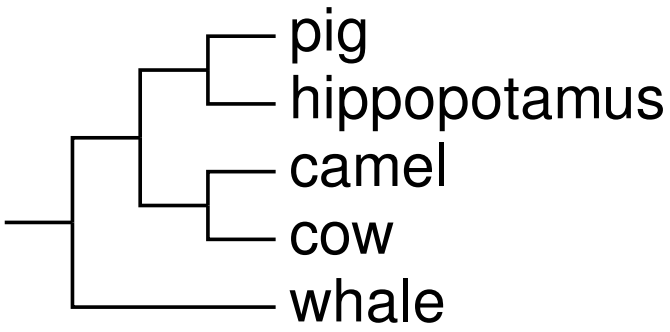
Announcements

- Homework 1 is due Tuesday November 9 22:00
  discussion regarding questions in MS Teams
- Work on the journal club
  (read the paper, plan the meeting no later than Nov. 23)

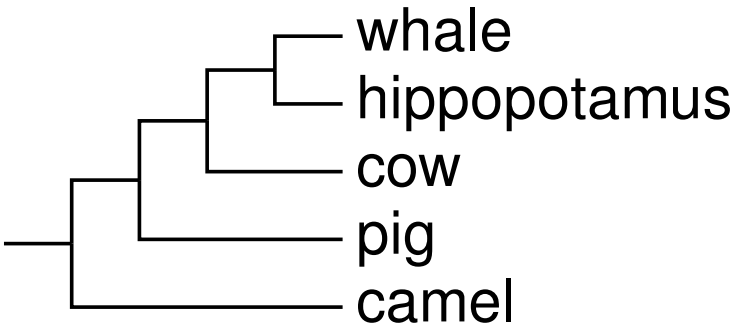
Evolution and Phylogenetic Trees

Broňa Brejová

October 28, 2021



OR



Phylogenetic tree reconstruction (fylogenetický strom)

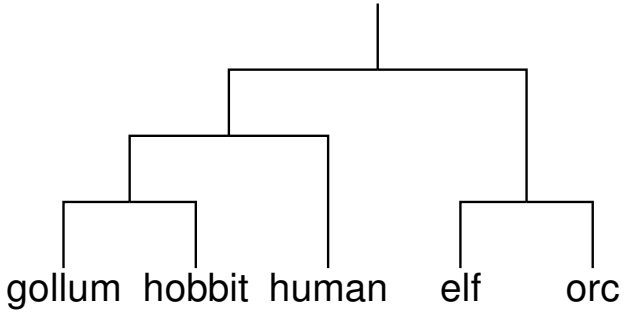
Input:

m **aligned** sequences,
each of length n

human	C	A	G	T	T	A
elf	A	A	T	A	G	A
Gollum	C	C	G	A	G	A
hobbit	C	C	G	T	T	C
orc	A	A	T	T	T	A

Output:

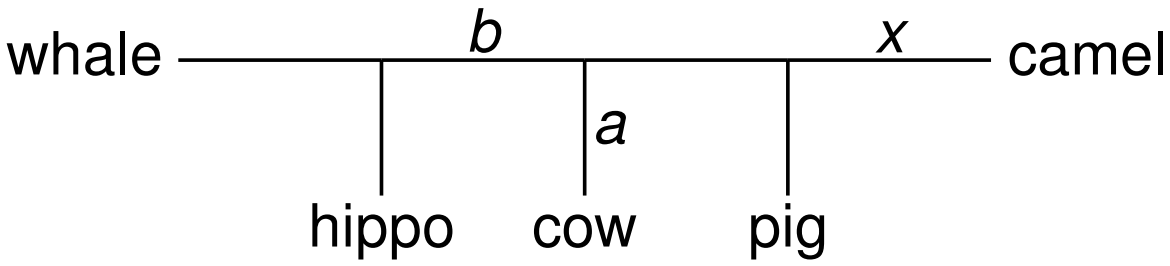
tree representing
their evolutionary history



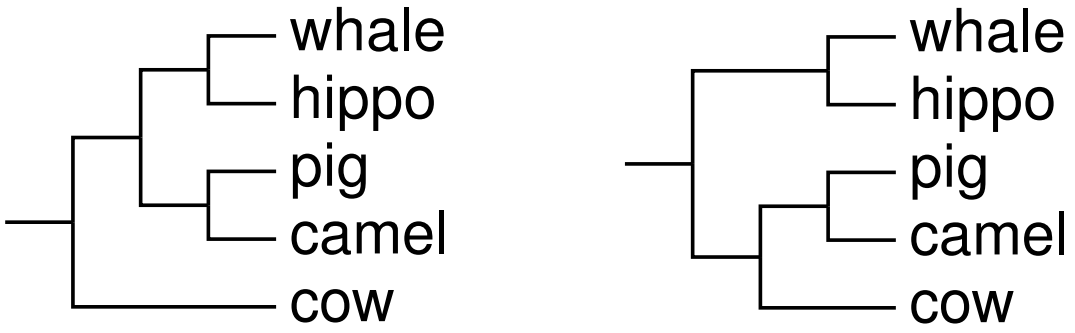
Newick format:
(((gollum,hobbit),human),(elf,orc))

Rooted and unrooted trees

Unrooted tree (nezakorenený strom)



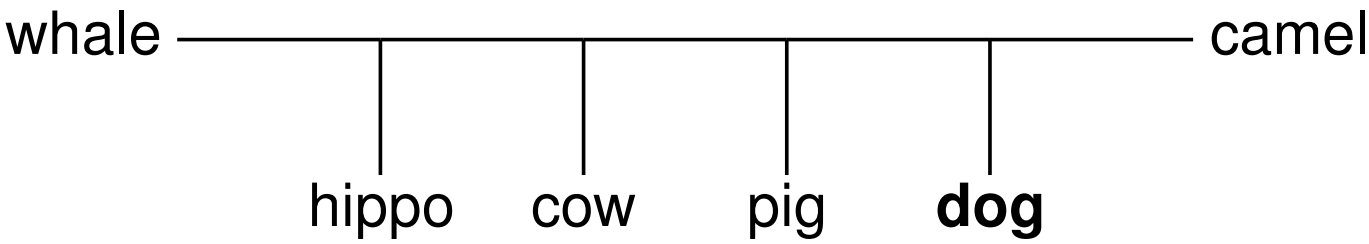
Two out of seven possible rooted versions of the tree

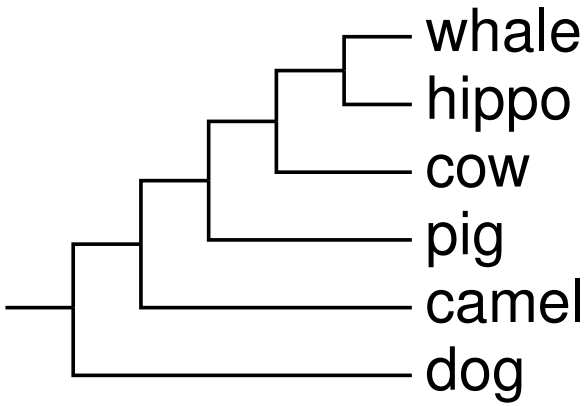


Most methods reconstruct unrooted trees

Rooting a tree using an outgroup

Add outgroup (dog) to the unrooted tree





Parsimony principle and maximum parsimony (úspornosť)

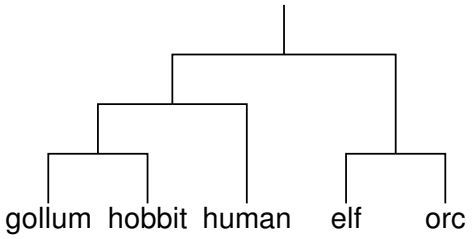
Input: (aligned) sequences of several extant species.

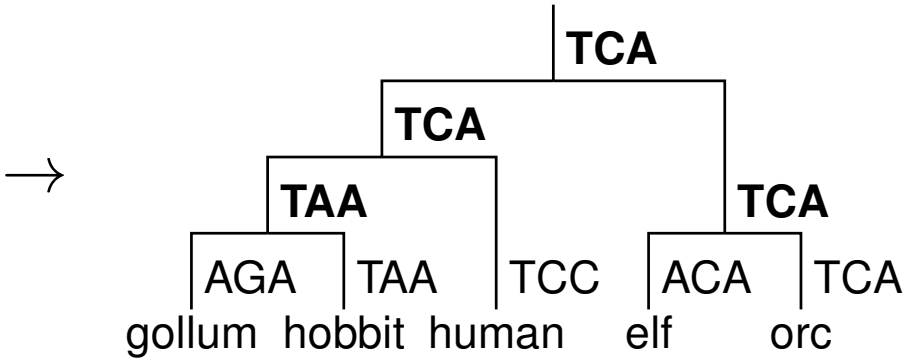
Task: Find a phylogenetic tree that explains the data by using the **minimum number of evolutionary events**.

Here: Evolutionary event = single base mutation

Subtask: For a given phylogenetic tree, find **ancestral sequences** that require the minimum number of events (score of the tree)

gollum	AGA
hobbit	TAA
human	TCC
elf	ACA
orc	TCA





5 changes

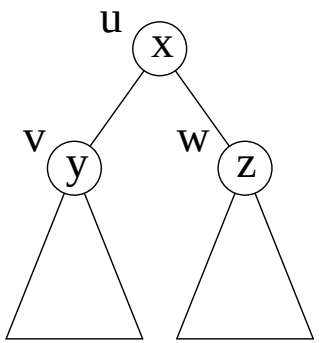
Computing cost of a given phylogenetic tree

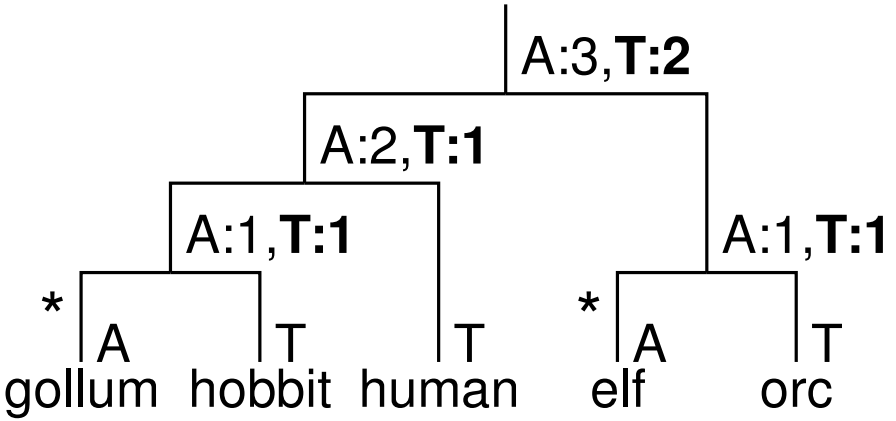
Use **dynamic programming** (separately for each alignment column).

For each internal vertex u and symbol x:

$N_{u,x}$: how many events are required in the subtree of u, assuming that the symbol in u is x?

$$N_{u,x} = \min_y\{N_{v,y} + [x \neq y]\} + \min_z\{N_{w,z} + [x \neq z]\}$$



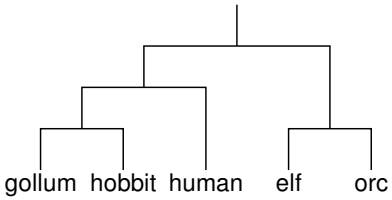


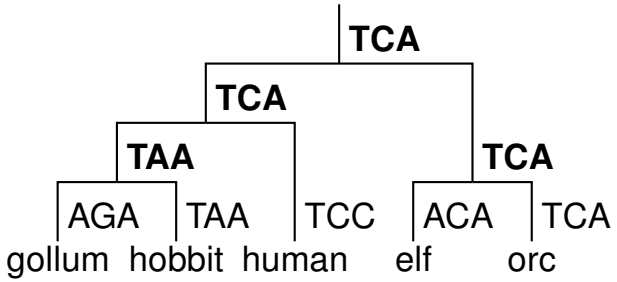
Time: $O(m)$

Repeat for each alignment column: $O(mn)$

What we have: compute the cost of a particular tree

gollum	AGA
hobbit	TAA
human	TCC
elf	ACA
orc	TCA

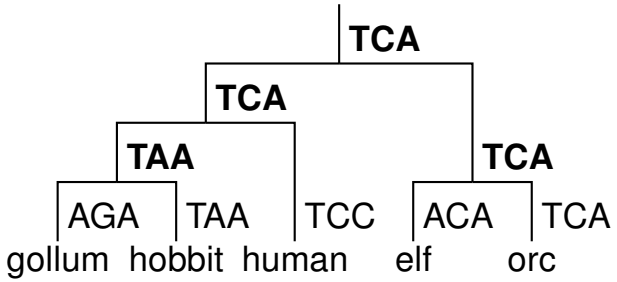




5 changes

What we want: Find the tree with the smallest cost

gollum	AGA
hobbit	TAA
human	TCC
elf	ACA
orc	TCA



Finding the most parsimonious tree

NP-hard problem

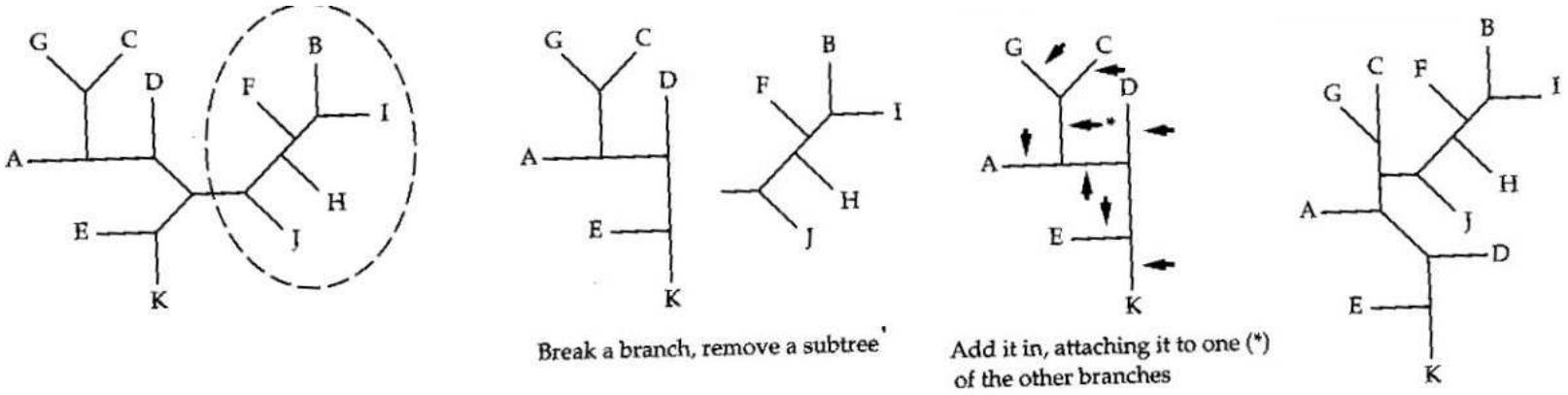
Trivial algorithm: try all possible trees.

For m species $1 \cdot 3 \cdot 5 \cdots (2m-5) = (2m-5)!!$

E.g. for 10 species cca 2 mil., for 20 species $2 \cdot 10^{20}$

Heuristic search:

- Start with some "sensible" tree
- Explore similar trees by using e.g. "subtree pruning and regraft":



Neighbour joining (metóda spájania susedov)

- We throw away "details" of which mutations happened
- Summarize by a **distance matrix** D_{ij}

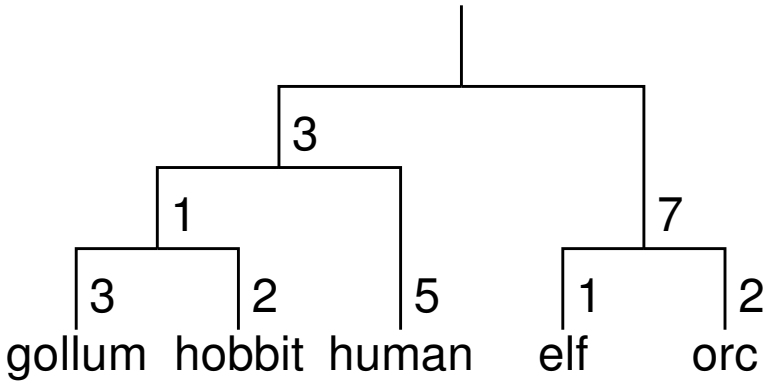
Example:

human	C	A	G	T	T	A
elf	A	A	T	A	G	A
gollum	C	C	G	A	G	A
hobbit	C	C	G	T	T	C
orc	A	A	T	T	T	A

	hu	e	h	ho	o
human	0	4	3	2	2
elf	4	0	3	6	2
gollum	3	3	0	3	5
hobbit	2	6	3	0	4
orc	2	2	5	4	0

Idea of neighbour joining

Assume that the distances $D_{i,j}$ correspond to the real distances in the tree (they are **additive**)



	gollum	hobbit	human	elf	orc
gollum	0	5	9	15	16
hobbit	5	0	8	14	15
human	9	8	0	16	17
elf	15	14	16	0	3
orc	16	15	17	3	0

$D_{\text{hobbit,human}} = 2 + 1 + 5 = 8$

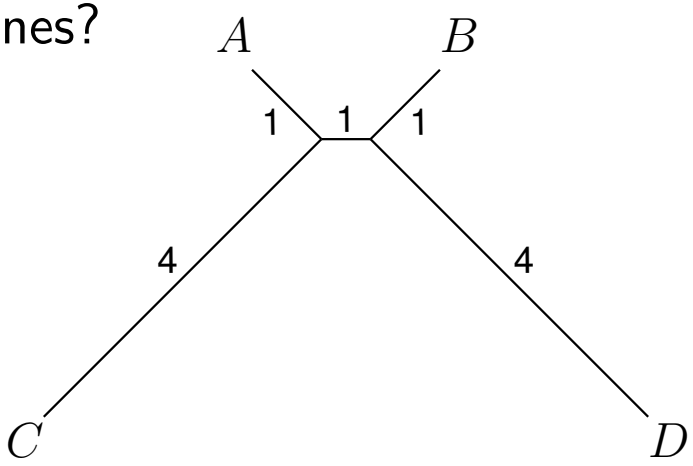
Idea of neighbour joining

- Assume that the distances $D_{i,j}$ correspond to the real distances in the tree (they are **additive**)
- Find two leaves i and j, for which we can **say with certainty**, that they have the same parent in the tree
- Join i and j and replace them with a parent node k with new distances to each other node ℓ:

$$D_{k,\ell} = \frac{D_{i,\ell} + D_{j,\ell} - D_{i,j}}{2}$$

How to find out which two leaves should be joined?

Why not two closest ones?



	A	B	C	D
A	-	3	5	6
B	3	-	6	5
C	5	6	-	9
D	6	5	9	-

Choose leaves i, j **minimizing**:

$$L_{i,j} = (m-2)D_{i,j} - \underbrace{\sum_{k \neq i} D_{i,k}}_{r_i} - \underbrace{\sum_{k \neq j} D_{j,k}}_{r_j}$$

m: the number of leaves

Connect leaves i, j, which minimize the following quantity:

$$L_{i,j} = (m-2)D_{i,j} - \underbrace{\sum_{k \neq i} D_{i,k}}_{r_i} - \underbrace{\sum_{k \neq j} D_{j,k}}_{r_j}$$

D

	g	ho	hu	e	o	r_i
g	0	5	9	15	16	45
ho	5	0	8	14	15	42
hu	9	8	0	16	17	50
e	15	14	16	0	3	48
o	16	15	17	3	0	51

L

	g	ho	hu	e	o
g	.	-72	-68	-58	-48
ho	-72	.	-68	-48	-48
hu	-68	-68	.	-50	-50
e	-58	-48	-50	.	**-90**
o	-48	-48	-50	**-90**	.

new D

	g	ho	hu	e+o
g	0	5	9	14
ho	5	0	8	13
hu	9	8	0	15
e+o	14	13	15	0

Running time of neighbor joining: $O(m^3)$ (m: number of leaves)

In 2009 a $O(m^2)$ version was developed (Elias and Lagergren)

Neighbour joining: summary

- If the distance matrix is additive and corresponds to the real evolutionary distances then neighbour joining finds the correct tree
- Longer sequences $\Rightarrow$ better distance estimates $\Rightarrow$ correct trees
- How to compute "real" evolutionary distances?
  Counting differences is not enough

human	C	A	G	T	T	A
elf	A	A	T	A	G	A
gollum	C	C	G	A	G	A
hobbit	C	C	G	T	T	C
orc	A	A	T	T	T	A

	hu	e	g	ho	o
human	0	4	3	2	2
elf	4	0	3	6	2
gollum	3	3	0	3	5
hobbit	2	6	3	0	4
orc	2	2	5	4	0

Problems with estimating distances

- One base may mutate multiple times during evolution (possibly even back to original base)
- When counting differences we see at most one change at each position $\Rightarrow$ we underestimate the real distance
- We want a correction to estimate the real number of mutations that have occurred

Jukes-Cantor substitution model

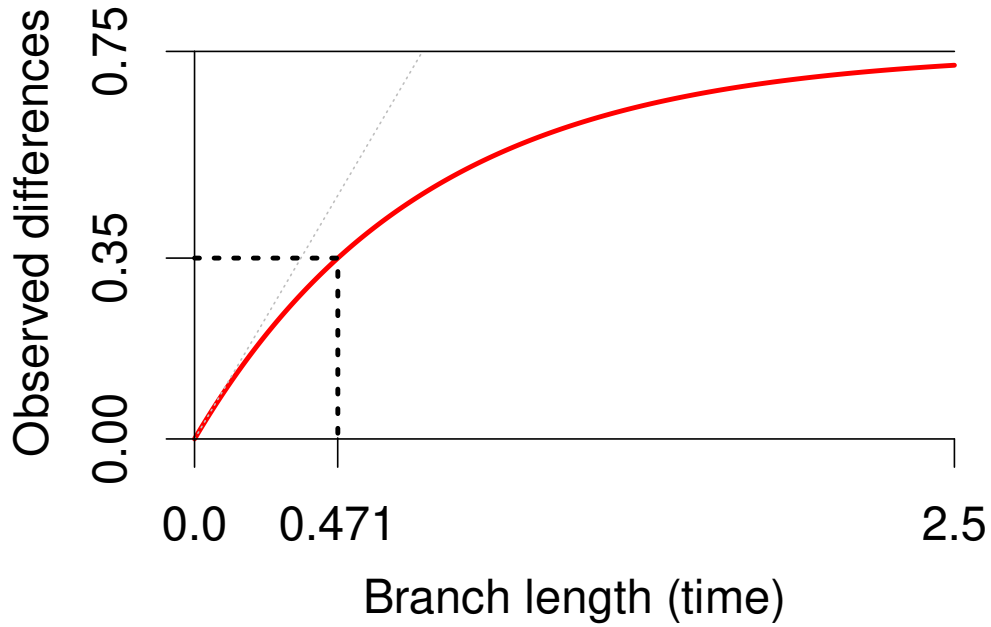
Probability that base A changes to C in time t:

$$\Pr(X_{t_0+t} = C \mid X_{t_0} = A)\frac{1}{4}(1 - e^{-\frac{4}{3}\alpha t})$$

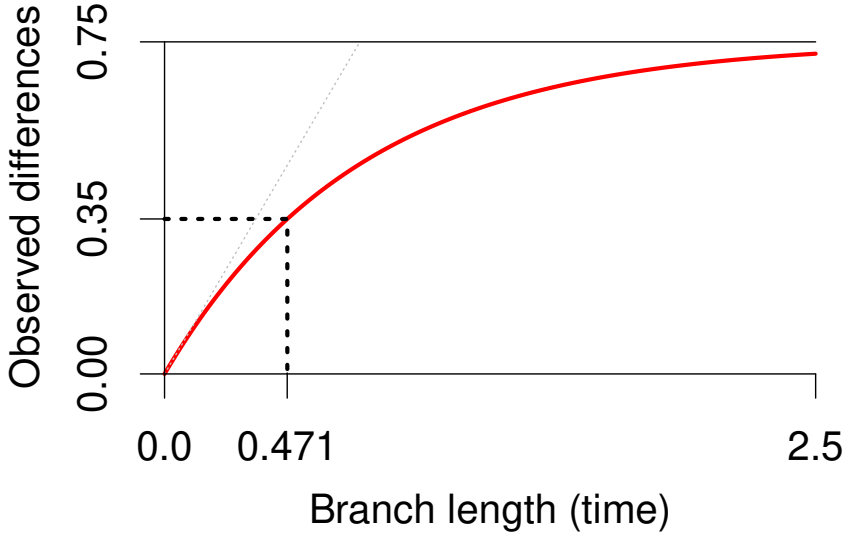
α: mutation rate (the number of substitutions per unit of time)

Expected number of mutations per base in time t:

$$D(t) = \Pr(X_{t_0+t} \neq X_{t_0}) = \frac{3}{4}(1 - e^{-\frac{4}{3}\alpha t})$$



Back to distances in neighbor joining



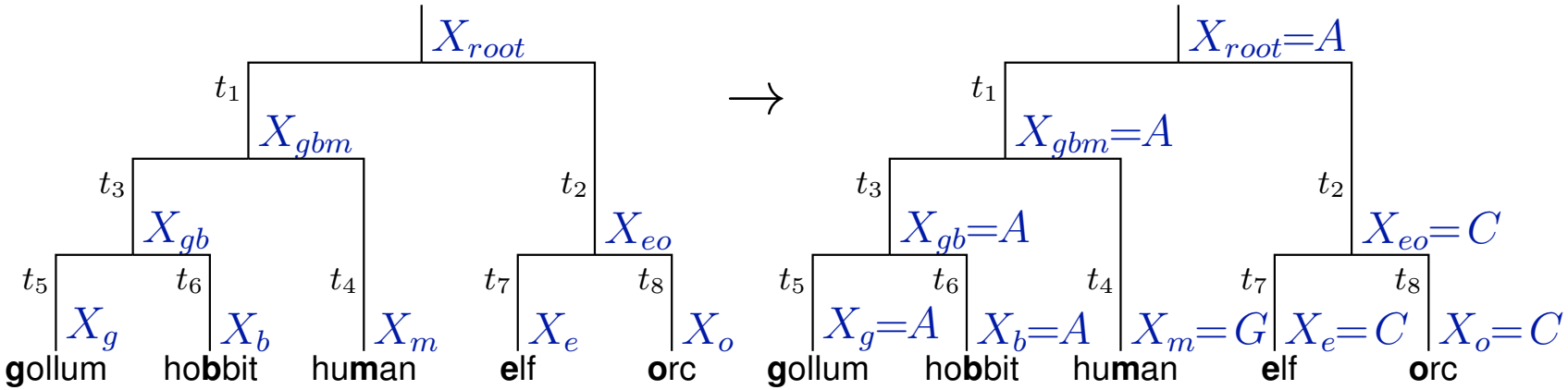
- Using this model, we can correct observed distances

$$D = \frac{3}{4}(1 - e^{-\frac{4}{3}\alpha t}) \qquad \Rightarrow \qquad \alpha t = -\frac{3}{4}\ln(1 - \frac{4}{3}D)$$

- Next week: more complex models of evolution

Maximum likelihood trees (najvierohodnejšie stromy)

A phylogenetic tree with branch lengths can be viewed
as a **simple generative model**

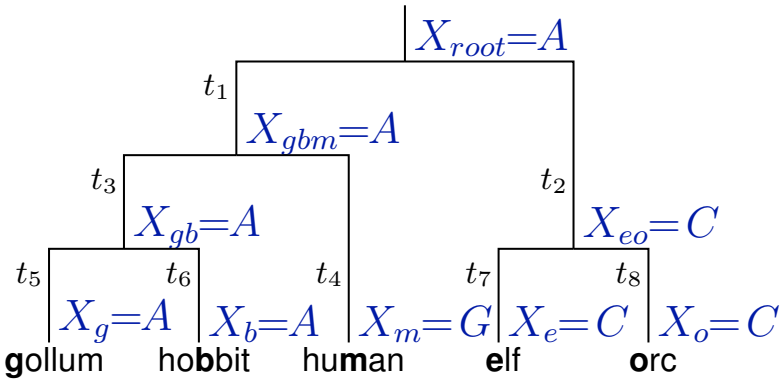


Probability that it generates particular bases in nodes:

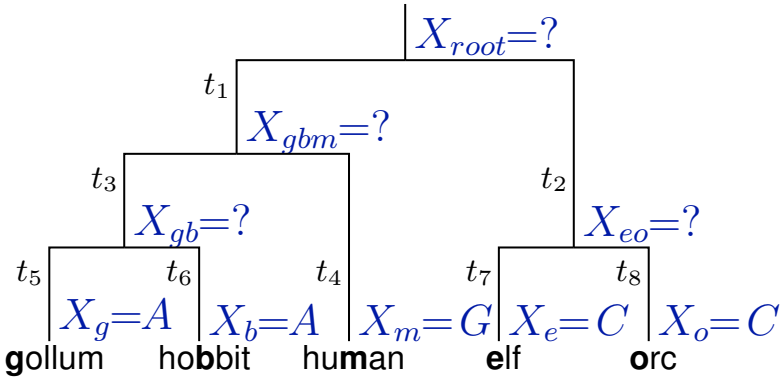
$$\Pr(X_g = A, X_b = A, X_m = G, X_e = C, X_o = C, X_{gb} = A,$$
$$X_{gbm} = A, X_{eo} = C, X_{root} = A)$$
$$= \Pr(X_{root} = A) \cdot \Pr(A \mid A, t_1) \cdot \Pr(C \mid A, t_2) \cdot \Pr(A \mid A, t_3) \cdot$$
$$\Pr(G \mid A, t_4) \cdot \Pr(A \mid A, t_5) \cdot \Pr(A \mid A, t_6) \cdot \Pr(C \mid C, t_7) \cdot \Pr(C \mid C, t_8)$$

$\Pr(C|A, t_2)$ is a abbreviation of $\Pr(X_{eo} = C|X_{root} = A)$, J.-C. model

We can compute (product):



We want to compute **tree likelihood**:



Likelihood of a tree (vierohodnosť stromu):

$\Pr(X_g = A, X_b = A, X_m = G, X_e = C, X_o = C)$

Add up probabilities of all letter combinations in ancestors X_{gb}, X_{gbm}, X_{eo}, X_{root}

Compute using **Felsenstein algorithm**
(simple dynamic programming similar to the parsimony)

For a given alignment, tree and branch lengths
we can compute likelihood in $O(nm)$ time

How to find the tree with the highest likelihood?

- Again NP-hard problem ;
  complicated because we also need **branch lengths**
- Typical heuristic algorithm:
  - Start with a "reasonable" tree
  - Compute its likelihood
    - Start with "reasonable" branch lengths
    - Compute likelihood using these branch lengths
    - Iteratively improve branch lengths to improve the likelihood (e.g. gradient descent)
  - Explore "similar" trees to improve likelihood
    (as with parsimony).

Consistency of algorithms for phylogeny

- "Well-behaved" algorithms: if the length of the sequences n increases, the answer should get closer to the correct answer.
- The algorithm for phylogeny is **consistent**, if the probability of obtaining the correct tree converges to 1 with $n \to \infty$.

Algorithm comparison

	Complexity	**Consistency**	**Data utilization**
Parsimony	NP-hard	NO	complete sequences
Neighbor Joining	$O(m^3)$	YES	distances only
Likelihood	NP-hard	YES	complete

Sources of data for phylogenetic trees

Some special sequences are often used
(e.g. ribosomal RNA genes, mitochondrial genome)

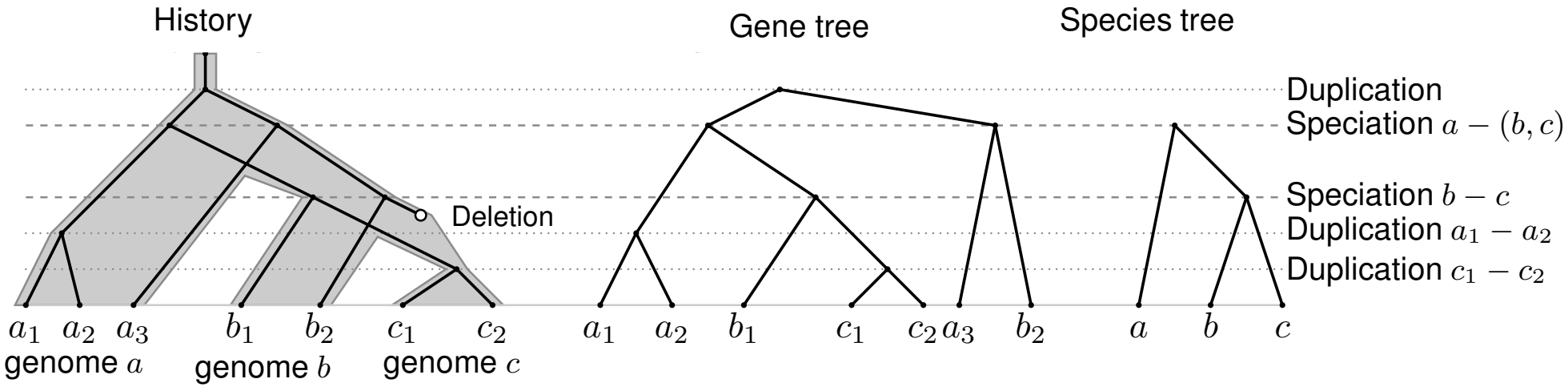
What about using DNA sequences of other genes?

- Choose a suitable gene
- Find its homologs in other species
- Use these to construct the tree
  (DNA sequences or proteins)

Problem: genes can be duplicated and lost in evolution

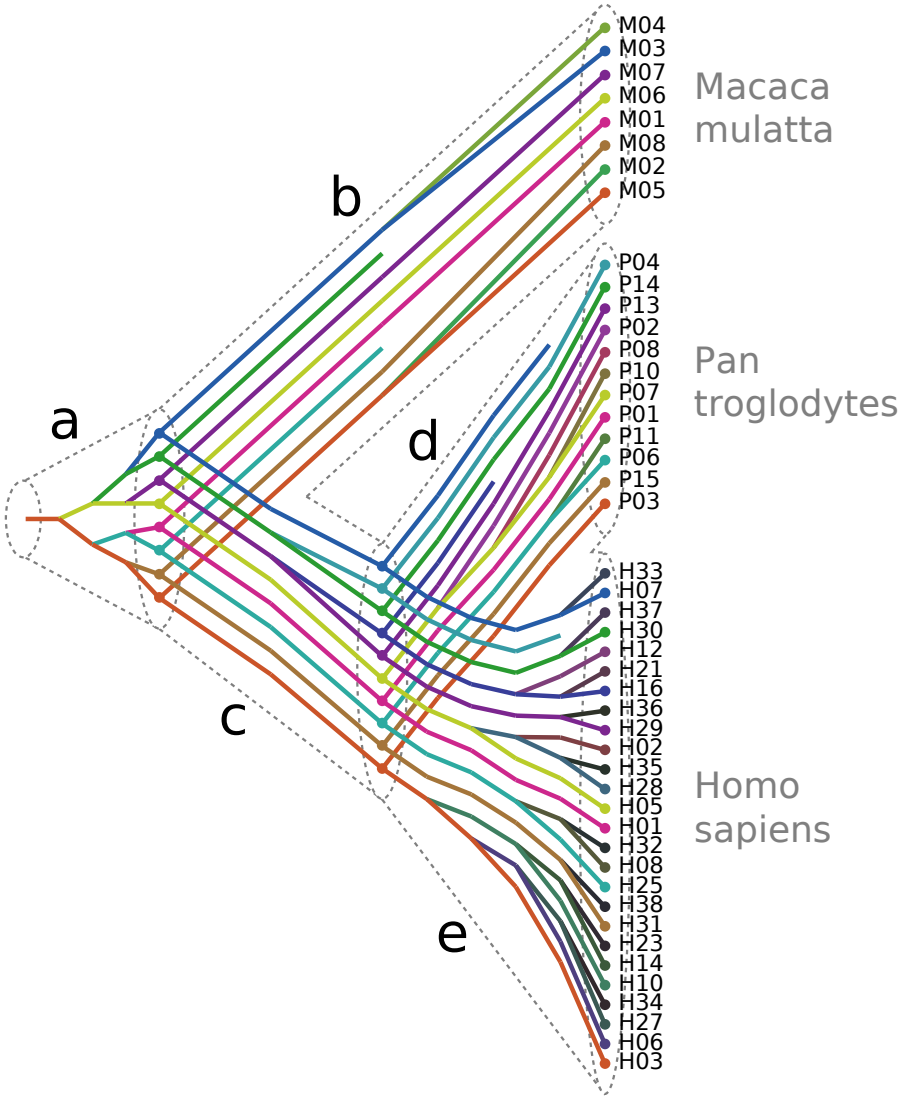
History of a duplicated gene

Example: species a, b, c, genes $a_1, a_2, a_3, b_1, b_2, c_1, c_2$



- **Homologs:** similar sequences evolved from a common ancestor
- **Orthologs:** closest common ancestor is a speciation
  (e.g. pairs of genes $a_1 - b_1$, $a_2 - b_1$)
- **Paralogs:** closest common ancestor is a duplication
  (e.g. pairs of genes $a_1 - a_2$, $a_1 - b_2$)

A more complex example of gene duplication:



Summary

Substitution models allow us to:

- estimate real evolutionary distance (the number of substitutions) from the observed difference count between two sequences
- compute the probability that we observe a particular nucleotide change over time t

Three methods for phylogeny inference:

- Parsimony
- Neighbour joining
- Maximum likelihood

Gene trees and species trees, complications in phylogeny reconstruction