

Global alignments – review

- Take two sequences: $X[j]$ and $Y[j]$
- The best alignment for $X[1..i]$ and $Y[1..j]$ is called $M[i, j]$
- Initiation: $M[0,0]=0$
- Apply the equation
- Find the alignment with backtracing

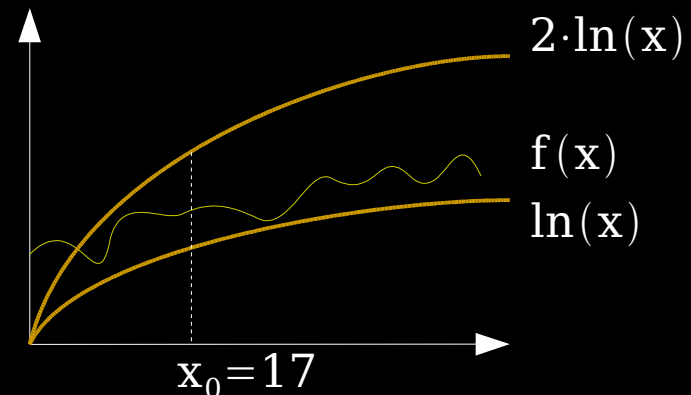
$$M[i, j] = \max \begin{cases} M[i-1, j-1] + 1 \\ M[i, j-1] - 2 \\ M[i-1, j] - 2 \end{cases}$$

		X[j]							
		0	1	2	3	4	5	6	
		-	G	A	G	T	G	A	
0	-	0							
1	G								
2	A								
3	G								
4	G								
5	C								
6	G								
7	A								2

Algorithm time/space complexity – Big-O Notation

- ♦ a simple description of complexity:
 - constant $O(1)$, linear $O(n)$, quadratic $O(n^2)$, cubic $O(n^3)$...
- ♦ asymptotic upper bound
- ♦ read: “order of”

$$f(n) = O(\underbrace{g(n)}_{\text{simple, e.g. } n^2}) \text{ iff. } \exists_{x_0, c} \forall_{x \geq x_0} f(x) \leq c g(x)$$



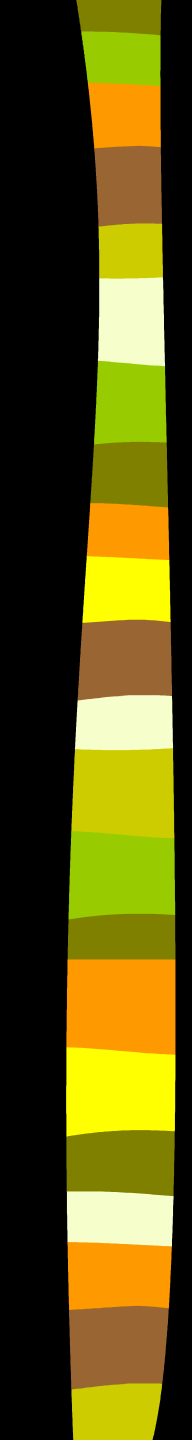
Big-O Notation

Example

- Time complexity of global alignment:

$$M[i, j] = \max \begin{cases} M[i-1, j-1] \pm 1 \\ M[i, j-1] - 2 \\ M[i-1, j] - 2 \end{cases}$$

$$\underbrace{n+m-1}_{\text{init}} + \underbrace{10nm}_{\text{calc M}} + \underbrace{1}_{\text{print}} = O(nm)$$



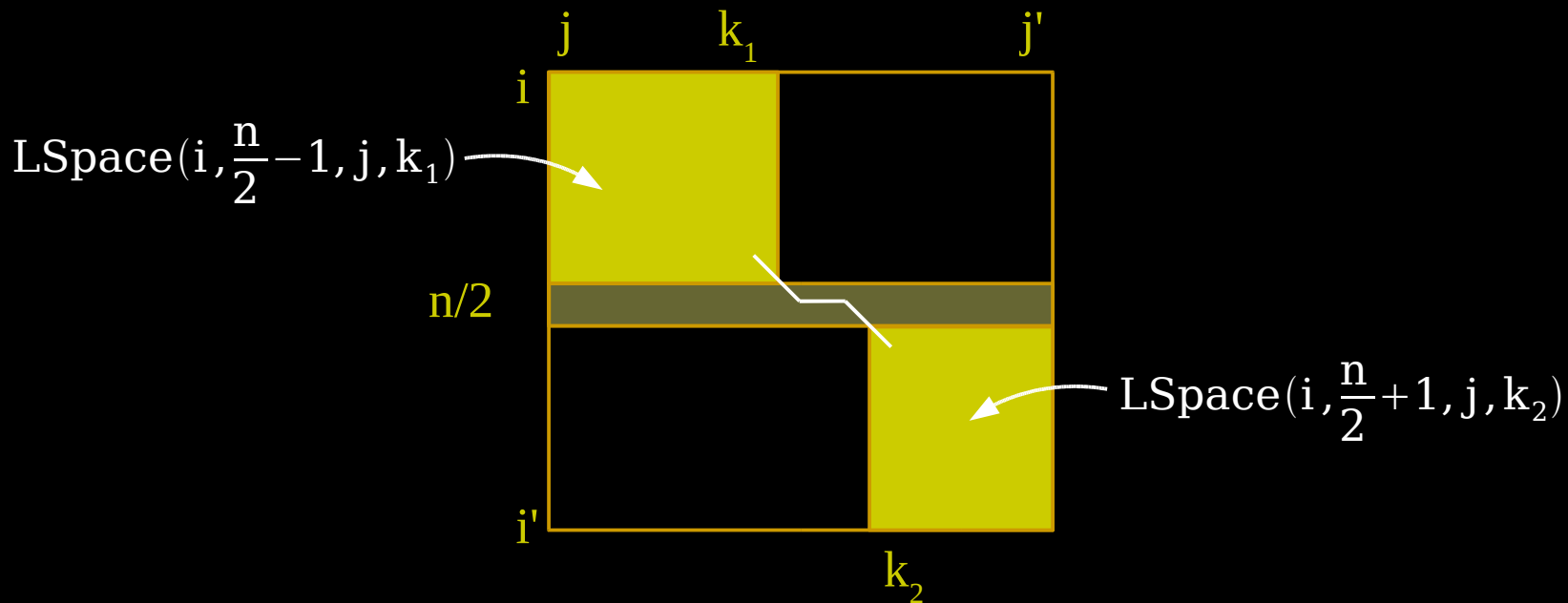
Global alignment – linear space

$$M[i, j] = \max \begin{cases} M[i-1, j-1] \pm 1 \\ M[i, j-1] - 2 \\ M[i-1, j] - 2 \end{cases}$$

- ♦ We need $O(nm)$ time, but only $O(m)$ space
 - how?
- ♦ problem with backtracking

Global alignment – linear space, recursion

LSpace(i, i', j, j'):



- ♦ space complexity: $O(m)$

Global alignment – linear space, algorithm

LSpace(i, i', j, j'):

return if area(i, i', j, j') empty

$$h := \frac{i' - i}{2}$$

calc. M using $O(m)$ memory

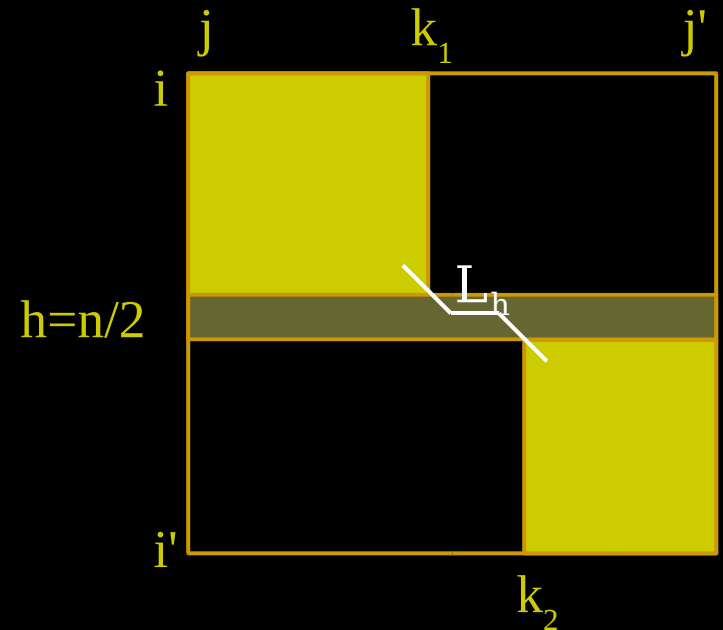
plus find path L_h

crossing the row h

LSpace($i, h-1, k_1, j'$)

print L_h

LSpace($i, h+1, k_2, j'$)



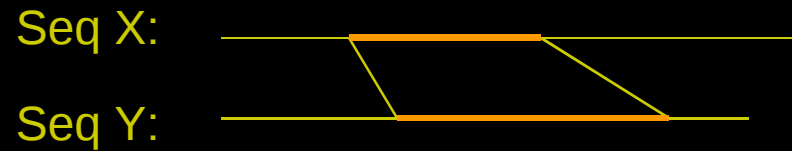
- time complexity: $\sum_{i=0}^{\log_2 n} \frac{nm}{2^i} \leq 2nm$

Alignments:

Local alignment

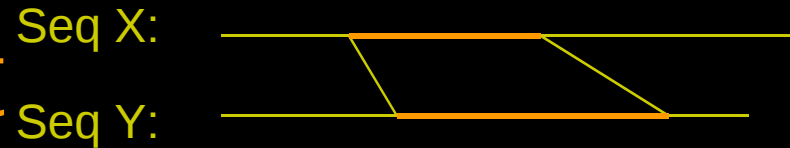


Local alignment: Smith–Waterman algorithm



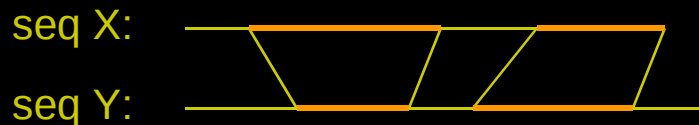
- ♦ What's local?
 - Allow only parts of the sequence to match
 - Locally maximal: can not make it better by trimming/extending the alignment

Local alignment

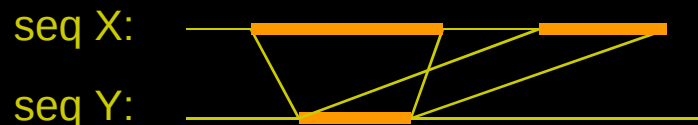


♦ Why local?

- Parts of sequence diverge faster
evolutionary pressure does not put constraints on *the whole* sequence



- Proteins have modular construction
sharing domains between sequences



Domains – example

Immunoglobulin domain

Representative ig domain proteins

[1A01_GORGO](#) [Gorilla gorilla gorilla (lowland gorilla)] class i histocompatibility antigen, gogo-a0101 alpha chain precursor



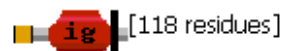
[ALC1_GORGO](#) [Gorilla gorilla gorilla (lowland gorilla)] ig alpha-1 chain c region



[AMAL_DROME](#) [Drosophila melanogaster (fruit fly)] amalgam protein precursor



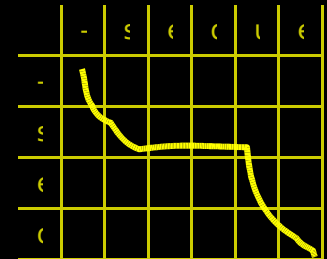
[B2MG_BOVIN](#) [Bos taurus (bovine)] beta-2-microglobulin precursor (lactollin)



Global → local alignment

- ♦ Take the *global* equation
- ♦ Look at the result of the *global* alignment

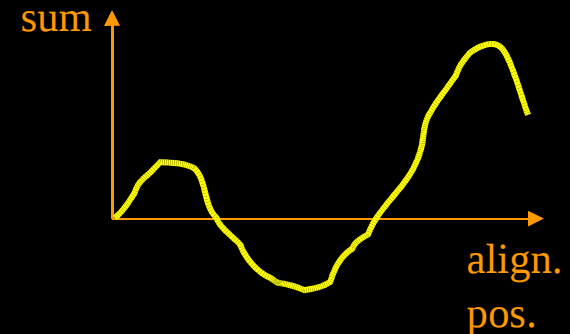
a) global align



b) retrieve the result

CAGCACTTGGATTCTCG-
CA-C-----GATTCGT-G

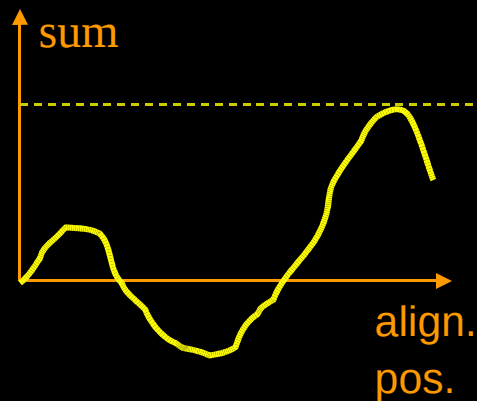
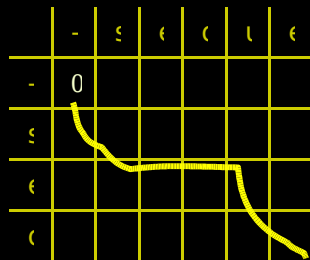
c) sum score along
the result



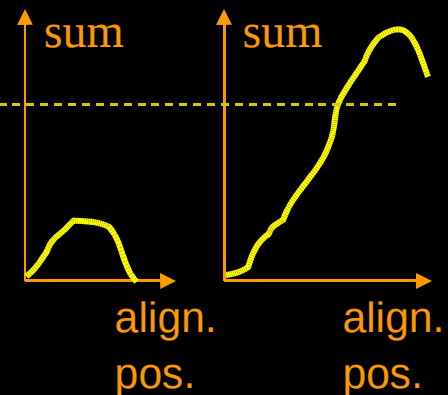
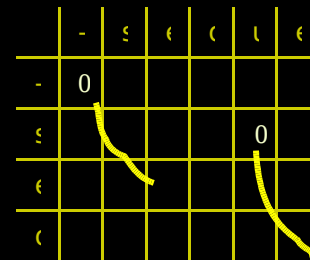
Local alignment – breaking the alignment

- ♦ A recipe
 - Just don't let the score go below 0
 - Start the new alignment when it happens
 - Where is the result in the matrix?

Before:



After:



Local alignment – the equation

$$M[i, j] = \max \begin{cases} M[i-1, j-1] + \text{score}(X[i], Y[j]) \\ M[i, j-1] - g \\ M[i-1, j] - g \\ 0 \end{cases}$$

Great contribution
to science!

- ♦ Init the boundaries with 0's
- ♦ Run the algorithm
- ♦ Read the maximal value from anywhere in the matrix
- ♦ Find the result with backtracking

	-	ξ	ε	(	l	ε
-						
ξ						
ε						
(						

Finding second best alignment

- We can find the best *local* alignment in the sequence
- But where is the second best one?

Scoring:

1 for match

-2 for a gap

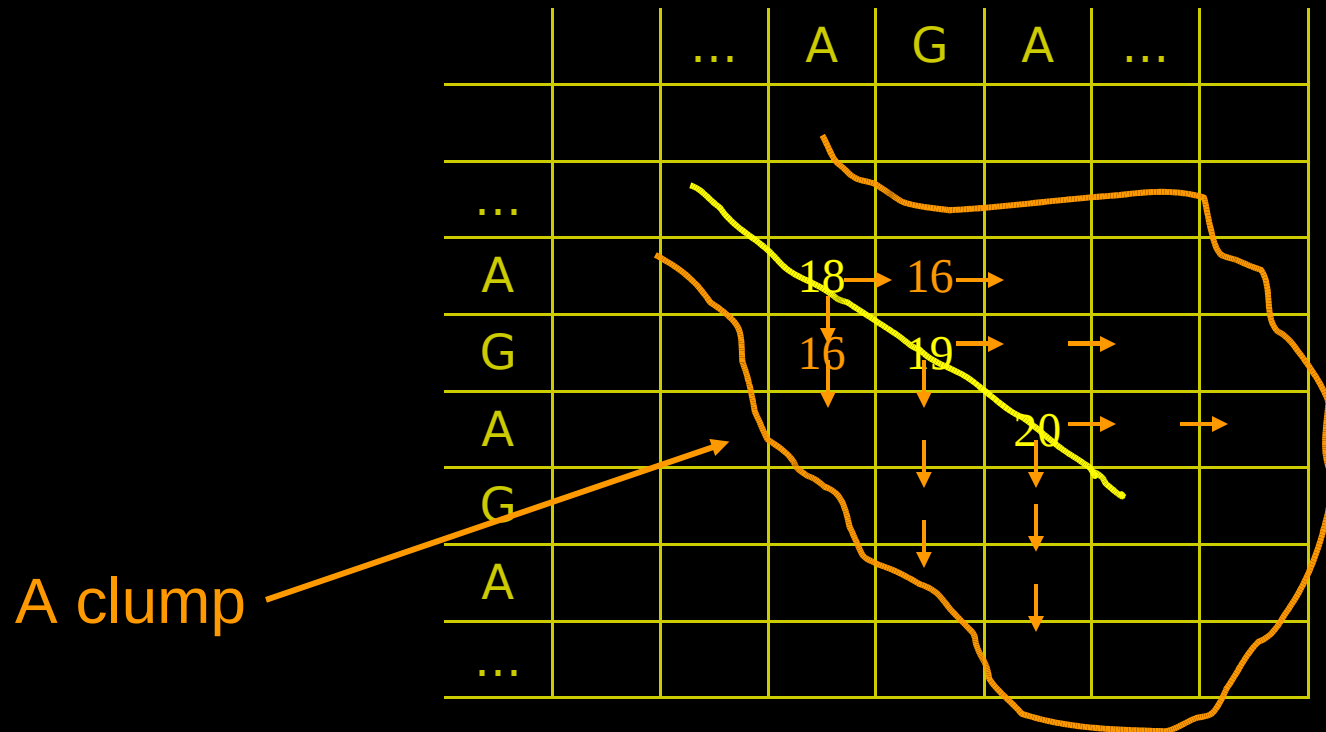
Best alignment

A clump

		...	A	G	A	...	
...							
A			18	16	14		
G			16	19	17	15	
A			14	17	20	18	16
G				15	18		
A				13	16		
...							

Clump of an alignment

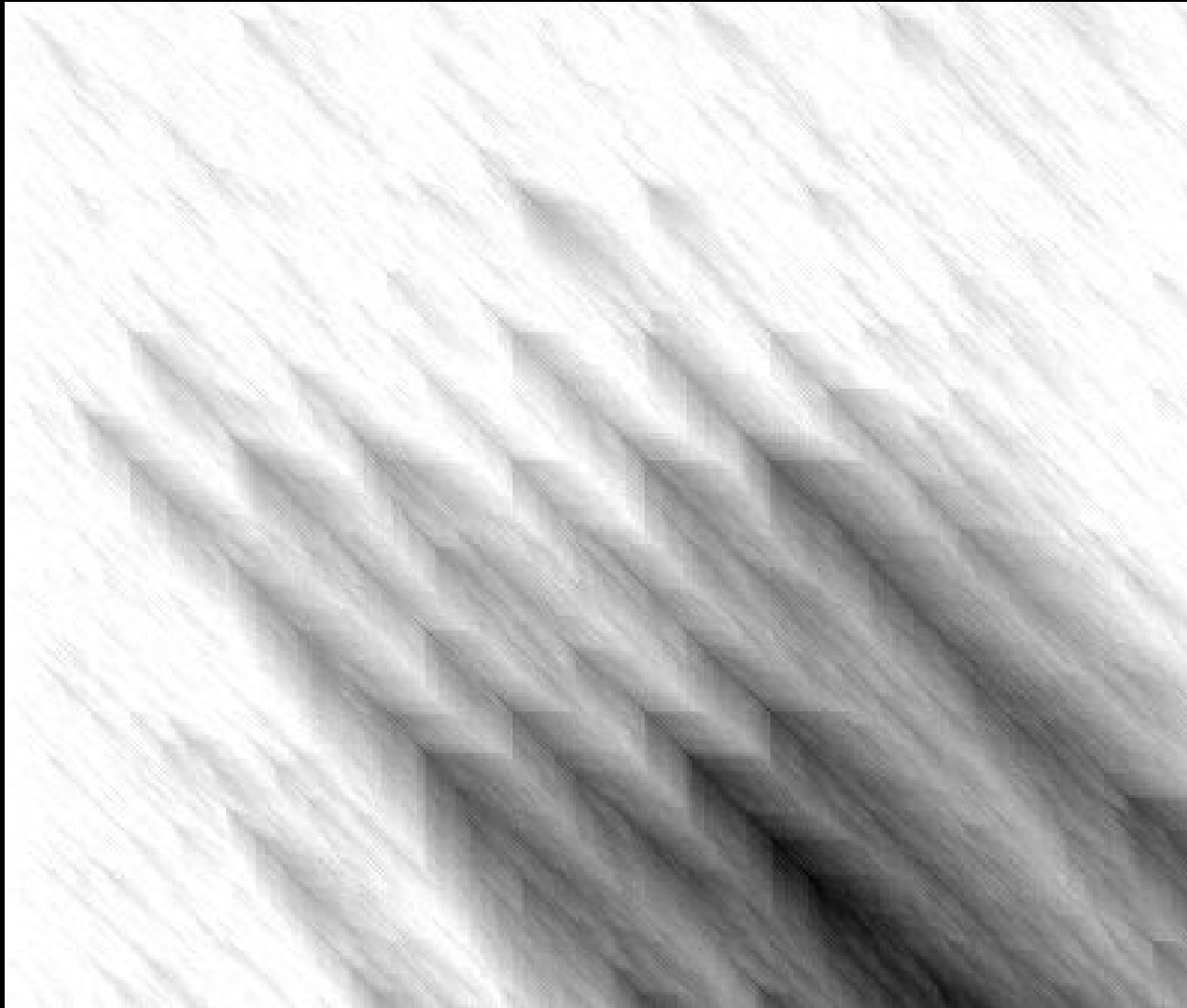
- Alignments sharing at least one aligned pair



Clumps

gene X

gene Y

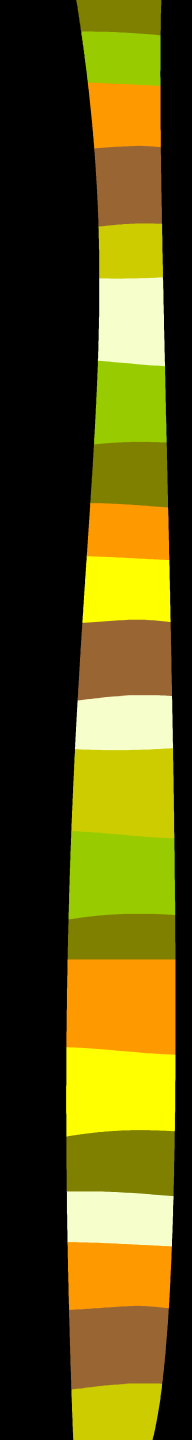


Finding second best alignment

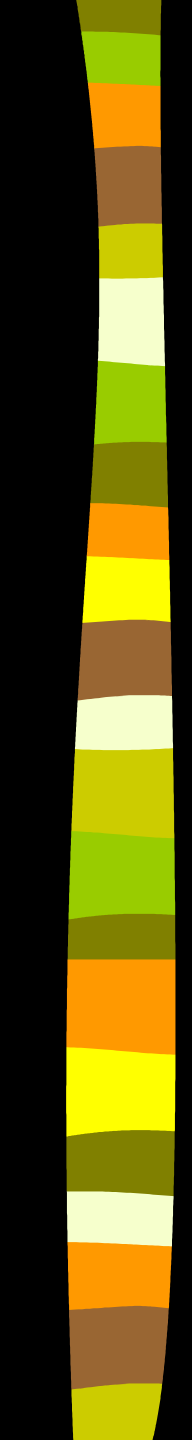
- Don't let any matched pair to contribute to the next alignment

“Clear” the best alignment

Recalculate the clump



		...	A	G	A	...	
...							
A			0	0	1		
G			0	0	0	0	
A			1	0	0	0	1
G				2	0		
A				0	3	1	
...					1		



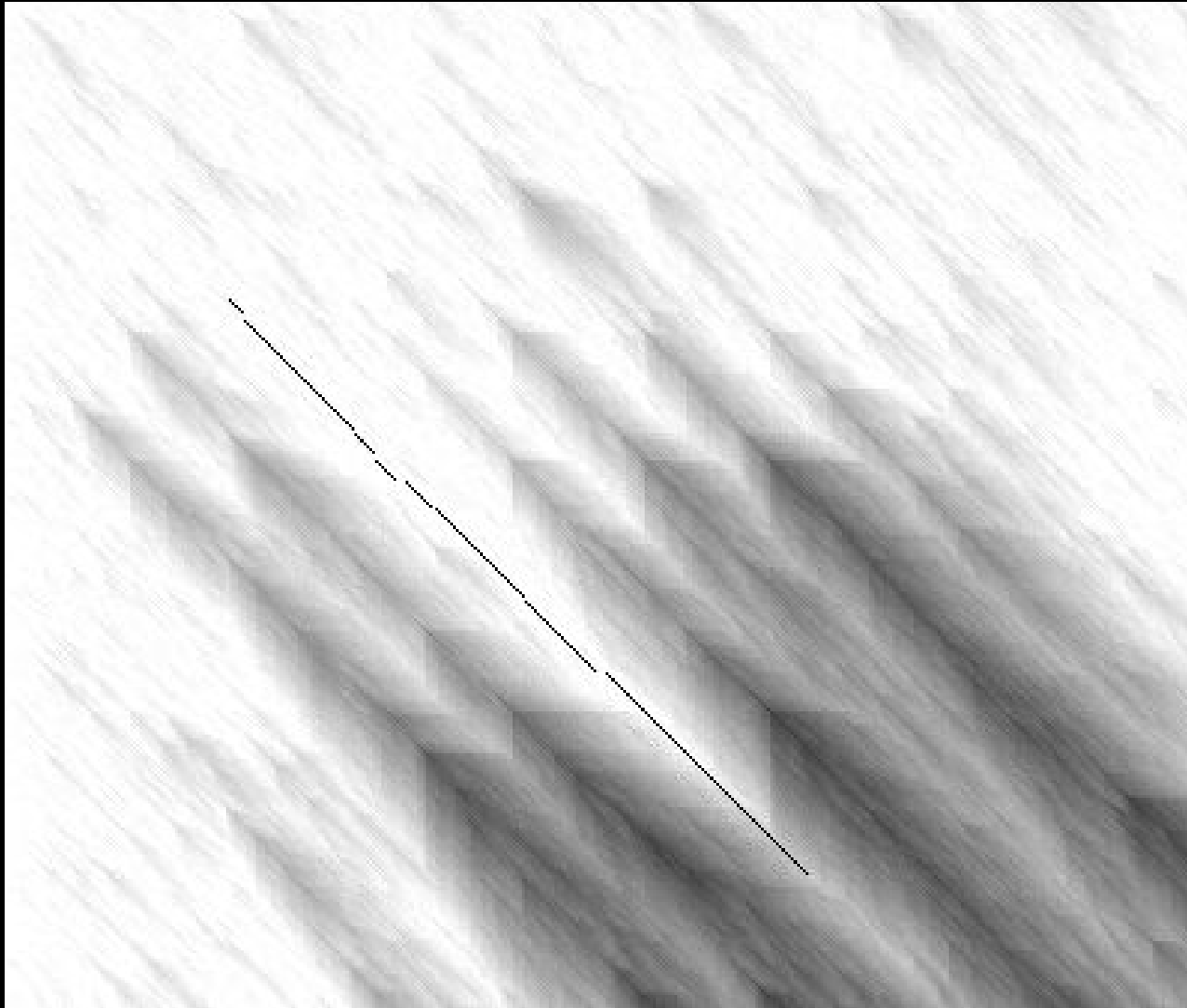
Extraction of local alignments – Waterman-Eggert algorithm

1. Repeat
 - a. Calc M without taking ☠ cells into account
 - b. Retrieve the highest scoring alignment
 - c. Set it's trace to ☠

Clumps

gene X

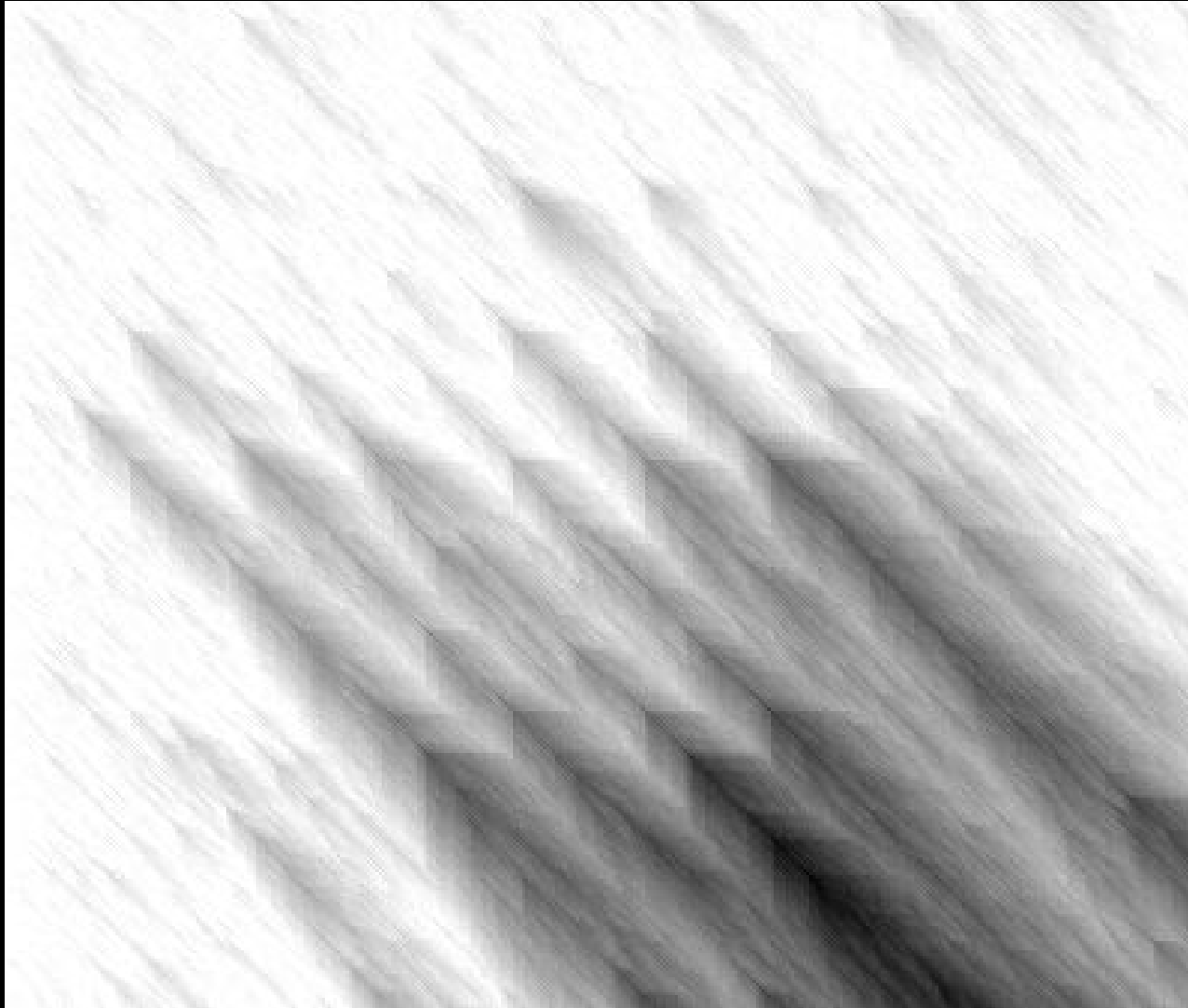
gene Y



Clumps

gene X

gene Y

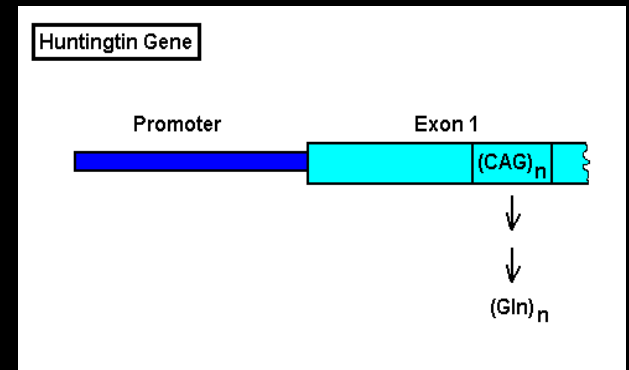


Low complexity regions

- ♦ Local composition bias
 - Replication slippage: e.g. triplet repeats
- ♦ Results in spurious hits
 - Breaks down statistical models
 - Different proteins reported as hits due to similar composition
 - Up to $\frac{1}{4}$ of the sequence can be biased

Huntington's disease

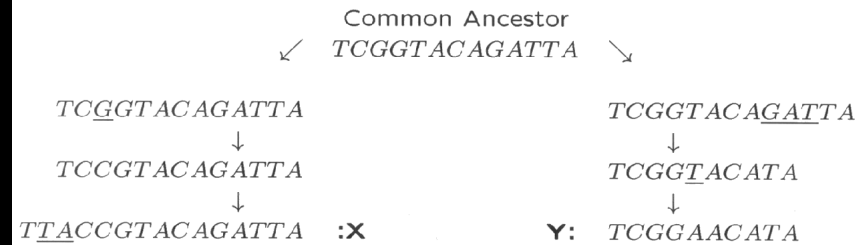
- *Huntingtin* gene of unknown (!) function
- Repeats #: 6-35: normal; 36-120: disease



Pitfalls of alignments

Alignment is not a reconstruction of the evolution
(common ancestor is *extinct* by the time of alignment)

Evolutionary History of Sequences



Viewed X to Y:



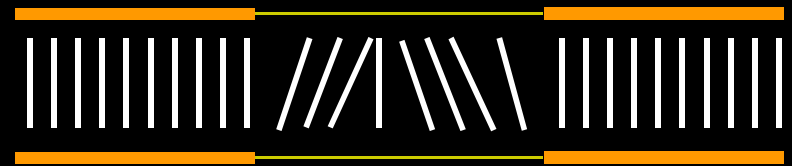
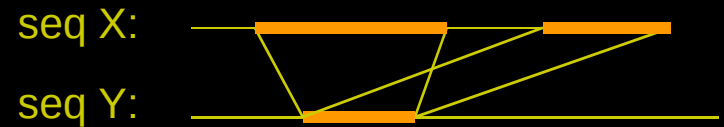
If we **line up** the parts that remain the same and those parts that have changed only by substitution, we get:

Seq. X:	T	T	A	C	G	T	A	C	A	G	A	T	T	A
Seq. Y:	T	-	-	G	G	A	A	C	A	-	-	-	T	A

This is called an **alignment**.

Pitfalls of alignments

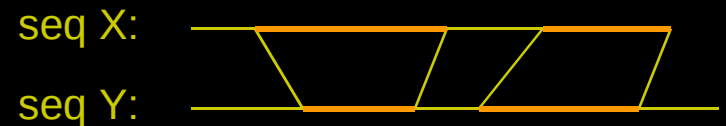
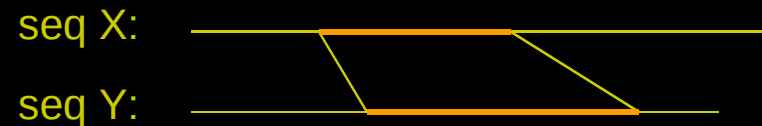
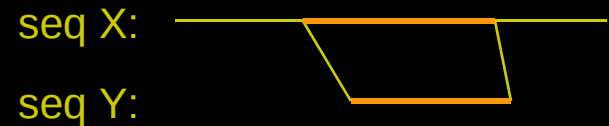
- ♦ Matches to the same fragment
- ♦ Arbitrary poor alignment regions



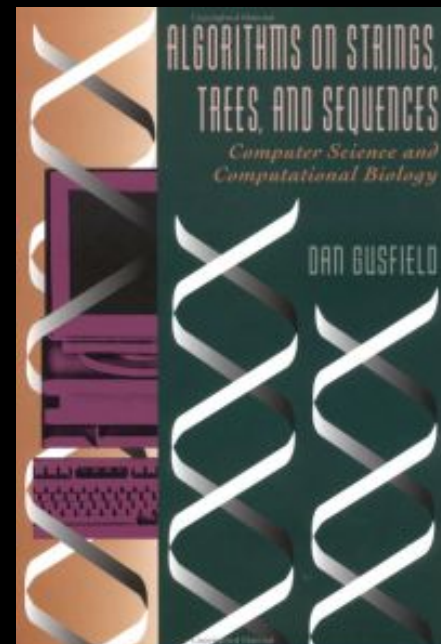
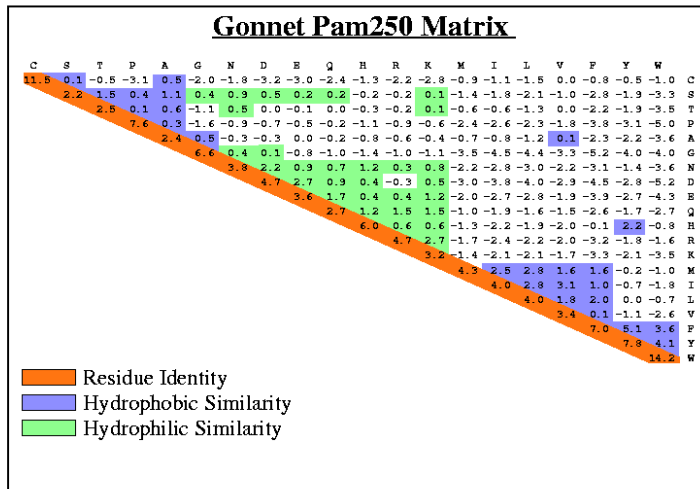
Summary

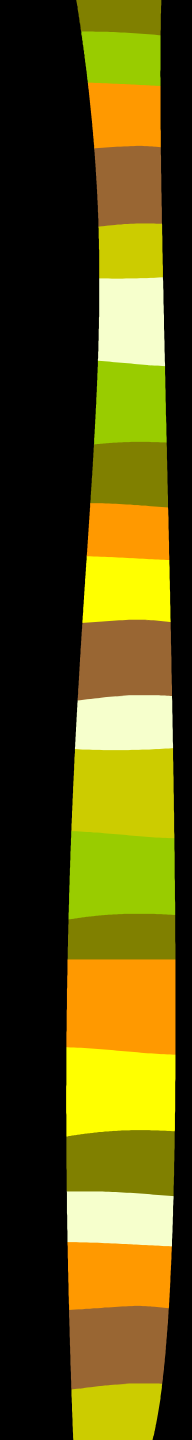
1. **Global**
a.k.a. Needleman-Wunsch algorithm
2. **Global-local**
3. **Local**
a.k.a. Smith-Waterman algorithm
4. **Many local alignments**
a.k.a. Waterman-Eggert algorithm

What's the number of steps in these algorithms?
How much memory is used?



Amino acid substitution matrices





Percent Accepted Mutations distance *and* matrices

- ♦ Accepted by natural selection
 - not lethal
 - not silent
- ♦ *Def.:* S_1 and S_2 are PAM 1 distant if on avg. there was one mutation per 100 aa
- ♦ *Q.:* If the seqs are PAM 8 distant, how many residues may be different?

PAM matrix

- Created from “easy” alignments
 - pairwise
 - gapless
 - 85% id

$f(\text{proline})$ – frequency of occurrence of proline

$f(\text{proline}, \text{valine})$ – frequency of substitution
proline with valine, for PAM 1

$$\text{i.e. } f(a, b) = \frac{\sum_{\text{aligns}} \text{count}(a, b)}{\sum_{\text{aligns } c, d \neq a, b} \text{count}(c, d)}$$

M – symmetric matrix,

$$\text{i.e. } M = \begin{bmatrix} f(a, a) & f(a, b) \\ f(b, a) & f(b, b) \end{bmatrix}$$

PAM matrix

- ♦ How to calculate M for PAM 2 distance?
 - Take more distant seqs
 - or extrapolate...

$$M^2 = \begin{bmatrix} f(a,a) & f(a,b) \\ f(b,a) & f(b,b) \end{bmatrix}^2 = \begin{bmatrix} f(a,a)f(a,a)+f(a,b)f(b,a) & f(a,a)f(a,b)+f(a,b)f(b,b) \\ \dots & \dots \end{bmatrix}$$

PAM *log odds* matrix

- ♦ Making of the PAM N matrix

$$\text{PAMN}[a,b] = \log_2 \frac{f(a)M^N[a,b]}{f(a)f(b)} = \log_2 \underbrace{\frac{M^N[a,b]}{f(b)}}_{\text{odds}} \underbrace{\phantom{\frac{M^N[a,b]}{f(b)}}}_{\log \text{ odds}}$$

observed

By chance alone

- ♦ Why log?
- ♦ Mutations and chance:
 - More freq: PAM N[a,b] > 0
 - Less freq: PAM N[a,b] < 0

PAM 250 matrix

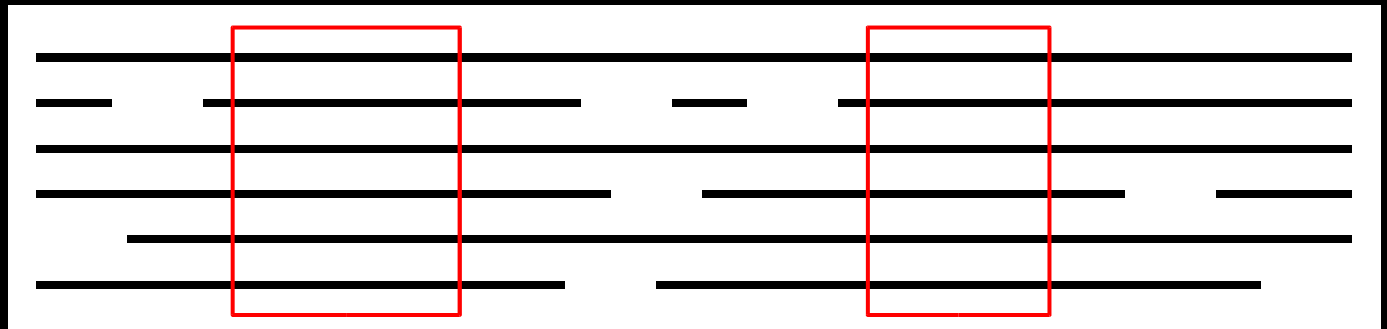
Gonnet Pam250 Matrix

C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
11.5	0.1	-0.5	-3.1	0.5	-2.0	-1.8	-3.2	-3.0	-2.4	-1.3	-2.2	-2.8	-0.9	-1.1	-1.5	0.0	-0.8	-0.5	-1.0	C
	2.2	1.5	0.4	1.1	0.4	0.9	0.5	0.2	0.2	-0.2	-0.2	0.1	-1.4	-1.8	-2.1	-1.0	-2.8	-1.9	-3.3	S
		2.5	0.1	0.6	-1.1	0.5	0.0	-0.1	0.0	-0.3	-0.2	0.1	-0.6	-0.6	-1.3	0.0	-2.2	-1.9	-3.5	T
			7.6	0.3	-1.6	-0.9	-0.7	-0.5	-0.2	-1.1	-0.9	-0.6	-2.4	-2.6	-2.3	-1.8	-3.8	-3.1	-5.0	P
				2.4	0.5	-0.3	-0.3	0.0	-0.2	-0.8	-0.6	-0.4	-0.7	-0.8	-1.2	0.1	-2.3	-2.2	-3.6	A
					6.6	0.4	0.1	-0.8	-1.0	-1.4	-1.0	-1.1	-3.5	-4.5	-4.4	-3.3	-5.2	-4.0	-4.0	G
						3.8	2.2	0.9	0.7	1.2	0.3	0.8	-2.2	-2.8	-3.0	-2.2	-3.1	-1.4	-3.6	N
							4.7	2.7	0.9	0.4	-0.3	0.5	-3.0	-3.8	-4.0	-2.9	-4.5	-2.8	-5.2	D
								3.6	1.7	0.4	0.4	1.2	-2.0	-2.7	-2.8	-1.9	-3.9	-2.7	-4.3	E
									2.7	1.2	1.5	1.5	-1.0	-1.9	-1.6	-1.5	-2.6	-1.7	-2.7	Q
										6.0	0.6	0.6	-1.3	-2.2	-1.9	-2.0	-0.1	2.2	-0.8	H
											4.7	2.7	-1.7	-2.4	-2.2	-2.0	-3.2	-1.8	-1.6	R
												3.2	-1.4	-2.1	-2.1	-1.7	-3.3	-2.1	-3.5	K
													4.3	2.5	2.8	1.6	1.6	-0.2	-1.0	M
														4.0	2.8	3.1	1.0	-0.7	-1.8	I
															4.0	1.8	2.0	0.0	-0.7	L
																3.4	0.1	-1.1	-2.6	V
																	7.0	5.1	3.6	F
																		7.8	4.1	Y
																			14.2	W

- Residue Identity
- Hydrophobic Similarity
- Hydrophilic Similarity

BLOSUM matrix

- ♦ BLOcks SUBstitution Matrix
- ♦ Based on gapless alignments
- ♦ More often used than PAM



BLOSUM N matrix

- Cluster together sequences N% identity

- $f(a,b)$ frequency of occurrence a and b in the same column

$$f(a) = f(a,a) + \sum_{a \neq b} \frac{f(a,b)}{2}$$

- $e(a,b)$ – chance alone $\sum_{a \leq b} e(a,b) = 1$

$$e(a,a) = f(a)^2$$

$$e(a,b) = 2 f(a)f(b) \quad \text{for } a \neq b$$

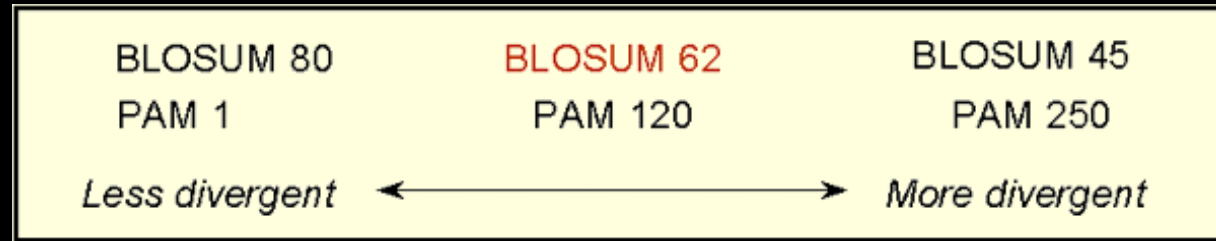
$$\text{BLOSUM}_N[a,b] = \underbrace{\log_2 \frac{f(a,b)}{e(a,b)}}_{\text{log odds}}$$

BLOSUM 62 matrix

```
# BLOSUM Clustered Scoring Matrix in 1/2 Bit Units
# Blocks Database = /data/blocks_5.0/blocks.dat
# Cluster Percentage: >= 62
# Entropy = 0.6979, Expected = -0.5209
```

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

PAM vs BLOSUM



<http://www.ncbi.nih.gov/Education/BLASTinfo/Scoring2.html>

- ♦ PAM is extrapolation from closely related seqs
- ♦ We are interested more distant relationships