

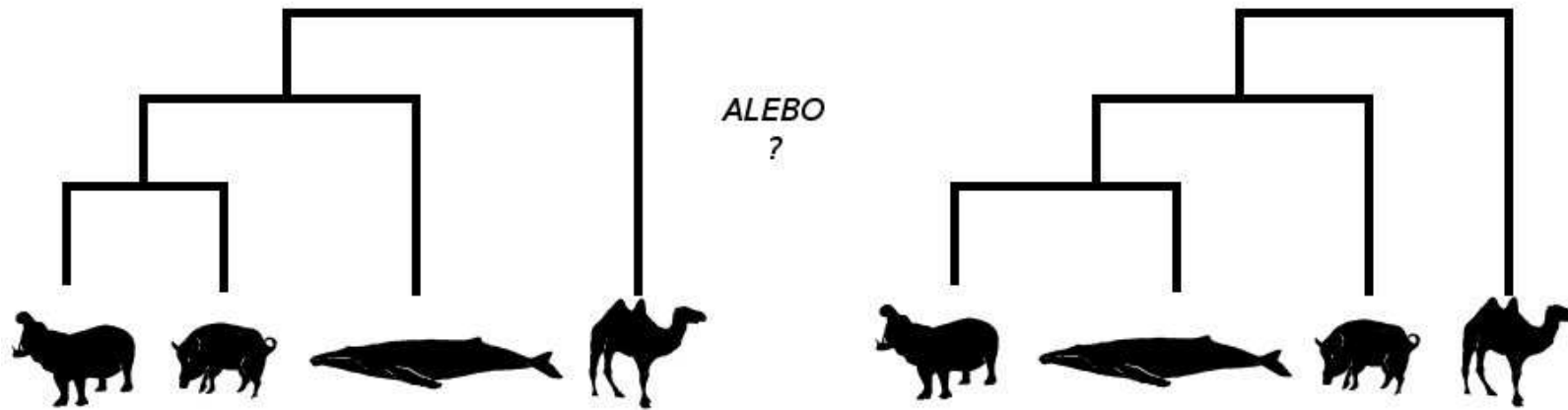
Organizačné poznámky

- Domáca úloha 1 bola do dnes
- Domáca úloha 2 zadania o dva týždne
- Pracujte na journal clube (prečítajte si článok, naplánujte si stretnutie)

Evoluční modely a stromy

Broňa Brejová

8.11.2018



Rekonštrukcia fylogenetických stromov

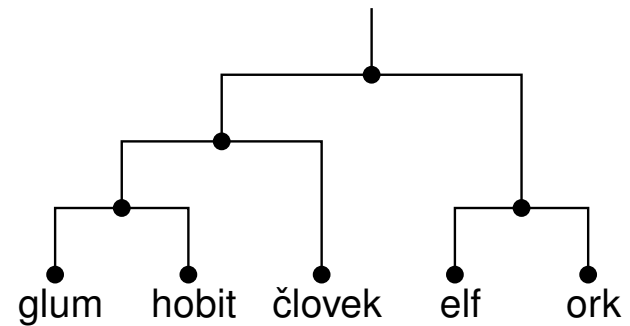
Vstup:

m zarovnaných sekvencií,
každá dĺžky n

človek	C	A	G	T	T	A
elf	A	A	T	A	G	A
Glum	C	C	G	A	G	A
hobit	C	C	G	T	T	C
ork	A	A	T	T	T	A

Výstup:

strom predstavujúci
ich evolučnú históriu

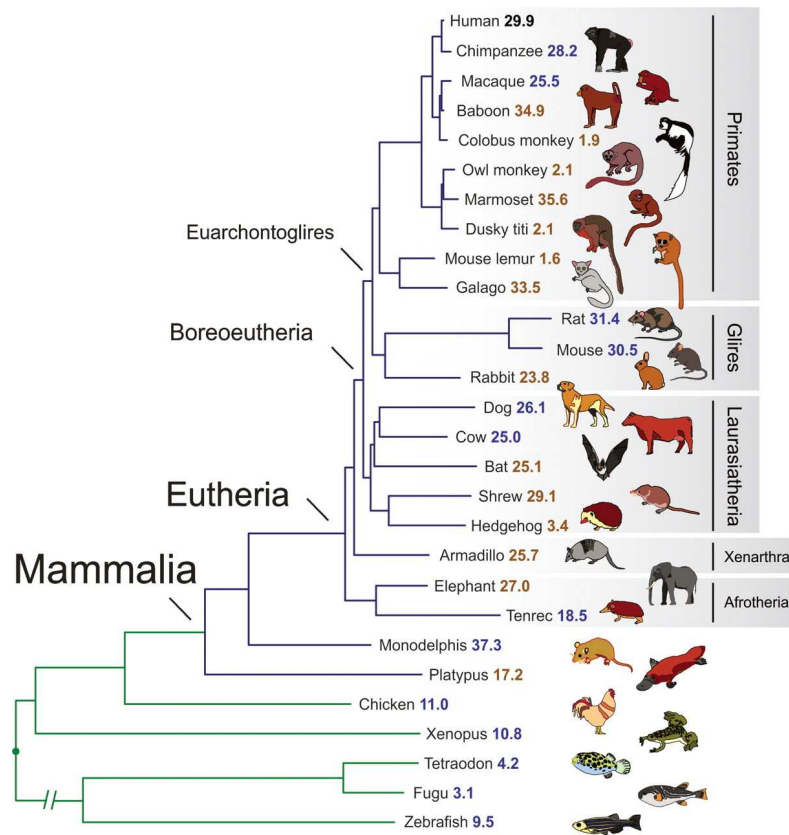


Newick format:

```
((glum,hobit),človek),(elf,ork))
```

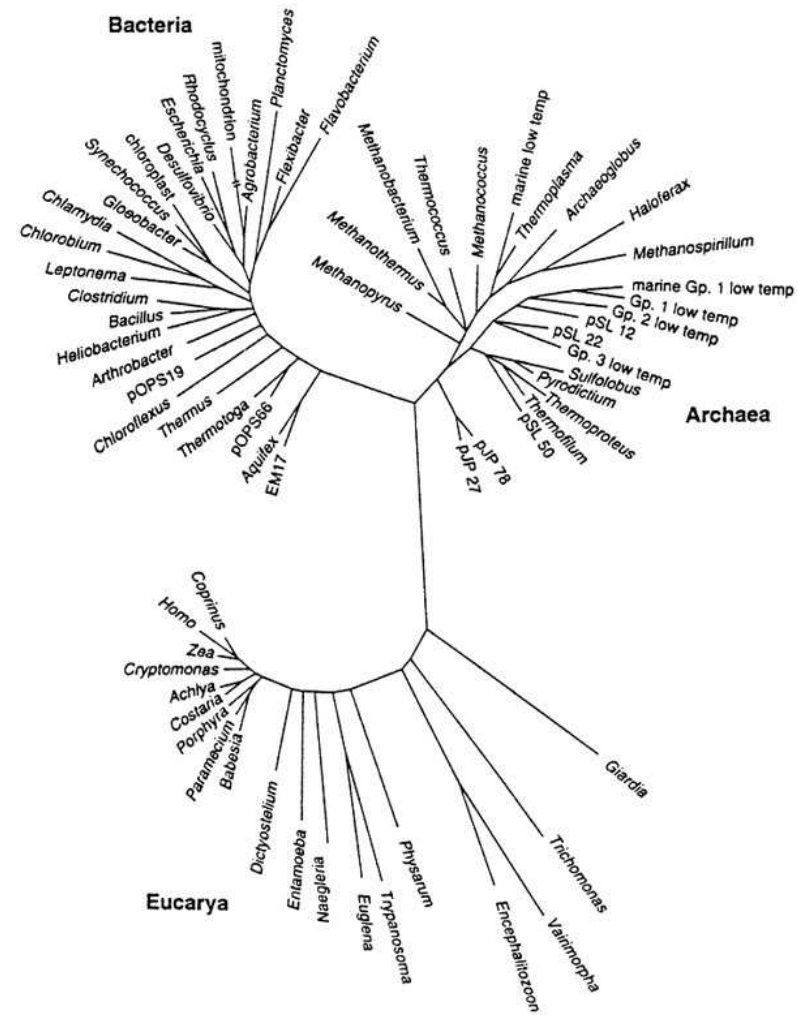
Zakorenené a nezakorenené stromy

[Margulies et al., 2007]



zakorenený pomocou
“outgroup”

[Pace, 1997]



Maximum parsimony (úsporné stromy)

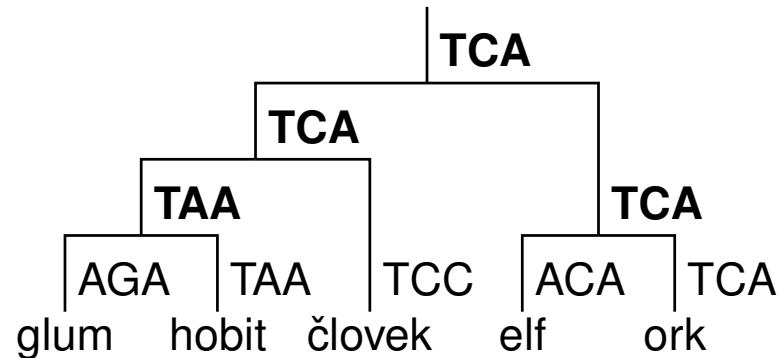
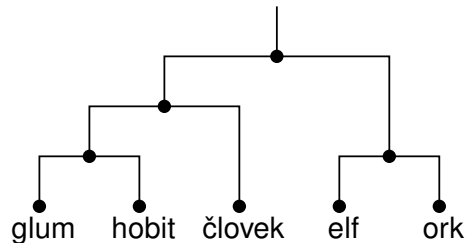
Úloha: Dané sú zarovnané sekvencie súčasných organizmov.

Chceme nájsť fylogenetický strom, ktorý vyžaduje **minimálny počet evolučných zmien**.

Evolučná zmena = mutácia jednej bázy na inú bázu

Podotázka: Pre daný fylogenetický strom, doplniť **ancestrálne sekvencie** tak, aby bol potrebný najmenší počet zmien.

glum	AGA
hobit	TAA
človek	TCC
elf	ACA
ork	TCA



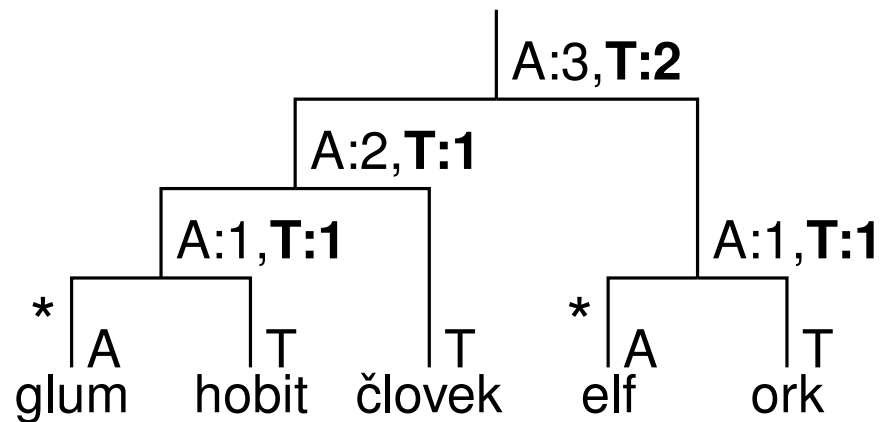
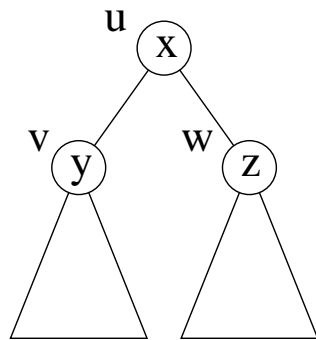
Výpočet ceny konkrétného stromu

Môžeme rátať **dynamickým programovaním** pre každý stĺpec zarovnania zvlášť.

Pre každý vnútorný vrchol u a symbol x :

$N_{u,x}$: koľko zmien treba v podstrome pod u , ak v u bude symbol x ?

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



Časová zložitosť: $O(m)$, lineárna

Hľadanie najúspornejšieho stromu

NP-ťažký problém

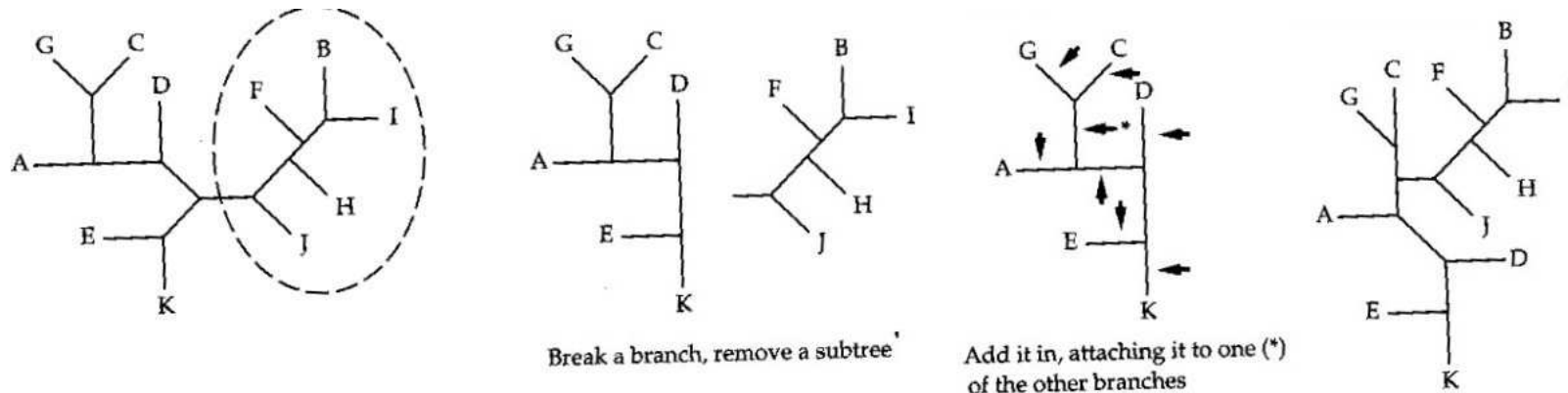
Triviálny algoritmus: vyskúšaj všetky možné stromy.

Pre m druhov $1 \cdot 3 \cdot 5 \cdots (2m - 5) = (2m - 5)!!$

Napr. pre 10 druhov cca 2 milióny, pre 20 druhov $2 \cdot 10^{20}$

Heuristické prehľadávanie:

- Začneme s “rozumným” stromom
- Pomocou stanovených operácií prehľadávame “podobné” stromy;
napr. “subtree pruning and regraft”:



Neighbor Joining (Metóda spájania susedov)

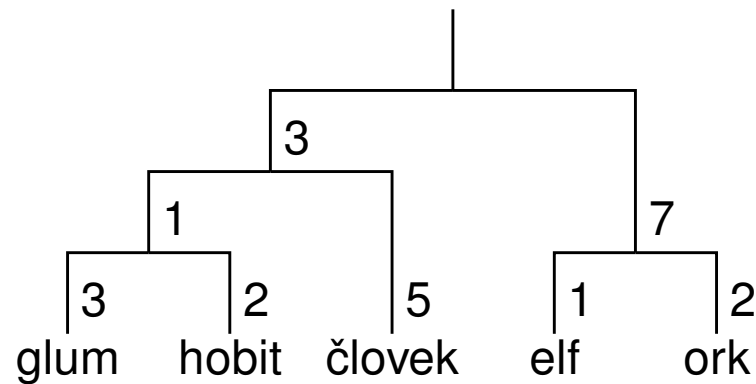
- Nevyužívame detaily rozdielov medzi sekvenciami
- Zosumarizujeme ich pomocou **matice vzdialeností** (D_{ij})

Jednoduchý príklad:

človek	C	A	G	T	T	A		Č	E	G	H	O
elf	A	A	T	A	G	A	človek	0	4	3	2	2
Glum	C	C	G	A	G	A	elf	4	0	3	6	2
hobit	C	C	G	T	T	C	Glum	3	3	0	3	5
ork	A	A	T	T	T	A	hobit	2	6	3	0	4
							ork	2	2	5	4	0

Idea spájania susedov

- Predpokladáme, že vzdialenosti $D_{i,j}$ skutočne zodpovedajú vzdialenostiam v strome (**aditivita**)



$$D_{\text{hobit}, \text{človek}} = 2 + 1 + 5 = 8$$

	glum	hobit	človek	elf	ork
glum	0	5	9	15	16
hobit	5	0	8	14	15
človek	9	8	0	16	17
elf	15	14	16	0	3
ork	16	15	17	3	0

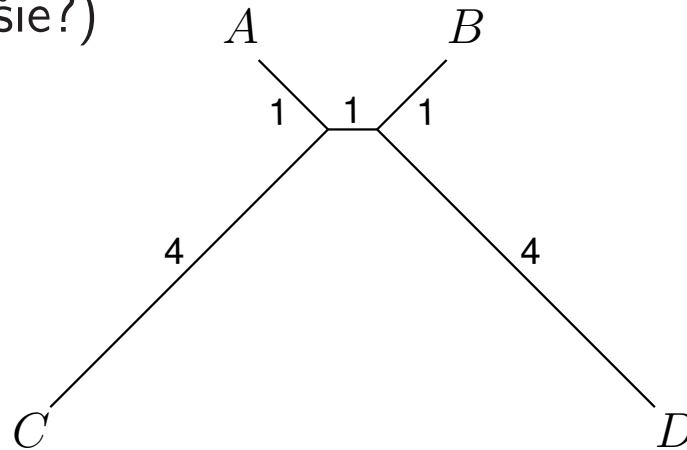
Idea spájania susedov

- Predpokladáme, že vzdialenosti $D_{i,j}$ skutočne zodpovedajú vzdialenostiam v strome (**aditivita**)
- Nájdeme dva listy i a j , o ktorých vieme **s určitosťou povedať**, že majú vo výslednom strome spoločného otca
- i a j spojíme a nahradíme ich ich otcom k s novými vzdialenosťami:

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

Ako určiť dva listy na spájanie?

(Prečo nie dva najbližšie?)



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

Vyber listy i, j , ktoré **minimalizujú** nasledujúci výraz:

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

m : počet listov

Časová zložitosť spájania susedov: $O(m^3)$

Ako určiť dva listy na spájanie?

Vyber listy i, j , ktoré **minimalizujú** nasledujúci výraz:

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

D	L	nové D																																																																																																							
<table style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <th style="width: 5%;"></th> <th style="width: 10%;">g</th> <th style="width: 10%;">h</th> <th style="width: 10%;">č</th> <th style="width: 10%;">e</th> <th style="width: 10%;">o</th> <th style="width: 15%;">r_i</th> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">g</th> <td>0</td> <td>5</td> <td>9</td> <td>15</td> <td>16</td> <td>45</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">h</th> <td>5</td> <td>0</td> <td>8</td> <td>14</td> <td>15</td> <td>42</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">č</th> <td>9</td> <td>8</td> <td>0</td> <td>16</td> <td>17</td> <td>50</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">e</th> <td>15</td> <td>14</td> <td>16</td> <td>0</td> <td>3</td> <td>48</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">o</th> <td>16</td> <td>15</td> <td>17</td> <td>3</td> <td>0</td> <td>51</td> </tr> </table>		g	h	č	e	o	r_i	g	0	5	9	15	16	45	h	5	0	8	14	15	42	č	9	8	0	16	17	50	e	15	14	16	0	3	48	o	16	15	17	3	0	51	<table style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <th style="width: 5%;"></th> <th style="width: 10%;">g</th> <th style="width: 10%;">h</th> <th style="width: 10%;">č</th> <th style="width: 10%;">e</th> <th style="width: 10%;">o</th> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">g</th> <td>.</td> <td>-72</td> <td>-68</td> <td>-58</td> <td>-48</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">h</th> <td>-72</td> <td>.</td> <td>-68</td> <td>-48</td> <td>-48</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">č</th> <td>-68</td> <td>-68</td> <td>.</td> <td>-50</td> <td>-50</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">e</th> <td>-58</td> <td>-48</td> <td>-50</td> <td>.</td> <td>-90</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">o</th> <td>-48</td> <td>-48</td> <td>-50</td> <td>-90</td> <td>.</td> </tr> </table>		g	h	č	e	o	g	.	-72	-68	-58	-48	h	-72	.	-68	-48	-48	č	-68	-68	.	-50	-50	e	-58	-48	-50	.	-90	o	-48	-48	-50	-90	.	<table style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <th style="width: 5%;"></th> <th style="width: 10%;">g</th> <th style="width: 10%;">h</th> <th style="width: 10%;">č</th> <th style="width: 10%;">eo</th> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">g</th> <td>0</td> <td>5</td> <td>9</td> <td>14</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">h</th> <td>5</td> <td>0</td> <td>8</td> <td>13</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">č</th> <td>9</td> <td>8</td> <td>0</td> <td>15</td> </tr> <tr> <th style="border-top: 1px solid black; border-bottom: 1px solid black;">eo</th> <td>14</td> <td>13</td> <td>15</td> <td>0</td> </tr> </table>		g	h	č	eo	g	0	5	9	14	h	5	0	8	13	č	9	8	0	15	eo	14	13	15	0
	g	h	č	e	o	r_i																																																																																																			
g	0	5	9	15	16	45																																																																																																			
h	5	0	8	14	15	42																																																																																																			
č	9	8	0	16	17	50																																																																																																			
e	15	14	16	0	3	48																																																																																																			
o	16	15	17	3	0	51																																																																																																			
	g	h	č	e	o																																																																																																				
g	.	-72	-68	-58	-48																																																																																																				
h	-72	.	-68	-48	-48																																																																																																				
č	-68	-68	.	-50	-50																																																																																																				
e	-58	-48	-50	.	-90																																																																																																				
o	-48	-48	-50	-90	.																																																																																																				
	g	h	č	eo																																																																																																					
g	0	5	9	14																																																																																																					
h	5	0	8	13																																																																																																					
č	9	8	0	15																																																																																																					
eo	14	13	15	0																																																																																																					

Spájanie susedov: zhrnutie

- Ak je vstupná matica aditívna a zodpovedá skutočným evolučným vzdialenostiam, spájanie susedov nám dá správny strom
- Čím dlhšie sekvencie, tým spoľahlivejší odhad vzdialenosti a tým väčšia šanca dostať správny strom
- Ako však prejdeme od sekvencií k odhadu vzdialenosti?
Len počítanie rozdielov nestačí

človek	C	A	G	T	T	A		Č	E	G	H	O
elf	A	A	T	A	G	A	človek	0	4	3	2	2
Glum	C	C	G	A	G	A	elf	4	0	3	6	2
hobit	C	C	G	T	T	C	Glum	3	3	0	3	5
ork	A	A	T	T	T	A	hobit	2	6	3	0	4
							ork	2	2	5	4	0

Problém so vzdialenosťami

- Počas evolúcie sa môže stať, že tá istá báza zmutuje **viackrát** (trebárs aj späť na pôvodnú bázu)
- Pri počítaní rozdielov ale vidíme nanajvýš jednu zmenu na každej pozícii \Rightarrow odhad vzdialenosti menší ako v skutočnosti
- Chceme korekciu na odhadovaný počet mutácií, ktoré sa naozaj stali

Jukesov-Cantorov model evolúcie

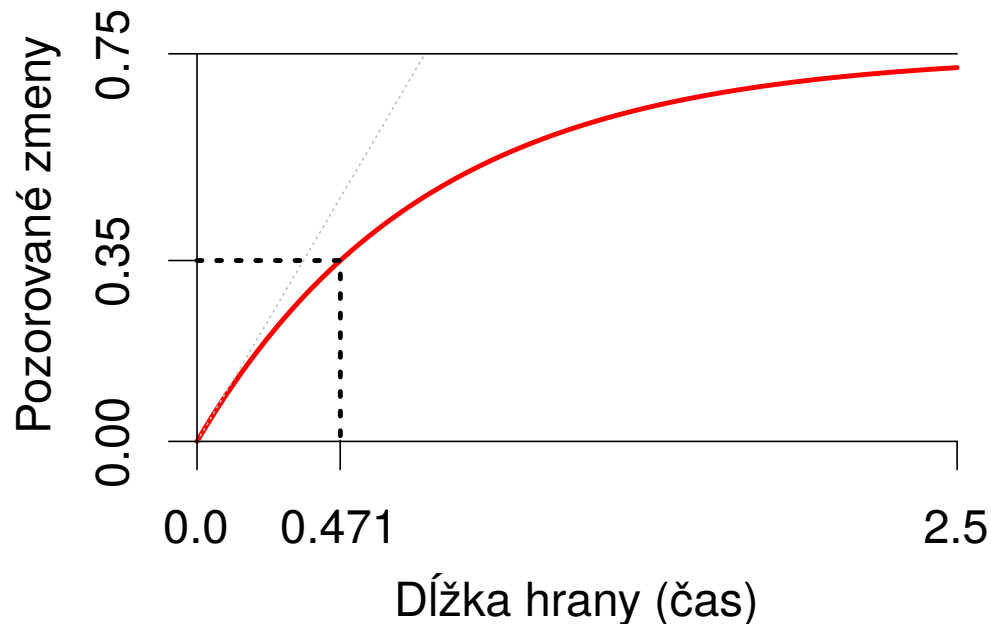
Pravdepodobnosť zmeny bázy na inú:

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

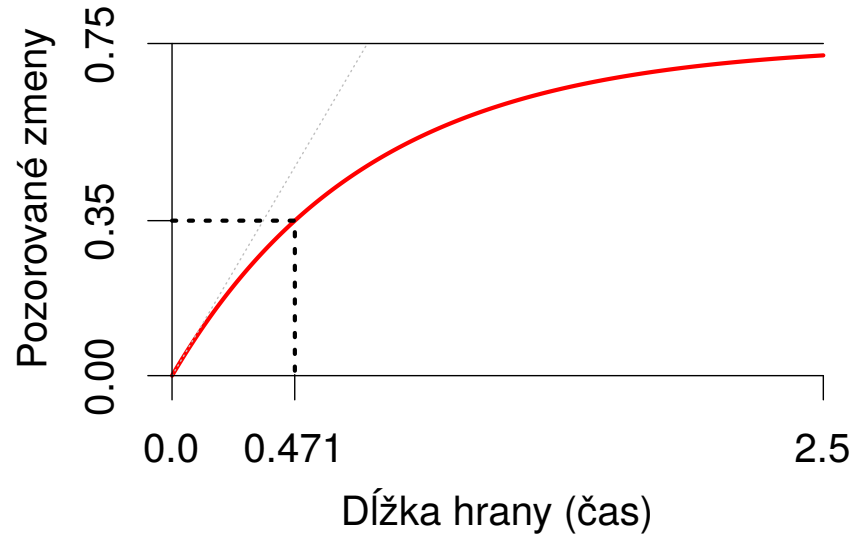
α : rýchlosť evolúcie (počet substitúcií na jednotku času)

Očakávaný počet pozorovaných zmien na bázu za čas t :

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



Späť ku spájaniu susedov (Neighbor Joining)



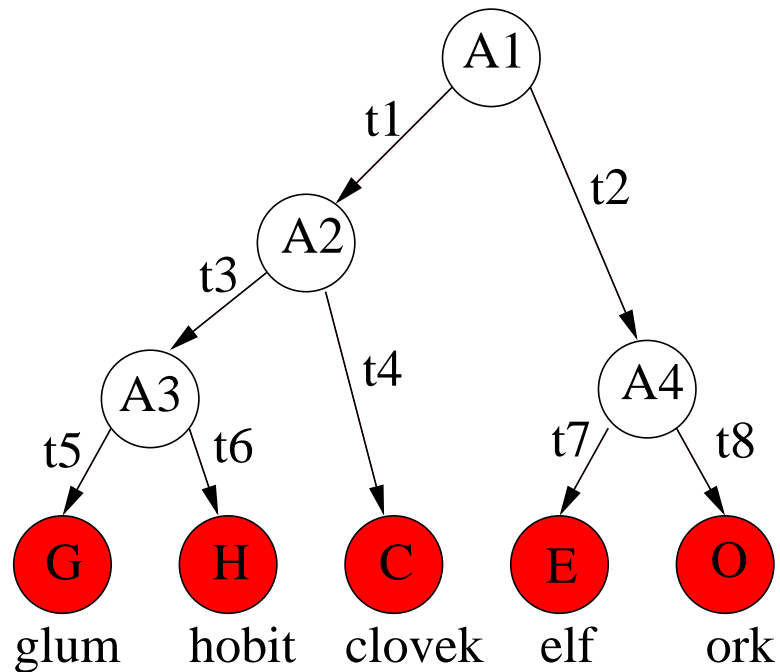
- Podľa takéhoto modelu môžeme korigovať pozorované vzdialenosti

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

- Často lepšie zložitejšie modely, ktoré zahŕňajú rôzne frekvencie báz, pomer tranzícií a transverzií, variabilnú rýchlosť evolúcie na rôznych pozíciách

Najvierohodnejšie stromy (Maximum likelihood)

- Strom môžeme chápať ako **jednoduchý generatívny model**



- $\Pr(G, H, C, E, O, A1, \dots, A4) = \Pr(A1) \cdot \Pr(A2 | A1, t_1) \cdot \Pr(A4 | A1, t_2) \cdot \Pr(A3 | A2, t_3) \cdot \Pr(G | A3, t_5) \cdot \Pr(H | A3, t_6) \cdot \Pr(C | A2, t_4) \cdot \Pr(E | A4, t_7) \cdot \Pr(O | A4, t_8)$

- $\Pr(G, H, C, E, O, A_1, \dots, A_4) = \Pr(A_1) \cdot \Pr(A_2 | A_1, t_1) \cdot \Pr(A_4 | A_1, t_2) \cdot \Pr(A_3 | A_2, t_3) \cdot \Pr(G | A_3, t_5) \cdot \Pr(H | A_3, t_6) \cdot \Pr(C | A_2, t_4) \cdot \Pr(E | A_4, t_7) \cdot \Pr(O | A_4, t_8)$
- Pre **daný strom** a **dané dĺžky hrán** možno jednotlivé pravdepodobnosti spočítať použitím evolučného modelu (napr. Jukes-Cantor)

- **Vierohodnosť (likelihood) stromu:**

$$\Pr(G, H, C, E, O) = \sum_{A_1, \dots, A_4} \Pr(G, H, C, E, O, A_1, \dots, A_4)$$

- Rátame pomocou **Felsensteinovho algoritmu** (jednoduché dynamické programovanie, podobne ako pre parsimony)
- \Rightarrow Pre daný strom a dĺžky hrán vieme spočítať vierohodnosť v čase $O(m)$

Ako nájsť najvieryhodnejší strom?

- Problém je NP-ťažký ;
navyše komplikovaný tým, že na výpočet vierohodnosti **potrebujeme aj dĺžky hrán**
- Opäť použijeme heuristické vyhľadávanie:
 - Začneme s “rozumným” stromom
 - Vypočítame vierohodnosť tohto stromu:
 - * Začneme s “rozumnými” dĺžkami hrán
 - * Vypočítame vierohodnosť stromu s dĺžkami
 - * Mierne zmeníme dĺžky tak, aby sa zlepšila vierohodnosť a opakujeme
 - Pomocou stanovených operácií (ako v prípade parsimony) skúsime “podobné” stromy, až kým nevieme zlepšiť

“Správnosť” fylogenetických algoritmov: Konzistentnosť

- “Rozumne” správajúce sa algoritmy: ak množstvo dát (n) rastie, ich odpoveď by sa mala približovať ku správnej odpovedi.
- Hovoríme, že algoritmus pre hľadanie fylogenetického stromu je **konzistentný**, ak v prípade, že n ide do nekonečna, pravdepodobnosť správneho stromu konverguje k 1.

Porovnanie algoritmov

	Zložitosť	Konzistentný	Využitie dát
Parsimony (úspornosť)	NP-ťažký	NIE	celé sekvencie
Neighbor Joining	$O(m^3)$	ÁNO	iba vzdialenosti
Likelihood (vierohodnosť)	NP-ťažký	ÁNO	celé sekvencie

Odkiaľ zohnať dáta pre fylogenetiku?

Často sa používajú špeciálne sekvencie
(napr. gény ribozomálnej RNA, mitochondriálny genóm)

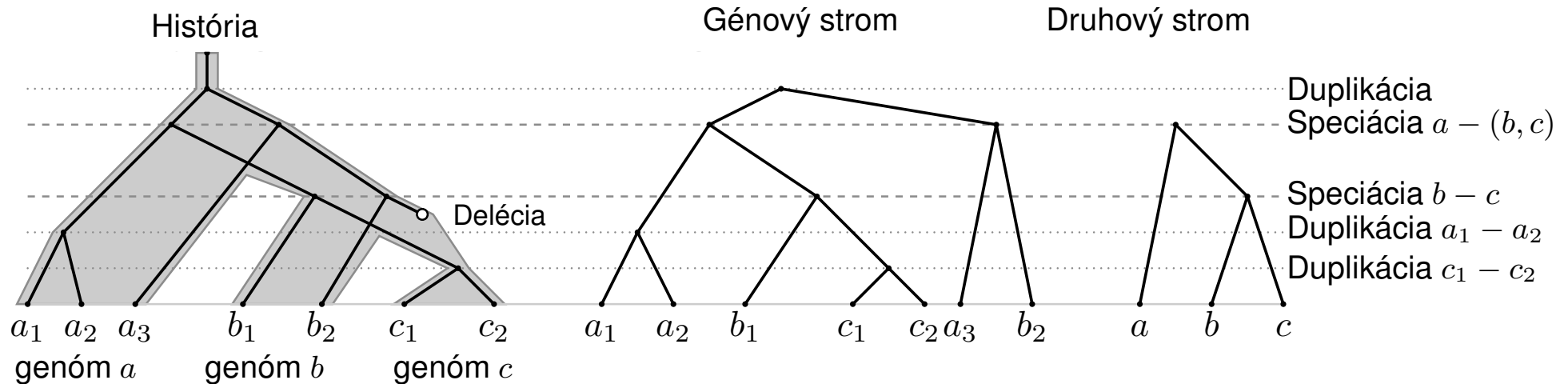
Chceme využiť aj ďalšie časti genómu. Čo tak:

- Vybrať si sympatický gén
- Nájsť jeho homológy v iných genómoch
- Použiť tieto na konštrukciu fylogenetického stromu
(DNA sekvencie alebo proteíny)

Problém: počas evolúcie sa časť genómu s vybraným génom mohla duplikovať

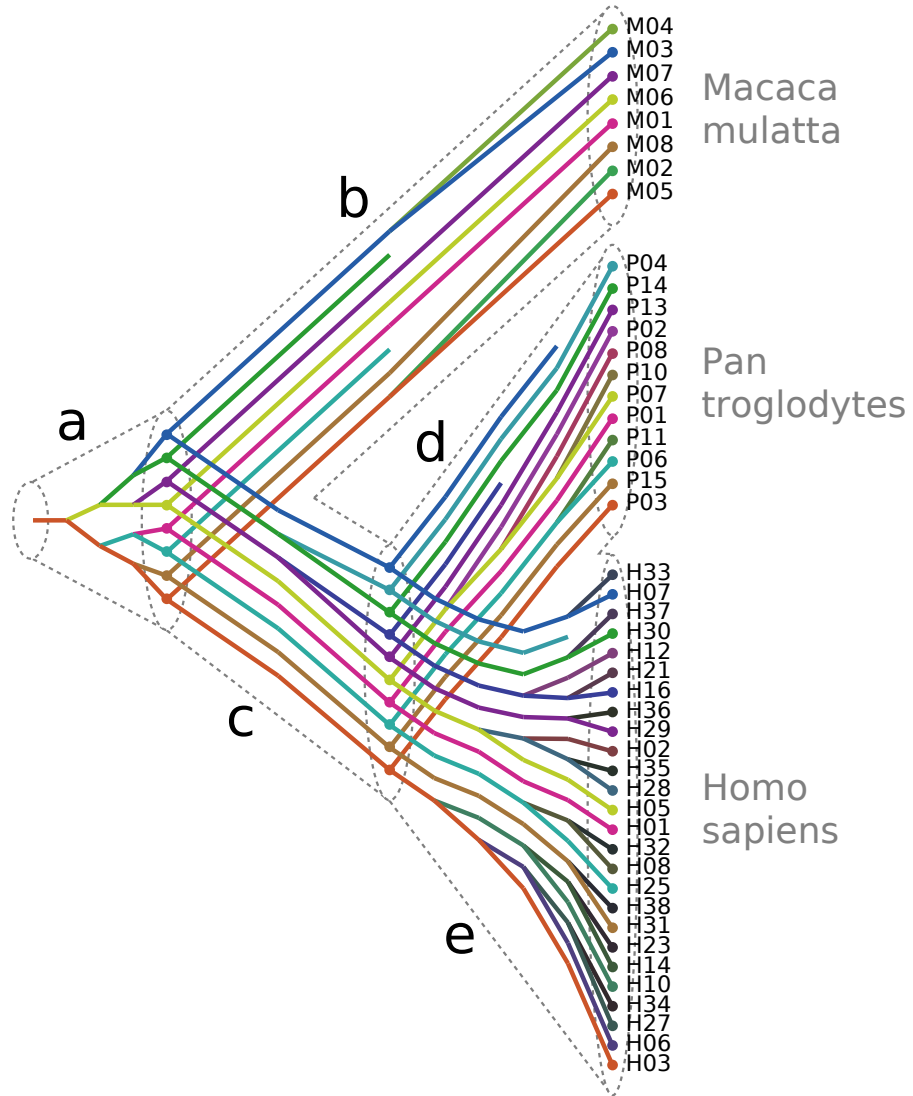
História duplikovaného génu

Príklad: organizmy a , b , c , gény $a_1, a_2, a_3, b_1, b_2, c_1, c_2$



- **Homológ:** vyvinuli sa zo spoločného predka, podobná sekvencia
- **Ortológ:** najbližší spoločný predok je speciácia (napr. dvojice génov $a_1 - b_1, a_2 - b_1$)
- **Paralóg:** najbližší spoločný predok je duplikácia (napr. dvojice génov $a_1 - a_2, a_1 - b_2$)

Zložitejší príklad duplikácie génu:



Zhrnutie

- Modely evolúcie nukleotidov nám dávajú možnosť:
 - Odhadovať skutočnú evolučnú vzdialenosť (počet substitúcií) z počtu pozorovaných zmien medzi sekvenciami
 - Počítať pravdepodobnosť, že uvidíme zmenu nukleotidu za určitý čas t
- Tri metódy na vytváranie evolučných stromov:
 - Úsporné stromy (parsimony)
 - Spájanie susedov (neighbour joining)
 - Vierohodnosť stromov (maximum likelihood)
- Génové a druhové stromy; komplikácie pri vytváraní stromov