Bioinformatics — Lecture 3
Sequence alignments
(EG Chps. 6, 10; MM Chps. 4–6)

Krzysztof Bartoszek

Linköping University

krzysztof.bartoszek@liu.se

20 XI 2023 (SH63)

Today

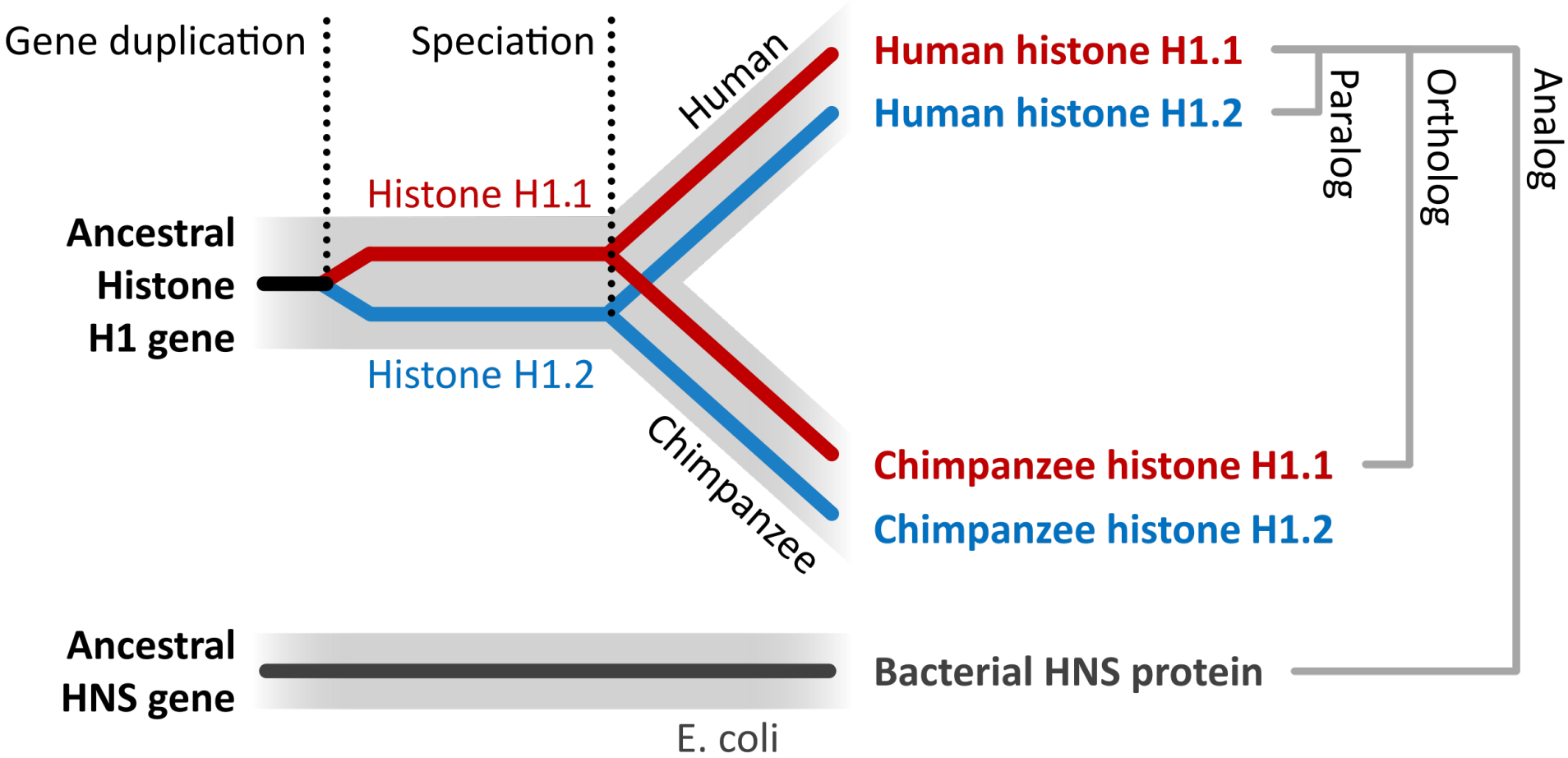
- Sequence similarity
  - Alignment
  - Are sequences related?
  - Sequence distances
- Global alignment algorithms
- Multiple sequence alignment
- Local alignment
- BLAST

Additional reading

A computer science perspective: chapter 7 in
[CHL] M. Crochemore, C. Hancart, T. Lecroq. Algorithms on Strings, Cambridge, 2007, Cambridge University Press.

Multiple alignments: chapter 6 in
[DEKM] R. Durbin, S. Eddy, A. Krogh, G. Mitchison. Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids, Cambridge, 1998, Cambridge University Press.

Gene homology (MM Ch. 3.4)



https://en.wikipedia.org/wiki/File:Ortholog_paralog_analog_examples.svg, by Thomas Shafee, CC BY 4.0

Gene homology

Orthologues: genes from different species that are descended from a common ancestral sequence, often carry out the same function

Paralogues: genes derived from a gene duplication event, can be in same or different species, and can have same or different functions

Identify genes in one species, given gene sequences in another species: *sequence alignment*

Alignment (simple) example

Original sequence: *CGGTATGCCA*

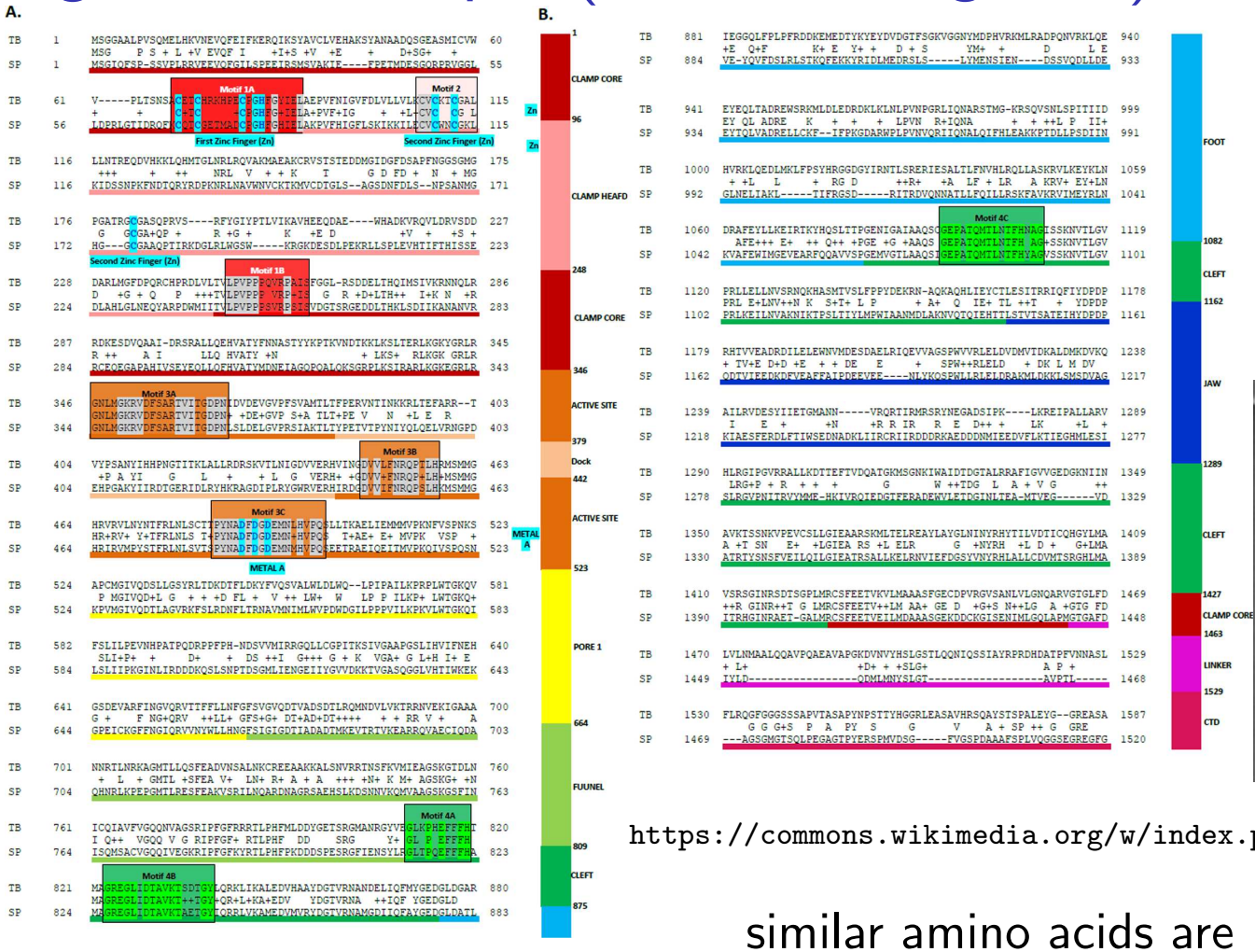
Descendant 1 sequence: *CGGGTATCCAA*
Descendant 2 sequence: *CCCTAGGTCCCA*

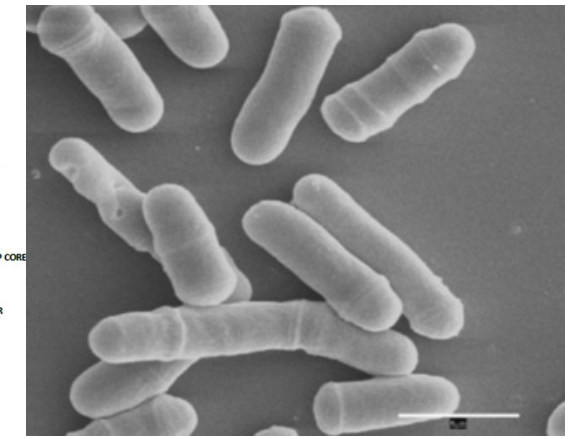
Alignment:

```
C G G G T A - - T C C A A
C C C - T A G G T C C C A
```

https://es.wikipedia.org/wiki/Trypanosoma_brucei#/media/Archivo:Tb_brucei.jpg, by Torsten Ochsenreiter, public domain

Alignment example (also MM Fig. 3.4)



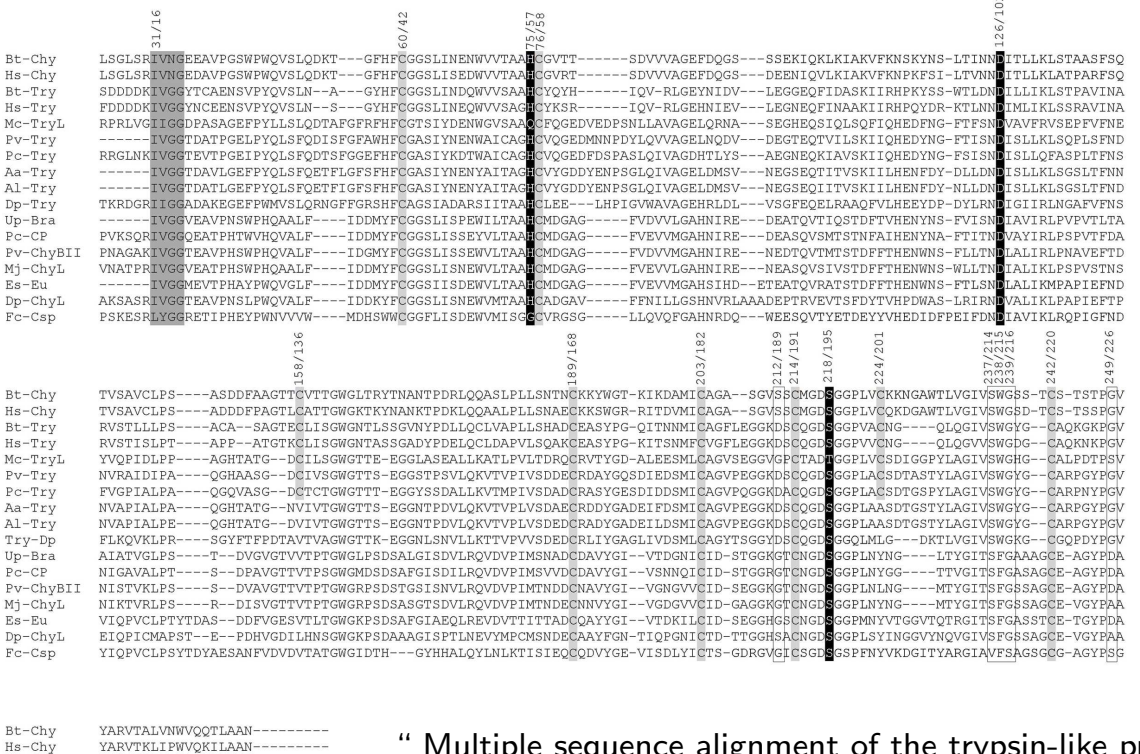


https://commons.wikimedia.org/w/index.php?curid=8627875, public domain

similar amino acids are marked, also . : |

Fig. 3 in L. Papageorgiou, V. Megalooikonomou, D. Vlachakis, 2017. Genetic and structural study of DNA-directed RNA polymerase II of *Trypanosoma brucei*, towards the designing of novel antiparasitic agents. PeerJ 5:e3061 doi:10.7717/peerj.3061. CC-BY 4.0

Alignment example (also MM Fig. 3.12)



" Multiple sequence alignment of the trypsin-like protease from *M. carcinus* (Mc-TryL), 12 mature brachyurins and 2 mammalian trypsins and 2 mammalian chymotrypsins as the reference. Grey shadows show sequence motif at the beginning of the mature protein, or conserved cysteines. The catalytic triad is denoted by black boxes, while substrate binding residues and residues that confer the peptide-bond specificity are shown by white boxes."
(trypsin: enzyme in small intestine starting digestion of protein molecules)

Fig. 2 in J.M Viader-Salvadó, J.A. Aguilar Briseño, J.A. Gallegos-López, J.A. Fuentes-Garibay, C.A. Alvarez-González, M. Guerrero-Olazarán, 2020. Identification and in silico structural and functional analysis of a trypsin-like protease from shrimp *Macrobrachium carcinus*. PeerJ 8:e9030 doi:10.7717/peerj.9030. CC-BY 4.0

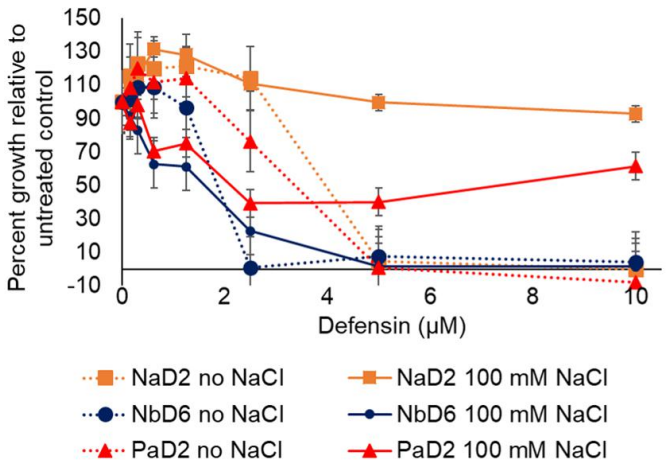
Sequence alignment (also MM Fig. 3.13)

National Museum of Anthropology (Mexico)

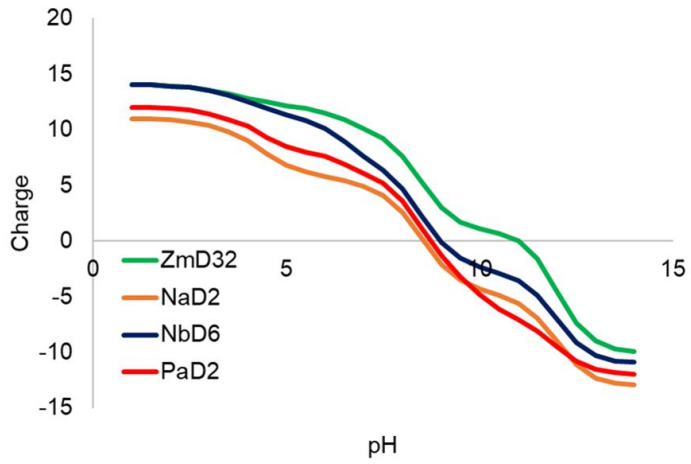
A

```
                                                          Ident   Sim   Charge
PaD2    RTCESQSHKFKGTCLSDTNCGNVCHSEGFPGGKCRGLRRRCFCTKNC   70.2    87.2  +6.1
NbD6    RTCESQSHRFKGLCFSRSNCASVCHTEGFNGGHCRGFRRRCFCTRHC   78.7    87.2  +7.6
NaD2    RTCESQSHRFKGPCARDSNCATVCLTEGFSGGDCRGFRRRCFCTRPC   78.7    85.1  +4.9
ZmD32   RTCQSQSHRFRGPCLRRSNCANVCRTEGFPGGRCRGFRRRCFCTTHC   100     100   +10.1
        ***:****:*:* *   :**..** :*** ** ***:*******  *
```

B



C



different defensin (against microbes) proteins share similar sequences behave differently with varying salinity

Fig. 8 in K. Kerenga Bomai et. al., 2019. Salt-Tolerant Antifungal and Antibacterial Activities of the Corn Defensin ZmD32. Front. Microbiol. 10 doi:10.3389/fmicb.2019.00795. CC BY.

Comparing sequences (frequency EG Ch. 6.1)

	nucleotide				Total
sequence 1	$\#A_1$	$\#C_1$	$\#G_1$	$\#T_1$	L_1
sequence 2	$\#A_2$	$\#C_2$	$\#G_2$	$\#T_2$	L_2
Total	$\#A$	$\#C$	$\#G$	$\#T$	L

H_0: both sequences are drawn from populations with identical nucleotide frequencies

$$p_{1N} = \#N_1/L_1, p_{N2} = \#N_2/L_2, p_N = \#N/L$$

$$2L\left(\frac{L_1}{L}\sum p_{1N}\log p_{1N} + \frac{L_2}{L}\sum p_{2N}\log p_{2N} - \sum p_N \log p_N\right) \sim \chi^2(3)$$

Comparing sequences (EG Ch. 6.3)

Ungapped alignment assumed (uniformly generated in R, 4 runs needed for LCS$\geq$ 3)

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

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

H_0: two sequences in ungapped alignment are random with respect to each other

Exact–matching test statistic: $Y_{\max}$—length of longest common subsequence (here 3)

$$Y_{\max} = y_{\max} \text{ has p} - \text{value } \cong 1 - \exp(-(1-p)Np^{y_{\max}})$$

(EG Eq. 5.15, Lec 2 slide 30), where
$p = P(A)^2 + P(G)^2 + P(C)^2 + P(T)^2$ (nucleotide frequencies)

Well–matching test statistic: $Y_{\max}^{(k)}$—length of longest subsequence with up to k mismatches

p–value approximate (see EG Eqs. 6.5. 6.6, 3.51)

Distance on X (CHL Ch. 7.1)

a function, $d : X \times X \to \mathbb{R}$ satisfying

positivity: $\forall_{u,v \in X} d(u, v) \geq 0$

separation: $\forall_{u,v \in X} d(u, v) = 0 \text{ iff } u = v$

symmetry: $\forall_{u,v \in X} d(u, v) = d(v, u)$

triangle inequality: $\forall_{u,v,w \in X} d(u, v) \leq d(u, w) + d(v, w)$

Similarity score $s(u, v) = (-1) \cdot d(u, v)$

Distances on sequences (CHL Ch. 7.1)

Prefix distance:

$$d_{pref}(u, v) = |u| + |v| - 2|\text{longest common prefix}(u, v)|$$

Suffix distance:

$$d_{pref}(u, v) = |u| + |v| - 2|\text{longest common suffix}(u, v)|$$

Factor distance:

$$d_{pref}(u, v) = |u| + |v| - 2|\text{longest common subsequence}(u, v)|$$

Hamming distance: for two strings of same length, number of positions on which they differ

Levenshtein (edit) distance (CHL Ch. 7.1)

$Lev(x, y) = \min(\text{cost of transformation } u \text{ to } v)$

permitted operations (single character edits):

substitution: replace a letter by another letter

deletion: remove a letter

insertion: insert a letter at a given position

TACA $\rightarrow$ TATATA

operation	result	alignment	cost
T$\rightarrow$T	TACA	(T,T)	0
A$\rightarrow$A	TACA	(A,A)	0
insert T	TATCA	(-,T)	1
insert A	TATACA	(- ,A)	1
C$\rightarrow$T	TATATA	(C,T)	1
A$\rightarrow$A	TATATA	(A,A)	0

see also Crochemore, Hancart, Lecroq : Figs. 7.1, 7.2

$$Lev(TACA, TATATA) = 3 \quad \text{cost} \begin{pmatrix} T & A & - & - & C & A \\ T & A & T & A & T & A \end{pmatrix} = 3$$

DEFINE CUSTOM COSTS FOR EACH OPERATION AND LETTER CHANGE

MATCH SCORE, MISMATCH PENALTY

The approach (EG Ch. 6.4, MM Ch. 5)

Define scores for matches and penalties for mismatches
(biology: substitution matrices, gap cost function)

Align two sequences by finding transformation steps minimizing
$Lev(Seq1, Seq2)$ under these scores
(dynamic programming: Needleman–Wunsch algorithm)

Gap cost function: $\delta(l)$, e.g. linear $\delta(l) = -ld$ (gap of length l)

Needleman–Wunsch (EG Ch. 6.4.2, MM Ch. 5.3–4)

Input: two sequences $\mathbf{x} = X_1 X_2 \dots X_m$ and $\mathbf{y} = Y_1 Y_2 \dots Y_n$

denote: $\mathbf{x}_{1i} = X_1 X_2 \dots X_i$

define: $B(i,j)$ score of the best alignment between $\mathbf{x}_{1i}$ and $\mathbf{y}_{1j}$

initial conditions: $B(i,0) = -id$, $B(0,j) = -jd$, $B(0,0) = 0$
recursion $w(X,Y)$ score/penalty of match/mismatch:

$$B(i,j) = \max \begin{cases} B(i-1,j-1) + w(X_i, Y_j) & \\ B(i-1,j) - d & \text{gap in sequence } \mathbf{y} \\ B(i,j-1) - d & \text{gap in sequence } \mathbf{x} \end{cases}$$

Complexity: O(mn)

Example

$w(x,x) = 1,\ w(x,y) = -1\ x \neq y,\ d = 1$

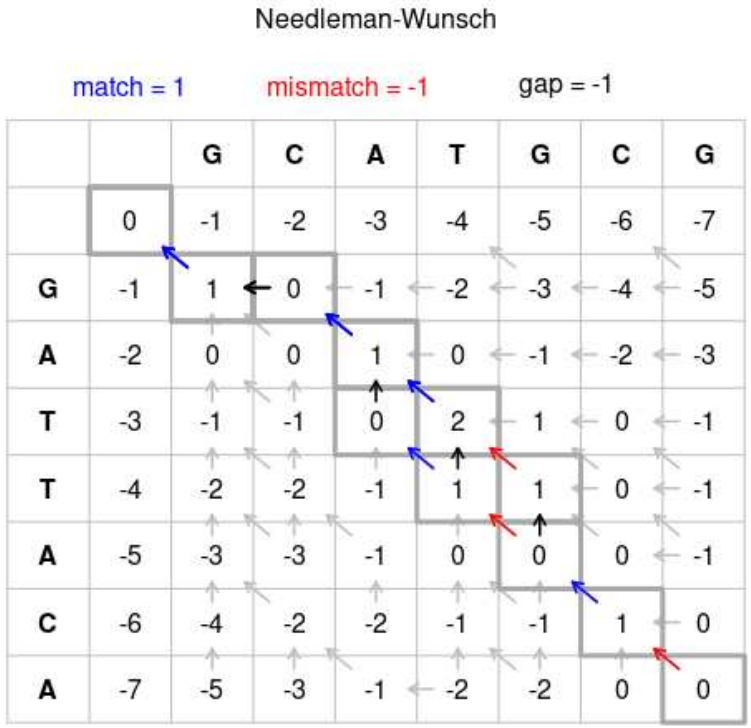
$\mathbf{x} = GCATGCG,\ \mathbf{y} = GATTACA$

Alignments with score$= 0$

$$\begin{array}{cccccccc} G & C & A & T & G & - & C & G \\ G & - & A & T & T & A & C & A \end{array}$$

$$\begin{array}{cccccccc} G & C & A & - & T & G & C & G \\ G & - & A & T & T & A & C & A \end{array}$$

$$\begin{array}{cccccccc} G & C & A & T & - & G & C & G \\ G & - & A & T & T & A & C & A \end{array}$$



https://commons.wikimedia.org/wiki/File:Needleman-Wunsch_pairwise_sequence_alignment.png, by Slowkow, public domain
code: https://gist.github.com/slowkow/508393, Kamil Slowikowski

see also Ewens, Grant: Fig. 6.1 (there $d = 2$)

Fitting one sequence into another (EG Ch. 6.4.3)

Input: two sequences $\mathbf{x} = X_1 X_2 \ldots X_m$ and $\mathbf{y} = Y_1 Y_2 \ldots Y_n$ but $n \gg m$ ($\mathbf{x}$ is to be aligned to a subsequence of $\mathbf{y}$)

$$\text{find}: \qquad \max\{B(\mathbf{x}, \mathbf{y}_{k,j}) : 1 \leq k \leq j \leq n\}$$

define $F(i,j) = \max\{B(\mathbf{x}_{1,i}, \mathbf{y}_{k,j}) : 1 \leq k \leq j\}$

initial conditions: $F(i,0) = -id$, $B(0,j) = 0$
recursion:

$$F(i,j) = \max \begin{cases} F(i-1, j-1) + w(X_i, Y_j) & \\ F(i-1, j) - d & \text{gap in sequence } \mathbf{y} \\ F(i, j-1) - d & \text{gap in sequence } \mathbf{x} \end{cases}$$

Variations

Other gap models (EG Ch. 6.4.5)

gap open penalty: $\delta(l) = -d - (l-1)e$

d: gap open, e: gap extenstion

Ends–free alignment (MM Ch. 5.5)

$B(i,0) = B(0,j) = 0$

(sequences of significantly different lengths)

R packages (BioConductor)

- msa
- DECIPHER
- DIAlignR

BONUS POINT(S): implement a package that plots the matrix, with paths resulting from an alignment of two sequences
FIRST DISCUSS YOUR IDEAS AND POSSIBLE POINTS

$w(x, y)$ (MM Ch. 4.3)

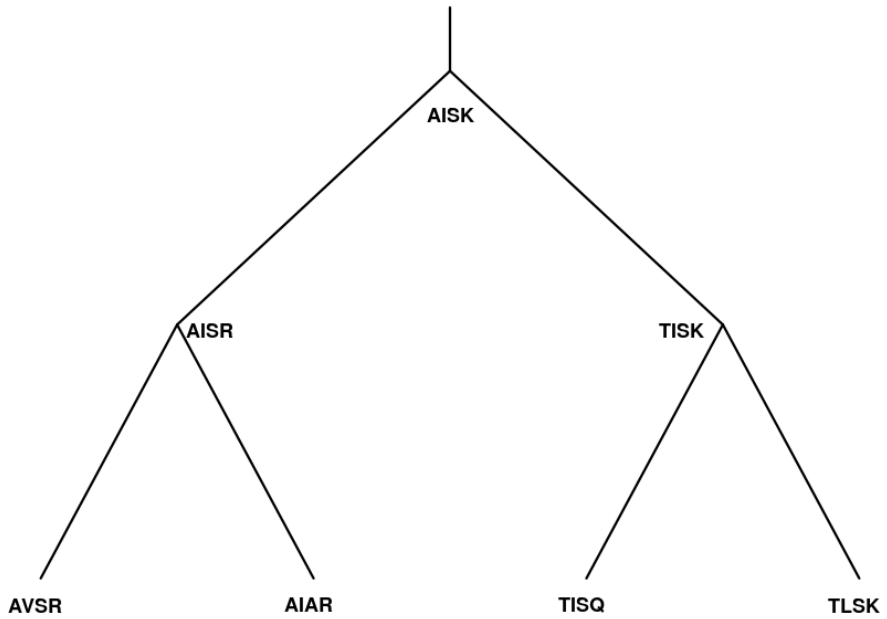
"Physical and chemical properties of replacement residues (amino acids) should be nearly the same as of the original one."

Radically different ones can compromise the protein's function.

Idea: Make a table of similarities of amino acids

Not possible: from sequence data see what replacements occured.

Nucleotide level (gene): synonymous changes "should" be neutral



	A	I	K	L	Q	R	S	T	V
A	0	0	0	0	0	0	1	1	0
I	0	0	0	1	0	0	0	0	1
K	0	0	0	0	1	1	0	0	0
L	0	1	0	0	0	0	0	0	0
Q	0	0	1	0	0	0	0	0	0
R	0	0	1	0	0	0	0	0	0
S	1	0	0	0	0	0	0	0	0
T	1	0	0	0	0	0	0	0	0
V	0	1	0	0	0	0	0	0	0

PAM matrix (EG Ch. 6.5.3, MM Ch. 4.4–4.10)

Accepted Point Mutations: collect closely related protein sequence and record matrix (**A**) of frequencies of point substitutions

Margaret Dayhoff (1978) 1572 substitutions recorded
(MM Fig. 4–4)

Jones, Taylor, Thornton (1992) 59190 substitutions recorded
(similar, MM Fig. 4–5)

PAM Matrix, Dayhoff (1978)

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

`https://github.com/brouwern/dayoff` GNU GPL

Margaret Dayhoff recorded 1572 substitutions

Fig 80 in M.O. Dayhoff, R.M. Schwartz RM, B.C. Orcutt BC 1978. A model of Evolutionary Change in Proteins. Atlas of protein sequence and structure (vol. 5, suppl 3ed.). Washington, DC.: National Biomedical Research Foundation. p.345-358.

D.T. Jones, W.R. Taylor, J.M Thornton JM 1992. The rapid generation of mutation data matrices from protein sequences. Comp. Appl. Biosci. 8:275-282, recorded 59190 substitutions (their Tab. I; MM Fig. 4.5)

PAM matrix (EG Ch. 6.5.3, MM Ch. 4.4–4.10)

Relative mutability: probability that one amino acid will mutate into another in a short period of time
$m(j) = \sum_{i \neq j} \mathbf{A}_{i,j} / n(i)$, $n(i)$ number of observed amino acids i

$$M_1(i,j) = \lambda m(i) \mathbf{A}_{i,j} / (\sum_k \mathbf{A}_{i,k}), i \neq j$$

$$M_1(i,i) = 1 - \lambda m_i$$

AISKTISK
AITKKISK

Amino acid	A	I	K	S	T
Changes	0	0	1	1	2
Freq	2	4	5	3	2
Rel. mutability	0	0	$\frac{1}{5}$	$\frac{1}{3}$	1

λ solves (99% sequence identity after mutation)

$$0.99 = 1 - \lambda \sum_j \frac{n(j)}{N} m(j) \quad N: \text{ total number of observed amino acids}$$

$M_n = M_1^n$, M_n: PAM$_n$ matrix, *PAM*250 is often used
$w_n(i,j) = 10 \log_{10}(\frac{M_n(i,j)}{n(j)/N})$

BLOSUM matrix (EG 6.5.2, MM Ch. 4.11)

BLOcks SUbstitution Matrix,
align multiple short, related sequences
Count the number of (ordered) pairs of positions where one amino acid changes into another, $\tilde{\mathbf{A}}$ matrix

$$\mathbf{Q}(i,j) = \tilde{\mathbf{A}} / \sum_{i,j} \tilde{\mathbf{A}}(i,j)$$

$$\tilde{\mathbf{R}}_{i,j} = \mathbf{Q}_{i,j} / ((n(i)/N)(n(j)/N))$$

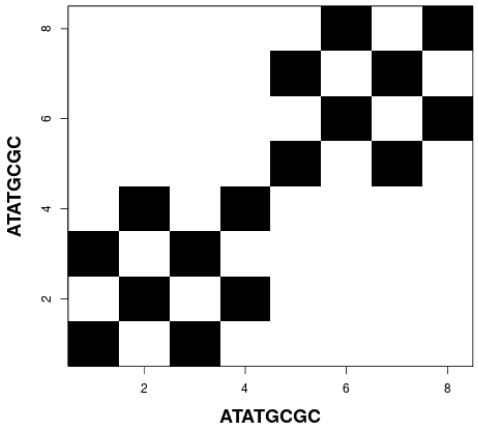
$$w(i,j) = 2 \log_2 \tilde{\mathbf{R}}_{i,j}$$

A	I	S	K	T	I	S	K
A	R	S	K	T	V	S	K
A	R	S	K	T	V	S	Q
A	R	S	K	T	V	S	Q
A	R	S	K	T	V	S	K
A	R	S	K	T	I	S	K
A	R	S	K	T	I	S	K
A	R	T	K	K	I	S	K

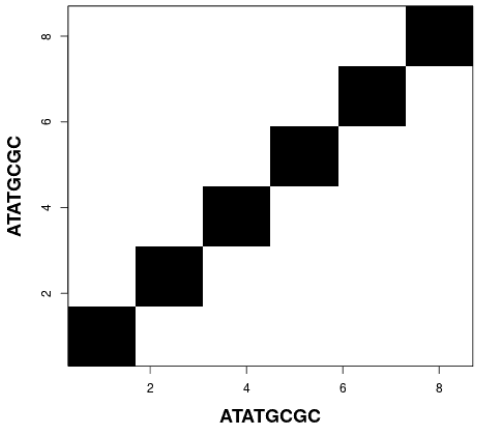
	A	I	K	Q	R	S	T	V
A	56							
I		12			7			16
K			86	12			7	
Q			12	2				
R		7			42			
S						98	7	
T			7			7	42	
V		16						12

Dot plots (MM Ch. 5.2)

`seqinr::dotPlot()`

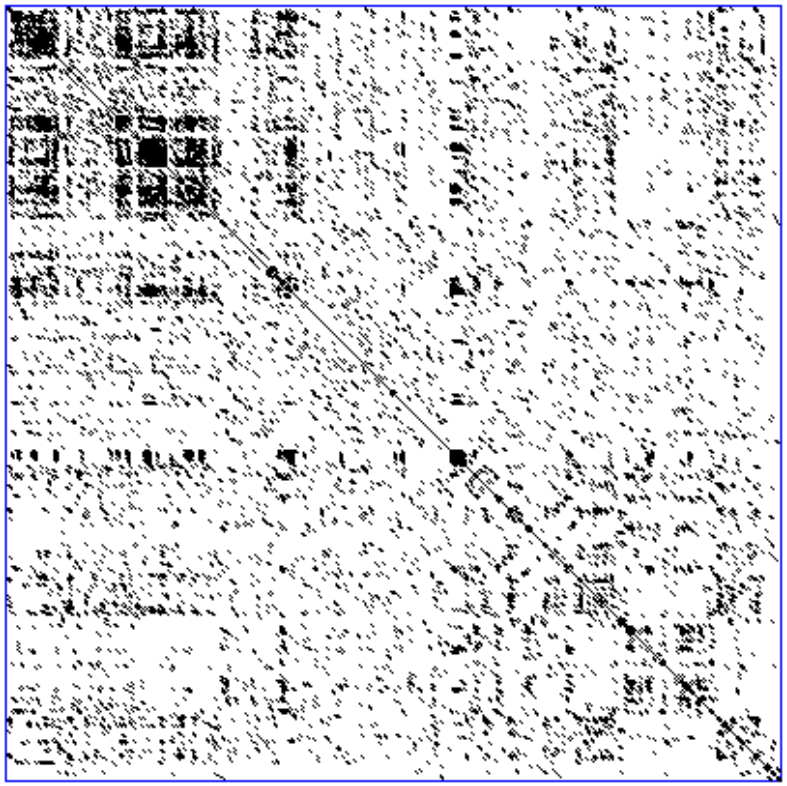


sliding window:1 `nmatch=wsize=1`



sliding window:3 `nmatch=wsize=3`

Human zincfinger TF versus itself dotplot.



`https://commons.wikimedia.org/w/index.php?curid=949276`, by Opabinia regalis, CC BY-SA 3.0

Multiple sequences (EG Ch. 6.6, MM Ch. 6.3)

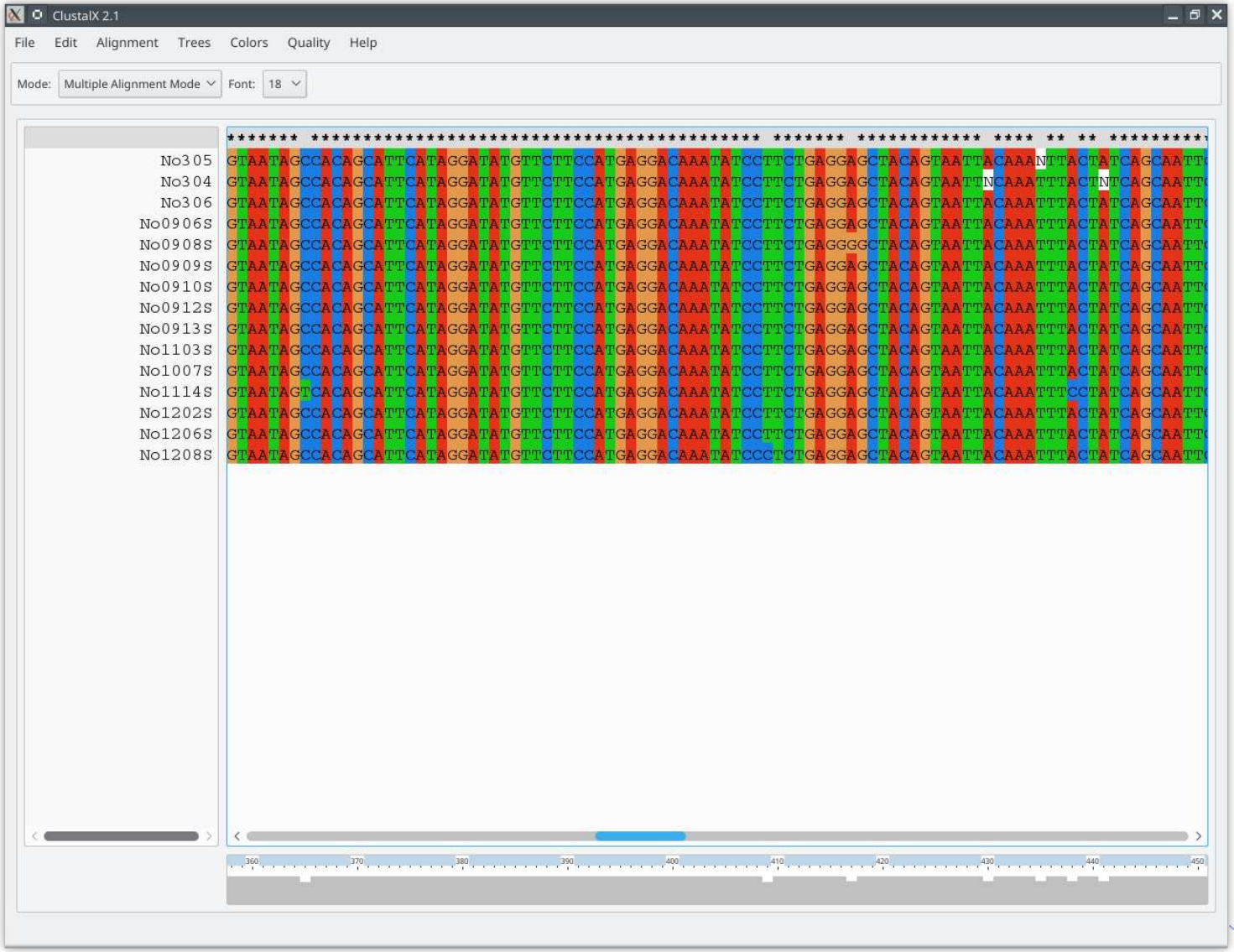
Needleman–Wunsch algorithm can be directly generalized but impractical for large number of sequences (more than 20 in 2005)

Heuristics developed e.g. CLUSTAL W
(`http://www.clustal.org/`);
(`https://www.ebi.ac.uk/Tools/msa/clustalo/`)

1. Create all pairwise alingnments and calculate scores
2. Create a guide tree (neighbour–joining) based on the scores
3. Align sets of sequences going from tips to root See also Ch. 6 in DEKM

ClustalX (woodmouse.fasta from phangorn)

http://www.clustal.org/



Variations: Smith–Waterman algorithm (EG Ch. 6.4.4, MM Ch. 5.6)

Local alignment: given two sequences, which sub–sequences have the highest–scoring alignment

Smith–Waterman: linear gap penalty

$$\text{find}: \quad \max\{B(\mathbf{x}_{h,i}, \mathbf{y}_{k,j}) : 1 \le h \le i \le m, 1 \le k \le j \le n\}$$

define : $L(i,j) = \max\{0, B(\mathbf{x}_{h,i}, \mathbf{y}_{k,j}) : 1 \le h \le i, 1 \le k \le j\}$

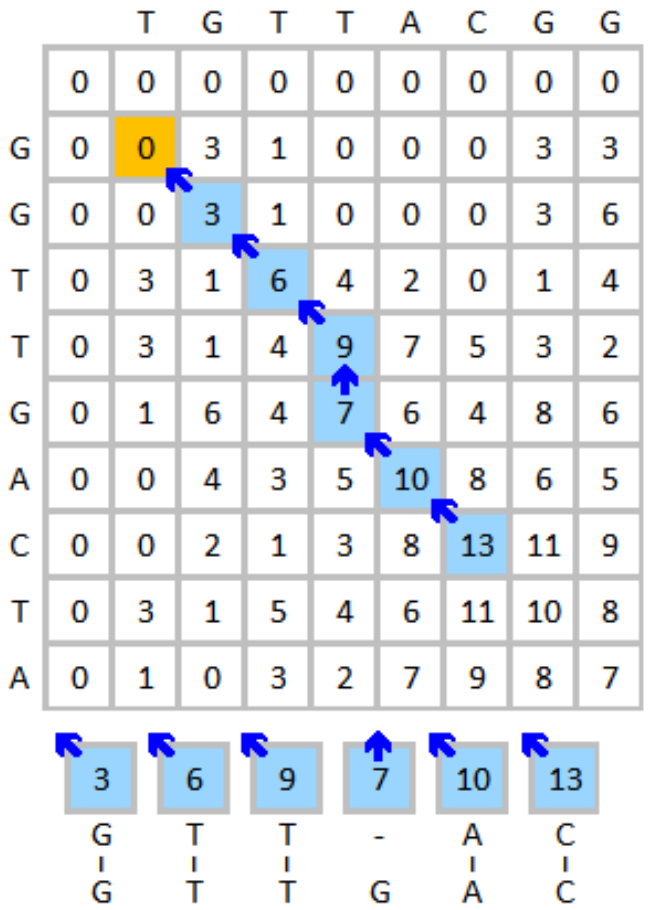
idea: $L(i,j)$ negative then remove first part of alignment

initial conditions: $L(i,0),\ L(0,j) = 0$

recursion:

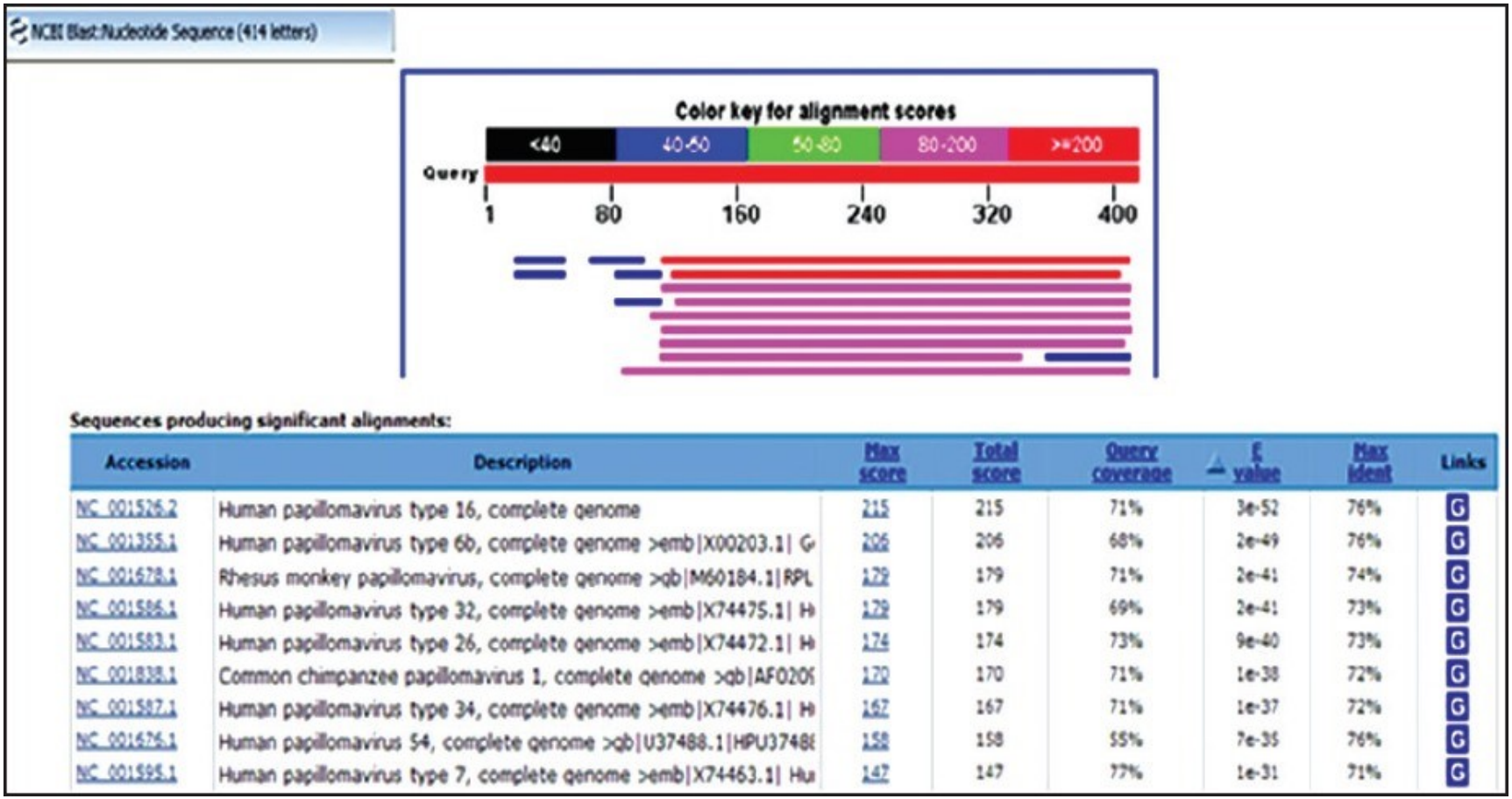
$$L(i,j) = \max \begin{cases} 0 & \\ L(i-1,j-1) + w(X_i, Y_j) & \\ L(i-1,j) - d & \text{gap in sequence } \mathbf{y} \\ L(i,j-1) - d & \text{gap in sequence } \mathbf{x} \end{cases}$$

Example



$d = 2$, match:$+3$, mismatch:-3

BLASTing (MM Ch. 6.2)



see also NCBI user's manual: `https://www.ncbi.nlm.nih.gov/books/NBK279690/`

BLAST phases (MM Fig. 6.2, https://en.wikipedia.org/wiki/BLAST_(biotechnology))

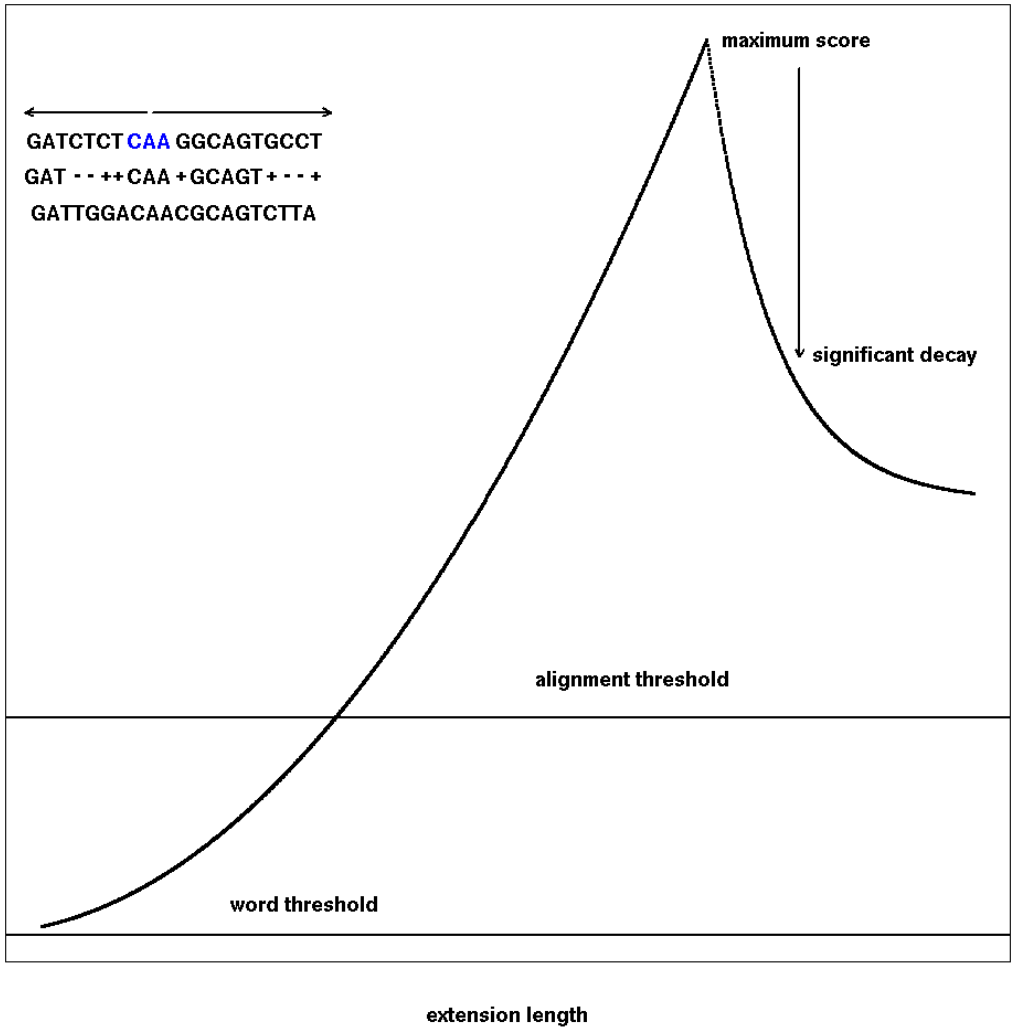
high–scoring word: (short subsequence of given length(s))

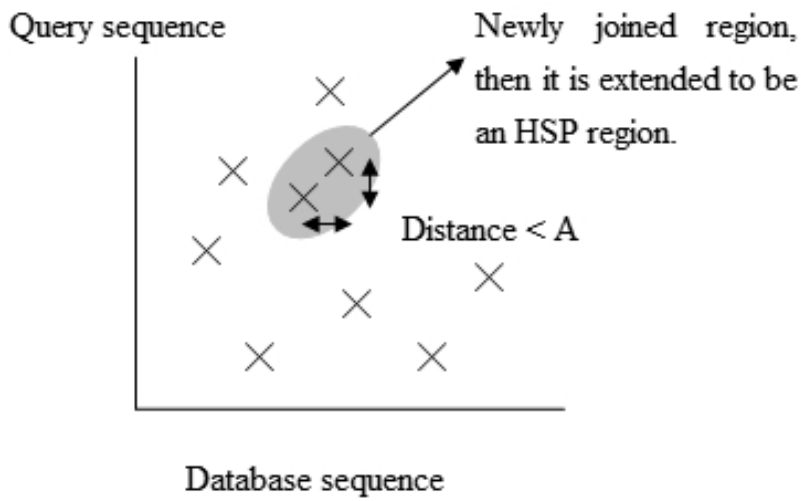
Phase 1: Remove regions of low complexity (i.e., those with few different elements)
Phase 2: Create a list of k (longer than word threshold) letter *high scoring* words derived from query sequence.

Phase 3: Using a chosen substitution (scoring) matrix (BLOSUM62) create a list of *high scoring pairs* (**HSP**; query/database; longer than word threshold) by searching through the sequence databse (against which query is *blasted*).
Phase 4: Extend each HSP, both ways, until a significant decay in its score is reached.
Phase 5: Reduce back each HSP until the maximum score for the pair is attained.

BLAST phases (MM Figs. 6.3, 6.4)

https://commons.wikimedia.org/w/index.php?curid=33587927,





by DISP, public domain

E–value (EG Ch. 10.3.4, 10.4, MM p. 104)

Multiple testing issue: multiple hits between query and database sequence, where to cut off?
H_0: The two sequences are unrelated
(each pair of aligned amino acids is independent)

Score $\mathbf{S}(j, k)$ (assumed substituion matrix) assigned to each position where amino acids j, k are observed

accumulated score(i): sum of scores from positions $1, \ldots, i$
(random walk)

EG Ch. 10.2.5–10.3.3 p–value derivations

E–value

E: number of hits of score at least s expected by chance
(given the query and database sizes)

$$E = 2N_1' N_2' K e^{-\lambda s}$$

N_1', N_2' sizes of query, database sequences corrected for edge effects
K, λ: estimated model parameters (from **S**)
s: similarity score

D: database size

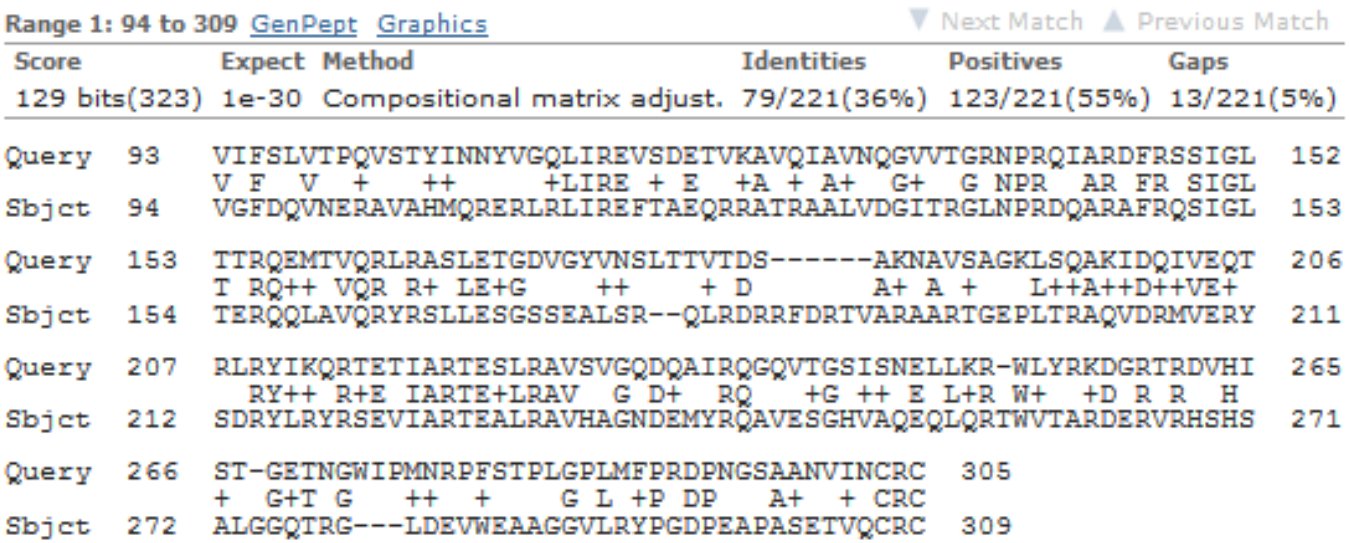
$$\text{Expect} = \frac{(1 - e^E)D}{N_2} \qquad \text{p} - \text{value} = 1 - e^{-\text{Expect}}$$

BLAST's applications (MM Tab. 6.1)

IDEA: Function based on sequence similarity to other sequences

- Identify genes in newly sequenced genomes.
- Identify non–coding DNA sequences.
- Identify distantly related proteins (similar substructures).
- Predict protein structures.
- Predict protein functions
  (similar sequences—similar structures).

BLAST printout (EG Ch. 10.5.1)



Range 1: 94 to 309 GenPept Graphics ▼ Next Match ▲ Previous Match

Score	Expect	Method	Identities	Positives	Gaps
129 bits(323)	1e-30	Compositional matrix adjust.	79/221(36%)	123/221(55%)	13/221(5%)

```
Query  93   VIFSLVTPQVSTYINNYVGQLIREVSDETVKAVQIAVNQGVVTGRNPRQIARDFRSSIGL  152
            V F  V  +   ++     +LIRE + E  +A + A+  G+  G NPR  AR FR SIGL
Sbjct  94   VGFDQVNERAVAHMQRERLRLIREFTAEQRRATRAALVDGITRGLNPRDQARAFRQSIGL  153

Query  153  TTRQEMTVQRLRASLETGDVGYVNSLTTVTDS------AKNAVSAGKLSQAKIDQIVEQT  206
            T RQ++ VQR R+ LE+G    ++    + D        A+ A +   L++A++D++VE+
Sbjct  154  TERQQLAVQRYRSLLESGSSEALSR--QLRDRRFDRTVARAARTGEPLTRAQVDRMVERY  211

Query  207  RLRYIKQRTETIARTESLRAVSVGQDQAIRQGQVTGSISNELLKR-WLYRKDGRTRDVHI  265
              RY++ R+E IARTE+LRAV  G D+  RQ   +G ++ E L+R W+  +D R R  H
Sbjct  212  SDRYLRYRSEVIARTEALRAVHAGNDEMYRQAVESGHVAQEQLQRTWVTARDERVRHSHS  271

Query  266  ST-GETNGWIPMNRPFSTPLGPLMFPRDPNGSAANVINCRC  305
            +  G+T G   ++  +    G L +P DP   A+  + CRC
Sbjct  272  ALGGQTRG---LDEVWEAAGGVLRYPGDPEAPASETVQCRC  309
```

https://cpt.tamu.edu/bich464/doc/C2.html, Center for Phage Technology, CC BY-SA 4.0

BioPerl for graphically presenting BLAST's output

https://bioperl.org/howtos/BioGraphics_HOWTO.html

Questions?