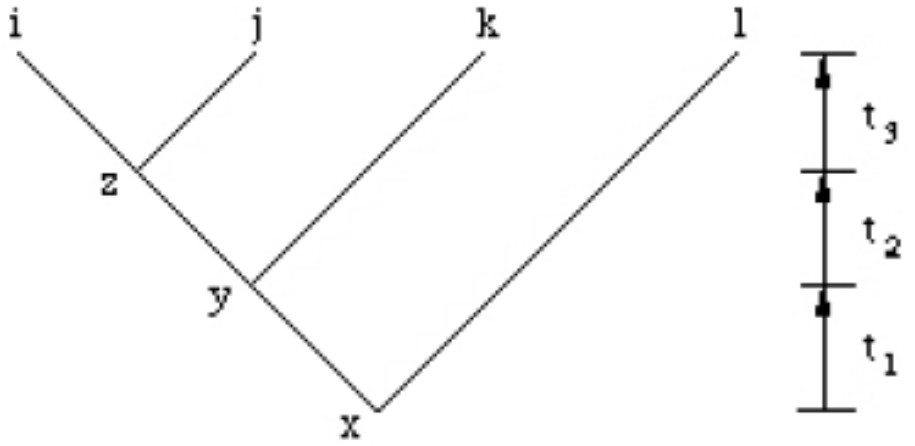
- Makes 2 independence assumptions
  - Different sites evolve independently
  - Diverged sequences (or species) evolve independently after diverging
- If D_i is data for ith site

$$Pr(D|T, M) = \prod_i Pr(D_i|T, M)$$

Maximum Likelihood

How to calculate $Pr(D_i|T,M)$?

$p_{xy}(t) \sim$ prob of going from x to y in time t



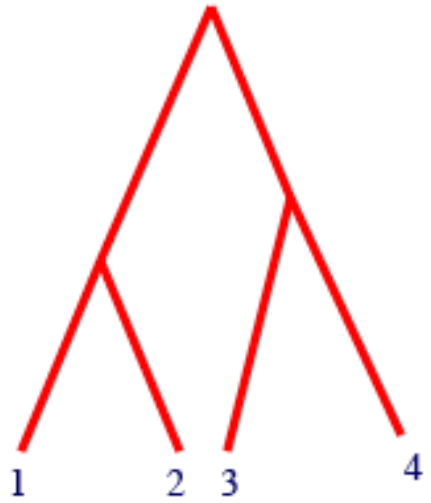
$$Pr(i,j,k,l|T,M) =$$
$$\sum_x \sum_y \sum_z pr(x)(p_{xl} \cdot (t_1 + t_2 + t_3) \cdot p_{xy}(t_1)$$
$$\cdot p_{yk}(t_2 + t_3) \cdot p_{yz}(t_2) \cdot p_{zi}(t_3) \cdot p_{zj}(t_3))$$

Maximum Likelihood

- Given tree topology and branch lengths, can efficiently calculate $Pr(D|T, M)$ using dynamic programming
  - I.e., don't have to enumerate over all internal states
- Finding best maximum likelihood tree is expensive
  - Must consider all topologies
  - Find best edge lengths for each topology
    - Idea: use some search procedure, e.g., EM, to optimize these lengths

Assessing Reliability -- The Bootstrap

Say we've inferred the following tree



Would like to get confidence levels that 1 & 2 belong together, and 3&4 belong together

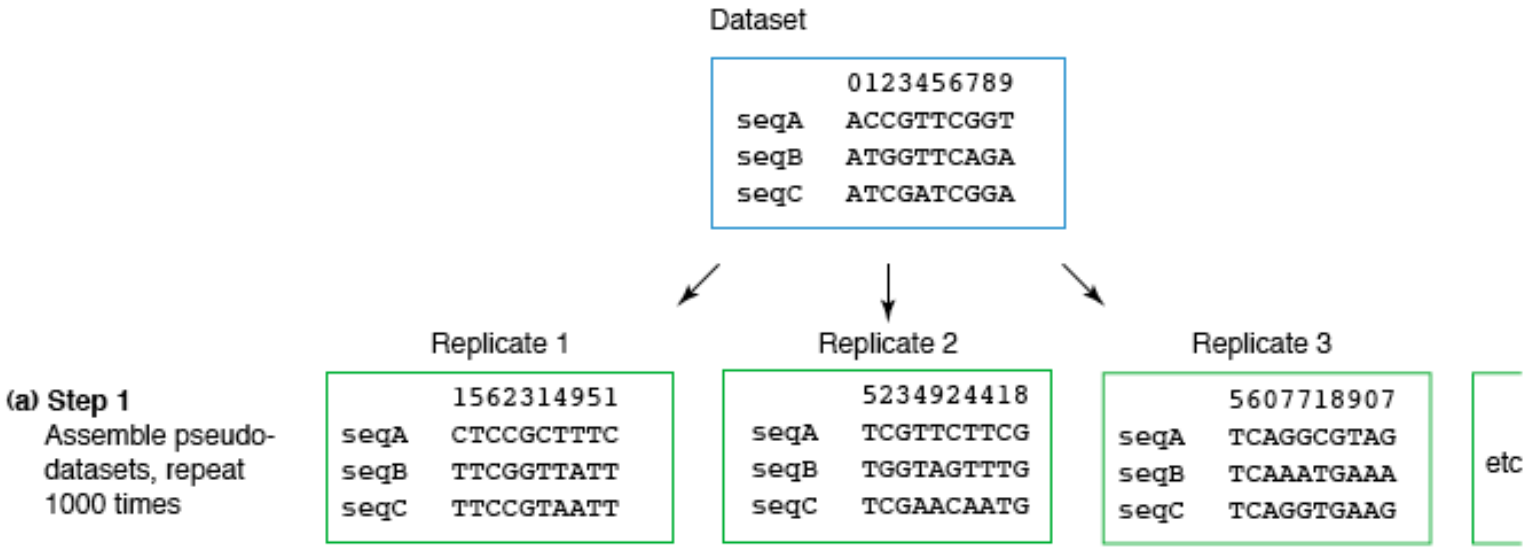
Assessing the Reliability - The Bootstrap

Say we're given following alignment:

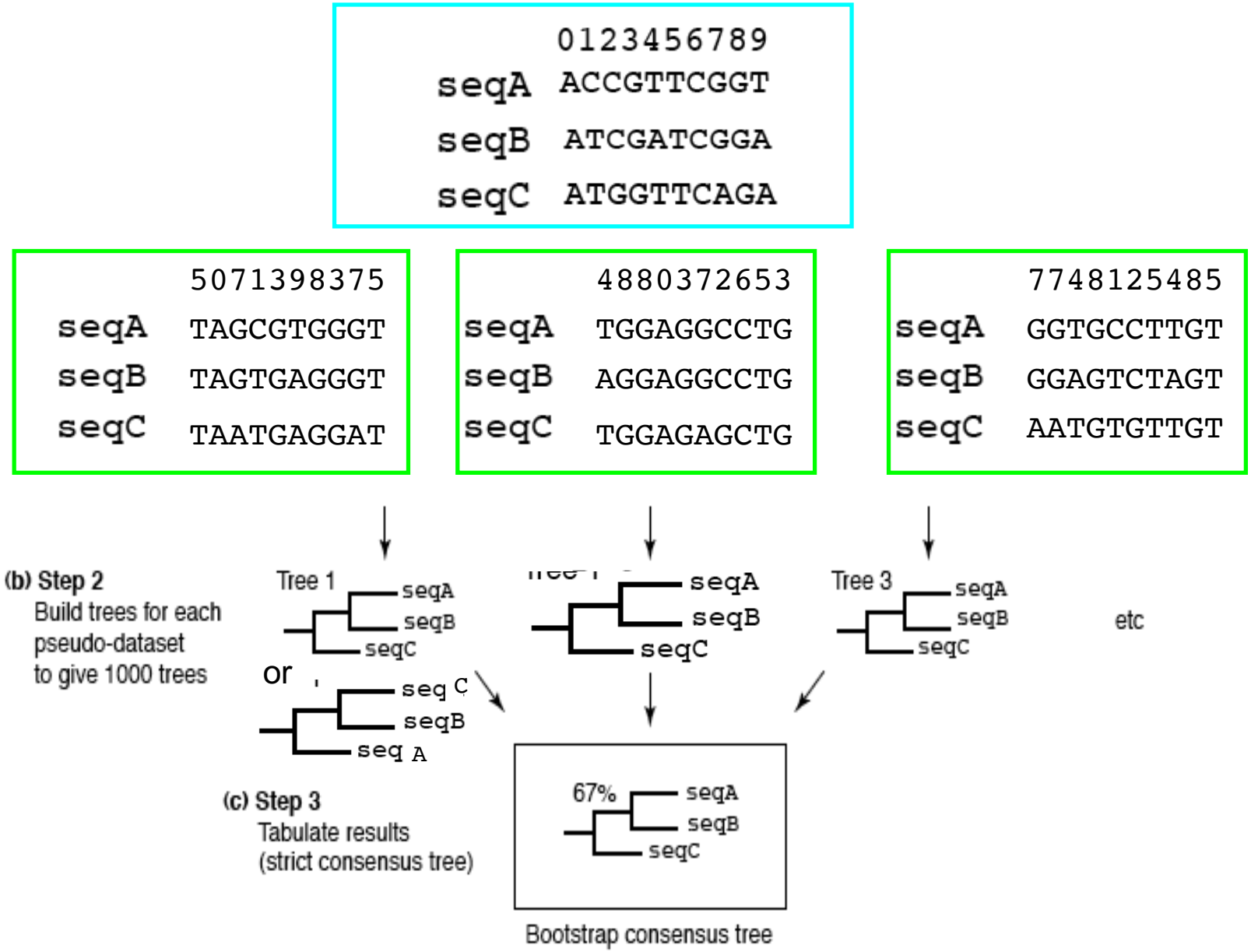
```
   12345678
1  GCAGTACT
2  GTAGTACT
3  ACAATACC
4  ACAACACT
```

We'll create a pseudosample by choosing sites randomly until N sites are chosen (N is length of alignment)

Bootstrapping



Bootstrapping



Many to choose from

- Serial Sequence Alignment
  - ClustalW
  - Contralign
  - MUSCLE
  - PROBCONS
  - PROBALIGN
  - Poy
- Serial Tree Inference
  - PAUP
  - Poyt

Large Data Issues

- Serial
  - Many memory requirements
  - Long time
- Parallel
  - Break into chunks
  - Less time

Parallel Align& TreeCodes

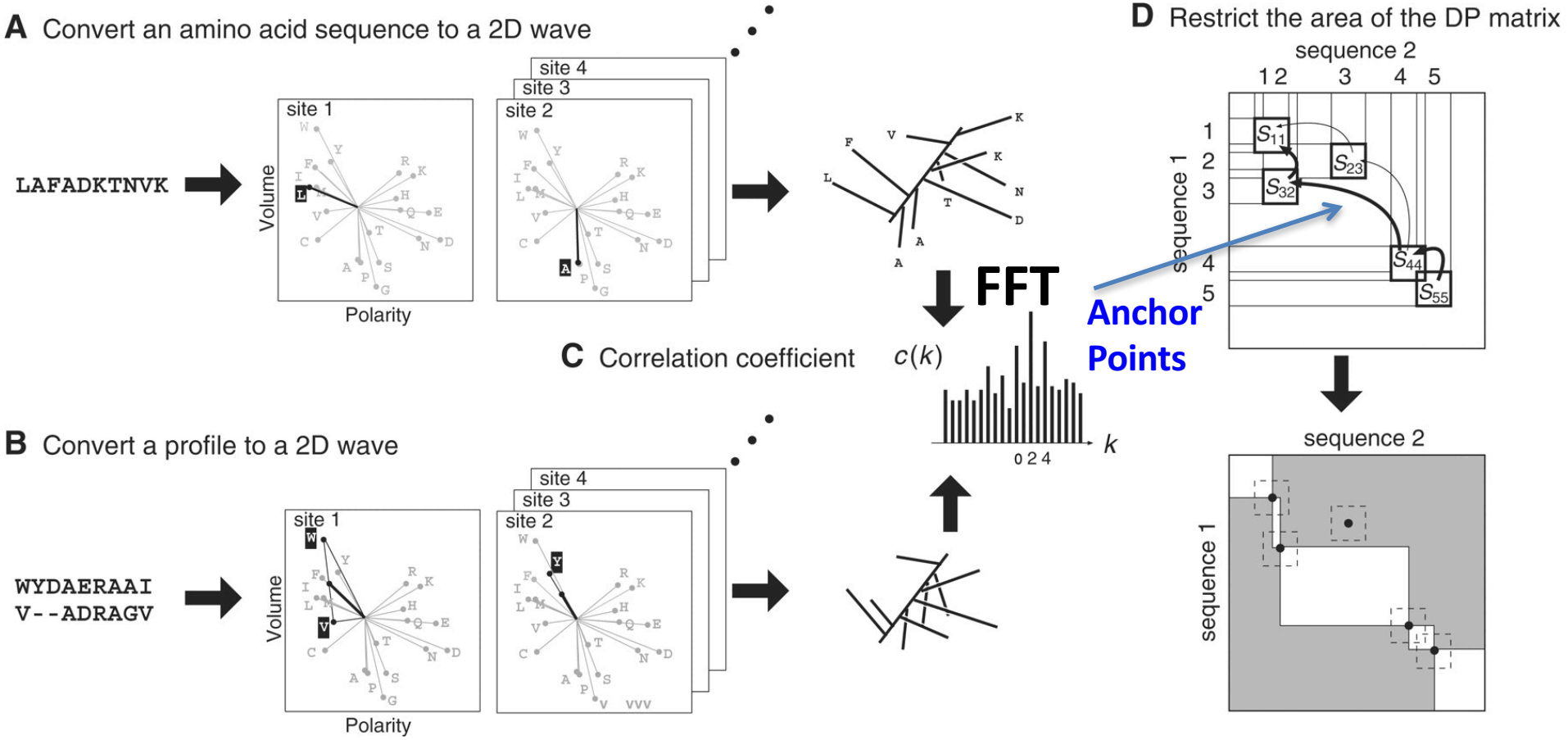
Alignment

- MAFFT

Tree

- RAxML
- MrBayes
- BEAST(2)
- GARLI
- PhyloBayes
- DPPDIV
- FastTree
- jModelTest2

MAFFT
(Multiple Alignment using Fourier Transform)



Tree Inference from alignments

ML

- **RaxML** (several heuristics to reduce search)
- **FastTree** (nearest neighbor exchanges for ml search)
- **Garli** (use genetic algorithm for ML search)

Bayesian MCMC

(Monte Carlo Markov Chains)

- **MrBayes**
- **PhyloBayes** (an infinite mixture model accounting for site-specific amino-acid or nucleotide preferences)
- **Beast** (relaxed molecular clock and demographic history)

Tree from different models of Nucleotide Substitution

- DPPDIV (Using Fixed tree topology, change parameters using MCMC)
- jModelTest2 (Likelihood ratio tests, information criterion, and decision theory to get candidate trees)

Tree Inference – What to use?!

- Almost no systematic comparisons
- ML techniques: RAxML to FastTree (http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0027731)
  - Says FastTree may be faster and just as good as raxml on large datasets
- MCMC: MrBayes is classic
- Ones that don't use alignment seem on the front of the state-of-the-art
  - Suggest UNCOMMON models of molecular evolution