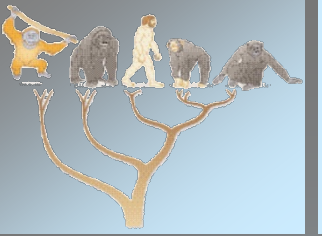
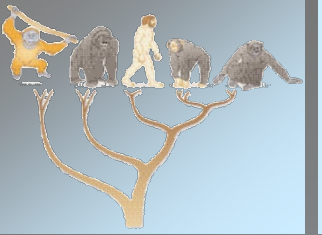


Sequence Analysis and Phylogenetics

Part 4

Sepp Hochreiter



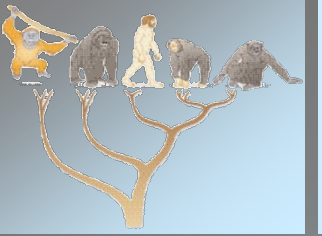
Klausur

Mo. 27.01.2014

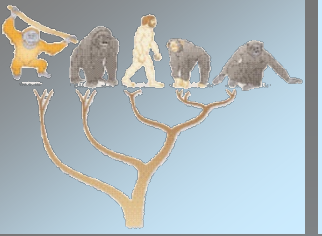
time: 15:30 – 17:00

room: S2 0120

register: Kusss



Contents



Motivation

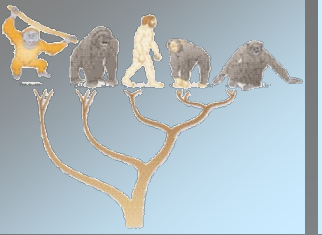
central field in biology: relation between species in form of a tree

Root: beginning of life
Leaves: *taxa* (current species)
Branches: relationship “is ancestor of" between nodes
Node: split of a species into two

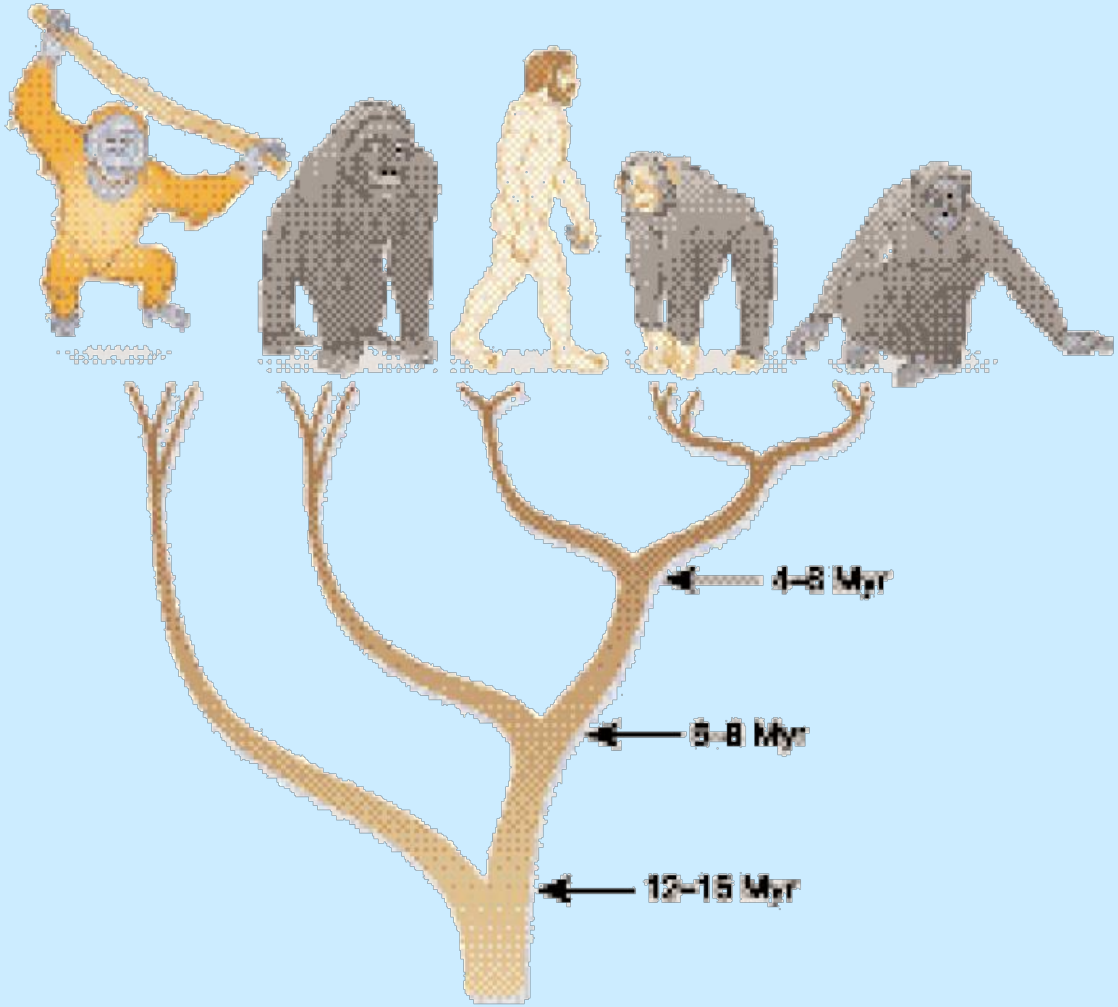
phylogeny (phylo = tribe and genesis)

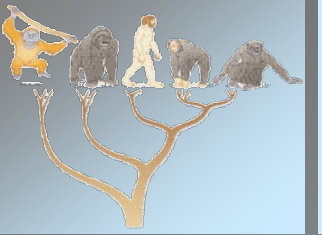
Cladistic trees: conserved characters
Phenetic trees: measure of distance between the leaves of the tree
(distance as a whole and not based on single features)

Phenetic problems: simultaneous development of features and different evolution rates
Convergent evolution e.g. finding the best form in water

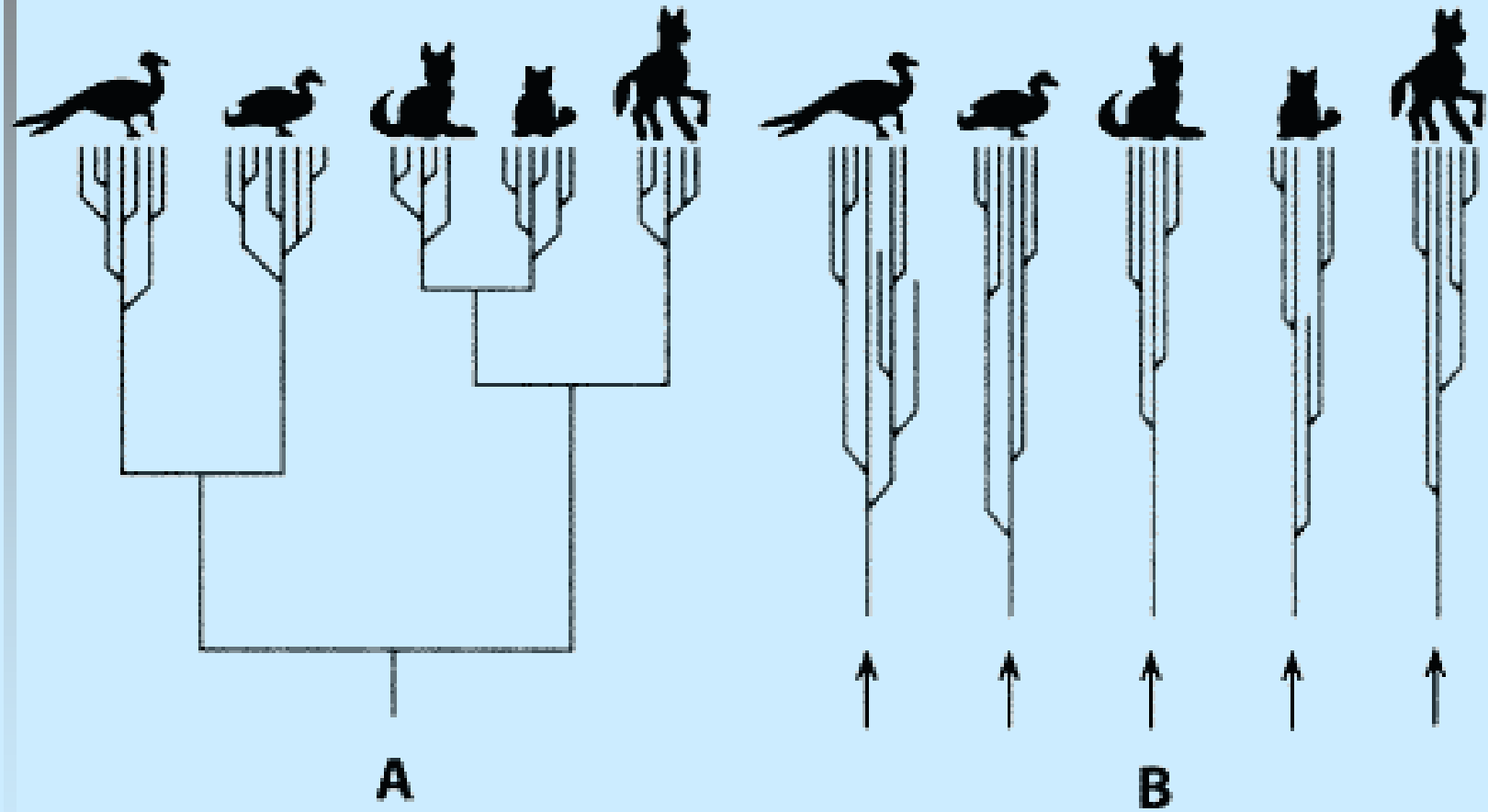


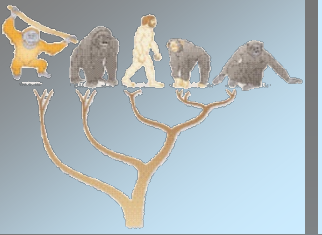
Motivation



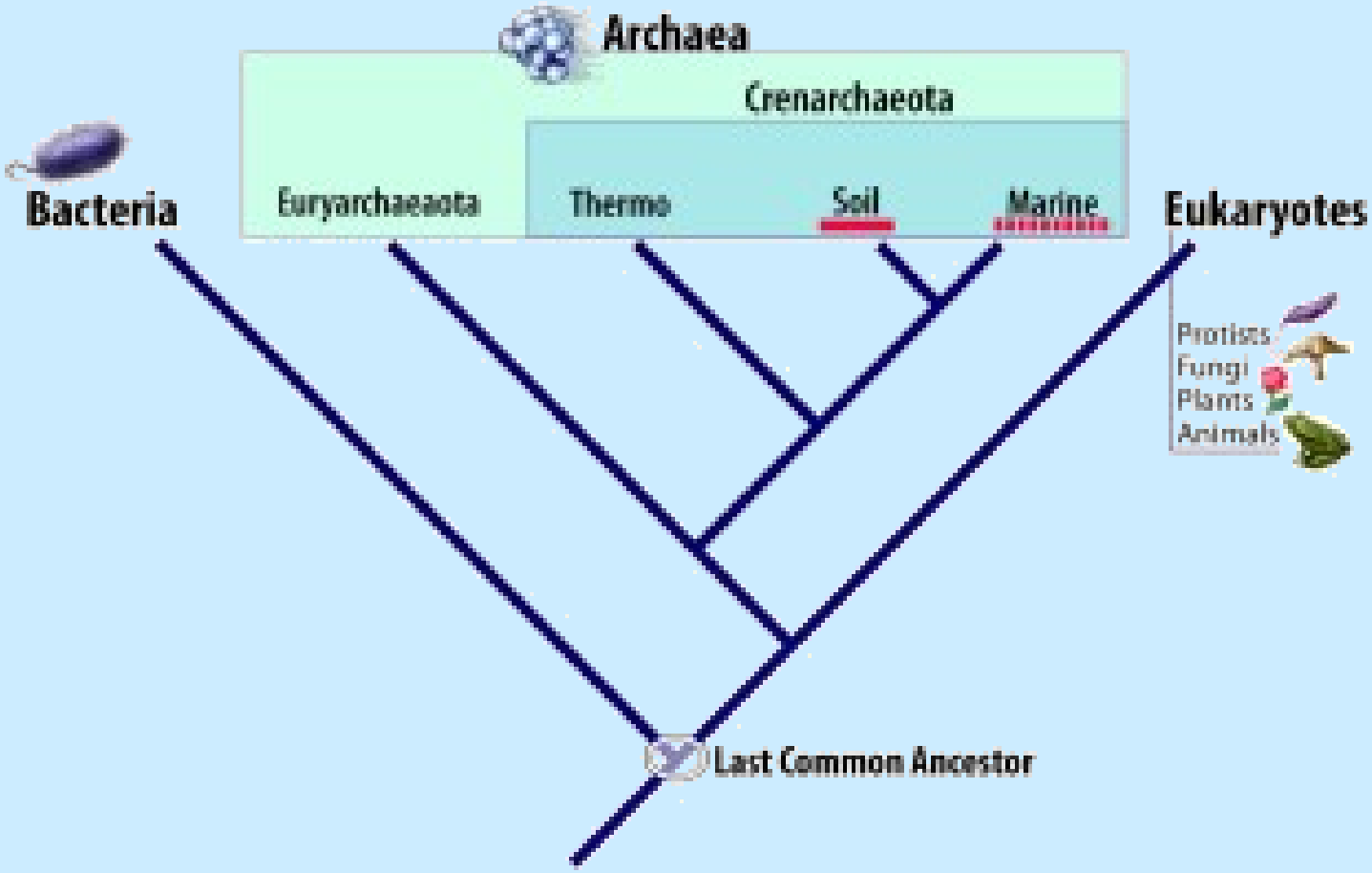


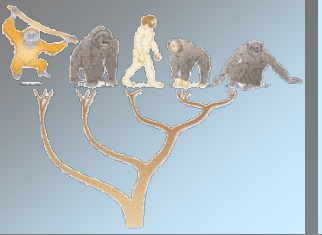
Motivation





Motivation

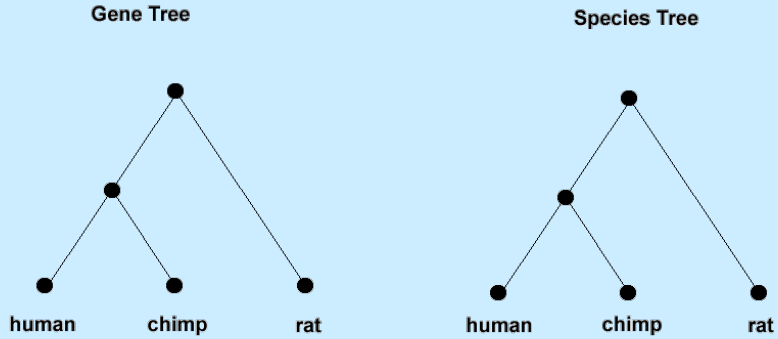




Molecular Phylogenies

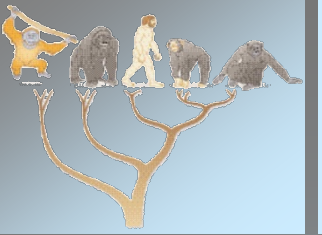
Not characteristics like wings, feathers, (morphological) differences between organisms: RNA, DNA, and proteins

α-hemoglobin



Molecular phylogenetics

- is more precise can also distinguish bacteria or viruses
- connects all species by DNA
- uses mathematical and statistical methods
- is model-based as mutations can be modeled
- detects remote homologies

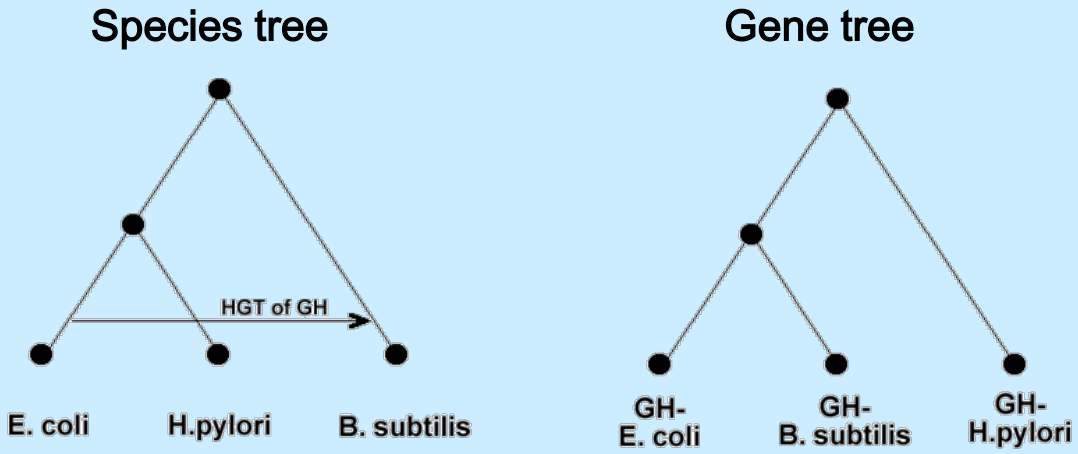


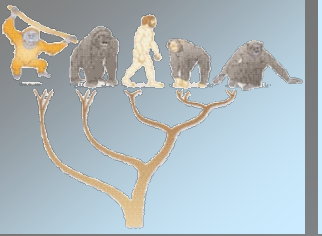
Molecular Phylogenies

Difficulties in constructing a phylogenetic tree:

➥ different mutation rates

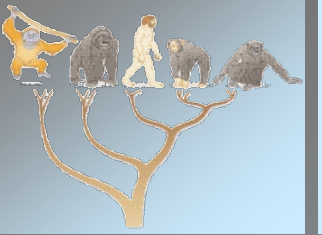
➥ horizontal transfer of genetic material (Horizontal Gene Transfer, Glycosyl Hydrolase from E.coli to B.subtilis)





Molecular Phylogenies

- Branches: time in number of mutations
- molecular clock: same evolution / mutation rate
- number of substitution: Poisson distribution
- mutation rate: equally distributed over the sequence



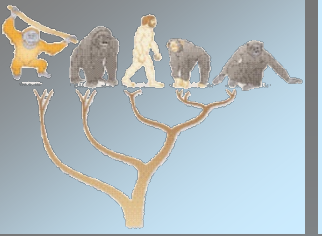
Methods

➥ choose the sequences e.g. rRNA (RNA of ribosomes) and mitochondrial genes (in most organism, enough mutations)

➥ pairwise and multiple sequence alignments

➥ method for constructing a phylogenetic tree
- distance-based
- maximum parsimony
- maximum likelihood

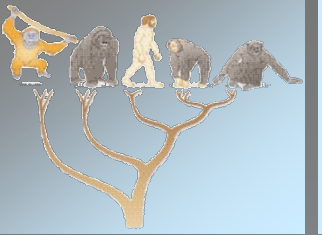
• Maximum parsimony: strong sequence similarities (few trees), few sequences
• Distance based (CLUSTALW): less similarity, many sequences
• Maximum likelihood: very variable sequences, high computational costs



Methods

Software

Name	Author	URL
PHYLIP	Felsenstein 89,96	http://evolution.genetics.washington.edu/phylip.html
PAUP	Sinauer Asssociates	http://www.lms.si.edu/PAUP

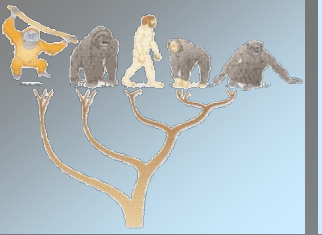


Maximum Parsimony

- minimize number of mutations
- mutations are branches in the tree
- tree explains the evolution of the sequences

- surviving mutations are rare → tree with minimal mutations is most likely explanation

- maximum parsimony PHYLIP programs: DNAPARS, DNAPENNY, DNACOMP, DNAMOVE, and PROTPARS



Tree Length

- *maximum parsimony tree:* tree with smallest tree length

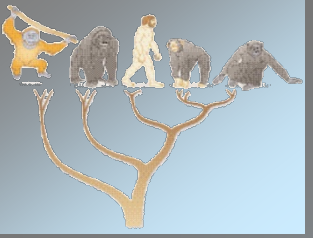
- *tree length*: number of substitutions in the tree

- Example protein triosephosphate isomerase for the taxa "Human", "Pig", "Rye", "Rice", and "Chicken":

Human	ISPGMI
Pig	IGPGMI
Rye	ISAEQL
Rice	VSAEML
Chicken	ISPAMI

- If we focus on column 4:

Human	G
Pig	G
Rye	E
Rice	E
Chicken	A



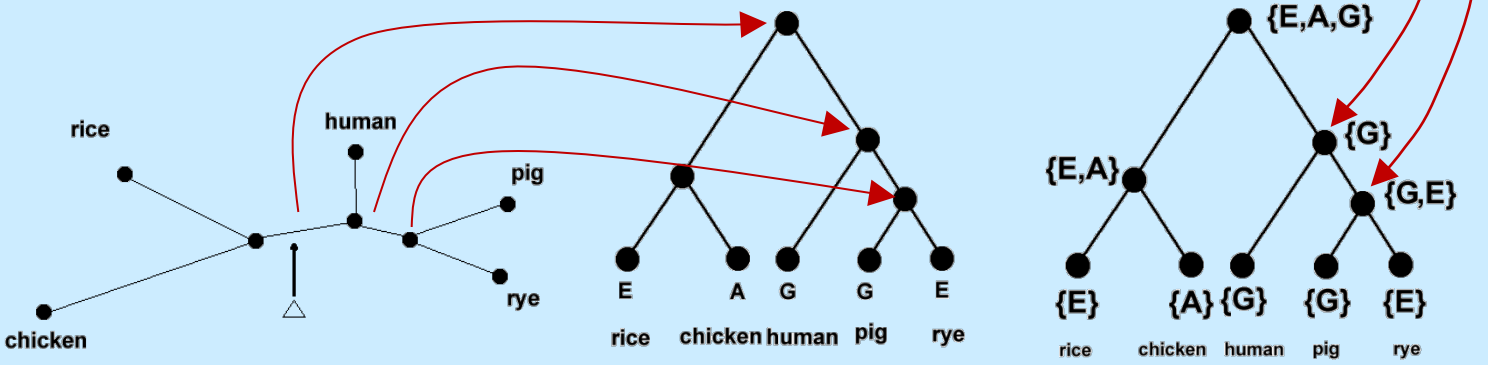
Tree Length

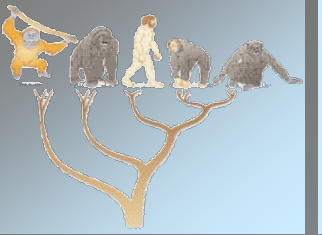
Fitch (71) alg. for computing the tree length with taxa at leaves:

1. Root node added to an arbitrary branch
2. bottom-up pass: sets of symbols (amino acids) for a hypothetical sequence at this node.
   Minimize the number mutations by maximal agreement of the subtrees ➔ avoid a mutation at the actual node

$$m_{12} = \begin{cases} \{\text{“leave symbol”}\} & \text{if} \quad m_1 = m_2 = \emptyset \\ m_1 \cup m_2 & \text{if} \quad m_1 \cap m_2 = \emptyset \\ m_1 \cap m_2 & \text{if} \quad m_1 \cap m_2 \neq \emptyset \end{cases}$$

In the first case m_{12} is leave, the second case enforces a mutation, and the third case avoids a mutation



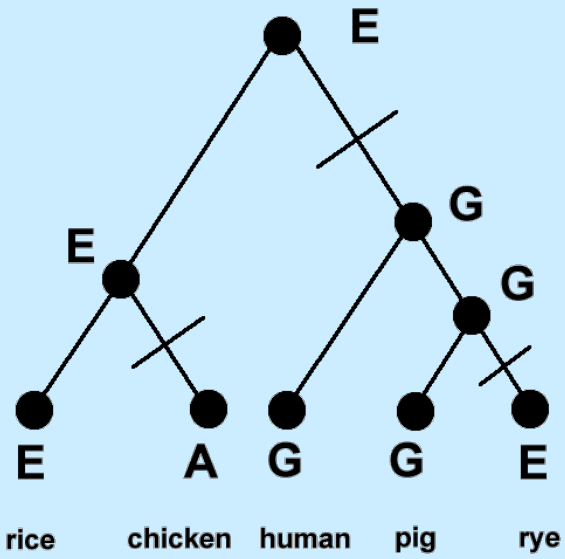


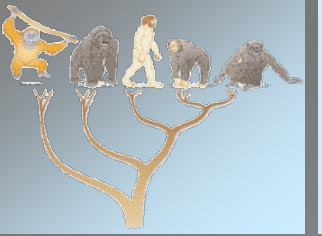
Tree Length

3. top town pass: hypothetical sequences at the interior nodes; counts the number of mutations

$$m_{1/2} = \begin{cases} x \in m_{1/2} \cap m_{12} & \text{if } m_{1/2} \cap m_{12} \neq \emptyset \\ x \in m_{1/2} & \text{if } m_{1/2} \cap m_{12} = \emptyset \end{cases}$$

$m_{1/2}$ means that the formula holds for m_1 and for m_2





Tree Length

Non-informative columns: not used

➥Columns with one symbol occur multiple an others only single (number of mutations independent of topology)

➥ Columns with only one symbol

minimal number of substitutions: m_i

maximal subs., star tree (center: most frequent symbol): g_i

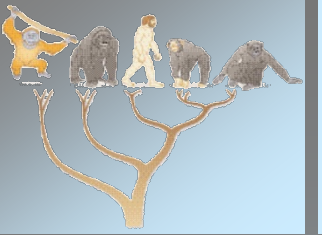
number of substitutions for the topology: s_i

consistency index : $c_i = \frac{m_i}{s_i}$

High values support the according tree as being plausible

retention index $r_i = \frac{g_i - s_i}{g_i - m_i}$

rescaled consistency index $rc_i = r_i \, c_i$

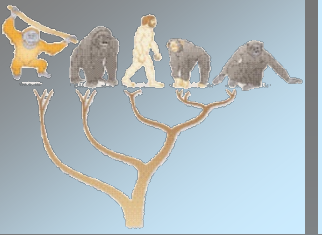


Tree Search

- few sequences: all trees and their length can be constructed
- for a larger number of sequences heuristics are used

Branch and Bound (Hendy & Penny, 82) for 20 and more taxa:

1. This step determines the addition order of the taxa as follows. First, compute the core tree of three taxa with **maximal** length of all three taxa trees. Next the taxa is added to one of the three branches which leads to **maximal** tree length. For the tree with four branches we determine the next taxa which leads to **maximal** tree length. (longest branches early in tree! Later bad trees are early found)
2. This step determines an upper bound for the tree length by either distance based methods (neighbor joining) or heuristic search (stepwise addition algorithm). (tight bound is important)



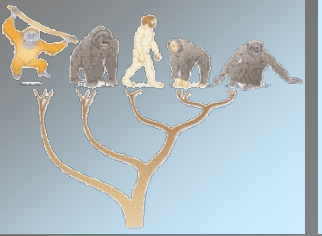
Tree Search

Branch and Bound (continued):

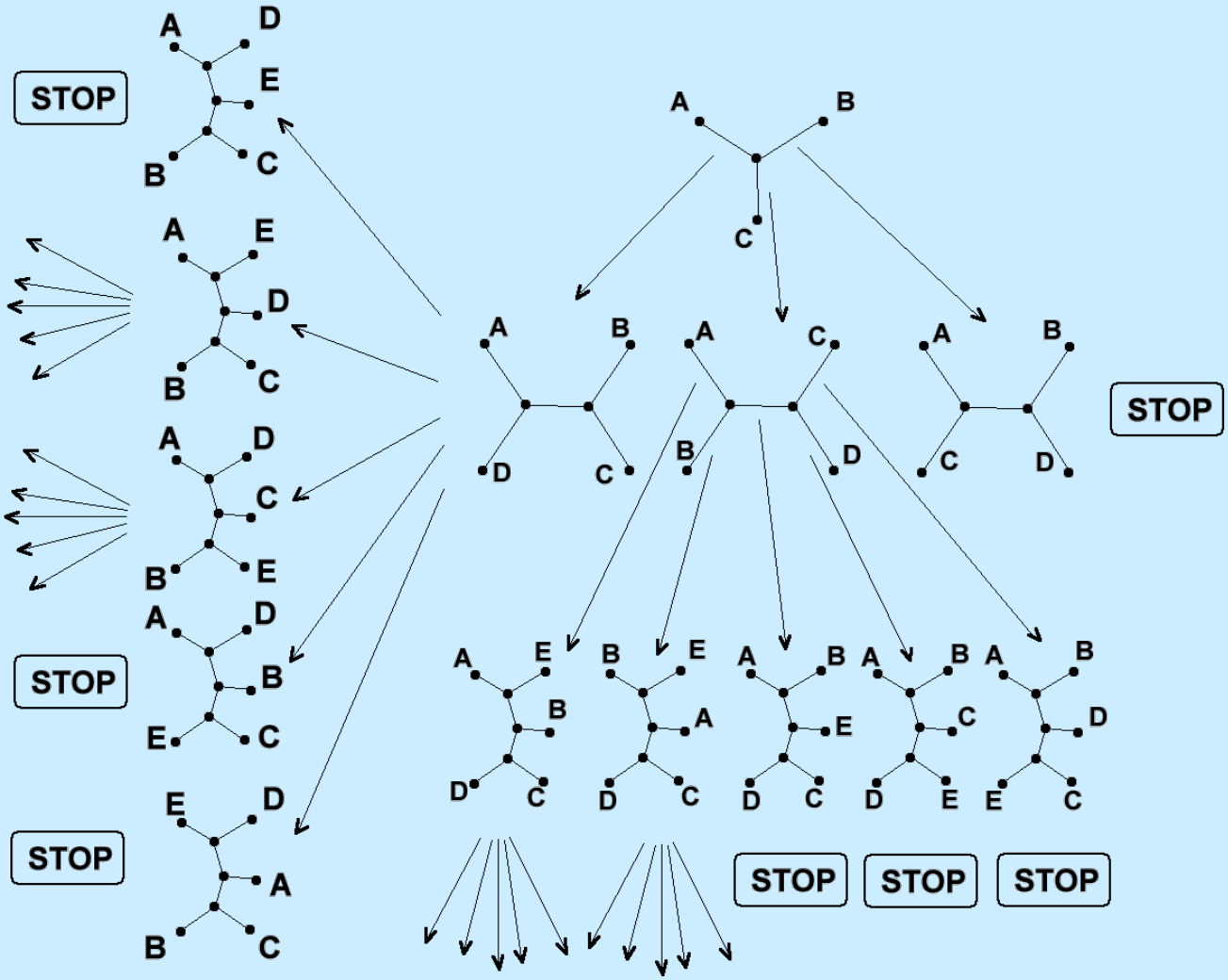
3. Start with the core tree of three taxa.

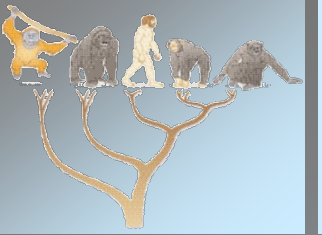
4. Construct new tree topologies by stepwise adding new taxa to the trees which do not possess a STOP mark. The next taxa is chosen according to the list in step 1 and added to each tree at all of its branches. Tree lengths are computed.

5. Assign STOP marks if upper bound is exceeded. Terminate if all trees possess a STOP mark. List of step 1 leads to many early STOP signals.



Tree Search



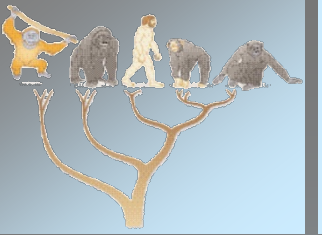


Tree Search

Heuristics for Tree Search (Step 2 in previous algorithm)

- *Stepwise Addition Algorithm*: extend only tree with shortest length. If all taxa are inserted then perform branch swapping (greedy, best first, here minimal tree not maximal).

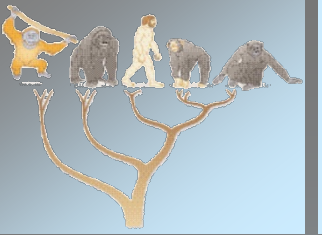
- *Branch Swapping*:
  (1) neighbor interchange: two taxa connected to the same node,
  (2) subtree pruning and regrafting: remove small tree and connect it with the root to a branch,
  (3) bisection-reconnection: remove branch to obtain two trees and reconnect them by inserting a new branch where each branch of the subtrees can be connected in contrast to (2) where the root is connected



Tree Search

Heuristics for Tree Search

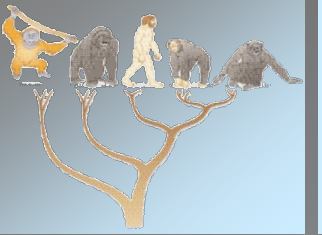
- *Branch and Bound Like*:
  Use step 1. of the branch-and-bound algorithm to obtain the minimal tree (not maximal!).
  Upper local bounds U_n for n taxa are constructed in this way.
  These upper bounds server for stopping signals.



Weighted Parsimony / Bootstrapping

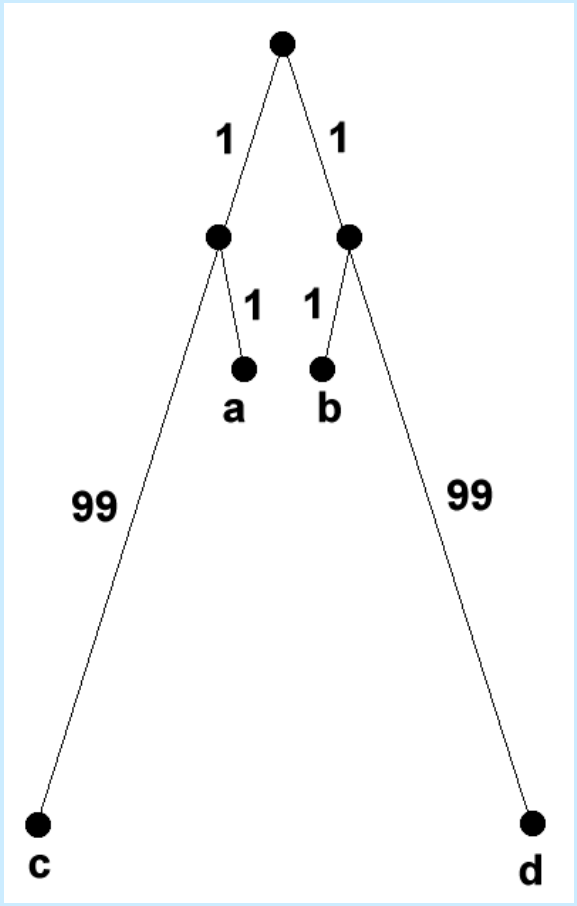
- Type of substitution is *weighted* according to PAM and BLOSUM matrices to address survival of substituition

- *Bootstrapping*
  - accesses the variability of the tree with respect to the data ("variance") and identifies stable substructures
  - is possible because the temporal order of the alignment columns does not matter
  - cannot access the quality of a method but only its robustness



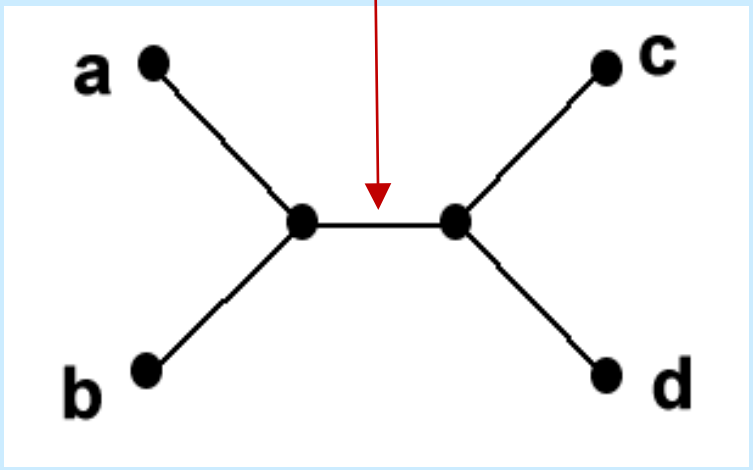
Inconsistency of Maximum Parsimony

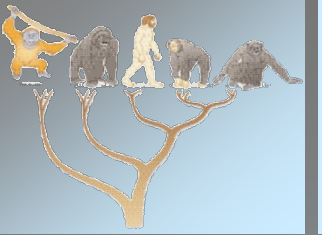
True Tree



Randomly shared between c and d

Maximum Parsimony





Inconsistency of Maximum Parsimony

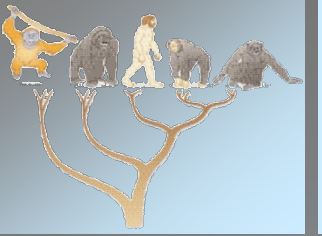
a and b are similar to each other and match to 99% whereas c and d are not similar to any other sequence and only match 5% by chance (1 out of 20).

Informative columns: only two symbols and each appears twice.
Probabilities of informative columns and their rate:

$$a_i = b_i,\ c_i = d_i: \quad \text{prob:} \quad 0.0495(0.99 \cdot 0.05) \quad \text{rate:} \quad 0.908$$
$$a_i = c_i,\ b_i = d_i: \quad \text{prob:} \quad 0.0025(0.05 \cdot 0.05) \quad \text{rate:} \quad 0.046$$
$$a_i = d_i,\ b_i = c_i: \quad \text{prob:} \quad 0.0025(0.05 \cdot 0.05) \quad \text{rate:} \quad 0.046$$

90% of the cases of informative columns we observe $c_i = d_i$.

Maximum parsimony will judge c and d as similar as a and b.



Distance-based Methods

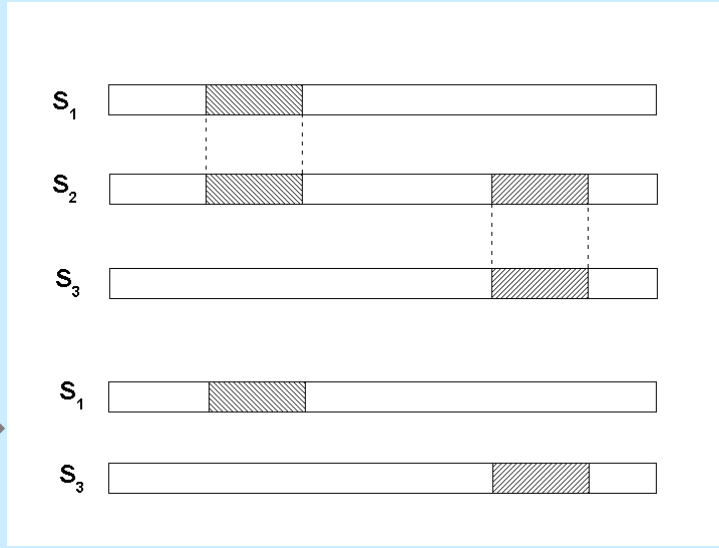
- matrix of pairwise distances between the sequences

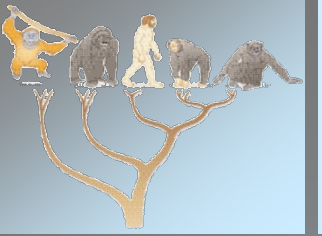
- A distance D is produced by a metric d (function) on objects x indexed by i,j,k: $D_{ij} = d(x_i, x_j)$

- metric d must fulfill

$$d(x_i, x_j) \geq 0 \ ,$$
$$d(x_i, x_j) = 0 \text{ for } i = j \ ,$$
$$d(x_i, x_j) = d(x_j, x_i) \ ,$$
$$d(x_i, x_j) \leq d(x_i, x_k) + d(x_k, x_j)$$

not all scoring schemes are a metric, e.g. the e-value →





UPGMA

Unweighted Pair Group Method using arithmetic Averages

(constructive clustering method based on joining pairs of clusters)

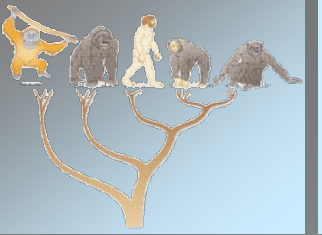
It works as follows:

1. each sequence i is a cluster c_i with one element n_i =1 and height l_i = 0. Put all i into a list.

2. Select cluster pair (i,j) from the list with minimal D_{ij} and create a new cluster c_k by joining c_i and c_j with height $l_k = D_{ij} / 2$ and number of elements $n_k = n_i + n_j$.

3. Compute the distance of c_k to c_m: $$D_{km} = \frac{n_i\ D_{mi} + n_j\ D_{mj}}{n_i + n_j}$$

4. Remove i and j from the list and add k to the list. If the list contains only one element then terminate else go to step 2.



UPGMA

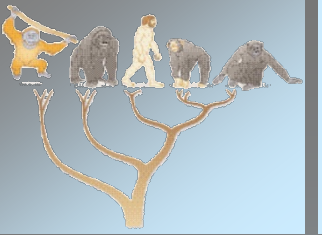
- assumption of constant rate of evolution in different lineages
- bootstrap can evaluate the reliability to data variation
- Positive interior branches contribute to the quality of the tree

UPGMA = average linkage clustering $D(X,Y) = \text{mean}(d(x,y))$

single linkage clustering $D(X,Y) = \min(d(x,y))$

complete linkage clustering $D(X,Y) = \max(d(x,y))$

$x \in X, y \in Y$



Least Squares

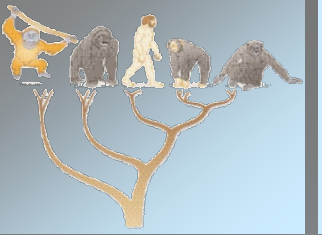
minimize $(D_{ij} - E_{ij})$, where E_{ij} is the sum of distances in the tree on the path from taxa i to taxa j (the path metric)

The objective is $\sum_{i<j} (D_{ij} - E_{ij})^2$

Fitch and Margoliash, 1967, introduced *weighted least squares*:

$$\sum_{i<j} (D_{ij} - E_{ij})^2 / D_{ij}^2$$

optimized under the constraint of nonnegative branch length



Least Squares

If matrix $\boldsymbol{A}$ is the binary topology matrix with $N(N-1)/2$ rows, one for each D_{ij}, and v columns for the v branches of the topology. In each row (i,j) all branches contained in the path from i to j are marked by 1 and all other branches are 0

$\boldsymbol{l}$ is the v -dimensional vector of branch weights, then $\boldsymbol{E} = \boldsymbol{A}\,\boldsymbol{l}$

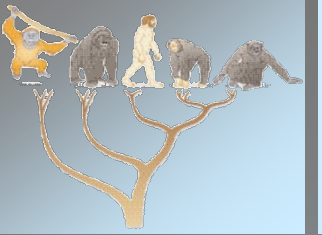
least squares assumption: D_{ij} deviates from E_{ij} according to a Gaussian distribution ε_{ij} with mean 0 and variance D^2_{ij}:

$$\boldsymbol{D} = \boldsymbol{E} + \boldsymbol{\epsilon} = \boldsymbol{A}\,\boldsymbol{l} + \boldsymbol{\epsilon}$$

maximum likelihood estimator (least squares) is

$$\hat{\boldsymbol{l}} = \left(\boldsymbol{A}^T\boldsymbol{A}\right)^{-1}\boldsymbol{A}^T\boldsymbol{D}$$

Gaussianity assumption: justified by sufficient large sequences → l_i are Gaussian and, therefore, also D_{ij}



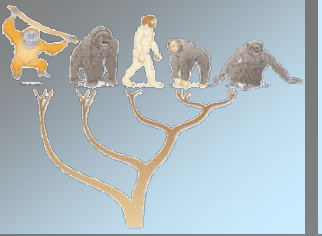
Minimum Evolution

The objective is the sum of branch length' l:

$$L = \sum_{ij} \hat{l}_{ij}$$

Given an unbiased branch length estimator, the expected value of L is smallest for the true topology independent of the number of sequences (Rzhetsky and Nei, 1993)

Minimum evolution is computational expensive



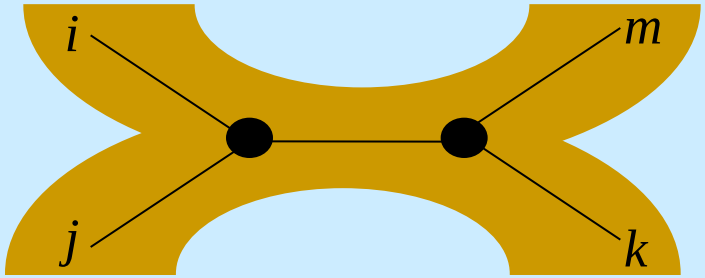
Neighbor Joining

The neighbor joining (Saitou and Nei, 1987) simplifies the minimum evolution method (for fewer than six taxa both methods give the same result)

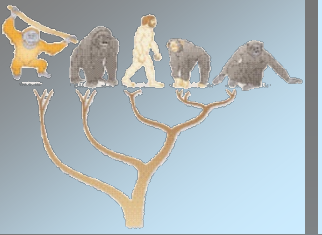
Neighbors: taxa that are connected by a single node

Additive metric d*:* any four elements fulfill

$$d(i,j) + d(k,m) \leq d(i,k) + d(j,m) = d(i,m) + d(j,k)$$

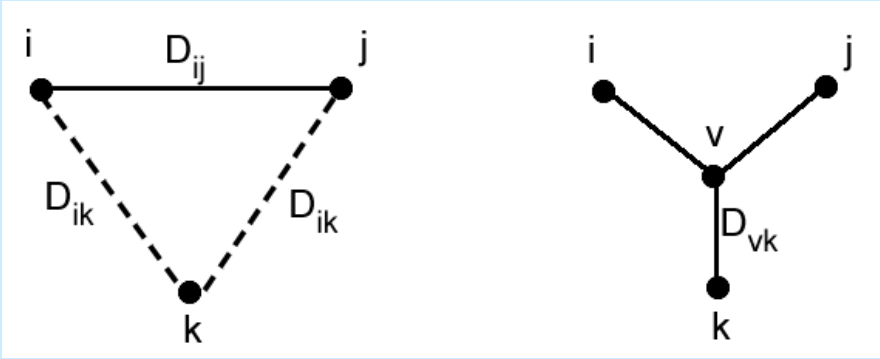


path metric (counting the branch weights) is an additive metric



Neighbor Joining

An additive metric can be represented by an unique additive tree



construction of an additive tree where a node v is inserted:

$$D_{vk} = \frac{1}{2}(D_{ik} + D_{jk} - D_{ij})$$

$$D_{iv} = \frac{1}{2}(D_{ij} + D_{ik} - D_{jk})$$

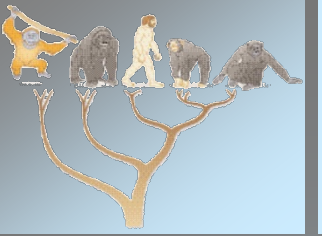
$$D_{jv} = \frac{1}{2}(D_{ij} + D_{jk} - D_{ik})$$

Path metric holds:

$$D_{ij} = D_{iv} + D_{vj}$$

$$D_{ik} = D_{iv} + D_{vk}$$

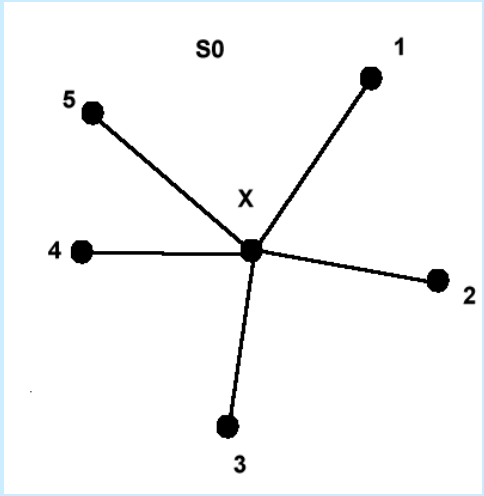
$$D_{jk} = D_{jv} + D_{vk}$$



Neighbor Joining

objective of the neighbor joining algorithm is S the sum of all branch length' l_{ij}

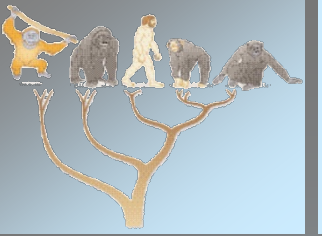
starts with a star tree



We assume N taxa with initial (star tree) objective S_0:

$$S_0 = \sum_{i=1}^{N} l_{iX} = \frac{1}{N-1} \sum_{i,j;i<j} D_{ij}$$

where the $\frac{1}{N-1}$ comes from the fact that $D_{ij} = l_{iX} + l_{Xj}$, therefore $l_{iX}$is part of $(N-1)$ distances D_{ij}

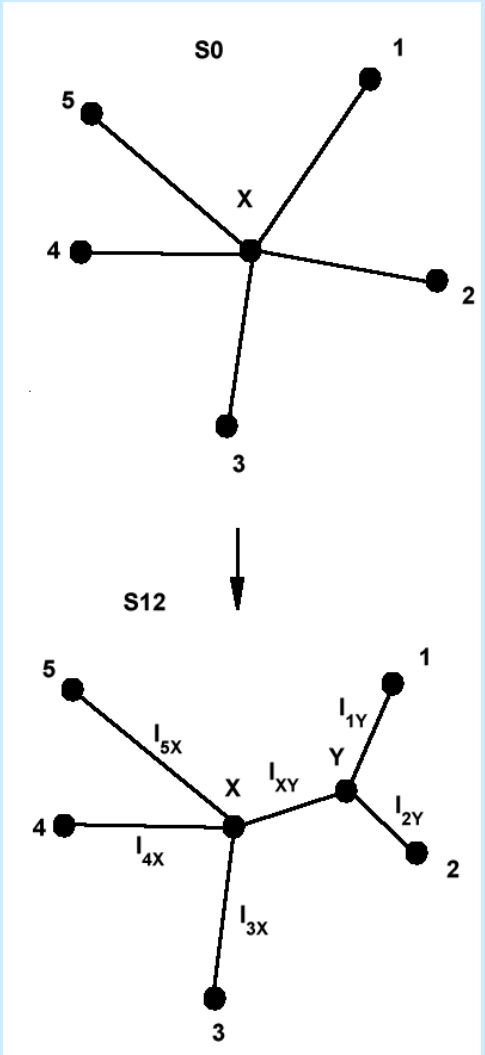


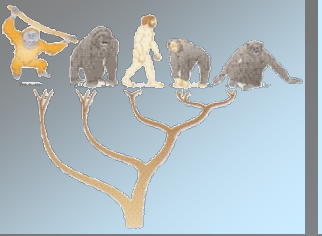
Neighbor Joining

In the next step taxa 1 and 2 are joined and a new internal node Y is introduced and l_{XY} computed as

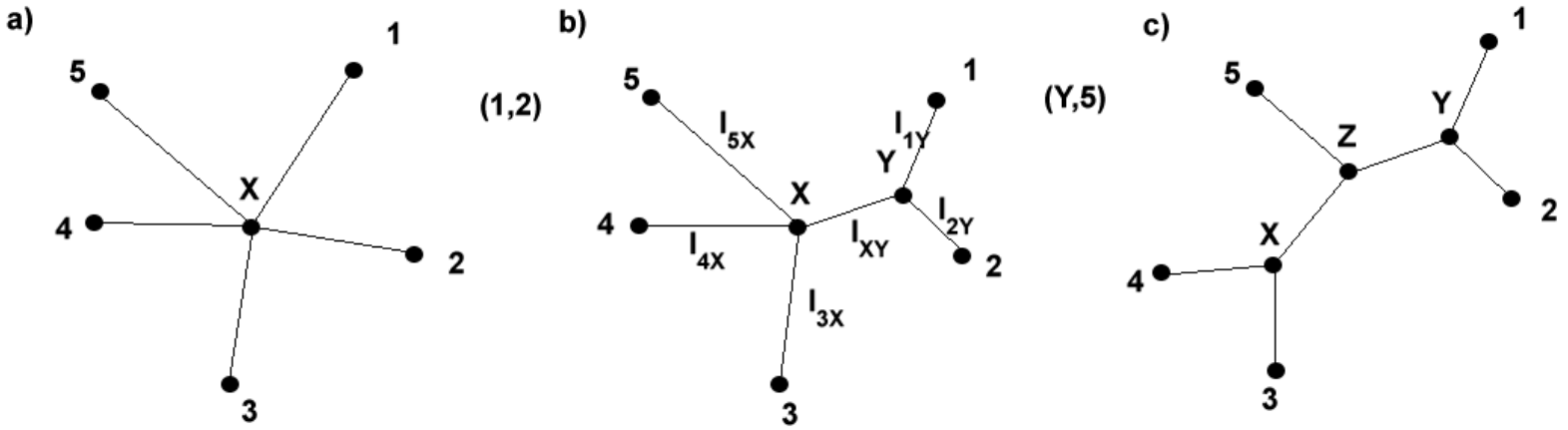
$$l_{XY} = \frac{1}{2(N-2)}\left(\sum_{i=3}^{N}(D_{1i}+D_{2i}) - (N-2)\,(l_{1Y}+l_{2Y}) - 2\sum_{i=3}^{N} l_{Xi}\right)$$

set all paths from i to j containing l_{XY} equal to D_{ij} and solve for l_{XY}. These are all path' from nodes 1 and 2 to $i \geq 3$. Therefore $(N-2)$ paths start from nodes 1 and 2, each, giving $2(N-2)$ paths. l_{1Y} is in all node 1 paths and l_{2Y} in all node 2 paths. The tail l_{iX} is in one node 1 and one node 2 path.
Above equation is obtained by averaging over these $2(N-2)$ equations for l_{XY} .





Neighbor Joining



$$\sum_{i=1}^{N} l_{Xi} = \frac{1}{N-3} \sum_{i,j;2<i<j} D_{ij}$$

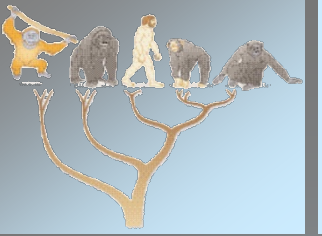
from l_{XY}

$$l_{1Y} + l_{2Y} - \frac{1}{2}(l_{1Y} + l_{2Y}) = \frac{1}{2}(l_{1Y} + l_{2Y}) = \frac{1}{2} D_{12}$$

$$\sum_{i=3}^{N} l_{Xi} - \frac{1}{N-2} \sum_{i=3}^{N} l_{Xi} = \frac{N-3}{N-2} \sum_{i=3}^{N} l_{Xi}$$

$$S_{12} = l_{1Y} + l_{2Y} + l_{XY} + \sum_{i=3}^{N} l_{Xi} =$$

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



Neighbor Joining

generalized from joining (1,2) to (k,l):

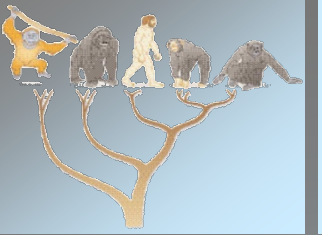
net divergences r_k are accumulated distances of k to all other taxa:

$$r_k = \sum_{i=1}^{N} D_{ki} \text{ giving } S_{kl} = \frac{2\sum_{i,j;i<j} D_{ij} - r_k - r_l}{2(N-2)} + \frac{D_{kl}}{2}$$

$\frac{2\sum_{i,j;i<j} D_{ij}}{2(N-2)}$ is constant for all objectives S_{kl}, therefore an equivalent objective is $\boxed{Q_{kl} = (N-2)\,D_{kl} - r_k - r_l}$

If k and l are evolutionary neighbors but D_{kl} is large due to fast evolution of k and/or l, then r_k and/or r_l are large and Q_{kl} small

Sum over all i and j, therefore subtract r_k and r_l



Neighbor Joining

The algorithm:

1. Given D_{ij} start with a star tree where the taxa are the leaves. Put all taxa in a set of objects.

2. For each leave i compute $r_i = \sum_{k=1}^{N} D_{ik}$

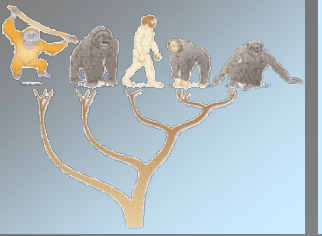
3. For each pair (i,j) compute $Q_{ij} = (N-2)\, D_{ij} - r_i - r_j$

4. Determine the minimal Q_{ij}. Join these (i,j) to new leave u. Compute new branch length' and new distances of u:

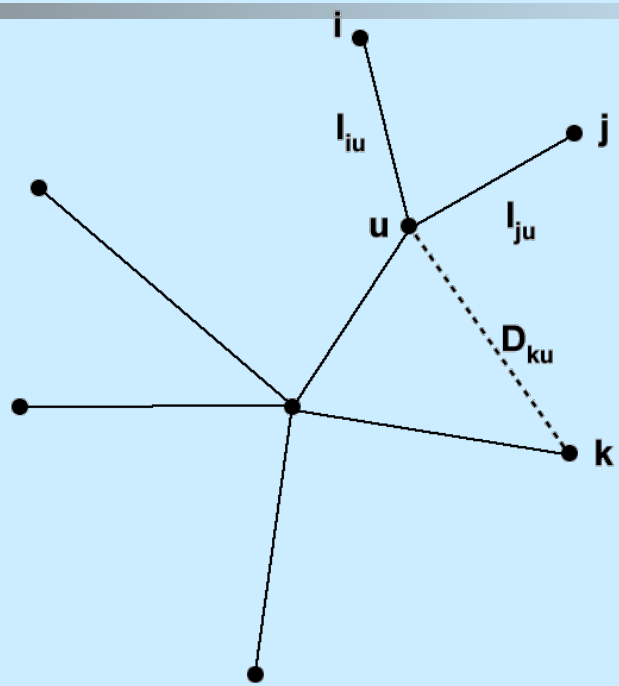
$$l_{iu} = \frac{D_{ij}}{2} + \frac{r_i - r_j}{2(N-2)} \qquad D_{ku} = \frac{D_{ik} + D_{jk} - D_{ij}}{2}$$

$$l_{ju} = D_{ij} - l_{iu}$$

Delete i and j from the set of objects and add.
Stop if the set of objects contains only u otherwise go to Step 1

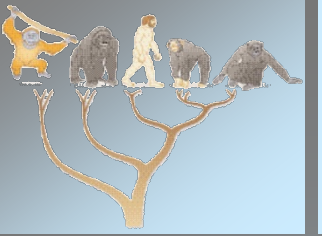


Neighbor Joining



Neighbor joining: $O(N^3)$ algorithm; for larger data sets

The formula for Q_{ij} accounts for differences in evolution rates
The objective S is only minimized approximatively
CLUSTALW uses neighbor-joining for multiple alignments



Distance Measures

we focus on nucleotides!

substitution rates:

Jukes Cantor

	A	T	C	G
A		α	α	α
T	α		α	α
C	α	α		α
G	α	α	α	

Kimura

	A	T	C	G
A		β	β	α
T	β		α	β
C	β	α		β
G	α	β	β	

Felsenstein / Tajima-Nei

	A	T	C	G
A		$\alpha\ g_T$	$\alpha\ g_C$	$\alpha\ g_G$
T	$\alpha\ g_A$		$\alpha\ g_C$	$\alpha\ g_G$
C	$\alpha\ g_A$	$\alpha\ g_T$		$\alpha\ g_G$
G	$\alpha\ g_A$	$\alpha\ g_T$	$\alpha\ g_C$	

Tamura

	A	T	C	G
A		$\beta\ (g_A + g_T)$	$\beta\ (g_G + g_C)$	$\alpha\ (g_G + g_C)$
T	$\beta\ (g_A + g_T)$		$\alpha\ (g_G + g_C)$	$\beta\ (g_G + g_C)$
C	$\beta\ (g_A + g_T)$	$\alpha\ (g_A + g_T)$		$\beta\ (g_G + g_C)$
G	$\alpha\ (g_A + g_T)$	$\beta\ (g_A + g_T)$	$\beta\ (g_G + g_C)$	

Hasegawa

	A	T	C	G
A		$\beta\ g_T$	$\beta\ g_C$	$\alpha\ g_G$
T	$\beta\ g_A$		$\alpha\ g_C$	$\beta\ g_G$
C	$\beta\ g_A$	$\alpha\ g_T$		$\beta\ g_G$
G	$\alpha\ g_A$	$\beta\ g_T$	$\beta\ g_C$	

Tamura-Nei

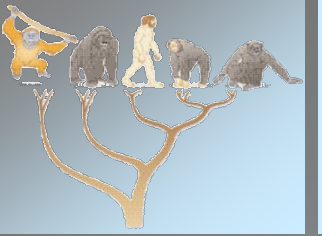
	A	T	C	G
A		$\beta\ g_T$	$\beta\ g_C$	$\alpha_{AG}\ g_G$
T	$\beta\ g_A$		$\alpha_{TC}\ g_C$	$\beta\ g_G$
C	$\beta\ g_A$	$\alpha_{TC}\ g_T$		$\beta\ g_G$
G	$\alpha_{AG}\ g_A$	$\beta\ g_T$	$\beta\ g_C$	

Reversible

	A	T	C	G
A		$\alpha_{AT}\ g_T$	$\alpha_{AC}\ g_C$	$\alpha_{AG}\ g_G$
T	$\alpha_{AT}\ g_A$		$\alpha_{TC}\ g_C$	$\alpha_{TG}\ g_G$
C	$\alpha_{AC}\ g_A$	$\alpha_{TC}\ g_T$		$\alpha_{CG}\ g_G$
G	$\alpha_{AG}\ g_A$	$\alpha_{TG}\ g_T$	$\alpha_{CG}\ g_C$	

General

	A	T	C	G
A		a_{12}	a_{13}	a_{14}
T	a_{21}		a_{23}	a_{24}
C	a_{31}	a_{32}		a_{34}
G	a_{41}	a_{42}	a_{43}	



Distance Measures

Jukes Cantor

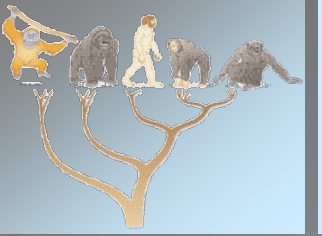
Mutation probability: $r = 3\alpha$

Identical positions of 2 seq. remain identical: $(1 - r)^2 \approx 1 - 2r$

different nucleotides will be identical: $\frac{2r}{3}$

One changes and the other not: $\alpha(1 - r) = \frac{r}{3}(1 - r)$

Two of these events: $\frac{2r}{3}(1 - r) \approx \frac{2r}{3}$



Distance Measures

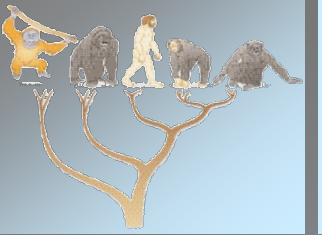
Jukes Cantor

difference equation: $q_{t+t} = (1 - 2\,r)\,q_t + \frac{2\,r}{3}(1 - q_t)$

$$q_{t+t} - q_t = \frac{2\,r}{3} - \frac{8\,r}{3}\,q_t$$

continuous model: $\dot{q} = \frac{2\,r}{3} - \frac{8\,r}{3}\,q$

The solution for $q(0) = 1$: $q(t) = 1 - \frac{3}{4}\left(1 - \exp\left(-\frac{8\,r\,t}{3}\right)\right)$



Distance Measures

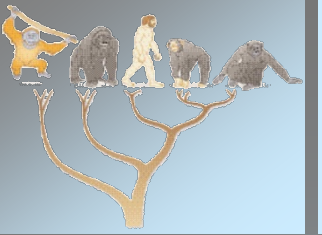
Jukes Cantor

The substitutions per position d for two sequences is $2r\,t$

$$d = -\frac{3}{4}\ln\left(1 - \frac{3}{4}p\right), \quad p = 1 - q$$

Estimating q and inserting in above equation: estimate d.

The variance of the estimate for d: $\mathrm{Var}(\hat{d}) = \frac{9p\,(1 - p)}{(3 - 4p)^2\,n}$



Distance Measures

Kimura

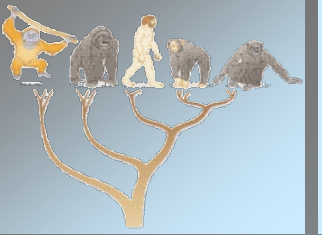
$$r = \alpha + 2\beta$$

group nucleotide pairs: $P = \{\mathrm{AG}, \mathrm{GA}, \mathrm{TC}, \mathrm{CT}\}$

$Q = \{\mathrm{AT}, \mathrm{TA}, \mathrm{AC}, \mathrm{CATG}, \mathrm{GT}, \mathrm{CG}, \mathrm{GC}\}$

$$P = \frac{1}{4}\left(1 - 2\,\exp\left(-4\,(\alpha + \beta)\,t\right) + \exp\left(-8\,\beta\,t\right)\right)$$

$$Q = \frac{1}{2}\left(1 - \exp\left(-8\,\beta\,t\right)\right)$$



Distance Measures

Kimura

$$d = 2\,r\,t = 2\,\alpha\,t + 4\,\beta\,t = -\frac{1}{2}\ln(1 - 2\,P - Q) - \frac{1}{2}\ln(1 - 2\,Q)$$

$$\mathrm{Var}(\hat{d}) = \frac{1}{n}\left(c_1^2\,P + c_2^2\,Q - (c_1\,P + c_2\,Q)^2\right)$$

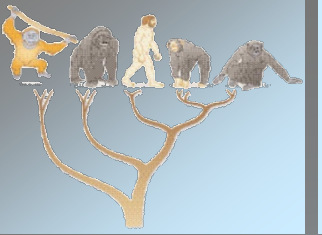
$$c_1 = (1 - 2\,P - Q)^{-1}$$

$$c_2 = \frac{1}{2}\left((1 - 2\,P - Q)^{-1} + (1 - 2\,Q)^{-1}\right)$$

transitional substitutions: $2\,\alpha\,t$
transversional substitutions: $4\,\beta\,t$

equilibrium frequency of each nucleotide: 0.25

However occurrence of GC *Drosophila* mitochondrial DNA is 0.1



Distance Measures

Felsenstein / Tajima-Nei

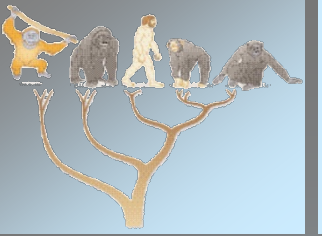
x_{ij}: relative frequency of nucleotide pair (i,j)

$$b = \frac{1}{2}\left(1 - \sum_{i=1}^{4} g_i^2 + \frac{p^2}{c}\right)$$

$$c = \sum_{i=1}^{3}\sum_{j=i+1}^{4} \frac{x_{ij}^2}{2\ g_i\ g_j}$$

$$d = -b\ \ln\left(1 - \frac{p}{b}\right)$$

$$\mathrm{Var}(\hat{d}) = \frac{b^2\ p\ (1 - p)}{(b - b)^2 n}$$



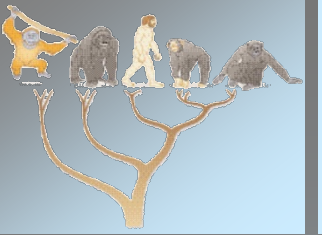
Distance Measures

Tamura

extends Kimura's model for GC content different from 0.5

$$d = -h \ln\left(1 - \frac{P}{h} - Q\right) - \frac{1}{2}(1 - h)\ln(1 - 2Q)$$

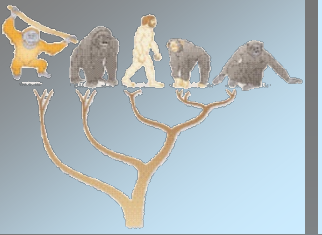
$$h = 2\theta(1 - \theta)$$



Distance Measures

Hasegawa (HKY)

hybrid of Kimuras and Felsenstein / Tajima-Nei: GC content and transition / transversion



Distance Measures

Tamura-Nei

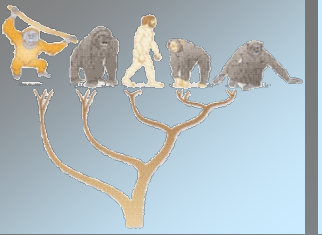
includes Hasegawa's model

$$c_1 = \frac{2\ g_A\ g_G}{g_R}$$

$$c_2 = \frac{2\ g_T\ g_C}{g_Y}$$

$$d = -\ c_1 \ln\left(1 - c_1^{-1}\ P_1 - (2\ g_R)^{-1}\ Q\right) -$$
$$c_2 \ln\left(1 - c_2^{-1}\ P_2 - (2\ g_Y)^{-1}\ Q\right) -$$
$$(2\ g_R\ g_Y - c_1\ g_Y - c_2\ g_R) \ln\left(1 - (2\ g_R\ g_Y)^{-1} Q\right)$$

P_1: proportion of transitional differences between `A` and `G`
P_2: proportion of transitional differences between `T` and `C`
Q: proportion of transversional differences

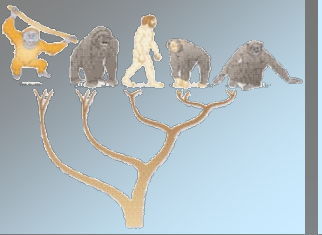


Maximum Likelihood Methods

Tree probability: product of the mutation rates in each branch
Mutation rate: product between substitution rate and branch length

$\boldsymbol{D}$: data, multiple alignment of N sequences (taxa)
$\boldsymbol{D}_k$: N-dimensional vector at position k of the multiple alignment
$\boldsymbol{A}$: tree topology (see least squares)
$\boldsymbol{l}$: vector of branch length
H: number of hidden nodes of the topology $\boldsymbol{A}$
$\mathcal{M}$: model for nucleotide substitution
$\mathcal{A}$: set of letters (e.g. the amino acids)

Hidden nodes are indexed from 1 to H
taxa are indexed from $H+1$ to $H+N$
Root node has index 1



Maximum Likelihood Methods

The likelihood of the tree at the k-th position:

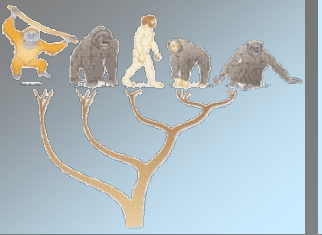
$$L(\boldsymbol{D}_k \mid \boldsymbol{l}, \boldsymbol{A}, \mathcal{M}) =$$

$$\sum_{h=1}^{H} \sum_{a_h \in \mathcal{A}} P_r(a_1) \prod_{i,j;\ 1 \le i \le H,\ i < j \le N+H,\ A_{ij}=1} P_{a_i a_j}(l_{ij})$$

$P_r(a_1)$: prior probability of the root node assigned with $a_1 \in \mathcal{A}$
$A_{ij} = 1$: indicates an existing branch $i \to j$

$P_{a_i a_j}(l_{ij})$: probability of branch length l_{ij} between a_i and a_j

hidden states are summed out
Prior $P_r(a_1)$ is estimated or given



Maximum Likelihood Methods

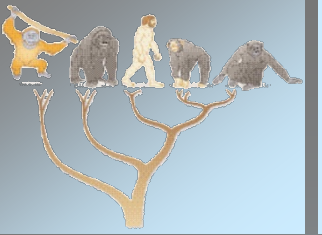
If $\mathcal{M}$ is the Felsenstein / Tajima-Nei equal-input model, the branch length probabilities are

$$P_{a_i a_i}(l_{ii}) = g_{a_i} + (1 - g_{a_i})\, e^{-l_{ii}}$$
$$P_{a_i a_j}(l_{ij}) = g_{a_j}\left(1 - e^{-l_{ij}}\right)$$

For $g_{a_i} = \frac{1}{4}$ and $l_{ij} = 4rt$ we obtain Jukes-Cantor: $P_{a_i a_i} = q$

reversible models: $g_{a_i}\, P_{a_i a_j}(l) = g_{a_j}\, P_{a_j a_i}(l)$

choice of the root does not matter because branch lengths count independent of their substitution direction



Maximum Likelihood Methods

$$L(\boldsymbol{D} \mid \boldsymbol{l}, \boldsymbol{A}, \mathcal{M}) = \prod_k L(\boldsymbol{D}_k \mid \boldsymbol{l}, \boldsymbol{A}, \mathcal{M})$$ (independent positions)

Felsenstein's (81) pruning algorithm to compute $L(\boldsymbol{D}_k \mid \boldsymbol{l}, \boldsymbol{A}, \mathcal{M})$

$P_i(a) = P_i(a \mid \boldsymbol{D}_k, \boldsymbol{l}, \boldsymbol{A}, \mathcal{M})$: probability of a letter a at node i

recursive formula:

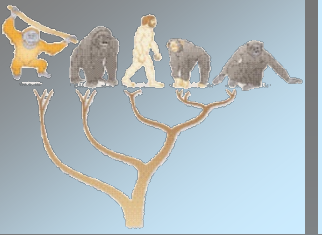
$$P_i(a_i) = \delta_{a_i \; D_{k(i-H)}} \quad \text{for} \quad i > H \;\; (i \text{ taxa})$$

$$\delta_{a\,b} = \begin{cases} 1 & \text{for} \quad a = b \\ 0 & \text{for} \quad a \neq b \end{cases}$$

$$P_i(a_i) = \prod_{j;\; A_{ij}=1} \left(\sum_{a_j \in \mathcal{A}} P_{a_i a_j}(l_{ij}) \;\; P_j(a_j) \right) \quad \text{for} \quad i \leq H \;\; (i \text{ hidden})$$

$P_i(a)$ computed by dynamic programming from leaves to root

Likelihood: $L(\boldsymbol{D}_k \mid \boldsymbol{l}, \boldsymbol{A}, \mathcal{M}) = \sum_{a_1 \in \mathcal{A}} P_r(a_1) \, P_1(a_1)$



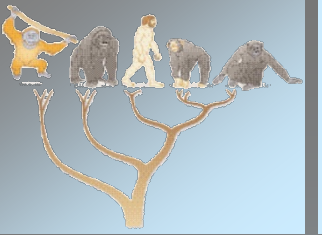
Maximum Likelihood Methods

best tree: both the branch length' and the topology optimized

- ➥ branch length‘: gradient based / EM (expectation-maximization)
- ➥ tree topology: Felsenstein (81) growing (constructive) algorithm start with 3 taxa, at *k*-th taxa test all ($2k$-5) branches for insertion further optimized by local changing the topology
- ➥ tree topology: small N all topologies can be tested, then local changes similar to parsimony tree

ML estimator is computationally expensive but unbiased (sequence length) and asymptotically efficient (minimal variance)

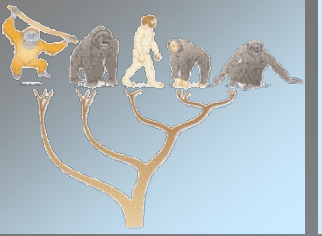
fast heuristics: Strimmer and v. Haeseler (96): all topologies of 4 taxa then build the final tree (software: http://www.tree-puzzle.de/)



Examples

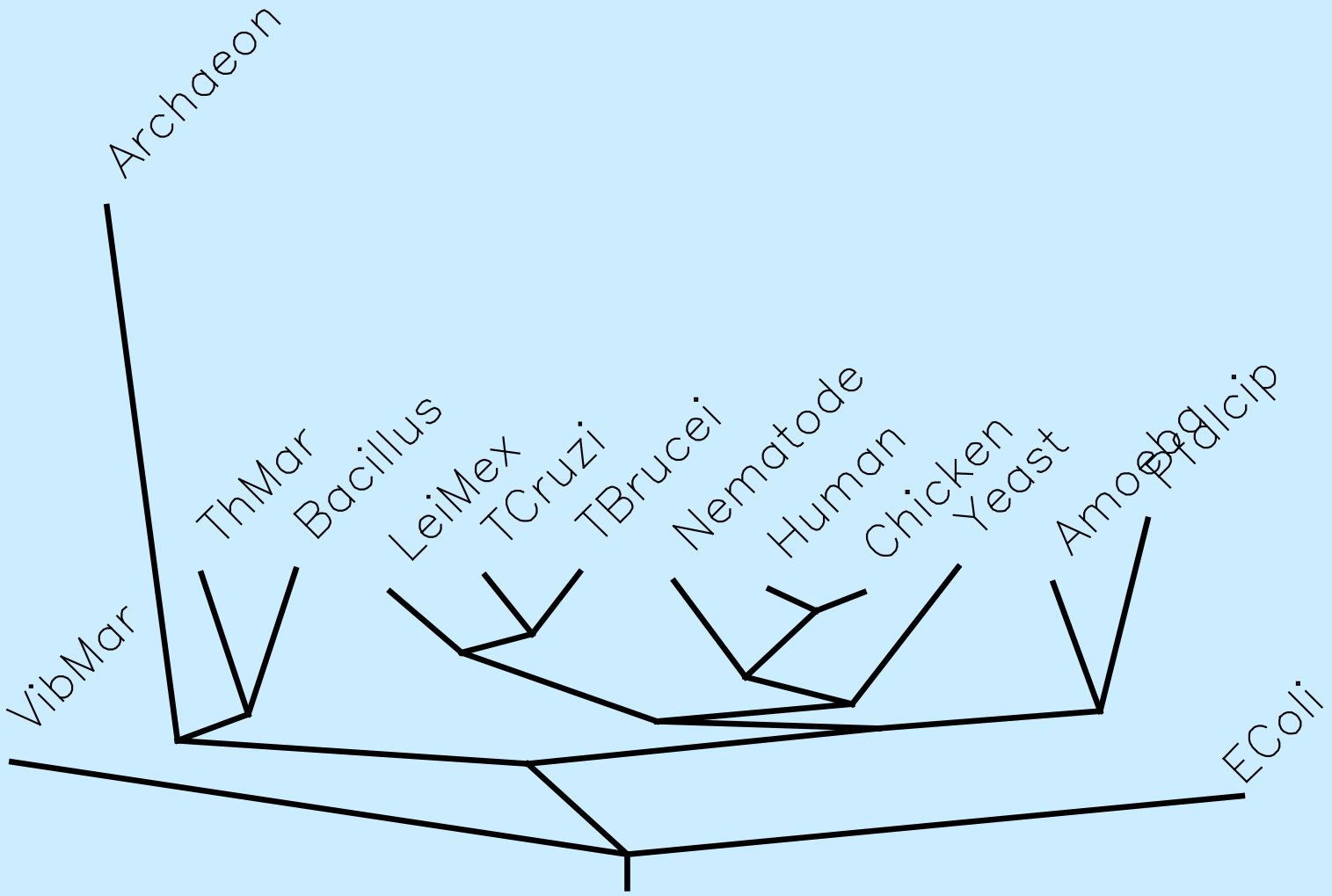
From triosephosphat isomerase of different species trees are constructed with PHYLIP (Phylogeny Inference Package) 3.5c

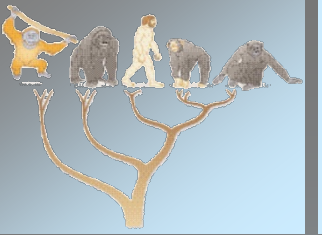
EColi	Escherichia coli	Bacterium
VibMar	Vibrio marinus	Bacterium
Chicken	Gallus gallus	Animal
Human	Homo sapiens	Animal
Nematode	Caenorhabditis elegans	Worm
Yeast	Saccharomyces cerevisiae	Yeast
Pfalcip	Plasmodium falciparum	single cell
Amoeba	Entamoeba histolytica	single cell
TBrucei	Trypanosoma brucei	single cell
TCruzi	Trypanosoma cruzi	single cell
LeiMex	Leishmania mexicana	single cell
Bacillus	Bacillus stearothermophilus	Bacterium
ThMar	Thermotoga maritima	Bacterium
Archaeon	Pyrococcus woesei	Archaeon



Examples

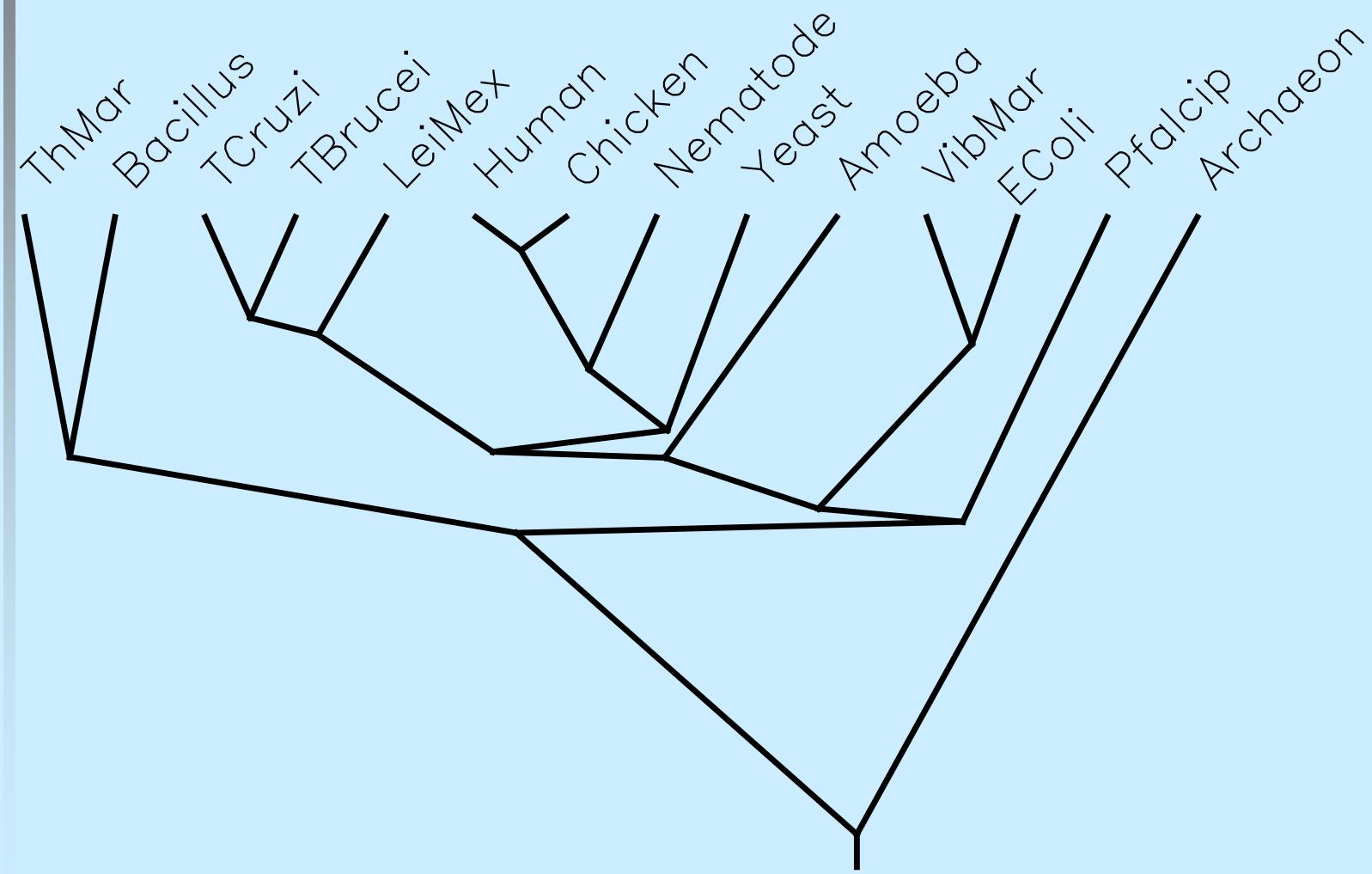
Fitch-Margoliash

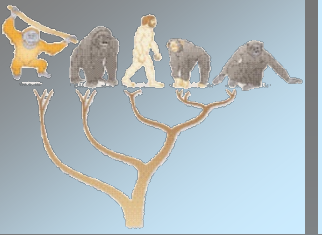




Examples

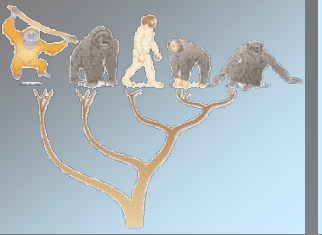
Fitch-Margoliash with assumption of a molecular clock (“kitsch”)





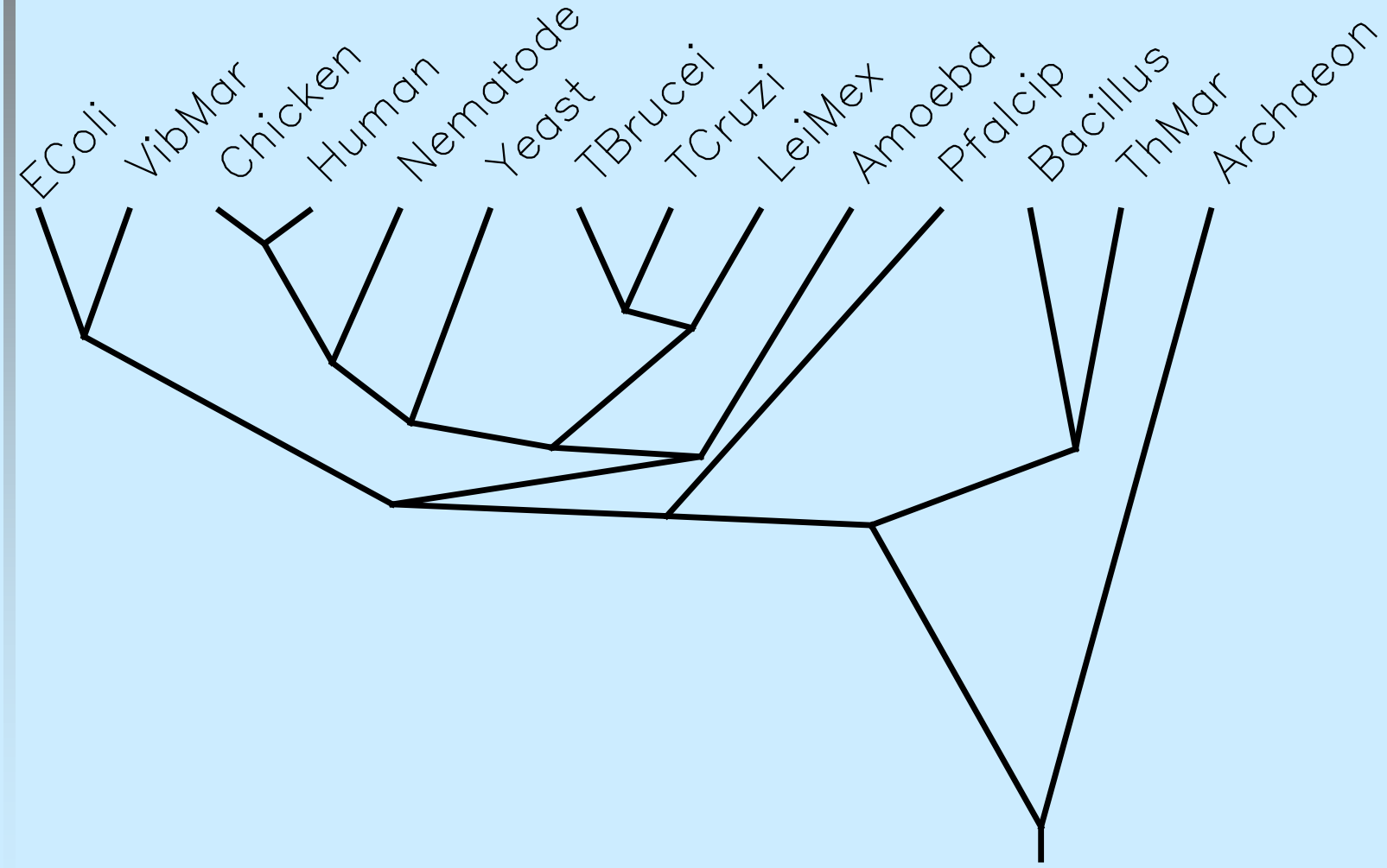
Examples

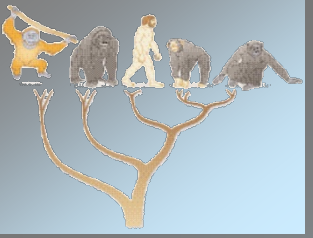
neighbor joining



Examples

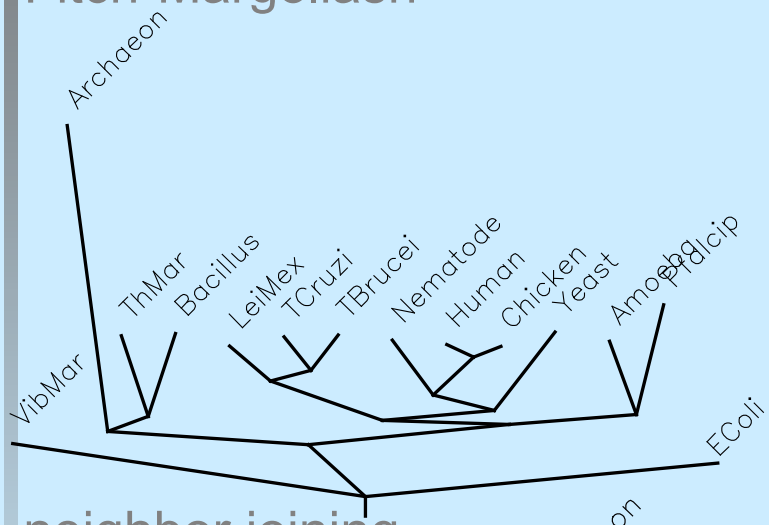
UPGMA



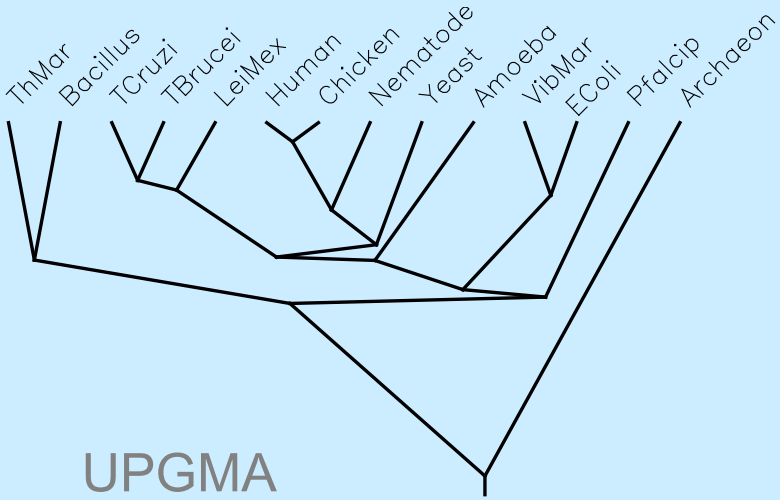


Examples

Fitch-Margoliash

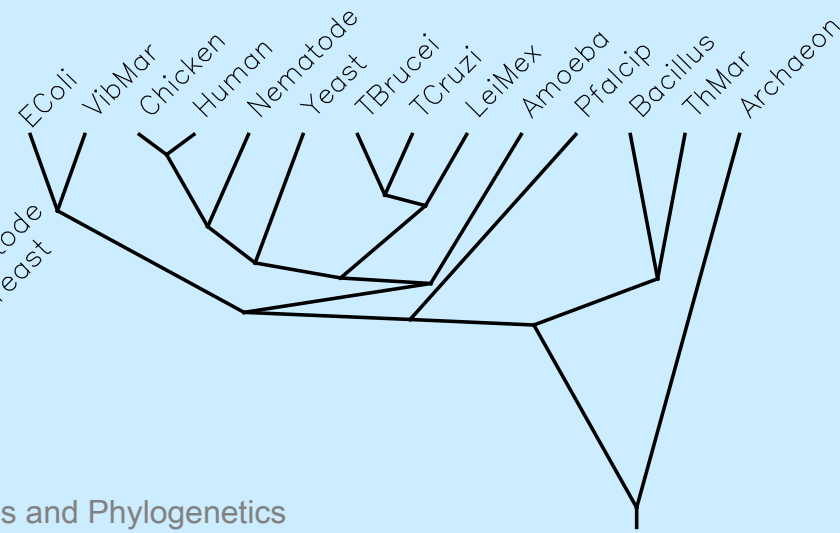


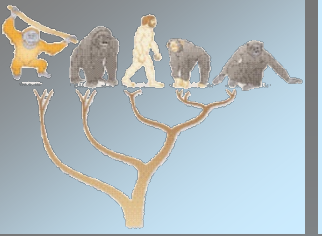
Kitsch



neighbor joining

UPGMA

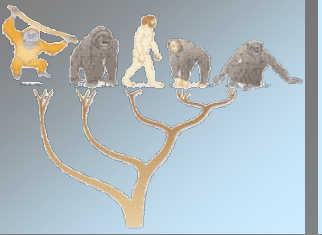




Examples

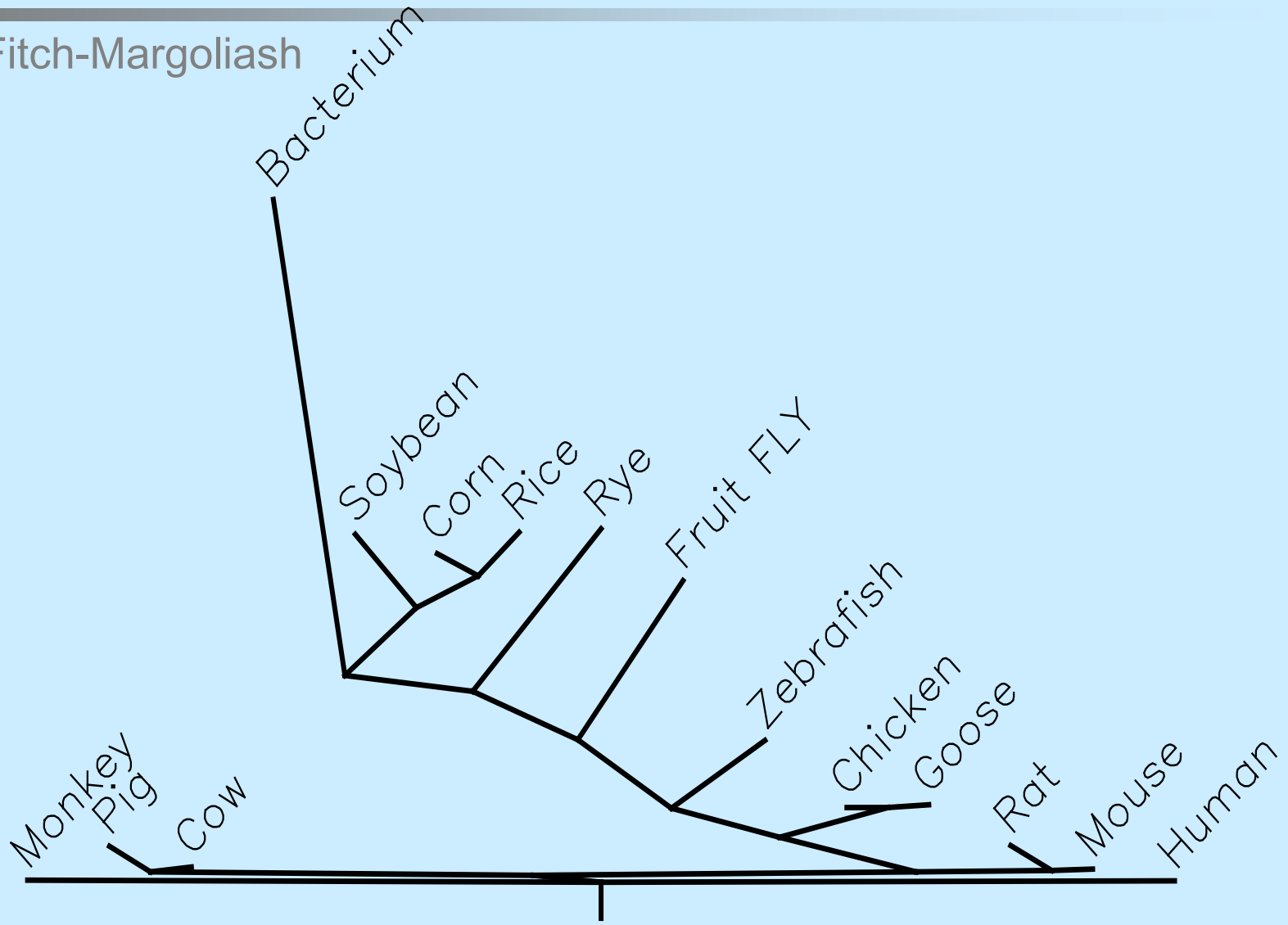
Phylogenetic tree based on triosephosphat isomerase for:

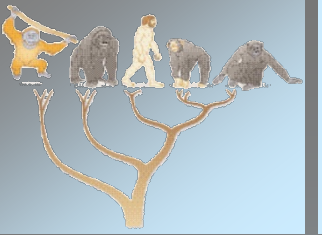
- Human
- Monkey
- Mouse
- Rat
- Cow
- Pig
- Goose
- Chicken
- Zebrafish
- Fruit FLY
- Rye
- Rice
- Corn
- Soybean
- Bacterium



Examples

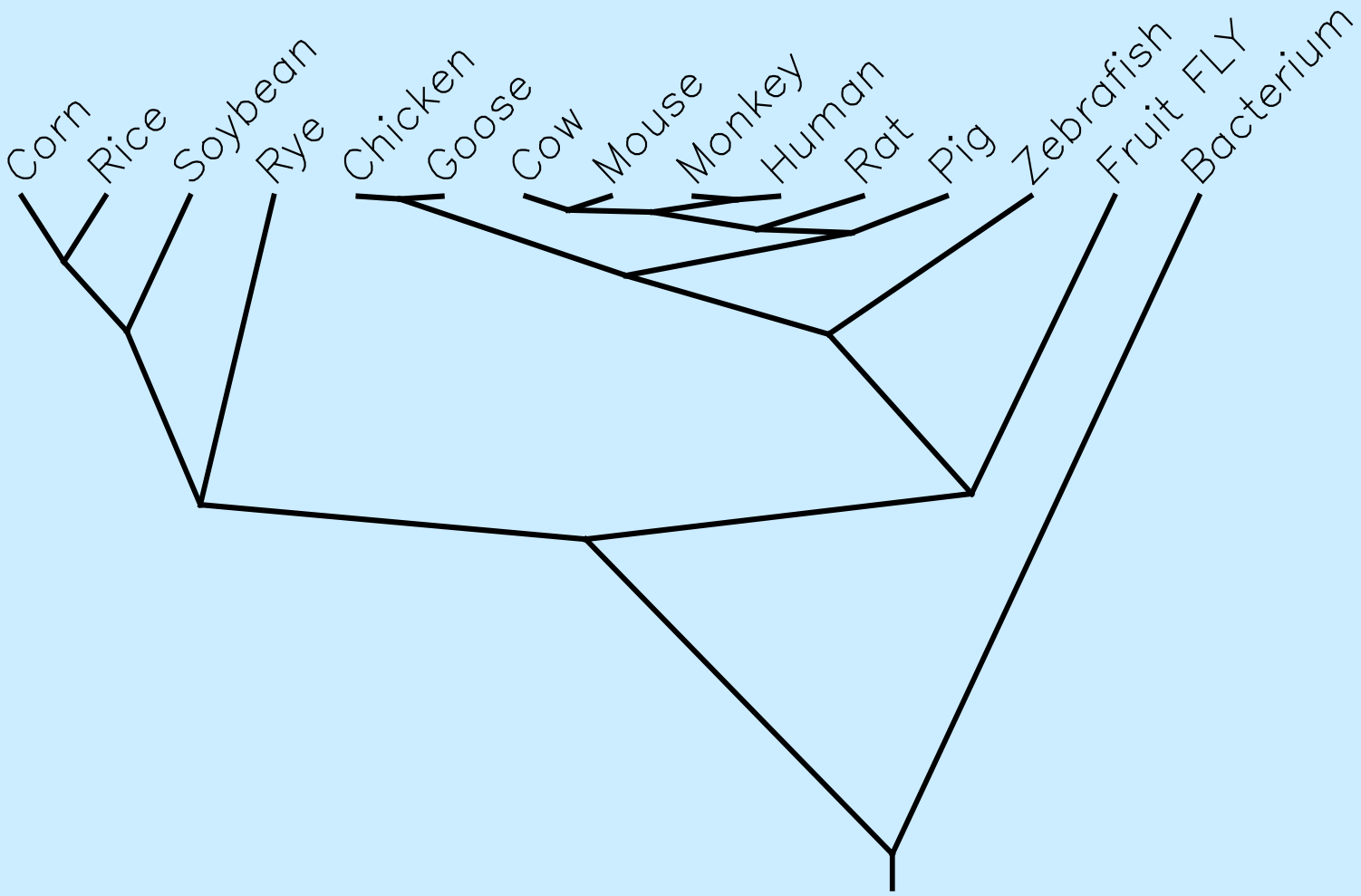
Fitch-Margoliash

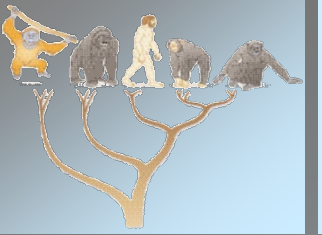




Examples

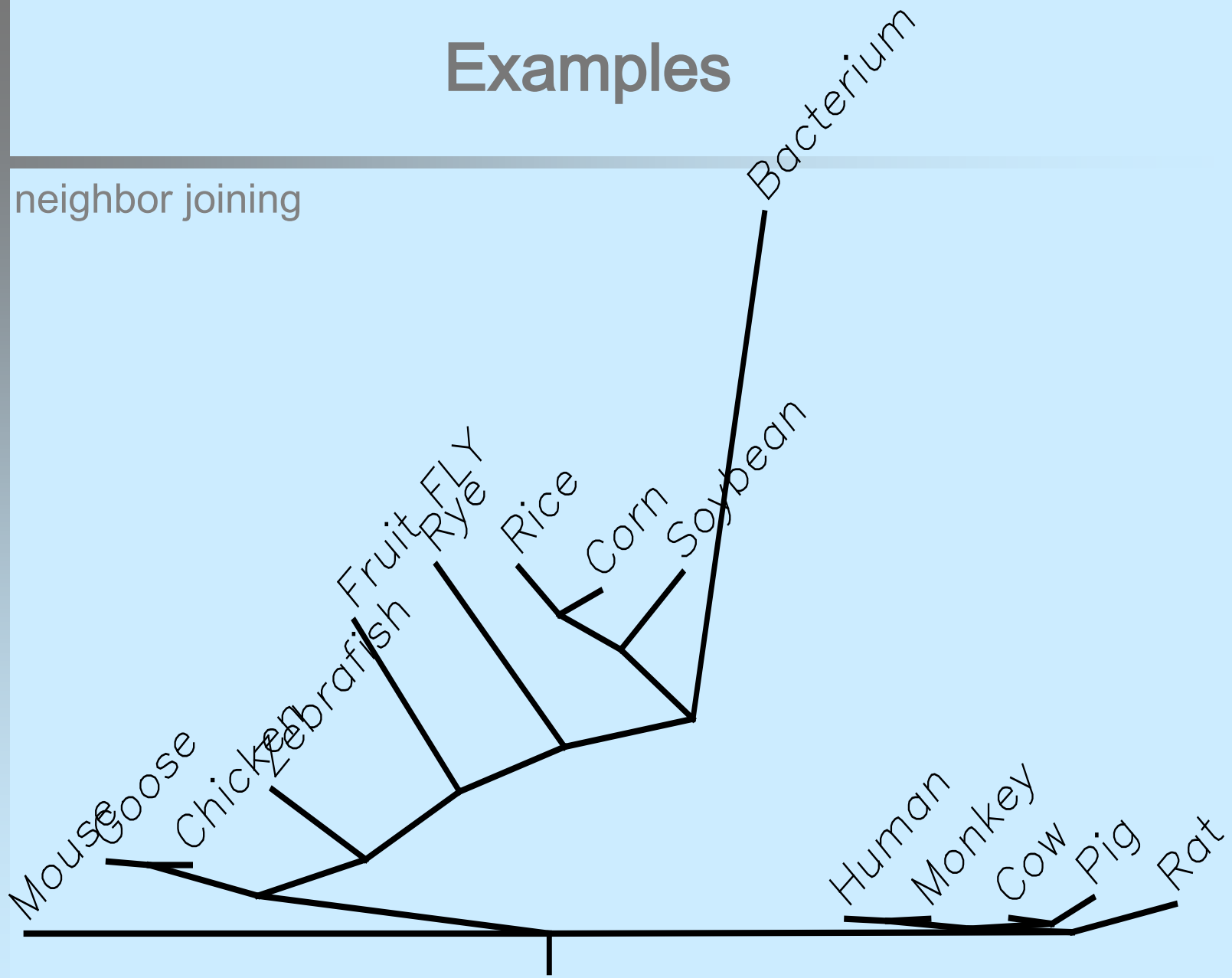
Fitch-Margoliash with assumption of a molecular clock (“kitsch”)

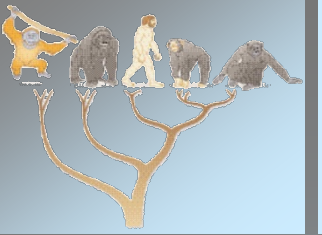




Examples

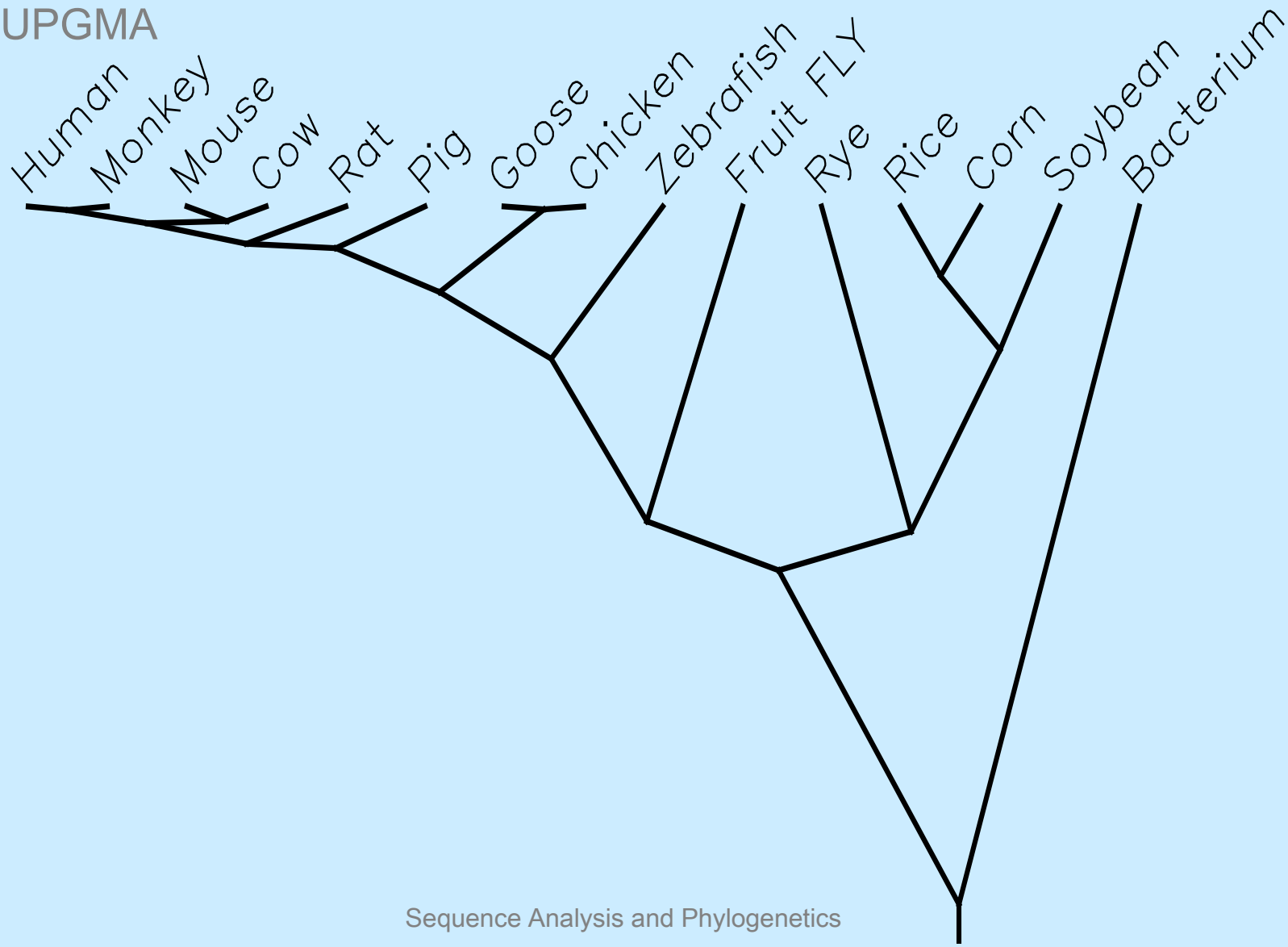
neighbor joining

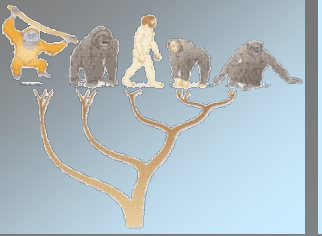




Examples

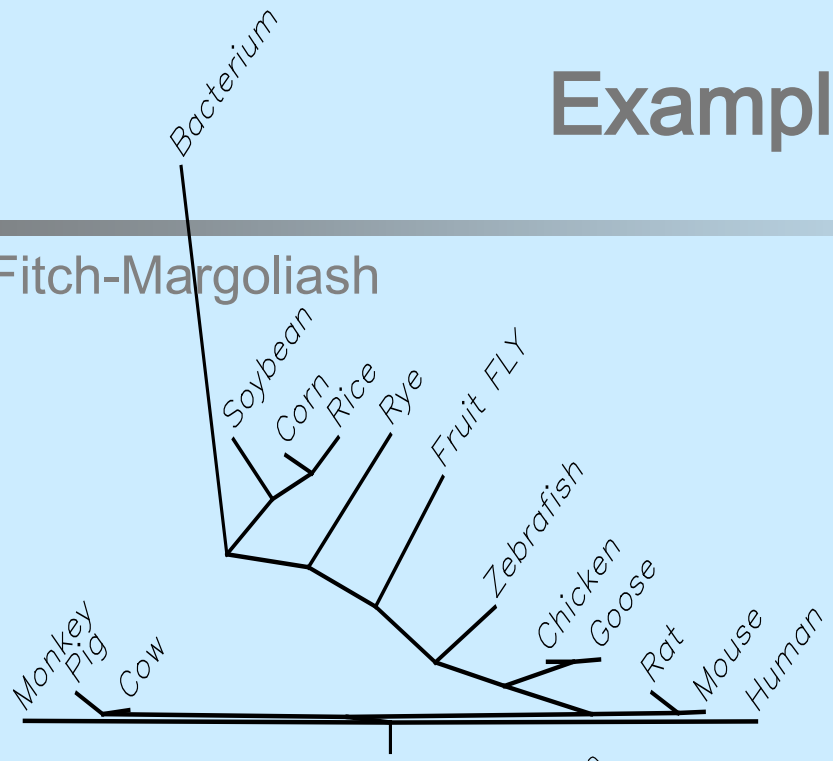
UPGMA



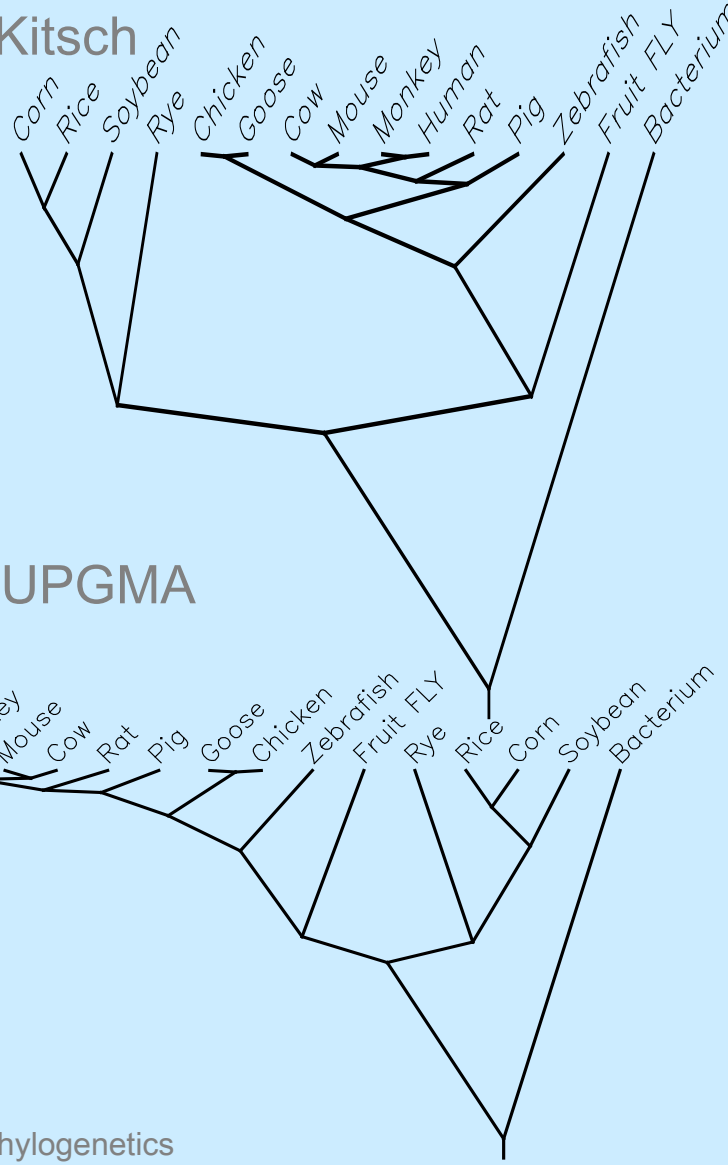


Examples

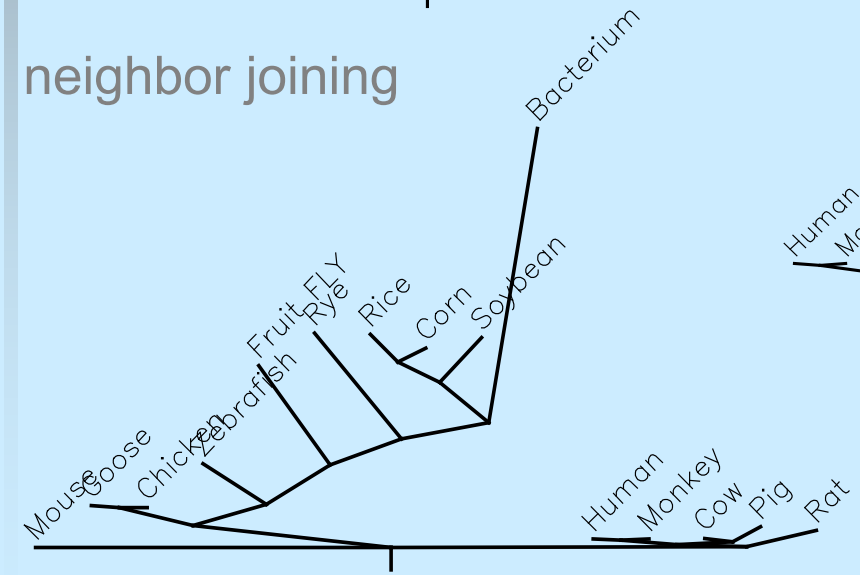
Fitch-Margoliash



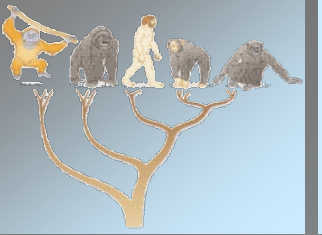
Kitsch



neighbor joining

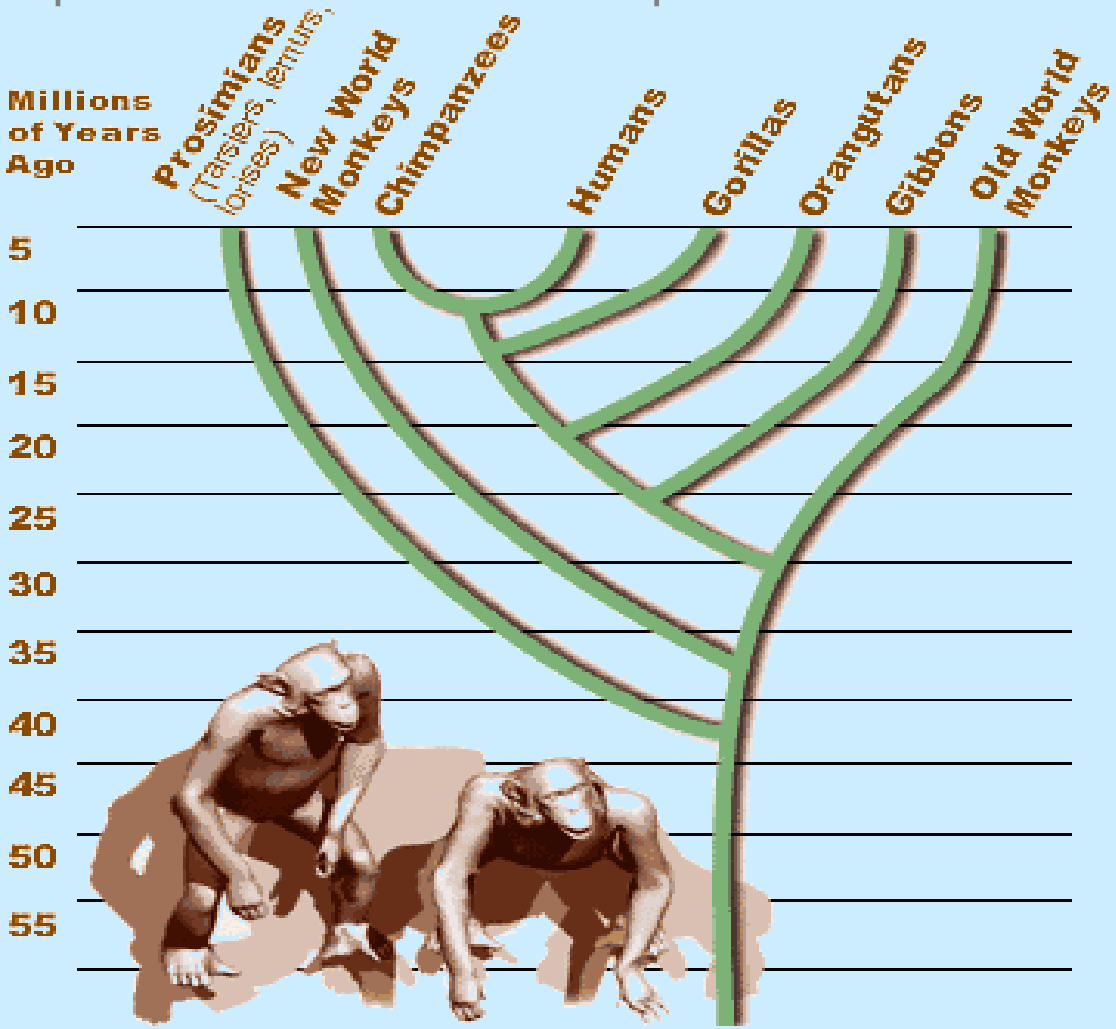


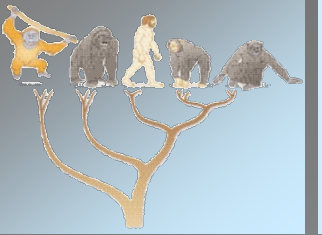
UPGMA



Examples

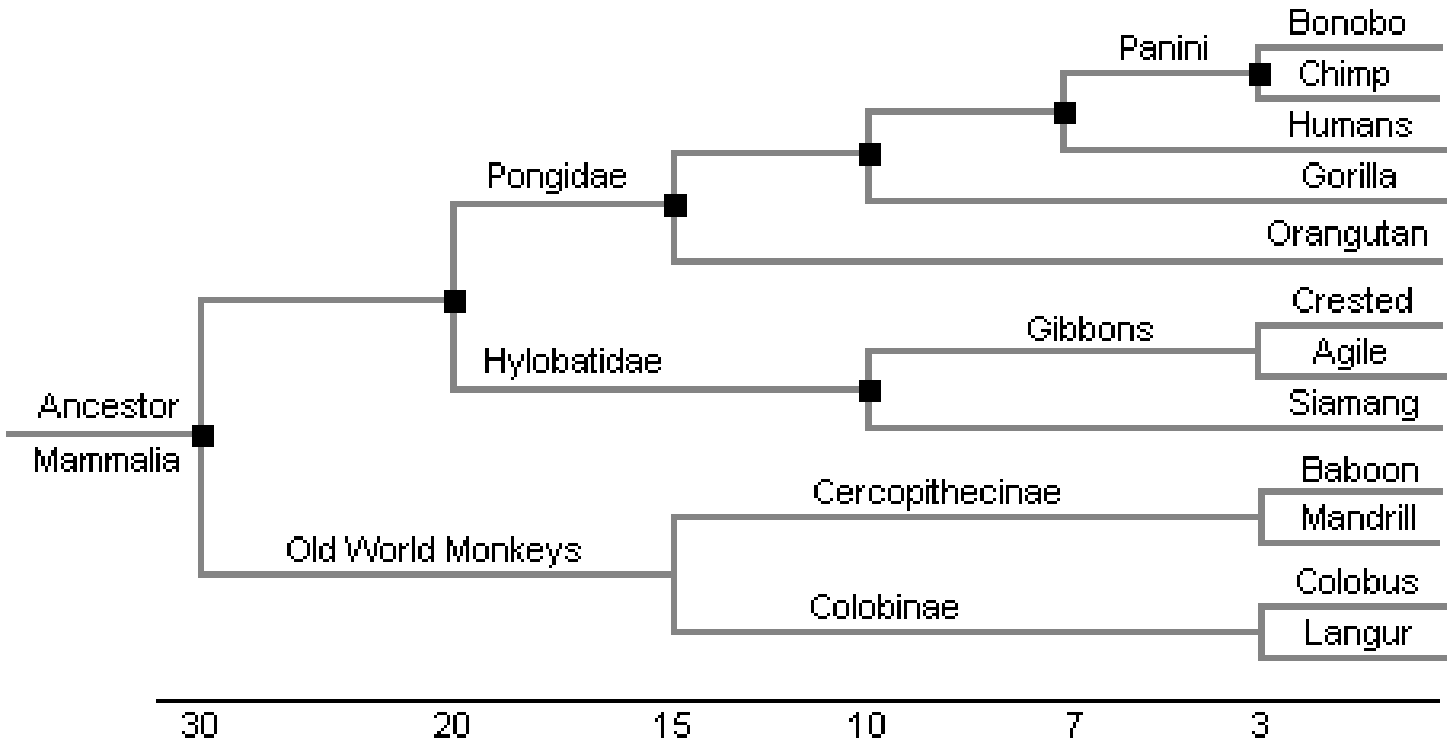
relationship between humans and apes

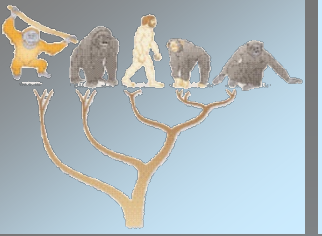




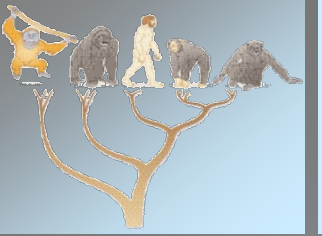
Examples

relationship between humans and apes





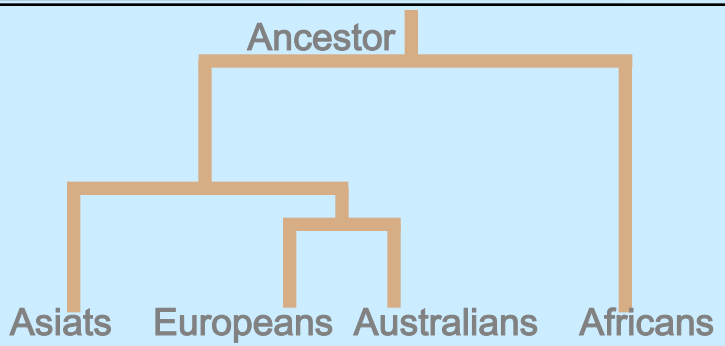
Examples



Three Anwers

➥From where came the first human?

Africa!



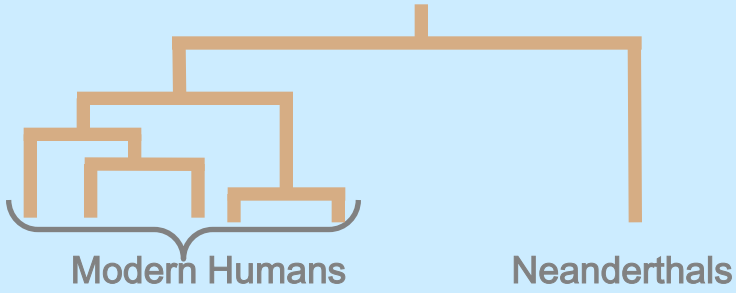
➥Is Anna Anderson the tsar‘s daughter Anastasia?

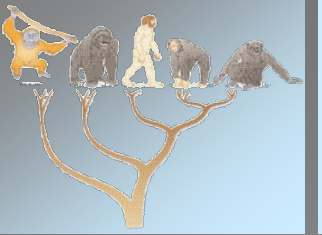
No!

	91 106	324 337
Anna Anderson	CCACCATGAATATTGC	TAGTCAAATCCCTT
Carl Maucher (Grand nephew F. Schanzkowska)	CCACCATGAATATTGC	TAGTCAAATCCCTT
Prince Philip (Grand nephew zar)	TCACCATGAATATTGT	CAGTCAAATCCCTC

➥Are the neanderthals the ancestors of the humans?

No! Separate Species





END