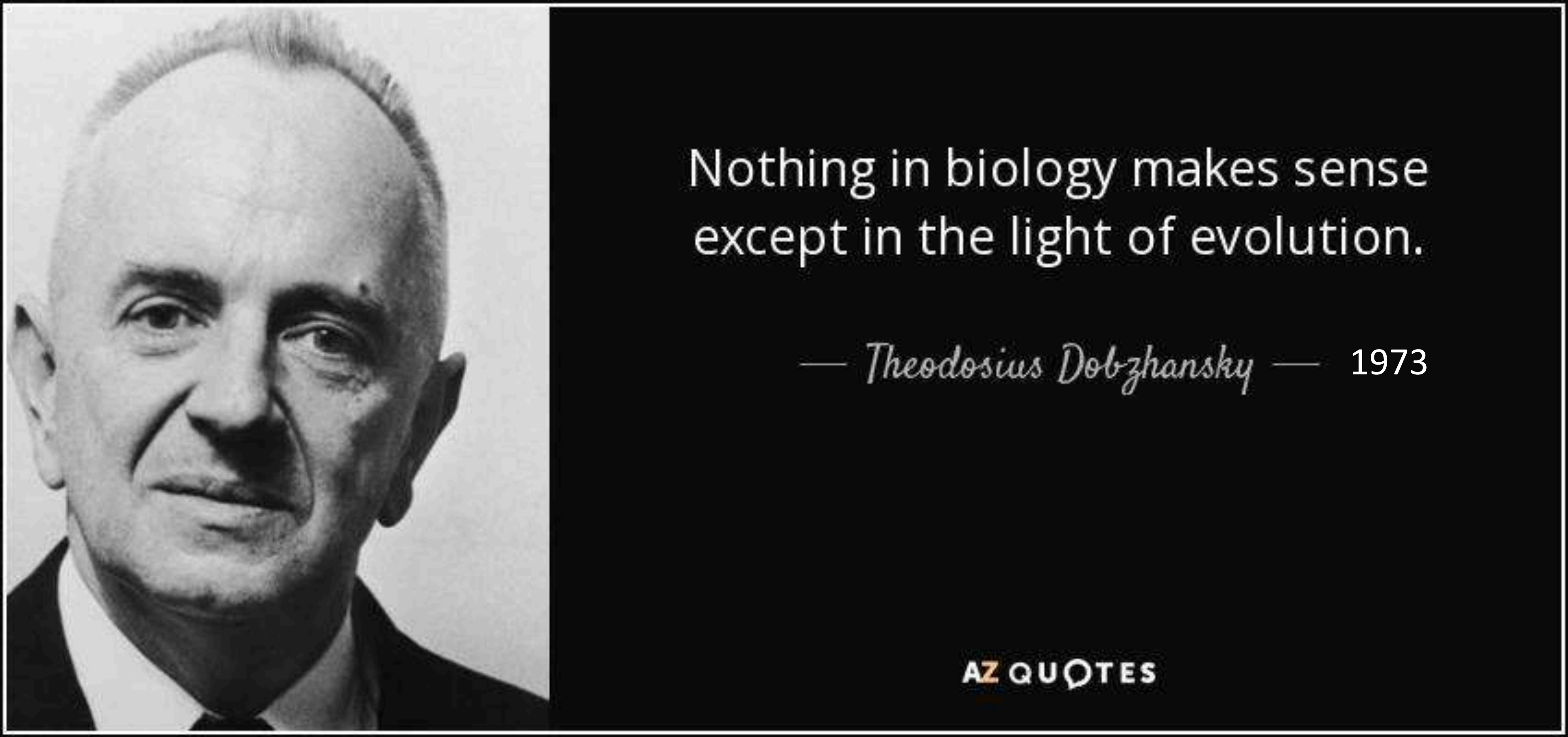
From sequence alignment to phylogenetic tree

Jia-Ming Chang

Department of Computer Science
National Chengchi University, Taiwan

政大資訊科學系
Department of Computer Science, National Chengchi University



Nothing in biology makes sense except in the light of evolution.

— Theodosius Dobzhansky — 1973

https://biologie-lernprogramme.de/daten/programme/js/homologer/daten/lit/Dobzhansky.pdf

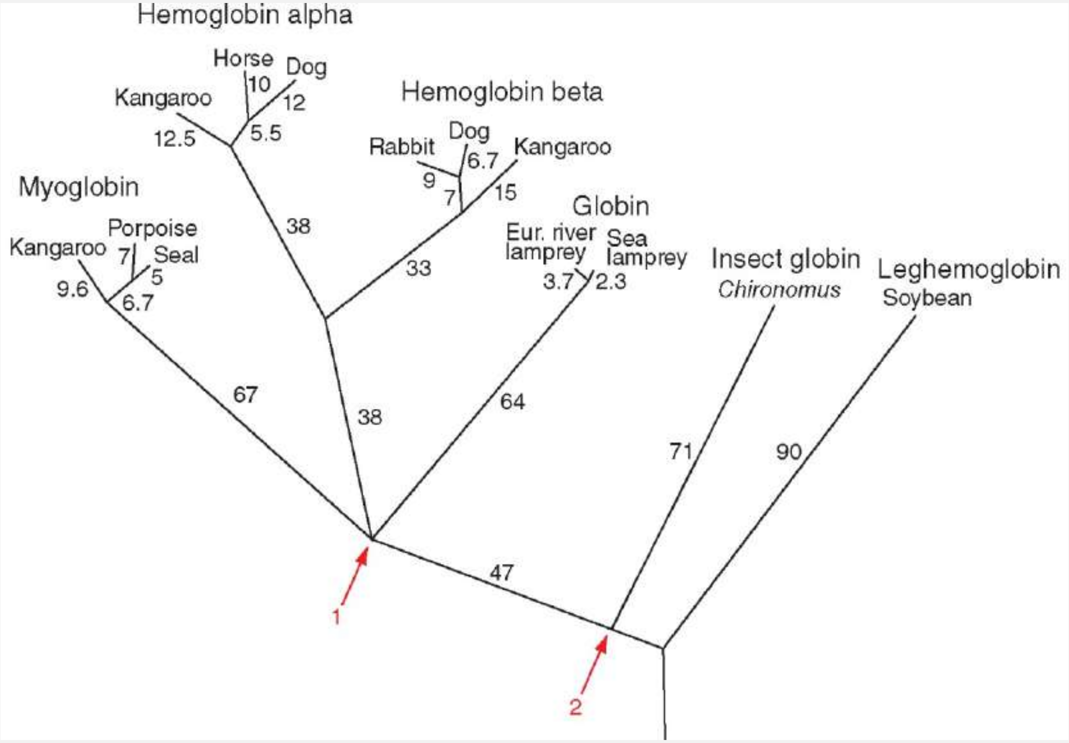
若不採用演化論，生物學的一切都說不通

Evolution

- Charles Darwin's 1859 book (*On the Origin of Species By Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*) introduced the theory of evolution.
- At the molecular level, evolution is a process of mutation with selection.
- Molecular evolution is the study of changes in genes and proteins throughout different branches of the tree of life.
- Phylogeny is the inference of evolutionary relationships.

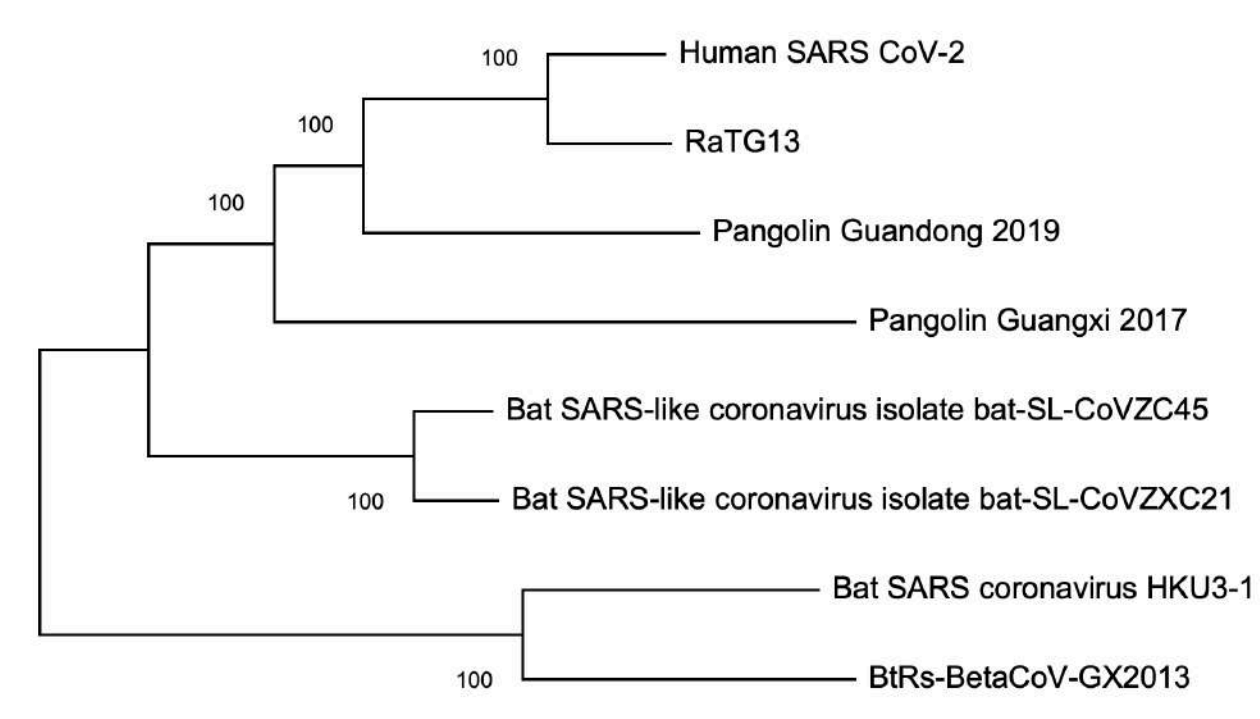
1960s: globin phylogeny

- tree of 13 orthologs by Margaret Dayhoff and colleagues
  - Arrow 1: node corresponding to last common ancestor of a group of vertebrate globins.
  - Arrow 2: ancestor of insect and vertebrate globins



The neighbor-joining tree of SARS-CoV-2 related coronaviruses

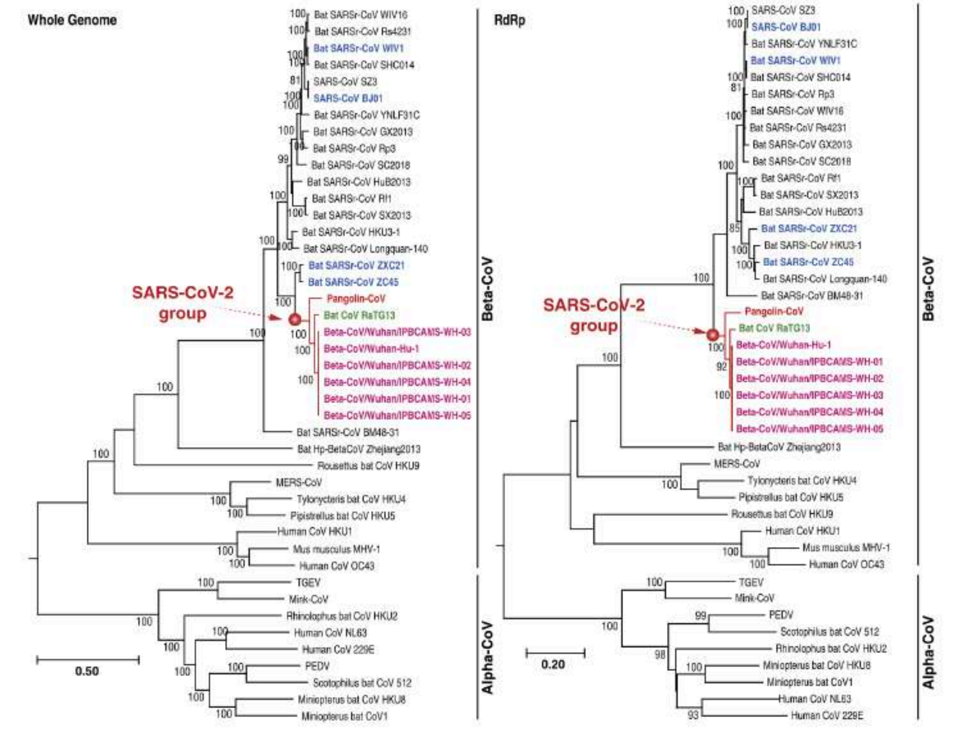
- CDSs were aligned based on translated amino acid sequences using MUSCLE v3.8.31 …
- Phylogenetic relationships were constructed using the neighbor-joining method based on Kimura’s two-parameter model.


100
Human SARS CoV-2
RaTG13
100
100
Pangolin Guandong 2019
Pangolin Guangxi 2017
Bat SARS-like coronavirus isolate bat-SL-CoVZC45
100
Bat SARS-like coronavirus isolate bat-SL-CoVZXC21
Bat SARS coronavirus HKU3-1
100
BtRs-BetaCoV-GX2013

The origin and underlying driving forces of the SARS-CoV-2 outbreak, Shu-Miaw Chaw, Jui-Hung Tai, Shi-Lun Chen, Chia-Hung Hsieh, Sui-Yuan Chang, Shiou-Hwei Yeh, Wei-Shiung Yang, Pei-Jer Chen, Hurng-Yi Wang bioRxiv 2020.04.12.038554; doi: https://doi.org/10.1101/2020.04.12.038554

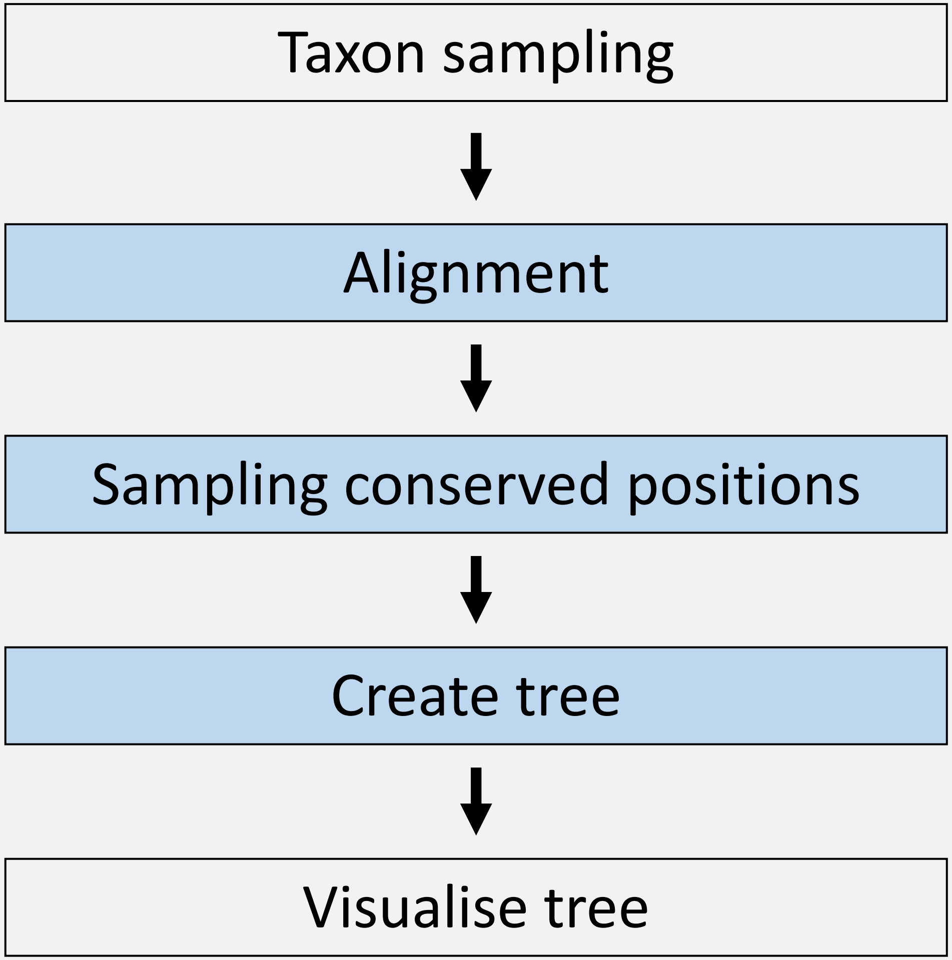
Phylogenetic Relationship of CoVs

- Sequence alignment was carried out using MUSCLE software.
- Gblocks was used to process the gap in the aligned sequence.
- Using MegaX, we inferred all maximum likelihood phylogenetic trees.



Zhang, T., Wu, Q. & Zhang, Z. Probable Pangolin Origin of SARS-CoV-2 Associated with the COVID-19 Outbreak. *Curr Biology Cb* **30**, 1346-1351.e2 (2020).

Flow to build Phylogenetic tree



Taxon sampling

Alignment

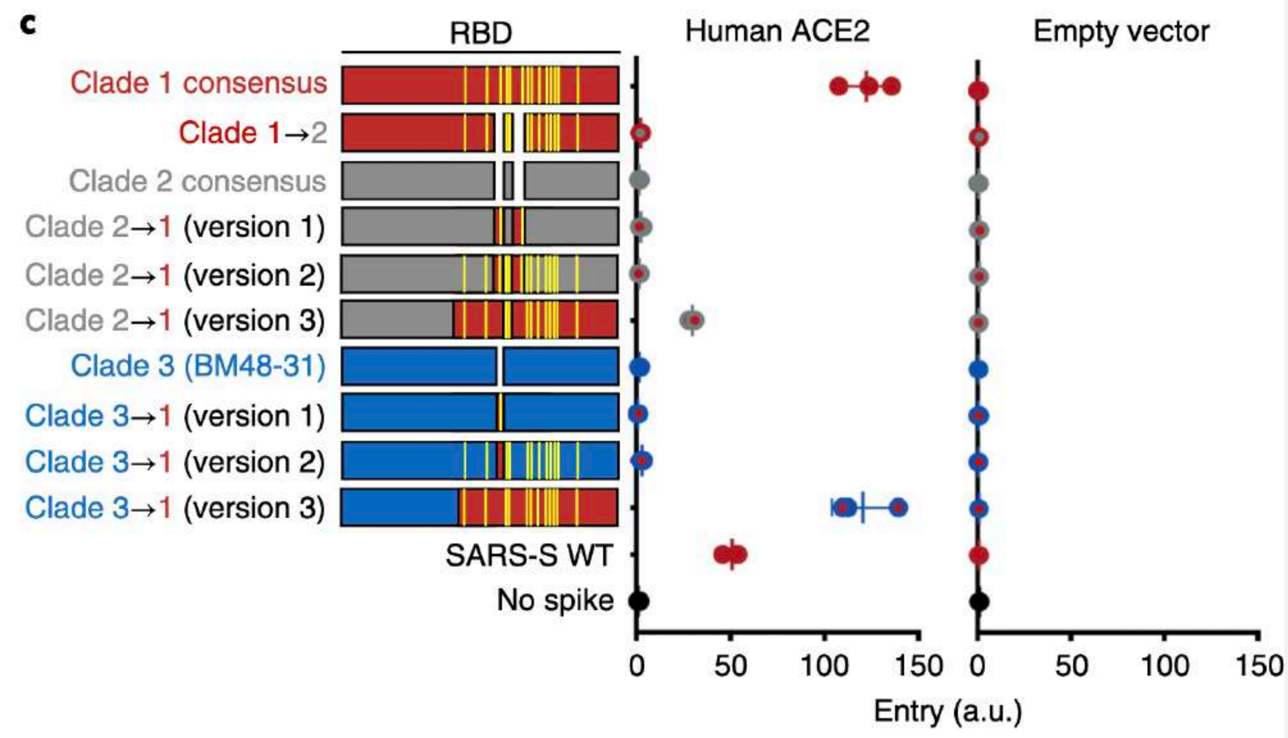
Sampling conserved positions

Create tree

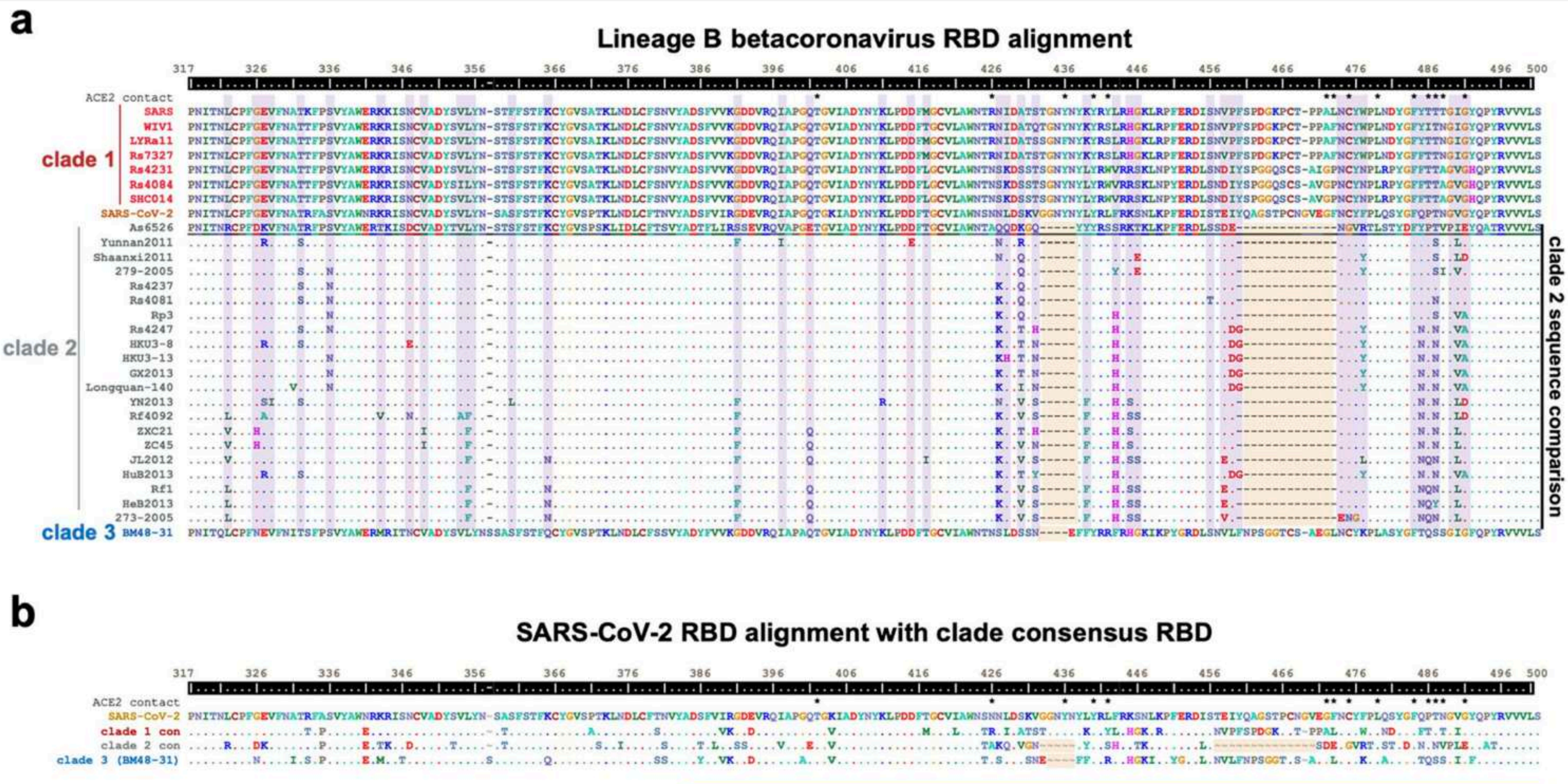
Visualise tree

Lineage B clade-specific determinants for human ACE2 usage

- Replacing all 14 contact points and the surrounding amino acids (known as the receptor-binding motif (RBM)) led to increased ACE2 entry with clade 2 and 3 RBDs
  - 2 → 1 (version 3) = clade 2 residues 322–400 + clade 1 residues 400–501
  - 3 → 1 (version 3) = clade 3 residues 322–385 + clade 1 residues 386–501



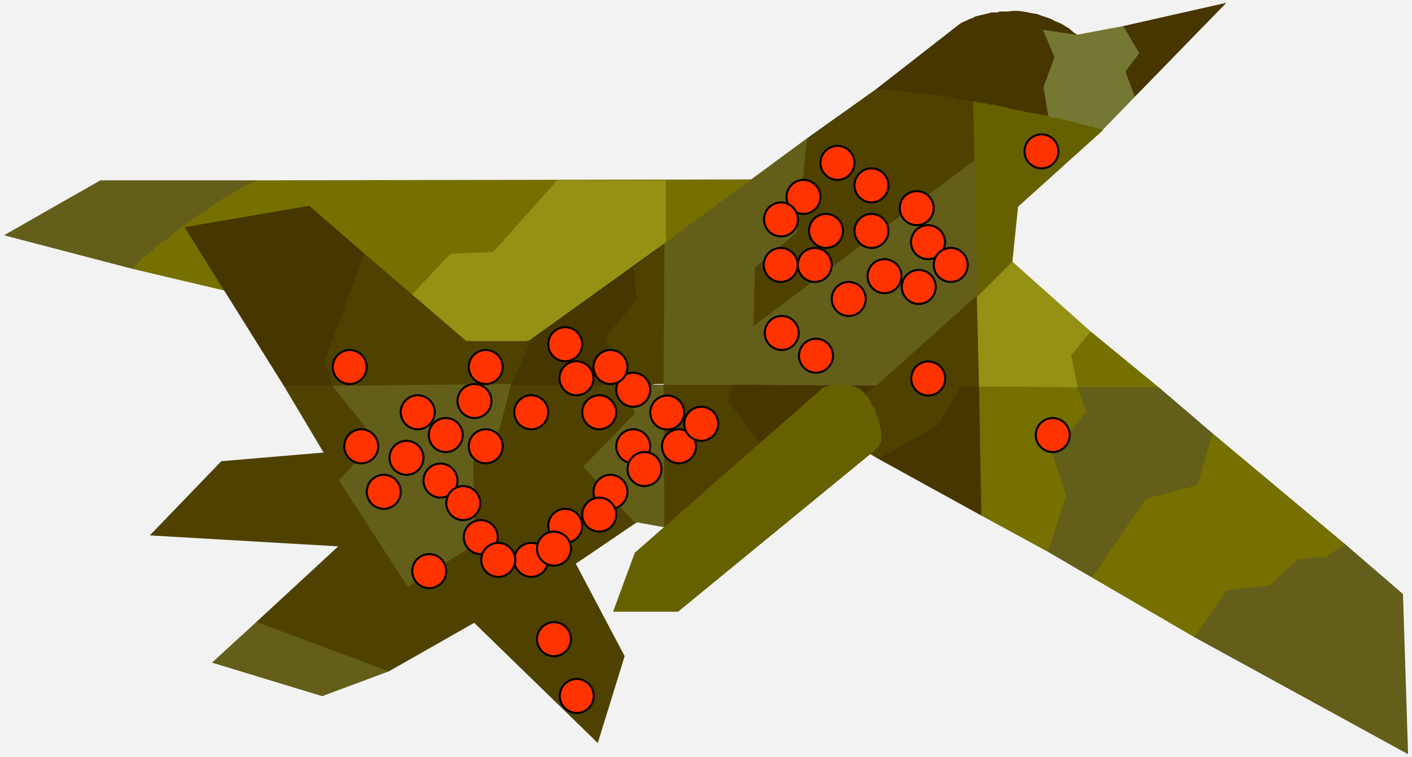
Letko, M., Marzi, A. & Munster, V. Functional assessment of cell entry and receptor usage for SARS-CoV-2 and other lineage B betacoronaviruses. *Nat Microbiol* **5**, 562–569 (2020).



Extended Data Fig. 4 | Lineage B panel RBD sequence features. a, Amino acid sequences corresponding to SARS-spike residues 317 through 500 were aligned with ClustalW. Contact points between SARS-spike and human ACE2 are indicated with an (*). Clade 2 sequences are shown as compared to clade 2 As6526, with identical residues indicated with a (.) and sites that vary between clade 2 viruses highlighted in purple. Loop deletions are highlighted in orange. **b**, Amino acid alignment of 2019-nCoV RBD and consensus RBD sequences for clade 1 and 2 and BM48-31 (clade 3). Loop deletions are highlighted in orange.

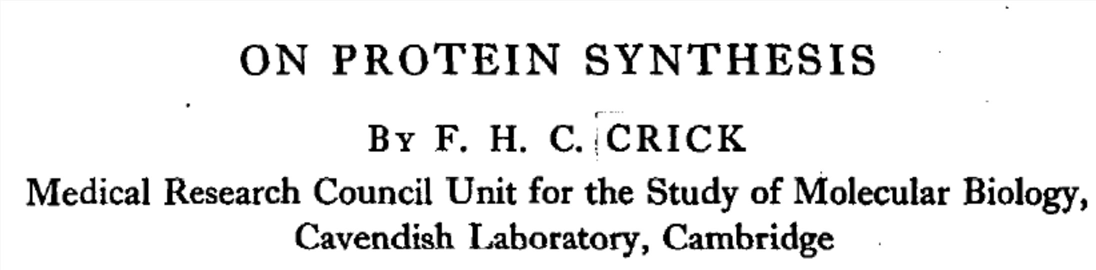
Letko, M., Marzi, A. & Munster, V. Functional assessment of cell entry and receptor usage for SARS-CoV-2 and other lineage B betacoronaviruses. *Nat Microbiol* **5**, 562–569 (2020).

1. Sequence Alignment



Manguel M, Samaniego F.J., ***Abraham Wald's Work on Aircraft Suvivability,*** J. American Statistical Association. 79, 259-270, (1984)

Adapted from Cedric Notredame



ON PROTEIN SYNTHESIS

By F. H. C. CRICK

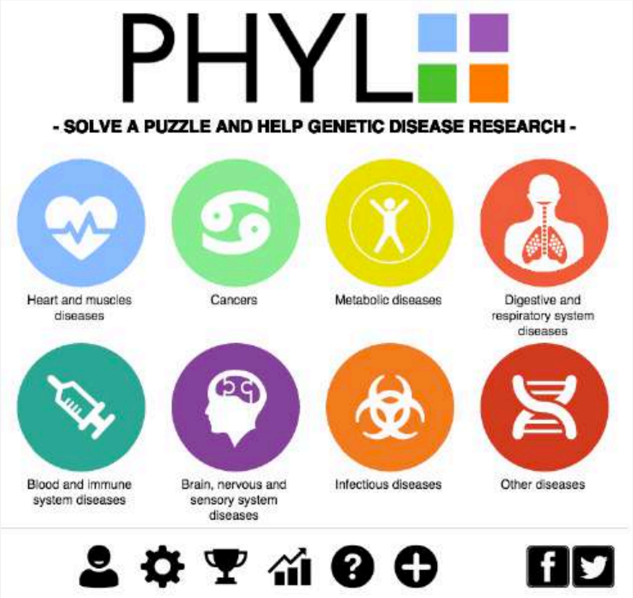
Medical Research Council Unit for the Study of Molecular Biology,
Cavendish Laboratory, Cambridge

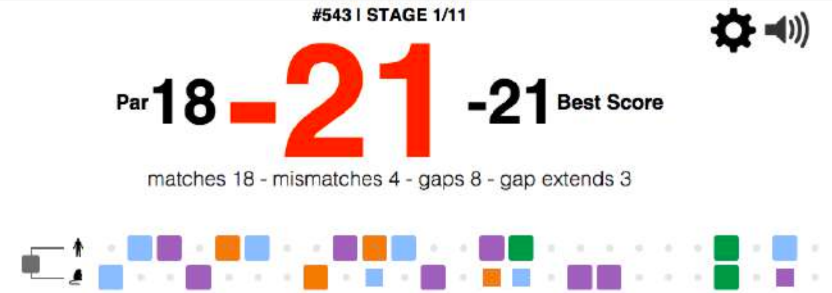
- Biologists should realise that before long we shall have a subject which might be called ‘protein taxonomy’—the study of the amino acid sequences of the proteins of an organism and the comparison of them between species.
- It can be argued that these sequences are the most delicate expression possible of the phenotype of an organism and that vast amounts of evolutionary information may be hidden away within them.

Crick FHC. On protein synthesis. *Symp Soc Exp Biol.* 1958; 12:138–163. PMID: 13580867

Sequence alignment

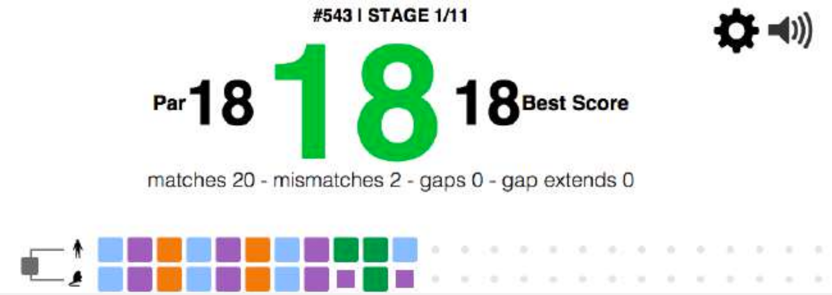
http://phylo.cs.mcgill.ca/


PHYLO
- SOLVE A PUZZLE AND HELP GENETIC DISEASE RESEARCH -
Heart and muscles diseases
Cancers
Metabolic diseases
Digestive and respiratory system diseases
Blood and immune system diseases
Brain, nervous and sensory system diseases
Infectious diseases
Other diseases


#543 I STAGE 1/11

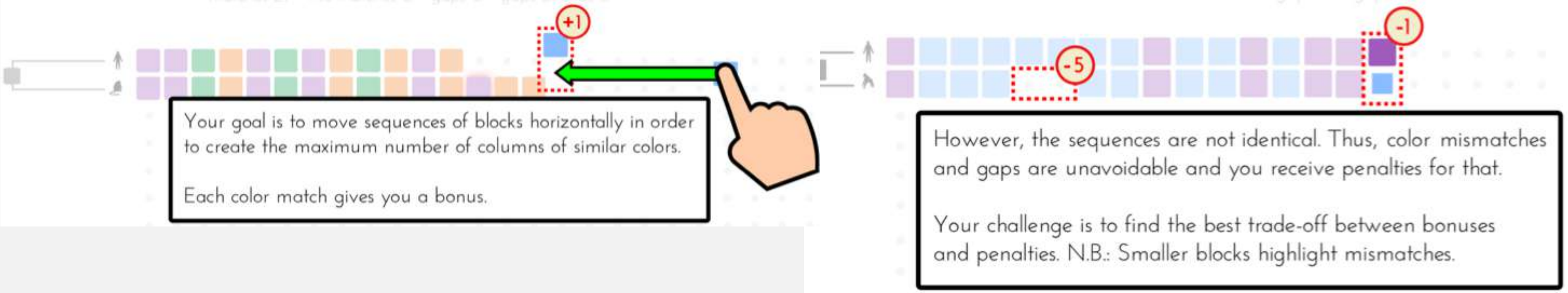
Par 18 -21 -21 Best Score

matches 18 - mismatches 4 - gaps 8 - gap extends 3


#543 I STAGE 1/11

Par 18 18 18 Best Score

matches 20 - mismatches 2 - gaps 0 - gap extends 0



Your goal is to move sequences of blocks horizontally in order to create the maximum number of columns of similar colors.

Each color match gives you a bonus.

However, the sequences are not identical. Thus, color mismatches and gaps are unavoidable and you receive penalties for that.

Your challenge is to find the best trade-off between bonuses and penalties. N.B.: Smaller blocks highlight mismatches.

1.1 Substitution Matrix

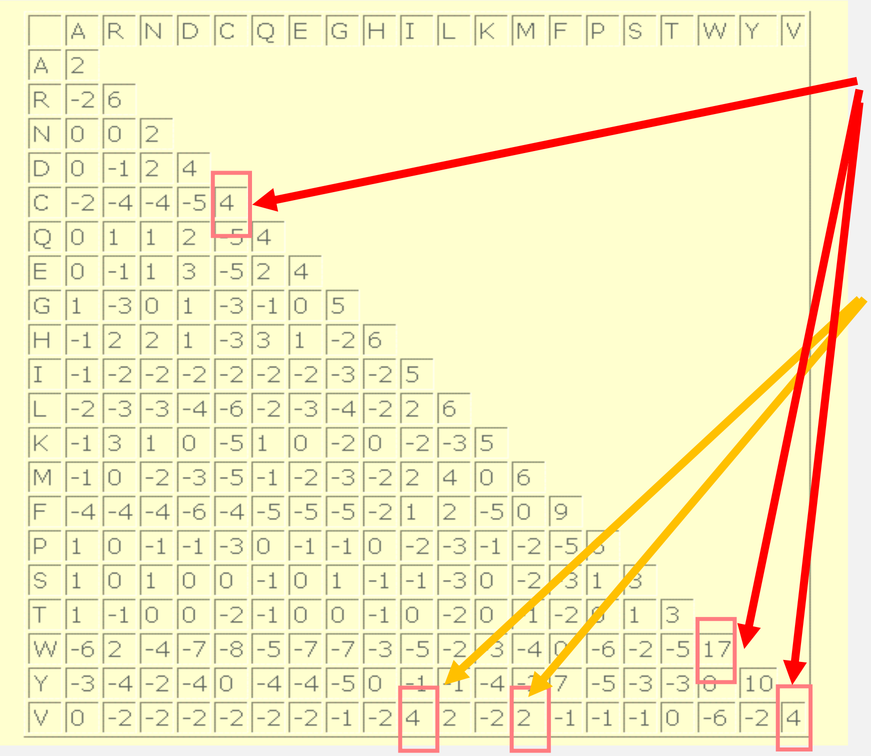
1. Sequence Alignment

How Can We Compare Sequences ?

- To compare Sequences, we need to compare residues
- We need to know how much it COSTS to SUBSTITUTE
  - an Alanine into an Isoleucine
  - a Tryptophan into a Glycine
- The table that contains the costs for all the possible substitutions is called the SUBSTITUTION MATRIX

Adapted from Cedric Notredame

Making a Substitution Matrix



	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	2																			
R	-2	6																		
N	0	0	2																	
D	0	-1	2	4																
C	-2	-4	-4	-5	4															
Q	0	1	1	2	-5	4														
E	0	-1	1	3	-5	2	4													
G	1	-3	0	1	-3	-1	0	5												
H	-1	2	2	1	-3	3	1	-2	6											
I	-1	-2	-2	-2	-2	-2	-2	-3	-2	5										
L	-2	-3	-3	-4	-6	-2	-3	-4	-2	2	6									
K	-1	3	1	0	-5	1	0	-2	0	-2	-3	5								
M	-1	0	-2	-3	-5	-1	-2	-3	-2	2	4	0	6							
F	-4	-4	-4	-6	-4	-5	-5	-5	-2	1	2	-5	0	9						
P	1	0	-1	-1	-3	0	-1	-1	0	-2	-3	-1	-2	-5	6					
S	1	0	1	0	0	-1	0	1	-1	-1	-3	0	-2	-3	1	3				
T	1	-1	0	0	-2	-1	0	0	-1	0	-2	0	-1	-2	0	1	3			
W	-6	2	-4	-7	-8	-5	-7	-7	-3	-5	-2	-3	-4	0	-6	-2	-5	17		
Y	-3	-4	-2	-4	0	-4	-4	-5	0	-1	-1	-4	-2	7	-5	-3	-3	0	10	
V	0	-2	-2	-2	-2	-2	-2	-1	-2	4	2	-2	2	-1	-1	-1	0	-6	-2	4

The Diagonal Indicates How Conserved a residue tends to be.

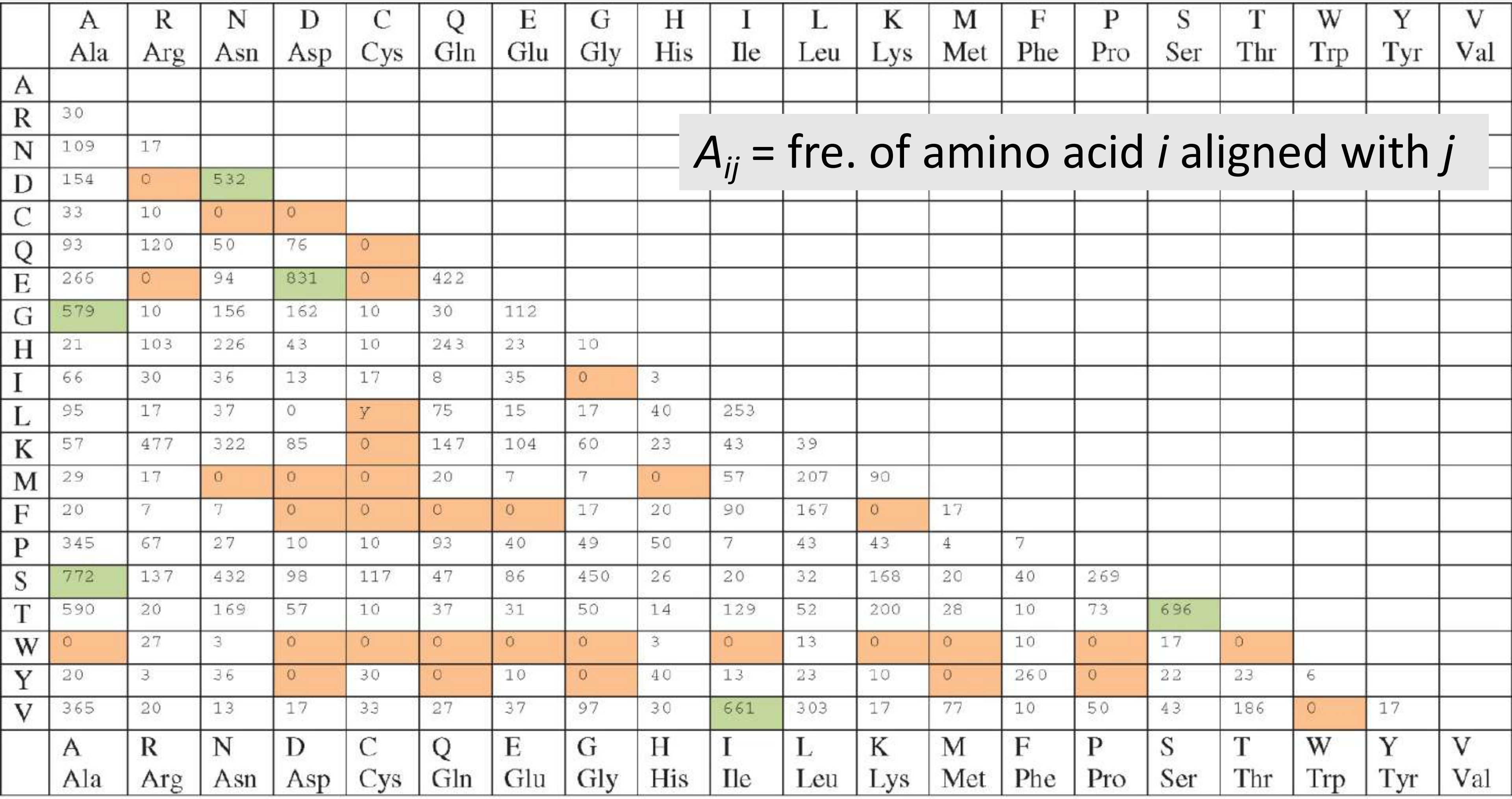
W is VERY Conserved

Some Residues are Easier To mutate into other similar.

Adapted from Cedric Notredame

How to derive that matrix?

PAM



	A Ala	R Arg	N Asn	D Asp	C Cys	Q Gln	E Glu	G Gly	H His	I Ile	L Leu	K Lys	M Met	F Phe	P Pro	S Ser	T Thr	W Trp	Y Tyr	V Val
A																				
R	30																			
N	109	17																		
D	154	0	532																	
C	33	10	0	0																
Q	93	120	50	76	0															
E	266	0	94	831	0	422														
G	579	10	156	162	10	30	112													
H	21	103	226	43	10	243	23	10												
I	66	30	36	13	17	8	35	0	3											
L	95	17	37	0	y	75	15	17	40	253										
K	57	477	322	85	0	147	104	60	23	43	39									
M	29	17	0	0	0	20	7	7	0	57	207	90								
F	20	7	7	0	0	0	0	17	20	90	167	0	17							
P	345	67	27	10	10	93	40	49	50	7	43	43	4	7						
S	772	137	432	98	117	47	86	450	26	20	32	168	20	40	269					
T	590	20	169	57	10	37	31	50	14	129	52	200	28	10	73	696				
W	0	27	3	0	0	0	0	0	3	0	13	0	0	10	0	17	0			
Y	20	3	36	0	30	0	10	0	40	13	23	10	0	260	0	22	23	6		
V	365	20	13	17	33	27	37	97	30	661	303	17	77	10	50	43	186	0	17	
	A Ala	R Arg	N Asn	D Asp	C Cys	Q Gln	E Glu	G Gly	H His	I Ile	L Leu	K Lys	M Met	F Phe	P Pro	S Ser	T Thr	W Trp	Y Tyr	V Val

A_{ij} = fre. of amino acid *i* aligned with *j*

FIGURE 3.8 Numbers of accepted point mutations, multiplied by 10, in 1572 cases of amino acid substitutions from closely related protein sequences. Amino acids are presented alphabetically according to the three-letter code. Notice that some substitutions (green shaded boxes) are very commonly accepted (such as V and I or S and T). Other amino acids, such as C and W, are rarely substituted by any other residue (orange shaded boxes).

Bioinformatics and Functional Genomics, Third Edition, Jonathan Pevsner.
© 2015 John Wiley & Sons, Ltd. Published 2015 by John Wiley & Sons, Ltd.
Companion Website: www.wiley.com/go/pevsnerbioinformatics

Normalized frequencies of amino acids, f_i

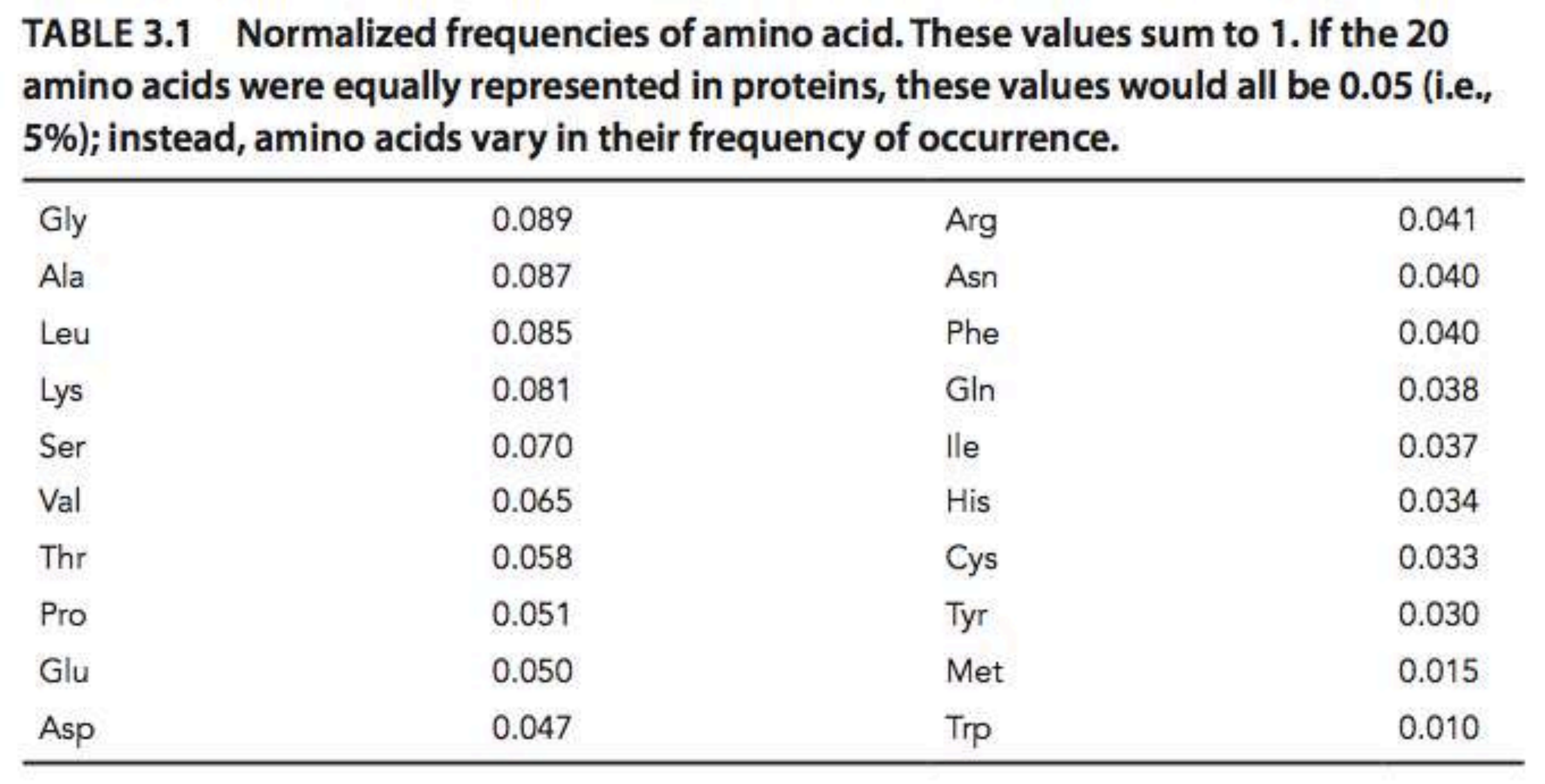


TABLE 3.1 **Normalized frequencies of amino acid. These values sum to 1. If the 20 amino acids were equally represented in proteins, these values would all be 0.05 (i.e., 5%); instead, amino acids vary in their frequency of occurrence.**

Gly	0.089	Arg	0.041
Ala	0.087	Asn	0.040
Leu	0.085	Phe	0.040
Lys	0.081	Gln	0.038
Ser	0.070	Ile	0.037
Val	0.065	His	0.034
Thr	0.058	Cys	0.033
Pro	0.051	Tyr	0.030
Glu	0.050	Met	0.015
Asp	0.047	Trp	0.010

some are more common (G, A, L, K) and some rare (C, Y, M, W).

Bioinformatics and Functional Genomics, 3rd Edition, Jonathan Pevsner, Fig. 3-1, Page 81

The relative mutability of amino acid *j*, m_j

- the # of *j* was observed to mutate / the overall occurrence frequency of *j* (f_j)
- In a scoring system alignment of two tryptophans will be weighted more heavily than two asparagines.

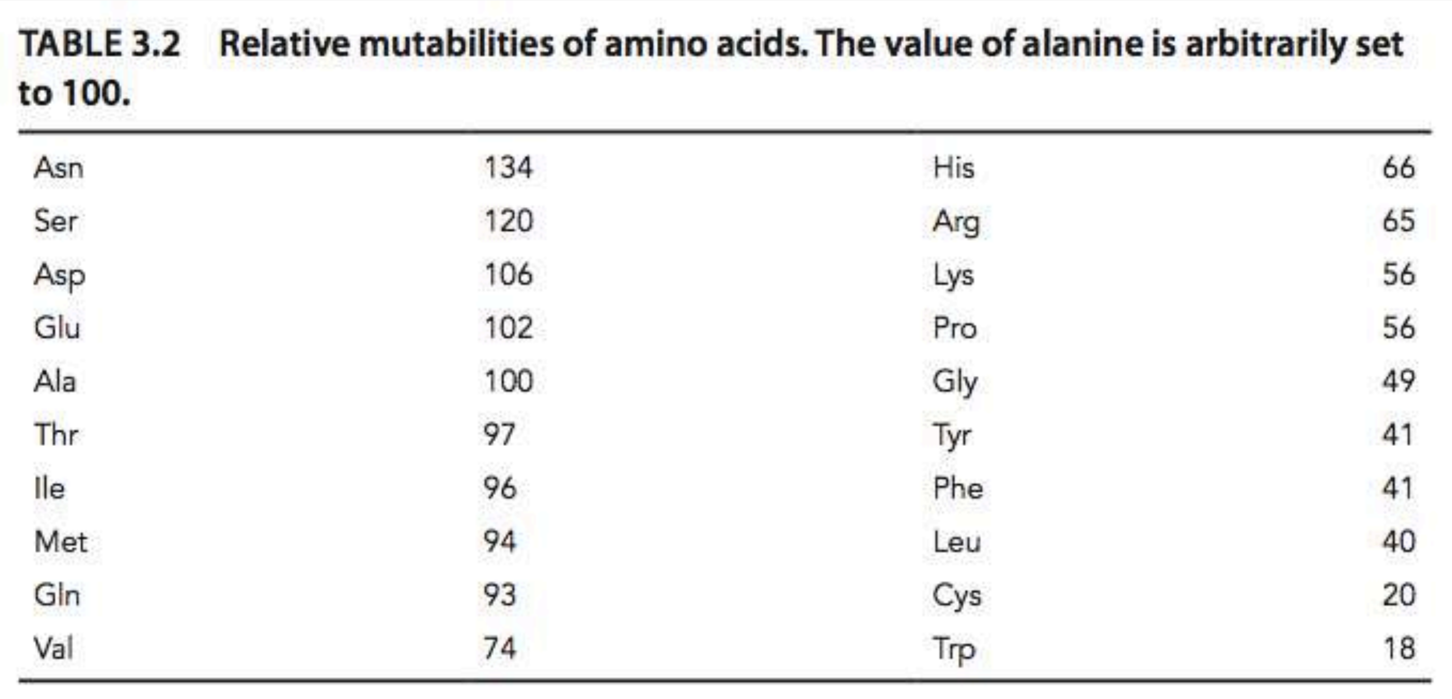


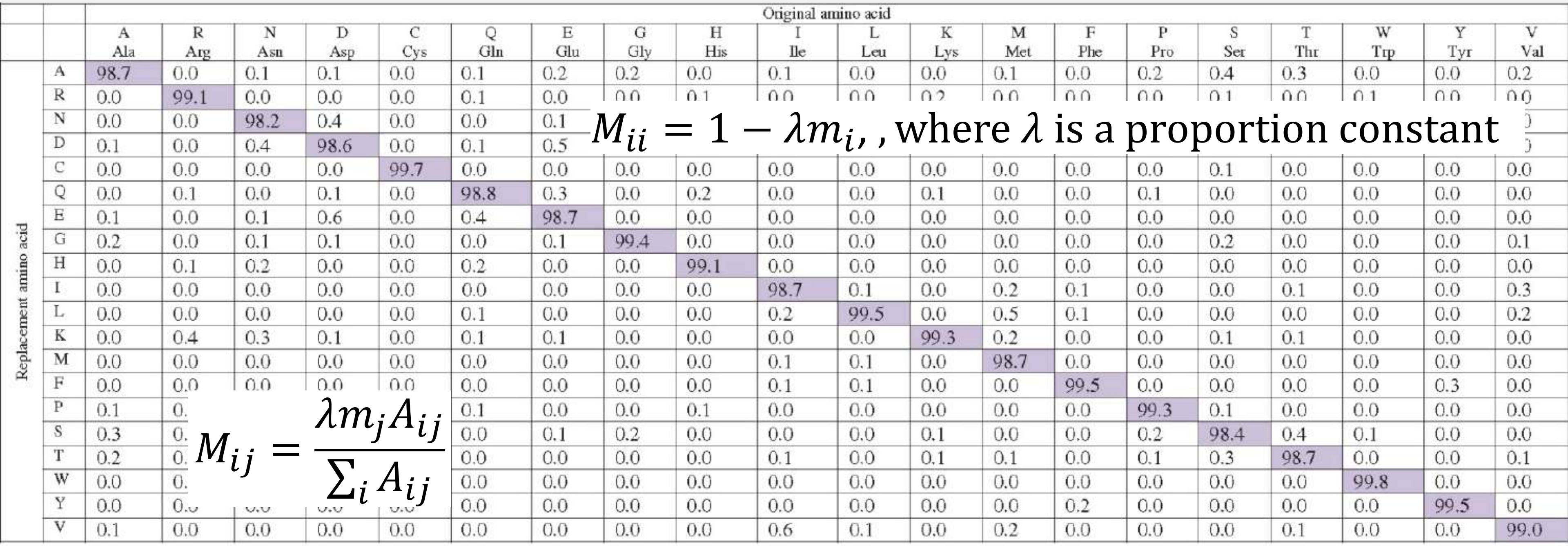
TABLE 3.2 Relative mutabilities of amino acids. The value of alanine is arbitrarily set to 100.

Asn	134	His	66
Ser	120	Arg	65
Asp	106	Lys	56
Glu	102	Pro	56
Ala	100	Gly	49
Thr	97	Tyr	41
Ile	96	Phe	41
Met	94	Leu	40
Gln	93	Cys	20
Val	74	Trp	18

Bioinformatics and Functional Genomics, 3rd Edition, Jonathan Pevsner, Table 3-2, Page 82

Mutation matrix – original amino acids (columns) and replacements (rows)

- The relative mutability of amino acid *j*



		Original amino acid																			
		A Ala	R Arg	N Asn	D Asp	C Cys	Q Gln	E Glu	G Gly	H His	I Ile	L Leu	K Lys	M Met	F Phe	P Pro	S Ser	T Thr	W Trp	Y Tyr	V Val
Replacement amino acid	A	98.7	0.0	0.1	0.1	0.0	0.1	0.2	0.2	0.0	0.1	0.0	0.0	0.1	0.0	0.2	0.4	0.3	0.0	0.0	0.2
	R	0.0	99.1	0.0	0.0	0.0	0.1	0.0	0.0	0.1	0.0	0.0	0.2	0.0	0.0	0.0	0.1	0.0	0.1	0.0	0.0
	N	0.0	0.0	98.2	0.4	0.0	0.0	0.1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	D	0.1	0.0	0.4	98.6	0.0	0.1	0.5	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	C	0.0	0.0	0.0	0.0	99.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0
	Q	0.0	0.1	0.0	0.1	0.0	98.8	0.3	0.0	0.2	0.0	0.0	0.1	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
	E	0.1	0.0	0.1	0.6	0.0	0.4	98.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	G	0.2	0.0	0.1	0.1	0.0	0.0	0.1	99.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.1
	H	0.0	0.1	0.2	0.0	0.0	0.2	0.0	0.0	99.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	I	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	98.7	0.1	0.0	0.2	0.1	0.0	0.0	0.1	0.0	0.0	0.3
	L	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.2	99.5	0.0	0.5	0.1	0.0	0.0	0.0	0.0	0.0	0.2
	K	0.0	0.4	0.3	0.1	0.0	0.1	0.1	0.0	0.0	0.0	0.0	99.3	0.2	0.0	0.0	0.1	0.1	0.0	0.0	0.0
	M	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.1	0.0	98.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	F	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.1	0.0	0.0	99.5	0.0	0.0	0.0	0.0	0.3	0.0
	P	0.1	0.	[illegible]	[illegible]	[illegible]	0.1	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	99.3	0.1	0.0	0.0	0.0	0.0
	S	0.3	0.	[illegible]	[illegible]	[illegible]	0.0	0.1	0.2	0.0	0.0	0.0	0.1	0.0	0.0	0.2	98.4	0.4	0.1	0.0	0.0
	T	0.2	0.	[illegible]	[illegible]	[illegible]	0.0	0.0	0.0	0.0	0.1	0.0	0.1	0.1	0.0	0.1	0.3	98.7	0.0	0.0	0.1
	W	0.0	0.	[illegible]	[illegible]	[illegible]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	99.8	0.0	0.0
	Y	0.0	0.	[illegible]	[illegible]	[illegible]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.0	99.5	0.0
	V	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.6	0.1	0.0	0.2	0.0	0.0	0.0	0.1	0.0	0.0	99.0

$M_{ii} = 1 - \lambda m_i, ,$ where λ is a proportion constant

$$M_{ij} = \frac{\lambda m_j A_{ij}}{\sum_i A_{ij}}$$

Bioinformatics and Functional Genomics, 3rd Edition, Jonathan Pevsner, Fig. 3-9,Page 84

From a mutation probability matrix to a log odds matrix

$$s_{ij} = 10 \times \log_{10}\left(\frac{M_{ij}}{f_i}\right) \text{, where } R_{ij} = \frac{M_{ij}}{f_i}$$

- M_{ij}: models the observed change
- f_i: the probability of a.a. *i* occurring in the second sequence by chance

- a log scoring matrix, why?
  - doing a pairwise alignment (or a BLAST search) we know what score to assign to two aligned amino acid residues.
  - Logarithms are easier to use for a scoring system => sum the scores of aligned residues rather than multiply them.

Bioinformatics and Functional Genomics, 3rd Edition, Jonathan Pevsner, Eq. 3-4, Page 89

What do the numbers mean in a log odds matrix?

- 0: neutral
- +2: indicates that the amino acid replacement occurs 1.6 times as frequently as expected by chance
- –10: that the correspondence of two amino acids in an alignment that accurately represents homology (evolutionary descent) is one tenth as frequent as the chance alignment of these amino acids

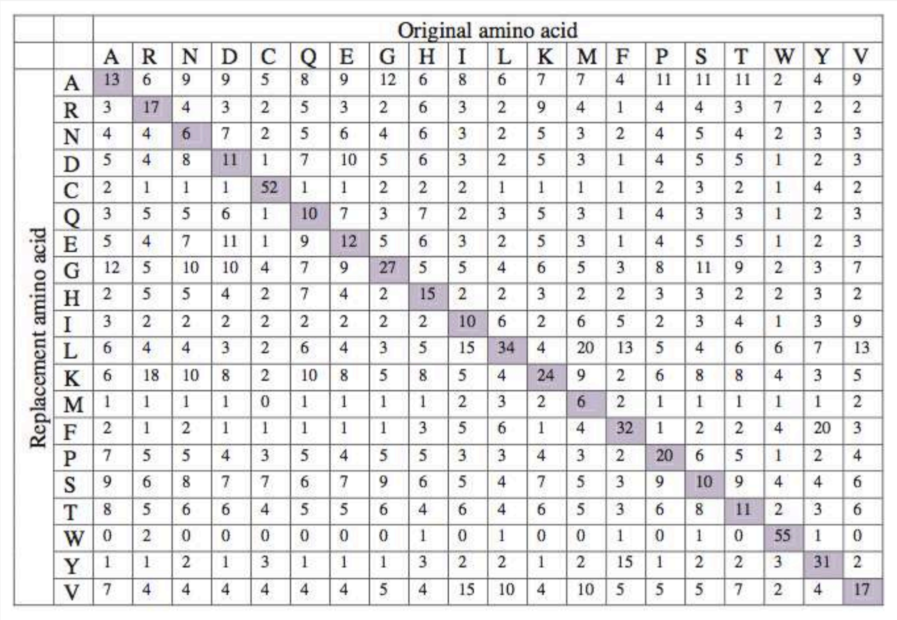
PAM matrices

- PAM1
  - At an evolutionary interval of PAM1, one change has occurred over a length of 100 amino acids.
- Other PAM matrices are extrapolated from PAM1
  - PAMx = multiplied PAM1 by itself
  - PAM250 matrix: for proteins that share ~20% identity

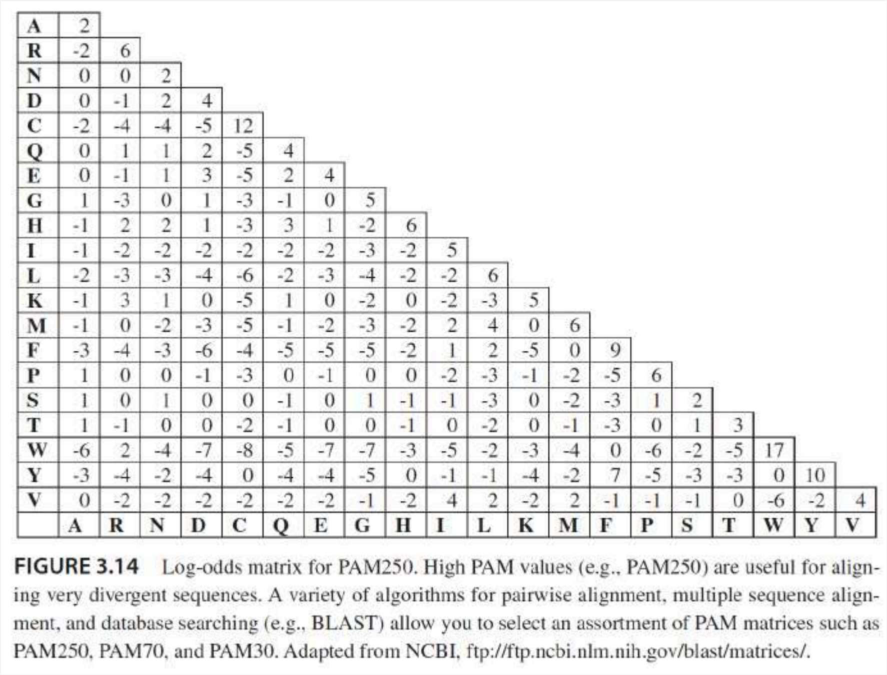
Adapted from Cedric Notredame

Mutation Matrix vs Log-odds score matrix

- Take PAM250 as an example, from asymmetric to symmetric, why?



		Original amino acid																			
		A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
Replacement amino acid	A	13	6	9	9	5	8	9	12	6	8	6	7	7	4	11	11	11	2	4	9
	R	3	17	4	3	2	5	3	2	6	3	2	9	4	1	4	4	3	7	2	2
	N	4	4	6	7	2	5	6	4	6	3	2	5	3	2	4	5	4	2	3	3
	D	5	4	8	11	1	7	10	5	6	3	2	5	3	1	4	5	5	1	2	3
	C	2	1	1	1	52	1	1	2	2	2	1	1	1	1	2	3	2	1	4	2
	Q	3	5	5	6	1	10	7	3	7	2	3	5	3	1	4	3	3	1	2	3
	E	5	4	7	11	1	9	12	5	6	3	2	5	3	1	4	5	5	1	2	3
	G	12	5	10	10	4	7	9	27	5	5	4	6	5	3	8	11	9	2	3	7
	H	2	5	5	4	2	7	4	2	15	2	2	3	2	2	3	3	2	2	3	2
	I	3	2	2	2	2	2	2	2	2	10	6	2	6	5	2	3	4	1	3	9
	L	6	4	4	3	2	6	4	3	5	15	34	4	20	13	5	4	6	6	7	13
	K	6	18	10	8	2	10	8	5	8	5	4	24	9	2	6	8	8	4	3	5
	M	1	1	1	1	0	1	1	1	1	2	3	2	6	2	1	1	1	1	1	2
	F	2	1	2	1	1	1	1	1	3	5	6	1	4	32	1	2	2	4	20	3
	P	7	5	5	4	3	5	4	5	5	3	3	4	3	2	20	6	5	1	2	4
	S	9	6	8	7	7	6	7	9	6	5	4	7	5	3	9	10	9	4	4	6
	T	8	5	6	6	4	5	5	6	4	6	4	6	5	3	6	8	11	2	3	6
	W	0	2	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	55	1	0
	Y	1	1	2	1	3	1	1	1	3	2	2	1	2	15	1	2	2	3	31	2
	V	7	4	4	4	4	4	4	5	4	15	10	4	10	5	5	5	7	2	4	17



A	2																			
R	-2	6																		
N	0	0	2																	
D	0	-1	2	4																
C	-2	-4	-4	-5	12															
Q	0	1	1	2	-5	4														
E	0	-1	1	3	-5	2	4													
G	1	-3	0	1	-3	-1	0	5												
H	-1	2	2	1	-3	3	1	-2	6											
I	-1	-2	-2	-2	-2	-2	-2	-3	-2	5										
L	-2	-3	-3	-4	-6	-2	-3	-4	-2	-2	6									
K	-1	3	1	0	-5	1	0	-2	0	-2	-3	5								
M	-1	0	-2	-3	-5	-1	-2	-3	-2	2	4	0	6							
F	-3	-4	-3	-6	-4	-5	-5	-5	-2	1	2	-5	0	9						
P	1	0	0	-1	-3	0	-1	0	0	-2	-3	-1	-2	-5	6					
S	1	0	1	0	0	-1	0	1	-1	-1	-3	0	-2	-3	1	2				
T	1	-1	0	0	-2	-1	0	0	-1	0	-2	0	-1	-3	0	1	3			
W	-6	2	-4	-7	-8	-5	-7	-7	-3	-5	-2	-3	-4	0	-6	-2	-5	17		
Y	-3	-4	-2	-4	0	-4	-4	-5	0	-1	-1	-4	-2	7	-5	-3	-3	0	10	
V	0	-2	-2	-2	-2	-2	-2	-1	-2	4	2	-2	2	-1	-1	-1	0	-6	-2	4
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V

FIGURE 3.14 Log-odds matrix for PAM250. High PAM values (e.g., PAM250) are useful for aligning very divergent sequences. A variety of algorithms for pairwise alignment, multiple sequence alignment, and database searching (e.g., BLAST) allow you to select an assortment of PAM matrices such as PAM250, PAM70, and PAM30. Adapted from NCBI, ftp://ftp.ncbi.nlm.nih.gov/blast/matrices/.

Why does PAM1 become symmetric?

- $M_{ij} = \frac{\lambda m_j A_{ij}}{\sum_{i=1,i\neq j}^{20} A_{ij}} = m_j \times \frac{\lambda A_{ij}}{\sum_{i=1,i\neq j}^{20} A_{ij}} = \frac{\sum_{i=1,i\neq j}^{20} A_{ij}}{f_j} \times \frac{\lambda A_{ij}}{\sum_{i=1,i\neq j}^{20} A_{ij}} = \frac{\lambda A_{ij}}{f_j}$

- $M_{ji} = \frac{\lambda m_i A_{ij}}{\sum_{i=1,i\neq j}^{20} A_{ij}} = m_i \times \frac{\lambda A_{ij}}{\sum_{i=1,i\neq j}^{20} A_{ij}} = \frac{\sum_{i=1,i\neq j}^{20} A_{ij}}{f_i} \times \frac{\lambda A_{ij}}{\sum_{i=1,i\neq j}^{20} A_{ij}} = \frac{\lambda A_{ij}}{f_i}$

- $R_{ij} = \frac{M_{ij}}{f_i} = \frac{\frac{\lambda A_{ij}}{f_j}}{f_i} = \frac{\lambda A_{ij}}{f_i f_j} = \frac{\frac{\lambda A_{ij}}{f_i}}{f_j} = \frac{M_{ji}}{f_j} = R_{ji}$

BLOcks SUbstitution Matrix (BLOSUM)

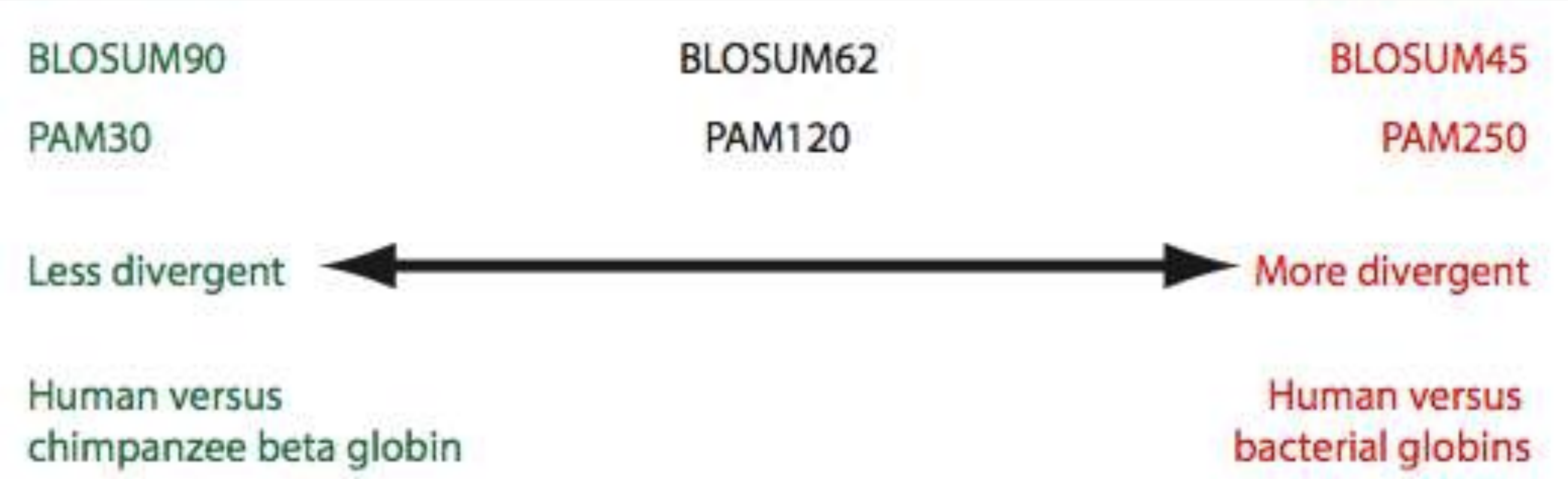
Henikoff, S.; Henikoff, J.G. (1992). "Amino Acid Substitution Matrices from Protein Blocks". *PNAS*. 89 (22): 10915–10919

Procedure of BLOSUM

- Cluster together sequences in a family whenever more than **L**% identical residues are shared, for BLOSUM-**L**.
- Based on local alignments & use aligned ungapped regions of protein families.
- Count number of substitutions across different clusters (in the same family).
- Estimate frequencies using the counts.

Summary of PAM and BLOSUM matrices

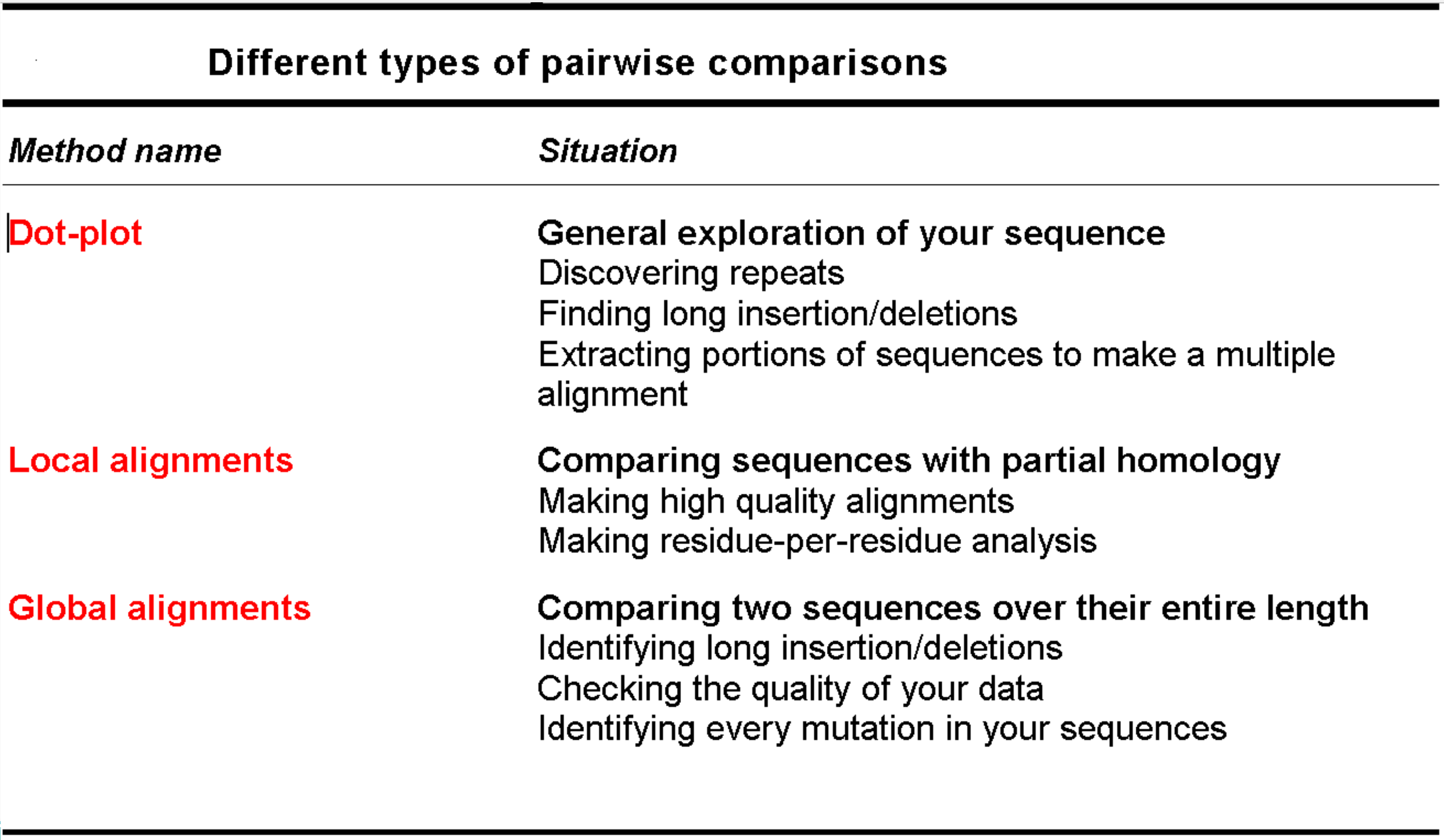
- BLOSUM62 is a matrix calculated from comparisons of sequences with no less than 62% divergence.
  - the default matrix in BLAST 2.0
  - Most widely used (PAM250)
- A higher PAM number, and a lower BLOSUM number, tends to correspond to a matrix tuned to more divergent proteins.



Bioinformatics and Functional Genomics, 3rd Edition, Jonathan Pevsner, Fig. 3-18, Page 94

1.2 Pairwise Alignment

HOW Can we Align Two Sequences?



Different types of pairwise comparisons

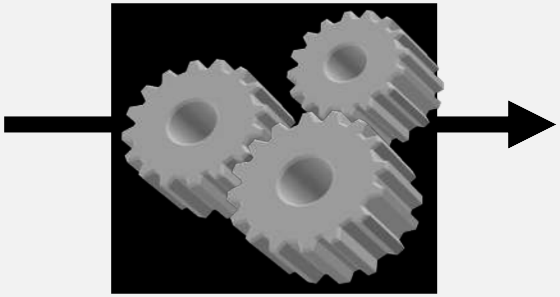
Method name	*Situation*
Dot-plot	**General exploration of your sequence** Discovering repeats Finding long insertion/deletions Extracting portions of sequences to make a multiple alignment
Local alignments	**Comparing sequences with partial homology** Making high quality alignments Making residue-per-residue analysis
Global alignments	**Comparing two sequences over their entire length** Identifying long insertion/deletions Checking the quality of your data Identifying every mutation in your sequences

Adapted from Cedric Notredame

Global Alignments

- Take 2 Nice Protein Sequences
- A good Substitution Matrix (Blosum62)
- DYNAMIC PROGRAMMING

>Seq1
THEFATCAT
>Seq2
THEFASTCAT

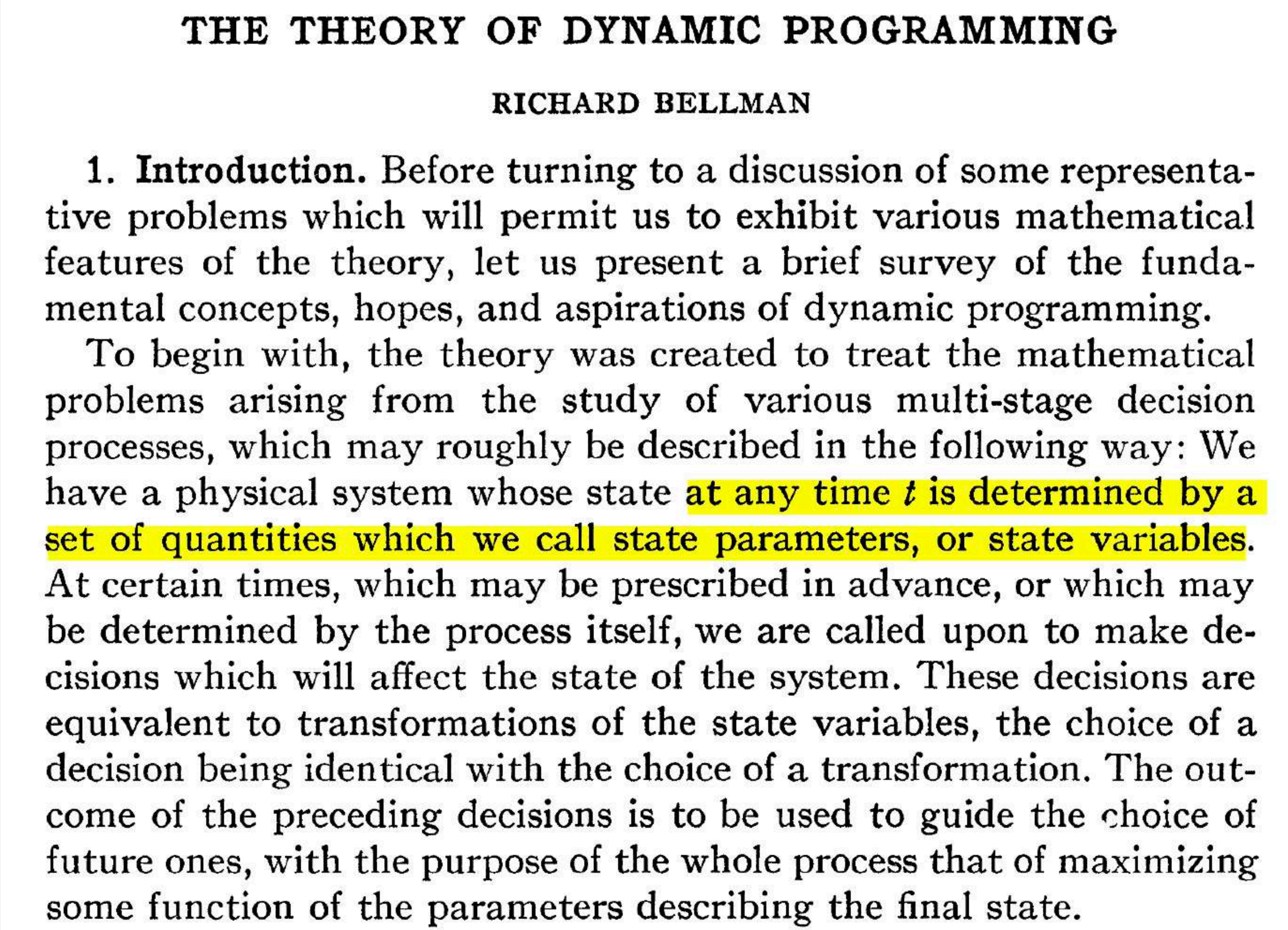


DYNAMIC
PROGRAMMING

```
THEFA-TCAT
THEFASTCAT
```

Adapted from Cedric Notredame

Dynamic Programming



THE THEORY OF DYNAMIC PROGRAMMING

RICHARD BELLMAN

1. Introduction. Before turning to a discussion of some representative problems which will permit us to exhibit various mathematical features of the theory, let us present a brief survey of the fundamental concepts, hopes, and aspirations of dynamic programming.

To begin with, the theory was created to treat the mathematical problems arising from the study of various multi-stage decision processes, which may roughly be described in the following way: We have a physical system whose state at any time t is determined by a set of quantities which we call state parameters, or state variables. At certain times, which may be prescribed in advance, or which may be determined by the process itself, we are called upon to make decisions which will affect the state of the system. These decisions are equivalent to transformations of the state variables, the choice of a decision being identical with the choice of a transformation. The outcome of the preceding decisions is to be used to guide the choice of future ones, with the purpose of the whole process that of maximizing some function of the parameters describing the final state.

Bellman, R. The theory of dynamic programming. *Bulletin of the American Mathematical Society* **60,** 503–515 (1954).

Using Dynamic Programming To Align Sequences

- DP invented in the 1950s by Bellman
    - Programming ⇔ Tabulation
- Re-invented in 1970 by Needlman and Wunsch
    - It took 10 year to find out…
    - *Needleman, Saul B. & Wunsch, Christian D. (1970). "A general method applicable to the search for similarities in the amino acid sequence of two proteins". Journal of Molecular Biology.* ***48*** *(3): 443–53*

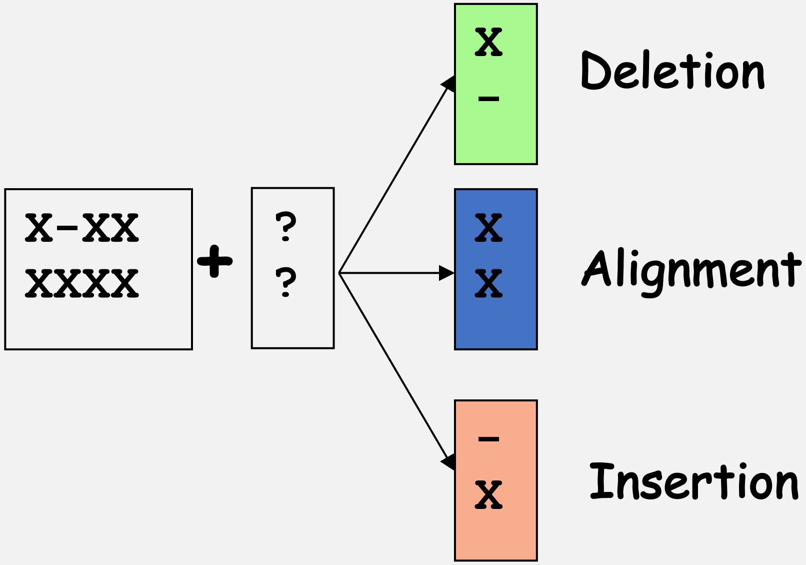
Adapted from Cedric Notredame

Global Alignment

Needleman, Saul B. & Wunsch, Christian D. (1970). "A general method applicable to the search for similarities in the amino acid sequence of two proteins". *Journal of Molecular Biology*. **48** (3): 443–53

The Principal of DP

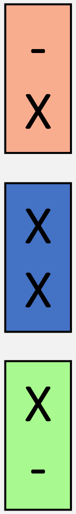
- If you extend optimally an optimal alignment of two sub-sequences, the result remains an optimal alignment



Adapted from Cedric Notredame

Finding the score of *i,j*

- Sequence 1: [1-*i*]
- Sequence 2: [1-*j*]
- The optimal alignment of [1-*i*] vs [1-*j*] can finish in three different manners:



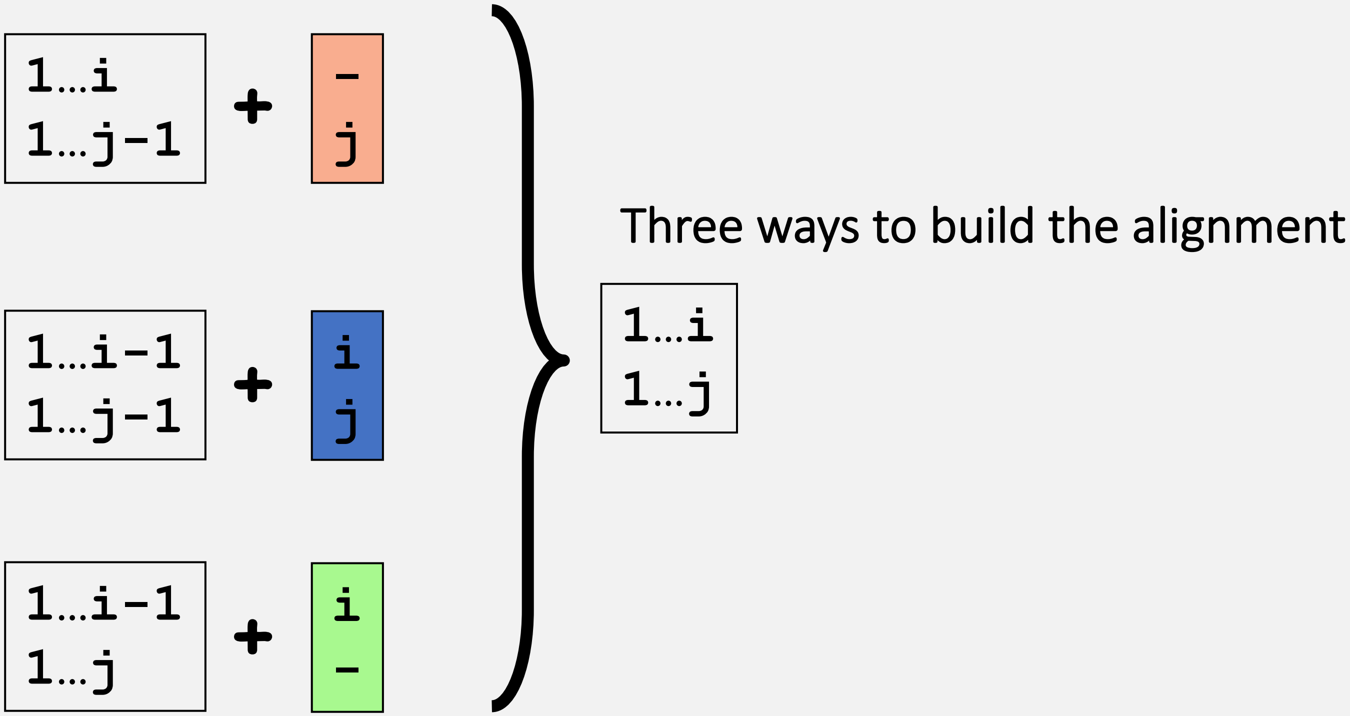
-
X

X
X

X
-

Adapted from Cedric Notredame

Finding the score of *i,j*



1…i
1…j-1
+
-
j

1…i-1
1…j-1
+
i
j

1…i-1
1…j
+
i
-

Three ways to build the alignment

1…i
1…j

Adapted from Cedric Notredame

Formalizing the algorithm

score_m(i,j)= best

- score_m(i,j-1) + gap_s
  1…i
  1…j-1
  +
  -
  x
- score_m(i-1,j-1) + match_s/mismatch_s
  1…i-1
  1…j-1
  +
  x
  x
- score_m(i-1,j) + gap_s
  1…i-1
  1…j
  +
  x
  -

Adapted from Cedric Notredame

Arranging Everything in a Table

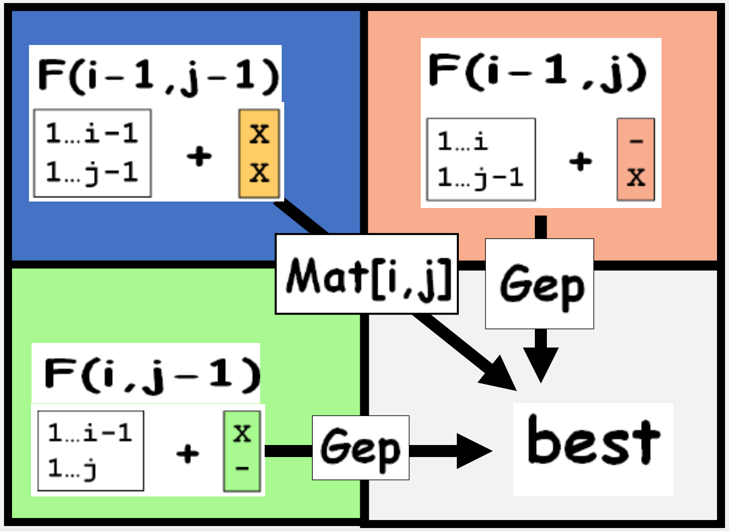
	–	F	A	T
–				
F		1…I-1 1…J-1	1…I 1…J-1	
A		1…I-1 1…J	1…I 1…J	
S				
T				

Adapted from Cedric Notredame

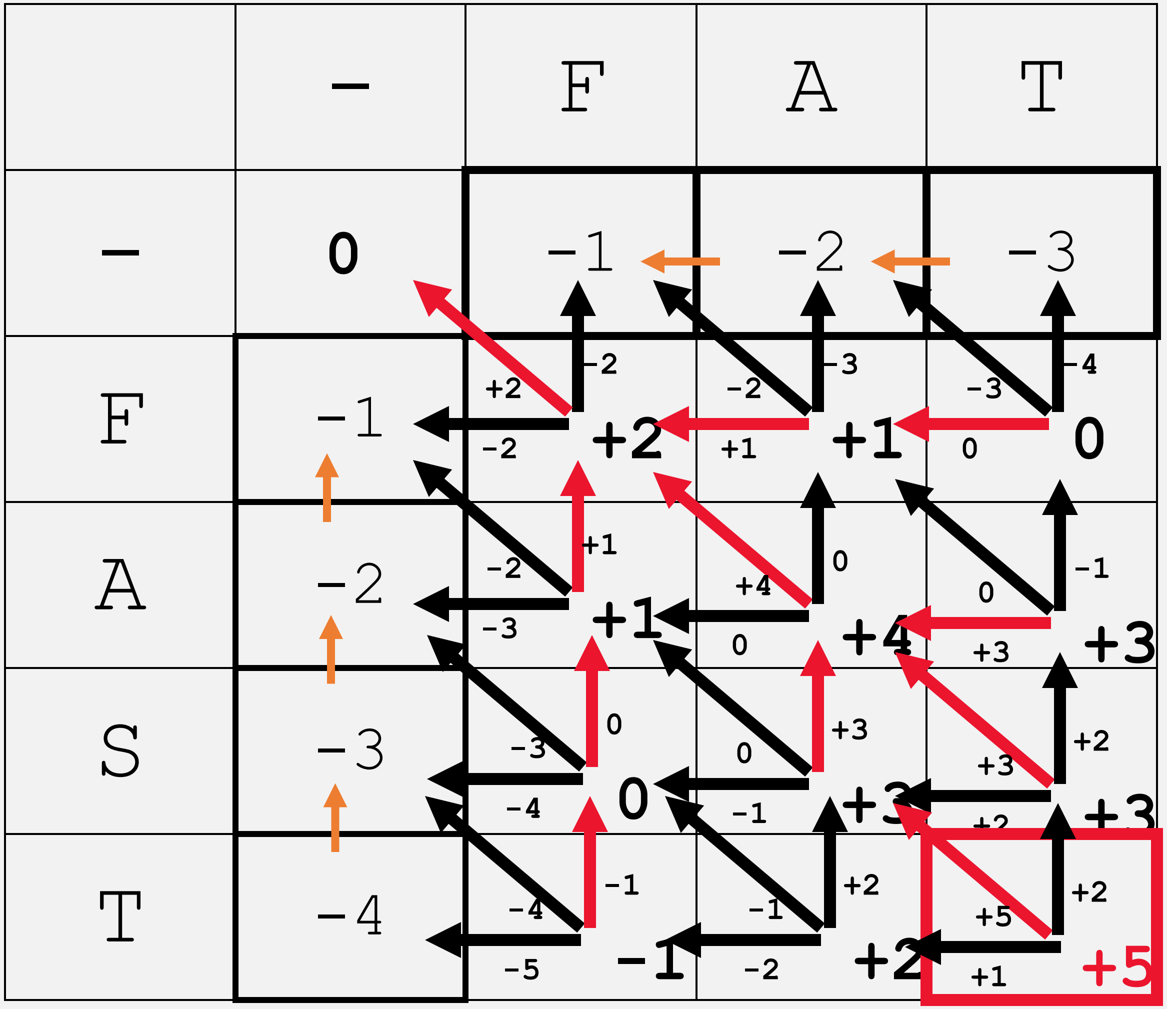
Filing Up The Matrix

$$\text{score\_m(i,j)= best}\begin{cases}\text{score\_m(i,j-1) + gap\_s} \\ \text{score\_m(i-1,j-1) +} \\ \text{match\_s/mismatch\_s} \\ \text{score\_m(i-1,j) + gap\_s}\end{cases}$$

1…i 1…j-1	+	- x
1…i-1 1…j-1	+	x x
1…i-1 1…j	+	x -



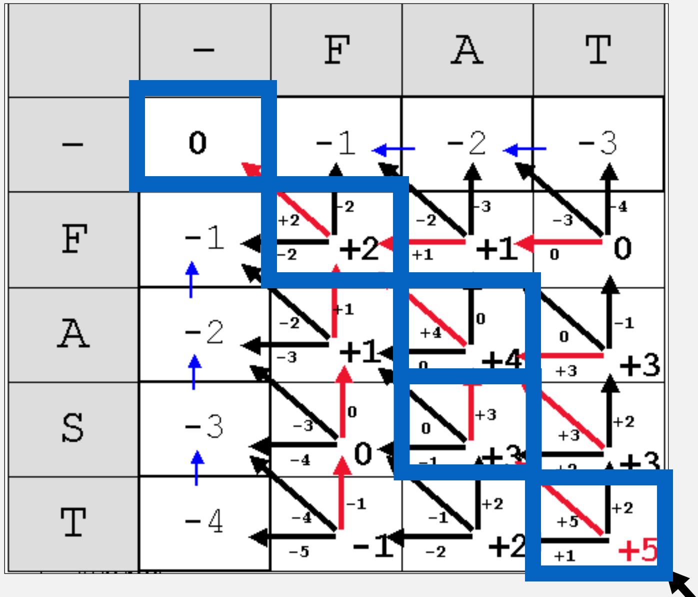
Adapted from Cedric Notredame



	–	F	A	T
–	0	–1	–2	–3
F	–1	+2	+1	0
A	–2	+1	+4	+3
S	–3	0	+3	+3
T	–4	–1	+2	+5

Adapted from Cedric Notredame

Delivering the alignment: Trace-back



	–	F	A	T
–	0	-1	-2	-3
F	-1	+2	+1	0
A	-2	+1	+4	+3
S	-3	0	+3	+3
T	-4	-1	+2	+5

T	S	A	F
T	–	A	F

Score of 1…3 Vs 1…4

⇔

Optimal Aln Score

Adapted from Cedric Notredame

Trace-back: possible implementation

```
while (!(i==0 && j==0)):
        if (direc_m[i][j]== ‘sub’):     #SUBSTITUTION
                aln1[aln_len]=pro1Seq[--i]

        aln2[aln_len]=pro2Seq[--j]
        elif (direc_m[i][j]==‘del’):      #DELETION
                aln1[aln_len]='-'
                aln2[aln_len]=pro2Seq[--j]
        elif (direc_m[i][j]==‘ins’):      #INSERTION
                aln1[aln_len]=pro1Seq[0][--i]
                aln2[aln_len]='-'
        aln_len++
}
```

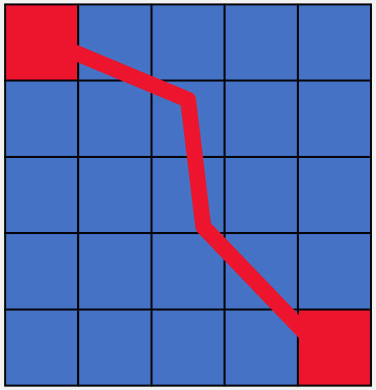
Adapted from Cedric Notredame

Local Alignment
Smith & Waterman algorithm

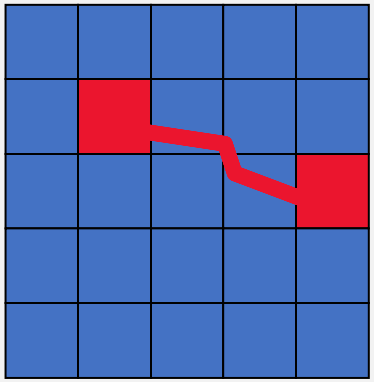
Smith, T. F. & Waterman, M. S. Identification of common molecular subsequences. *J. Mol. Biol.* **147,** 195–7 (1981).

Global alignment VS local alignment

- Global : extends from one end of each sequence to the other.
- Local : finds optimally matching regions within two sequences,
  - Subsequences useful to find domains (or limited regions of homology) within sequences
  - Smith and Waterman (1981) solved the problem of performing optimal local sequence alignment.
  - Other methods (BLAST, FASTA) are faster but less thorough.



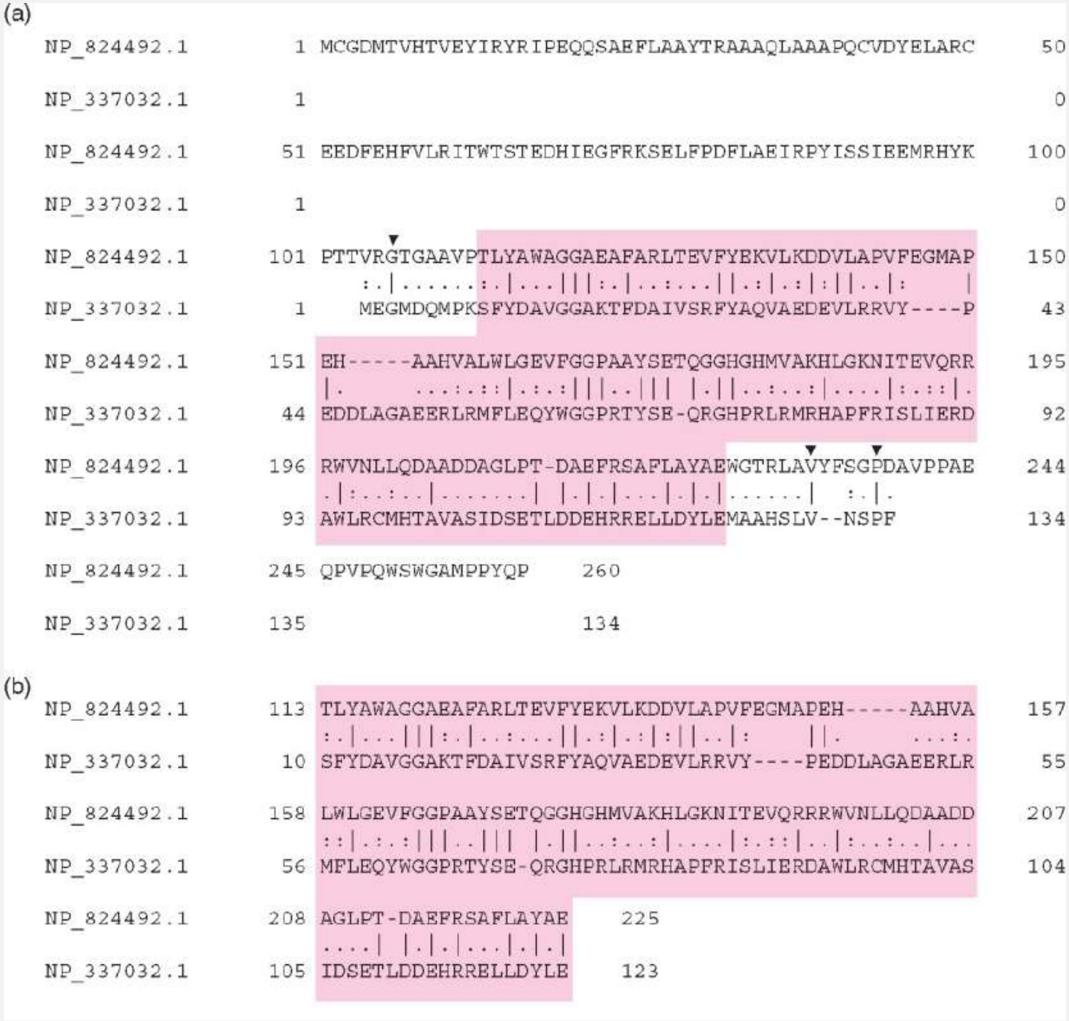
GLOBAL Alignment



LOCAL Alignment

Adapted from Cedric Notredame

Global alignment (top) includes matches ignored by local alignment (bottom)



(a)

```
NP_824492.1       1 MCGDMTVHTVEYIRYRIPEQQSAEFLAAYTRAAAQLAAAPQCVDYELARC   50

NP_337032.1       1                                                       0

NP_824492.1      51 EEDFEHFVLRITWTSTEDHIEGFRKSELFPDFLAEIRPYISSIEEMRHYK  100

NP_337032.1       1                                                       0
                         ▼
NP_824492.1     101 PTTVRGTGAAVPTLYAWAGGAEAFARLTEVFYEKVLKDDVLAPVFEGMAP  150
                       :.|......:.|...|||:.|..:...||.:|.:|:||..|:    |
NP_337032.1       1    MEGMDQMPKSFYDAVGGAKTFDAIVSRFYAQVAEDEVLRRVY----P   43

NP_824492.1     151 EH-----AAHVALWLGEVFGGPAAYSETQGGHGHMVAKHLGKNITEVQRR  195
                    |.     ...:.::|.:.:|||..||| |.||..:..:|....|:.::|.
NP_337032.1      44 EDDLAGAEERLRMFLEQYWGGPRTYSE-QRGHPRLRMRHAPFRISLIERD   92
                                                         ▼    ▼
NP_824492.1     196 RWVNLLQDAADDAGLPT-DAEFRSAFLAYAEWGTRLAVYFSGPDAVPPAE  244
                    .|:..:..|.......| |.|.|...|.|.|......|  :.|.
NP_337032.1      93 AWLRCMHTAVASIDSETLDDEHRRELLDYLEMAAHSLV--NSPF        134

NP_824492.1     245 QPVPQWSWGAMPPYQP      260

NP_337032.1     135                       134
```

(b)

```
NP_824492.1     113 TLYAWAGGAEAFARLTEVFYEKVLKDDVLAPVFEGMAPEH-----AAHVA  157
                    :.|...|||:.|..:...||.:|.:|:||..|:    ||.     ...:.
NP_337032.1      10 SFYDAVGGAKTFDAIVSRFYAQVAEDEVLRRVY----PEDDLAGAEERLR   55

NP_824492.1     158 LWLGEVFGGPAAYSETQGGHGHMVAKHLGKNITEVQRRRWVNLLQDAADD  207
                    ::|.:.:|||..||| |.||..:..:|....|:.::|..|:..:..|...
NP_337032.1      56 MFLEQYWGGPRTYSE-QRGHPRLRMRHAPFRISLIERDAWLRCMHTAVAS  104

NP_824492.1     208 AGLPT-DAEFRSAFLAYAE      225
                    ....| |.|.|...|.|.|
NP_337032.1     105 IDSETLDDEHRRELLDYLE      123
```

Global:
15% identity

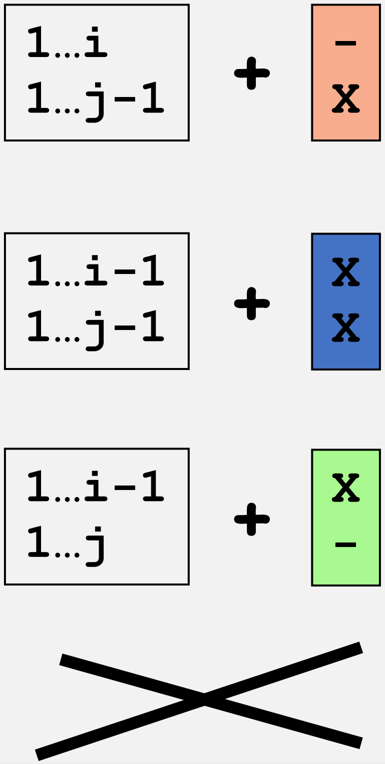
Local:
30% identity

B&FG 3e, Fig. 3-23, Page 101

The Smith and Waterman Algorithm

- 0 => Ignore the rest of the Matrix => terminate a local alignment

$$F(i,j)= \text{best} \begin{cases} F(i-1,j) + Gep \\ F(i-1,j-1) + Mat[i,j] \\ F(i,j-1) + Gep \\ 0 \end{cases}$$



1…i
1…j-1
+
-
x

1…i-1
1…j-1
+
x
x

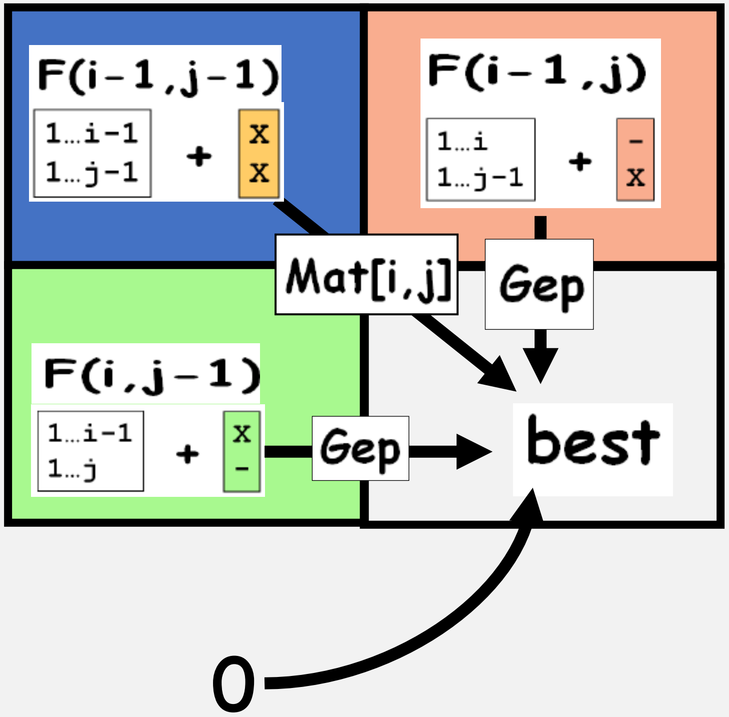
1…i-1
1…j
+
x
-

Adapted from Cedric Notredame

Filing Up a SW Matrix

F(i,j)= best

- F(i-1,j) + Gep 1…i / 1…j-1 + – / x
- F(i-1,j-1) + Mat[i,j] 1…i-1 / 1…j-1 + x / x
- F(i,j-1) + Gep 1…i-1 / 1…j + x / –
- 0



Adapted from Cedric Notredame

Filling up a SW matrix: borders

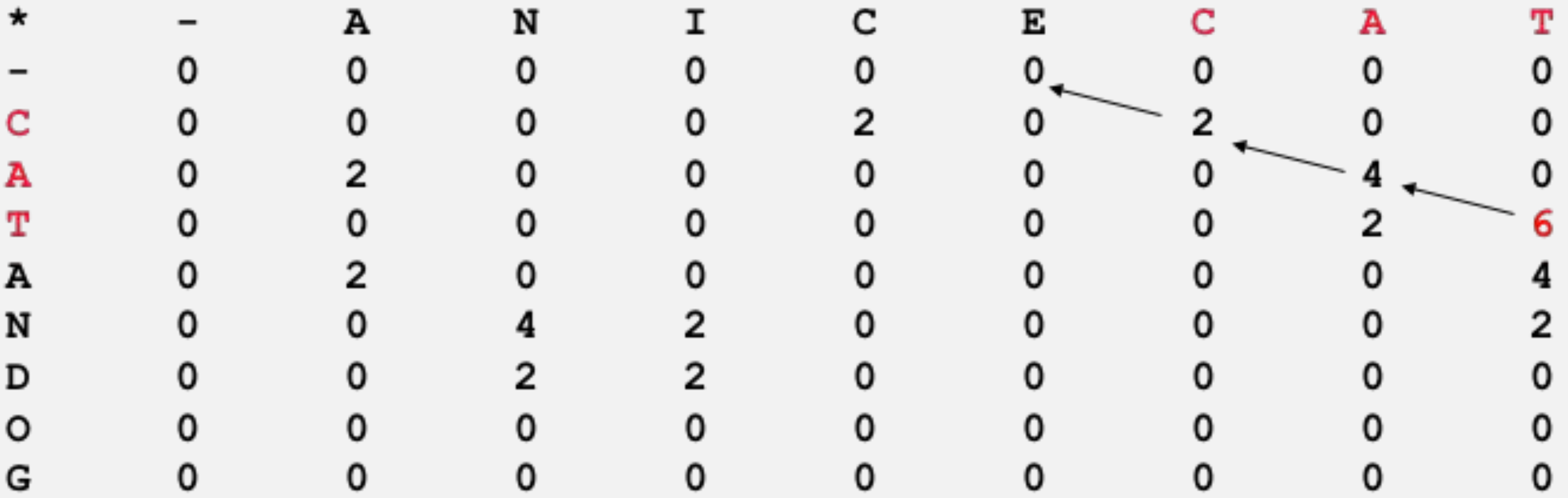
*	-	A	N	I	C	E	C	A	T
-	0	0	0	0	0	0	0	0	0
C	0								
A	0								
T	0								
A	0								
N	0								
D	0								
O	0								
G	0								

Local alignments NEVER start/end with a gap…

Adapted from Cedric Notredame

Filling up a SW matrix

- Best Local score ⇔ Beginning of the trace-back



*	-	A	N	I	C	E	C	A	T
-	0	0	0	0	0	0	0	0	0
C	0	0	0	0	2	0	2	0	0
A	0	2	0	0	0	0	0	4	0
T	0	0	0	0	0	0	0	2	6
A	0	2	0	0	0	0	0	0	4
N	0	0	4	2	0	0	0	0	2
D	0	0	2	2	0	0	0	0	0
O	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0

Adapted from Cedric Notredame

Adding Affine Gap Penalties
Forcing a bit of Biology into your alignment

Gotoh, O. An improved algorithm for matching biological sequences. *J. Mol. Biol.* **162,** 705–8 (1982).

Adapted from Cedric Notredame

Gap Penalties: Opening & extension

- Gaps : Positions at which a letter is paired with a null are called.
- Gap scores are typically negative.
- Opening a gap is more expensive than extending it
  - Since a single mutational event may cause the insertion or deletion of more than one residue, the presence of a gap is ascribed more significance than the length of the gap.
- Thus there are separate penalties for <u>gap open</u> and <u>gap extension</u>.

Gap Opening Penalty

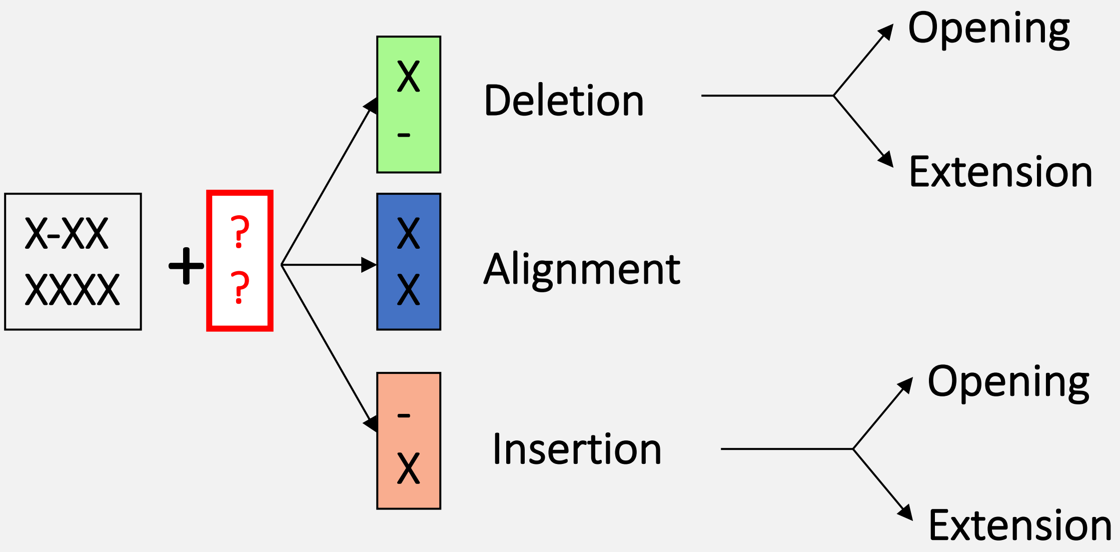
Gap Extension Penalty

```
Seq A GARFIELDTHE----CAT
      |||||||||||    |||
Seq B GARFIELDTHELASTCAT
```

Adapted from Cedric Notredame

But Harder To compute…

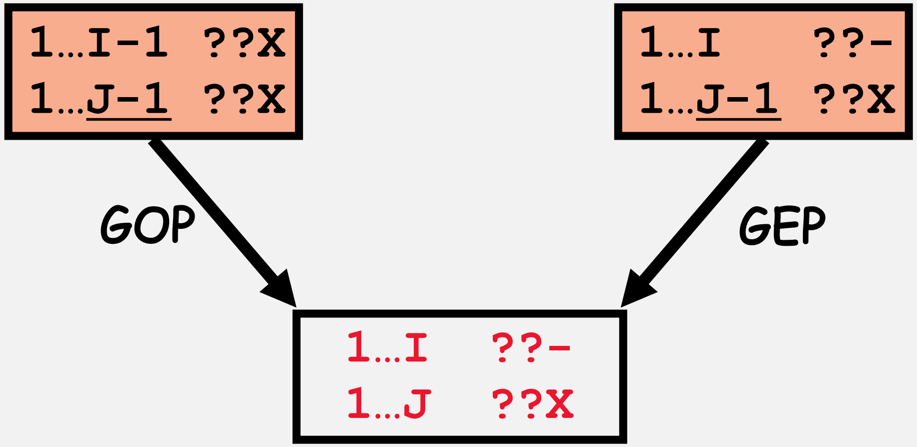
- More Than 3 Ways to extend an Alignment



Adapted from Cedric Notredame

More Questions Need to be asked

- For instance, what is the cost of an insertion ?



Adapted from Cedric Notredame

Solution: Maintain 3 Tables

$$M(i,j) = \text{best} \begin{cases} M(i-1,j-1) + Mat(i,j) \\ Ix(i-1,j-1) + Mat(i,j) \\ Iy(i-1,j-1) + Mat(i,j) \end{cases}$$

1…i-1
1…j-1

X
X

$$Ix(i,j) = \text{best} \begin{cases} M(i-1,j) + gop \\ Ix(i-1,j) + gep \end{cases}$$

1…i-1 X
1…j X

X
-

1…i-1 X
1…j -

X
-

$$Iy(i,j) = \text{best} \begin{cases} M(i,j-1) + gop \\ Iy(i,j-1) + gep \end{cases}$$

1…i X
1…j-1 X

-
X

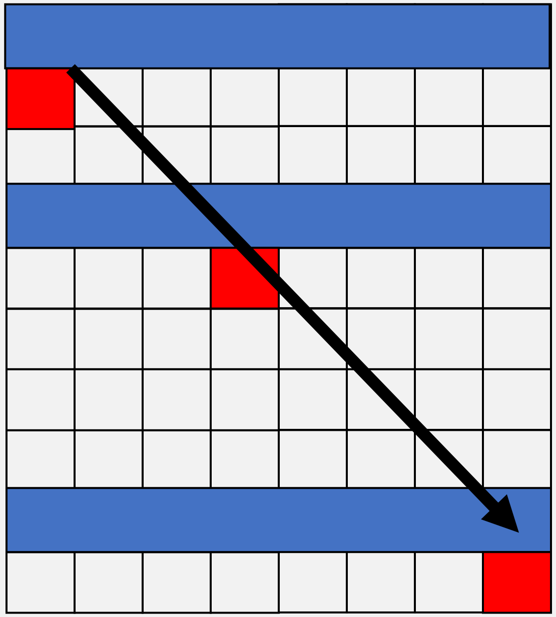
1…i -
1…j-1 X

-
X

Adapted from Cedric Notredame

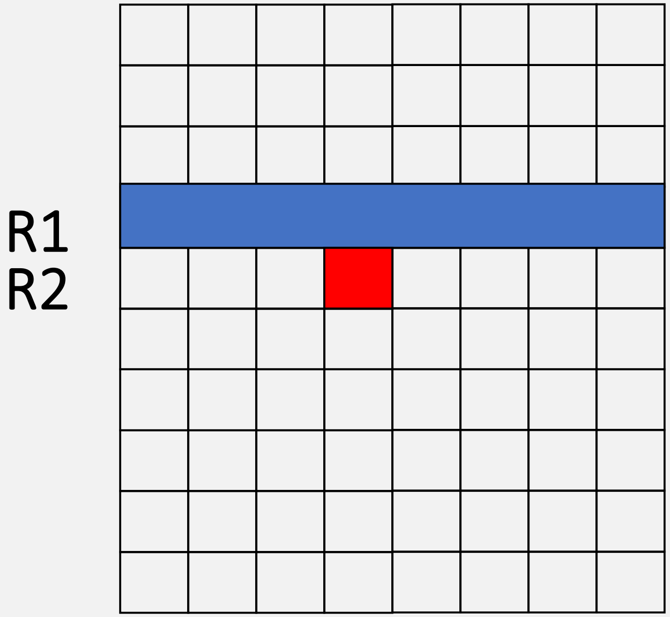
A Score in Linear Space

- You never Need More Than The Previous Row To Compute the optimal score



Adapted from Cedric Notredame

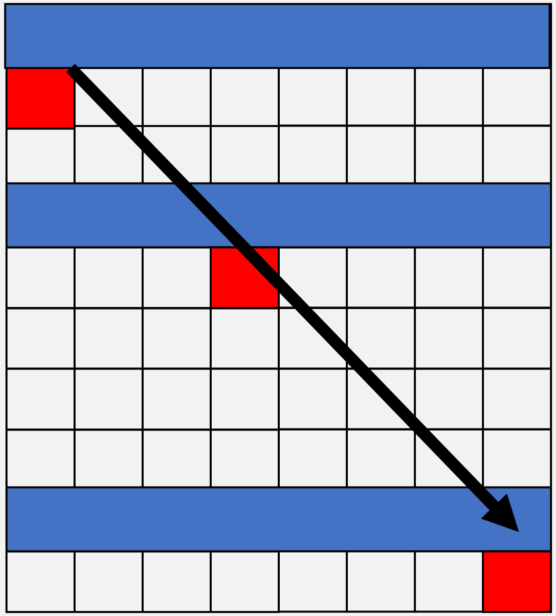
A Score in Linear Space



```
for i=1:I
   for j=1:J
      R2[i][j]=best
            R2[j-1], +gep
            R1[j-1]+mat
            R1[j]+gep
   for J,
         R1[j]=R2[j]
```

Adapted from Cedric Notredame

A Score in Linear Space



You never Need More Than The Previous Row To Compute the optimal score

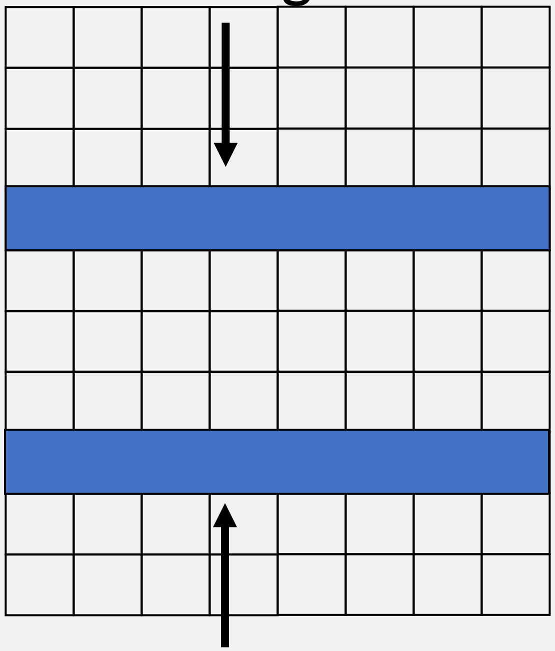
You only need the matrix for the Trace-Back,

Or do you ????

Adapted from Cedric Notredame

An Alignment in Linear Space

Forward Algorithm



Backward algorithm

F(i,j)=Optimal score of
0…i Vs 0…j

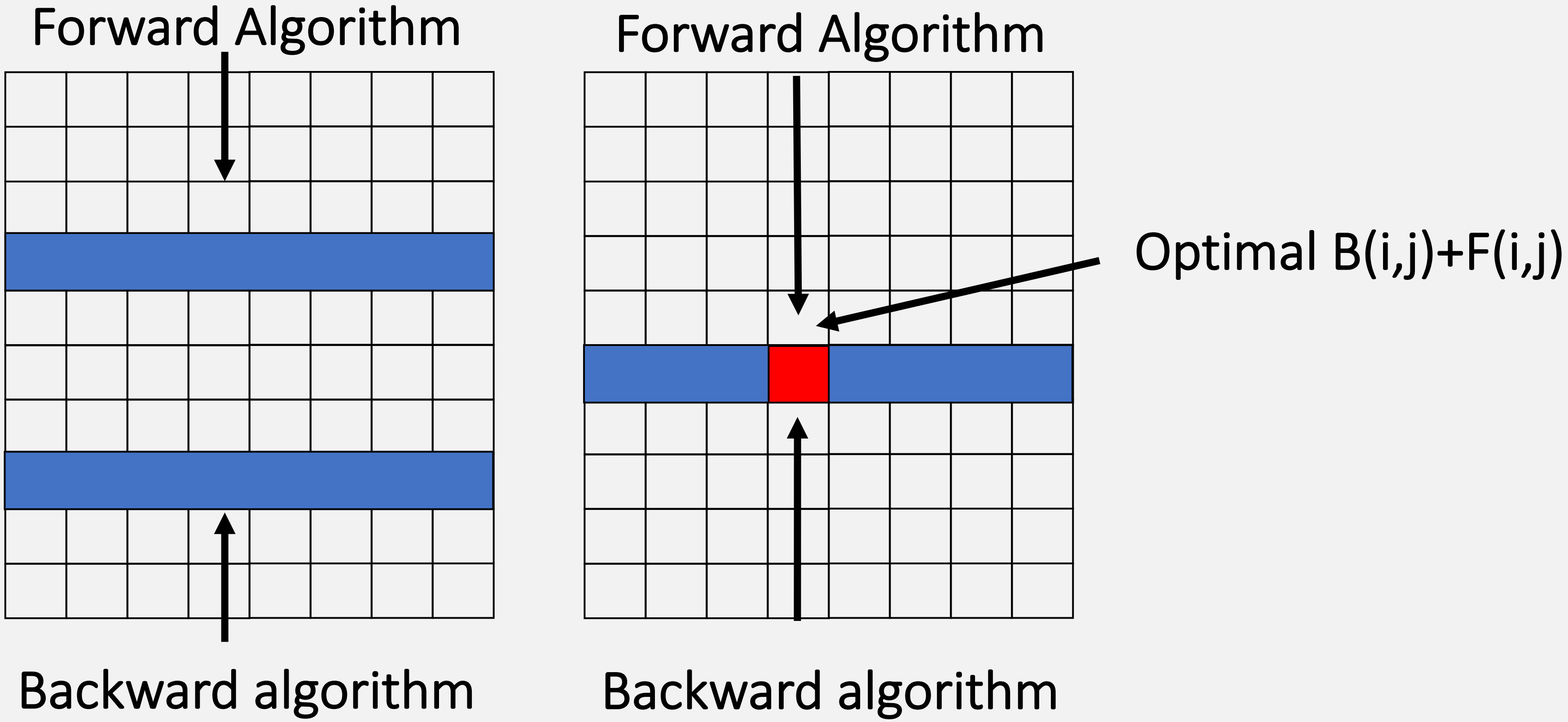
B(i,j)=Optimal score of
M…i Vs N…j

B(i,j)+F(i,j)=
Optimal score of the alignment that passes through pair i,j

Myers, E. W. & Miller, W. Optimal alignments in linear space.*Comput. Appl. Biosci.* **4,** 11–7 (1988).

Adapted from Cedric Notredame

An Alignment in Linear Space



Forward Algorithm

Backward algorithm

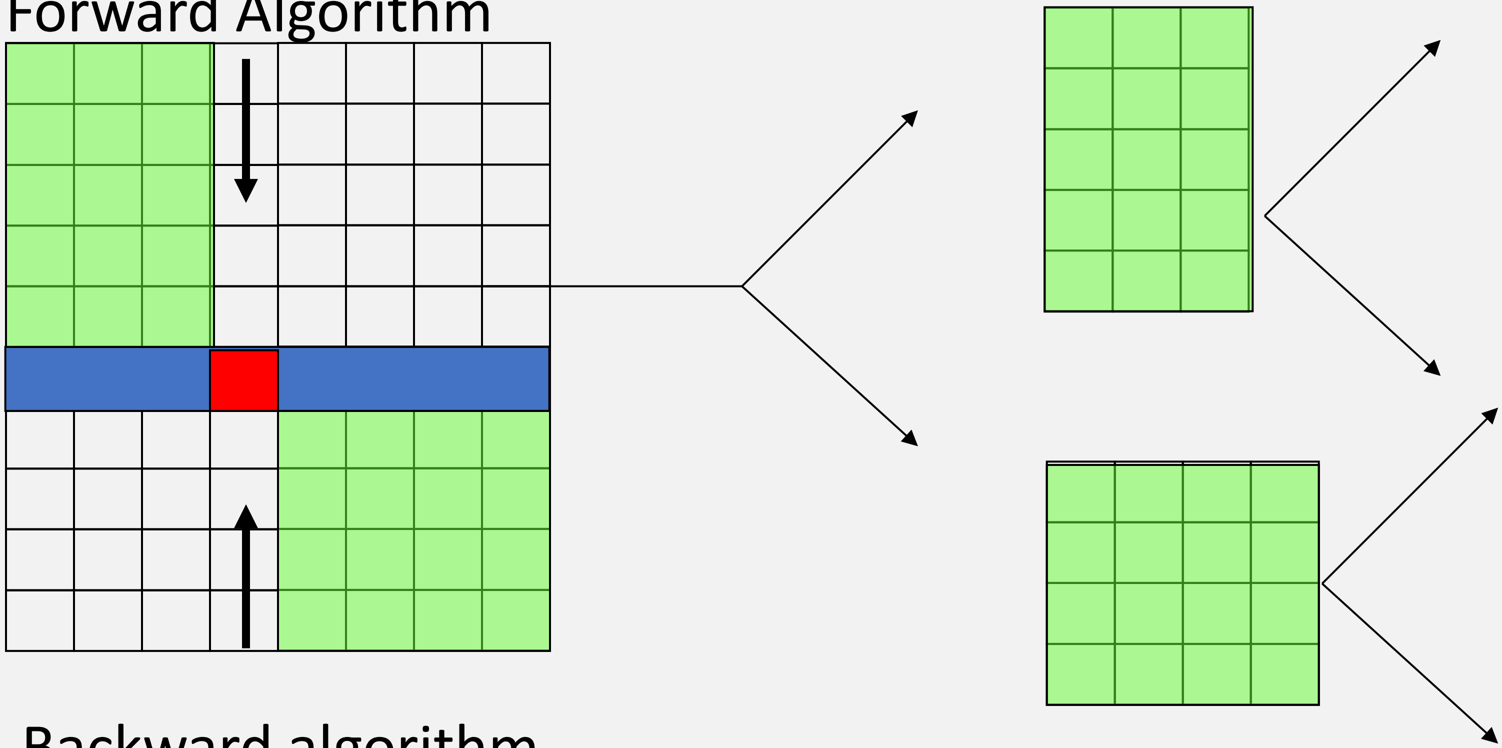
Forward Algorithm

Optimal B(i,j)+F(i,j)

Backward algorithm

Adapted from Cedric Notredame

An Alignment in Linear Space



Recursive divide and conquer strategy:
Myers and Miller (Durbin p35)

Adapted from Cedric Notredame

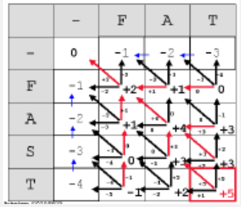
Remember Not To Run Out of Memory

- A survey paper
  - Chao, K.-M., Hardison R. C. and Miller, W., 1994, Recent Developments in Linear-Space Alignment Methods: a Survey, *Journal of Computational Biology*, 1: 271-291.

趙坤茂 (Kun-Mao Chao)
台大資工系

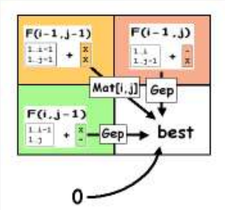
Recap: Pairwise alignment

- Needleman and Wunsch: Delivers the best scoring global alignment

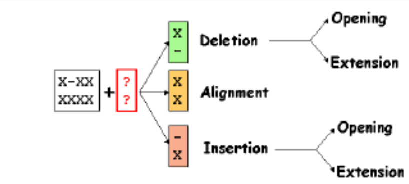


	-	F	A	T
-	0	-1	-2	-3
F	-1	+2	+1	0
A	-2	+1	+4	+3
S	-3	0	+3	+3
T	-4	-1	+2	+5

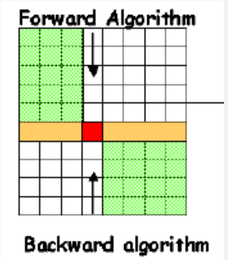
- Smith and Waterman: NW with an extra state 0



- Affine Gap Penalties: Making DP more realistic



- Linear space: Using Divide and Conquer Strategies Not to run out of memory



Adapted from Cedric Notredame

1.3 Multiple Sequence Alignment

Sometimes two sequences are not enough…

- The man with TWO watches NEVER knows the time

Adapted from Cedric Notredame

The COMPUTATIONAL Problem

- A nice set of Sequences
- Substitution Matrix (Blosum)
- Gap Penalties
- An Evaluation/Scoring Function
- An Alignment Algorithm

Adapted from Cedric Notredame

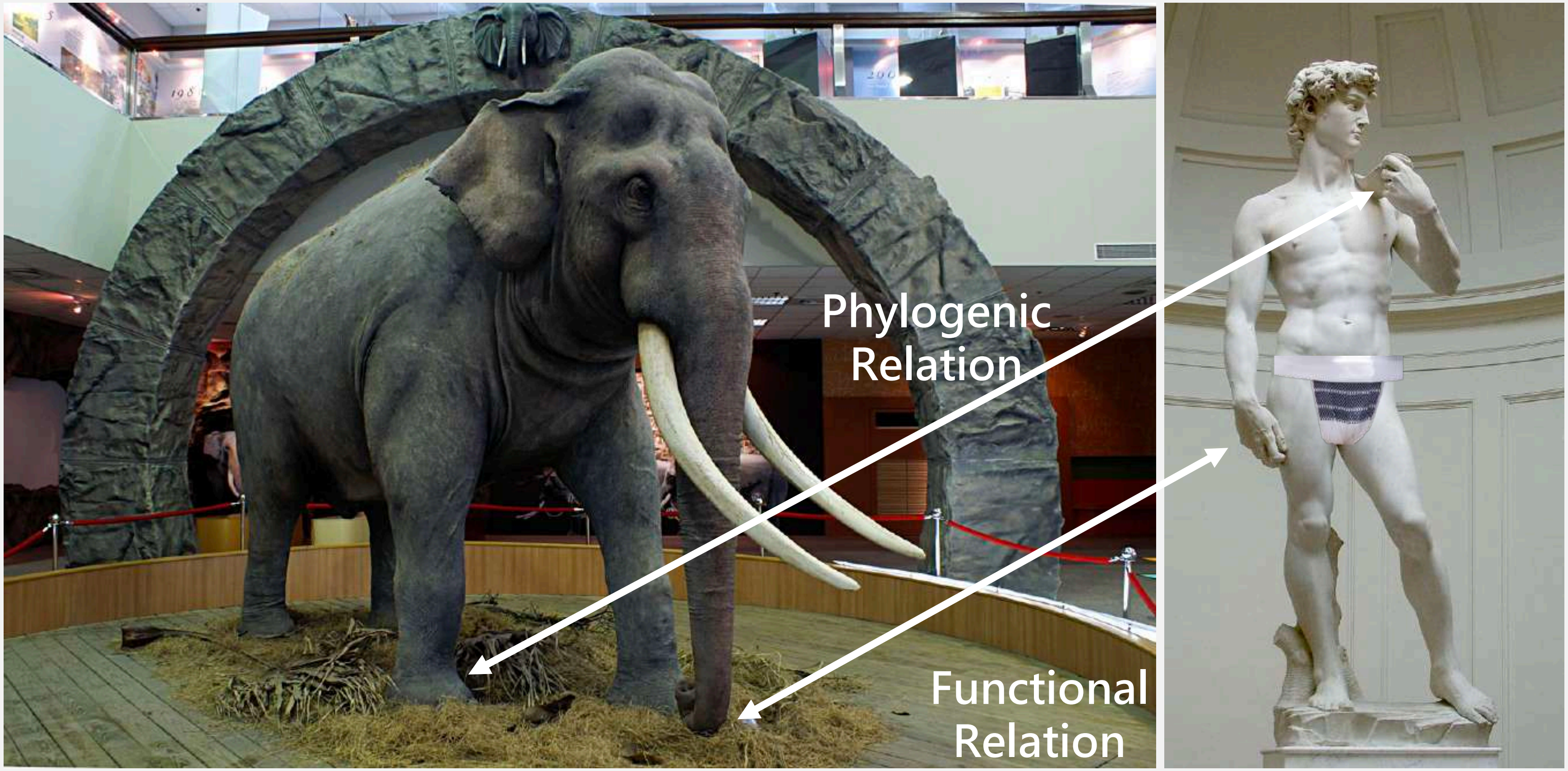
What is A Multiple Sequence Alignment?

- Structural Criteria
  - Residues are arranged so that those playing a similar role end up in the same column.
- Evolution Criteria
  - Residues are arranged so that those having the same ancestor end up in the same column.

```
chite   ---ADKPKRPLSAYMLWLNSARESIKRENPDFK-VTEVAKKGGELWRGLKD
wheat   --DPNKPKRAPSAFFVFMGEFREEFKQKNPKNKSVAAVGKAAGERWKSLSE
trybr   KKDSNAPKRAMTSFMFFSSDFRS----KHSDLS-IVEMSKAAGAAWKELGP
mouse   -----KPKRPRSAYNIYVSESFQ----EAKDDS-AQGKLKLVNEAWKNLSP
              ***. ::: .: ..  .    :  . .      *  .  *: *

chite   AATAKQNYIRALQEYERNGG-
wheat   ANKLKGEYNKAIAAYNKGESA
trybr   AEKDKERYKREM---------
mouse   AKDDRIRYDNEMKSWEEQMAE
        *   : .* . :
```

Adapted from Cedric Notredame



Phylogenic Relation

Functional Relation

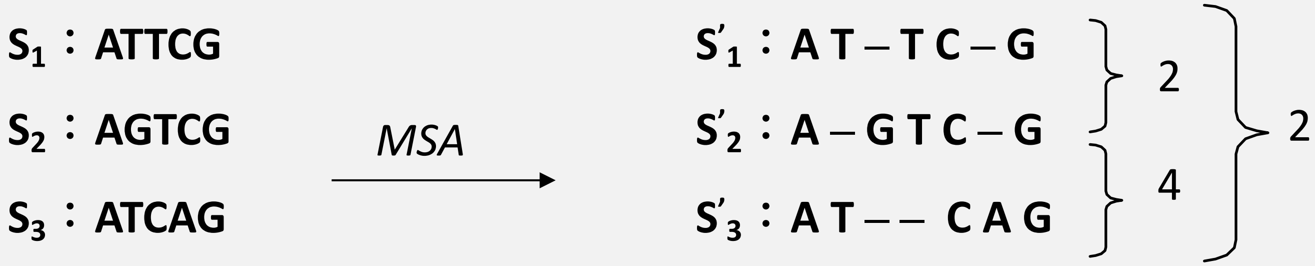
By peellden - 自己的作品

By Rico Heil (User:Silmaril) - private photo

Adapted from Cedric Notredame

Scoring function

- Sum of Pair (SP)
- Tree Cost: MSA with tree cost will be called tree alignment.
- Circular Sum(CS)



S_1 : ATTCG

S_2 : AGTCG

S_3 : ATCAG

MSA

S'_1 : A T – T C – G

S'_2 : A – G T C – G

S'_3 : A T – – C A G

Cost = 8

MSA with SP-Score: Exact Algorithm

- Given
  - k : # of Sequences
  - n : Sequences of length
- Exactly by Dynamic Programming
  - $O(2n^k)$ ： D.Snakoff, Simultaneous solution of RNA folding, alignment and Protosequence prolblems, *SIAM J. Appl. Math.*,(1985)
  - Exact methods of multiple alignment use dynamic programming and are guaranteed to find optimal solutions. But they are not feasible for more than a few sequences.

MSA with SP-Score: Complexity

- Wang L. Jiang T. On the complexity of multiple sequence alignment, *J Comput Biol* 1994 Winter;1(4):337-48
  - multiple alignment with SP-Score => NP-complete reduction from shortest common supersequence (non-metric : not symmetry)

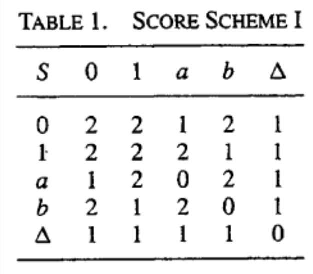


TABLE 1. SCORE SCHEME I

S	0	1	a	b	Δ
0	2	2	1	2	1
1	2	2	2	1	1
a	1	2	0	2	1
b	2	1	2	0	1
Δ	1	1	1	1	0

  - multiple tree alignment => MAX SNP-hard
- Paola Bonizzoni, Gianluca Della Vedoa The complexity with Multiple sequence alignment with SP-score that is a metric, *Theoretical Computer Science*; 259 (2001) 63-79
  - multiple alignment with SP-Score => NP-complete reduction from node cover

Feng-Doolittle algorithm

D.F.Feng, R.F.Doolittle, Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J. Mol. Evol.* 25, 351-360., (1987)

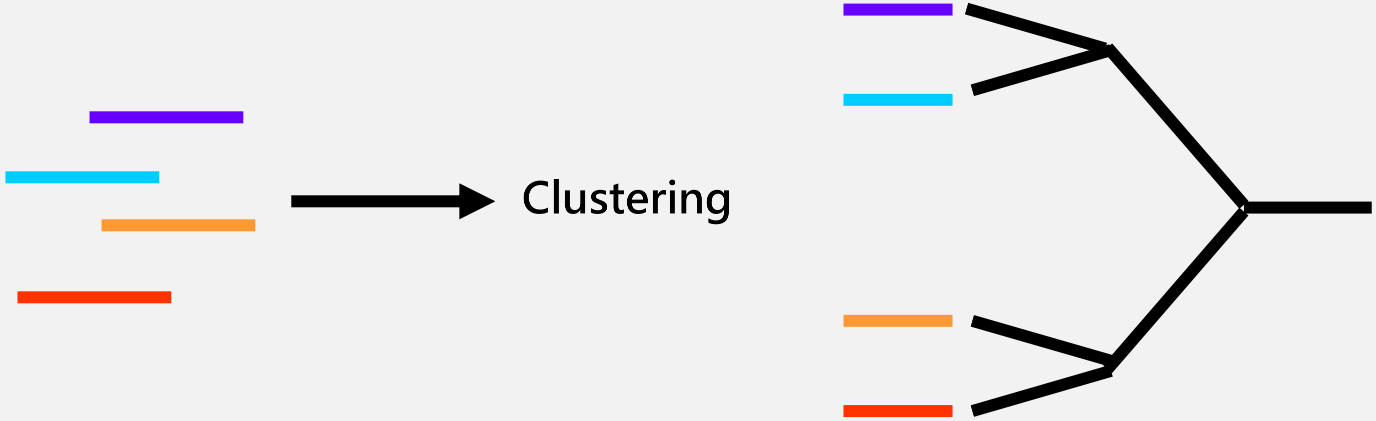
Progressive alignment

- Any exact method would be TOO SLOW.
- We will use a heuristic algorithm.
  - Progressive alignment algorithm is the most popular
    - use a guide tree (related to a phylogenetic tree) to determine how to combine pairwise alignments one by one to create a multiple alignment.
    - Examples
      - ClustalW
      - MUSCLE
  - - Greedy Heuristic (No Guaranty)
  - + Fast

Adapted from Cedric Notredame

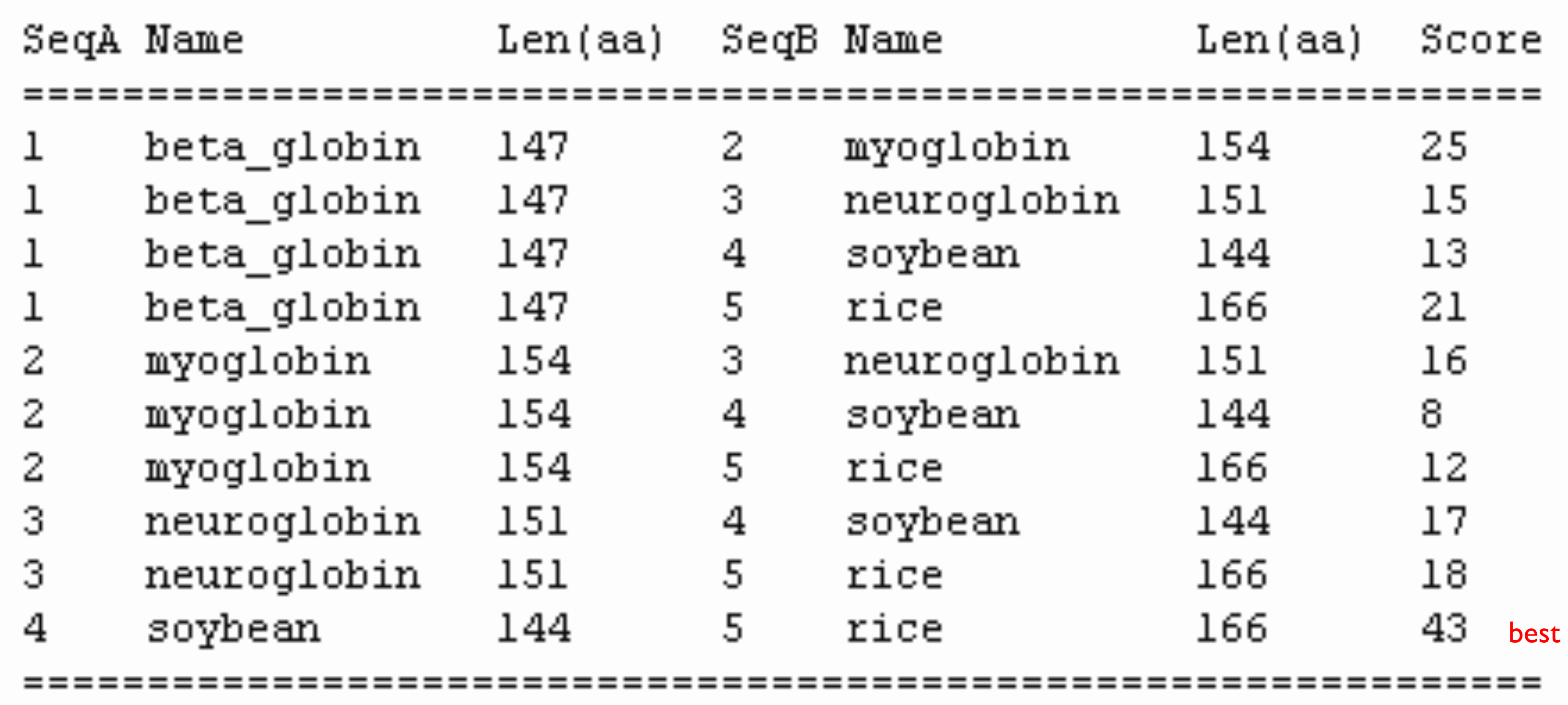
Feng-Doolittle MSA occurs in 3 stages

- Feng and Dolittle, 1988; Taylor 1989
  1. Do a set of global pairwise alignments
     - Needleman and Wunsch’ s dynamic programming algorithm
  2. Create a guide tree
  3. Progressively align the sequences



Adapted from Cedric Notredame

Generate global pairwise alignments (Progressive 1/3)

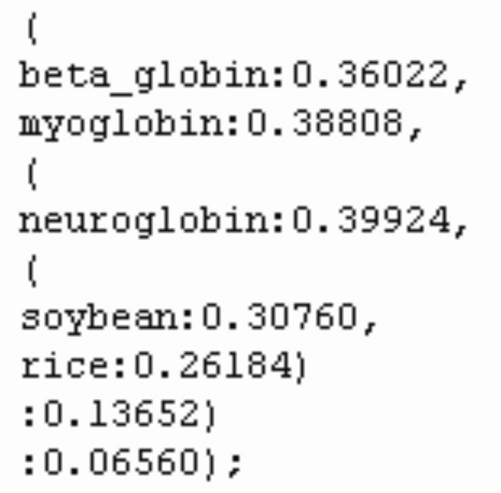


```
SeqA Name         Len(aa)  SeqB Name          Len(aa)  Score
=============================================================
1    beta_globin  147      2    myoglobin     154      25
1    beta_globin  147      3    neuroglobin   151      15
1    beta_globin  147      4    soybean       144      13
1    beta_globin  147      5    rice          166      21
2    myoglobin    154      3    neuroglobin   151      16
2    myoglobin    154      4    soybean       144      8
2    myoglobin    154      5    rice          166      12
3    neuroglobin  151      4    soybean       144      17
3    neuroglobin  151      5    rice          166      18
4    soybean      144      5    rice          166      43
=============================================================
```

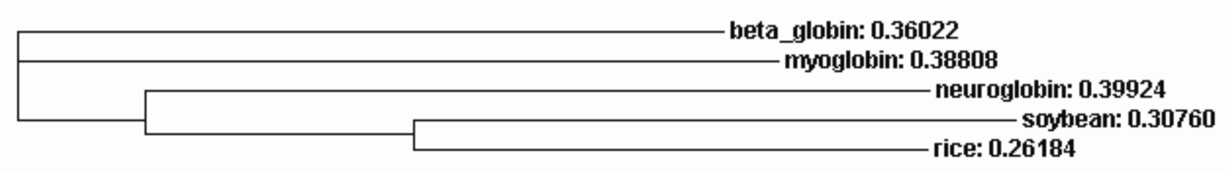
best score

Guide tree (Progressive 2/3)

- Convert similarity scores to distance scores
- A tree shows the distance between objects
- Use UPGMA (defined in the phylogeny chapter)
- ClustalW provides a syntax to describe the tree



```
(
beta_globin:0.36022,
myoglobin:0.38808,
(
neuroglobin:0.39924,
(
soybean:0.30760,
rice:0.26184)
:0.13652)
:0.06560);
```



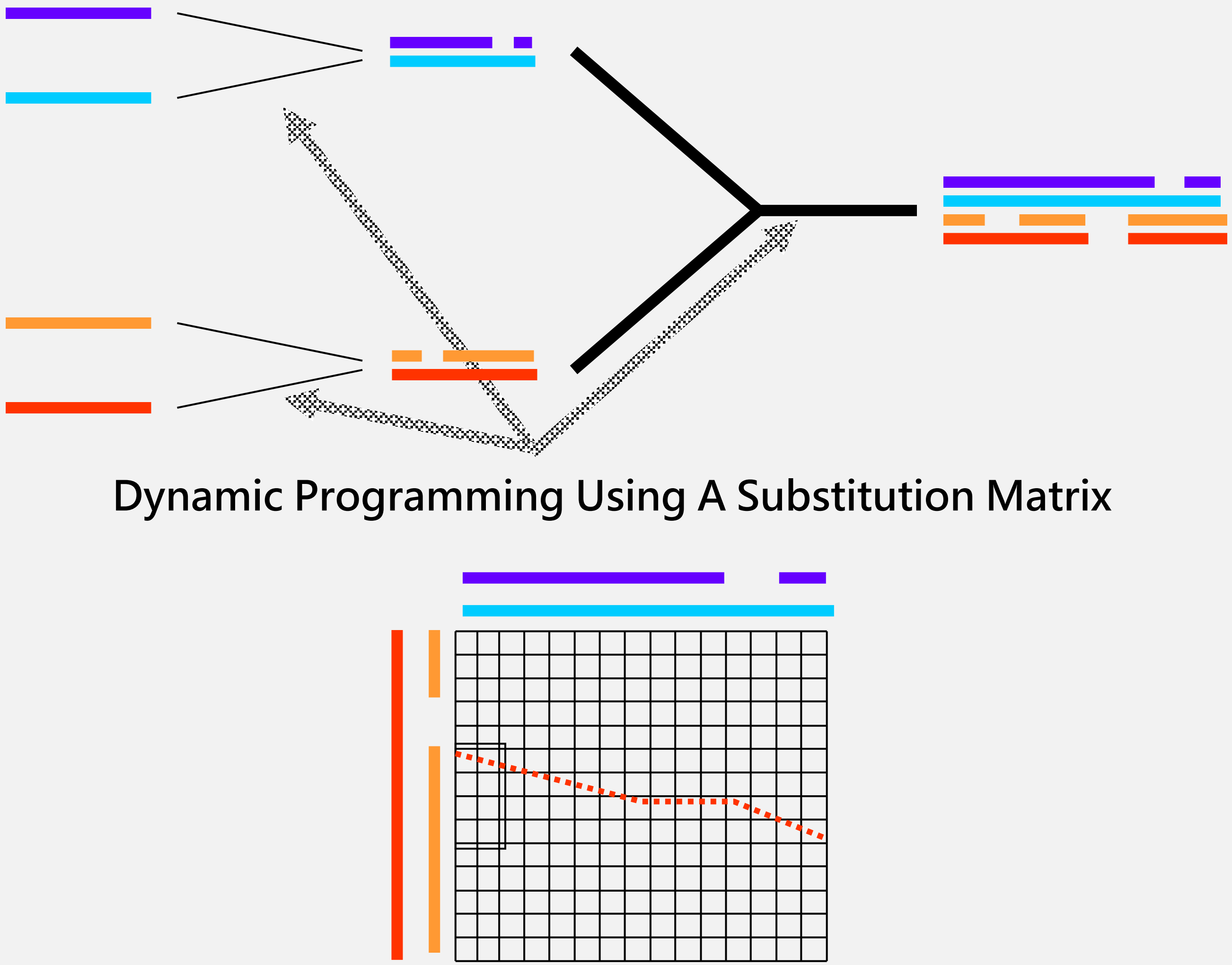
Credit by B&FG 3e, Jonathan Pevsner

Progressive alignment (Progressive 3/3)

- Make a MSA based on the order in the guide tree
- Start with the two most closely related sequences
- Then add the next closest sequence
- Continue until all sequences are added to the MSA
- Rule: *once a gap, always a gap,* why?
  - Gaps are often added to the first two (closest) sequences
  - To change the initial gap choices later on would be to give more weight to distantly related sequences
  - To maintain the initial gap choices is to trust that those gaps are most believable

Credit by B&FG 3e, Jonathan Pevsner

Progressive Alignment



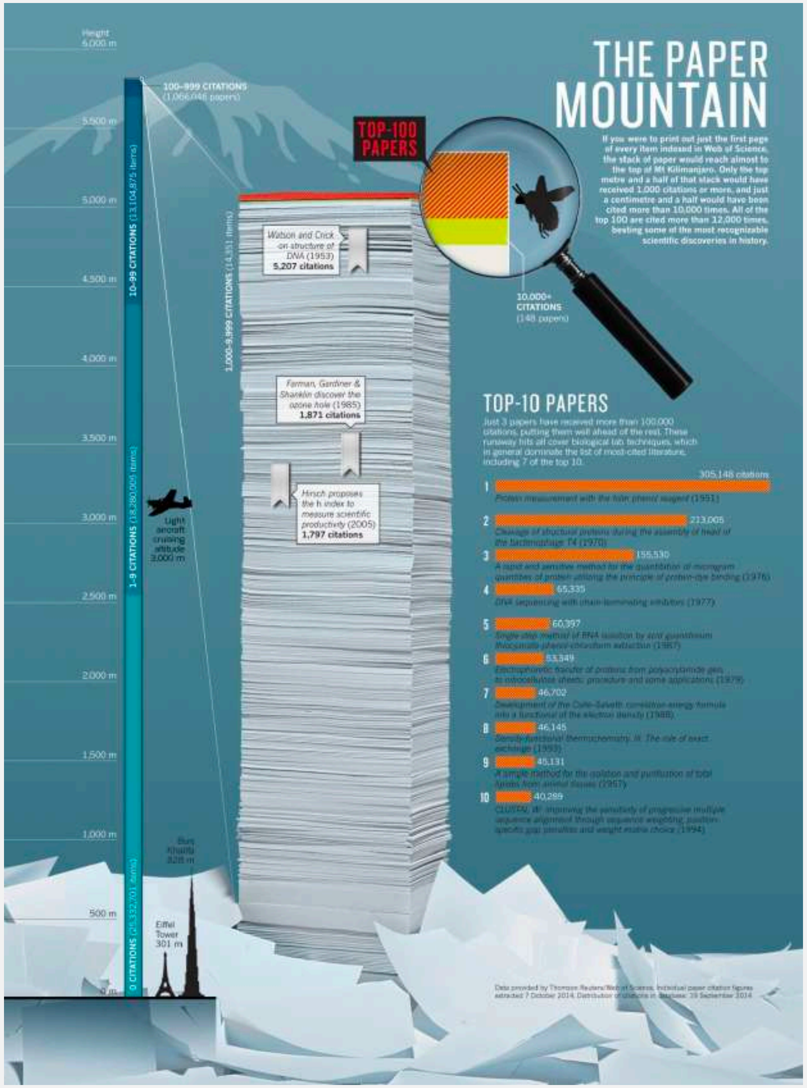
Dynamic Programming Using A Substitution Matrix

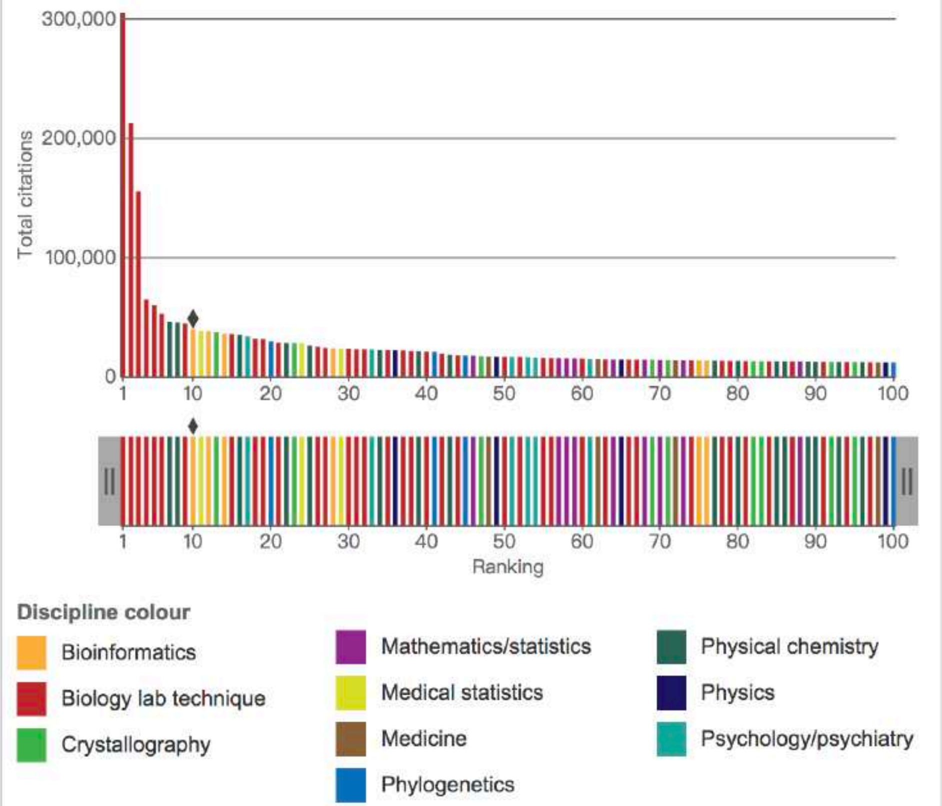
Adapted from Cedric Notredame

ClustalW

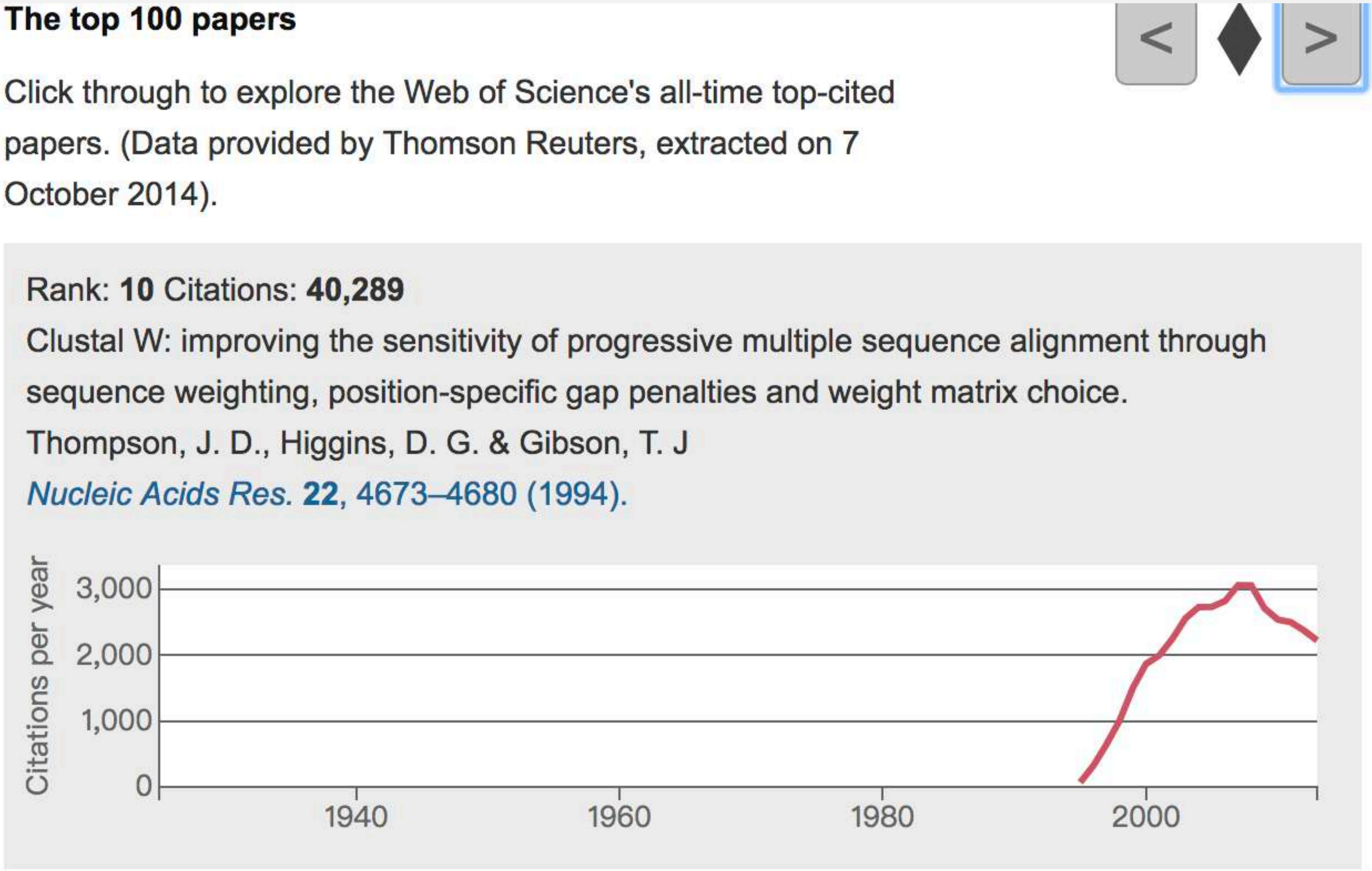
Thompson, J. D., Higgins, D. G. & Gibson, T. J. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* 22, 4673–80 (1994).

The top 100 papers


THE PAPER MOUNTAIN
If you were to print out just the first page of every item indexed in Web of Science, the stack of paper would reach almost to the top of Mt Kilimanjaro. Only the top metre and a half of that stack would have received 1,000 citations or more, and just a centimetre and a half would have been cited more than 10,000 times. All of the top 100 are cited more than 12,000 times, beating some of the most recognizable scientific discoveries in history.
Height 6,000 m
5,500 m
5,000 m
4,500 m
4,000 m
3,500 m
3,000 m
2,500 m
2,000 m
1,500 m
1,000 m
500 m
0 m
100–999 CITATIONS (1,066,046 papers)
10–99 CITATIONS (13,104,875 items)
1–9 CITATIONS (18,280,006 items)
0 CITATIONS (25,332,701 items)
1,000–9,999 CITATIONS (14,351 items)
TOP-100 PAPERS
10,000+ CITATIONS (148 papers)
Watson and Crick on structure of DNA (1953) 5,207 citations
Farman, Gardiner & Shanklin discover the ozone hole (1985) 1,871 citations
Hirsch proposes the h index to measure scientific productivity (2005) 1,797 citations
Light aircraft cruising altitude 3,000 m
Burj Khalifa 828 m
Eiffel Tower 301 m
TOP-10 PAPERS
Just 3 papers have received more than 100,000 citations, putting them well ahead of the rest. These runaway hits all cover biological lab techniques, which in general dominate the list of most-cited literature, including 7 of the top 10.
1 305,148 citations Protein measurement with the folin phenol reagent (1951)
2 213,005 Cleavage of structural proteins during the assembly of head of the bacteriophage T4 (1970)
3 155,530 A rapid and sensitive method for the quantitation of microgram quantities of protein utilizing the principle of protein-dye binding (1976)
4 65,335 DNA sequencing with chain-terminating inhibitors (1977)
5 60,397 Single-step method of RNA isolation by acid guanidinium thiocyanate-phenol-chloroform extraction (1987)
6 53,349 Electrophoretic transfer of proteins from polyacrylamide gels to nitrocellulose sheets: procedure and some applications (1979)
7 46,702 Development of the Colle-Salvetti correlation-energy formula into a functional of the electron density (1988)
8 46,145 Density-functional thermochemistry. III. The role of exact exchange (1993)
9 45,131 A simple method for the isolation and purification of total lipides from animal tissues (1957)
10 40,289 CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice (1994)
Data provided by Thomson Reuters/Web of Science. Individual paper citation figures extracted 7 October 2014. Distribution of citations in database: 19 September 2014



Credit by Nature, http://www.nature.com/news/the-top-100-papers-1.16224



The top 100 papers

Click through to explore the Web of Science's all-time top-cited papers. (Data provided by Thomson Reuters, extracted on 7 October 2014).

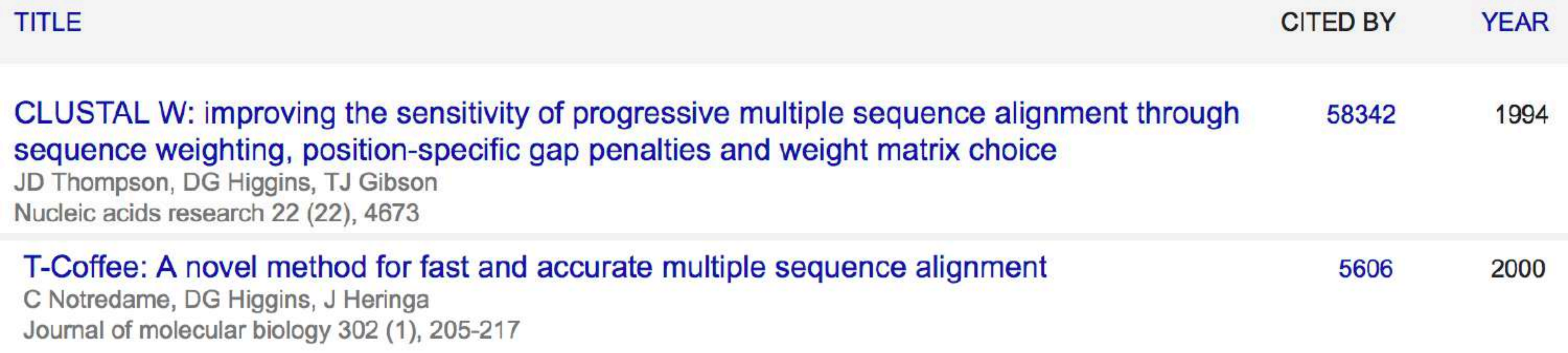
Rank: **10** Citations: **40,289**

Clustal W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice.

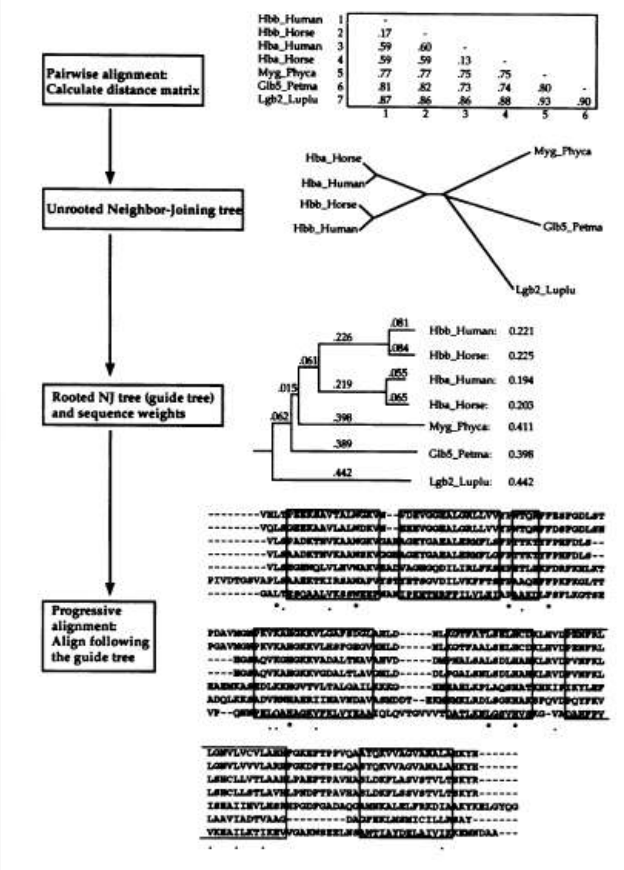
Thompson, J. D., Higgins, D. G. & Gibson, T. J

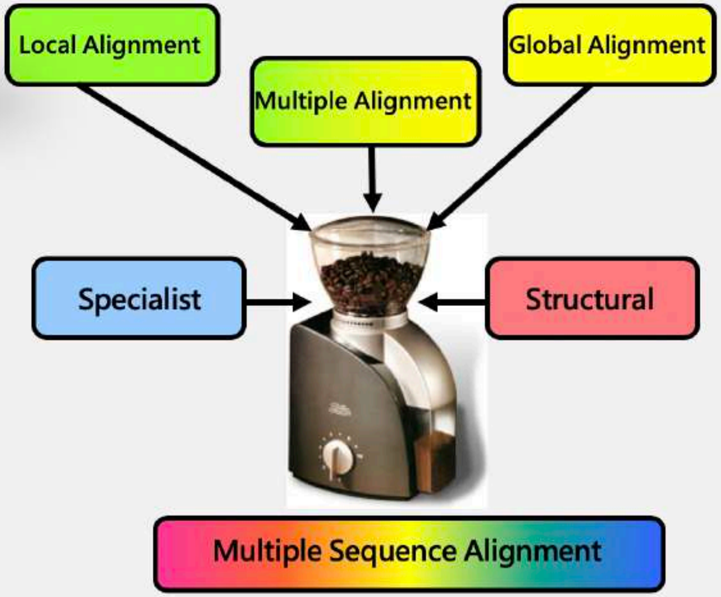
Nucleic Acids Res. **22**, 4673–4680 (1994).

Credit by The top 100 papers. Van Noorden R, Maher B, Nuzzo R. *Nature*. 2014 Oct 30;514(7524):550-3



TITLE	CITED BY	YEAR
CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice JD Thompson, DG Higgins, TJ Gibson Nucleic acids research 22 (22), 4673	58342	1994
T-Coffee: A novel method for fast and accurate multiple sequence alignment C Notredame, DG Higgins, J Heringa Journal of molecular biology 302 (1), 205-217	5606	2000





ClustalW

But owing to the vagaries of citation habits, BLAST has been bumped down the list by Clustal, a complementary programme for aligning multiple sequences at once. Clustal allows researchers to describe the evolutionary relationships between sequences from different organisms, to find matches among seemingly unrelated sequences and to predict how a change at a specific point in a gene or protein might affect its function. A 1994 paper describing ClustalW, a user-friendly version of the software, is currently number 10 on the list. A 1997 paper on a later version called ClustalX is number 28.

Credit by The top 100 papers. Van Noorden R, Maher B, Nuzzo R. *Nature*. 2014 Oct 30;514(7524):550-3

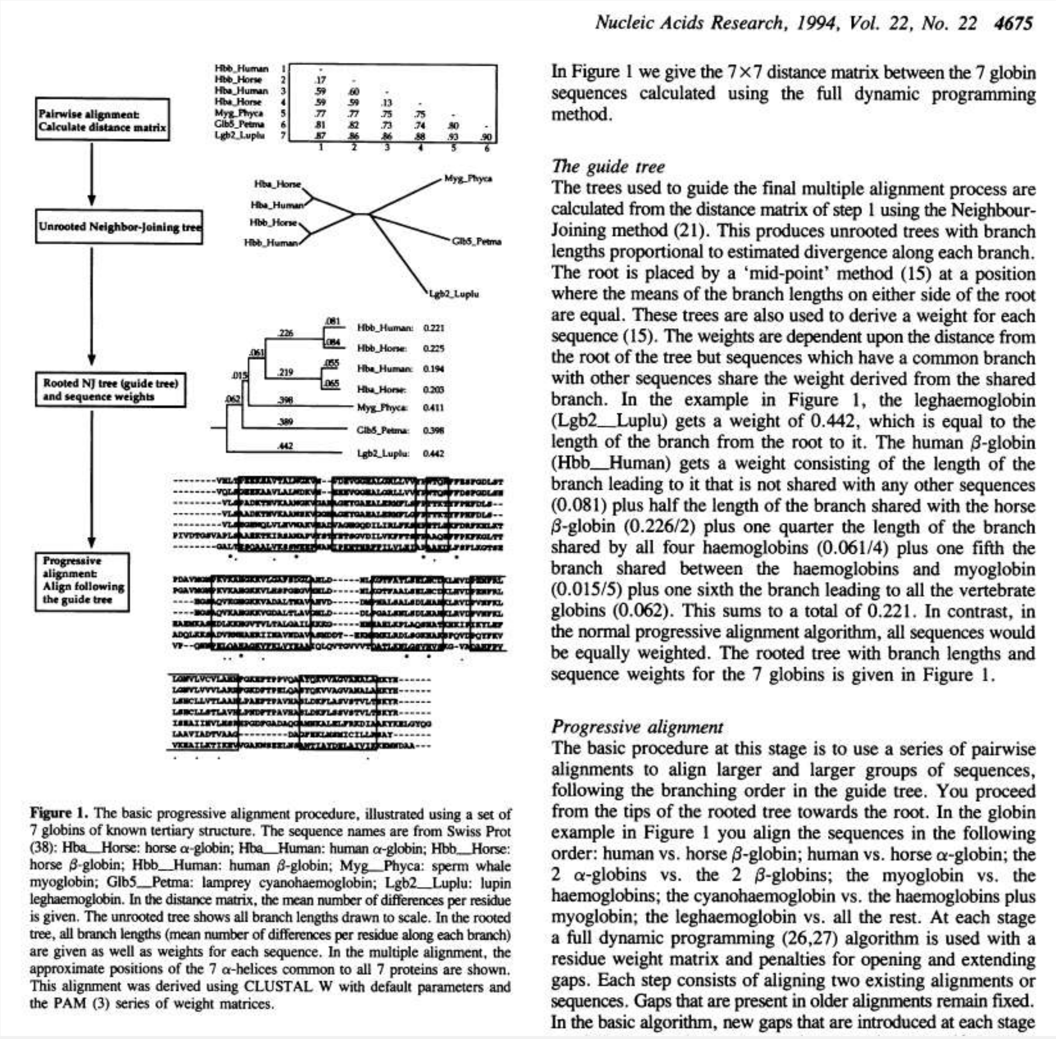
ClustalW

The team that developed ClustalW, at the European Molecular Biology Laboratory in Heidelberg, Germany, had created the program to work on a personal computer, rather than a mainframe. But the software was transformed when Julie Thompson, a computer scientist from the private sector, joined the lab in 1991. “It was a program written by biologists; I’m trying to find a nice way to say that,” says Thompson, who is now at the Institute of Genetics and Molecular and Cellular Biology in Strasbourg, France. Thompson rewrote the program to ready it for the volume and complexity of the genome data being generated at the time, while also making it easier to use.

The teams behind BLAST and Clustal are competitive about the ranking of their papers. It is a friendly sort of competition, however, says Des Higgins, a biologist at University College Dublin, and a member of the Clustal team. “BLAST was a game-changer, and they’ve earned every citation that they get.”

Credit by The top 100 papers. Van Noorden R, Maher B, Nuzzo R. *Nature*. 2014 Oct 30;514(7524):550-3

Thompson et al. (1994) for an explanation of the three stages of progressive alignment implemented in ClustalW



Pairwise alignment: Calculate distance matrix
Unrooted Neighbor-Joining tree
Rooted NJ tree (guide tree) and sequence weights
Progressive alignment: Align following the guide tree
Hba_Horse
Hba_Human
Hbb_Horse
Hbb_Human
Myg_Phyca
Glb5_Petma
Lgb2_Luplu
.226
.081
.084
.061
.219
.055
.065
.015
.062
.398
.389
.442
Hbb_Human: 0.221
Hbb_Horse: 0.225
Hba_Human: 0.194
Hba_Horse: 0.203
Myg_Phyca: 0.411
Glb5_Petma: 0.398
Lgb2_Luplu: 0.442

		1	2	3	4	5	6
Hbb_Human	1	-					
Hbb_Horse	2	.17	-				
Hba_Human	3	.59	.60	-			
Hba_Horse	4	.59	.59	.13	-		
Myg_Phyca	5	.77	.77	.75	.75	-	
Glb5_Petma	6	.81	.82	.73	.74	.80	-
Lgb2_Luplu	7	.87	.86	.86	.88	.93	.90

Figure 1. The basic progressive alignment procedure, illustrated using a set of 7 globins of known tertiary structure. The sequence names are from Swiss Prot (38): Hba__Horse: horse α-globin; Hba__Human: human α-globin; Hbb__Horse: horse β-globin; Hbb__Human: human β-globin; Myg__Phyca: sperm whale myoglobin; Glb5__Petma: lamprey cyanohaemoglobin; Lgb2__Luplu: lupin leghaemoglobin. In the distance matrix, the mean number of differences per residue is given. The unrooted tree shows all branch lengths drawn to scale. In the rooted tree, all branch lengths (mean number of differences per residue along each branch) are given as well as weights for each sequence. In the multiple alignment, the approximate positions of the 7 α-helices common to all 7 proteins are shown. This alignment was derived using CLUSTAL W with default parameters and the PAM (3) series of weight matrices.

In Figure 1 we give the 7×7 distance matrix between the 7 globin sequences calculated using the full dynamic programming method.

The guide tree

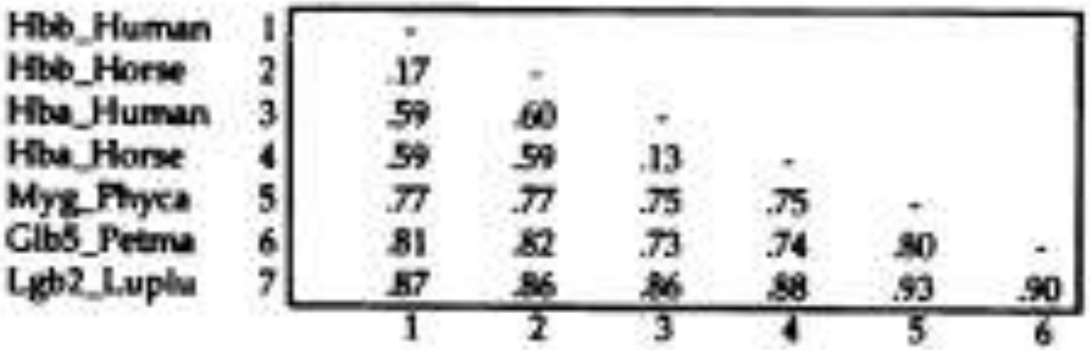
The trees used to guide the final multiple alignment process are calculated from the distance matrix of step 1 using the Neighbour-Joining method (21). This produces unrooted trees with branch lengths proportional to estimated divergence along each branch. The root is placed by a 'mid-point' method (15) at a position where the means of the branch lengths on either side of the root are equal. These trees are also used to derive a weight for each sequence (15). The weights are dependent upon the distance from the root of the tree but sequences which have a common branch with other sequences share the weight derived from the shared branch. In the example in Figure 1, the leghaemoglobin (Lgb2__Luplu) gets a weight of 0.442, which is equal to the length of the branch from the root to it. The human β-globin (Hbb__Human) gets a weight consisting of the length of the branch leading to it that is not shared with any other sequences (0.081) plus half the length of the branch shared with the horse β-globin (0.226/2) plus one quarter the length of the branch shared by all four haemoglobins (0.061/4) plus one fifth the branch shared between the haemoglobins and myoglobin (0.015/5) plus one sixth the branch leading to all the vertebrate globins (0.062). This sums to a total of 0.221. In contrast, in the normal progressive alignment algorithm, all sequences would be equally weighted. The rooted tree with branch lengths and sequence weights for the 7 globins is given in Figure 1.

Progressive alignment

The basic procedure at this stage is to use a series of pairwise alignments to align larger and larger groups of sequences, following the branching order in the guide tree. You proceed from the tips of the rooted tree towards the root. In the globin example in Figure 1 you align the sequences in the following order: human vs. horse β-globin; human vs. horse α-globin; the 2 α-globins vs. the 2 β-globins; the myoglobin vs. the haemoglobins; the cyanohaemoglobin vs. the haemoglobins plus myoglobin; the leghaemoglobin vs. all the rest. At each stage a full dynamic programming (26,27) algorithm is used with a residue weight matrix and penalties for opening and extending gaps. Each step consists of aligning two existing alignments or sequences. Gaps that are present in older alignments remain fixed. In the basic algorithm, new gaps that are introduced at each stage

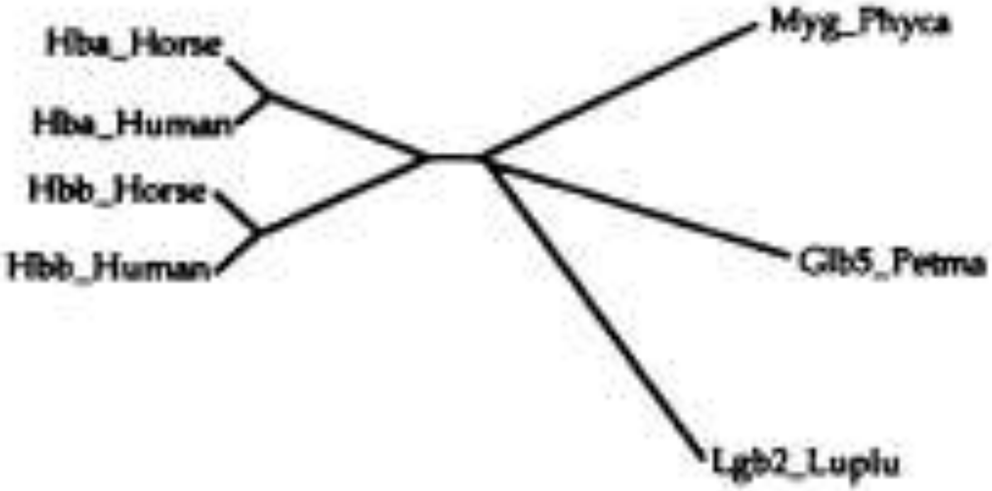
Credit by B&FG 3e, Jonathan Pevsner

Pairwise alignment:
Calculate distance matrix



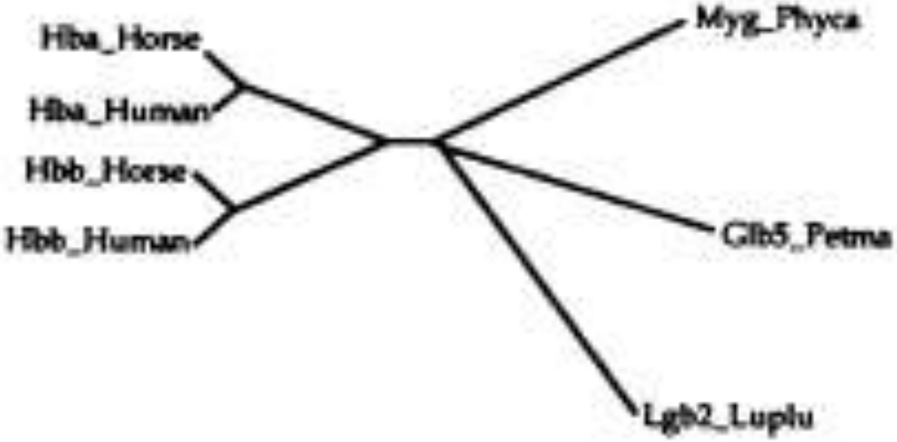
		1	2	3	4	5	6
Hbb_Human	1	-					
Hbb_Horse	2	.17	-				
Hba_Human	3	.59	.60	-			
Hba_Horse	4	.59	.59	.13	-		
Myg_Phyca	5	.77	.77	.75	.75	-	
Glb5_Petma	6	.81	.82	.73	.74	.80	-
Lgb2_Luplu	7	.87	.86	.86	.88	.93	.90

Unrooted neighbor-joining tree

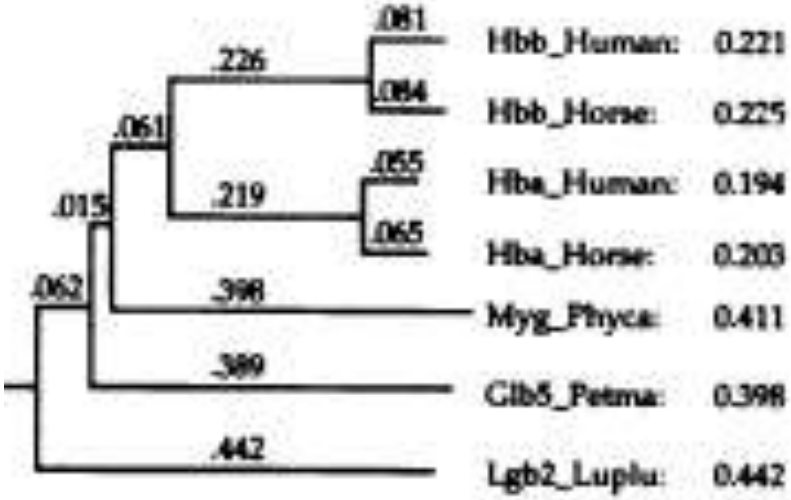


Credit by B&FG 3e, Jonathan Pevsner

Unrooted neighbor-joining tree

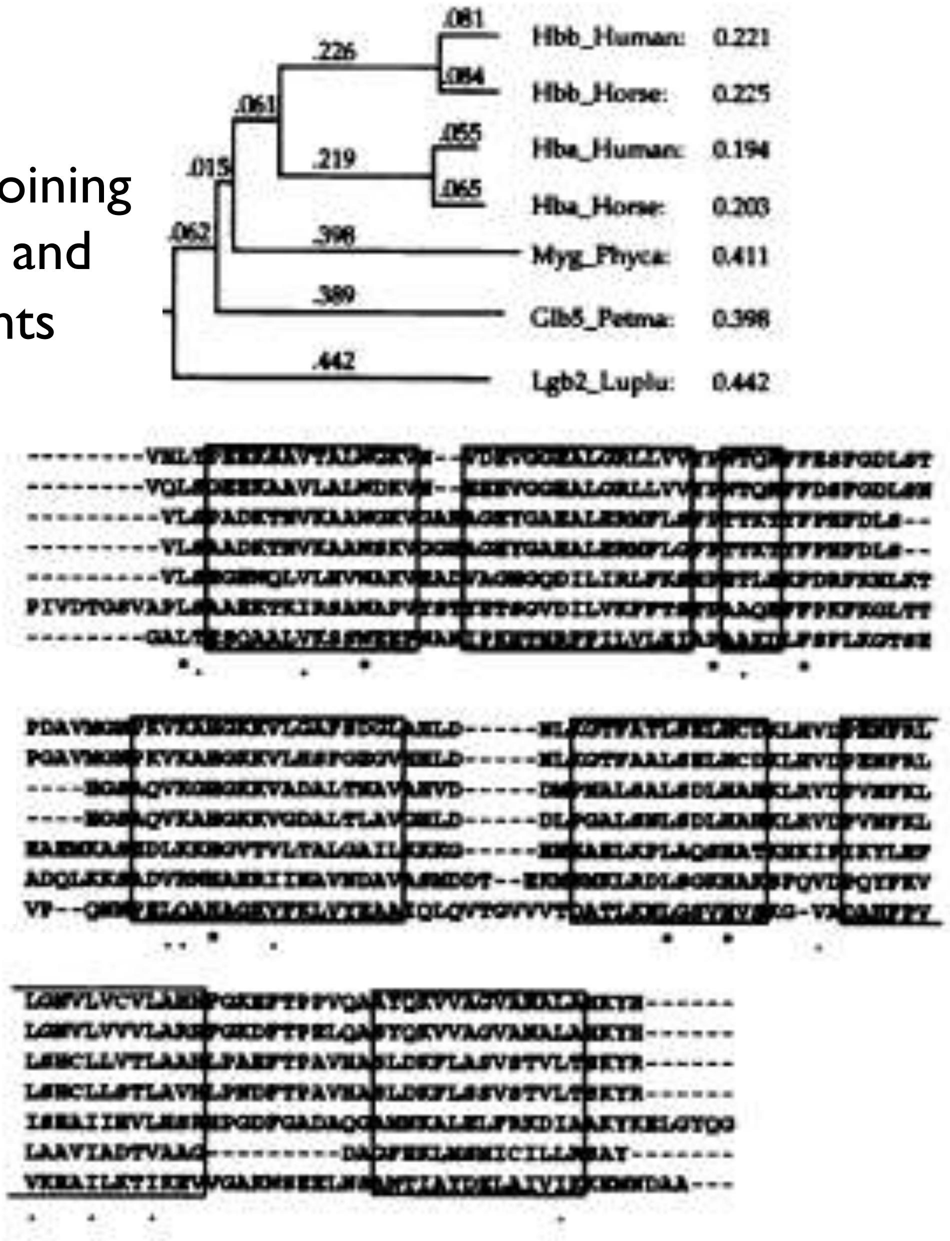


Rooted neighbor-joining tree (guide tree) & sequence weights



Credit by B&FG 3e, Jonathan Pevsner

Rooted neighbor-joining tree (guide tree) and sequence weights



Progressive alignment: Align following the guide tree

Credit by B&FG 3e, Jonathan Pevsner

Additional features of ClustalW improve its ability to generate accurate MSAs

- Individual weights are assigned to sequences; very closely related sequences are given less weight, while distantly related sequences are given more weight
- Scoring matrices are varied dependent on the presence of conserved or divergent sequences, e.g.:

PAM20	80-100% id
PAM60	60-80% id
PAM120	40-60% id
PAM350	0-40% id

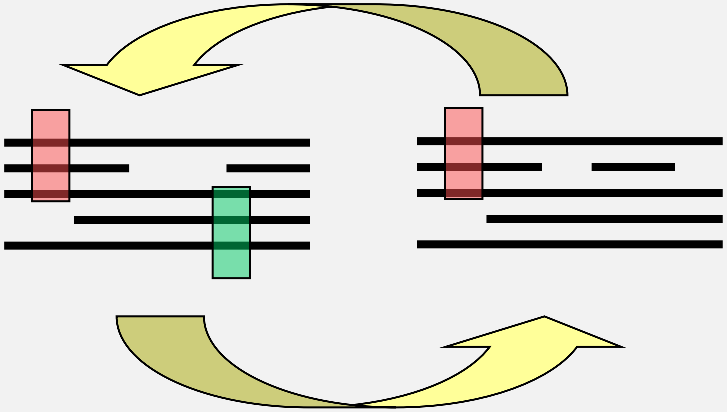
- Residue-specific gap penalties are applied

Credit by B&FG 3e, Jonathan Pevsner

Iterative approaches

Iterative methods

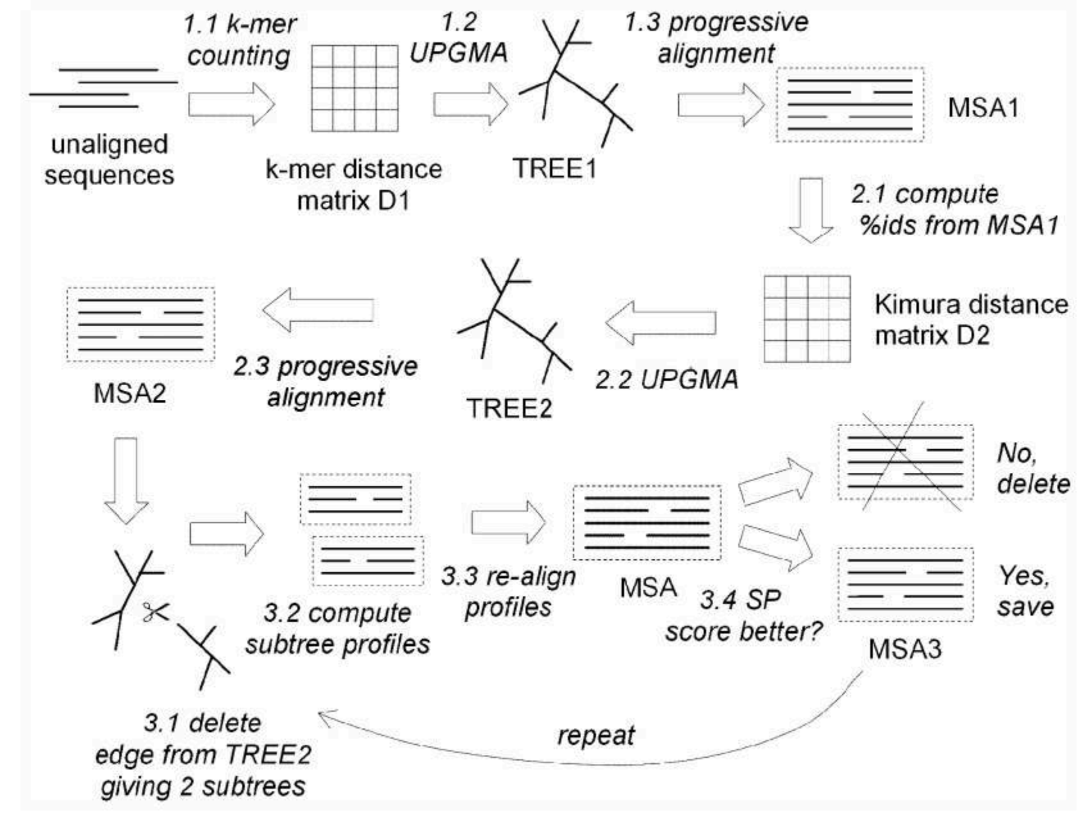
- compute a sub-optimal solution and keep modifying that intelligently using dynamic programming or other methods until the solution converges.
- MUSCLE, Mafft, HMMs, HMMER, SAM,, IterAlign, Praline
- +: Good Profile Generators
- -: Slow, Sometimes Inaccurate



Adapted from Cedric Notredame

Muscle

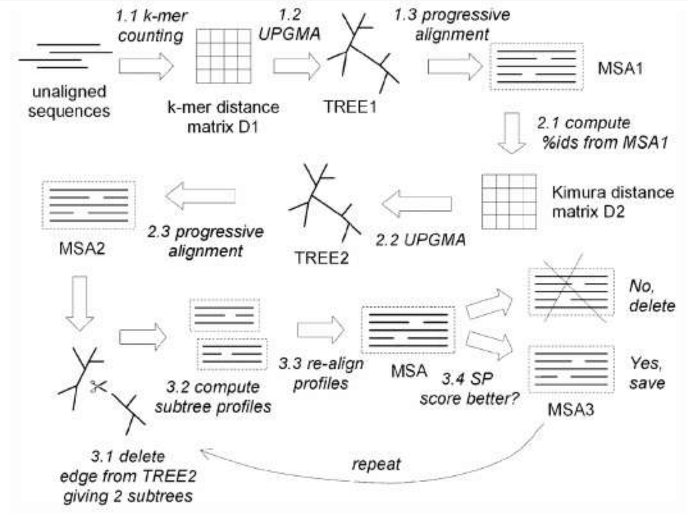
- Edgar, R. MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *Bmc Bioinformatics* **5**, 1–19 (2004).
- Edgar, R. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res* **32**,1792–1797 (2004)


1.1 k-mer counting
1.2 UPGMA
1.3 progressive alignment
unaligned sequences
k-mer distance matrix D1
TREE1
MSA1
2.1 compute %ids from MSA1
Kimura distance matrix D2
2.2 UPGMA
TREE2
2.3 progressive alignment
MSA2
3.1 delete edge from TREE2 giving 2 subtrees
3.2 compute subtree profiles
3.3 re-align profiles
MSA
3.4 SP score better?
No, delete
Yes, save
MSA3
repeat

Adapted from Cedric Notredame

MUSCLE: Improve the progressive alignment

- Build a draft progressive alignment
- Determine pairwise similarity through *k*-mer counting
- Compute triangular distance matrix
- Construct tree using UPGMA
- Construct draft progressive alignment following tree
- Compute pairwise identity through current MSA
- Construct new tree with *Kimura* distance measures
- Compare new and old trees: if improved, repeat this step, if not improved, then we're done
- Refinement of the MSA
- Split tree in half by deleting one edge
- Make profiles of each half of the tree
- Re-align the profiles
- Accept/reject the new alignment

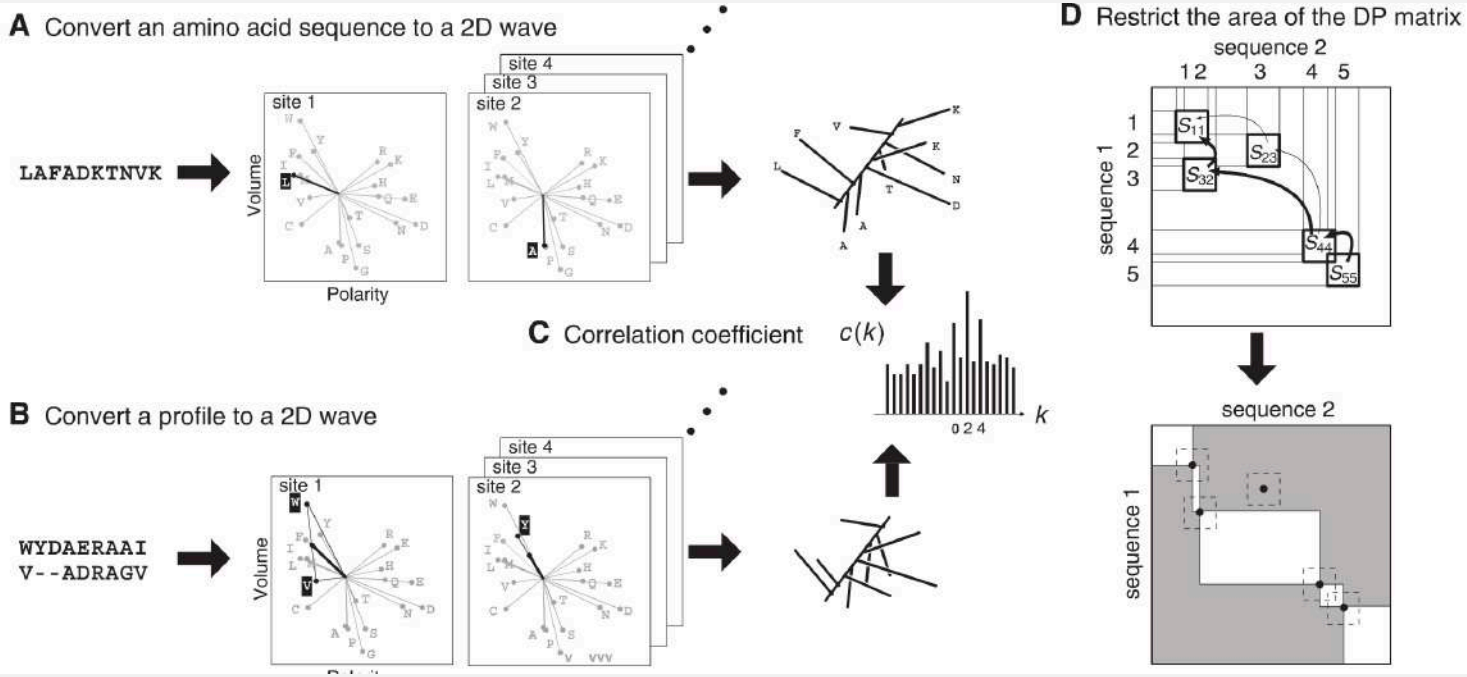


1.1 k-mer counting
1.2 UPGMA
1.3 progressive alignment
unaligned sequences
k-mer distance matrix D1
TREE1
MSA1
2.1 compute %ids from MSA1
Kimura distance matrix D2
2.2 UPGMA
TREE2
2.3 progressive alignment
MSA2
No, delete
3.3 re-align profiles
MSA
3.4 SP score better?
Yes, save
MSA3
3.2 compute subtree profiles
3.1 delete edge from TREE2 giving 2 subtrees
repeat

Credit by B&FG 3e, Jonathan Pevsner

MAFFT : Fast Fourrier Transforme

- Katoh, K., Misawa, K., Kuma, K. & Miyata, T. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res.* **30,** 3059–66 (2002).
- Katoh, K. & Toh, H. Recent developments in the MAFFT multiple sequence alignment program. *Brief. Bioinformatics* **9,** 286–98 (2008).



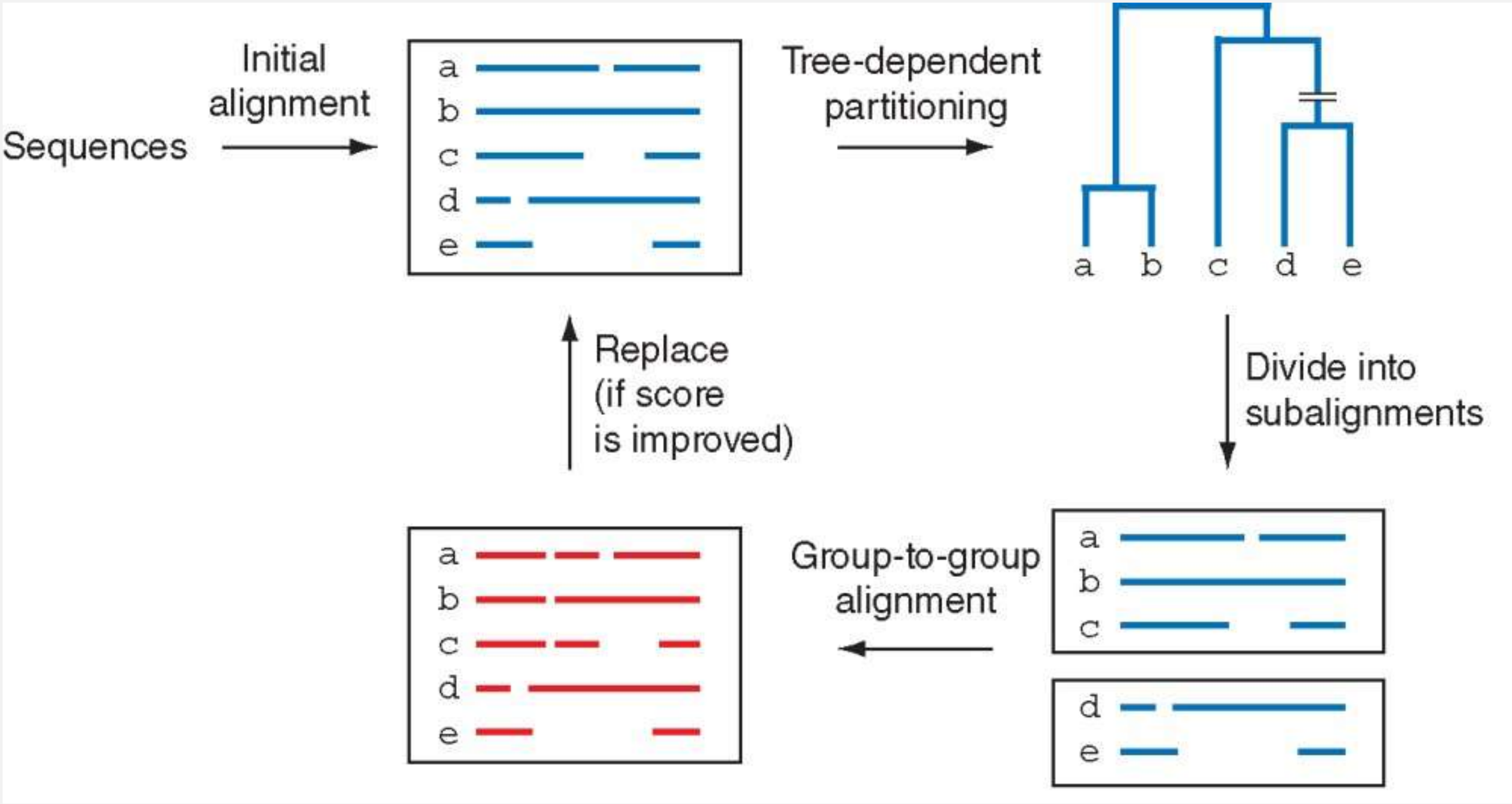
Adapted from Cedric Notredame

MAFFT

- Uses Fast Fourier Transform to speed up profile alignment
- Uses fast two-stage method for building alignments using *k*-mer frequencies
- Offers many different scoring and aligning techniques
- One of the more accurate programs available
- Available as standalone or web interface
- Many output formats, including interactive phylogenetic trees

Credit by B&FG 3e, Jonathan Pevsner

Iterative method of MAFFT

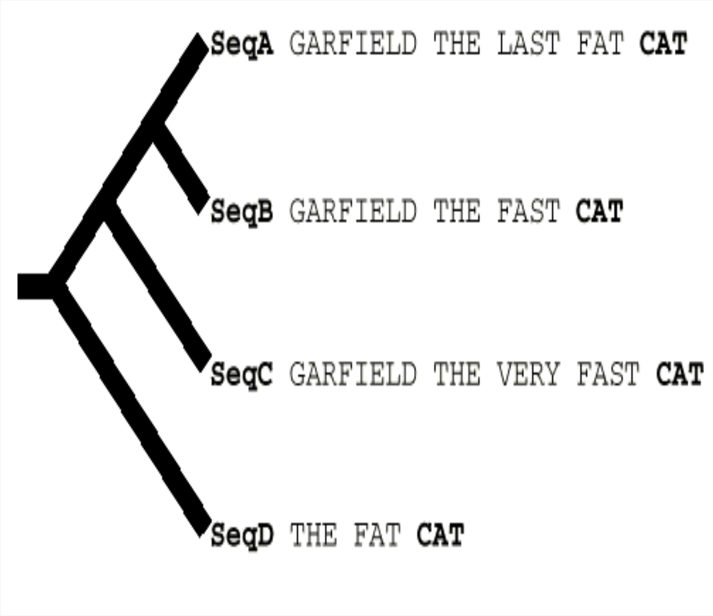


Initial alignment
Sequences
a
b
c
d
e
Tree-dependent partitioning
a
b
c
d
e
Divide into subalignments
a
b
c
d
e
Group-to-group alignment
a
b
c
d
e
Replace (if score is improved)

B&FG 3e Fig. 6.6 Page 215

Consistency-based approaches

Progressive Alignment
When It Doesn’t Work


SeqA GARFIELD THE LAST FAT CAT
SeqB GARFIELD THE FAST CAT
SeqC GARFIELD THE VERY FAST CAT
SeqD THE FAT CAT

CLUSTALW (Score=20, Gop=-1, Gep=0, M=1)

```
SeqA GARFIELD THE LAST FA-T CAT
SeqB GARFIELD THE FAST CA-T ---
SeqC GARFIELD THE VERY FAST CAT
SeqD -------- THE ---- FA-T CAT
```

CORRECT (Score=24)

```
SeqA GARFIELD THE LAST FA-T CAT
SeqB GARFIELD THE FAST ---- CAT
SeqC GARFIELD THE VERY FAST CAT
SeqD -------- THE ---- FA-T CAT
```

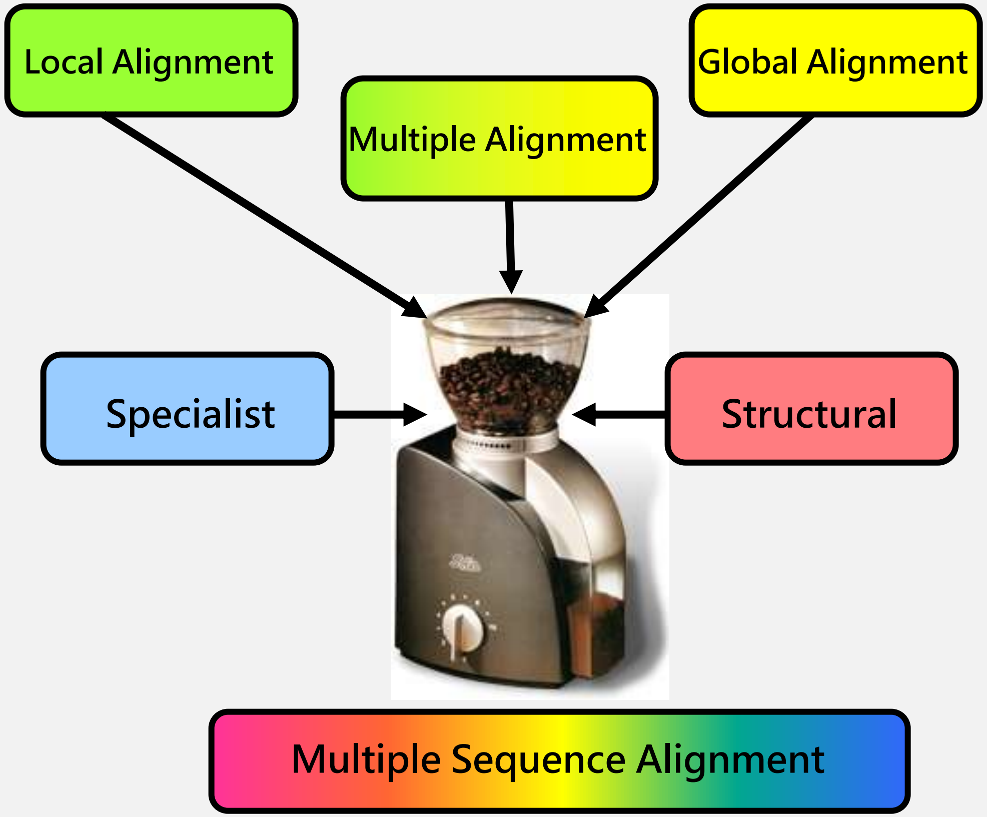
Adapted from Cedric Notredame

Multiple sequence alignment: consistency

- generally use a database of both local high-scoring alignments and long-range global alignments to create a final alignment
- These are very powerful and very accurate methods
- Examples: T-Coffee, Prrp, DiAlign, ProbCons

Credit by B&FG 3e, Jonathan Pevsner

Mixing Heterogenous Data With T-Coffee



Local Alignment

Multiple Alignment

Global Alignment

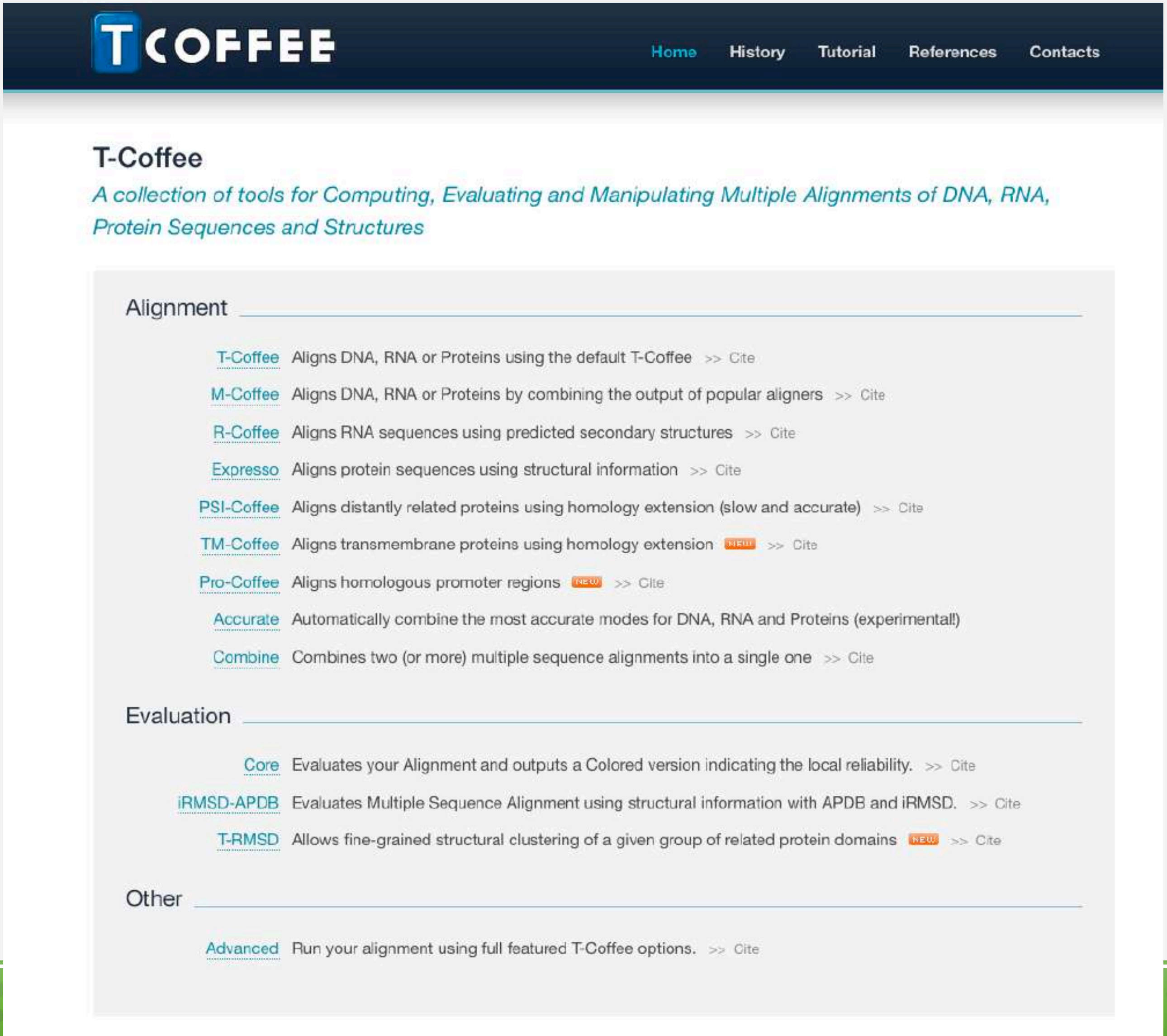
Specialist

Structural

Multiple Sequence Alignment

Adapted from Cedric Notredame

www.tcoffee.org



T-Coffee

A collection of tools for Computing, Evaluating and Manipulating Multiple Alignments of DNA, RNA, Protein Sequences and Structures

Alignment

T-Coffee	Aligns DNA, RNA or Proteins using the default T-Coffee >> Cite
M-Coffee	Aligns DNA, RNA or Proteins by combining the output of popular aligners >> Cite
R-Coffee	Aligns RNA sequences using predicted secondary structures >> Cite
Expresso	Aligns protein sequences using structural information >> Cite
PSI-Coffee	Aligns distantly related proteins using homology extension (slow and accurate) >> Cite
TM-Coffee	Aligns transmembrane proteins using homology extension NEW >> Cite
Pro-Coffee	Aligns homologous promoter regions NEW >> Cite
Accurate	Automatically combine the most accurate modes for DNA, RNA and Proteins (experimental!)
Combine	Combines two (or more) multiple sequence alignments into a single one >> Cite

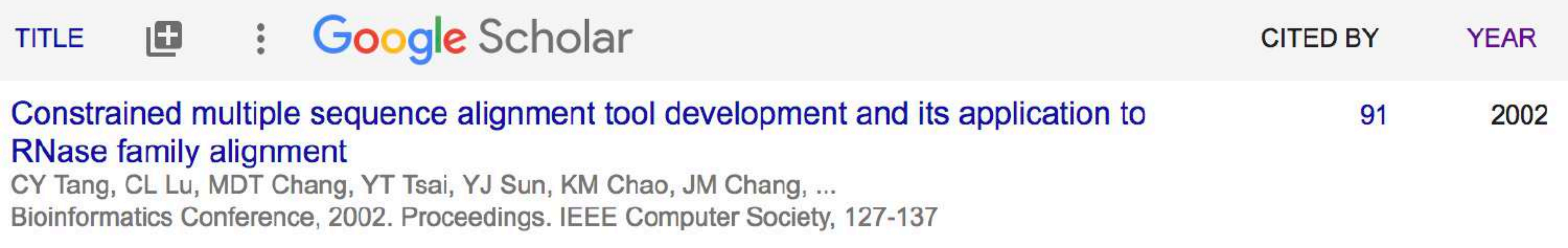
Evaluation

Core	Evaluates your Alignment and outputs a Colored version indicating the local reliability. >> Cite
iRMSD-APDB	Evaluates Multiple Sequence Alignment using structural information with APDB and iRMSD. >> Cite
T-RMSD	Allows fine-grained structural clustering of a given group of related protein domains NEW >> Cite

Other

Advanced	Run your alignment using full featured T-Coffee options. >> Cite

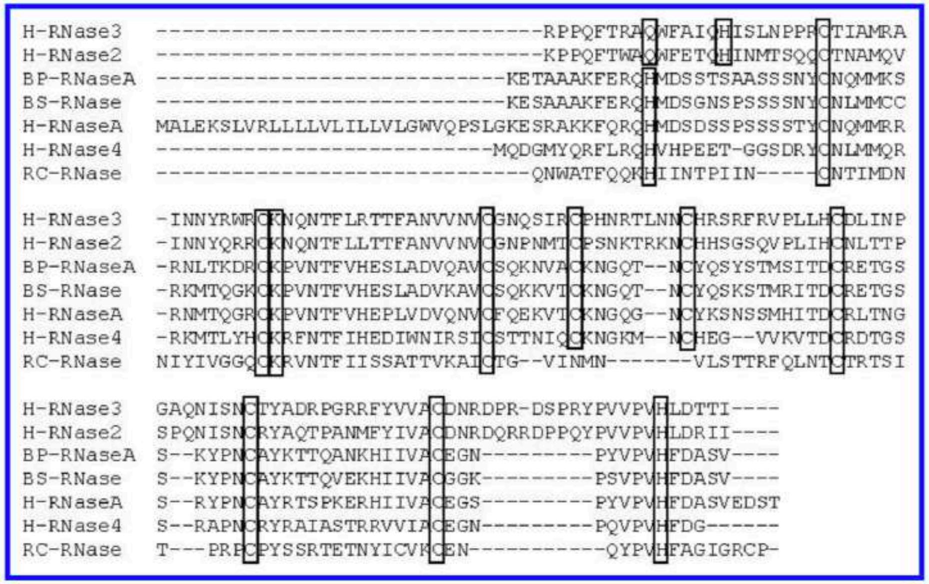
Constrained MSA



TITLE ⊞ ⋮ Google Scholar CITED BY YEAR

Constrained multiple sequence alignment tool development and its application to RNase family alignment — 91 — 2002

CY Tang, CL Lu, MDT Chang, YT Tsai, YJ Sun, KM Chao, JM Chang, ...
Bioinformatics Conference, 2002. Proceedings. IEEE Computer Society, 127-137

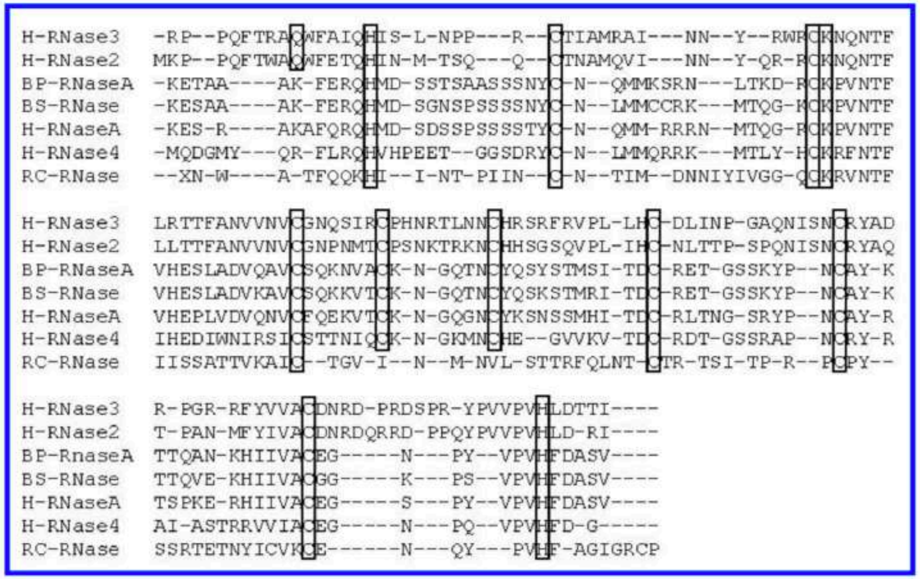


```
H-RNase3   -------------------------------RPPQFTRAQWFAIQHISLNPPRCTIAMRA
H-RNase2   -------------------------------KPPQFTWAQWFETQHINMTSQQCTNAMQV
BP-RNaseA  ----------------------------KETAAAKFERQHMDSSTSAASSSNYCNQMMKS
BS-RNase   ----------------------------KESAAAKFERQHMDSGNSPSSSSNYCNLMMCC
H-RNaseA   MALEKSLVRLLLLVLILLVLGWVQPSLGKESRAKKFQRQHMDSDSSPSSSSTYCNQMMRR
H-RNase4   ---------------------------MQDGMYQRFLRQHVHPEET-GGSDRYCNLMMQR
RC-RNase   ------------------------------QNWATFQQKHIINTPIIN-----CNTIMDN

H-RNase3   -INNYRWRCKNQNTFLRTTFANVVNVCGNQSIRCPHNRTLNNCHRSRFRVPLLHCDLINP
H-RNase2   -INNYQRRCKNQNTFLLTTFANVVNVCGNPNMTCPSNKTRKNCHHSGSQVPLIHCNLTTP
BP-RNaseA  -RNLTKDRCKPVNTFVHESLADVQAVCSQKNVACKNGQT--NCYQSYSTMSITDCRETGS
BS-RNase   -RKMTQGKCKPVNTFVHESLADVKAVCSQKKVTCKNGQT--NCYQSKSTMRITDCRETGS
H-RNaseA   -RNMTQGRCKPVNTFVHEPLVDVQNVCFQEKVTCKNGQG--NCYKSNSSMHITDCRLTNG
H-RNase4   -RKMTLYHCKRFNTFIHEDIWNIRSICSTTNIQCKNGKM--NCHEG--VVKVTDCRDTGS
RC-RNase   NIYIVGGQCKRVNTFIISSATTVKAICTG--VINMN-------VLSTTRFQLNTCTRTSI

H-RNase3   GAQNISNCTYADRPGRRFYVVACDNRDPR-DSPRYPVVPVHLDTTI----
H-RNase2   SPQNISNCRYAQTPANMFYIVACDNRDQRRDPPQYPVVPVHLDRII----
BP-RNaseA  S--KYPNCAYKTTQANKHIIVACEGN---------PYVPVHFDASV----
BS-RNase   S--KYPNCAYKTTQVEKHIIVACGGK---------PSVPVHFDASV----
H-RNaseA   S--RYPNCAYRTSPKERHIIVACEGS---------PYVPVHFDASVEDST
H-RNase4   S--RAPNCRYRAIASTRRVVIACEGN---------PQVPVHFDG------
RC-RNase   T---PRPCPYSSRTETNYICVKCEN-----------QYPVHFAGIGRCP-
```

Fig. 1. The multiple sequence alignment of seven RNases by WorkBench 3.2: The key active site residues homologous to His12, Lys41, and His119 of BP-RNaseA, the cysteine residues responsible for disulfide bond linkage and two matched Gln residues are shown in boxes.



```
H-RNase3   -RP--PQFTRAQWFAIQHIS-L-NPP---R--CTIAMRAI---NN--Y--RWRCKNQNTF
H-RNase2   MKP--PQFTWAQWFETQHIN-M-TSQ---Q--CTNAMQVI---NN--Y-QR-RCKNQNTF
BP-RNaseA  -KETAA----AK-FERQHMD-SSTSAASSSNYC-N--QMMKSRN---LTKD-RCKPVNTF
BS-RNase   -KESAA----AK-FERQHMD-SGNSPSSSSNYC-N--LMMCCRK---MTQG-KCKPVNTF
H-RNaseA   -KES-R----AKAFQRQHMD-SDSSPSSSSTYC-N--QMM-RRRN--MTQG-RCKPVNTF
H-RNase4   -MQDGMY---QR-FLRQHVHPEET--GGSDRYC-N--LMMQRRK---MTLY-HCKRFNTF
RC-RNase   --XN-W----A-TFQQKHI--I-NT-PIIN--C-N--TIM--DNNIYIVGG-QCKRVNTF

H-RNase3   LRTTFANVVNVCGNQSIRCPHNRTLNNCHRSRFRVPL-LHC-DLINP-GAQNISNCRYAD
H-RNase2   LLTTFANVVNVCGNPNMTCPSNKTRKNCHHSGSQVPL-IHC-NLTTP-SPQNISNCRYAQ
BP-RNaseA  VHESLADVQAVCSQKNVACK-N-GQTNCYQSYSTMSI-TDC-RET-GSSKYP--NCAY-K
BS-RNase   VHESLADVKAVCSQKKVTCK-N-GQTNCYQSKSTMRI-TDC-RET-GSSKYP--NCAY-K
H-RNaseA   VHEPLVDVQNVCFQEKVTCK-N-GQGNCYKSNSSMHI-TDC-RLTNG-SRYP--NCAY-R
H-RNase4   IHEDIWNIRSICSTTNIQCK-N-GKMNCHE--GVVKV-TDC-RDT-GSSRAP--NCRY-R
RC-RNase   IISSATTVKAIC--TGV-I--N--M-NVL-STTRFQLNT-CTR-TSI-TP-R--PCPY--

H-RNase3   R-PGR-RFYVVACDNRD-PRDSPR-YPVVPVHLDTTI----
H-RNase2   T-PAN-MFYIVACDNRDQRRD-PPQYPVVPVHLD-RI----
BP-RnaseA  TTQAN-KHIIVACEG-----N---PY--VPVHFDASV----
BS-RNase   TTQVE-KHIIVACGG-----K---PS--VPVHFDASV----
H-RNaseA   TSPKE-RHIIVACEG-----S---PY--VPVHFDASV----
H-RNase4   AI-ASTRRVVIACEG-----N---PQ--VPVHFD-G-----
RC-RNase   SSRTETNYICVKCE------N---QY---PVHF-AGIGRCP
```

Fig. 2. The multiple sequence alignment of seven RNases by our CMSA: The key active site residues homologous to His12, Lys41, and His119 of BP-RNaseA, the cysteine residues responsible for disulfide bond linkage, and two matched Gln residues are shown in boxes.

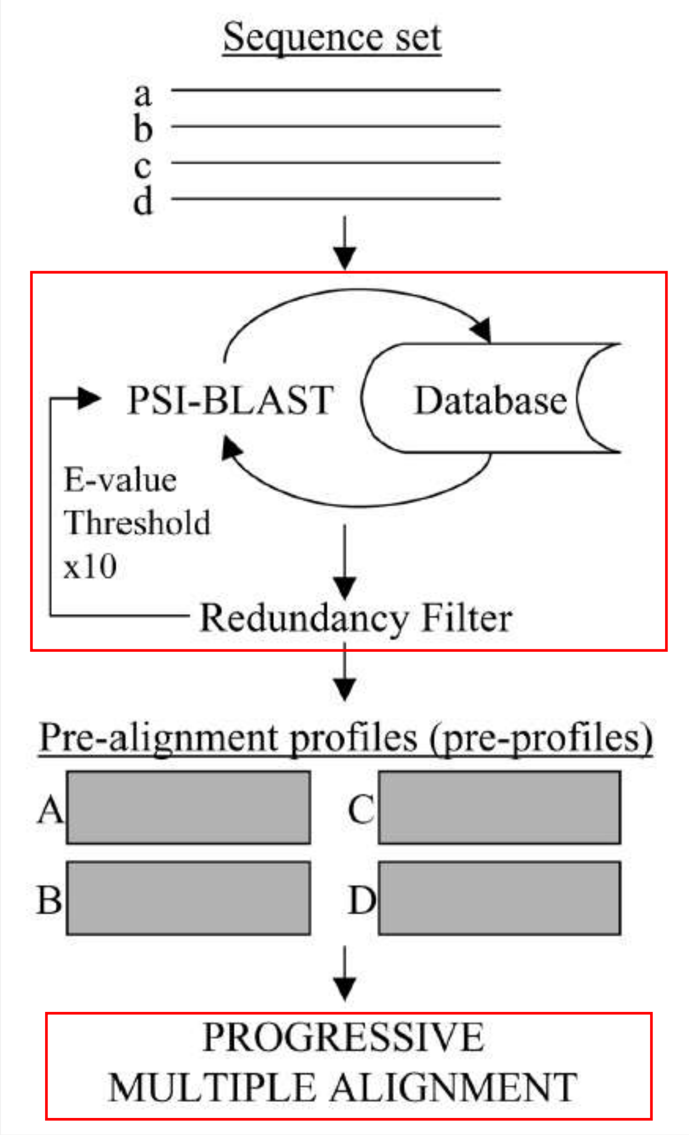
Homology extension approach

Chang, J.-M., Tommaso, P., Taly, J.-F. & Notredame, C. Accurate multiple sequence alignment of transmembrane proteins with PSI-Coffee. *Bmc Bioinformatics* **13,** 1–7 (2012).

Homology-extended

Que1: how to build a profile?

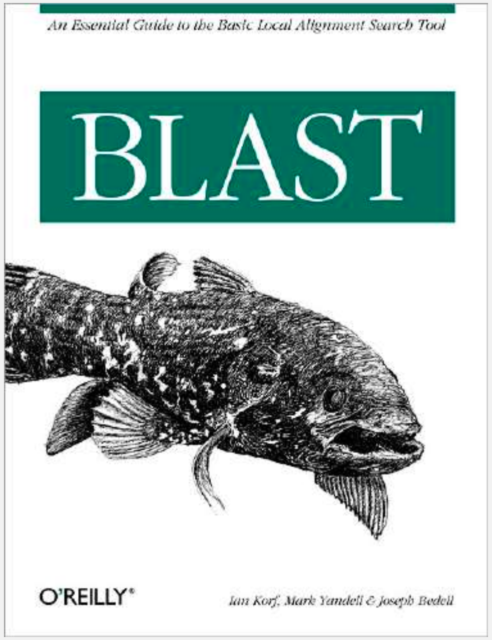
Que2: how to score profiles?



Simossis VA, Kleinjung J, Heringa J: **Homology-extended sequence alignment**. *Nucleic Acids Res* 2005, **33**(3):816-824.

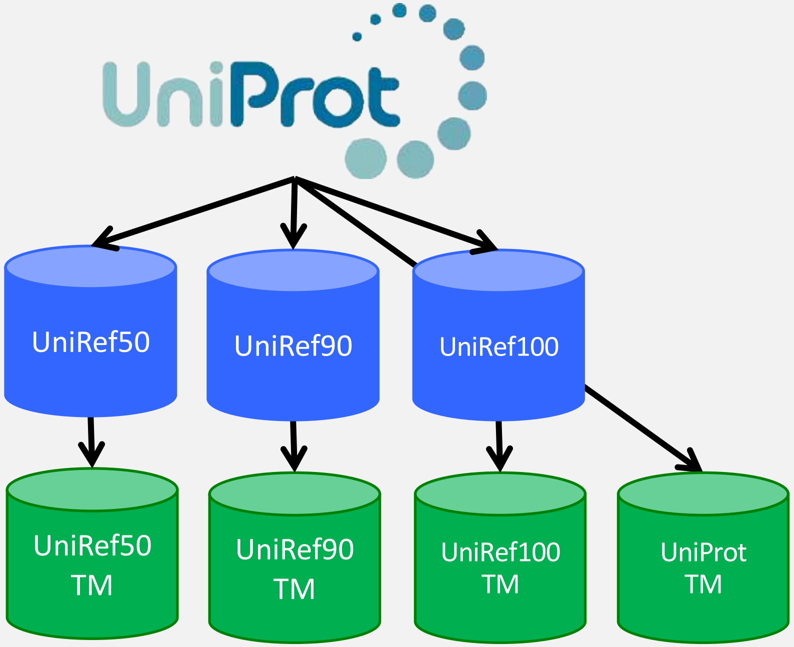
Searching parameters

- Fast, Insensitive search
  - High percent identity
  - blastp –F “m S” –f 999 –M BLOSUM80 –G 9 –E 2 –e 1e-5
- Slow, Sensitive search
  - Increase sensitivity, decrease specificity
  - blastp –F “m S” –f 9 –M BLOSUM45 –e 100 –b 10000 –v 10000
- BLAST, page 146, 147
  - By Ian Korf, Joseph Bedell, Mark Yandell
  - Publisher: O'Reilly Media
  - Release Date: July 2003


An Essential Guide to the Basic Local Alignment Search Tool
BLAST
O'REILLY
Ian Korf, Mark Yandell & Joseph Bedell

Database Size

NCBI non-redundant (NR)
UniProt (release 15.15 – 2010)

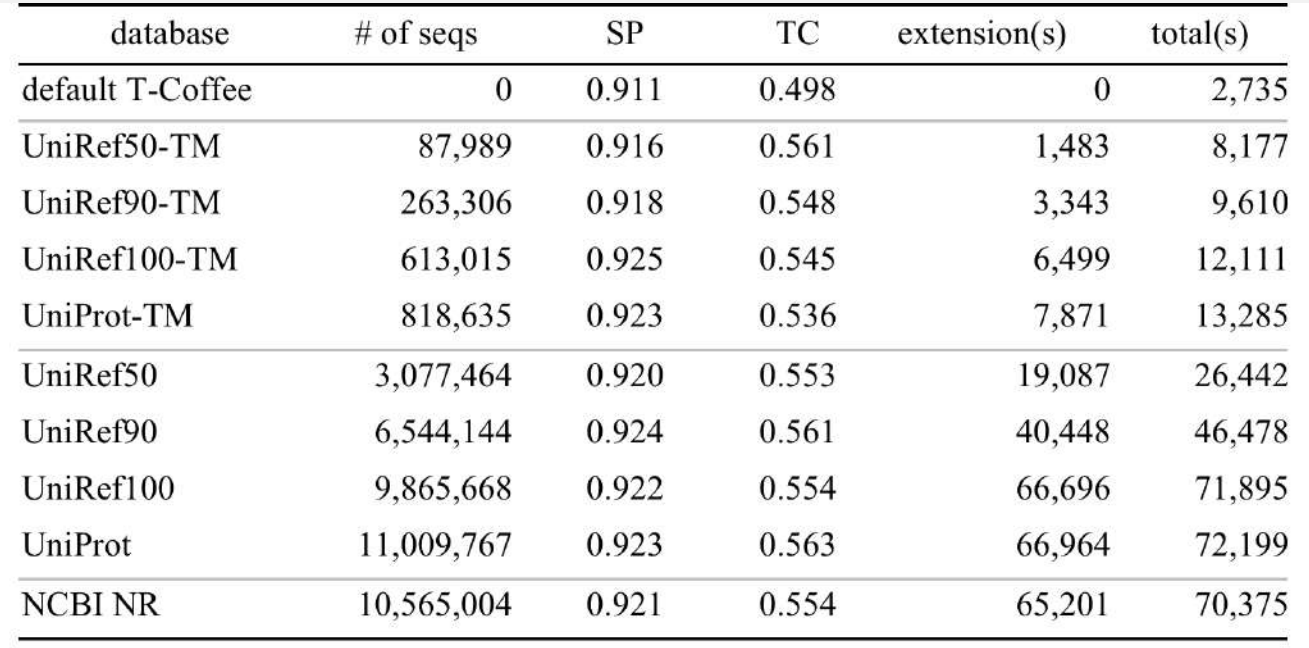


keyword:"Transmembrane [KW-0812]"

Data Set	No.
UniRef50-TM	87,989
UniRef90-TM	263,306
UniRef100-TM	613,015
UniProt-TM	818,635
UniRef50	3,077,464
UniRef90	6,544,144
UniRef100	9,865,668
UniProt	11,009,767
NCBI NR	10,565,004

Performance comparison of different database sizes for the BAliBASE2-ref7.

- UniRef50-TM contains about 100 times fewer sequences than the full UniProt.
- The level accuracy is comparable and even superior to that achieved with the default PSI-Coffee while the CPU time requirements are dramatically decreased by a factor 10.

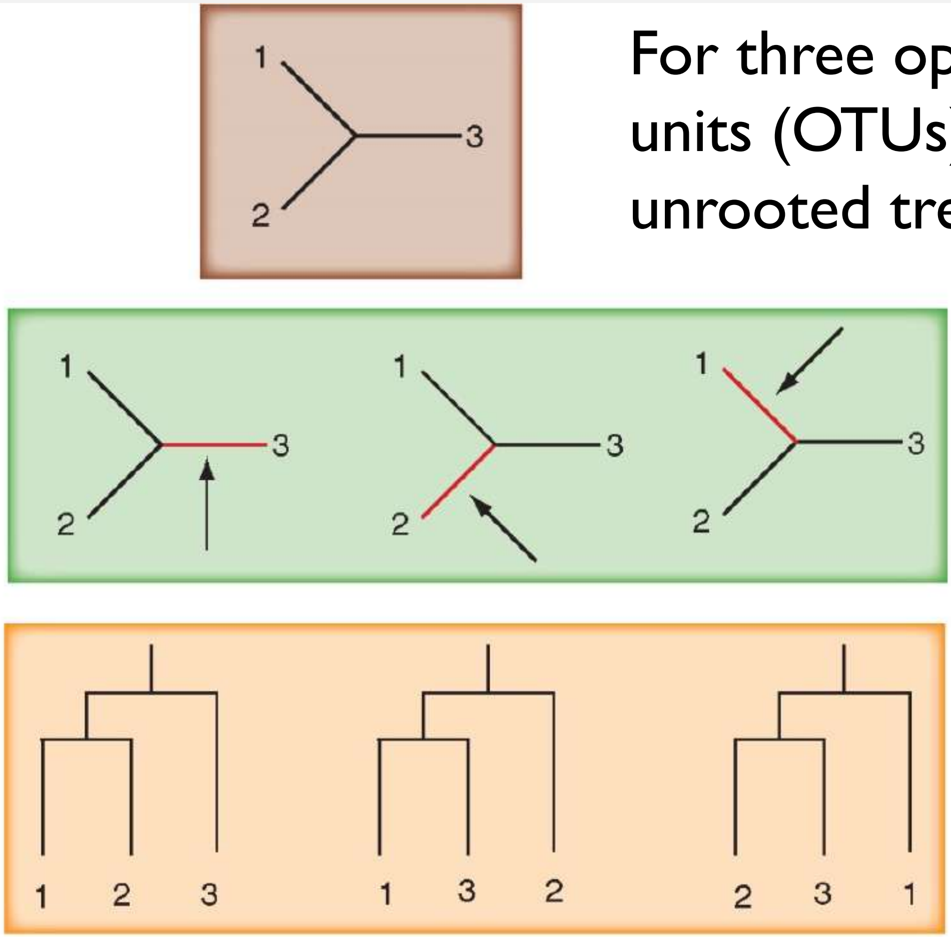


database	# of seqs	SP	TC	extension(s)	total(s)
default T-Coffee	0	0.911	0.498	0	2,735
UniRef50-TM	87,989	0.916	0.561	1,483	8,177
UniRef90-TM	263,306	0.918	0.548	3,343	9,610
UniRef100-TM	613,015	0.925	0.545	6,499	12,111
UniProt-TM	818,635	0.923	0.536	7,871	13,285
UniRef50	3,077,464	0.920	0.553	19,087	26,442
UniRef90	6,544,144	0.924	0.561	40,448	46,478
UniRef100	9,865,668	0.922	0.554	66,696	71,895
UniProt	11,009,767	0.923	0.563	66,964	72,199
NCBI NR	10,565,004	0.921	0.554	65,201	70,375

2.Phylogenetic trees

2.1 Enumerating trees and selecting search strategies

# of rooted and unrooted trees: 3 OTUs



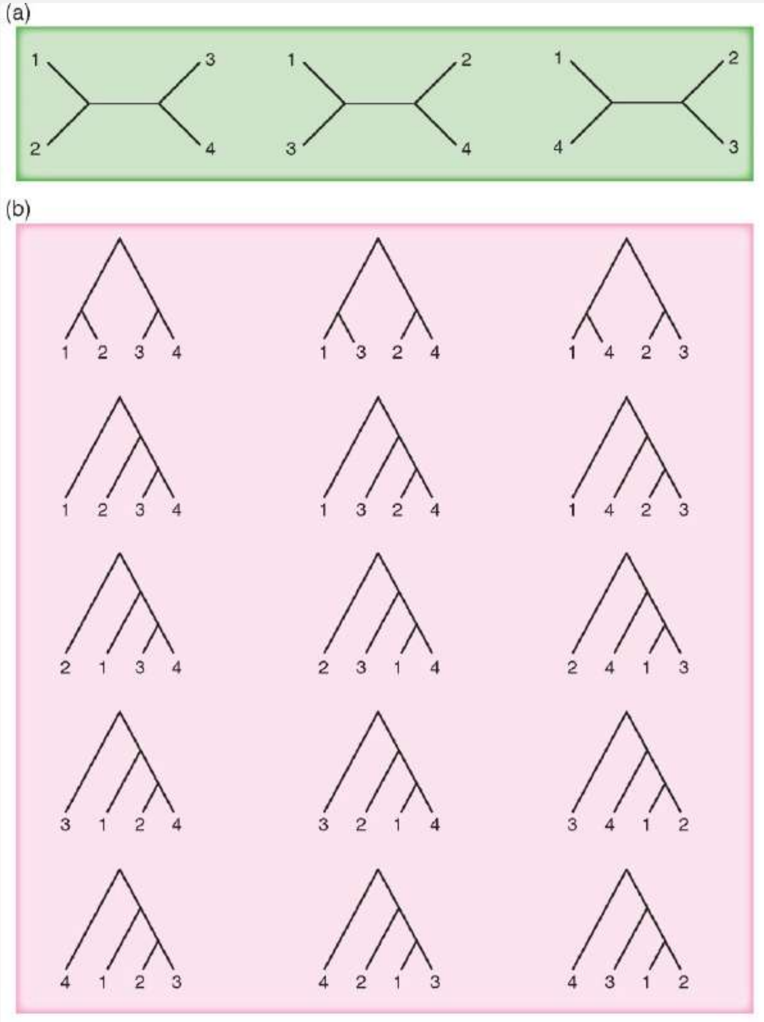
For three operational taxonomic units (OTUs) there is one possible unrooted tree.

Any of the three edges can be selected to form a root.

Three rooted trees are possible.

B&FG 3e Fig. 7.10 Page 264

# of rooted and unrooted trees: 4 OTUs



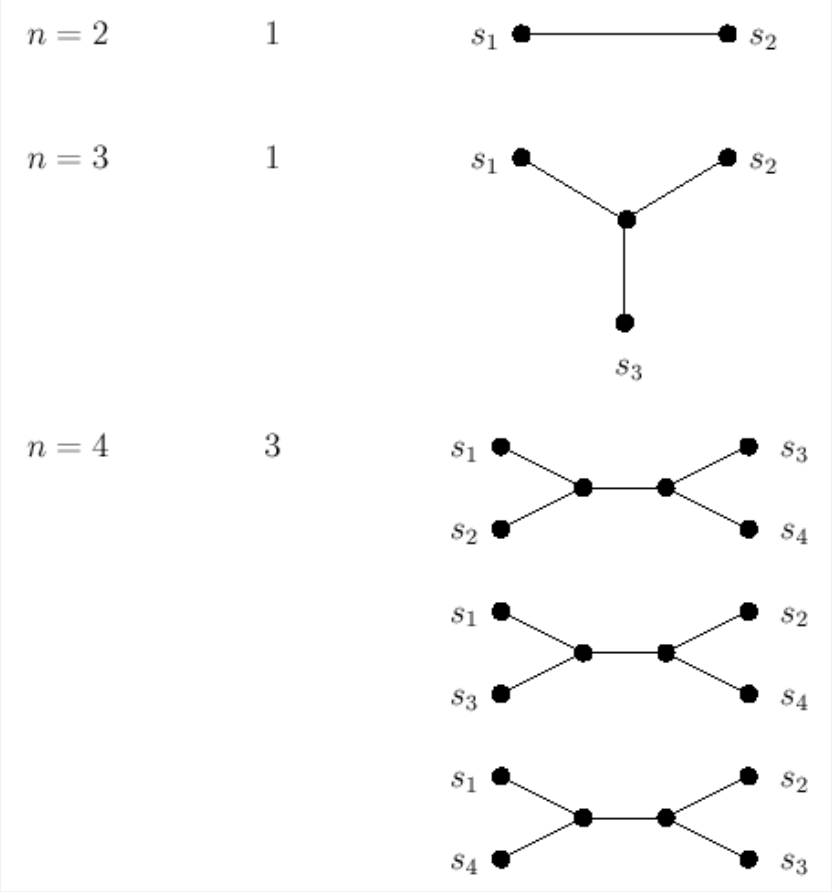
For 4 OTUs there are three possible unrooted trees.

For 4 OTUs there are 15 possible rooted trees.

There is only one of these 15 trees that accurately describes the evolutionary process by which these four sequences evolved.

B&FG 3e Fig. 7.11 Page 265

TU(*k*): the # of unrooted tree for *n* taxa



Let *E*(*k*) denote the # of edges in the unrooted tree for *k* species.

$$E(k) = 2k - 3$$

Let *TU*(*k*) denote the # of unrooted trees for *k* species.

$$\begin{aligned} TU(k) &= TU(k-1)\times E(k-1) \\ &= \big(TU(k-2)\times E(k-2)\big)\times E(k-1) \\ &= \prod_{i=1}^{k-2} E(k-i) \\ &= \prod_{i=1}^{k-2} (2k-2i-3) \\ &= (2k-5)\times\cdots\times 5\times 3\times 1 \end{aligned}$$

Credit by 李家同 計算生物學

TU(*k*) function

$$TU(k) = 1\times3\times5\times\cdots\times(2k-5)$$

$$= \frac{1\times2\times3\times4\times5\times\cdots\times(2k-6)\times(2k-5)}{2\times4\times\cdots\times(2k-6)}$$

$$= \frac{(2k-5)!}{(2\times1)\times(2\times2)\cdots(2\times(k-3))}$$

$$= \frac{(2k-5)!}{(2\times2\times\cdots\times2)\times(1\times2\times\cdots\times(k-3))}$$

$$= \frac{(2k-5)!}{2^{k-3}(k-3)!}$$

k	TU(k)
4	3
5	15
6	105
7	945
8	10395
9	135,135
10	2,027,025
…	…
20	~2x10^{20}

TR(*k*): the # of rooted tree for *k* species

$$TR(k) = TU(k) \times (2k - 3)$$
$$= TU(k) \times E(k)$$
$$= TU(k + 1)$$

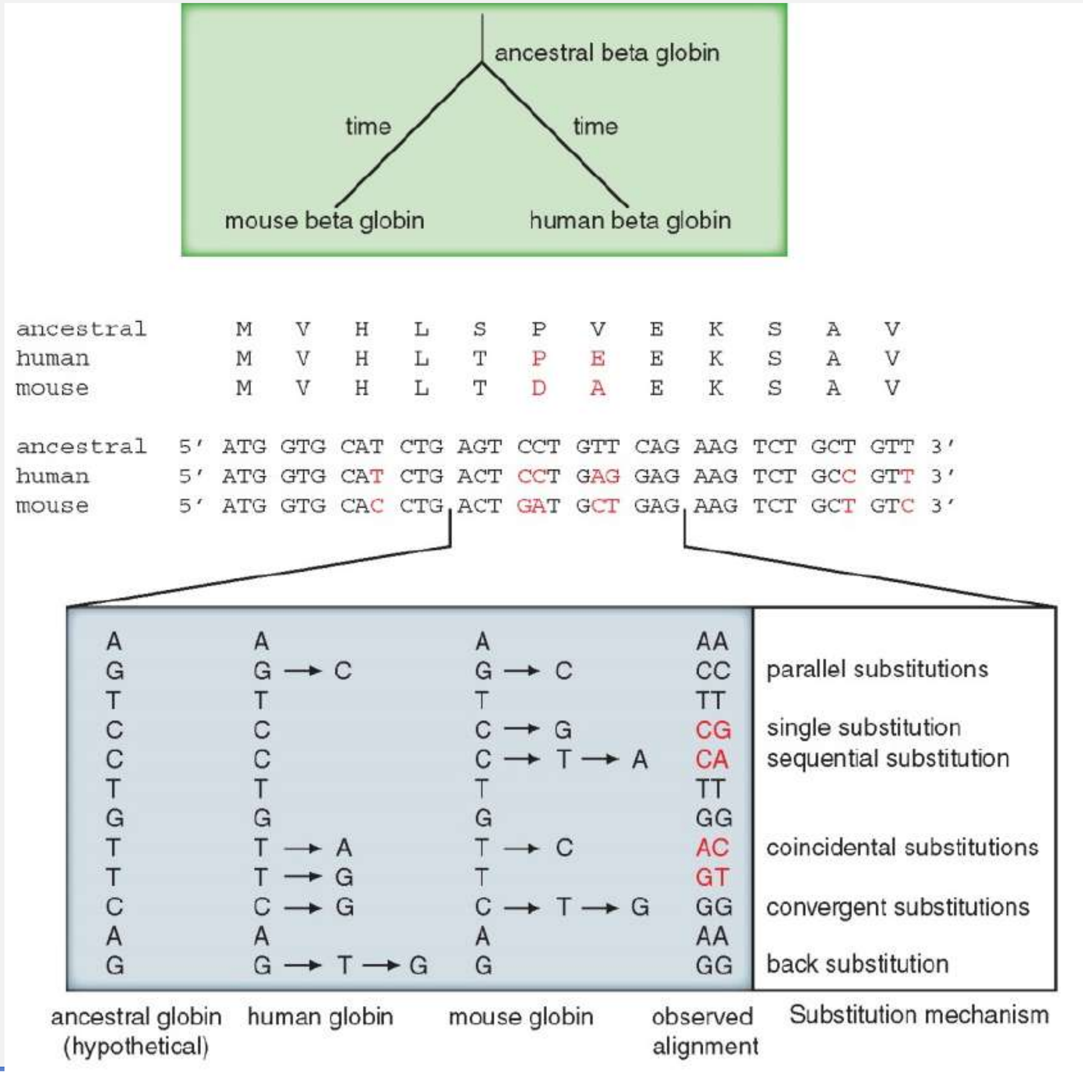
k	***TU*(*k*)**	***TR*(*k*)**
2	1	1
3	1	3
4	3	15
5	15	105
10	2,027,025	34,459,425
20	~2x10^{20}	~8 x 10^{21}

Credit by 李家同 計算生物學

Stage 1: Use of DNA, RNA, or protein

- For phylogeny, DNA can be more informative.
- Some substitutions in a DNA sequence alignment can be directly observed: single nucleotide substitutions, sequential substitutions, coincidental substitutions.
- Additional mutational events can be inferred by analysis of ancestral sequences.

Two sequences (human and mouse) and their common ancestor: we can infer which DNA changes occurred over time



```
ancestral      M   V   H   L   S   P   V   E   K   S   A   V
human          M   V   H   L   T   P   E   E   K   S   A   V
mouse          M   V   H   L   T   D   A   E   K   S   A   V

ancestral   5′ ATG GTG CAT CTG AGT CCT GTT CAG AAG TCT GCT GTT 3′
human       5′ ATG GTG CAT CTG ACT CCT GAG GAG AAG TCT GCC GTT 3′
mouse       5′ ATG GTG CAC CTG ACT GAT GCT GAG AAG TCT GCT GTC 3′
```

protein

DNA

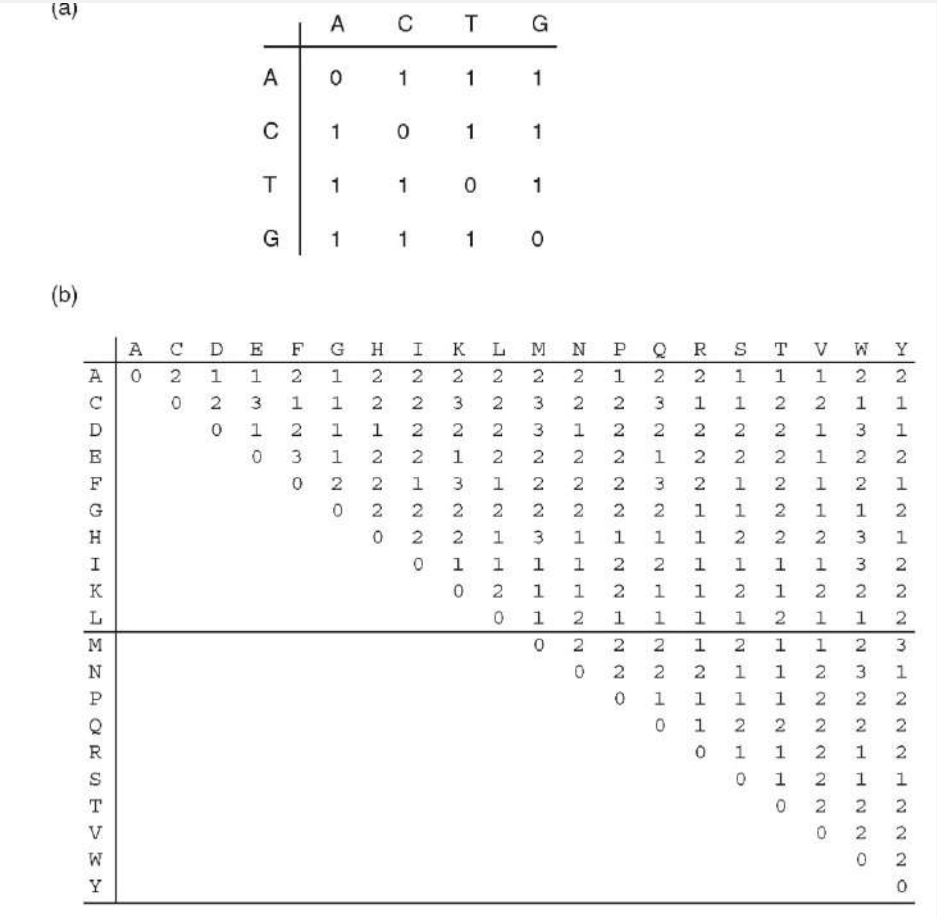
Stage 2: Multiple sequence alignment

1. Confirm that all sequences are homologous
2. Adjust gap creation and extension penalties as needed to optimize the alignment
3. Restrict phylogenetic analysis to regions of the multiple sequence alignment for which data are available for all taxa (delete columns having incomplete data).

Stage 3: models of DNA and amino acid substitution

- The simplest approach to measuring distances between sequences
  - align pairs of sequences
  - count the number of differences.
- For an alignment of length N with n sites at which there are differences, the degree of divergence D is (Hamming distance)
  - $D = n / N$
- But observed differences do not equal genetic distance!
- Genetic distance involves mutations that are not observed directly.

Step matrices: number of steps required to change a character



(a)

	A	C	T	G
A	0	1	1	1
C	1	0	1	1
T	1	1	0	1
G	1	1	1	0

(b)

	A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y
A	0	2	1	1	2	1	2	2	2	2	2	2	1	2	2	1	1	1	2	2
C		0	2	3	1	1	2	2	3	2	3	2	2	3	1	1	2	2	1	1
D			0	1	2	1	1	2	2	2	3	1	2	2	2	2	2	1	3	1
E				0	3	1	2	2	1	2	2	2	2	1	2	2	2	1	2	2
F					0	2	2	1	3	1	2	2	2	3	2	1	2	1	2	1
G						0	2	2	2	2	2	2	2	2	1	1	2	1	1	2
H							0	2	2	1	3	1	1	1	1	2	2	2	3	1
I								0	1	1	1	1	2	2	1	1	1	1	3	2
K									0	2	1	1	2	1	1	2	1	2	2	2
L										0	1	2	1	1	1	1	2	1	1	2
M											0	2	2	2	1	2	1	1	2	3
N												0	2	2	2	1	1	2	3	1
P													0	1	1	1	1	2	2	2
Q														0	1	2	2	2	2	2
R															0	1	1	2	1	2
S																0	1	2	1	1
T																	0	2	2	2
V																		0	2	2
W																			0	2
Y																				0

nucleotide step matrix

amino acid step matrix

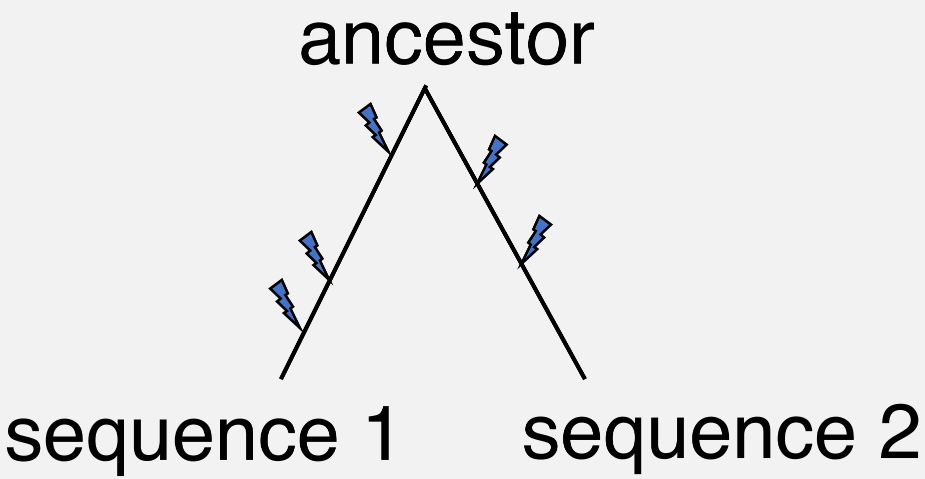
For amino acids, between 1 and 3 nucleotide changes are required to change one residue to another.

B&FG 3e Fig. 7.16 Page 271

Quantification of evolutionary distances

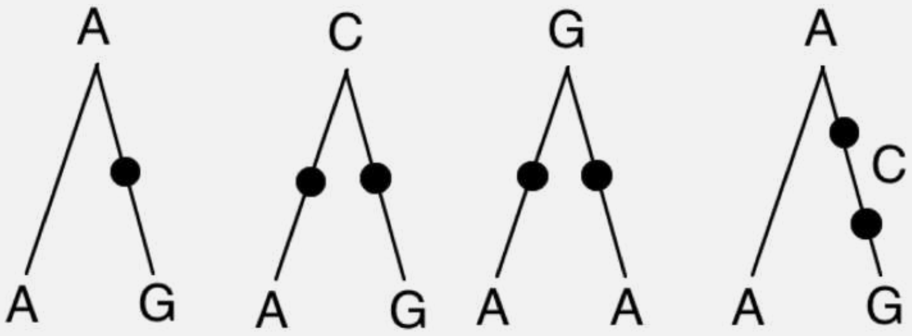
Evolutionary Distances

- They measure the total number of substitutions that occurred on both lineages since divergence from last common ancestor.
- Divided by sequence length.
- Expressed in substitutions / site



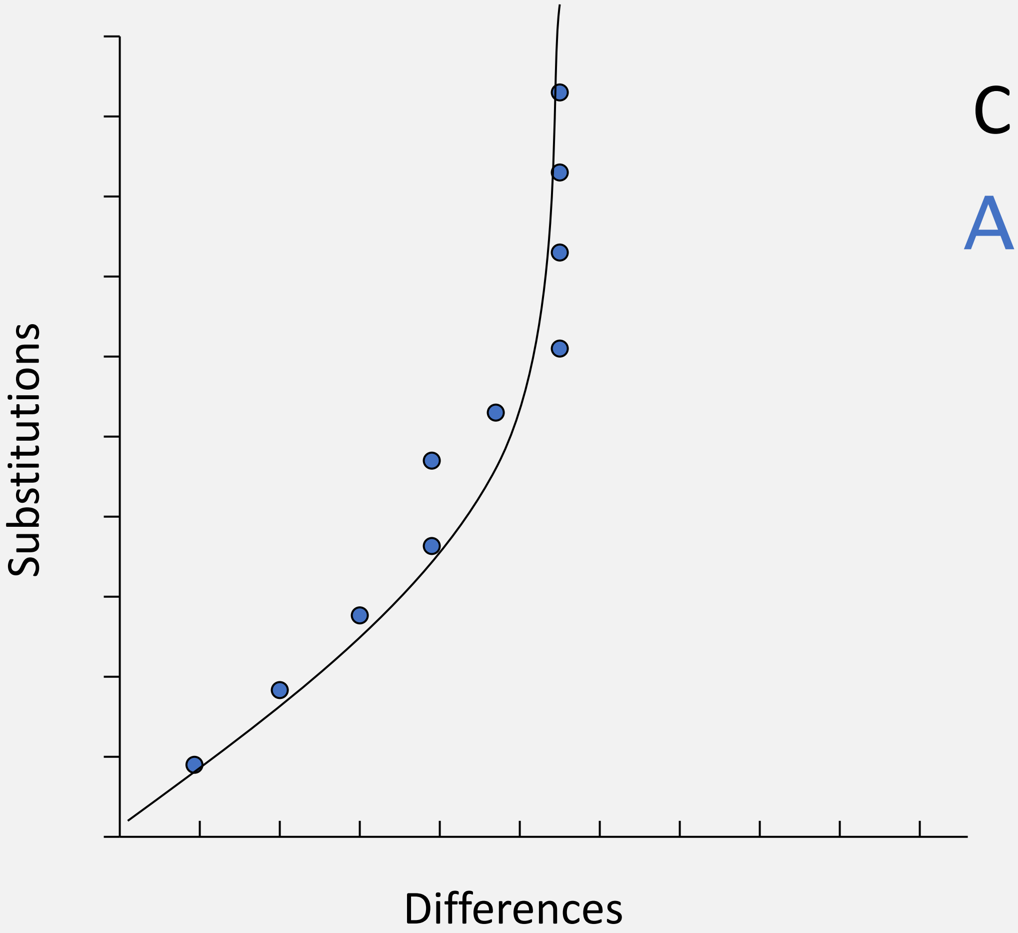
The problem of hidden or multiple changes

- *D* (true evolutionary distance) ≥ fraction of observed differences (*p*)



- $D = p +$ hidden changes
- Through hypotheses about the nature of the residue substitution process, it becomes possible to estimate *D* from observed differences between sequences.

Correcting for multiple substitutions



C T T C C A C

A T A A T A T

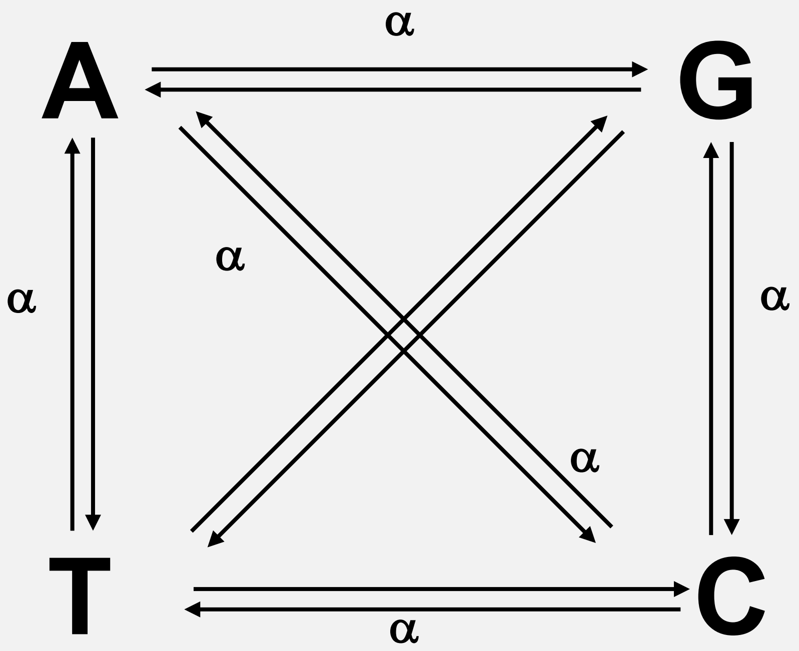
Correcting for multiple substitutions

- Requires a statistical ‘model’ of how the process of substitution works to correct for
- Differences in the rates of different substitution types
  - Jukes and Cantor – all substitutions are treated the same
  - Kimura 2-parameter model – distinguishes between transitions and transversions
- Different frequencies of different nucleotides
  - GC content – the HKY model adds nucleotide frequency parameters to the Kimura 2-parameter model
- Different rates at different sites (*often modelled using a distribution – e.g. Gamma distribution*)

Stage 3: Jukes and Cantor one-parameter model of nucleotide substitution

- This model describes the probability that one nucleotide will change into another. It assumes that each residue is equally likely to change into any other.
- Jukes and Cantor (1969) proposed a corrective formula:

$$D = \left(-\frac{3}{4}\right) ln \left(l - \frac{4}{3}p\right)$$



JC model: $D = \left(-\frac{3}{4}\right) ln\left(1 - \frac{4}{3}p\right)$

- Consider an alignment where 3/60 aligned residues differ
  - The normalized Hamming distance, 3/60 = 0.05.
  - The Jukes-Cantor correction is

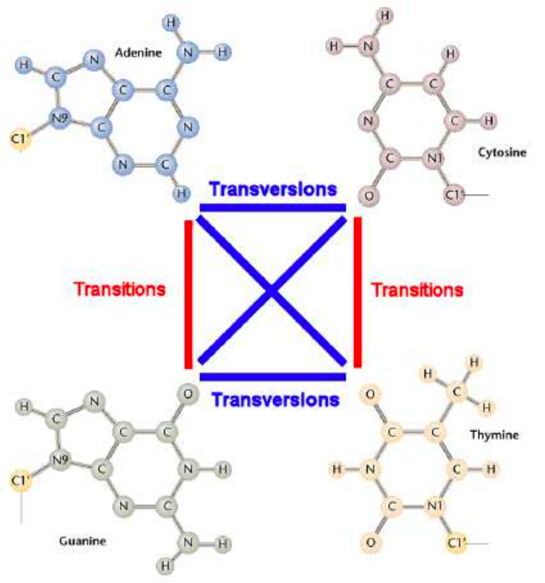
$$D = \left(-\frac{3}{4}\right) ln\left(1 - \frac{4}{3}0.05\right) = 0.052$$

- When 30/60 aligned residues differ, the Jukes-Cantor correction is more substantial:

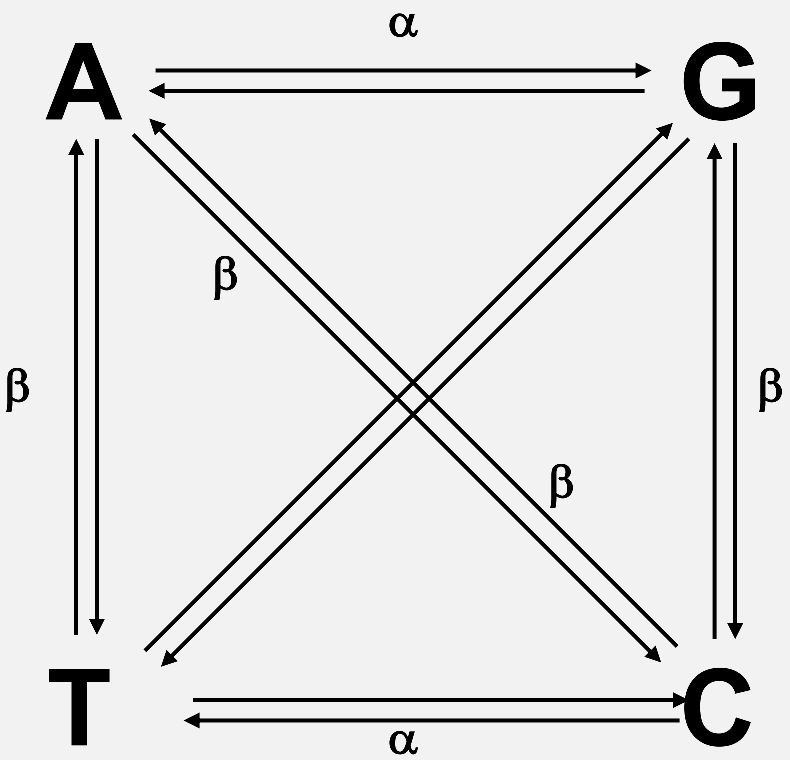
$$D = \left(-\frac{3}{4}\right) ln\left(1 - \frac{4}{3}0.5\right) = 0.82$$

Two DNA substitution mutations

- Transitions: interchanges of two-ring purines or of one-ring pyrimidines : they therefore involve bases of similar shape.
  - A <–> G, C <–> T
- Transversions: interchanges of purine for pyrimidine bases, which therefore involve exchange of one-ring and two-ring structures.
  - A <–> C, A <–> T, G <–> C, G <–> T



Kimura two-parameter model of nucleotide substitution (assumes a ≠ b)



Kimura’s two parameter distance (DNA)

- Hypotheses of the model
  - All sites evolve independently and following the same process.
  - Substitutions occur according to two probabilities
    - Transitions : G <–>A or C <–>T
    - Transversions : other changes
  - The base substitution process is constant in time.
- Quantification of evolutionary distance (*d*) as a function of the fraction of observed differences *p*: transitions, *q*: transversions
  - $d = -\frac{1}{2}\ln\left[(1-2p-q)\sqrt{1-2q}\right]$

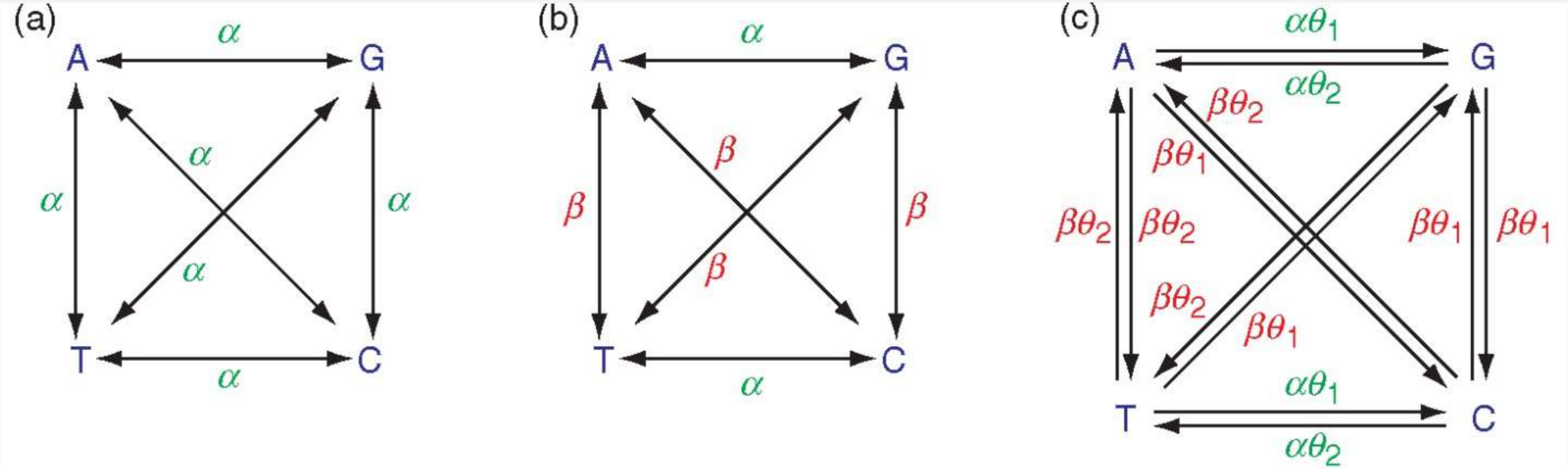
Kimura (1980) J. Mol. Evol. 16:111

There are dozens of models

Jukes-Cantor model

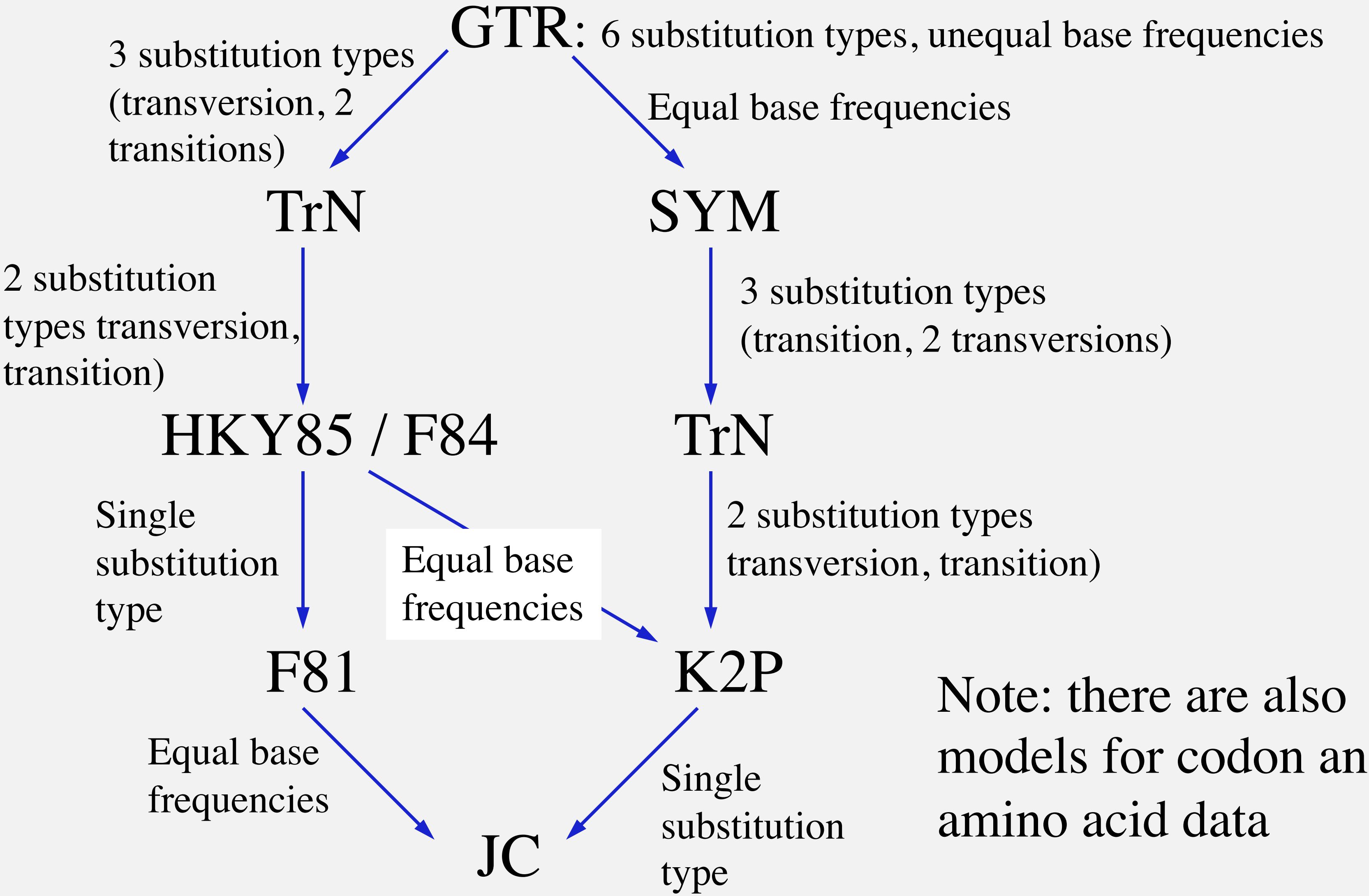
Kimura model

Tamura model



B&FG 3e
Fig. 7.20
Page 277

Substitution model categories



GTR: 6 substitution types, unequal base frequencies

3 substitution types (transversion, 2 transitions)

Equal base frequencies

TrN

SYM

2 substitution types transversion, transition)

3 substitution types (transition, 2 transversions)

HKY85 / F84

TrN

Single substitution type

Equal base frequencies

2 substitution types transversion, transition)

F81

K2P

Equal base frequencies

Single substitution type

JC

Note: there are also models for codon and amino acid data

Stage 4: tree-building methods

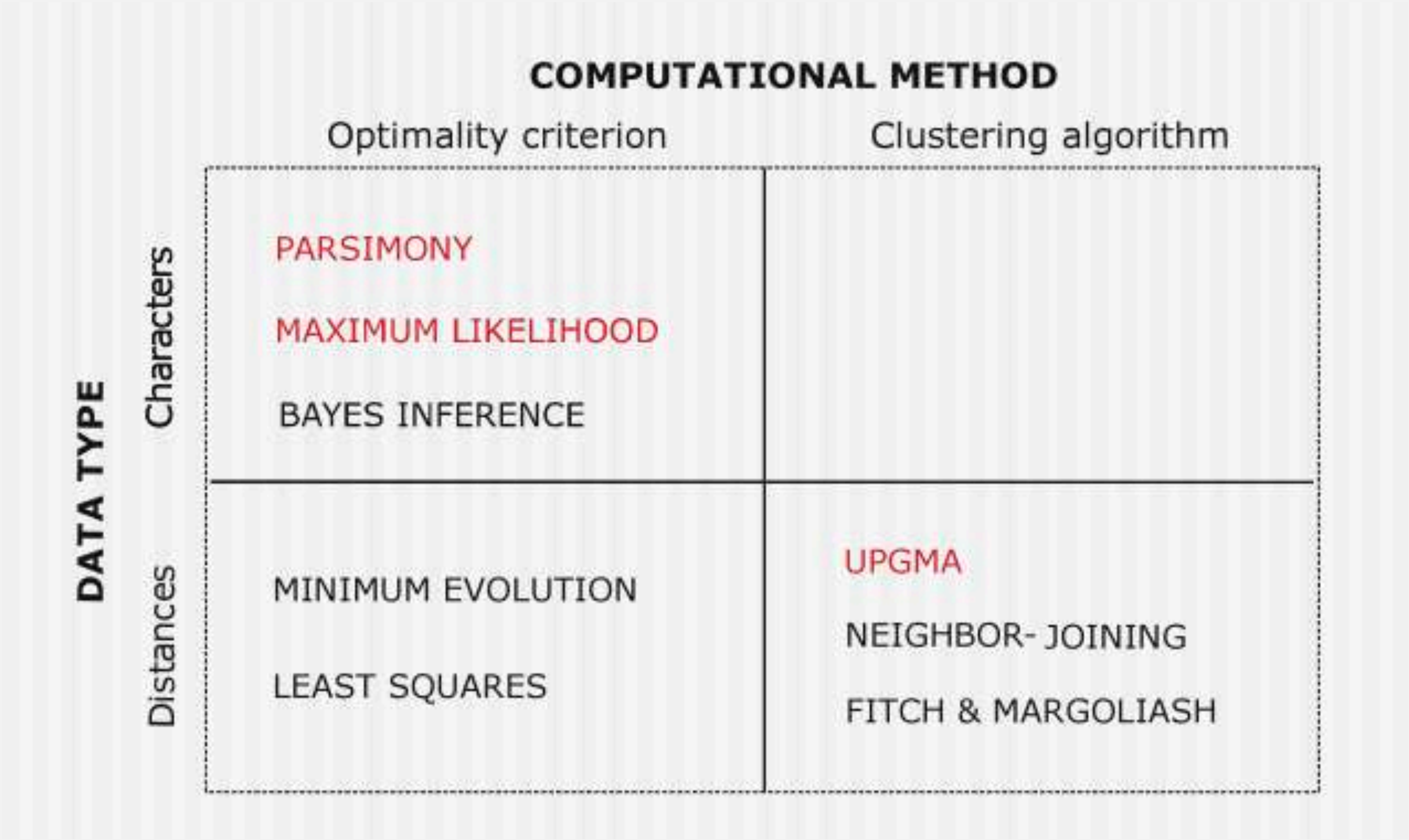
distance-based

maximum parsimony

maximum likelihood

Bayesian methods

Main families of Methods for Phylogenetic reconstruction



COMPUTATIONAL METHOD

DATA TYPE		Optimality criterion	Clustering algorithm
	Characters	PARSIMONY MAXIMUM LIKELIHOOD BAYES INFERENCE	
	Distances	MINIMUM EVOLUTION LEAST SQUARES	UPGMA NEIGHBOR- JOINING FITCH & MARGOLIASH

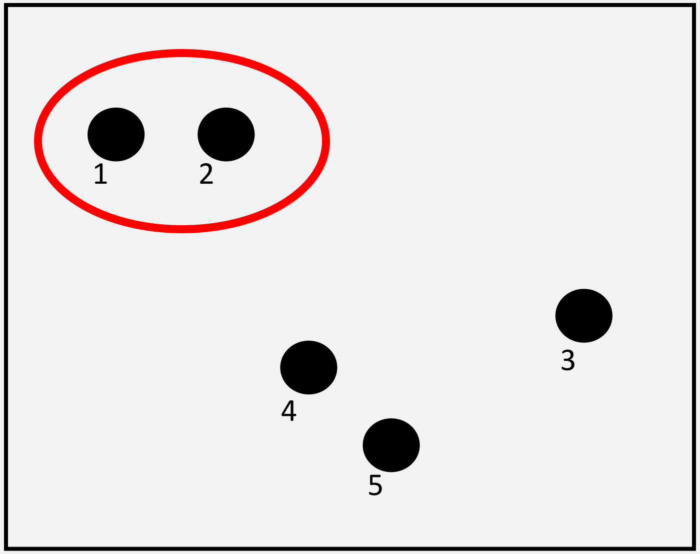
Adapted from Cedric Notredame

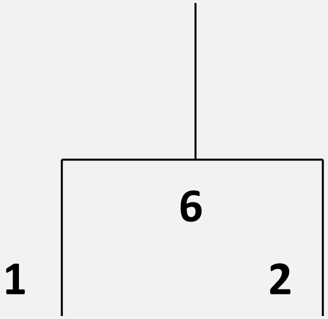
UPGMA

Unweighted Pair-Group Method with Arithmetic Mean

Tree-building methods: UPGMA

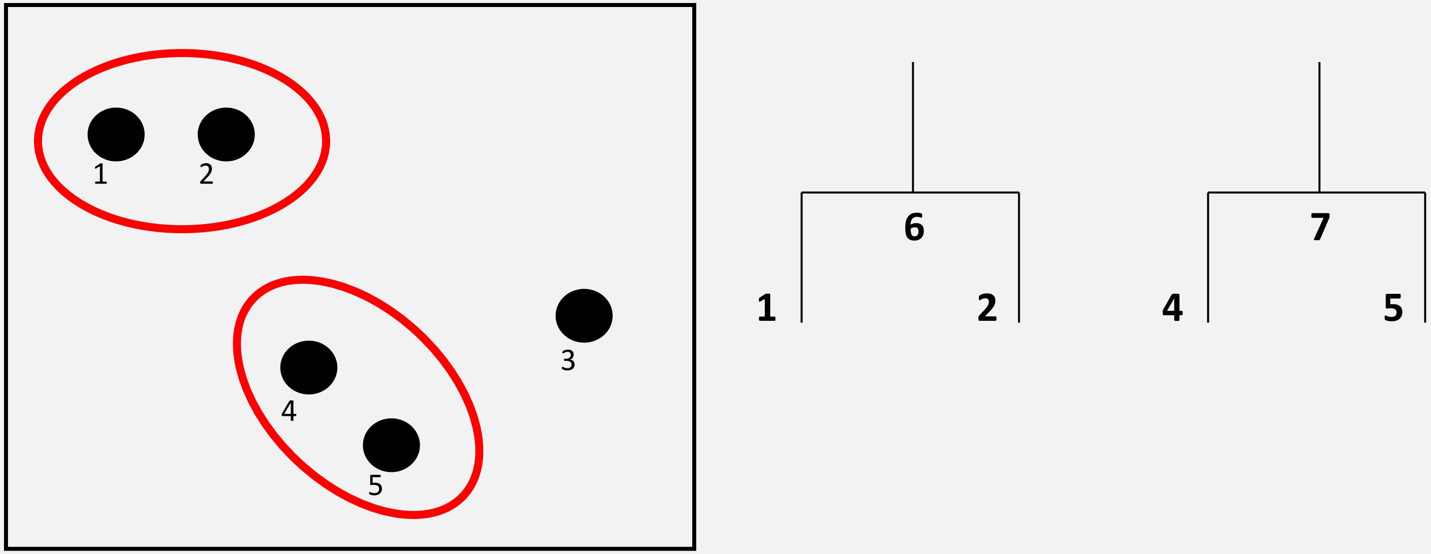
- Step 2: Find the two proteins with the smallest pairwise distance. Cluster them.





Tree-building methods: UPGMA

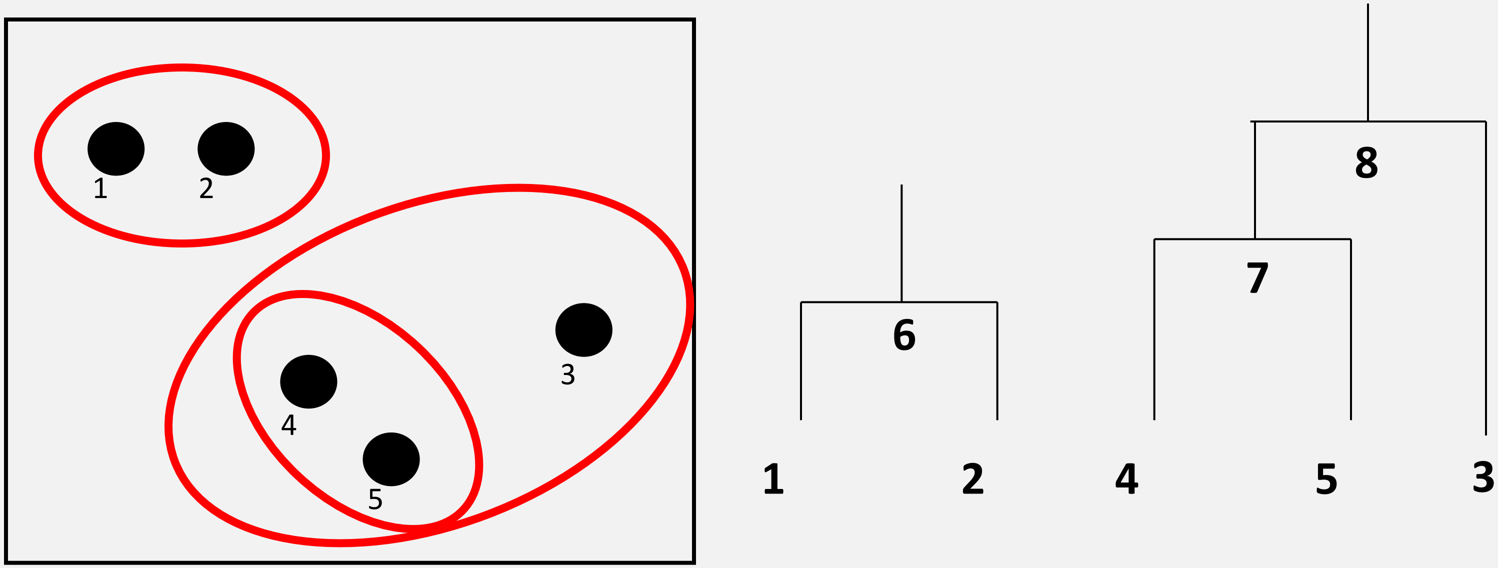
- Step 3: Do it again. Find the next two proteins with the smallest pairwise distance. Cluster them.



B&FG 3eFig. 7.24 Page 284

Tree-building methods: UPGMA

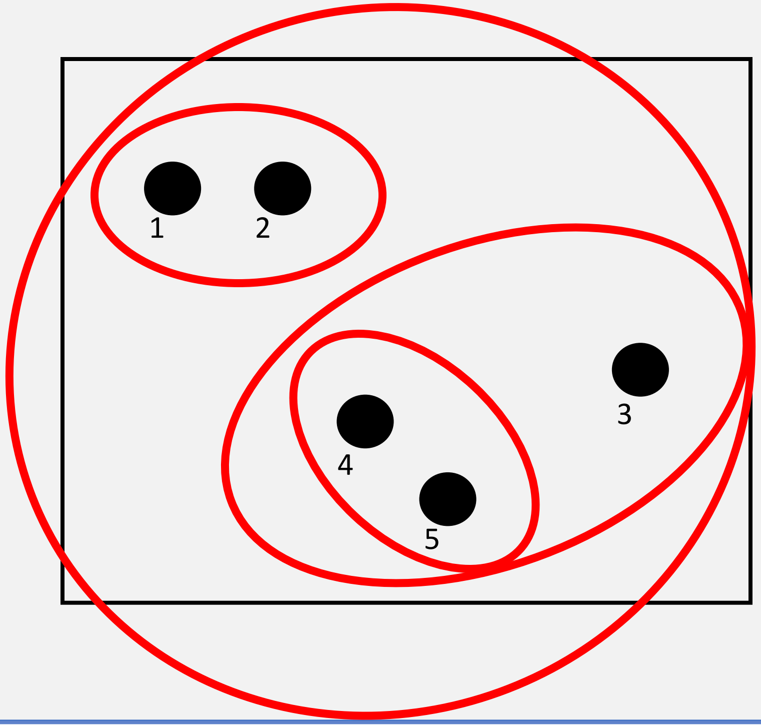
- Step 4: Keep going. Cluster.

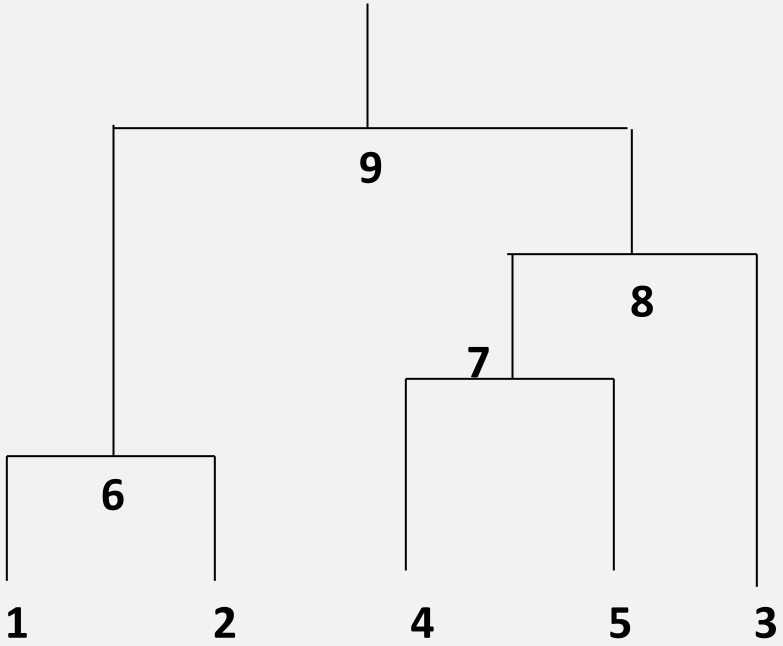


B&FG 3e Fig. 7.24 Page 284

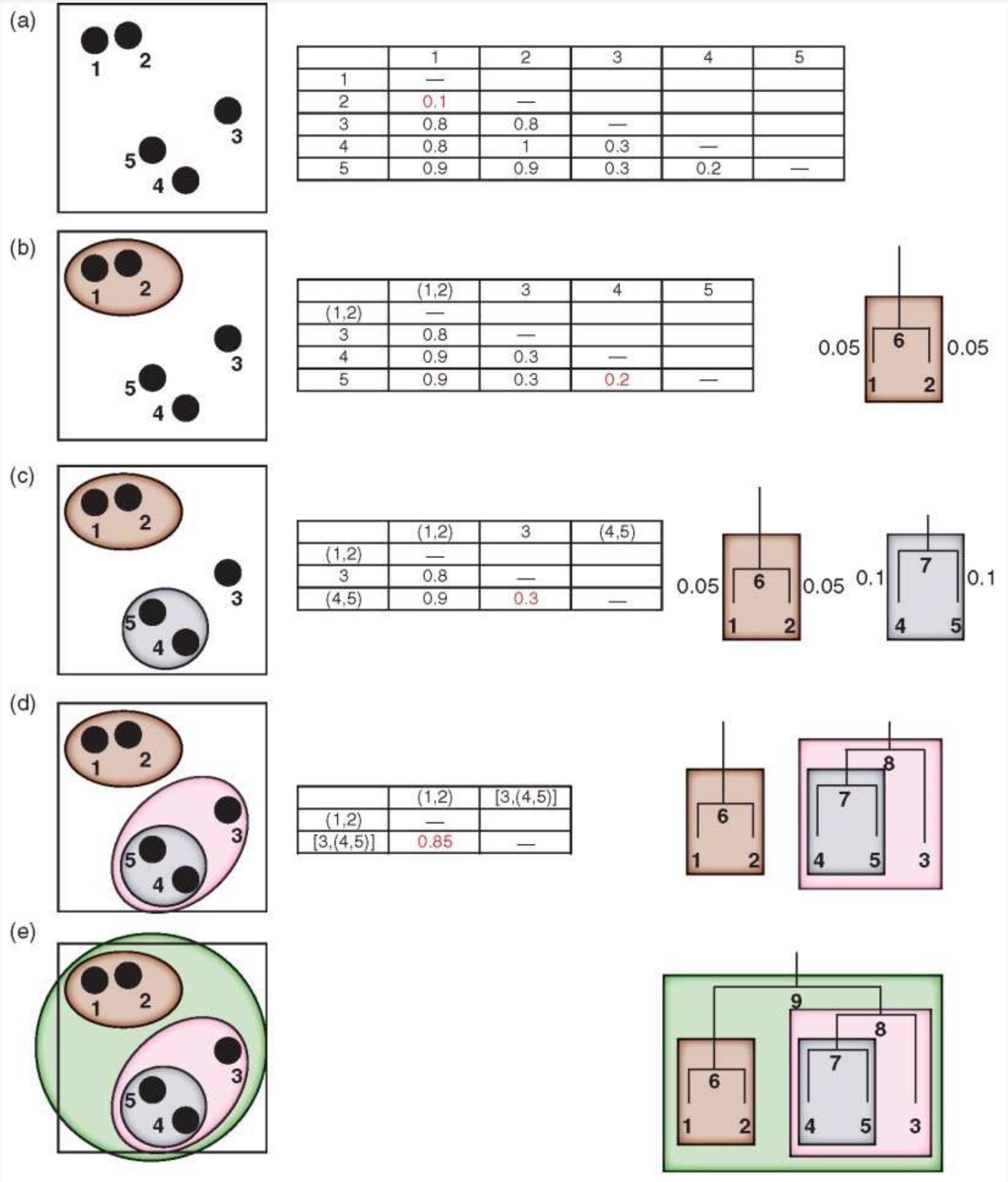
Tree-building methods: UPGMA

- Step 4: Last cluster! This is your tree.





B&FG 3e Fig. 7.24 Page 284



	1	2	3	4	5
1	—				
2	0.1	—			
3	0.8	0.8	—		
4	0.8	1	0.3	—	
5	0.9	0.9	0.3	0.2	—

	(1,2)	3	4	5
(1,2)	—			
3	0.8	—		
4	0.9	0.3	—	
5	0.9	0.3	0.2	—

	(1,2)	3	(4,5)
(1,2)	—		
3	0.8	—	
(4,5)	0.9	0.3	—

	(1,2)	[3,(4,5)]
(1,2)	—	
[3,(4,5)]	0.85	—

B&FG 3e Fig. 7.24 Page 284

Distance-based methods: UPGMA trees

- UPGMA is a simple approach for making trees.
- An UPGMA tree is always rooted.
- An assumption of the algorithm is that the molecular clock is constant for sequences in the tree. If there are unequal substitution rates, the tree may be wrong.
- While UPGMA is simple, it is less accurate than the neighbor-joining approach (described next).

Neighbor-Joining

N Saitou, M Nei, The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, Volume 4, Issue 4, Jul 1987, Pages 406–425

Saitou, N. & Nei, M. *Mol. Biol. Evol.* **4**, 406–425 (1987).

Number 20 on the list is a paper[12] that introduced the “neighbor-joining” method, a fast, efficient way of placing a large number of organisms into a phylogenetic tree according to some measure of evolutionary distance between them, such as genetic variation.

It links related organisms together one pair at a time until a tree is resolved. Physical anthropologist Naruya Saitou helped to devise the technique when he joined Masatoshi Nei’s lab at the University of Texas in Houston in the 1980s to work on human evolution and molecular genetics, two fields that were starting to burst at the seams with information.

Credit by The top 100 papers. Van Noorden R, Maher B, Nuzzo R. *Nature*. 2014 Oct 30;514(7524):550-3

Saitou, N. & Nei, M. *Mol. Biol. Evol.* **4**, 406–425 (1987)

Another field buoyed by the growth in genome sequencing is phylogenetics, the study of evolutionary relationships between species.

“We physical anthropologists were facing kind of the big data of that time,” says Saitou, now at Japan’s National Institute of Genetics in Mishima. The technique made it possible to devise trees from large data sets without eating up computer resources. (And, in a nice cross-fertilization within the top-10, Clustal’s algorithms use the same strategy.)

Credit by The top 100 papers. Van Noorden R, Maher B, Nuzzo R. *Nature*. 2014 Oct 30;514(7524):550-3

Why NJ instead of UPGMA?

In the original CLUSTAL programs, the initial guide trees, used to guide the multiple alignment, were calculated using the UPGMA method.

We now use the Neighbour-Joining method which is more robust against the effects of unequal evolutionary rates in different lineages and which gives better estimates of individual branch lengths.

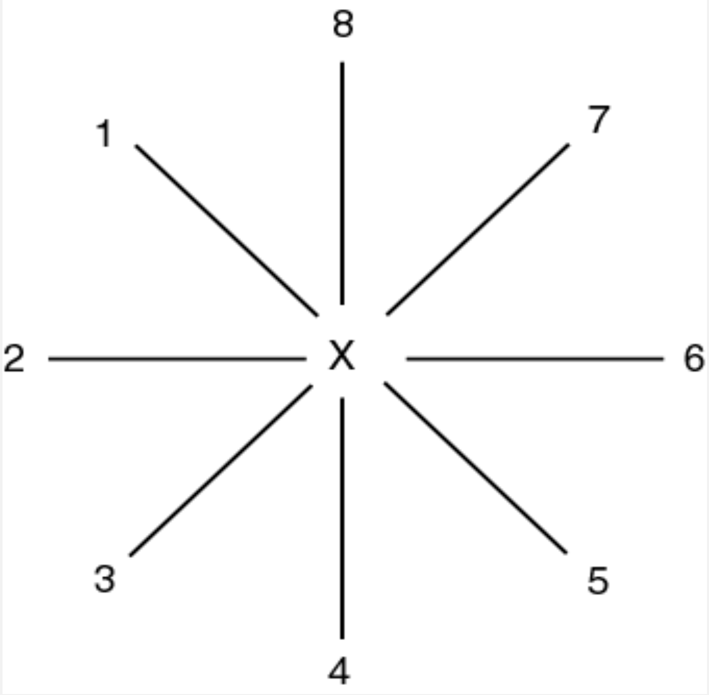
This is useful because it is these branch lengths which are used to derive the sequence weights.

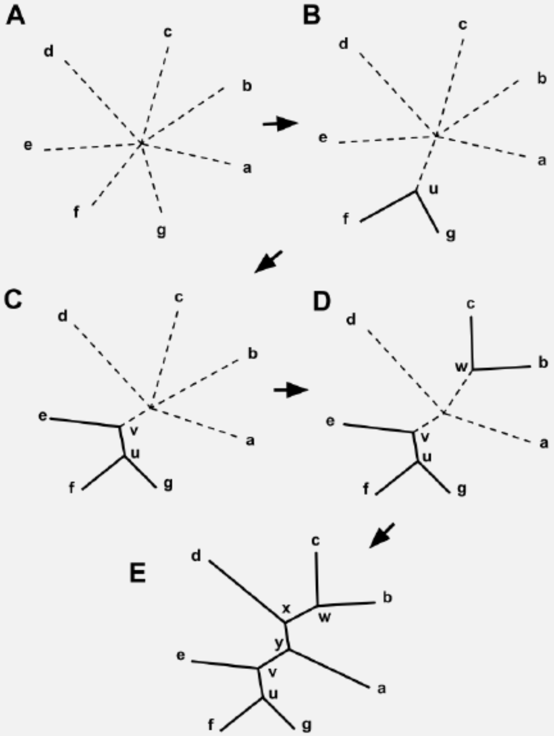
We also allow users to choose between fast approximate alignments or full dynamic programming for the distance calculations used to make the guide tree.

Thompson, J. D., Higgins, D. G. & Gibson, T. J. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* **22,** 4673–80 (1994).

Making trees using neighbor-joining

- useful for making a tree having a large number of taxa.
- Begin by placing all the taxa in a star-like structure.





Credit by https://en.wikipedia.org/wiki/Neighbor_joining

The algorithm

- Based on the current distance matrix calculate the matrix *Q*.
- Find the pair of distinct taxa *i* and *j* for which *Q*(*i*, *j*) has its lowest value. These taxa are joined to a newly created node, which is connected to the central node.
- Calculate the distance from each of the taxa in the pair to this new node.
- Calculate the distance from each of the taxa outside of this pair to the new node.
- Start the algorithm again, replacing the pair of joined neighbors with the new node and using the distances calculated in the previous step.

Credit by https://en.wikipedia.org/wiki/Neighbor_joining

The algorithm

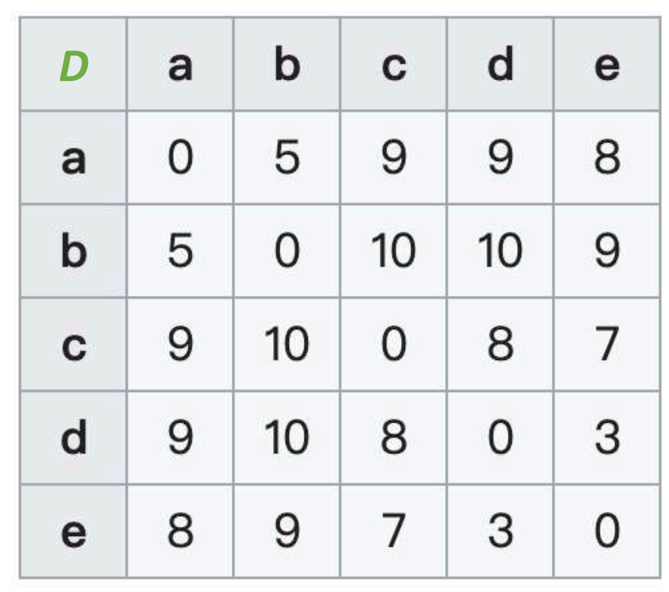
1. Based on the current distance matrix calculate the matrix *Q*.
2. Find the pair of distinct taxa *i* and *j* for which *Q*(*i*, *j*) has its lowest value. These taxa are joined to a newly created node, which is connected to the central node.
3. Calculate the distance from each of the taxa in the pair to this new node.
4. Calculate the distance from each of the taxa outside of this pair to the new node.
5. Start the algorithm again, replacing the pair of joined neighbors with the new node and using the distances calculated in the previous step.

Credit by https://en.wikipedia.org/wiki/Neighbor_joining

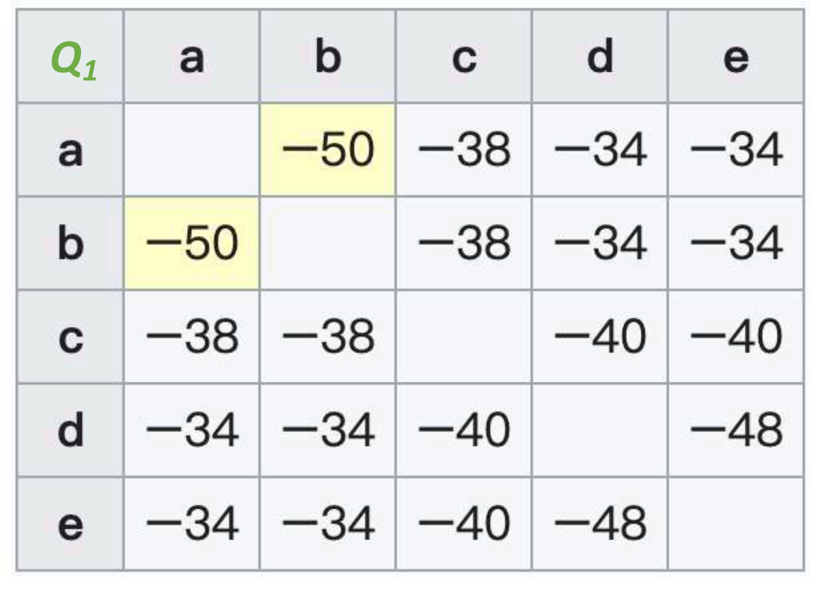
The matrix Q

1. Based on the current distance matrix D calculate the matrix Q.

- $Q(i,j) = (n-2)d(i,j) - \sum_{k=1}^{n} d(i,k) - \sum_{k=1}^{n} d(n,k)$



D	a	b	c	d	e
a	0	5	9	9	8
b	5	0	10	10	9
c	9	10	0	8	7
d	9	10	8	0	3
e	8	9	7	3	0



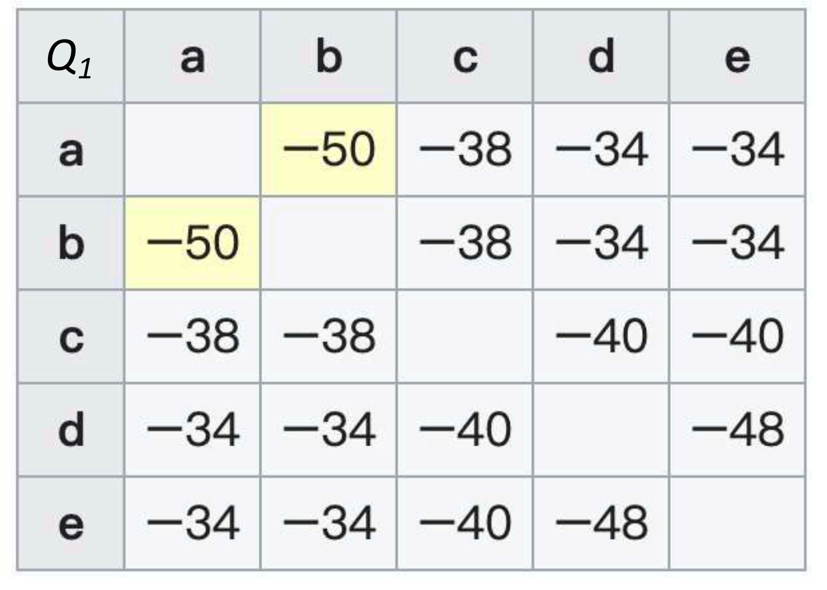
Q_1	a	b	c	d	e
a		−50	−38	−34	−34
b	−50		−38	−34	−34
c	−38	−38		−40	−40
d	−34	−34	−40		−48
e	−34	−34	−40	−48	

Credit by https://en.wikipedia.org/wiki/Neighbor_joining

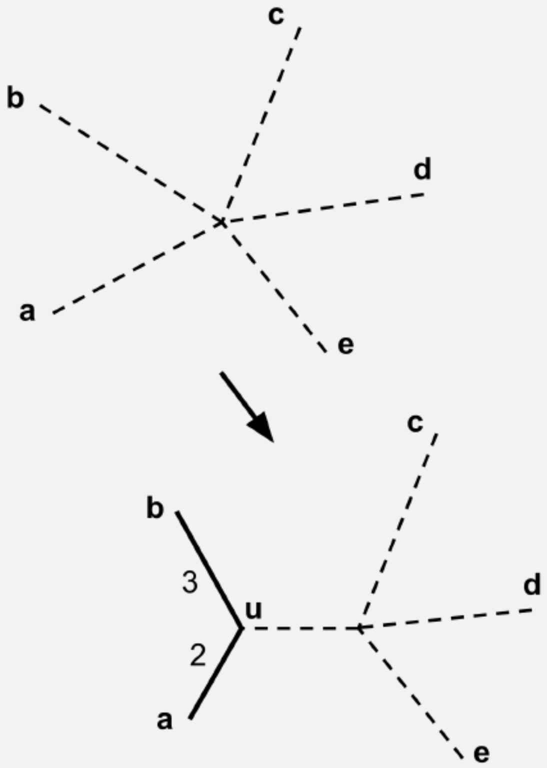
Join two nodes

2. Find the pair of distinct taxa *i* and *j* for which *Q*(*i*, *j*) has its lowest value. These taxa are joined to a newly created node, which is connected to the central node.

- Merge nodes *a* and *b* into *u*



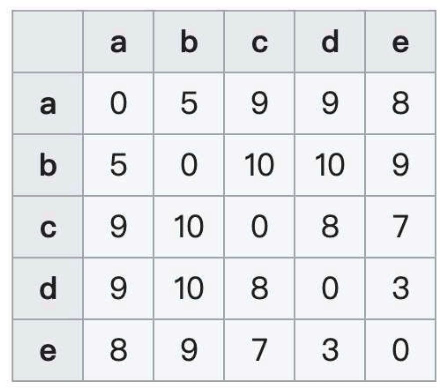
Q_1	a	b	c	d	e
a		−50	−38	−34	−34
b	−50		−38	−34	−34
c	−38	−38		−40	−40
d	−34	−34	−40		−48
e	−34	−34	−40	−48	



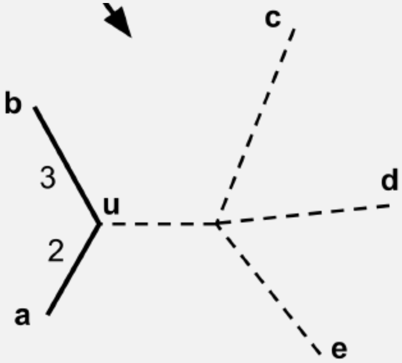
Credit by https://en.wikipedia.org/wiki/Neighbor_joining

Distance from the new node

- 3. Calculate the distance from each of the taxa in the pair to this new node. i.e, Merge nodes *f* and *g* into *u*
  - $\delta(f,u) = \frac{1}{2}d(f,g) + \frac{1}{2(n-2)}[\sum_{k=1}^{n} d(f,k) - \sum_{k=1}^{n} d(g,k)]$
  - $\delta(g,u) = d(f,g) - \delta(f,u)$
- Example: Merge nodes *a* and *b* into *u*
  - $\delta(a,u) = \frac{1}{2}d(a,b) + \frac{1}{2(5-2)}\left[\sum_{k=1}^{5} d(a,k) - \sum_{k=1}^{5} d(b,k)\right] = \frac{5}{2} + \frac{31-34}{6} = 2$
  - $\delta(b,u) = d(a,b) - \delta(a,u) = 5 - 2 = 3$



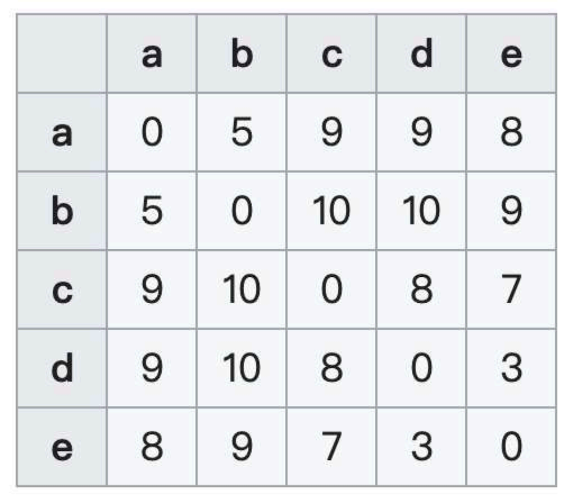
	a	b	c	d	e
a	0	5	9	9	8
b	5	0	10	10	9
c	9	10	0	8	7
d	9	10	8	0	3
e	8	9	7	3	0



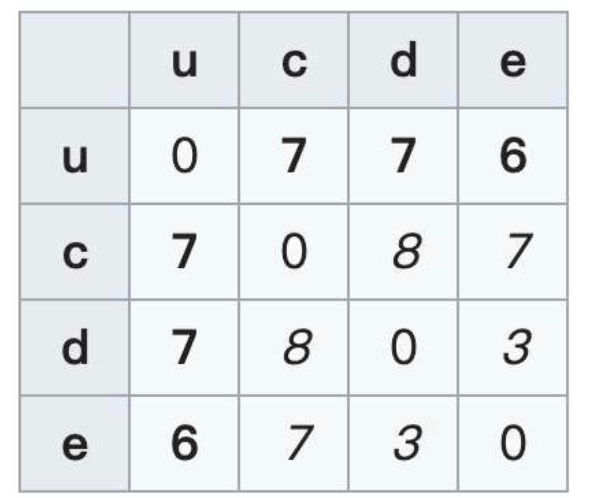
Credit by https://en.wikipedia.org/wiki/Neighbor_joining

Distance of the other taxa from the new node

- 4. Calculate the distance from each of the taxa outside of this pair to the new node.
- $d(u,k) = \frac{1}{2}[d(f,k) + d(g,k) - d(f,g)]$
- Example: Merge nodes *a* and *b* into *u*
  - $d(u,c) = \frac{1}{2}[d(a,c) + d(b,c) - d(a,b)] = \frac{9+10-5}{2} = 7$
  - $d(u,d) = \frac{1}{2}[d(a,d) + d(b,d) - d(a,b)] = \frac{9+10-5}{2} = 7$
  - $d(u,e) = \frac{1}{2}[d(a,e) + d(b,e) - d(a,b)] = \frac{8+9-5}{2} = 6$



	a	b	c	d	e
a	0	5	9	9	8
b	5	0	10	10	9
c	9	10	0	8	7
d	9	10	8	0	3
e	8	9	7	3	0

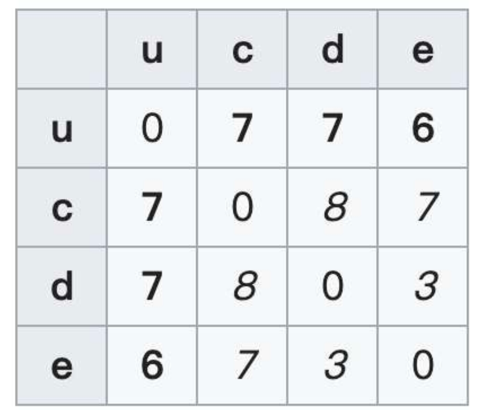


	u	c	d	e
u	0	**7**	**7**	**6**
c	**7**	0	*8*	*7*
d	**7**	*8*	0	*3*
e	**6**	*7*	*3*	0

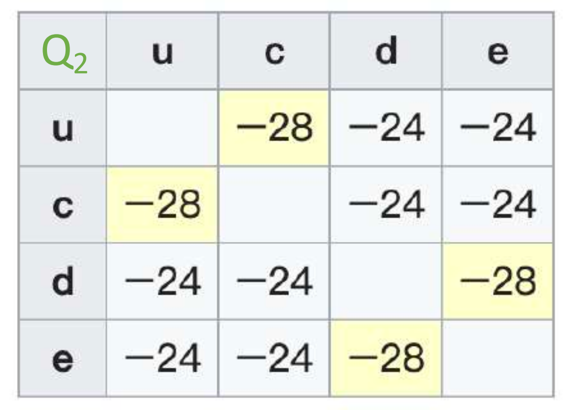
Credit by https://en.wikipedia.org/wiki/Neighbor_joining

Repeat

5. Start the algorithm again, replacing the pair of joined neighbors with the new node and using the distances calculated in the previous step.

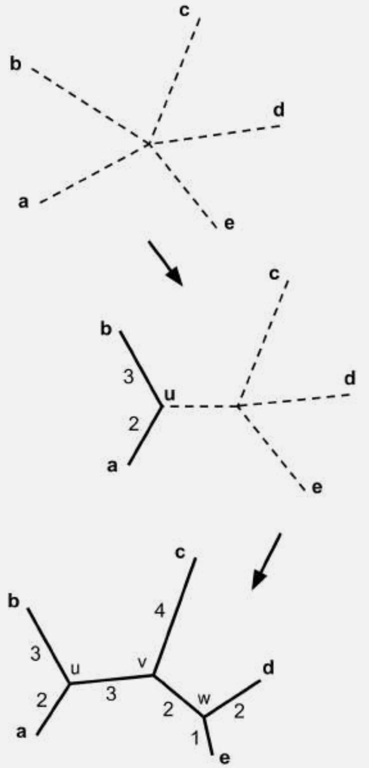


	u	c	d	e
u	0	7	7	6
c	7	0	8	7
d	7	8	0	3
e	6	7	3	0



Q_2	u	c	d	e
u		−28	−24	−24
c	−28		−24	−24
d	−24	−24		−28
e	−24	−24	−28	

…



Credit by https://en.wikipedia.org/wiki/Neighbor_joining

Maximum Parsimony Method

Tree-building methods: character based

- Rather than pairwise distances between proteins, evaluate the aligned columns of amino acid residues (characters).
- Tree-building methods based on characters include
  - maximum parsimony
  - maximum likelihood

Maximum Parsimony (MP)

- Find the tree with the shortest branch lengths possible. Thus we seek the most parsimonious (“simple”) tree.
- Identify informative sites.
  - Constant characters are not parsimony-informative.
- Construct trees, counting the number of changes required to create each tree.
  - <= 12 taxa : evaluate all possible trees exhaustively
  - >12 taxa : perform a heuristic search.
- Select the shortest tree (or trees).

An example of tree-building using MP

- Consider these four taxa

AAG

AAA

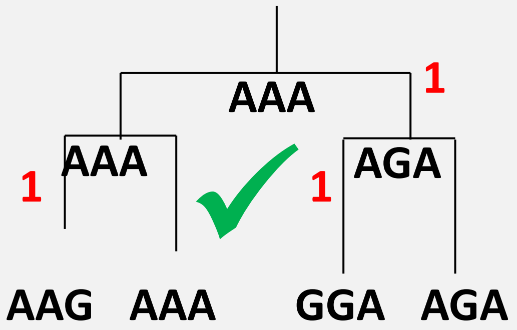
GGA

AGA

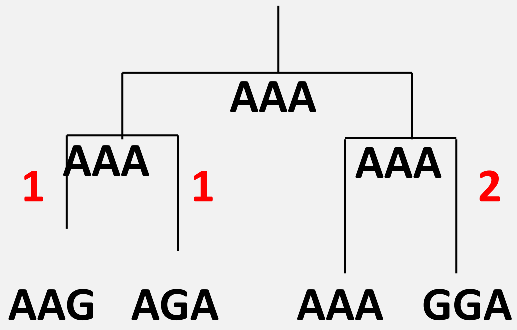
- How might they have evolved from a common ancestor such as AAA?

MP

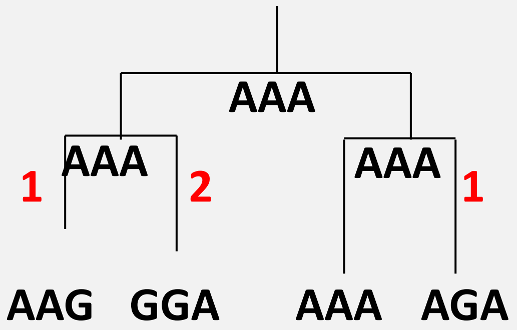
- Choose the tree(s) with the lowest cost (shortest branch lengths)



Cost = 3



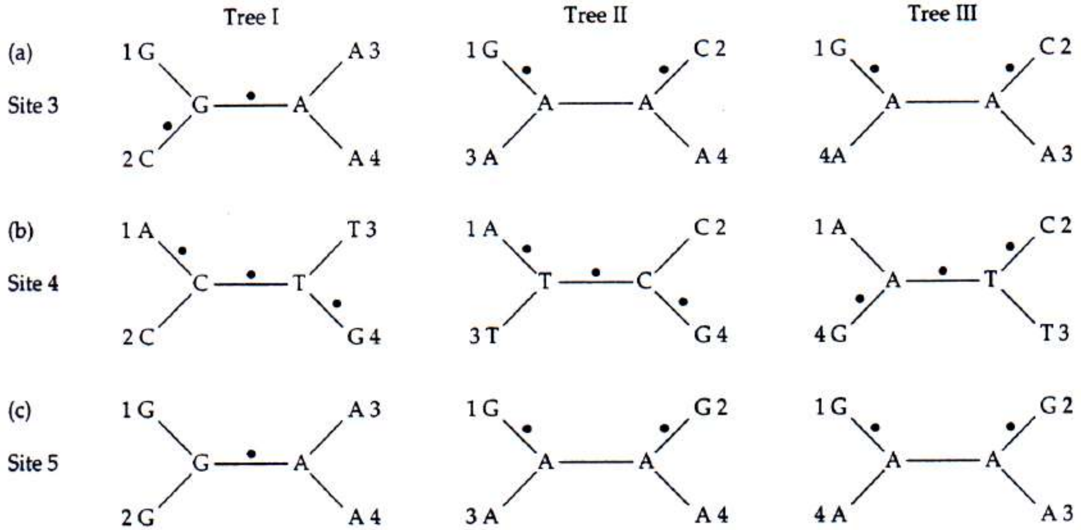
Cost = 4



Cost = 4

Select the tree supported by the largest number of Informative Site

Sequence	Site 1	2	3	4	5	6	7	8	9
1	A	A	G	A	G	T	G	C	A
2	A	G	C	C	G	T	G	C	G
3	A	G	A	T	A	T	C	C	A
4	A	G	A	G	A	T	C	C	G
					*		*		*



Site 5, 7 and 9 are *informative site*

For site5:

- Tree II and III require 2 changes
- Tree I requires 1 change

Credit by 李家同 計算生物學

Maximum Likelihood Method

Making trees using maximum likelihood

- An alternative to maximum parsimony.
- What are the tree topology and branch lengths that have the greatest likelihood of producing the observed data set?
- ML is implemented in the TREE-PUZZLE program, as well as MEGA5, PAUP and PHYLIP.

Likelihood

- Given some data (*D*) a decision must be made about an adequate explanation (*H*, hypothesis)
  - *D*: alignment
  - *H*: Model of evolution, tree topology, branch lengths, parameters of the model
- $L = \Pr(D \mid H)$
  - Each *H* will have a certain probability of producing the data

Likelihood vs Probability

- https://youtu.be/pYxNSUDSFH4
- The likelihood function **!**= the probability of a hypothesis being correct!
- The likelihood function is defined in terms of probability of producing the observed events not of the unknown parameters
- Thus: the probability of observing the data has nothing to do with the probability that the underlying model is correct.

Maximum Likelihood

- https://youtu.be/XepXtl9YKwc
- Given some data (*D*) a decision must be made about an adequate explanation (*H*, hypothesis)
- $L = \Pr(D \mid H)$
  - Each *H* will have a certain probability of producing the data
- The best *H* is that of the greatest *P*

Coin Example

- Data: flipping coins and counting the number of times "heads" appear
  - You throw the coin twice and observe "heads" both times.
- Hypotheses : You might have two hypotheses to explain these data.
  - H_1, the coin is normal: *p* = 0.5, of appearing head.
  - H_2, the coin is rigged with an 80% chance of getting a head , *p* = 0.8.
- What is the likelihood of H_1?
- What is the likelihood of H_2?

Likelihood of the coin example

- The probability of observing "heads" in each of two flips
  - under H1, L(data|H1) = 0.5 x 0.5 = 0.25
  - under H2, L(data|H2) = 0.8 x 0.8 = 0.64
- Since the probability of observing the data under H2 is greater than under H1, you might argue that the "rigged" coin hypothesis is the more likely.

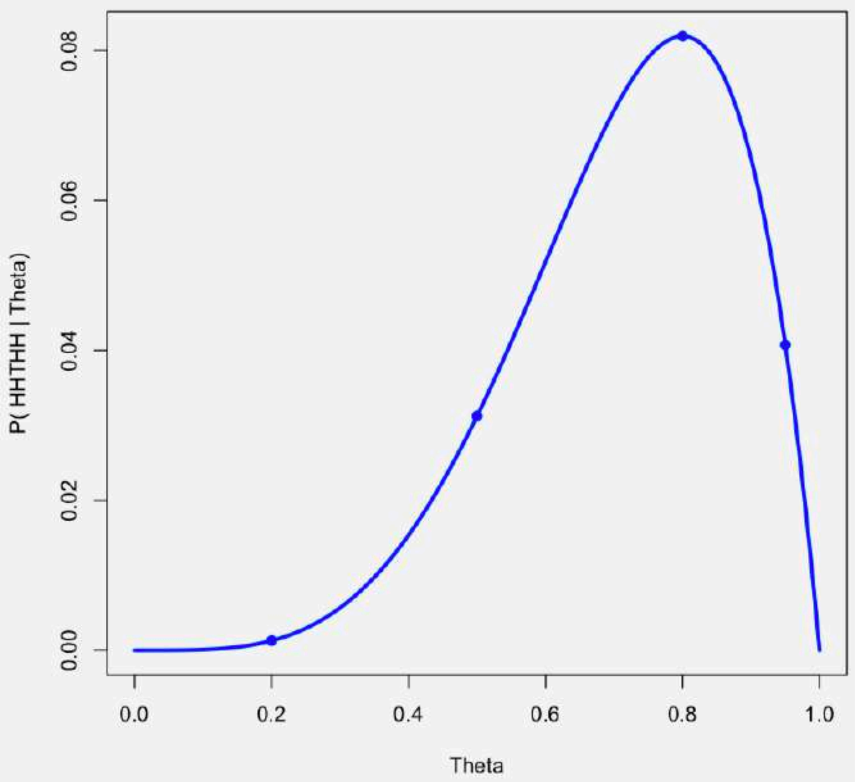
Parameter Estimation

- Assuming sample x_1, x_2, ..., x_n is from a parametric distribution $f(x|\theta)$, estimate θ.
  - Given sample HHTHH of (possibly biased) coin flips, estimate θ = probability of Heads
  - Pr(HHTHH | .6) > Pr(HHTHH | .5), event HHTHH is *more likely* when θ = .6 than θ = .5
  - And what θ make HHTHH *most* likely?

Likelihood Function

- Probability of HHTHH, given θ:

θ	$\Theta^4(1-\theta)$
0.20	0.0013
0.50	0.0313
0.80	0.0819
0.95	0.0407



Maximum Likelihood Parameter Estimation

- As a function of θ, what θ maximizes the likelihood of the data actually observed by taking derivative of L (Pr) with respect to θ

$$\frac{dL}{d\theta} = 4\theta^3 - 5\theta^4 = \theta^3(4 - 5\theta)$$

  - equating to 0, and solving

$$\frac{dL}{d\theta} = 4\theta^3 - 5\theta^4 = \theta^3(4 - 5\theta) = 0 \rightarrow \theta = \frac{4}{5}$$

- More easily, likelihood are often maximized by maximizing their logarithm

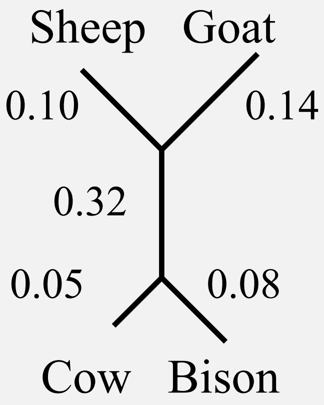
$$\ln L = 4 \ln \theta + \ln(1 - \theta)$$

  - whose derivative is

$$\frac{d \ln L}{d\theta} = \frac{4}{\theta} - \frac{1}{1 - \theta} = 0 \rightarrow \theta = \frac{4}{5}$$

First use in phylogenetics

- Cavalli-Sforza and Edwards (1967) for gene frequency data
- Felsenstein (1981) for DNA sequences
- In phylogenetics, the hypothesis is
  - a tree topology
  - its branch-lengths
  - a model under which the data evolved



Branch-lengths as expected numbers of substitutions per site

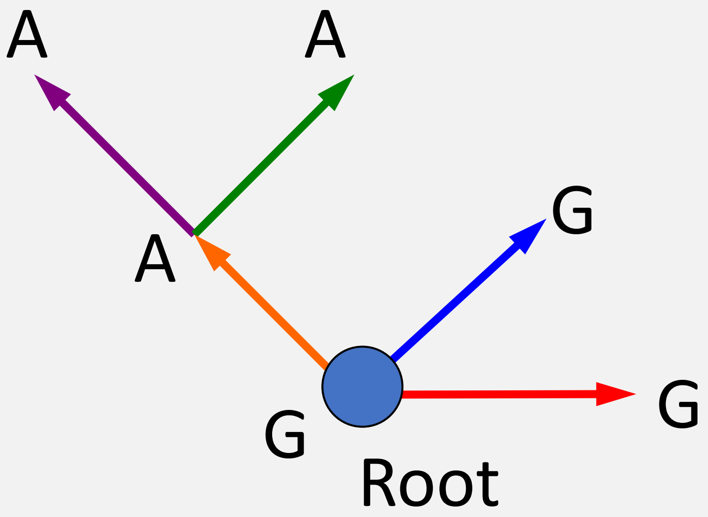
Maximum Likelihood Method(con’t)

- *s* homologous sequences each with *N* nucleotides
- X_k=(X$_{1k}$,…,X_{sk}) the nucleotide configuration at *k*th site
- The likelihood function of tree *T* at the *k*th site
- The likelihood function for the entire sequence for tree *T*

$$L\left(\theta_1, \ldots, \theta_\eta \middle| X_1, \ldots, X_N, T\right) = \prod_{k=1}^{N} f(X_k|\theta, T)$$

An example

- The model is reversible, ie. p(A→G) = p(A→G), so the root can be placed at any node
- Pattern probability = $p(G \rightarrow G) \times p(G \rightarrow G) \times p(G \rightarrow A) \times p(A \rightarrow A) \times p(A \rightarrow A)$



Credit by Joe Felsenstein, Maximum Likelihood and model selection

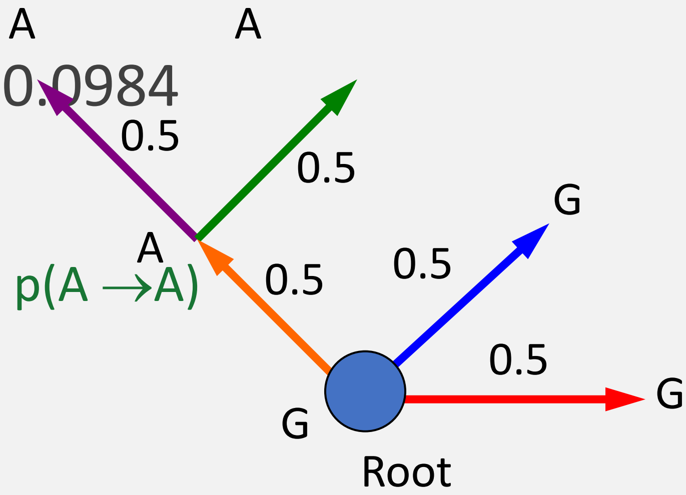
Site pattern probability

- Under the simple Jukes-Cantor model, all base frequencies=0.25, all substitutions equally probable.
- $P_{ij} = \begin{cases} 0.25 + 0.75e^{-b}, i = j \\ 0.25 - 0.75e^{-b}, i \neq j \end{cases}$, where b is branch−length (subs/site)
- Where *b* =0.5, P_{ij} (i=j) = 0.7049, P_{ij}(i≠j) = 0.0984
- Site pattern probability

  = p(G →G) × p(G →G) × p(G →A) × p(A →A) × p(A →A)

  = 0.7049 × 0.7049 × 0.0984 × 0.7049 × 0.7049

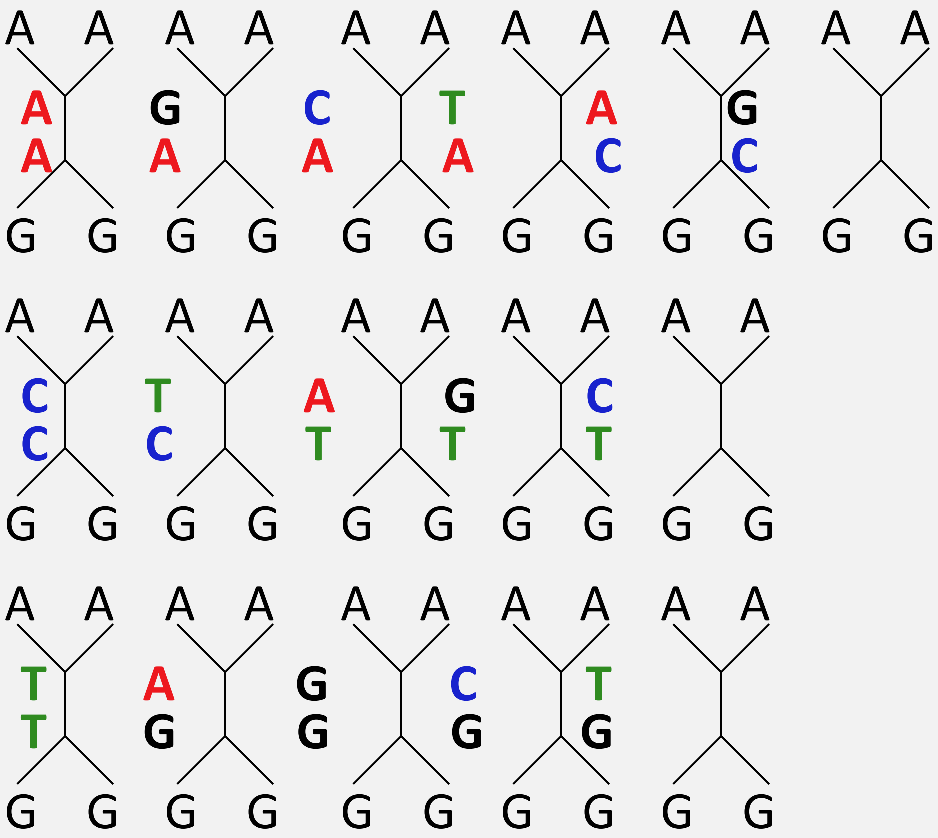
  = 0.0243



Credit by Joe Felsenstein, Maximum Likelihood and model selection

The likelihood of a tree

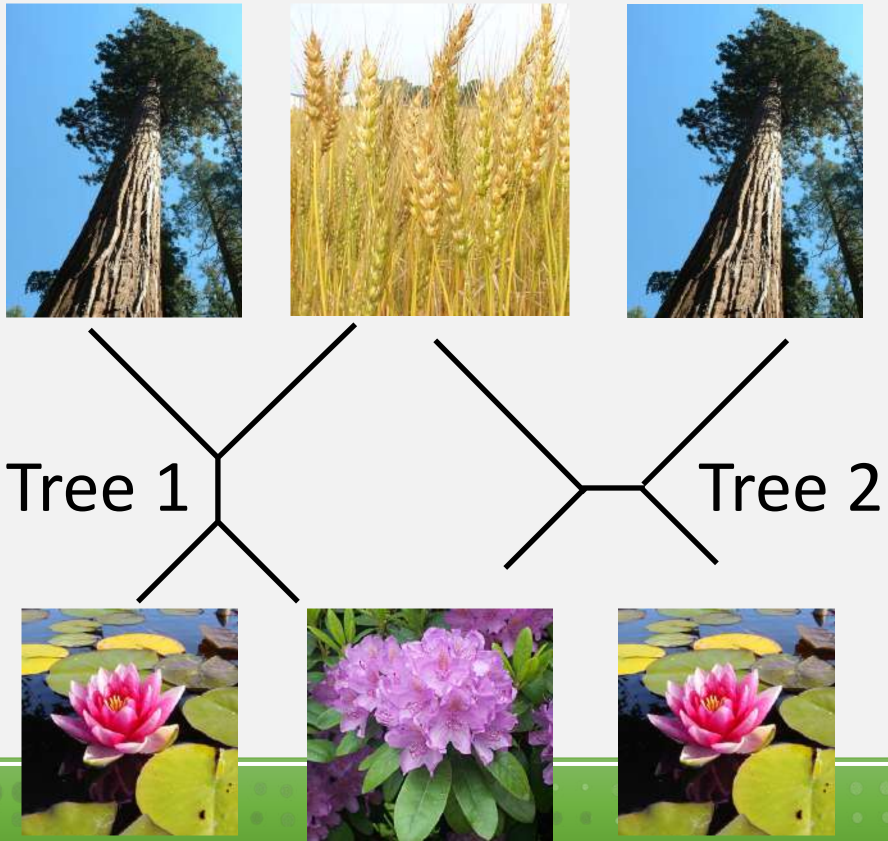
- The likelihood of a tree = the product of the site likelihoods
  - Taken as natural logs, the site likelihoods can be summed to give the log likelihood
- The sum of the probabilities for the 16 possible site patterns = 0.0333
- Hence, the site $-\ln L = 3.402$



Credit by Joe Felsenstein, Maximum Likelihood and model selection

the ML tree with the highest likelihood

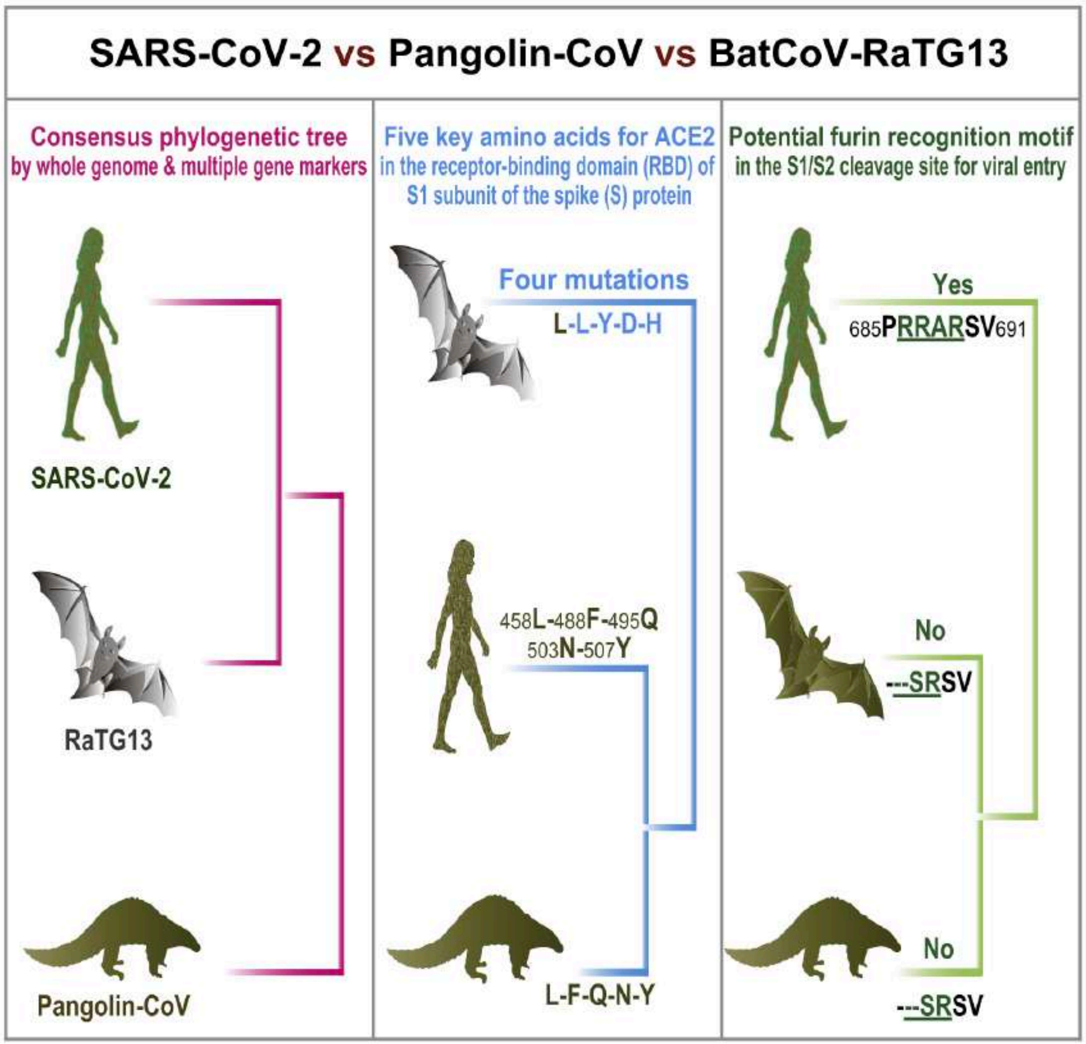
- The tree with the highest likelihood (lowest –ln*L*)
- Tree 2 is the ML tree by 8.801 –ln*L* units(=2052.456-2043.655)



Site	–ln*L*(1)	–ln*L*(2)
1	2.457	2.891
2	1.568	1.943
.	..	..
.	..	..
1206	2.541	1.943
	2052.456	2043.655

Credit by Joe Felsenstein, Maximum Likelihood and model selection

Phylogenetic Relationship of CoVs


SARS-CoV-2 vs Pangolin-CoV vs BatCoV-RaTG13
Consensus phylogenetic tree by whole genome & multiple gene markers
SARS-CoV-2
RaTG13
Pangolin-CoV
Five key amino acids for ACE2 in the receptor-binding domain (RBD) of S1 subunit of the spike (S) protein
Four mutations
L-L-Y-D-H
458L-488F-495Q
503N-507Y
L-F-Q-N-Y
Potential furin recognition motif in the S1/S2 cleavage site for viral entry
Yes
685PRRARSV691
No
---SRSV
No
---SRSV

Zhang, T., Wu, Q. & Zhang, Z. Probable Pangolin Origin of SARS-CoV-2 Associated with the COVID-19 Outbreak. *Curr Biology Cb* **30**, 1346-1351.e2 (2020).

If you build a tree,

Clustal

Dalign

DCA

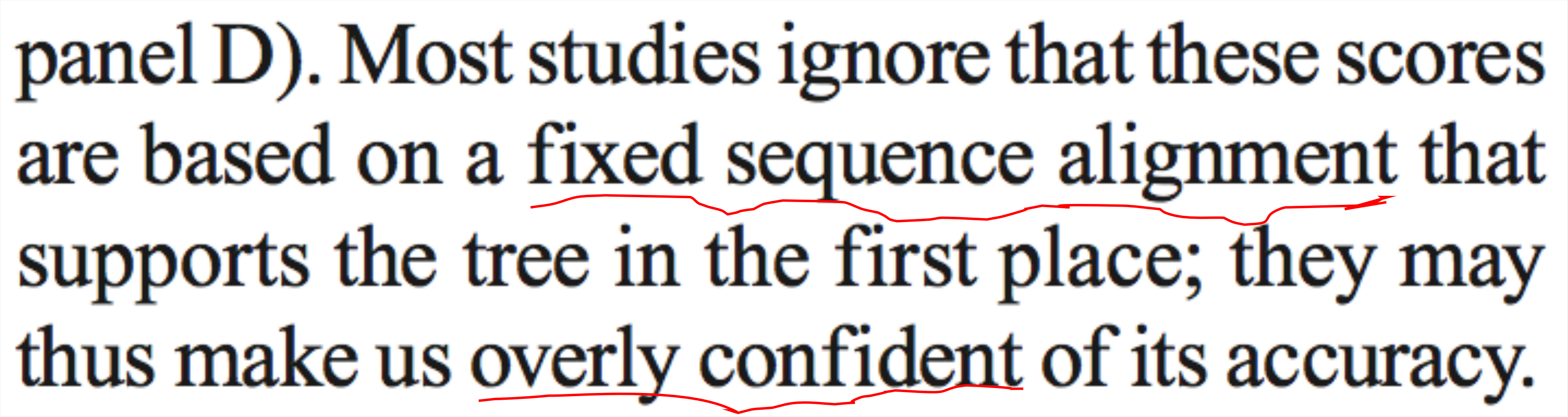
Mafft

Muscle

ProbCons

T-Coffee

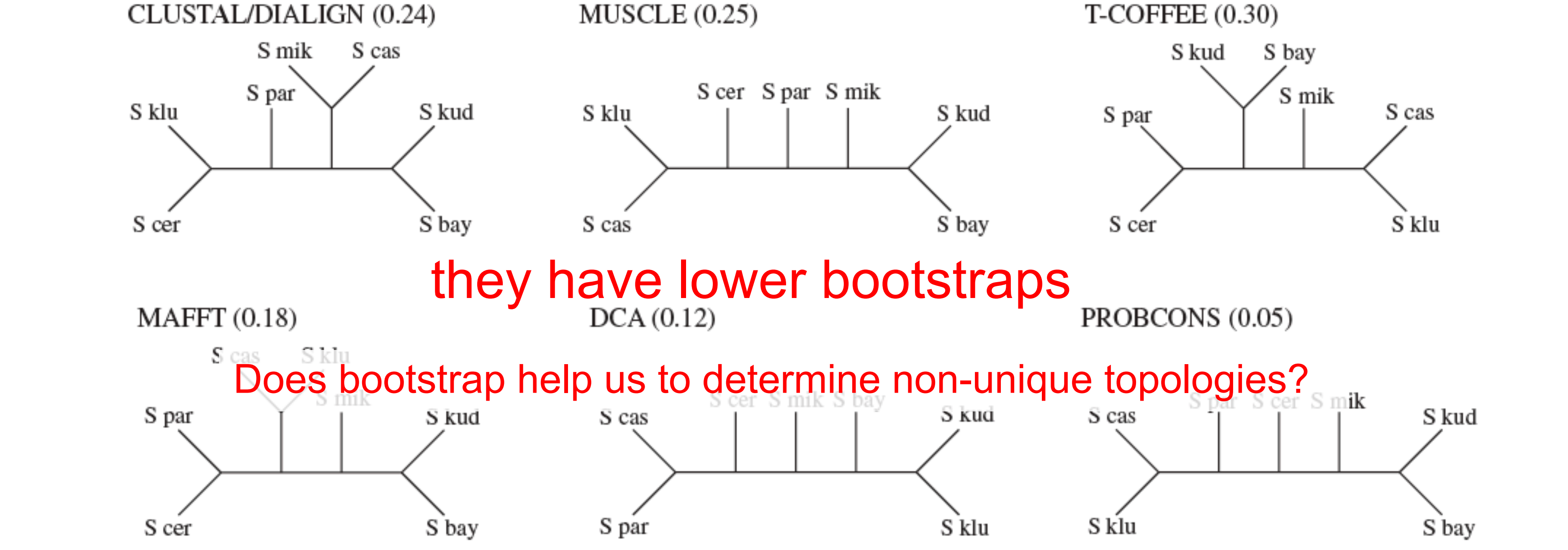
Which guy should I trust?



panel D). Most studies ignore that these scores are based on a fixed sequence alignment that supports the tree in the first place; they may thus make us overly confident of its accuracy.

Ari Löytynoja and Nick Goldman, “Uniting Alignments and Trees,” *Science* 324, no. 5934 (June 19, 2009): 1528 -1529.

YPL077C with six topologies

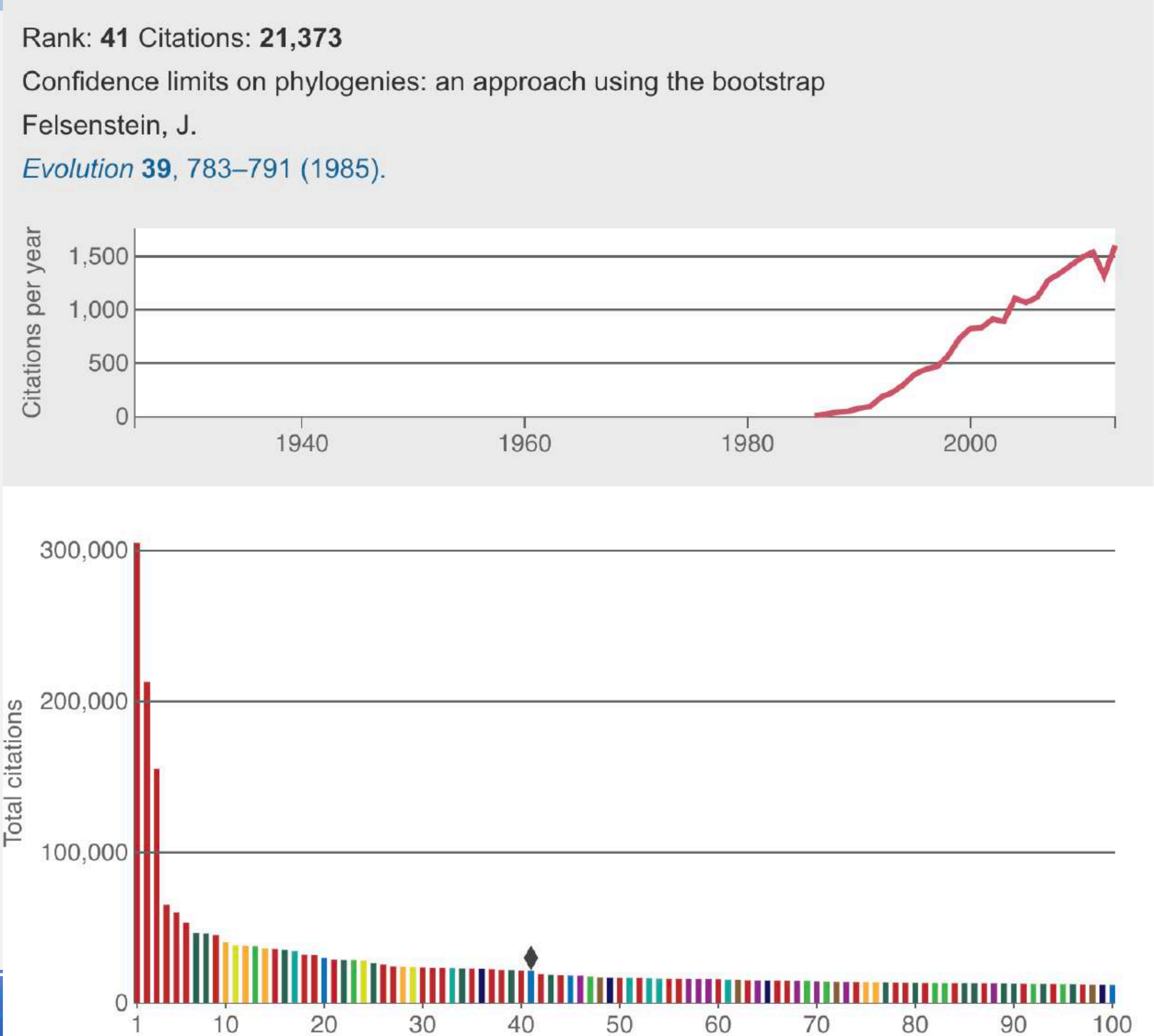


they have lower bootstraps

Does bootstrap help us to determine non-unique topologies?

Fig. 1. An example, involving ORF YPL077C, in which alignments produced by seven different alignment methods produce six different estimated trees, albeit with low bootstrap support (bootstrap proportions shown parenthetically for each tree).

Karen M Wong, Marc A Suchard, and John P Huelsenbeck, "Alignment uncertainty and genomic analysis", Science 319, no. 5862 (January 25, 2008): 473-476.



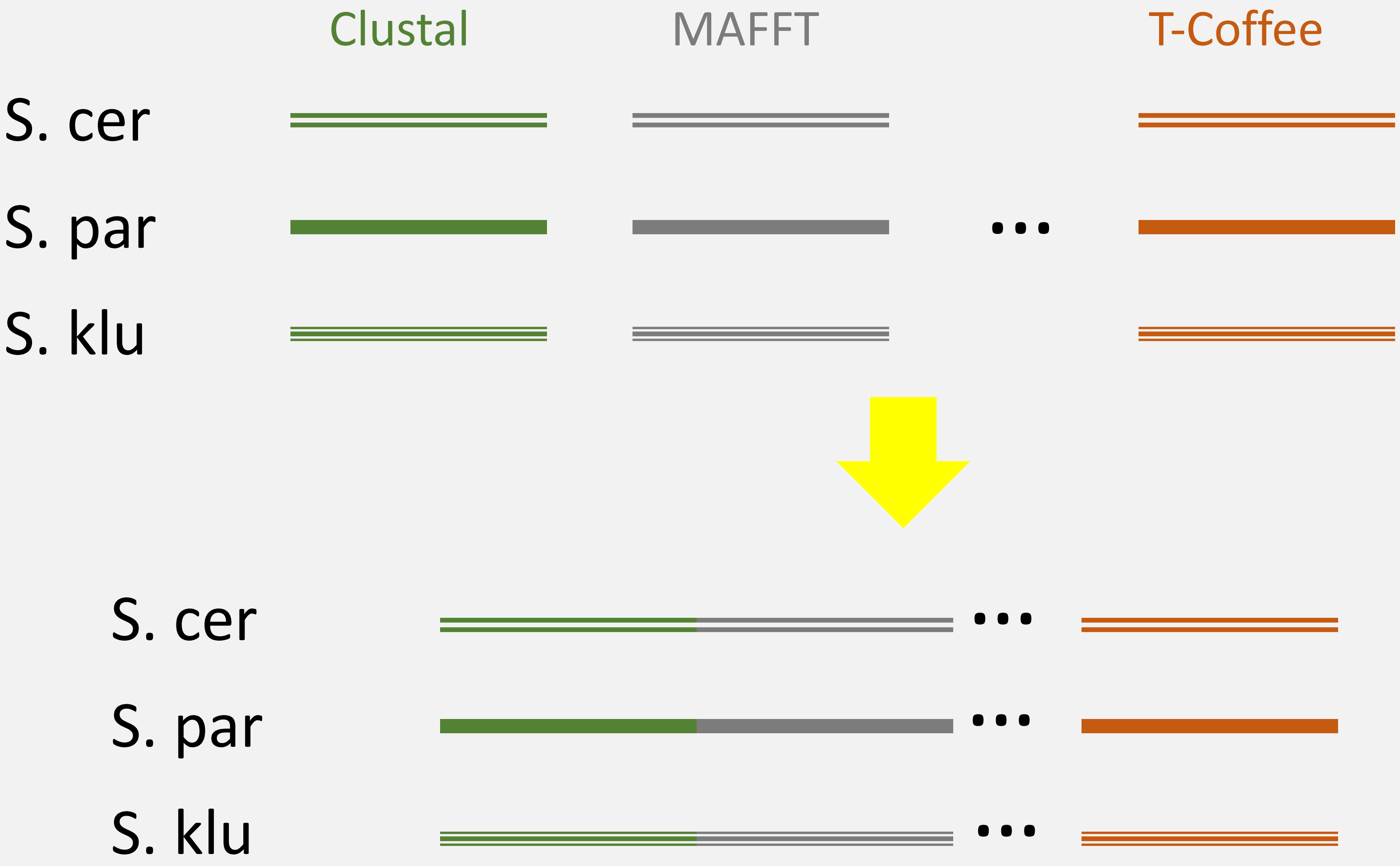
Rank: **41** Citations: **21,373**

Confidence limits on phylogenies: an approach using the bootstrap

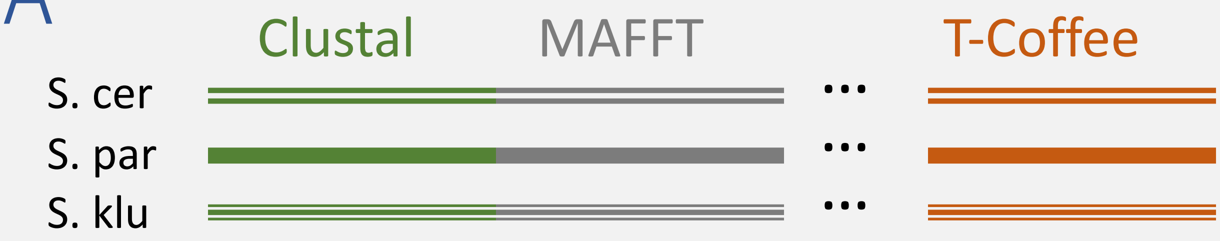
Felsenstein, J.

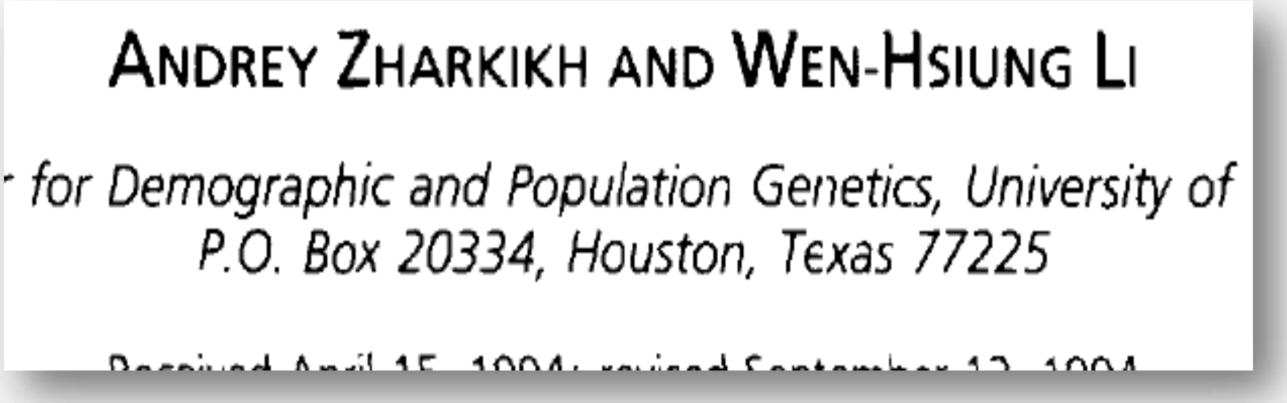
Evolution **39**, 783–791 (1985).

Super multiple sequence alignment (SMSA)



SMSA



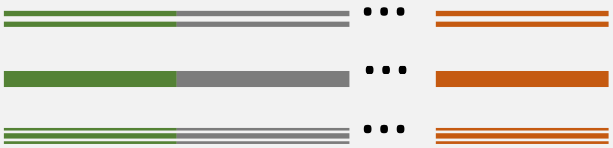


ANDREY ZHARKIKH AND WEN-HSIUNG LI

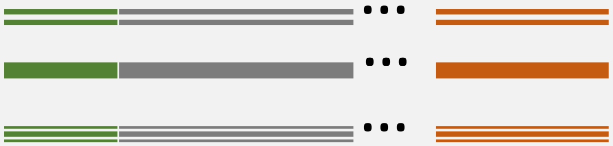
r for Demographic and Population Genetics, University of
P.O. Box 20334, Houston, Texas 77225

Received April 15, 1994; revised September 12, 1994

partial

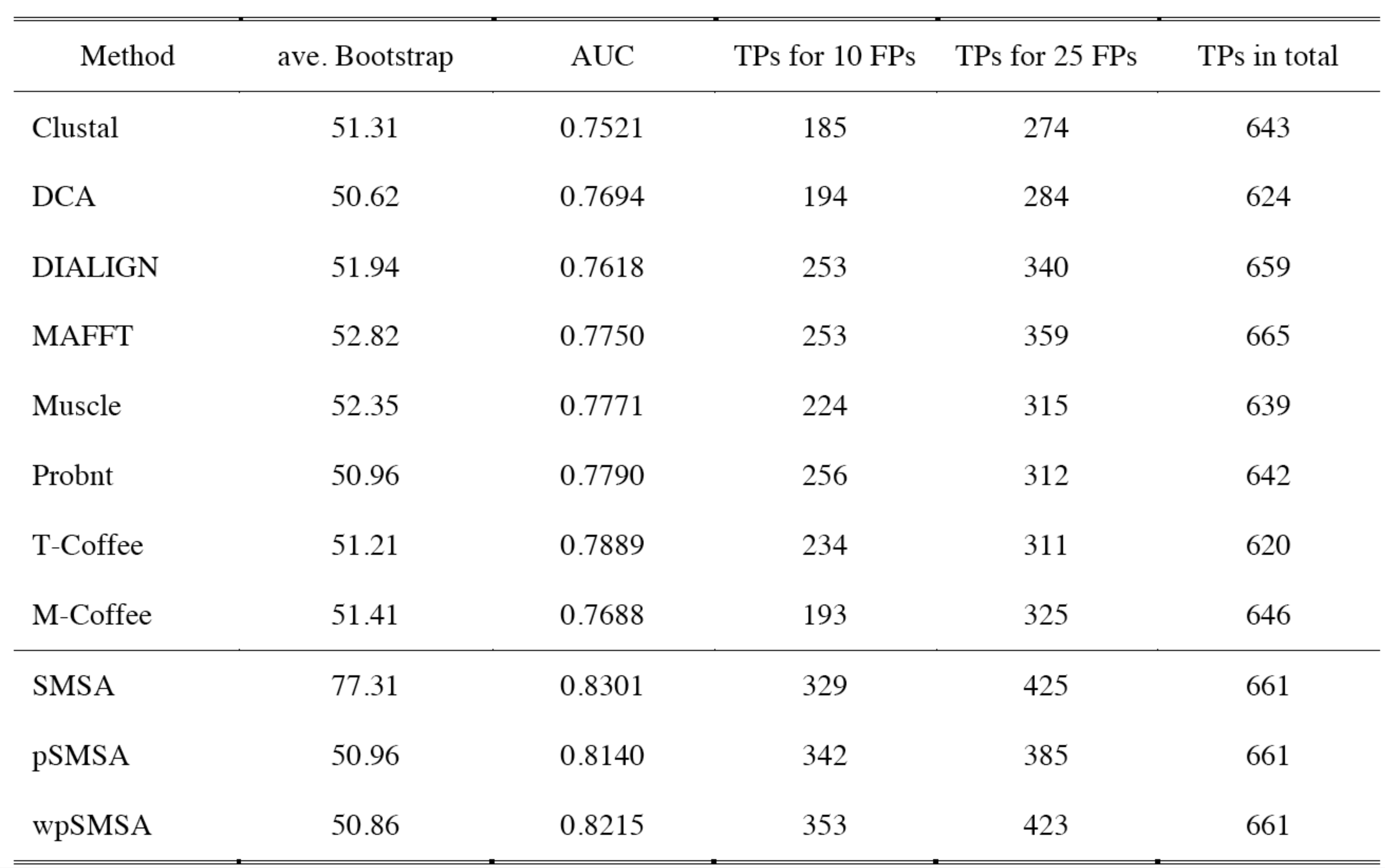


weighted



Zharkikh A, Li WH (1995) Estimation of confidence in phylogeny: the complete-and-partial bootstrap technique. Mol Phylogenet Evol 4: 44–63. doi:10.1006/mpev.1995.1005.

Average bootstrap, AUC values and the number of TPs for 10 and 25 accepted FPs of each method



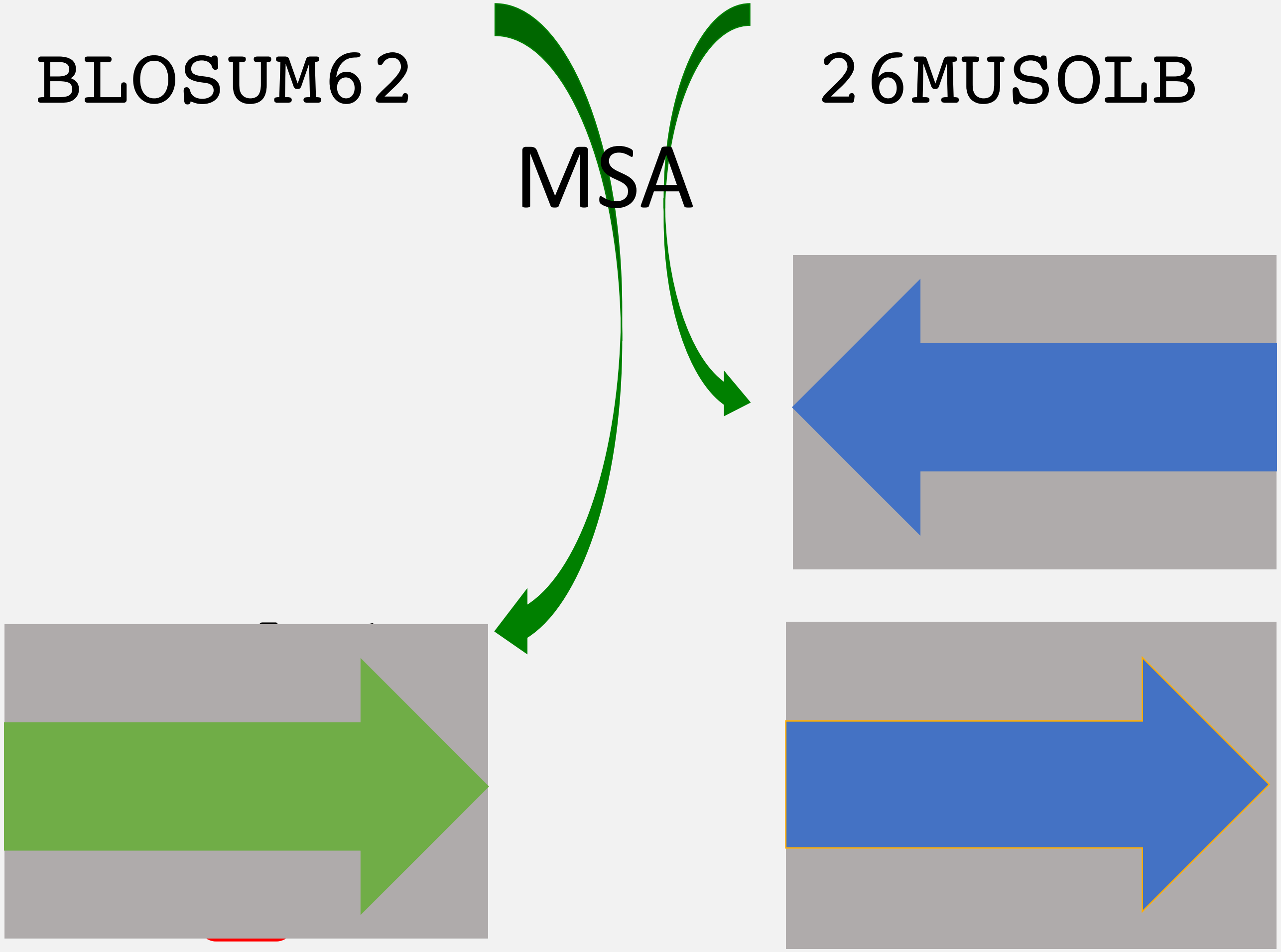
Method	ave. Bootstrap	AUC	TPs for 10 FPs	TPs for 25 FPs	TPs in total
Clustal	51.31	0.7521	185	274	643
DCA	50.62	0.7694	194	284	624
DIALIGN	51.94	0.7618	253	340	659
MAFFT	52.82	0.7750	253	359	665
Muscle	52.35	0.7771	224	315	639
Probnt	50.96	0.7790	256	312	642
T-Coffee	51.21	0.7889	234	311	620
M-Coffee	51.41	0.7688	193	325	646
SMSA	77.31	0.8301	329	425	661
pSMSA	50.96	0.8140	342	385	661
wpSMSA	50.86	0.8215	353	423	661

OPOSSUM
BLOSUM62

aligners

```
OPOSSUM-- OPOSSUM--
BLOS-UM62 BLO-SUM62
```

alignment uncertainty - data



OPOSSUM

BLOSUM62

MUSSOPO

26MUSOLB

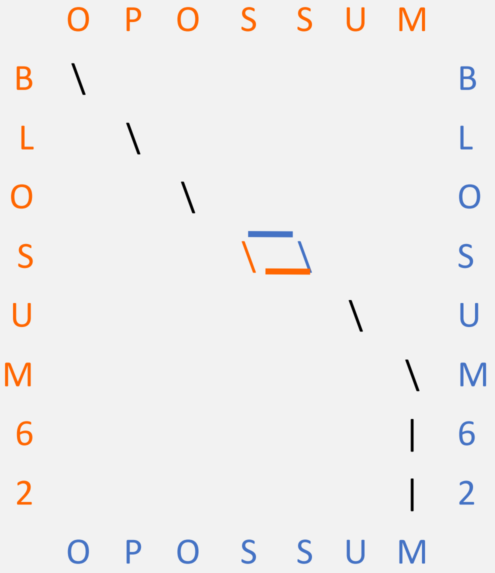
MSA

Landan G, Graur D (2007) Heads or Tails: A Simple Reliability Check for Multiple Sequence Alignments. Molecular Biology and Evolution 24: 1380 –1383.

alignment uncertainty - data

```
Aln1          Aln2
OPOSSUM--     OPOSSUM--
BLOS-UM62     BLO-SUM62
```

```
If there are two paths
{
    chooses low-road;
}
```



Landan G, Graur D (2007) Heads or Tails: A Simple Reliability Check for Multiple Sequence Alignments. Molecular Biology and Evolution 24: 1380–1383.

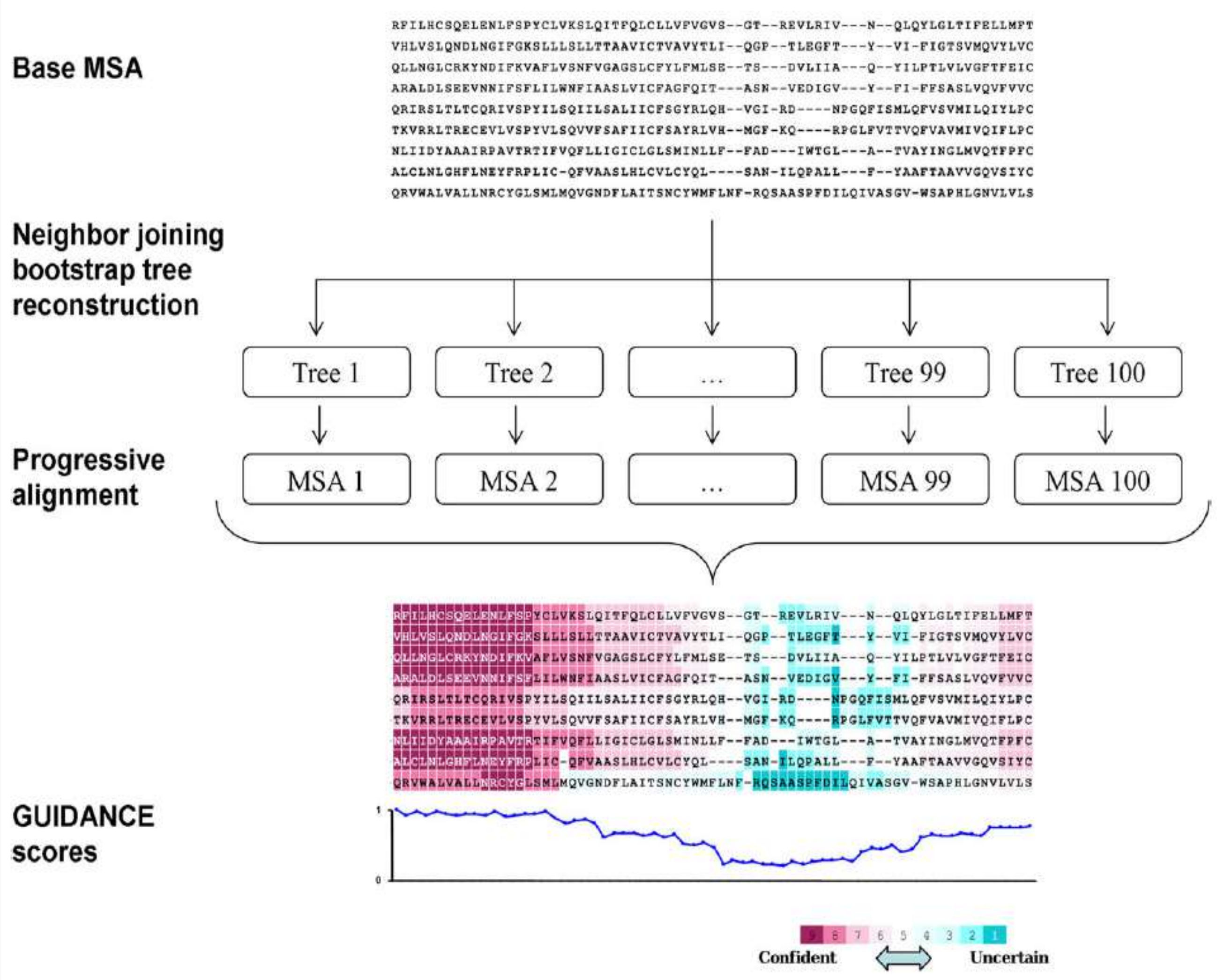
alignment uncertainty - data

- It gets worse with a multiple sequence alignment.

```
   Aln1      Aln2      Aln3      Aln4
BLOS-UM45 BLO-SUM45 BLO-SUM45 BLOS-UM45
OPOSSUM-- OPOSSUM-- OPOSSUM-- OPOSSUM--
BLOS-UM62 BLOS-UM62 BLO-SUM62 BLO-SUM62
```

Telling apart Uncertainty parts of the alignment is more important than the overall accuracy.

Guidance

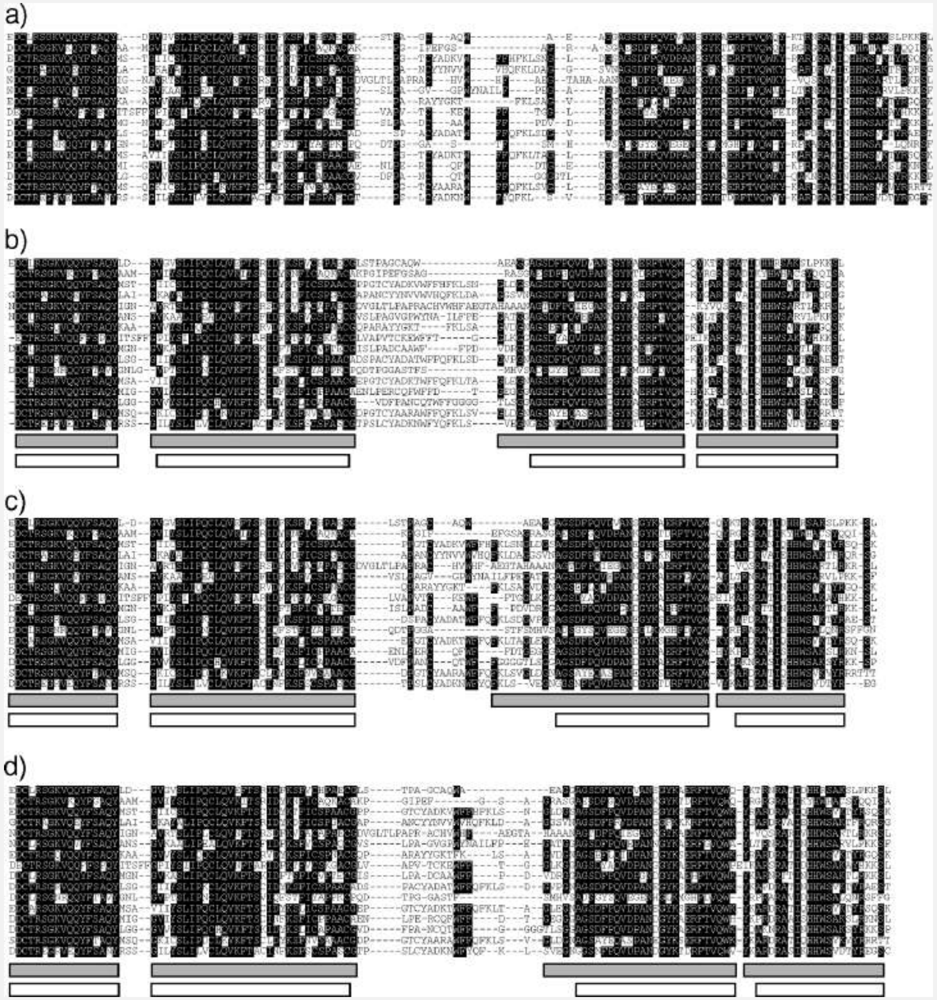


```
RFILHCSQELENLFSPYCLVKSLQITFQLCLLVFVGVS--GT--REVLRIV---N--QLQYLGLTIFELLMFT
VHLVSLQNDLNGIFGKSLLLSLLTTAAVICTVAVYTLI--QGP--TLEGFT---Y--VI-FIGTSVMQVYLVC
QLLNGLCRKYNDIFKVAFLVSNFVGAGSLCFYLFMLSE--TS---DVLIIA---Q--YILPTLVLVGFTFEIC
ARALDLSEEVNNIFSFLILWNFIAASLVICFAGFQIT---ASN--VEDIGV---Y--FI-FFSASLVQVFVVC
QRIRSLTLTCQRIVSPYILSQIILSALIICFSGYRLQH--VGI-RD----NPGQFISMLQFVSVMILQIYLPC
TKVRRLTRECEVLVSPYVLSQVVFSAFIICFSAYRLVH--MGF-KQ----RPGLFVTTVQFVAVMIVQIFLPC
NLIIDYAAAIRPAVTRTIFVQFLLIGICLGLSMINLLF--FAD---IWTGL---A--TVAYINGLMVQTFPFC
ALCLNLGHFLNEYFRPLIC-QFVAASLHLCVLCYQL----SAN-ILQPALL---F--YAAFTAAVVGQVSIYC
QRVWALVALLNRCYGLSMLMQVGNDFLAITSNCYWMFLNF-RQSAASPFDILQIVASGV-WSAPHLGNVLVLS
```

```
RFILHCSQELENLFSPYCLVKSLQITFQLCLLVFVGVS--GT--REVLRIV---N--QLQYLGLTIFELLMFT
VHLVSLQNDLNGIFGKSLLLSLLTTAAVICTVAVYTLI--QGP--TLEGFT---Y--VI-FIGTSVMQVYLVC
QLLNGLCRKYNDIFKVAFLVSNFVGAGSLCFYLFMLSE--TS---DVLIIA---Q--YILPTLVLVGFTFEIC
ARALDLSEEVNNIFSFLILWNFIAASLVICFAGFQIT---ASN--VEDIGV---Y--FI-FFSASLVQVFVVC
QRIRSLTLTCQRIVSPYILSQIILSALIICFSGYRLQH--VGI-RD----NPGQFISMLQFVSVMILQIYLPC
TKVRRLTRECEVLVSPYVLSQVVFSAFIICFSAYRLVH--MGF-KQ----RPGLFVTTVQFVAVMIVQIFLPC
NLIIDYAAAIRPAVTRTIFVQFLLIGICLGLSMINLLF--FAD---IWTGL---A--TVAYINGLMVQTFPFC
ALCLNLGHFLNEYFRPLIC-QFVAASLHLCVLCYQL----SAN-ILQPALL---F--YAAFTAAVVGQVSIYC
QRVWALVALLNRCYGLSMLMQVGNDFLAITSNCYWMFLNF-RQSAASPFDILQIVASGV-WSAPHLGNVLVLS
```

Penn O, Privman E, Landan G, Graur D, Pupko T (2010) An alignment confidence score capturing robustness to guide tree uncertainty. Mol Biol Evol 27: 1759–1767.

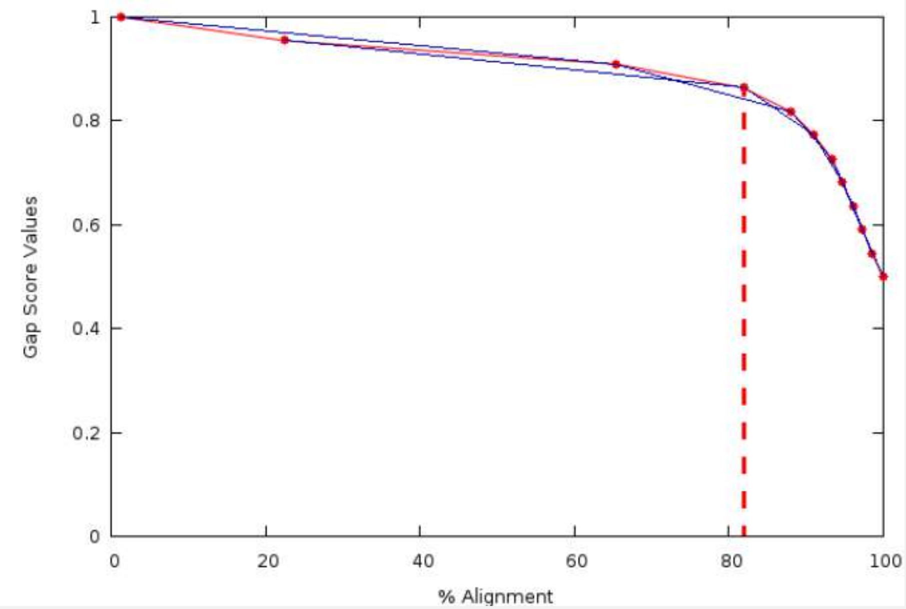
Gblocks



Talavera G, Castresana J (2007) Improvement of Phylogenies after Removing Divergent and Ambiguously Aligned Blocks from Protein Sequence Alignments. Syst Biol 56: 564–577.

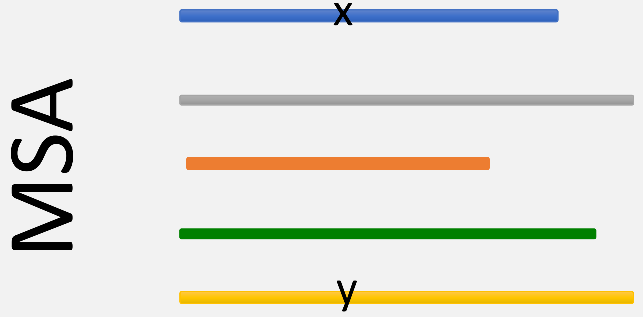
2691 citation by Google

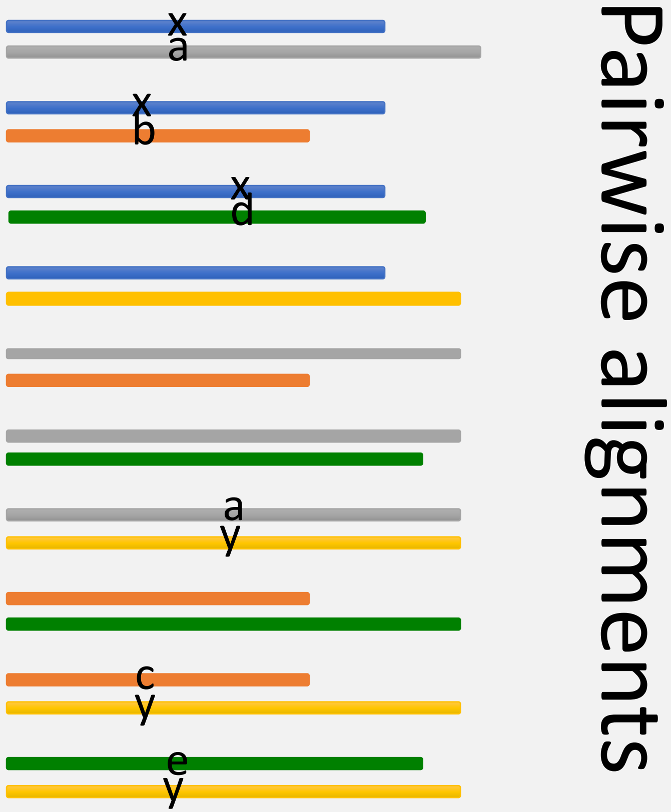
trimAl



Capella-Gutiérrez S, Silla-Martínez JM, Gabaldón T (2009) trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses. Bioinformatics 25: 1972–1973.

2270 citation by Google

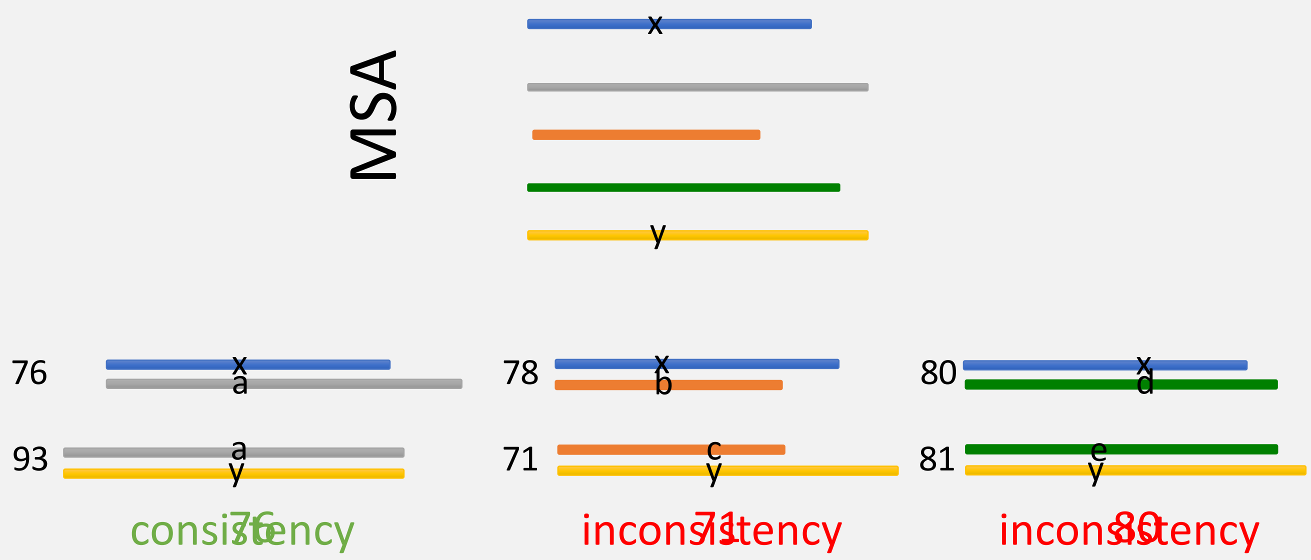




consistency

inconsistency

inconsistency



consistency 76

inconsistency 71

inconsistency 80

$$TCS\ (x,y)= \frac{76}{76 + 71 + 80}$$

```
CLUSTAL W (1.83) multiple sequence alignment

1j46_A          MQ------DRVKRP---MNAFIVWSRDQRRKMALENPRMRN--SEISKQL
2lef_A          MH--------IKKP---LNAFMLYMKEMRANVVAESTLKES--AAINQIL
1k99_A          MKKLKKHPDFPKKP---LTPYFRFFMEKRAKYAKLHPEMSN--LDLTKIL
1aab_           GK------GDPKKPRGKMSSYAFFVQTSREEHKKKHPDASVNFSEFSKKC
                 :         *:*   :..:  :    * :     .        :.:
```

TCS

Residue level

Col	row	row	TCS
1	1	2	0.762
1	1	3	0.748
1	1	4	0.741
1	2	3	0.651
1	2	4	0.677
1	3	4	0.693
2	1	3	0.562
2	1	4	0.632
2	3	4	0.526

…

```
T-COFFEE, Version_9.01 (2012-01-27 09:40:38)
Cedric Notredame
CPU TIME:0 sec.
SCORE=76
*
 BAD AVG GOOD
*
1j46_A   :  74
2lef_A   :  75
1k99_A   :  77
1aab_    :  72
cons     :  76

1j46_A   75------4566---6777777777777777777776666--7789999
2lef_A   6--------566---677777777777777777777766--7789999
1k99_A   865454445667---777788887888888888877877--7789999
1aab_    76------566533356667666666666666666655336789999
cons     641111113455122566777666666777777666655215689999
```

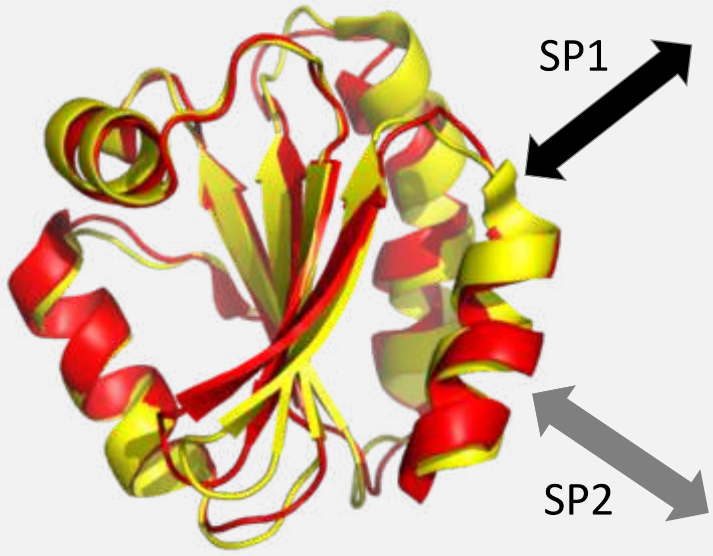
Alignment level

Column level

Test2 - structural modeling @ alignment level

reference alignment

Guidence/TCS



SP1

SP2

```
Seq1 …SALMLWLSARESIKREN…YPD…
Seq2 …SAYNIYVSFQ----RESA…KD…
…
Seqn …SAYNIYVSAQ----RENA…KD…
```

confidence1

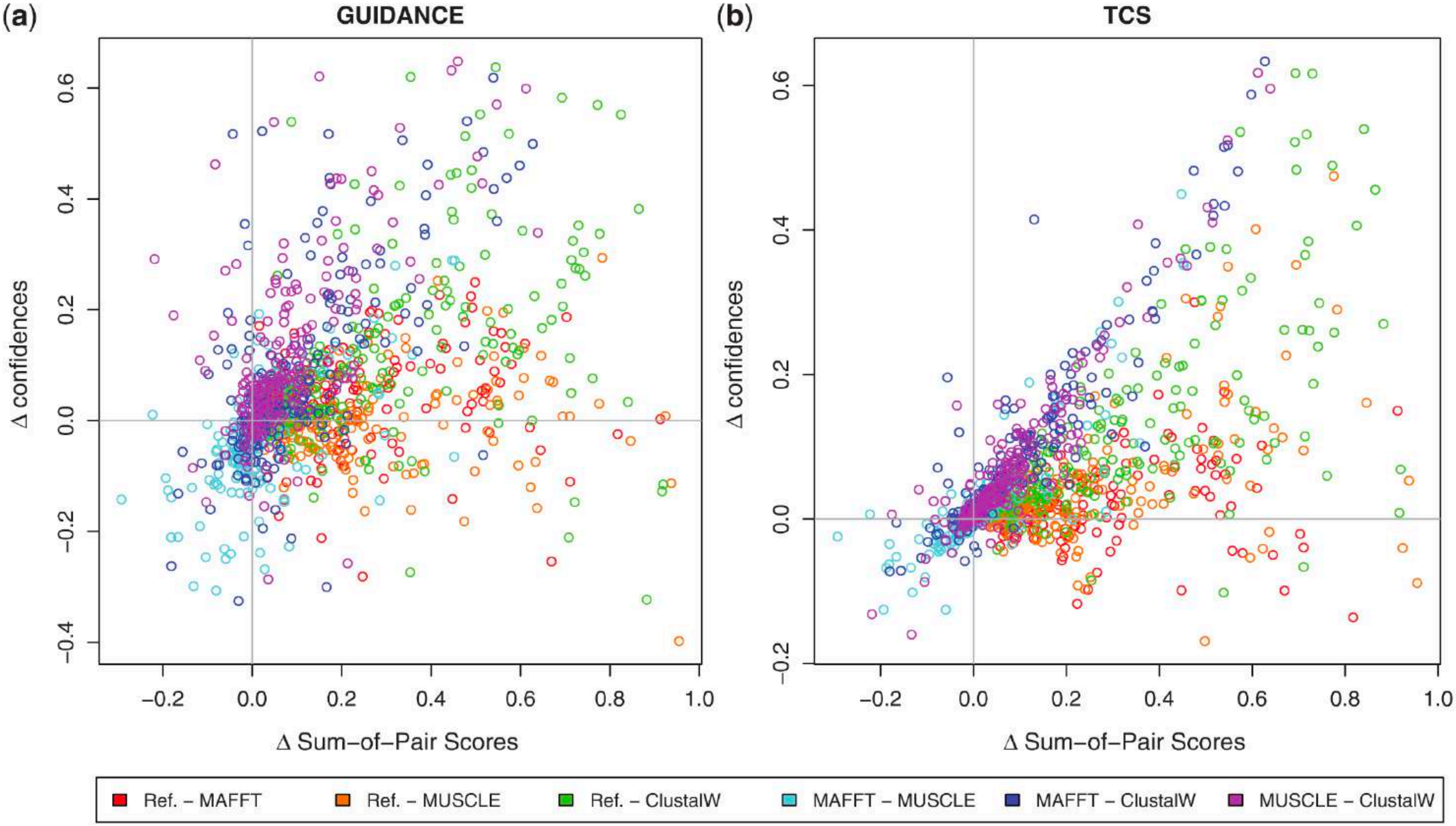
```
Seq1 …SALMLWLSARESIKREN…YPD…
Seq2 …SAYNIYVSF----QRESA…KD…
…
Seqn …SAYNIYVSA----QRENA…KD…
```

confidence2

SP1 – SP2 ? confidence1 – confidence2

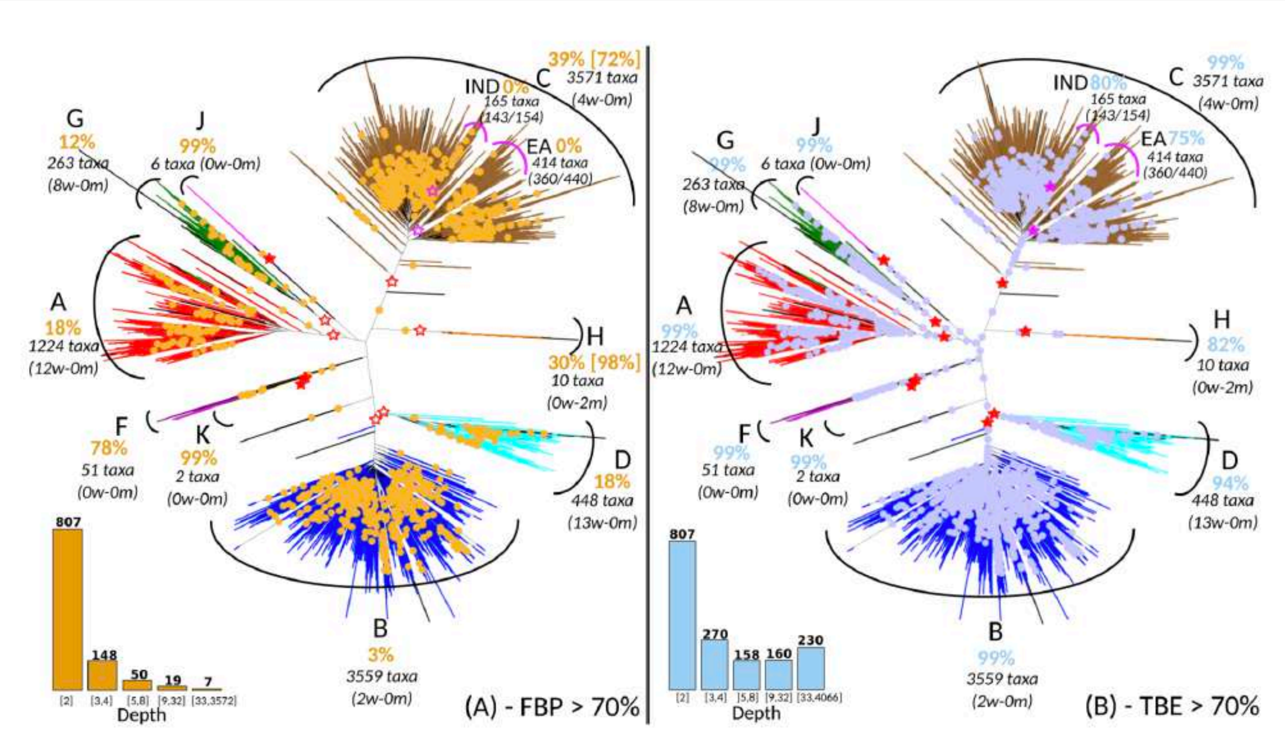
Guidance = 71.10%

TCS = 83.5%



Renewing Felsenstein’s phylogenetic bootstrap in the era of big data

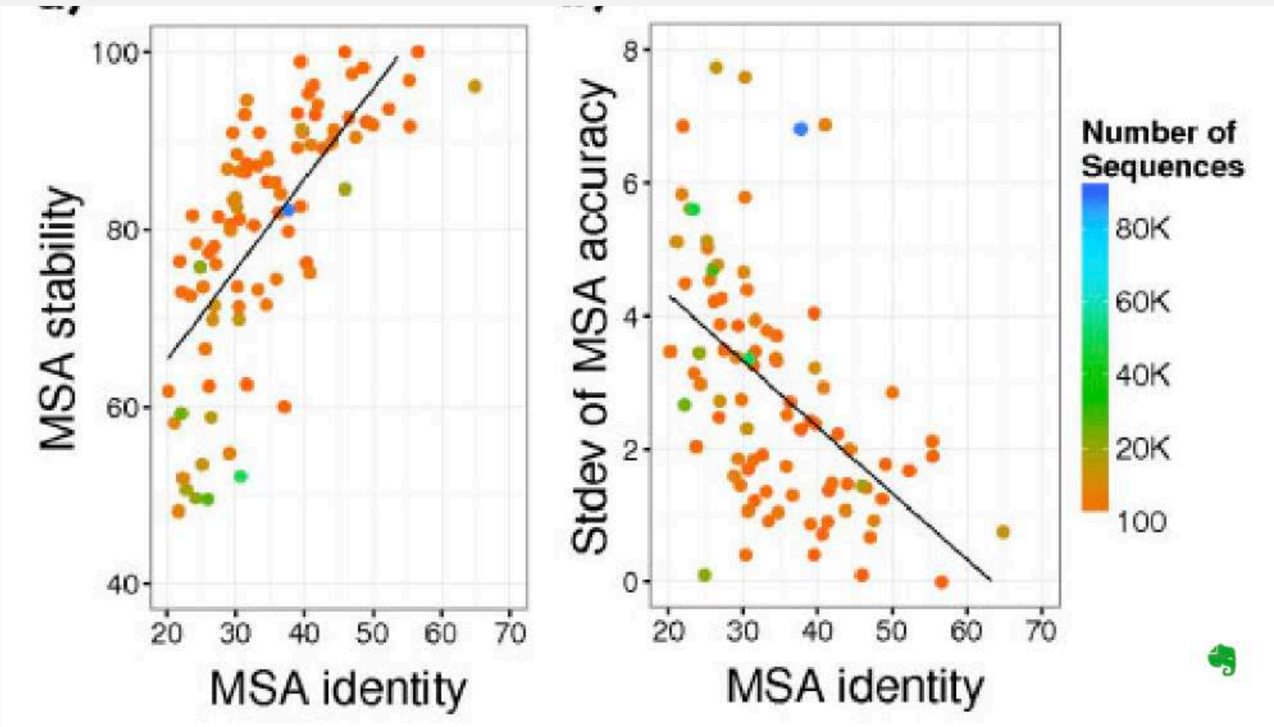
- *transfer distance*, (b,b*) : a branch b of the reference tree T and a branch b* of a bootstrap tree T* is equal to the number of taxa that must be transferred (or removed), in order to make both branches identical
- Felsenstein (FBP) and transfer (TBE) bootstrap supports on the same tree with **9,147** HIV-1M pol sequences



Lemoine, F. et al. Nature 556, 452–456 (2018).

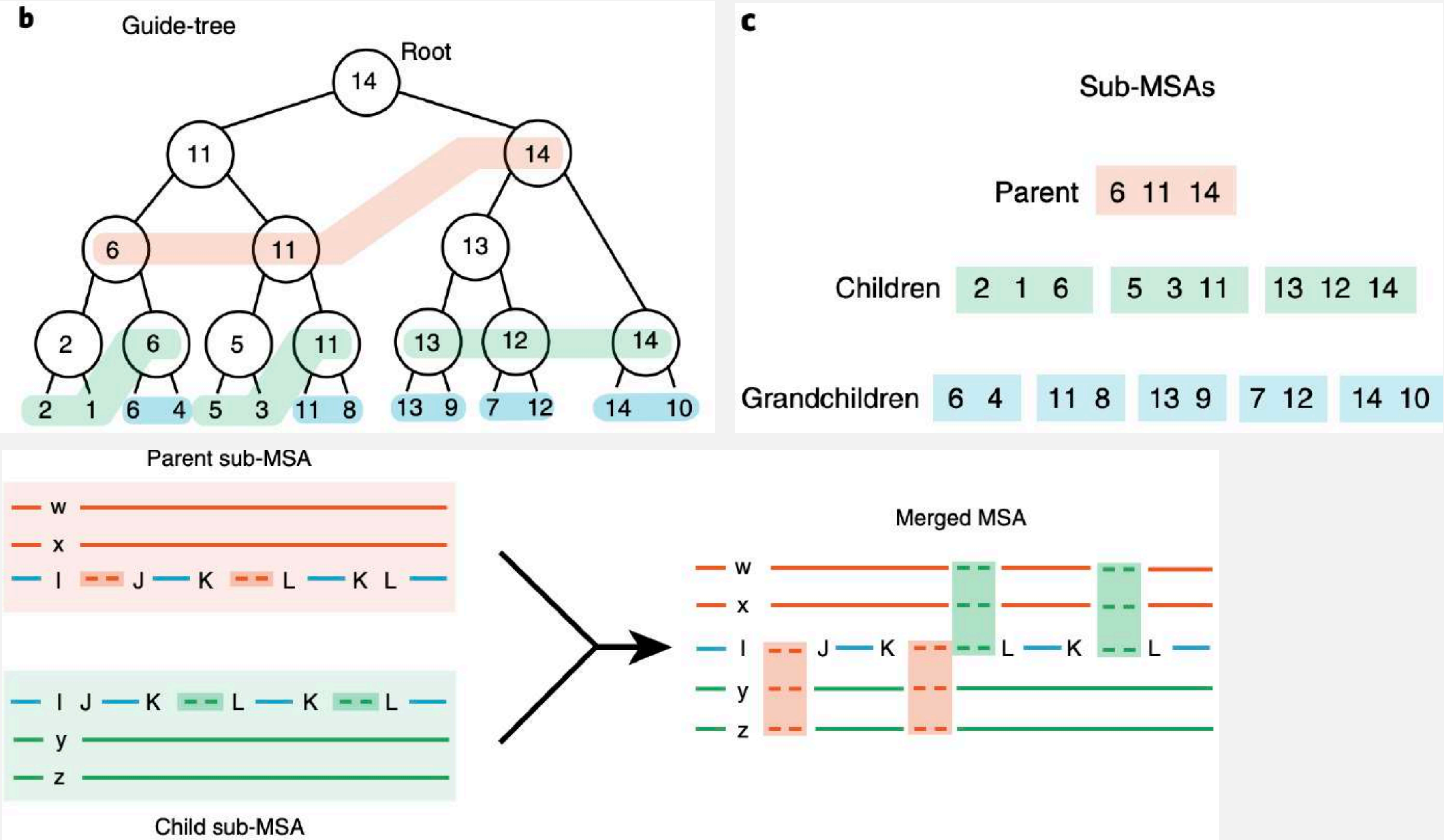
Significantly different output when changing sequence input order

- S-o-P comparison vs average identity (Spearman correlation rs = 0.79).
- a high MSA structural accuracy variability vs correlating with MSA identity (rs = −0.51)



Maria Chatzou, Evan W Floden, Paolo Di Tommaso, Olivier Gascuel, Cedric Notredame, *Systematic Biology*, 2018

Regressive algorithm enables MSA of up to 1.4 million sequences on



Edgar Garriga et al, Nature Biotechnology 2019

GCAT
TACG
GCAT

genes

an Open Access Journal by MDPI

Impact Factor 3.331

24 days to first decision

Special Issue “Phylogenetic Methods in the Genomic Era: Challenges in Multiple Sequence Alignment and Phylogenetics for Genome-Scale Data”

Guest Editors

Dr. Cedric Notredame
Pompeu Fabra University,
Barcelona, Spain

Dr. Jia-Ming Chang
National Chengchi University,
Taipei, Taiwan

Dr. Minh Bui
University of Melbourne,
Melbourne, Australia

Dr. Ding He
University of Copenhagen,
Copenhagen, Denmark

Deadline for submissions:
1 June 2020

Short Information about the Special Issue:

Grand-scale genome sequencing projects dedicating a systematic approach to targeting well-recognized taxonomic groups have started to appear.

To highlight the importance of this exciting moment for phylogenetic method development and evolutionary data inference in facing the big data era, this Special Issue welcomes contributions of methods and metrics addressing challenges from sequence alignment to tree reconstruction in phylogenomics.

Author Benefits

Open Access

No Copyright Constraints Retain copyright of your work and free use of your article

Rapid Publication

No Space Constraints or Color Charges

Thorough peer-review

MDPI

www.mdpi.com

Thank You
Any Question?