

---

# Sequence Analysis

---

- ❖ ECS129
- ❖ Patrice Koehl

## Sequence Analysis: Outline

1. Why do we compare sequences?
2. Sequence comparison: from qualitative to quantitative methods
3. Deterministic methods: Dynamic programming
4. Heuristic methods: BLAST
5. Multiple Sequence Alignment

# Similarity: Homology vs Analogy

**Homology:** Similarity in characteristics resulting from shared ancestry.

**Analogy:** The similarity of characteristics between two species that are not closely related; attributable to convergent evolution.



***Two sisters: homologs***



***Two “Elvis”: analogs***

# Homology: Orthologs and Paralogs

**Homology:** Similarity in characteristics resulting from shared ancestry.

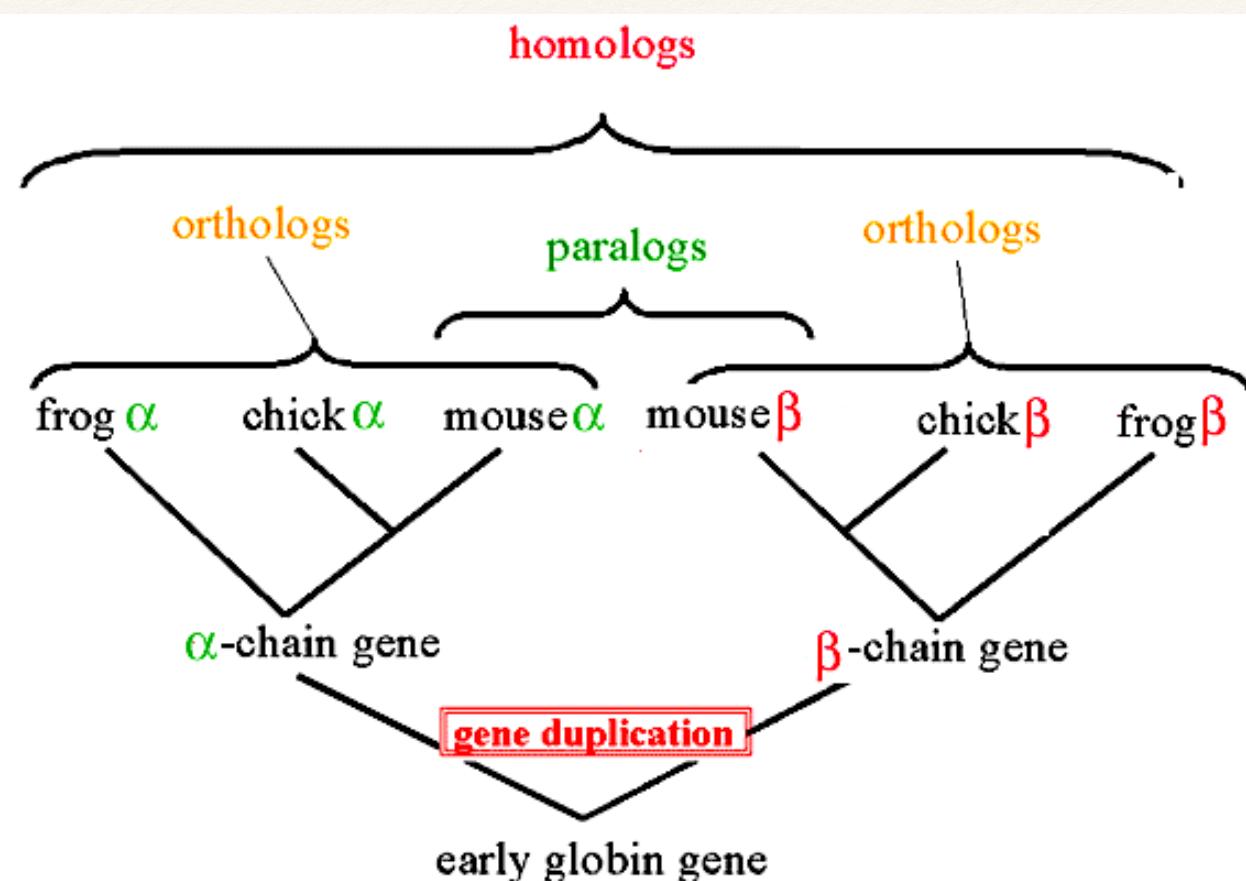
**Paralogy:** Homologous sequences are paralogous if they were separated by a gene duplication event

**Orthology:** Homologous sequences are orthologous if they were separated by a speciation event

**Further reading:**

Koonin EV (2005). “Orthologs, paralogs, and evolutionary genomics”. *Annu. Rev. Genet.* 39:309-338.

# Homology: Orthologs and Paralogs

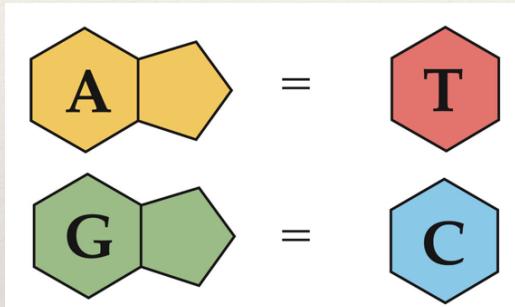


# Applications of Sequence Analysis

- Sequencing projects, assembly of sequence data
- Evolutionary history
- Identification of functional elements in sequences
- gene prediction
- Classification of proteins
- Comparative genomics
- RNA structure prediction
- Protein structure prediction
- Health Informatics

## DNA sequence: Chargaff's rules

**Rule 1:** In double stranded DNA, the amount of guanine is equal to cytosine and the amount of adenine is equal to thymine



(basis of Watson Crick base pairing)

**Rule 2:** the composition of DNA varies from one species to another; in particular in the relative amounts of A, G, T, and C bases

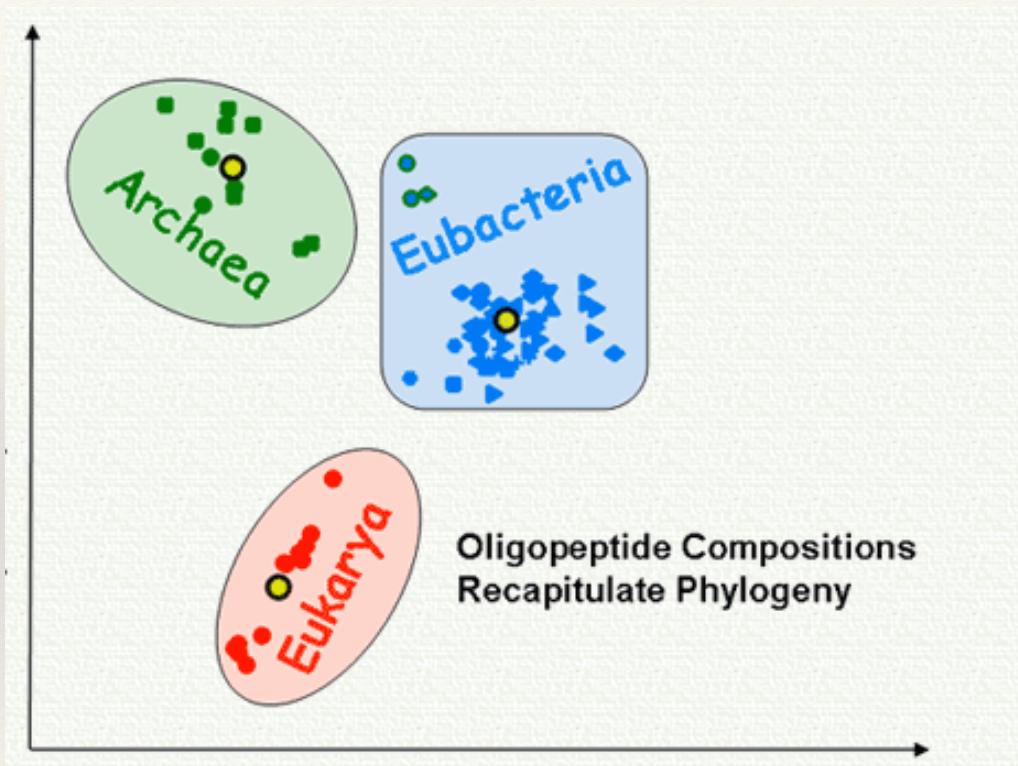
## DNA sequence: Chargaff's rules

Table 3-2 Data Leading to the Formulation of Chargaff's Rules

Source	Adenine to Guanine	Thymine to Cytosine	Adenine to Thymine	Guanine to Cytosine	Purines to Pyrimidines
Ox	1.29	1.43	1.04	1.00	1.1
Human	1.56	1.75	1.00	1.00	1.0
Hen	1.45	1.29	1.06	0.91	0.99
Salmon	1.43	1.43	1.02	1.02	1.02
Wheat	1.22	1.18	1.00	0.97	0.99
Yeast	1.67	1.92	1.03	1.20	1.0
<i>Hemophilus influenzae</i>	1.74	1.54	1.07	0.91	1.0
<i>E-coli</i> K2	1.05	0.95	1.09	0.99	1.0
Avian tubercle bacillus	0.4	0.4	1.09	1.08	1.1
<i>Serratia marcescens</i>	0.7	0.7	0.95	0.86	0.9
<i>Bacillus schatz</i>	0.7	0.6	1.12	0.89	1.0

SOURCE: After E. Chargaff et al., *J. Biol. Chem.* 177 (1949).

## Comparing sequences based on their tri-peptide content



*Proteins: Structure, Function and Genetics* 54, 20-40 (2004)

## Comparing individual letters

*Scores are usually stored in a “weight” matrix also called “substitution” matrix or “matching” matrix.*

Defining the “proper” matrix is still an active area of research:

**1. Identity matrix**

**2. Chemical property matrix**

In this matrix amino acids or nucleotides are intuitively classified on the basis of their chemical properties

**3. Substitution-based matrix**

Dayhoff matrix

PAM matrices

Blosum matrices

# Substitution Matrices

**Dayhoff matrix** was created in 1978 based on few closely related (> 85% identity) sequences available this time (1500 aligned amino-acids).

**PAM-family of matrices** is a simple update of the original Dayhoff matrix.

**Gonnet matrices** were created by exhaustive alignment of all Database sequences in 1992.

**BLOSUM matrix** is based on local similarities (blocks) of proteins rather than overall alignments.

## Most common Scoring Matrices

### BLOSUM matrices (*Henikoff and Henikoff, 1992*)

- Start from “reliable” alignments of sequences with at least **XX** % identity
- Compute mutation probabilities
- Convert into Scores: -> BLOSUM**XX** matrix

### PAM matrices (*Dayhoff, 1974*)

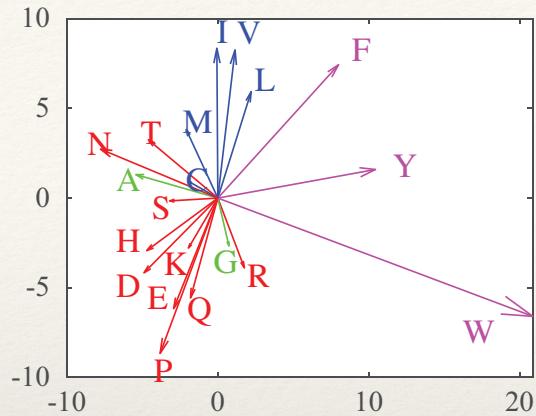
- Point Accepted Mutation
- Start with PAM score = 1: alignments of sequences with 1 mutation -> PAM1 matrix
- Generate successive PAM matrices:

$$\text{PAM}\text{XX} = (\text{PAM1})^{\text{XX}}$$

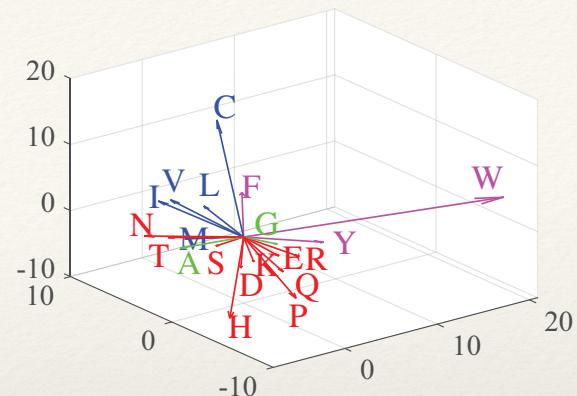
## Example of a Scoring matrix: Blosum62

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

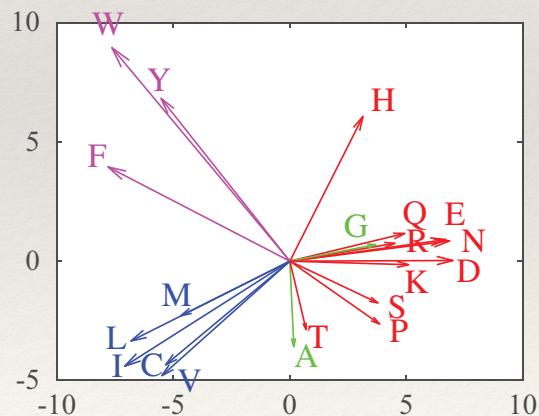
BLOSUM30: 2D Projection



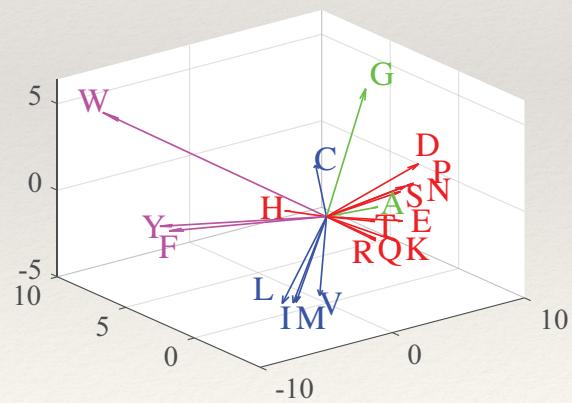
BLOSUM30: 3D Projection



BLOSUM62: 2D Projection



BLOSUM62: 3D Projection



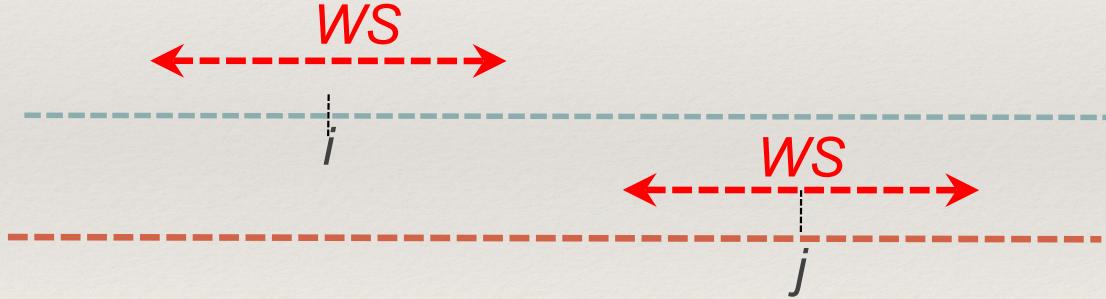
# DotPlot: Overview of Sequence Similarity

**Build a table  $S$ :**

- rows: Sequence 1
- columns: Sequence 2

**Assign a score  $S(i,j)$  to each entry in the table:**

- select a window size WS

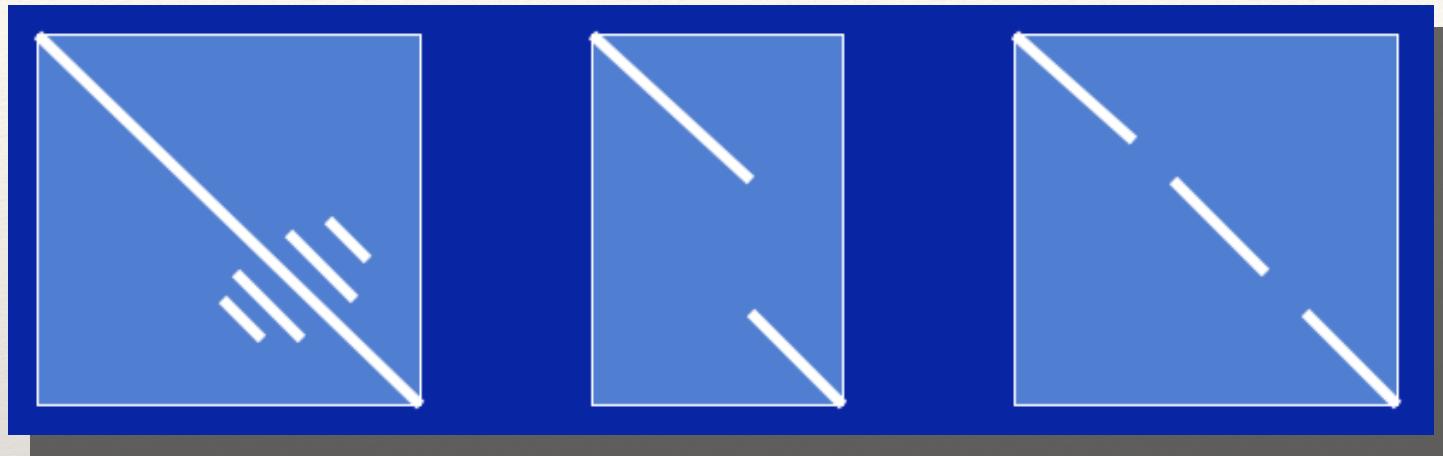


- Compare window around  $i$  with window around  $j$   $\rightarrow$  Score( $i,j$ )

**Display table of scores  $S$**

- show a dot at position  $(i,j)$  if Score( $i,j$ )  $>$  Threshold

## Patterns on DotPlot



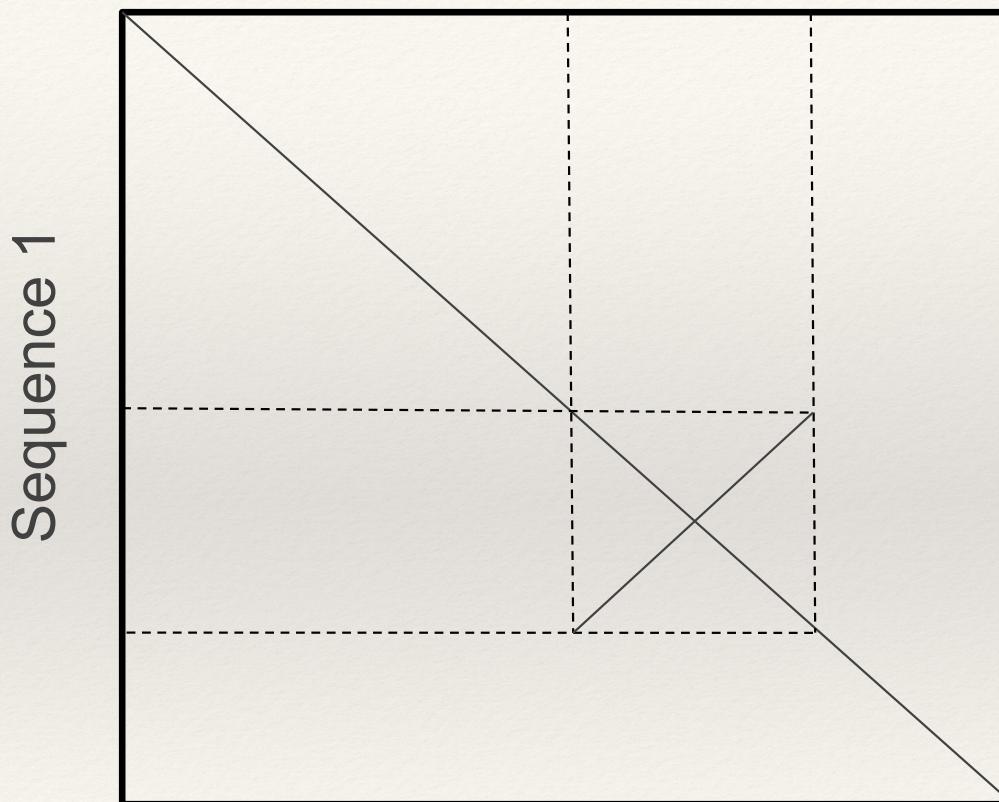
Internal Repeat

Insertion (Deletion)

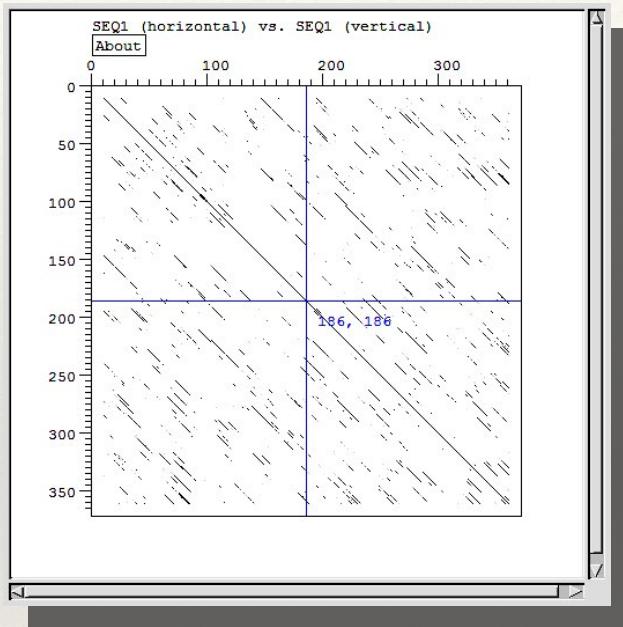
Divergence

# Patterns on DotPlot

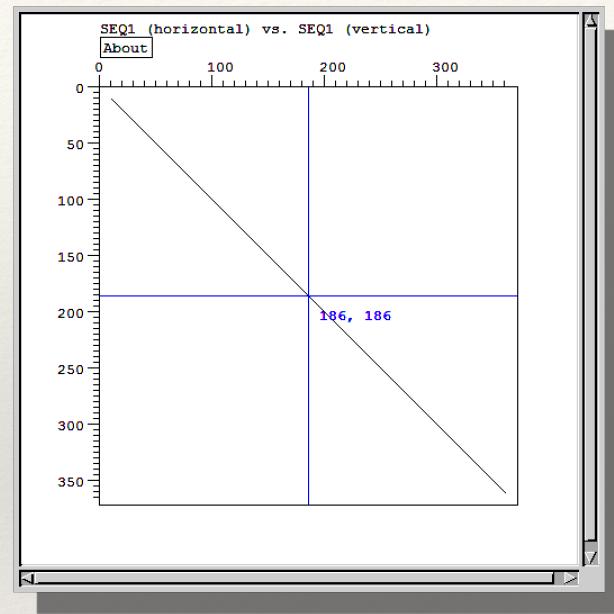
Sequence 2



# Patterns on DotPlot



With many details



Overall view - no details

## What is sequence alignment?

Given two sequences of letters and a **scoring scheme** for evaluating letter matching, find the optimal pairing of letters from one sequence to the other.

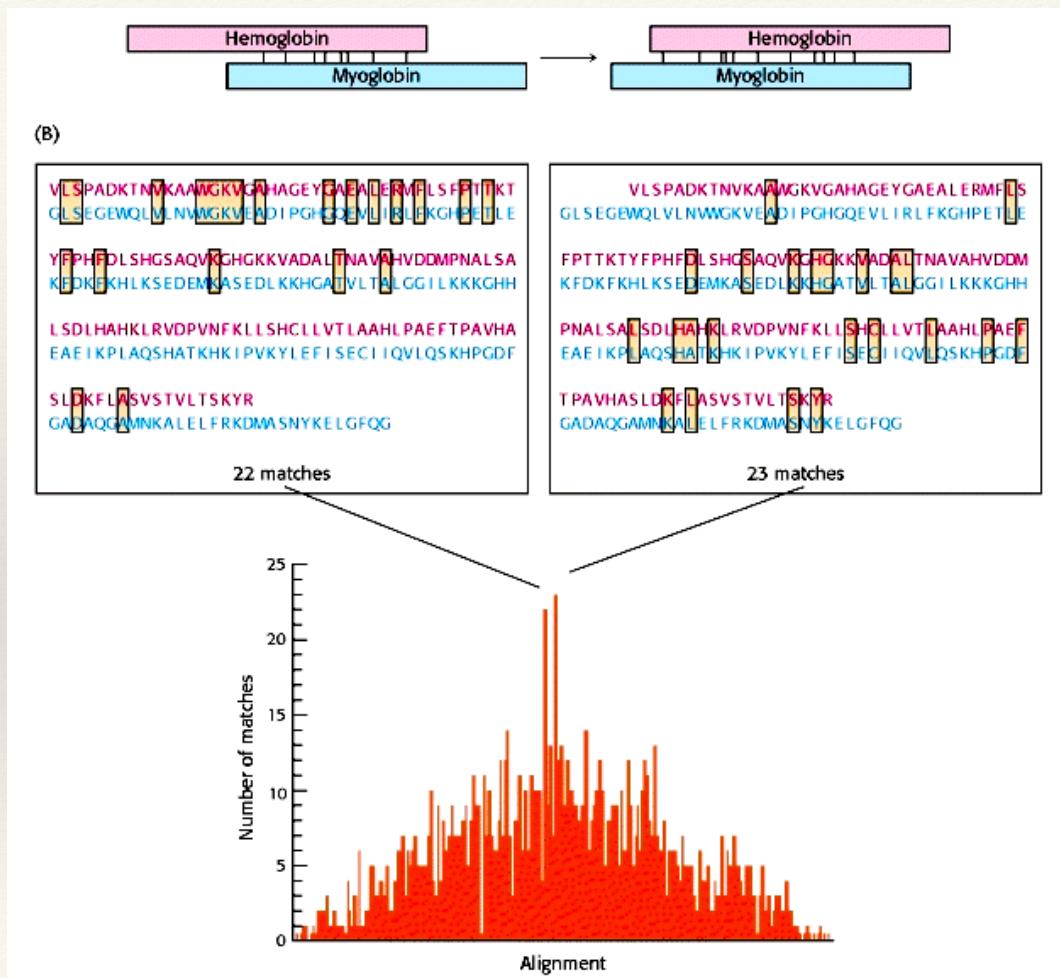
### Human hemoglobin ( $\alpha$ chain)

VLSPADKTNVKAAWGKVGAAHAGEYGAEALERMFLSFPTTKTYFPHFDLSHG  
SAQVKGHGKKVADALTNAVAHVDDMPNALSDLHAHKLRVDPVNFKLLS  
HCLLVTLAAHLPAEFTPASLDKFLASVSTVLTSKYR

### Human myoglobin

GLSDGEWQLVLNWNGKVEADIPGHGQEVLIRLFKGHPETLEKFDKFKHLKS  
EDEMKAESDLKKHGATVLTALGGILKKKGHEAEIKPLAQSHATKHKIPVK  
YLEFISECIIQVLQSKHPGDFGADAQGAMNKALELFRKDMASNYKELGFQG

# Ungapped Alignment



*(From Biochemistry,  
Stryer, fifth edition)*

## Alignment with gap(s)



How do we generate the “best” gapped alignment ?

Total number of possible gapped alignment:

$$\sum_{k=1}^{\min(N,M)} \binom{N}{k} \binom{M}{k}$$

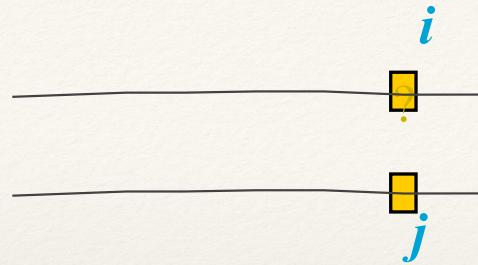
# DP and Sequence Alignment

## *Key idea:*

The score of the optimal alignment that ends at a given pair of positions in the sequences is the score of the best alignment previous to these positions plus the score of aligning these two positions.

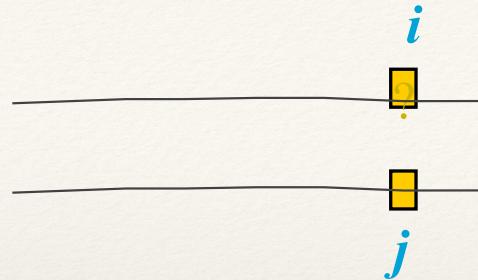
# DP and Sequence Alignment

*Test all alignments that can lead to  $i$  aligned with  $j$*



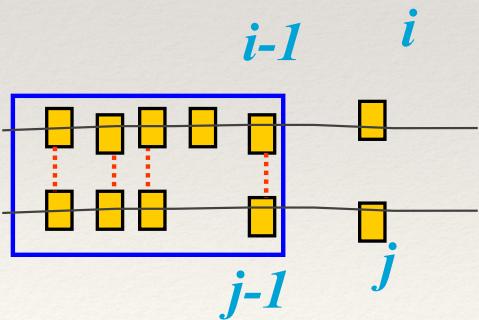
# DP and Sequence Alignment

*Test all alignments that can lead to  $i$  aligned with  $j$*



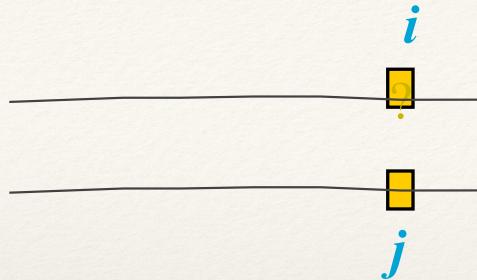
3 possibilities:

1)  $i-1$  aligned with  $j-1$



# DP and Sequence Alignment

*Test all alignments that can lead to  $i$  aligned with  $j$*

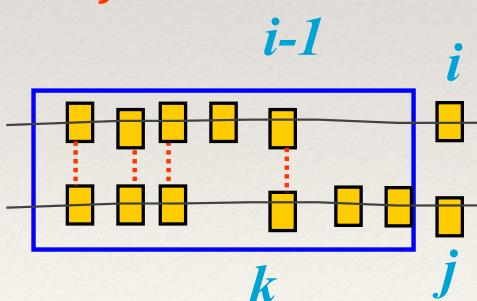
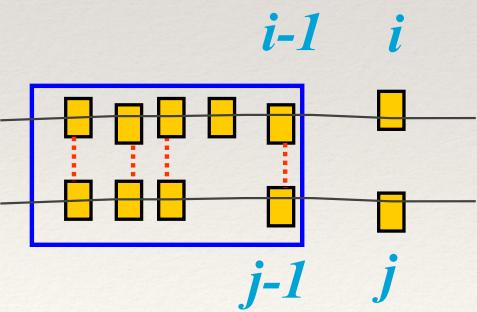


3 possibilities:

1)  $i-1$  aligned with  $j-1$

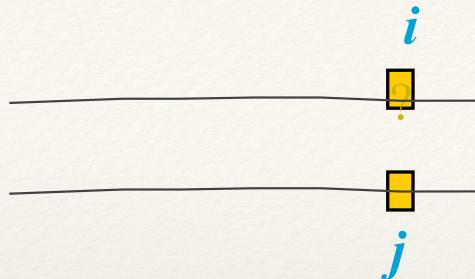
2)  $i-1$  aligned with  $k$ ,

$1 \leq k \leq j-2$



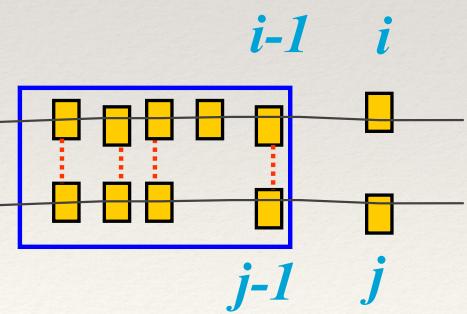
# DP and Sequence Alignment

*Test all alignments that can lead to  $i$  aligned with  $j$*



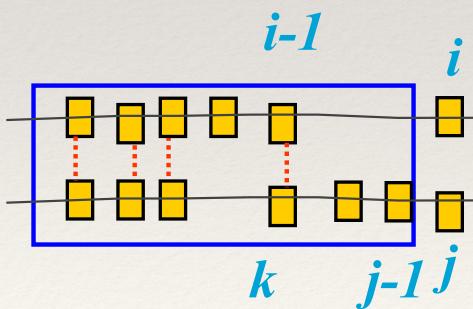
*3 possibilities:*

1)  $i-1$  aligned with  $j-1$



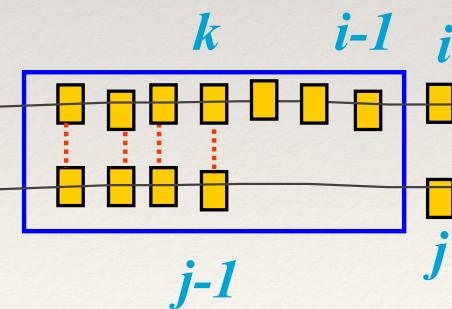
2)  $i-1$  aligned with  $k$ ,

$$1 \leq k \leq j-2$$



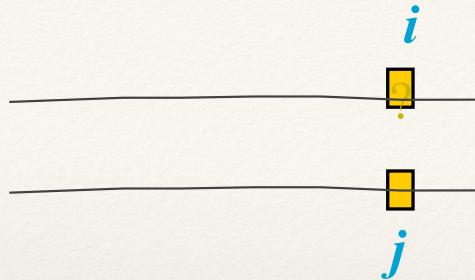
3)  $j-1$  aligned with  $l$ ,

$$1 \leq l \leq i-2$$



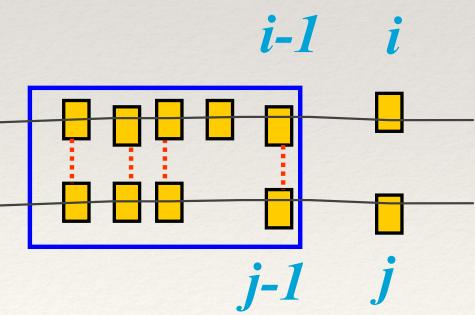
# DP and Sequence Alignment

*Test all alignments that can lead to  $i$  aligned with  $j$*



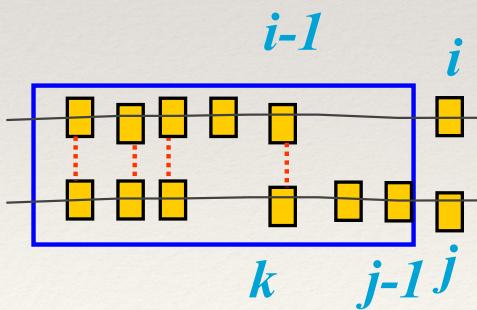
*3 possibilities:*

1)  $i-1$  aligned with  $j-1$



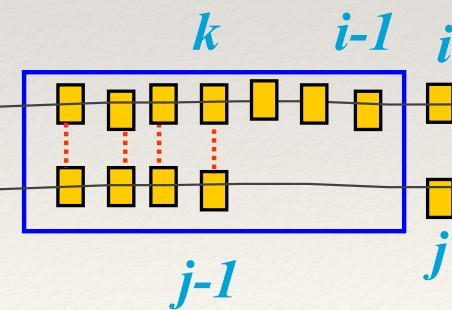
2)  $i-1$  aligned with  $k$ ,

$$1 \leq k \leq j-2$$



3)  $j-1$  aligned with  $l$ ,

$$1 \leq l \leq i-2$$



*Choose option that leads to the best score*

# Implementing the DP algorithm for sequences

*Aligning 2 sequence S1 and S2 of lengths N and M:*

- 1) Build a NxM alignment matrix A such that  
 $A(i,j)$  is the optimal score for alignments  
up to the pair  $(i,j)$
- 2) Find the best score in A
- 3) Track back through the matrix to get  
the optimal alignment of S1 and S2.

# Implementing the DP algorithm for sequences

*Aligning 2 sequence  $S1$  and  $S2$  of lengths  $N$  and  $M$ :*

# Example

Sequence 1: AWVCDEC

Sequence 2: AWEC

$\text{Score}(i,j) = 10 \text{ if } i=j, 0 \text{ otherwise}$

no gap penalty

## Example

### 1) Initialize

	A	W	V	C	D	E	C
A	10	0	0	0	0	0	0
W	0						
E	0						
C	0						

## Example

### 2) Propagate

	A	W	V	C	D	E	C
A	10	0	0	0	0	0	0
W	0	20					
E	0						
C	0						

## Example

### 2) Propagate

	A	W	V	C	D	E	C
A	10	0	0	0	0	0	0
W	0	20	10				
E	0						
C	0						

## Example

### 2) Propagate

	A	W	V	C	D	E	C
A	10	0	0	0	0	0	0
W	0	20	10	10	10	10	10
E	0	10	20	20	20	30	20
C	0	10	20	30	20	20	20

## Example

### 3) Trace back

	A	W	V	C	D	E	C
A	10	0	0	0	0	0	0
W	0	20	10	10	10	10	10
E	0	10	20	20	20	30	20
C	0	10	20	30	20	20	40

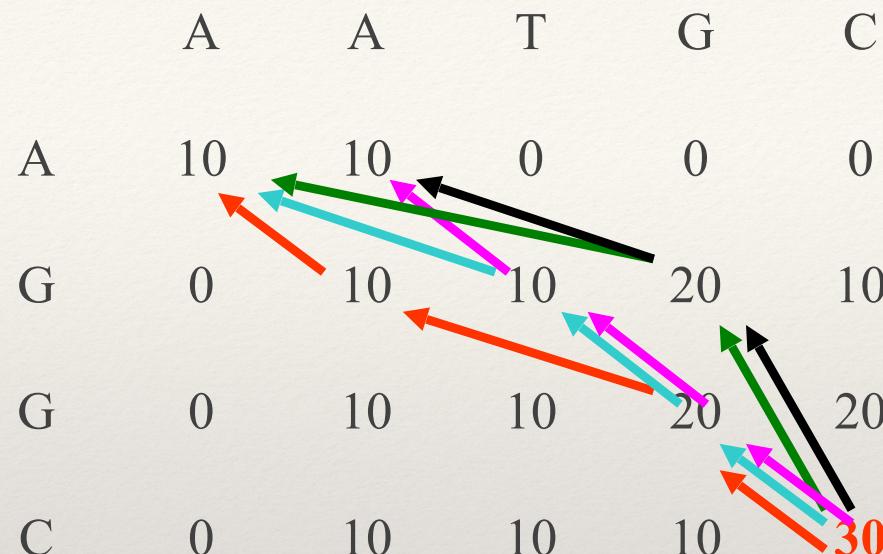
*Alignment:*

AWVCDEC

*Total score: 40*

AW-----EC

## Example 2



Alignments:

High Score: 30

AATGC

AATGC

AATGC

AATG C

AATG C

AG GC

A GGC

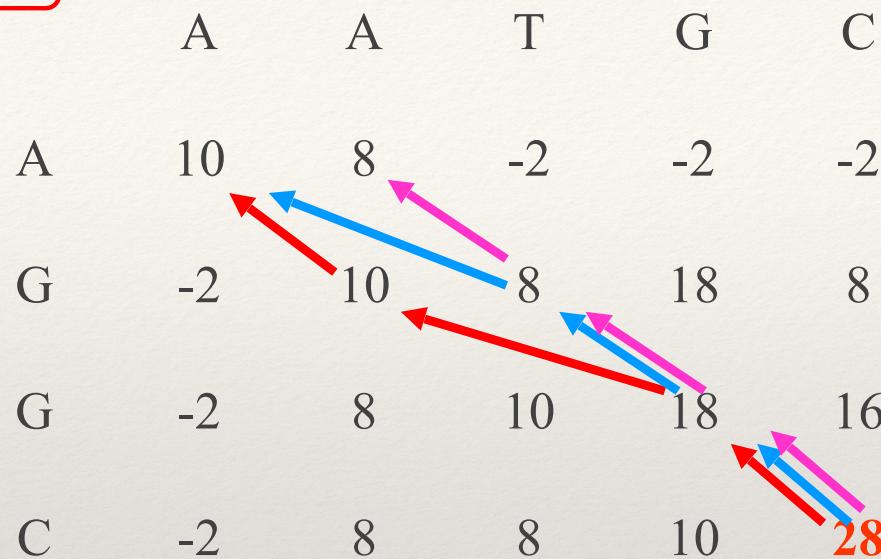
AGGC

A GGC

A GGC

## Example 3

Gap cost: -2



Alignments:

AATGC

AATGC

AATGC

AG GC

A GGC

AGGC

High Score: 28

## Statistical Significance of alignment: Shuffling

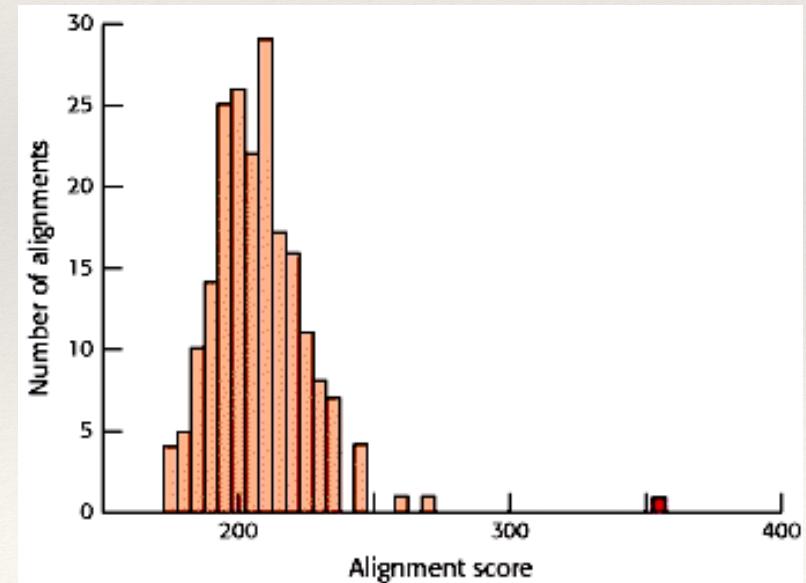


Score: 355

*Shuffling a sequence:*

THISISTHECORRECTSEQUENCE

TSTCRTQNHIHOESUCISERCEEE



## Gap penalty

Most common model:

$$W_N = G_0 + N * G_1$$

$W_N$ : gap penalty for a gap of size  $N$

$G_0$  : cost of opening a gap

$G_1$  : cost of extending the gap by one

$N$  : size of the gap

## Global versus Local Alignment

**Global alignment** finds the arrangement that maximizes total score

*Best known algorithm: Needleman and Wunsch.*

**Local alignment** identifies highest scoring subsequences,  
sometimes at the expense of the overall score.

*Best known algorithm: Smith and Waterman.*

***Local alignment algorithm is just a variation of the global alignment algorithm!***

## Modifications for local alignment

- 1) The scoring matrix has negative values for mismatches
- 1) The minimum score for any  $(i, j)$  in the alignment matrix is 0.
- 1) The best score is found anywhere in the filled alignment matrix

*These 3 modifications cause the algorithm to search for matching sub-sequences which are not penalized by other regions (modif. 2), with minimal poor matches (modif 1), which can occur anywhere (modif 3).*

## Global versus Local Alignment

*Match: +1; Mismatch: -2; Gap: -1*

	A	C	C	T	G	S
A	1	-3	-3	-3	-3	-3
C	-3	2	1	-2	-2	-2
C	-3	1	3	-1	-1	-1
N	-3	-2	-1	1	0	0
S	-3	-2	-1	0	-1	1

	A	C	C	T	G	S
A	1	0	0	0	0	0
C	0	2	1	0	0	0
C	0	1	3	0	0	0
N	0	0	0	1	0	0
S	0	0	0	0	0	1

*Global:*    ACCTGS    ACCTGS  
                  ACC-NS    ACCN-S

*Local:*    ACC  
                  ACC

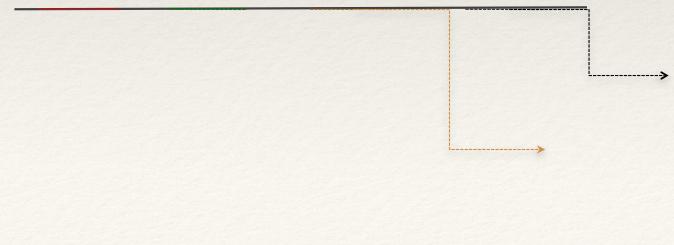
# BLAST

(Basic Local Alignment Search Tool)

*Main ideas:*

1. Construct a list of all words in the query sequence
2. Scan database for sequences that contain one or more of the query words
3. Initiate a local alignment for each word match between query and database

*Query sequence*



*Database*



# Original BLAST

## 1. Define dictionary

All words of length  $k$

(typically  $k=11$ )

## 2. Scan database sequences for matches

with alignment score  $\geq T$

(typically  $T = k$ )

## 3. Generate alignment

ungapped extensions until score

below statistical threshold

## 4. Output all local alignments with scores

above the statistical threshold

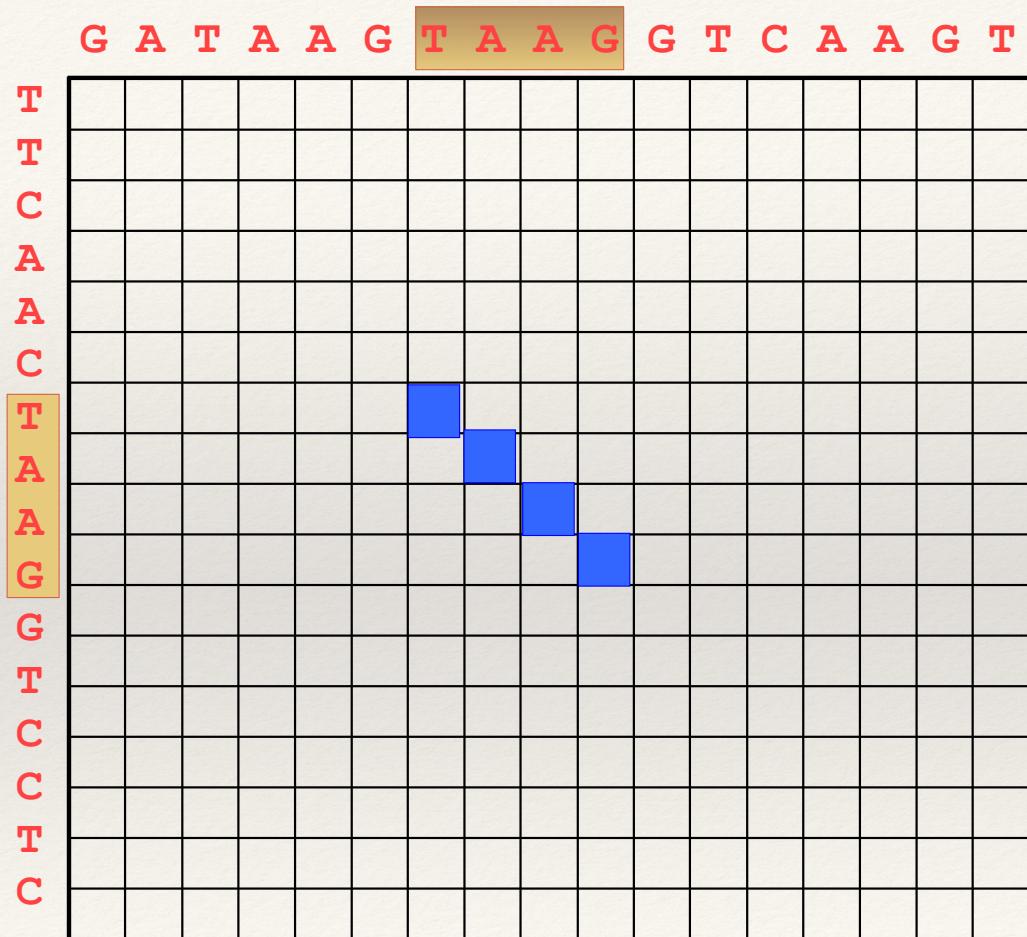


## Original BLAST

*An example:*

$k = 4, T = 4$

1) The matching word TAAG initiates an alignment

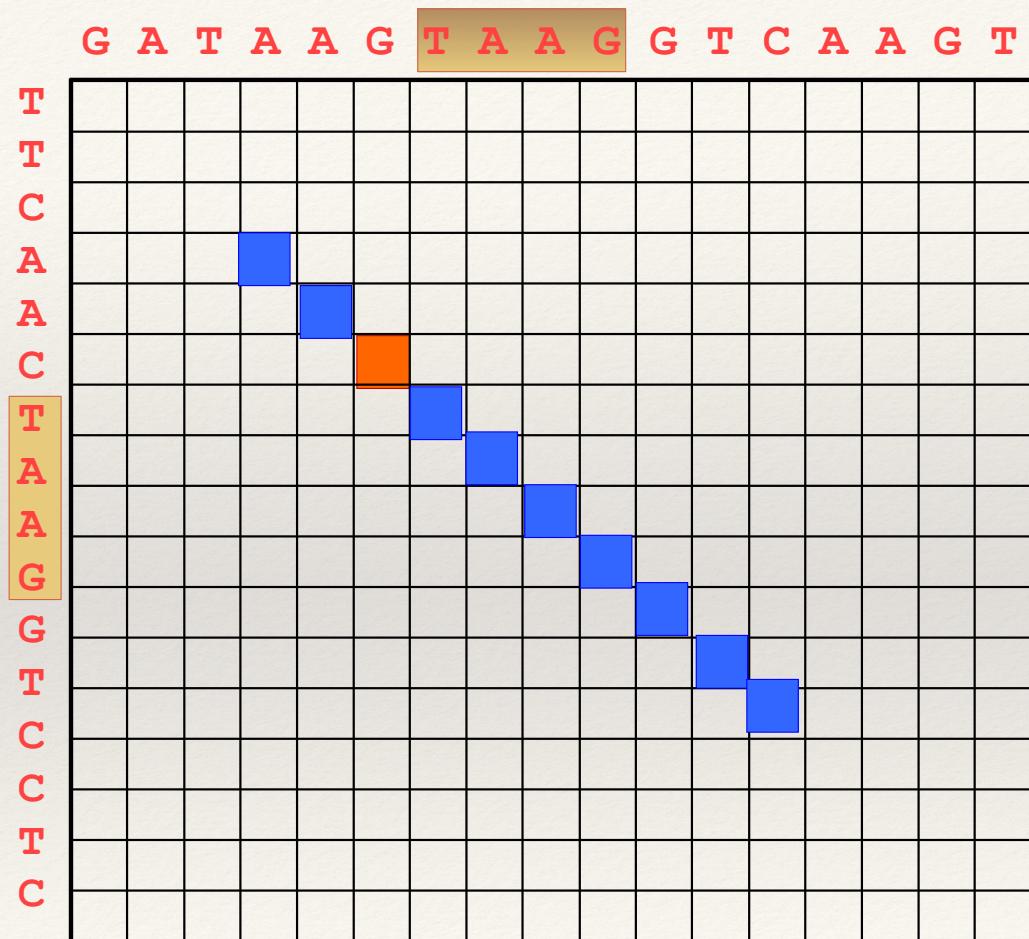


## Original BLAST

*An example:*

$k = 4, T = 4$

- 1) The matching word AGGT initiates an alignment
- 2) Extension of the alignment to the left and right with no gap until alignment score falls below 50%



## Original BLAST

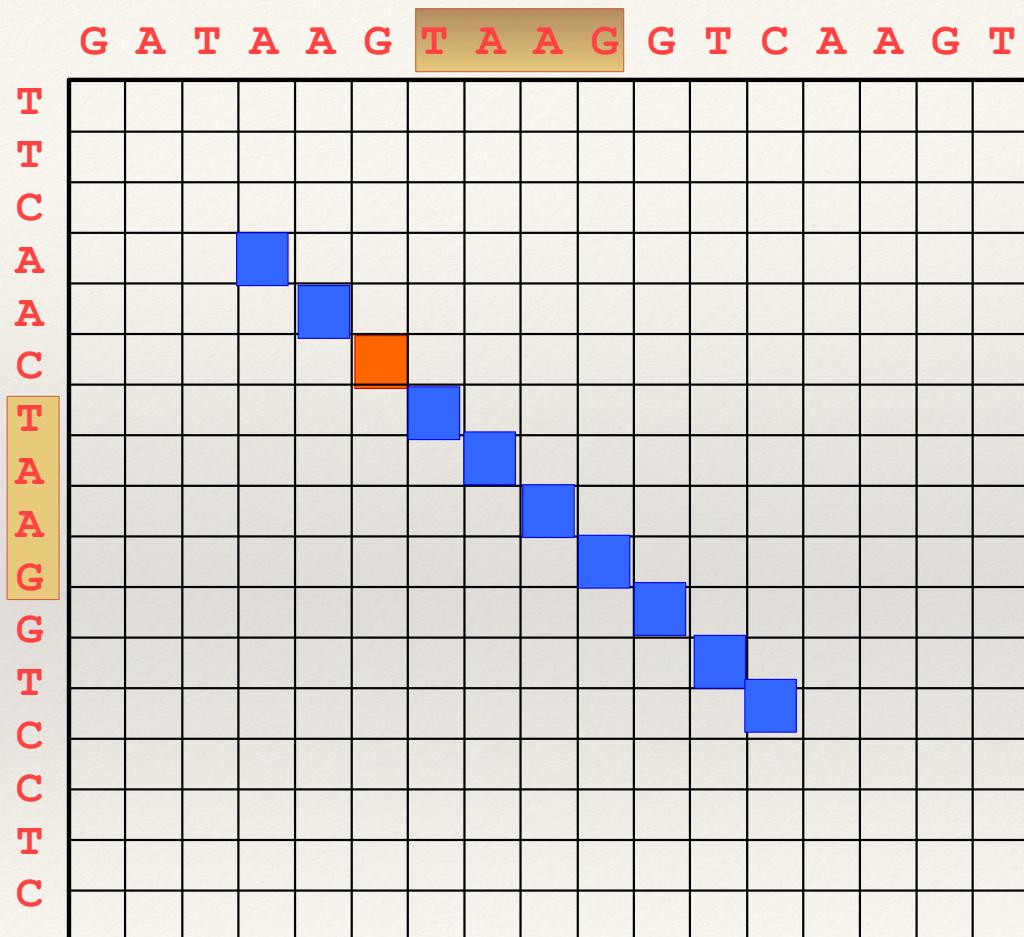
*An example:*

$k = 4, T = 4$

- 1) The matching word AGGT initiates an alignment
- 2) Extension of the alignment to the left and right with no gap until alignment score falls below 50%

3) Output:

**AAGTAAGGTC**  
**AACTAAGGTC**



# Gapped BLAST

*An example:*

$k = 4, T = 4$

- 1) The matching word GGTC  
initiates an alignment
- 2) Extend alignment in a band  
around anchor

# BLAST Portal

 **BLAST** *Basic Local Alignment Search Tool*

[Home](#) [Recent Results](#) [Saved Strategies](#) [Help](#)

► [NCBI/BLAST Home](#)

BLAST finds regions of similarity between biological sequences. [more...](#)

[Learn more about how to use the new BLAST design](#)

---

## BLAST Assembled Genomes

Choose a species genome to search, or [list all genomic BLAST databases](#).

<input type="checkbox"/> <a href="#">Human</a>	<input type="checkbox"/> <a href="#">Oryza sativa</a>	<input type="checkbox"/> <a href="#">Gallus gallus</a>
<input type="checkbox"/> <a href="#">Mouse</a>	<input type="checkbox"/> <a href="#">Bos taurus</a>	<input type="checkbox"/> <a href="#">Pan troglodytes</a>
<input type="checkbox"/> <a href="#">Rat</a>	<input type="checkbox"/> <a href="#">Danio rerio</a>	<input type="checkbox"/> <a href="#">Microbes</a>
<input type="checkbox"/> <a href="#">Arabidopsis thaliana</a>	<input type="checkbox"/> <a href="#">Drosophila melanogaster</a>	<input type="checkbox"/> <a href="#">Apis mellifera</a>

---

## Basic BLAST

Choose a BLAST program to run.

<a href="#">nucleotide blast</a>	Search a <b>nucleotide</b> database using a <b>nucleotide</b> query <i>Algorithms:</i> blastn, megablast, discontiguous megablast
<a href="#">protein blast</a>	Search <b>protein</b> database using a <b>protein</b> query <i>Algorithms:</i> blastp, psi-blast, phi-blast
<a href="#">blastx</a>	Search <b>protein</b> database using a <b>translated nucleotide</b> query
<a href="#">tblastn</a>	Search <b>translated nucleotide</b> database using a <b>protein</b> query
<a href="#">tblastx</a>	Search <b>translated nucleotide</b> database using a <b>translated nucleotide</b> query

# BLAST: Input

► [NCBI/ BLAST/ blastp suite: BLASTP programs search protein databases using a protein query.](#) [more...](#) [Reset page](#) [Bookmark](#)

**Enter Query Sequence**

Enter accession number, gi, or FASTA sequence [?](#) [Clear](#)

>1CTF:A|PDBID|CHAIN|SEQUENCE  
AAEEKTEFDVILKAAGANKVAVIKAVRGATGLGLKEAKDLVESAPAALKEGVSKDDAEALKALEE  
AGAEVEVK

Query subrange [?](#)

From   
To

Or, upload file [Choose File](#) no file selected [?](#)

Job Title   
Enter a descriptive title for your BLAST search [?](#)

**Choose Search Set**

Database [Non-redundant protein sequences \(nr\)](#) [?](#)

Organism [Optional](#)  
Enter organism name or id--completions will be suggested

Enter organism common name, binomial, or tax id. Only 20 top taxa will be shown. [?](#)

Entrez Query [Optional](#)  
Enter an Entrez query to limit search [?](#)

**Program Selection**

Algorithm  **blastp (protein-protein BLAST)**  
 **PSI-BLAST (Position-Specific Iterated BLAST)**  
 **PHI-BLAST (Pattern Hit Initiated BLAST)**  
Choose a BLAST algorithm [?](#)

**BLAST**   
Search database **nr** using **Blastp (protein-protein BLAST)**  
 Show results in a new window

► [Algorithm parameters](#)

# BLAST Parameters

▼ Algorithm parameters

General Parameters

**Max target sequences**  Select the maximum number of aligned sequences to display 

**Short queries**  Automatically adjust parameters for short input sequences 

**Expect threshold**  

**Word size**  

Scoring Parameters

**Matrix**  

**Gap Costs** Existence:  Extension:  

**Compositional adjustments**  

Filters and Masking

**Filter**  Low complexity regions 

**Mask**  Mask for lookup table only   
 Mask lower case letters 

# BLAST Results

Distance tree of results [NEW](#) [Related Structures](#)

Sequences producing significant alignments:

		Score (Bits)	E Value	
prf 0601198A	polymerase beta, RNA	114	2e-24	
ref NP_660396.2	50S ribosomal protein L7/L12 [Buchnera aphid...	85.5	1e-15	<b>G</b>
ref NP_239876.1	50S ribosomal protein L7/L12 [Buchnera aphid...	84.0	3e-15	<b>G</b>
sp P41188 RL7_BUCAPI	50S ribosomal protein L7/L12 >gb AAM67607...	82.8	5e-15	
ref YP_001337990.1	50S ribosomal protein L7/L12 [Klebsiella ...	80.5	3e-14	<b>G</b>
ref YP_001454539.1	hypothetical protein CKO_03003 [Citrobact...	80.1	4e-14	<b>G</b>
ref YP_001174937.1	ribosomal protein L7/L12 [Enterobacter sp...	79.3	7e-14	<b>G</b>
ref YP_001439732.1	hypothetical protein ESA_03692 [Enterobac...	79.3	7e-14	<b>G</b>
ref NP_457918.1	50S ribosomal protein L7/L12 [Salmonella ent...	79.3	7e-14	<b>G</b>
ref YP_453813.1	50S ribosomal subunit protein L7/L12 [Sodali...	79.0	7e-14	<b>G</b>
ref YP_312899.1	50S ribosomal subunit protein L7/L12 [Shigel...	79.0	8e-14	<b>G</b>
ref NP_290617.1	50S ribosomal protein L7/L12 [Escherichia co...	78.6	1e-13	<b>G</b>
ref YP_001476514.1	ribosomal protein L7/L12 [Serratia protea...	78.2	1e-13	<b>G</b>
ref YP_588936.1	ribosomal protein L7/L12 [Baumannnia cicadell...	77.8	2e-13	<b>G</b>
ref NP_927791.1	50S ribosomal protein L7/L12 (L8) [Photorhab...	77.4	2e-13	<b>G</b>
pdb 2CYA 3	Chain 3, Structure Of The 50s Subunit Of A Pre-Tra...	77.4	2e-13	<b>S</b>
pdb 1RQU A	Chain A, Nmr Structure Of L7 Dimer From E.Coli >pd...	77.4	2e-13	<b>S</b>
ref NP_777674.1	50S ribosomal protein L7/L12 [Buchnera aphid...	77.4	2e-13	<b>G</b>
ref YP_219023.1	50S ribosomal protein L7/L12 [Salmonella ent...	77.4	3e-13	<b>G</b>
ref YP_048349.1	50S ribosomal protein L7/L12 [Erwinia caroto...	77.4	3e-13	
ref ZP_00798031.1	COG0222: Ribosomal protein L7/L12 [Yersinia p...	75.5	9e-13	
ref ZP_00827822.1	COG0222: Ribosomal protein L7/L12 [Yersini...	75.5	1e-12	
ref ZP_00821082.1	COG0222: Ribosomal protein L7/L12 [Yersini...	75.5	1e-12	

# Statistics of Protein Sequence Alignment

- ❖ *Statistics of global alignment:*

Unfortunately, not much is known! Statistics based on Monte Carlo simulations (shuffle one sequence and recompute alignment to get a distribution of scores)

- ❖ *Statistics of local alignment*

Well understood for ungapped alignment. Same theory probably apply to gapped-alignment

# Statistics of Protein Sequence Alignment

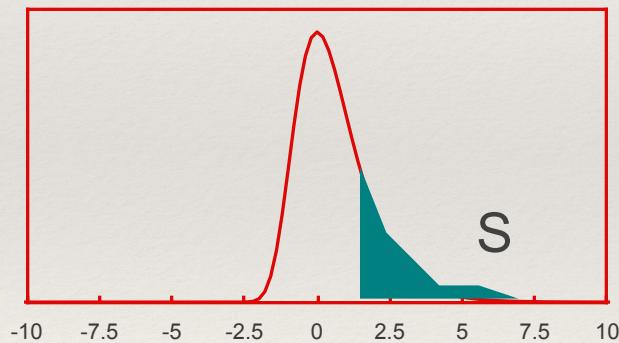
## *What is a local alignment ?*

“Pair of equal length segments, one from each sequence, whose scores can not be improved by extension or trimming. These are called high-scoring pairs, or HSP”

<http://www.people.virginia.edu/~wrp/cshl98/Altschul/Altschul-1.html>

# The E-value for a sequence alignment

HSP scores follow an extreme value distribution, characterized by two parameters,  $K$  and  $\lambda$ .



The expected number of HSP with score at least  $S$  is given by:

$$E = Kmn \exp(-\lambda S)$$

$m, n$  : sequence lengths

$E$  : E-value

Raw scores have little meaning without knowledge of the scoring scheme used for the alignment, or equivalently of the parameters  $K$  and  $\lambda$ .

Scores can be normalized according to:

$$S' = \frac{\lambda S - \ln(K)}{\ln(2)}$$

$S'$  is the **bit score** of the alignment.

The E-value can be expressed as:

$$E = mn2^{-S'}$$

## The P-value of a sequence alignment

The number of random HSP with score greater or equal to S follows a Poisson distribution:

$$P(X \text{ random HSP with score } \geq S) = \exp(-E) \frac{E^X}{X!}$$

(E: E-value)

Then:

$$P(0 \text{ random HSP with score } \geq S) = \exp(-E)$$

$$P_{val} = P(\text{at least 1 random HSP with score } \geq S) = 1 - \exp(-E)$$

Note: when  $E \ll 1$ ,  $P \approx E$

## The database E-value for a sequence alignment

2) Longer sequences are more likely to be related to the query:

$$E_{DB} = N_S K m n \exp(-\lambda S)$$

BLAST reports  $E_{DB2}$

$$E_{DB2} = K m N_R \exp(-\lambda S)$$

## Why multiple sequence alignment?

Seq1 : AALG**C**LVKDYFPEP--VTVS**W**NSG---

Seq2 : VSLT**C**LVKGFYPSD--IAVE**WW**SNG--

## Why multiple sequence alignment?

Seq1 : AALG**C**LVKDYFPEP--VTVS**W**NSG---

Seq2 : VSLT**C**LVKGFYPSD--IAVE**W**WSNG--

Seq3 : VTIS**C**TGSSSNIGAG-NHVK**W**Y**Q****Q****L****P****G**

Seq4 : VTIS**C**TGTSSNIIGS--ITVN**W**Y**Q****Q****L****P****G**

Seq5 : LRLS**C**SSSGFIFSS--YAMY**W**VR**Q****A****P****G**

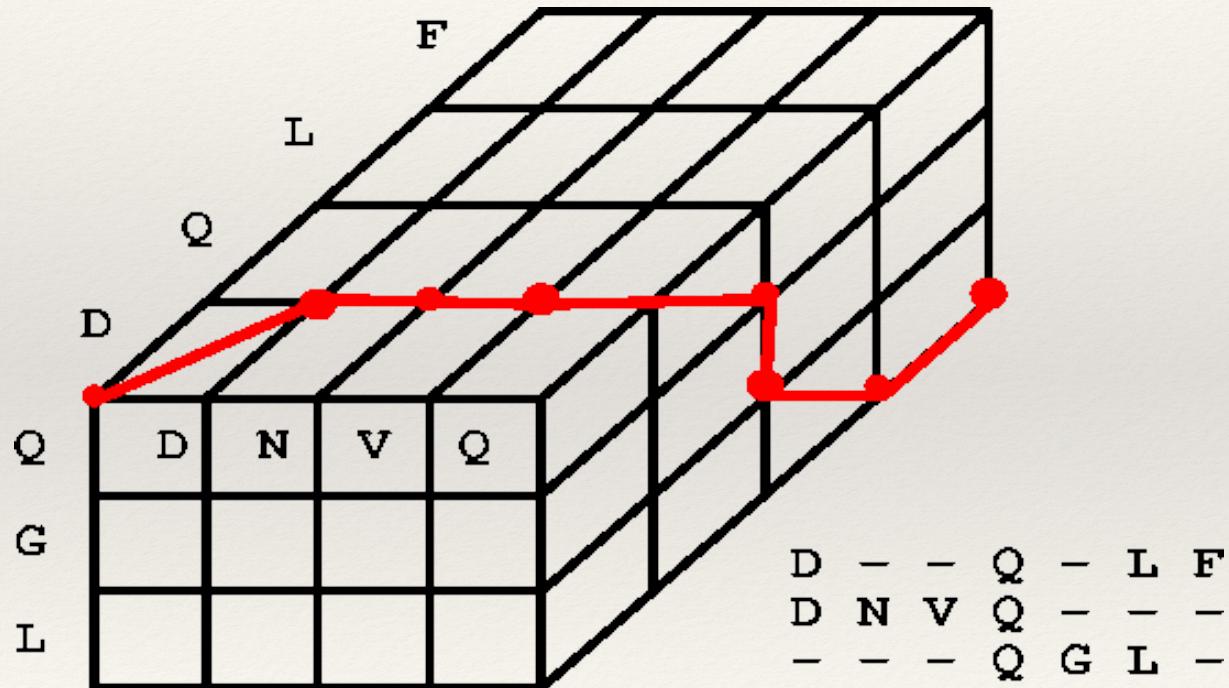
Seq6 : LSLT**C**TVSGTSFDD--YYST**W**VR**Q****P****P****G**

Seq7 : PEVT**C**VVVVDVSHEDPQVKFN**W**YVDG--

Seq8 : ATLV**C**LISDFYPGA--VTVA**W**KADS--

## MSA: Dynamic programming?

*Theoretically, it is possible to extend the dynamic programming technique to N sequences.*



## MSA: Dynamic programming?

- One of the most important properties of an algorithm is how its execution time increases as the problem is made larger.  
This is **the computational complexity** of the algorithm
- There is a notation to describe the algorithmic complexity, called **the big-O notation**.  
If we have a problem of size (i.e. number of input data points)  $n$ , then an algorithm takes  **$O(n)$**  time if the time increases linearly with  $n$ .
- It is important to realize that an algorithm that is **quick on small problems may be totally useless on large problems** if it has a bad  $O()$  behavior.

## MSA: Dynamic programming?

Standard description of algorithms, where n is the size of the problem, and c is a constant:

Complexity	Type	Computing time for n=1000 (1 operation=1s)
$O(c)$	Dream...	Seconds
$O(\log(n))$	Really good	10 seconds
$O(n)$	good	1000 seconds = 5 mins
$O(n^2)$	Not so good	$10^6$ seconds = 11.5 days
$O(n^3)$	Bad	$10^9$ seconds = 31 years
$O(c^n)$	Catastrophic!	Millions of years!!

## MSA: Dynamic programming?

*Computational complexity of dynamic programming:*

- Two sequences of length M :  **$O(M^2)$**
- Three sequences of length M:  **$O(M^3)$**
- N sequences of length M:  **$O(M^N)$**

-> dynamic programming is not a reasonable option for aligning multiple sequences!

## MSA: Approximate methods

### 1. Progressive global alignment

Start with the most similar sequences and builds the alignment by adding the rest of the sequences

### 2. Iterative methods

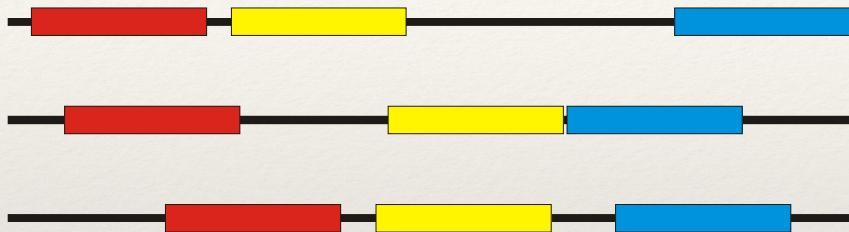
Start by making alignments of small group of sequences and then revise the alignment for better results

### 3. Alignment based on small conserved domains

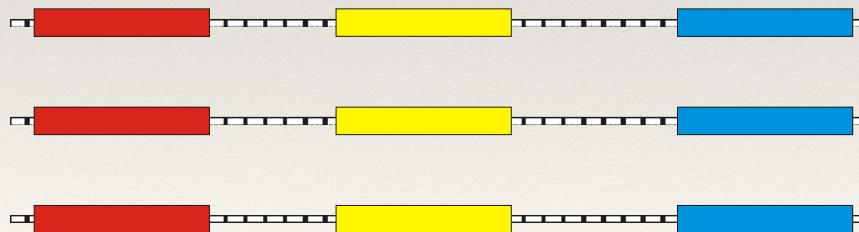
### 4. Alignment based on statistical or probabilistic models of the sequence

## Multiple sequence alignment: using conserved domains

*Sequences often contain highly  
conserved regions*



*These regions can be used for an initial alignment*



## How to generate a multiple sequence alignment?

### Raw Alignment

Human N Y L S

Chimp N K Y L S

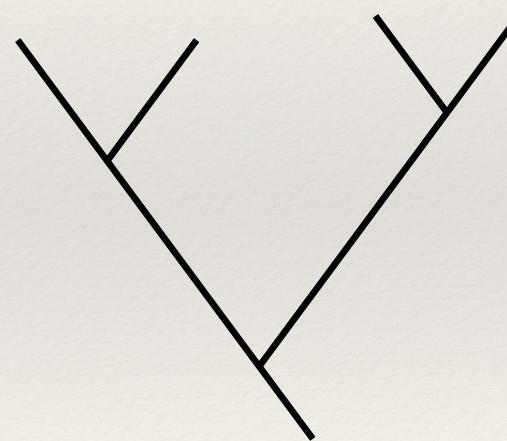
Gorilla N F S

Orangutan N F L S

## How to generate a multiple sequence alignment?

*Sequence elements are not truly independent but related by phylogeny:*

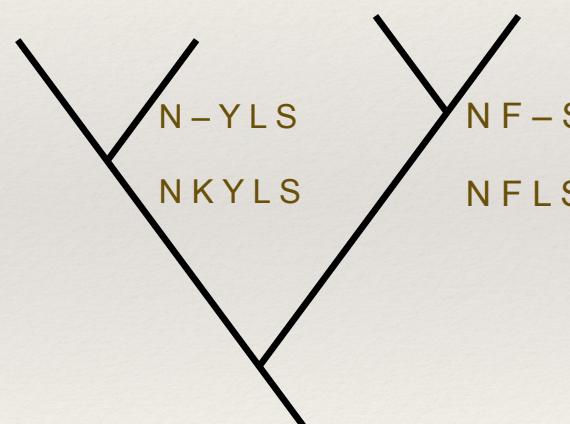
	Raw Alignment	NYLS	NKYLS	NFS	NFLS
		Human	Chimp	Gorilla	Orangutan
Human	N Y L S				
Chimp	N K Y L S				
Gorilla	N F S				
Orangutan	N F L S				



## How to generate a multiple sequence alignment?

*Sequence elements are not truly independent but related by phylogeny:*

Raw Alignment		NYLS	NKYLS	NFS	NFLS
		Human	Chimp	Gorilla	Orangutan
Human	NYLS				
Chimp	NKYLS		N-YLS		NF-S
Gorilla	NFS		NKYLS		NFLS
Orangutan	NFLS				



## How to generate a multiple sequence alignment?

*Sequence elements are not truly independent but related by phylogeny:*

### Raw Alignment

Human NYLS

Chimp NKYS

Gorilla NFS

OrangutanNFLS

NYLS NKYLS NFS NFLS  
Human Chimp Gorilla Orangutan



## Multiple sequence alignment: Progressive method

### A) Perform pairwise alignments

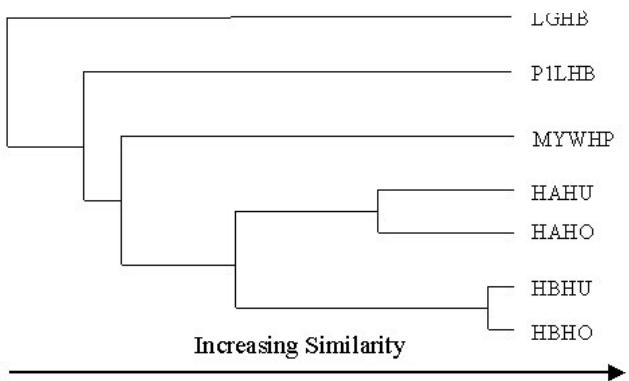
	HAHU	HBHU	HAHO	HBHO	MYWHP	P1LHB	LGHB
HAHU							
HBHU	21.1						
HAHO	32.9	19.7					
HBHO	20.7	<b>39.0</b>	20.4				
MYWHP	11.0	9.8	10.3	9.7			
P1LHB	9.3	8.6	9.6	8.4	7.0		
LGHB	7.1	7.3	7.5	7.4	7.3	4.3	

## Multiple sequence alignment: Progressive method

### A) Perform pairwise alignments

	HAHU	HBHU	HAHO	HBHO	MYWHP	P1LHB	LGHB
HAHU							
HBHU	21.1						
HAHO	32.9	19.7					
HBHO	20.7	<b>39.0</b>	20.4				
MYWHP	11.0	9.8	10.3	9.7			
P1LHB	9.3	8.6	9.6	8.4	7.0		
LGHB	7.1	7.3	7.5	7.4	7.3	4.3	

### B) Cluster based on similarity

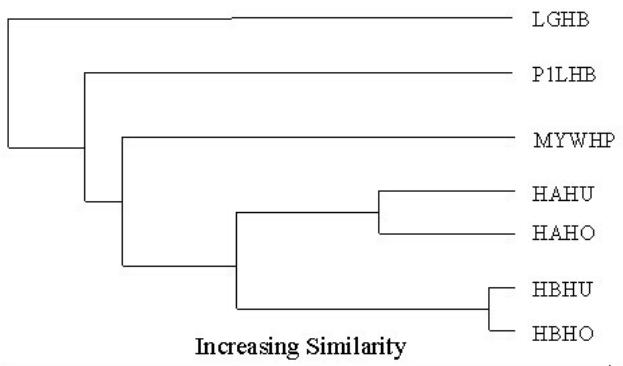


## Multiple sequence alignment: Progressive method

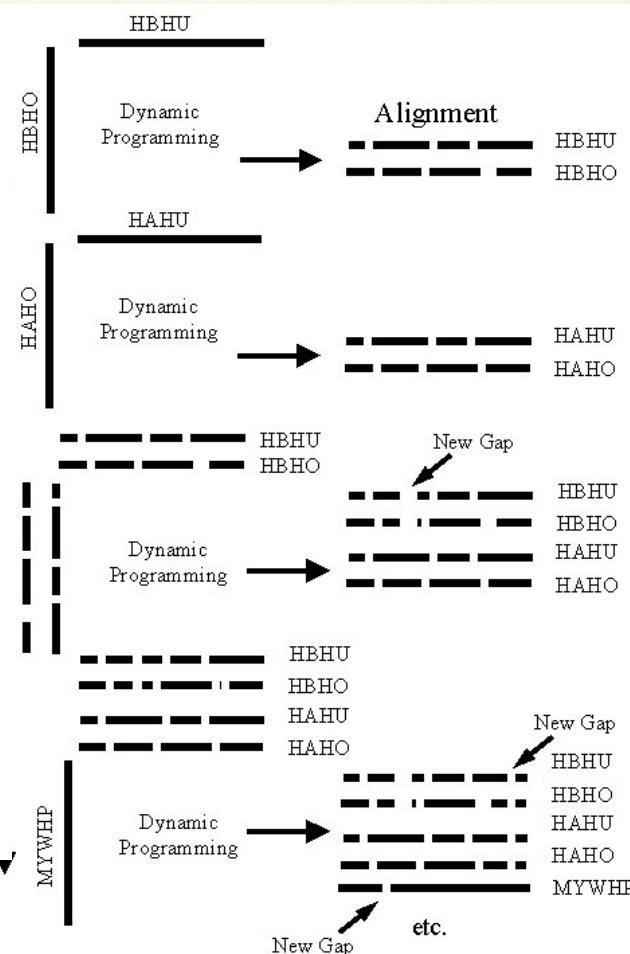
### A) Perform pairwise alignments

	HAHU	HBHU	HAHO	HBHO	MYWHP	P1LHB	LGHB
HAHU							
HBHU	21.1						
HAHO	32.9	19.7					
HBHO	20.7	<b>39.0</b>	20.4				
MYWHP	11.0	9.8	10.3	9.7			
P1LHB	9.3	8.6	9.6	8.4	7.0		
LGHB	7.1	7.3	7.5	7.4	7.3	4.3	

### B) Cluster based on similarity



### C) Generate Multiple Sequence Alignment



## Some References on Alignments

### **Global Alignment:**

Needleman, S.B. and Wunsch, C.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

### **Local alignment:**

Smith, T.F. and Waterman, M.S. (1981) "Identification of Common Molecular Subsequences". *Journal of Molecular Biology* 147: 195–197

### **ClustalW:**

Thompson, J. D., Higgins, D.G. and Gibson, T.J. (1994) "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice".

*Nucleic Acids Research*, 22:4673-4680

# What have we learnt?

1) **Sequence analysis** is one of the keys that will help us unravel the information coming from Genomics

## 2) Vocabulary

**Analogy:** The similarity of characteristics between two species that are not closely related

**Homology:** Similarity in characteristics resulting from shared ancestry

- **Paralog:** Homologous sequences are paralogous if they were separated by a gene duplication event
- **Ortholog:** Homologous sequences are orthologous if they were separated by a speciation event

3) In bioinformatics we often assume that **sequence similarity implies homology**. However we do need to be cautious.

## What have we learnt?

- 4) Sequence analysis starts with **an analysis of its content**

- 1) **DNAs:**

- Chargaff rule2:** the composition of DNA varies from one species to another

- 2) **Proteins:**

- Tri-peptide content identifies the kingdom of life  
(bacteria, archea or eukaryot)

- 5) **DotPlots** are very useful, qualitative tools for sequence comparison
- 4) **Scoring** between sequences is usually based on **substitution matrices**

Most common matrices: **PAM** and **BLOSUM**

## What have we learnt?

1. **Dynamic programming (DP)** is an algorithm for aligning two sequences that is guaranteed to generate the **optimal alignment**, under the hypothesis that the **scores are additive**.
2. There are two variants of DP used for sequence analysis  
**Global alignment:** Needleman and Wunsch  
**Local alignment:** Smith and Waterman
3. DP is too slow for comparing a sequence with a large database
4. **BLAST** provides a heuristic method for detecting sequences that are similar
5. **BLAST is best for detection** and should not be trusted for the alignment itself

# What have we learnt?

## 6) Multiple sequence alignment: definition

A multiple sequence alignment is an alignment of  $n > 2$  sequences obtained by inserting gaps ("–") into sequences such that the resulting sequences have all length  $L$ . MSA can help to reveal biological facts about proteins, to establish homology,...

## 7) Difficulties in generating MSA

Most pairwise alignment algorithms are too complex to be used for  $N$ -wise alignments

## 8) Three main types of MSA algorithms:

- Progressive global alignment (starts with the most alike sequences)
  - \* e.g., ClustalW, ClustalX
- Iterative methods (initial alignment of groups of sequences that are revised)
  - \* MultAlin, PRRP, SAGA
- Alignments based on locally conserved patterns