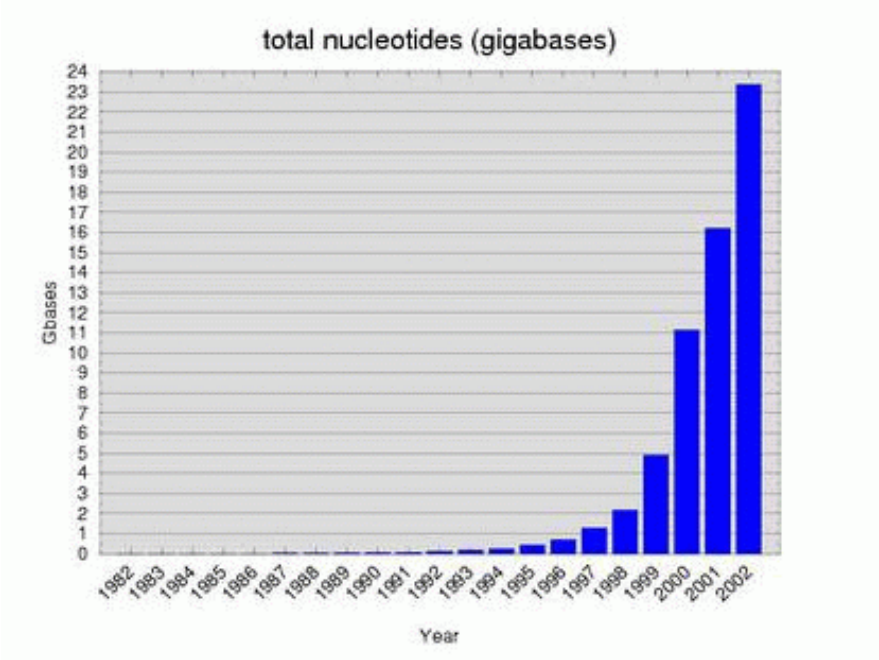
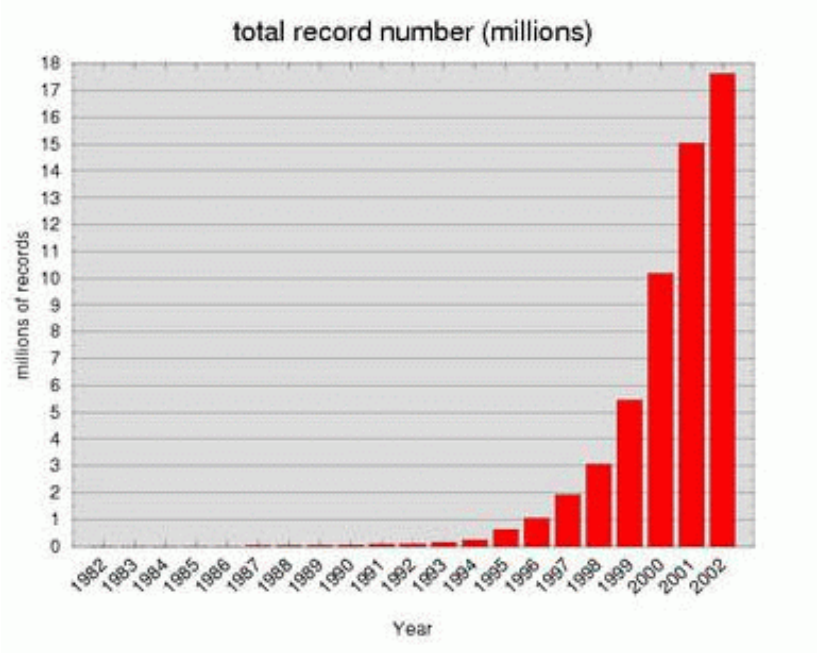
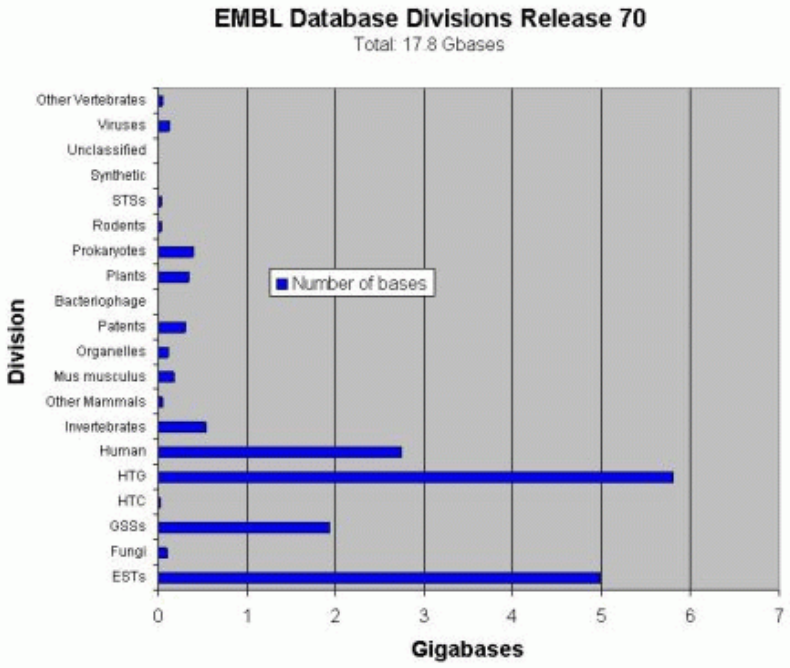
EMBL Database Growth

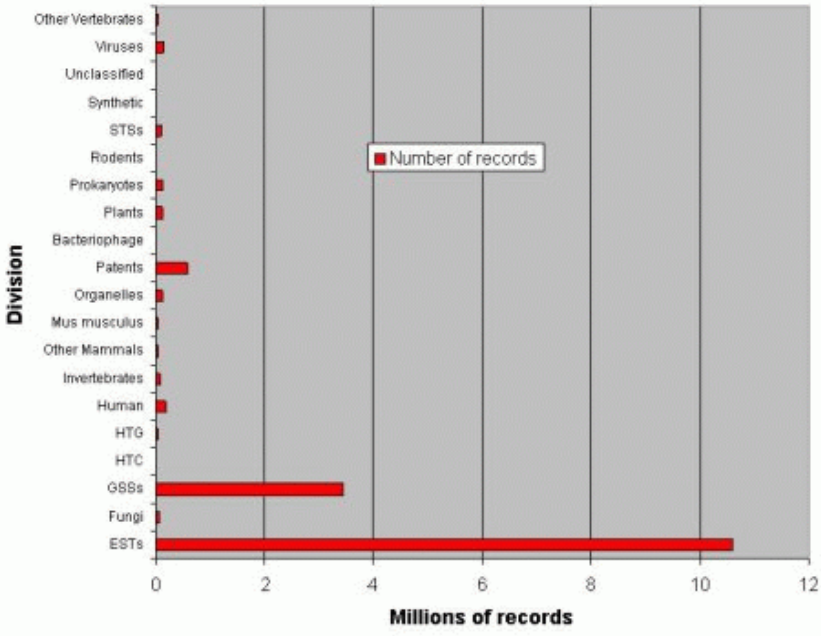
(growth statistics as of 05/30/02)





EMBL Divisions (05/30/02)





Computational Genomics

Sequence determination

Signal processing

Assembly of shotgun sequence fragments and determination of a consensus sequence

Sequence analysis

Search for genes, regulatory patterns, repeats, etc, in the inferred genomic sequence

Conceptual translation of the predicted genes into protein sequences

Inferences about the structural, functional, evolutionary, etc, properties of the putative proteins

Looking for similarities in biological Databases

Biological Databases :

Primary Data: **sequences : nucleic acids or proteins**
curated or not
complete or not
redundant or not
general or specialized (organisms...)

structures

Derived data: **patterns, motifs, profiles etc**

Query :

Sequence : nucleic acid or protein, "finished" or not, etc

Structural data

Motif, profile...

Similarity :

Global vs local, and other variations

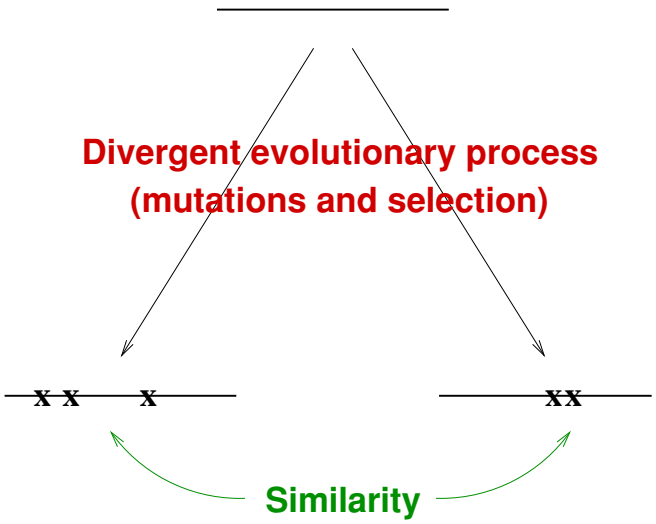
How to measure it?

What is its biological interpretation? significance?

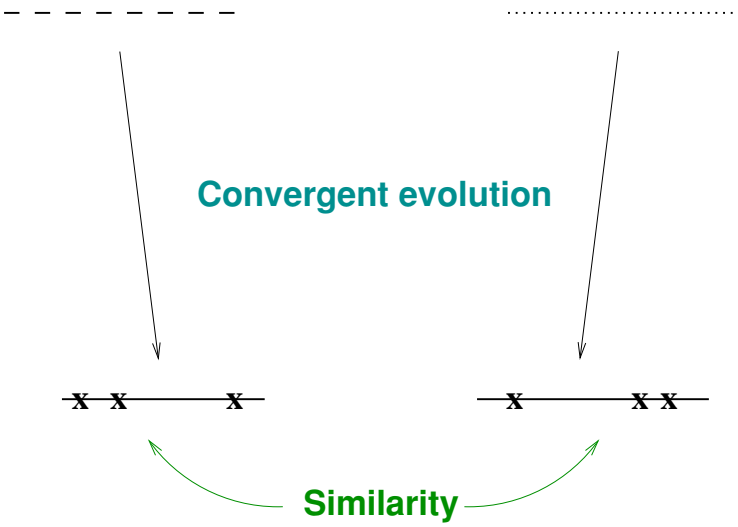
→ **Need to be better defined and formalized.**

Interpretation of biological similarity

HOMOLOGY



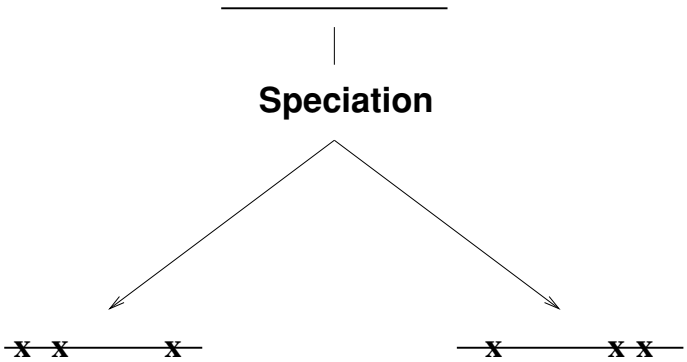
ANALOGY



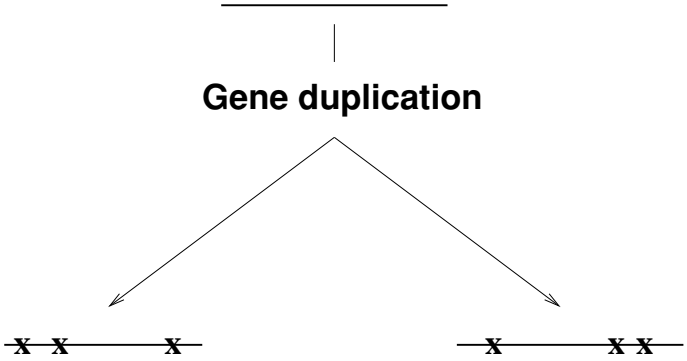
From W. M. Fitch, 2000: "Homology; a personal view on some of the problems", Trends in genetics, 16, p227–231

HOMOLOGIES

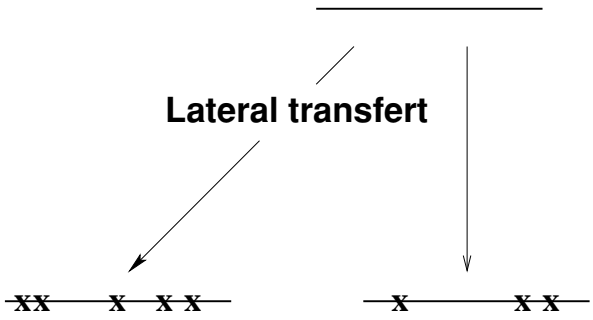
ORTHOLOGY



PARALOGY



XENOLOGY



Introduction to Computational sequence analysis

1. Pairwise sequence comparison

- 1.1 Algorithms for sequence alignments
- 1.2 Scoring schemes

2. Similarity searching in sequence databases

- 2.1 Sequence databases
- 2.2 The Fasta and Blast programs
- 2.3 Scores and statistics

3. Multiple alignments and motifs

- 3.1 Algorithms: how to compute global or local multiple alignments?
- 3.2 Representation of the information contained in a multiple alignment
- 3.3 Examples of resources and applications of these concepts in the context of similarity searches
- 3.4 Back to the algorithms: motif inference

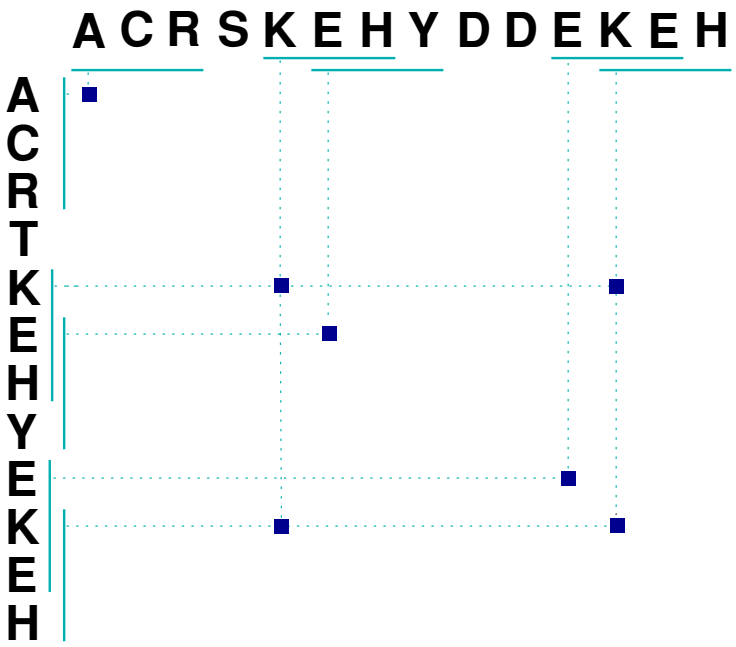
Other important aspects of computational sequence analysis:

- Gene prediction
- Protein structure prediction and homology modeling
- Phylogenetic inference

Part 1:
Pairwise sequence comparison

DOT–PLOT

Word matches



Approximate word matches

Parameters:

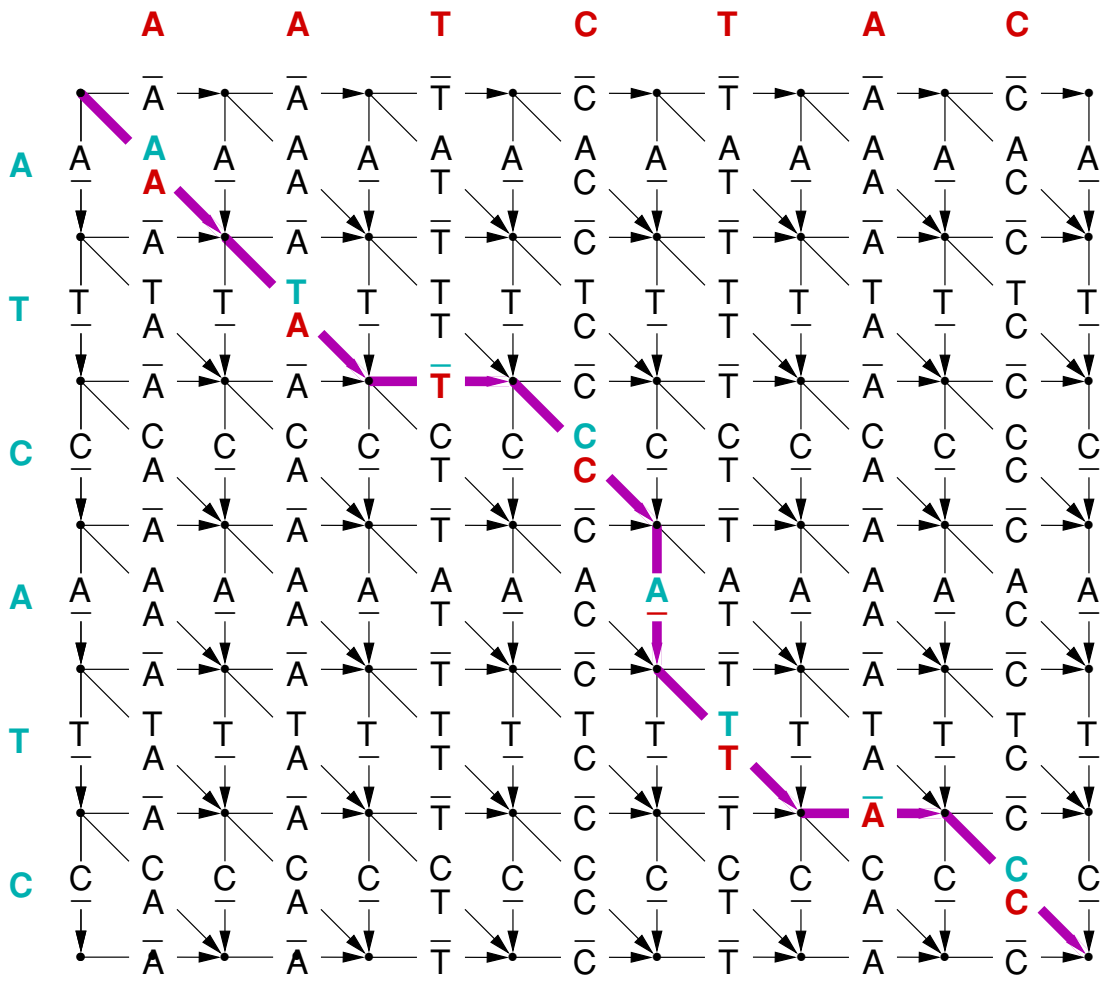
w = size of the sliding words or windows

t = threshold which is applied to the score of the comparison between two windows (or words). The score itself may be computed as the number of identities between the two words, or using a scoring matrice for amino acids.

Global alignment

```
seq A  MAKSKSSQGASGARRKPAPSLYQHISSFKPQFSTRVDDVLHFSKTLTWRSEIIPDKSKGT
       | |.        . .. .|| |.||||..|.| .|||.|||  |.||. . ..|||...|
seq B  MPKK--------VWKSSTPSTYEHISSLRPKFVSRVDNVLHQRKSLTFSNVVVPDKKNNT

seq A  LTTSLLYSQGSDIYEIDTTLPLKTFYDDDDDDDNDDDDEEGNGKTKSAATPNPEYGDAFQ
       ||.|..||||||||||| ..||.                       ..|. |  .|||||.
seq B  LTSSVIYSQGSDIYEIDFAVPLQ----------------------EAASEPVKDYGDAFE

seq A  DVEGKPLRPKWIYQGETVAKMQYLESSDDSTAIAMSKNGSLAWFRDEIKVPVHIVQEMMG
        .|.  | ||..|||||||.|| ||... ..| ..||||||||||.. ||||.||||||.||
seq B  GIENTSLSPKFVYQGETVSKMAYLDKTGETTLLSMSKNGSLAWFKEGIKVPIHIVQELMG

seq A  PATRYSSIHSLTRPG-------SLAVSDFDVSTNMDTVVKSQSNGYEEDSILKIIDNSDR
       ||| |.|||||||||       |||.||| .|.. .|.|||||| ||||||||||. .
seq B  PATSYASIHSLTRPGDLPEKDFSLAISDFGISNDTETIVKSQSNGDEEDSILKIIDNAGK

seq A  PGDILRTVHVPGTNVAHSVRFFNNHLFASCSDDNILRFWDTRTADKPLWTLSEPKNGRLT
       ||.||||||||||.|.|.||||.||.||||||||||||||||.|||.|.|.||||||.||
seq B  PGEILRTVHVPGTTVTHTVRFFDNHIFASCSDDNILRFWDTRTSDKPIWVLGEPKNGKLT

seq A  SFDSSQVTENLFVTGFSTGVIKLWDARAVQLATTDLTHRQNGEEPIQNEIAKLFHSGGDS
       ||| |||..||||||||||.||||||.. ||||||.||||||.|||||||...|.||||
seq B  SFDCSQVSNNLFVTGFSTGIIKLWDARAAEAATTDLTYRQNGEDPIQNEIANFYHAGGDS

seq A  VVDILFSQTSATEFVTVGGTGNVYHWDMEYSFSRNDDDNEDEVRVAAPEELQGQCLKFFH
       |||. || ||..|| |||||||.|||. .||.|. . |.       | || | . |.|.|
seq B  VVDVQFSATSSSEFFTVGGTGNIYHWNTDYSLSKYNPDDTIAPPQDATEESQTKSLRFLH

seq A  TGGTRRSSNQFGKRNTVALHPVINDFVGTVDSDSLVTAYKPFLASDFIGRGYDD
        ||.||| .|.|.|||.| ||||...|||||.||||. |||.   .       .
seq B  KGGSRRSPKQIGRRNTAAWHPVIENLVGTVDDDSLVSIYKPYTEES-------E
```

Alignment as a path in a graph



Alignment corresponding to the colored path:

A T – C A T – C
A A T C – T A C

Scoring Schemes

Similarity measures:

. the scores may be either positive or negative

. the greater the similarity between two compared symbols, the greater the score of their comparison,

. when using a similarity scoring scheme, we want to maximize the score of the alignment.

Distance measures:

. scores are always positive and increase when the similarity decreases

. a simple distance scoring scheme:

if x = y, then d(x,y) = 0, else d(x,y) = 1
d(x,–) = d(–,y) = 1

. using a distance measure, we want to minimize the score of the alignment

Alignments

An alignment is defined as a series of paired symbols, that are either letters from the alphabet of the sequences, or the symbol for a gap.

Given two sequences

$$A = a_1 a_2 \cdots a_m \quad \textbf{and} \quad B = b_1 b_2 \cdots b_n$$

one alignment between the two sequences is represented as:

$$\mathcal{A} = \begin{pmatrix} a_1 a_2 \cdots a_m \\ b_1 b_2 \cdots b_n \end{pmatrix} = \begin{pmatrix} \bar{a_1} \\ \bar{b_1} \end{pmatrix} \begin{pmatrix} \bar{a_2} \\ \bar{b_2} \end{pmatrix} \cdots \begin{pmatrix} \bar{a_i} \\ \bar{b_i} \end{pmatrix} \cdots \begin{pmatrix} \bar{a_p} \\ \bar{b_p} \end{pmatrix}$$

The score of an alignment is defined as the sum of the scores of all the paired symbols:

$$score\mathcal{A} = \sum_{i=1}^{p} score \begin{pmatrix} \bar{a_i} \\ \bar{b_i} \end{pmatrix}$$

which can be written as:

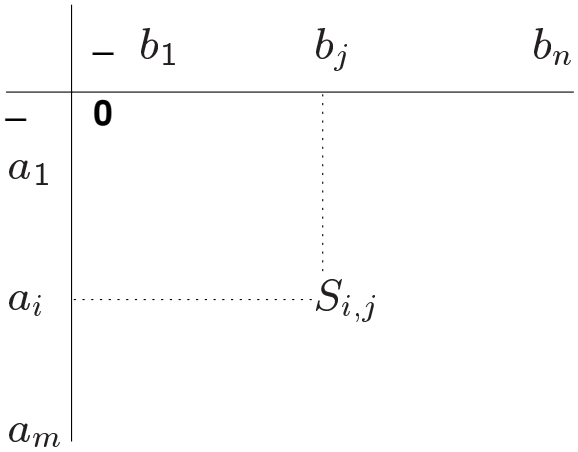
$$score\mathcal{A} = score\mathcal{A}' + score \begin{pmatrix} \bar{a_p} \\ \bar{b_p} \end{pmatrix}$$

with: $$\mathcal{A}' = \begin{pmatrix} \bar{a_1} \bar{a_2} \cdots a_{\bar{p-1}} \\ \bar{b_1} \bar{b_2} \cdots b_{\bar{p-1}} \end{pmatrix} = \begin{pmatrix} \bar{a_1} \\ \bar{b_1} \end{pmatrix} \begin{pmatrix} \bar{a_2} \\ \bar{b_2} \end{pmatrix} \cdots \begin{pmatrix} a_{\bar{p-1}} \\ b_{\bar{p-1}} \end{pmatrix}$$

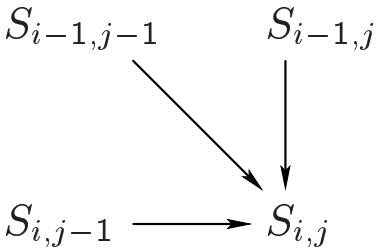
The optimal alignment is defined as the one having the best score ("best" meaning "lowest" or "highest", depending the kind of scoring scheme that is used)

Dynamic Programming

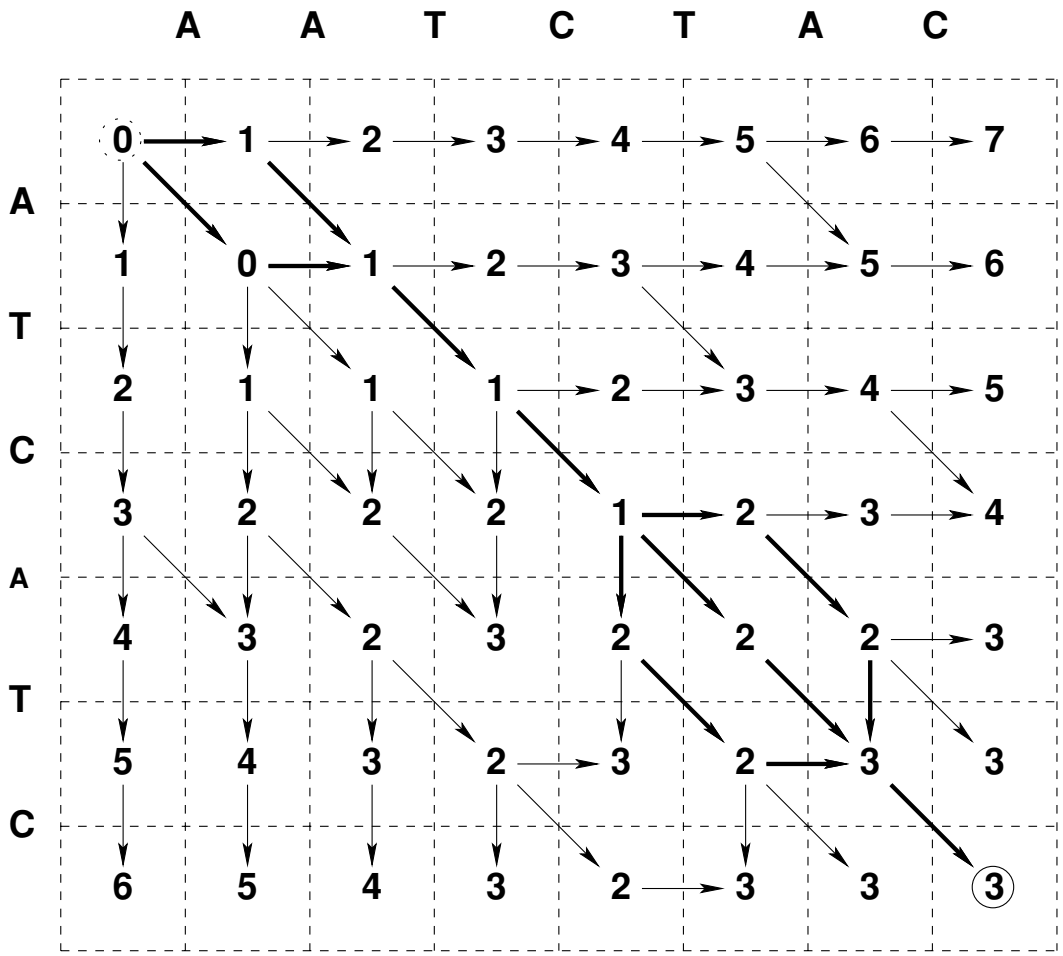
$$S_{i,j} = score\ \ opt\_\begin{pmatrix} a_1 \cdots a_i \\ b_1 \cdots b_j \end{pmatrix}$$



$$S_{i,j} = min \begin{cases} score\ \ opt\_\begin{pmatrix} a_1 \cdots a_{i-1} \\ b_1 \cdots b_{j-1} \end{pmatrix} + score\begin{pmatrix} a_i \\ b_i \end{pmatrix} \\ score\ \ opt\_\begin{pmatrix} a_1 \cdots a_i \\ b_1 \cdots b_{j-1} \end{pmatrix} + score\begin{pmatrix} - \\ b_j \end{pmatrix} \\ score\ \ opt\_\begin{pmatrix} a_1 \cdots a_{i-1} \\ b_1 \cdots b_j \end{pmatrix} + score\begin{pmatrix} a_i \\ - \end{pmatrix} \end{cases}$$



Dynamic Programming – Global alignment

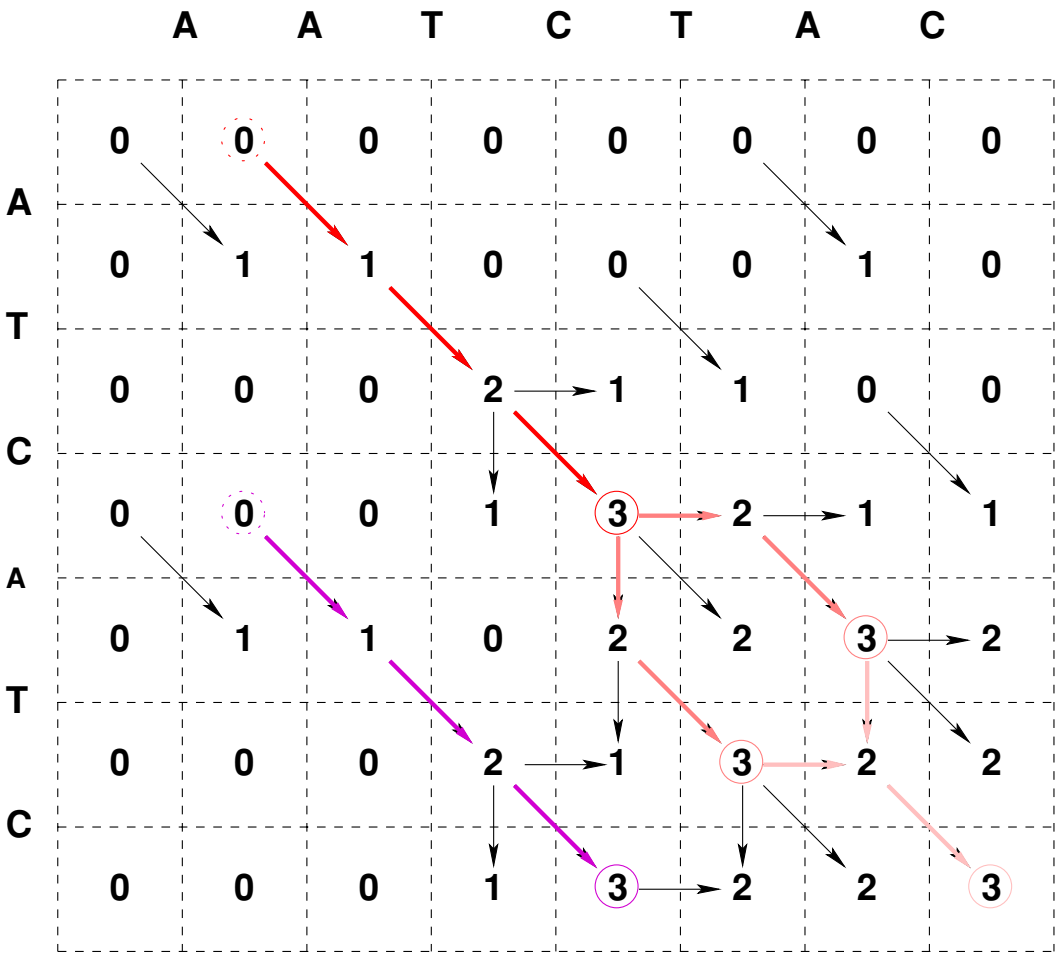


Scoring parameters (distance scheme):

Matches = 0

Mismatches and insertions/deletions = 1

Dynamic Programming – Local Alignment



Scoring parameters (similarity scheme):

Matches: –1

Mismatches and insertions/deletions = –1

Dynamic Programming alignment of two sequences

Global alignment :

S. Needleman and C. Wunsch, 1970.

P. Sellers, 1974.

Local alignment :

T. Smith and M. Waterman , 1981.

Improvments to the algorithms:

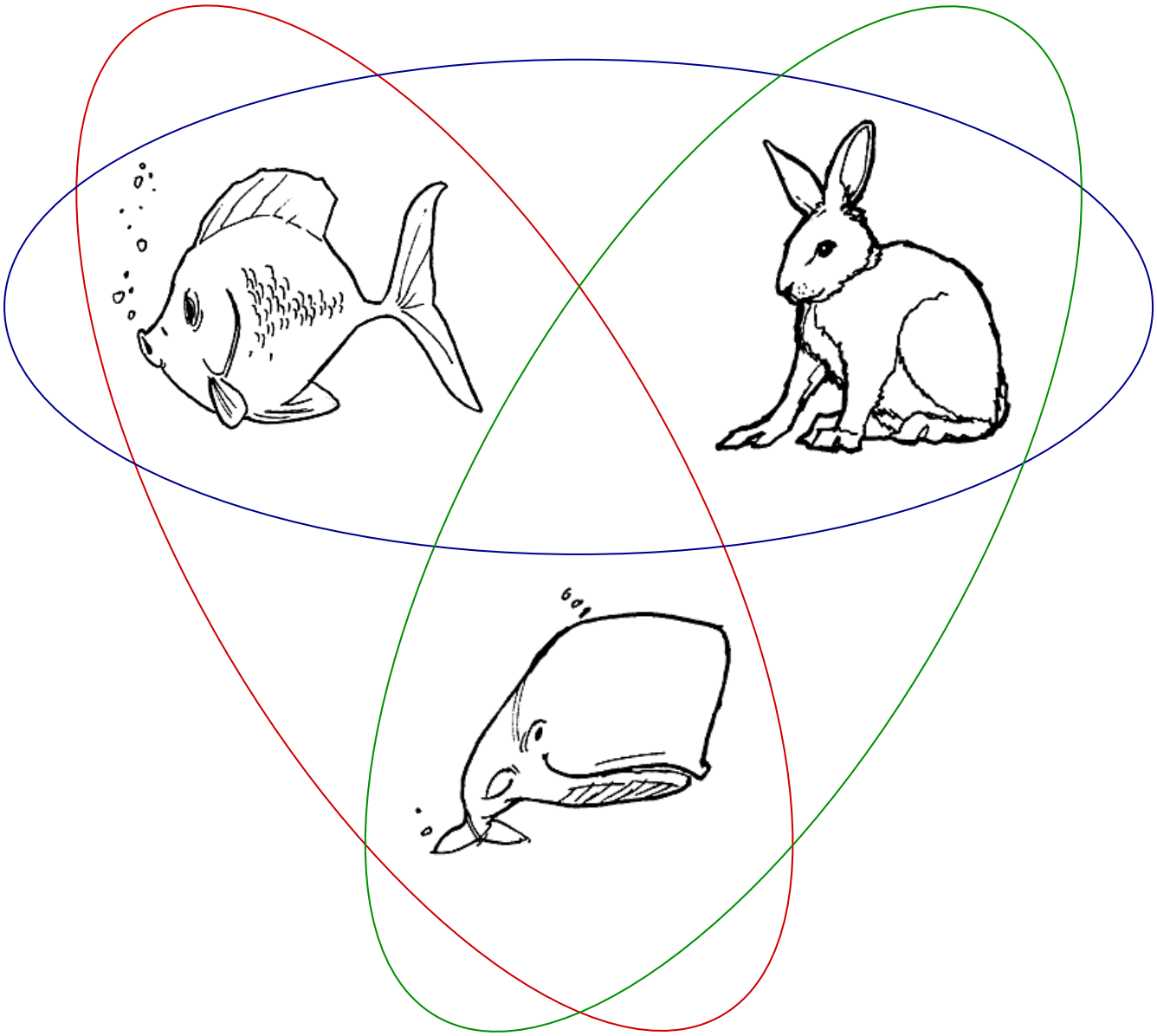
O. Gotoh, 1982

E. W. Myers and W. Miller, 1988

The PAM250 similarity matrice

(M. O. Dayhoff et al, 1978)

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

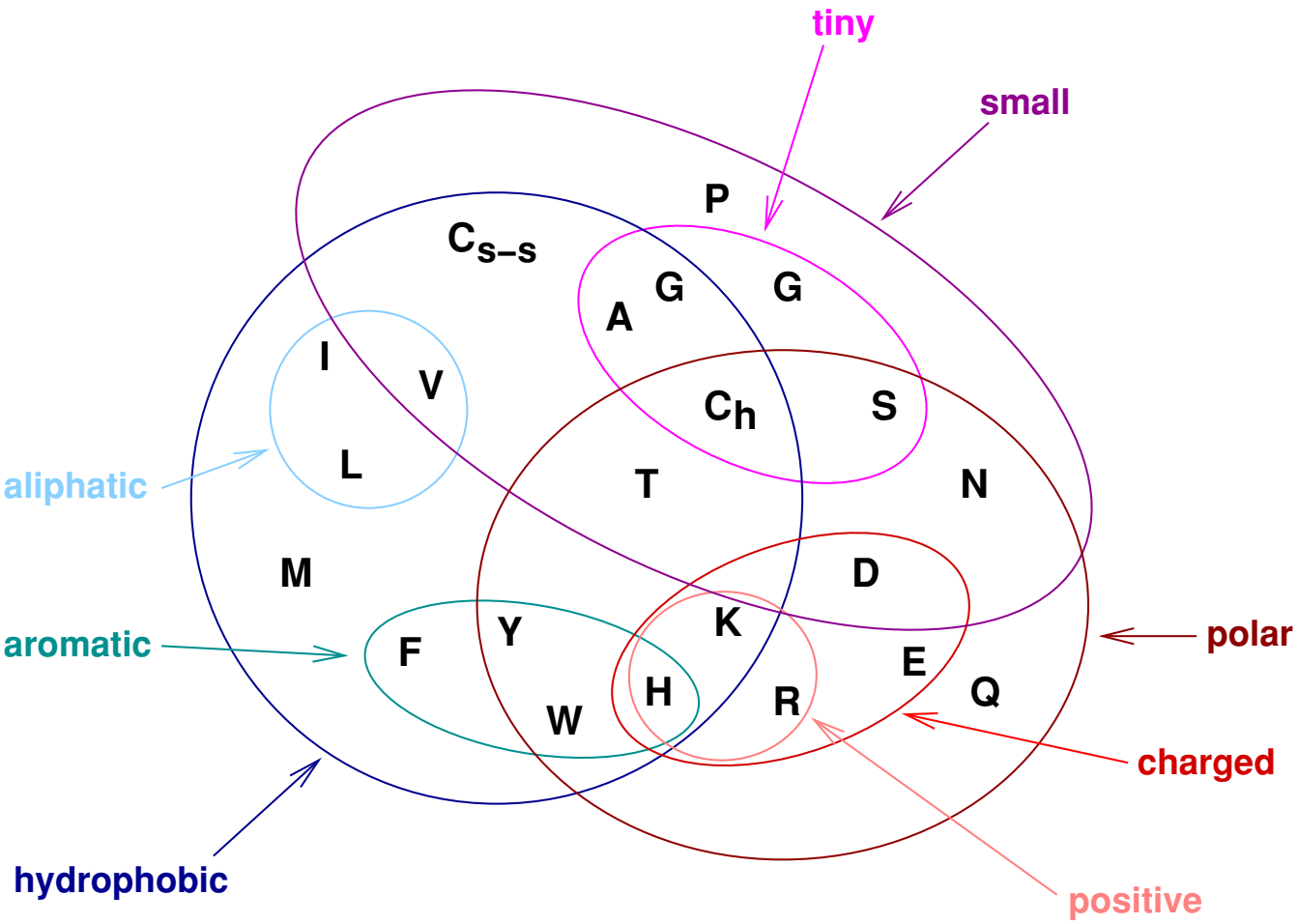


Different viewpoints

(From M–F. Sagot, PhD dissertation, 1996)

Venn Diagram for amino acids

Proposed by W. R. Taylor, 1986



PAM Matrices

(M. Dayhoff et al., 1968, 1972, 1978)

PAM = Accepted Point Mutation

1. Raw PAM Matrix :

Observed frequencies of occurence of the substitutions.

2. Mutation Data Matrix (MDM) :

From the observed frequencies of change occurence and the individual frequencies of the amino acids, for each pair of amino acids i and j, compute the probability of i mutating into j during a given amount of evolution.

Amounts of evolution are expressed in PAM units : one PAM is the amount of evolution during which one expects on average 1% of change.

Derive the probabilities for one PAM and extrapolate to other PAM distances.

3. Relatedness Odds Matrix :

For each pair i, j of amino acids, $R_{i,j}$ represents the ratio of the probability of i and j being aligned because corresponding positions in the sequence are homologous, by the probability of i and j being aligned by chance.

$$R_{i,j} = \frac{q_{i,j}}{p_i \, p_j}$$

4. Scoring Matrix :

$$S_{i,j} = \log R_{i,j}$$

Log–Odds Matrices

Old PAM series = Dayhoff et al.

New PAM series = Jones et al, Gonnet et al.

PAMx : x stands for the evolutionary distance represented by the matrix.

BLOSUM series = Henikoff & Henikoff

BLOSUMy : y is the minimum percent of identity in the set of sequences from which the matrix is derived.

$$S_{i,j} = \log \frac{q_{i,j}}{p_i p_j}$$

$q_{i,j}$: target frequencies

p_i, p_j : background frequencies

Any matrix used for scoring local alignments is implicitly a log–odds matrix, best suited for distinguishing local alignments in which i and j are aligned with frequency $q_{i,j}$. (Stephen Altschul)

Scoring gaps

Simplest model:

The same penalty for each inserted or deleted element.
$\longrightarrow$ the score of a gap will be proportional to its length.

Affine gap costs:

The score for a gap of length l is of the form a + b l :
"a" is the cost for the presence of a gap per se
"b" is the cost (per residue) for extending the gap

Concave functions:

Empirical studies :

Benner, Cohen and Gonnet, 1993 (and previous studies):
From the observed distribution of gap lengths, they infer that the score of a gap should be of the form: a + b log l

Theoretical work :

Waterman, 1984; Miller and Myers, 1988
Development of algorithms for sequence alignment with concave gap costs.

Different treatments for different kind of gaps:

Distinguishing gaps inside alignments from gaps between aligned regions: implicitly done by combining several local alignments in WU–Blast2

Considering regions with gaps as "unaligned" regions: generalized affine gap costs (Altschul, 1998).