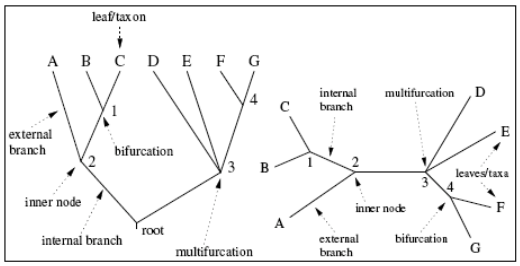
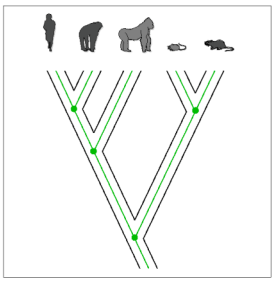
Phylogenetic inference

Notations:



sequence evolution



Sequence Alignment

seq 1	a	g	c	t	t	a	c	c	t	g	t	t	a	c	t
seq 2	c	g	t	a	a	a	t	t	t	c	c	c	g	a	t
seq 3	c	g	c	a	a	g	t	t	t	c	c	c	g	a	t
seq 4	c	a	c	t	t	a	t	t	a	g	t	c	a	a	c

Classification of phylogenetic methods

Data	Method	Evaluation Criterion
Characters (Alignment)	**Maximum Parsimony**	Parsimony
	Statistical Approaches: Likelihood, Bayesian	Evolutionary Models
Distances	**Distance Methods**	

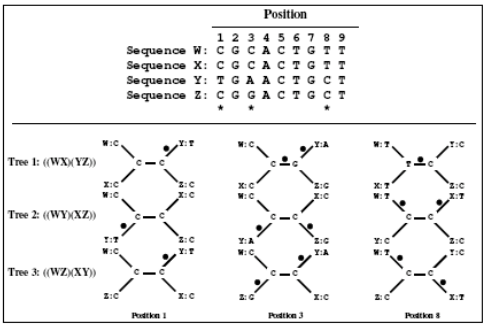
Maximum Parsimony I

A rule in science and philosophy stating that entities should not be multiplied needlessly.

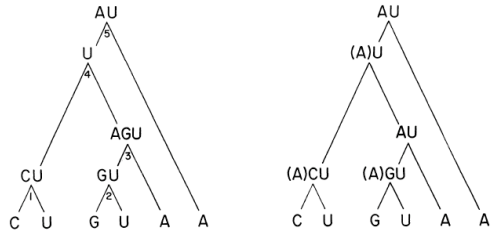
This rule is interpreted to mean that the simplest of two or more competing theories is preferable and that an explanation for unknown phenomena should first be attempted in terms of what is already known.

Also called law of parsimony. (Ockham's razor, ca 1285-1350)

Maximum Parsimony II



Maximum Parsimony III



Maximum Parsimony IV

Find the tree τ that minimizes the following expression:

$$L(\tau) = \sum_{k=1}^{B} \sum_{j=1}^{L} \omega_j \cdot \mathrm{diff}(x_{k'j}, x_{k''j})$$

where `diff` measures the distance between two characters
ω_j is an alignment specific weight factor
L alignment length
B number of branches in the tree
k' and *k''* edges of a branch

Maximum Parsimony V

Typical cost matrices for `diff`:

A =

	A	*C*	*G*	*T*
A	–	1	1	1
C	1	–	1	1
G	1	1	–	1
T	1	1	1	–

B =

	A	*C*	*G*	*T*
A	–	5	1	5
C	5	–	5	1
G	1	5	–	5
T	5	1	5	–

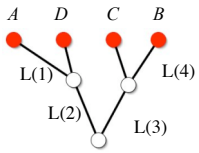
Distance based methods I:

seq 1	a	g	c	t	t	a	c	c	t	g	t	t	a	c	t
seq 2	c	g	t	a	a	a	t	t	t	c	c	c	g	a	t
seq 3	c	g	c	a	a	g	t	t	t	c	c	c	g	a	t
seq 4	c	a	c	t	t	a	t	t	a	g	t	c	a	a	c

$\downarrow (d_{ij})_{i,j=1,\ldots,4}$

	Seq 1	Seq 2	Seq 3	Seq 4
Seq 1	0	11	11	8
Seq 2	11	0	2	10
Seq 3	11	2	0	9
Seq 4	8	10	9	0

Distance based methods II:

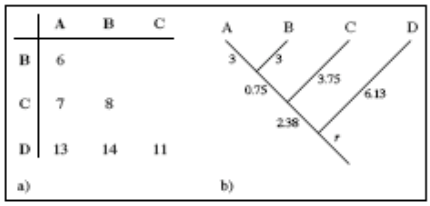


Find branch lengths L(b) such that the sum of the branch lengths connecting any two leaves gets close to the measured distances between all pairs of leaves. That is

$D_{measured}(A,B) = L(1)+L(2)+L(3)+L(4)$

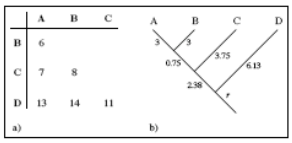
Distance based methods III:

Clustering methods: UPGMA = Unweighted Pair Group Methods using Arithmetic means.



Distance based methods IV:

Clustering methods work well, if sequences evolve according to a molecular clock



or equivalently: if the ultrametric inequality is holds:

$$d(A,B) \le \max\{d(A,C),d(B,C)\}$$

for each triple (*A*,*B*,*C*)

Distance based methods V:

Theorem: Four-Point-Condition

$(d_{i,j})_{i,j=1,..,n}$ is representable as a tree, if and only if

$$d(u,v)+d(x,z) \le \max\{d(u,x)+d(v,z),d(u,z)+d(v,x)\}$$

for all

$$u,v,x,z \in \{1,2,...,n\}$$

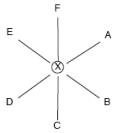
or equivalently:

For all sets of four elements exists a labelling of the elements, say *A,B,C,D* such that

$$d(A,B)+d(C,D) \le d(A,C)+d(B,D) = d(A,D)+d(B,C)$$

Distance based methods: Neighbor joining I

1. begin with **star tree**:



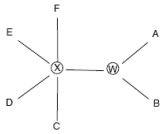
2. compute for each pair (*1,2*) the **net-divergence**

$$\frac{1}{2(N-2)}\sum_{k=3}^{N}(D_{1k}+D_{2k})+\frac{1}{2}D_{12}+\frac{1}{N-2}\sum_{3\le i<j}D_{ij}. \quad (1)$$

3. take the pair (*A*,*B*) that minimizes Eq. (1)

Distance based methods: Neighbor joining II

4. cluster (*A*,*B*) and define an interior node *W*



5. compute branch lengths for the external edges:

$$L(A,W) = \frac{1}{2}\left(D(A,B)+\frac{1}{m-2}\sum_{k=1}^{m}D(A,k)-D(B,k)\right)$$

$$L(B,W) = \frac{D(A,B)}{2} - L(A,W)$$

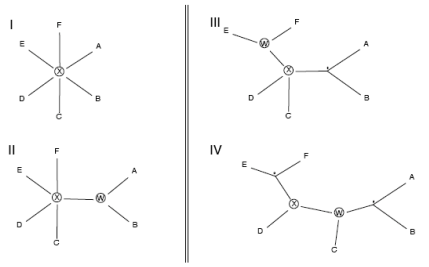
Distance based methods: Neighbor joining III

6. compute distance *W* to the remaining *m-2* leaves:

$$D(W,k) = \frac{1}{2}\left(D(A,k)+D(B,k)-D(A,B)\right)$$

7. continue with the reduced set of leaves

Distance based methods: Neighbor joining IV



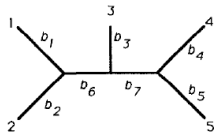
Distance based methods: Least Square I

Find a tree τ that minimizes

$$S(\tau) = \sum_{i,k} \left(\rho(i,k) - D(i,k)\right)^2$$

where $\rho(i,k)$ is the length of the unique path connecting leaves i and k in the tree.

Distance based methods: Least Square II



$$\begin{aligned}
d_{12} &= b_1 + b_2 \\
d_{13} &= b_1 + b_3 + b_6 \\
d_{14} &= b_1 + b_4 + b_6 + b_7 \\
d_{15} &= b_1 + b_5 + b_6 + b_7 \\
d_{23} &= b_2 + b_3 + b_6 \\
d_{24} &= b_2 + b_4 + b_6 + b_7 \\
d_{25} &= b_2 + b_5 + b_6 + b_7 \\
d_{34} &= b_3 + b_4 + b_7 \\
d_{35} &= b_3 + b_5 + b_7 \\
d_{45} &= b_4 + b_5
\end{aligned}$$

Distance based methods: Least Square III

$$\begin{aligned}
d_{12} &= b_1 + b_2 \\
d_{13} &= b_1 + b_3 + b_6 \\
d_{14} &= b_1 + b_4 + b_6 + b_7 \\
d_{15} &= b_1 + b_5 + b_6 + b_7 \\
d_{23} &= b_2 + b_3 + b_6 \\
d_{24} &= b_2 + b_4 + b_6 + b_7 \\
d_{25} &= b_2 + b_5 + b_6 + b_7 \\
d_{34} &= b_3 + b_4 + b_7 \\
d_{35} &= b_3 + b_5 + b_7 \\
d_{45} &= b_4 + b_5
\end{aligned}$$

$$\boldsymbol{d} = \boldsymbol{A}\boldsymbol{b}$$

$$\boldsymbol{A} = \begin{bmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 1 & 0 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 0
\end{bmatrix},$$

Distance based methods: Least Square IV

$$\boldsymbol{d} = \boldsymbol{A}\boldsymbol{b}$$

Least square estimates of the branch lengths

$$\hat{\boldsymbol{b}} = (\boldsymbol{A}^t\boldsymbol{A})^{-1}\boldsymbol{A}^t\boldsymbol{d}$$