



# BGGN 213

## Foundations of Bioinformatics

Barry Grant  
UC San Diego

<http://thegrantlab.org/bggcn213>

### Recap From Last Time:

25 Responses:

<https://tinyurl.com/bggcn213-02-F17>

## ALIGNMENT FOUNDATIONS

- **Why...**
  - ▶ Why compare biological sequences?
- **What...**
  - ▶ Alignment view of sequence changes during evolution (matches, mismatches and gaps)
- **How...**
  - ▶ Dot matrices
  - ▶ Dynamic programming
    - Global alignment
    - Local alignment
  - ▶ BLAST heuristic approach

## ALIGNMENT FOUNDATIONS

- **Why...**
  - ▶ Why compare biological sequences?
- **What...**
  - ▶ Alignment view of sequence changes during evolution (matches, mismatches and gaps)
- **How...**
  - ▶ Dot matrices
  - ▶ Dynamic programming
    - Global alignment
    - Local alignment
  - ▶ BLAST heuristic approach

**Basic Idea:** Display one sequence above another with spaces (termed **gaps**) inserted in both to reveal **similarity** of nucleotides or amino acids.

Seq1 : C A T T C A C  
 Seq2 : C T C G C A G C

[Screencast Materia

**Basic Idea:** Display one sequence above another with spaces (termed **gaps**) inserted in both to reveal **similarity** of nucleotides or amino acids.

Seq1 : C A T T C A C  
 Seq2 : C T C G C A G C

Two types of character correspondence

**Basic Idea:** Display one sequence above another with spaces (termed **gaps**) inserted in both to reveal **similarity** of nucleotides or amino acids.

Seq1 : C A T - T C A - C  
 Seq2 : C - T C G C A G C

Add gaps to increase number of matches  
**gaps**

**Basic Idea:** Display one sequence above another with spaces (termed **gaps**) inserted in both to reveal **similarity** of nucleotides or amino acids.

Seq1 : C A T - T C A - C  
 Seq2 : C - T C G C A G C

match } mutation  
 mismatch }  
 insertion } indels  
 deletion }

Gaps represent 'indels'  
 mismatch represent mutations

## Why compare biological sequences?

- To obtain **functional or mechanistic insight** about a sequence by inference from another potentially better characterized sequence
- To find whether two (or more) genes or proteins are **evolutionarily related**
- To find **structurally or functionally similar regions** within sequences (e.g. catalytic sites, binding sites for other molecules, etc.)
- Many practical bioinformatics applications...

## Practical applications include...

- **Similarity searching of databases**
  - Protein structure prediction, annotation, etc...
- **Assembly of sequence reads** into a longer construct such as a genomic sequence
- **Mapping sequencing reads to a known genome**
  - "Resequencing", looking for differences from reference genome - SNPs, indels (insertions or deletions)
  - Mapping transcription factor binding sites via ChIP-Seq (chromatin immuno-precipitation sequencing)
  - Pretty much all next-gen sequencing data analysis

## Practical applications include...

- **Similarity searching of databases**
  - Protein structure prediction
- **Assembly of sequence reads** into a longer construct such as a genomic sequence
- **Mapping sequencing reads to a known genome**
  - "Resequencing", looking for differences from reference genome - SNPs, indels (insertions or deletions)
  - Mapping transcription factor binding sites via ChIP-Seq (chromatin immuno-precipitation sequencing)
  - Pretty much all next-gen sequencing data analysis

**N.B.** Pairwise sequence alignment is arguably the most fundamental operation of bioinformatics!

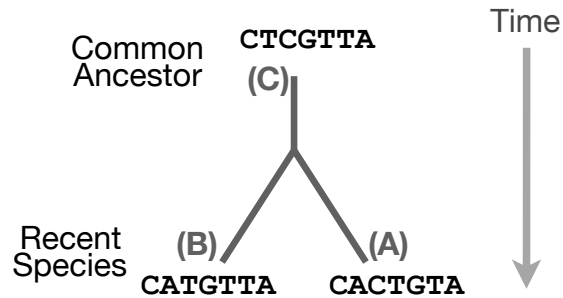
## ALIGNMENT FOUNDATIONS

- **Why...**
  - Why compare biological sequences?
- **What...**
  - ▶ Alignment view of sequence changes during evolution (matches, mismatches and gaps)
- **How...**
  - ▶ Dot matrices
  - ▶ Dynamic programming
    - Global alignment
    - Local alignment
  - ▶ BLAST heuristic approach

## Sequence changes during evolution

There are three major types of sequence change that can occur during evolution.

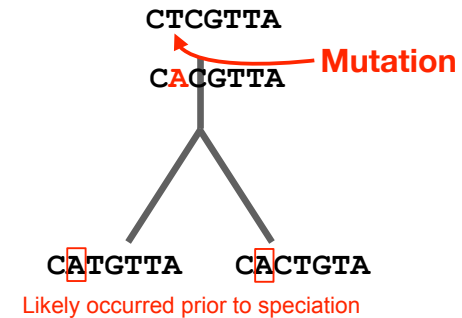
- Mutations/Substitutions
- Deletions
- Insertions



## Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

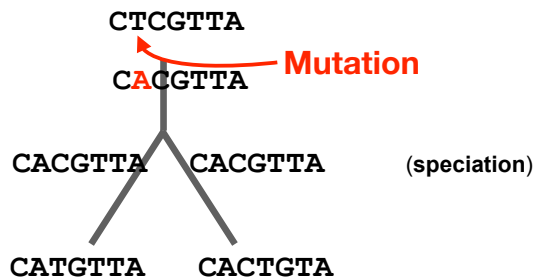
- **Mutations/Substitutions**    CTCGTTA → CACGTTA
- Deletions
- Insertions



## Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

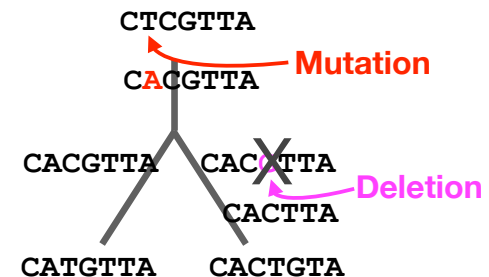
- Mutations/Substitutions    CTCGTTA → CACGTTA
- Deletions
- Insertions



## Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

- Mutations/Substitutions    CTCGTTA → CACGTTA
- **Deletions**    CACGTTA → CACTTA
- Insertions

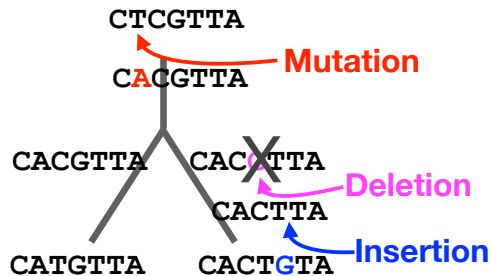


## Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

- Mutations/Substitutions
- Deletions
- Insertions

CTCGTTA → CACGTTA  
 CACGTTA → CACTTA  
 CACTTA → CACTGTA

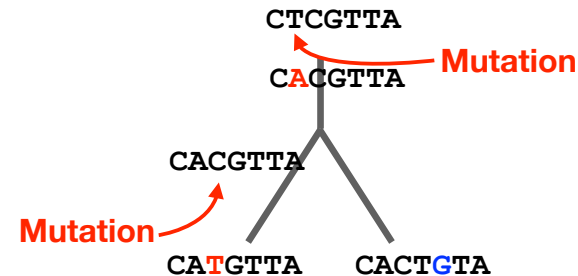


## Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

- **Mutations/Substitutions**
- Deletions
- Insertions

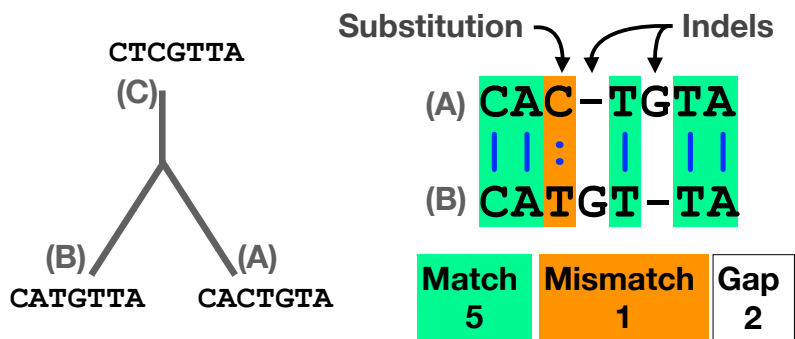
CTCGTTA → CACGTTA  
 CACGTTA → CATGTTA



## Alignment view

Alignments are great tools to visualize sequence similarity and evolutionary changes in homologous sequences.

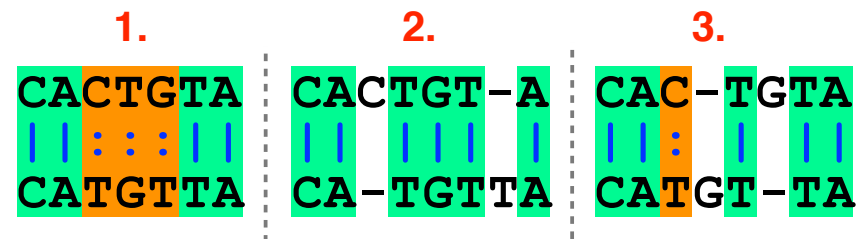
- **Mismatches** represent mutations/substitutions
- **Gaps** represent insertions and deletions (indels)



## Alternative alignments

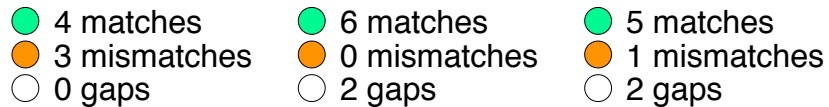
- Unfortunately, finding the correct alignment is difficult if we do not know the evolutionary history of the two sequences

Q. Which of these 3 possible alignments is best?



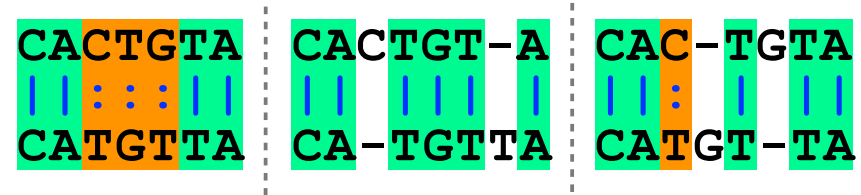
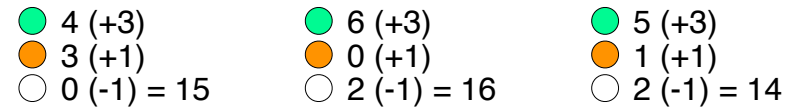
## Alternative alignments

- One way to judge alignments is to compare their number of matches, insertions, deletions and mutations



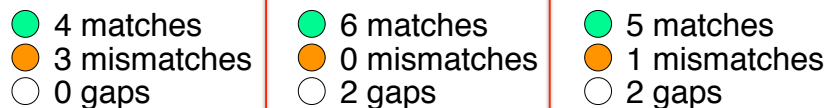
## Scoring alignments

- We can assign a score for each match (+3), mismatch (+1) and indel (-1) to identify the **optimal alignment for this scoring scheme**



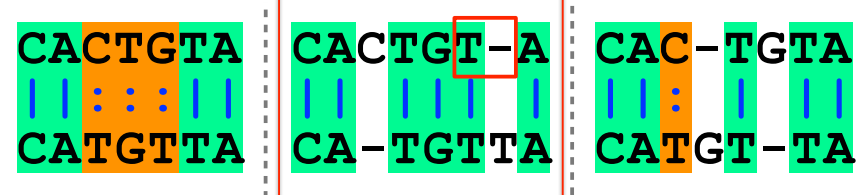
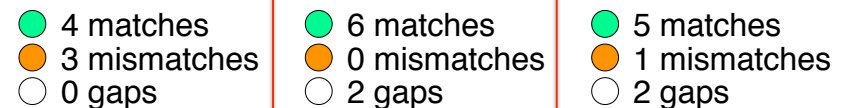
## Optimal alignments

- Biologists often prefer **parsimonious alignments**, where the number of postulated sequence changes is minimized.



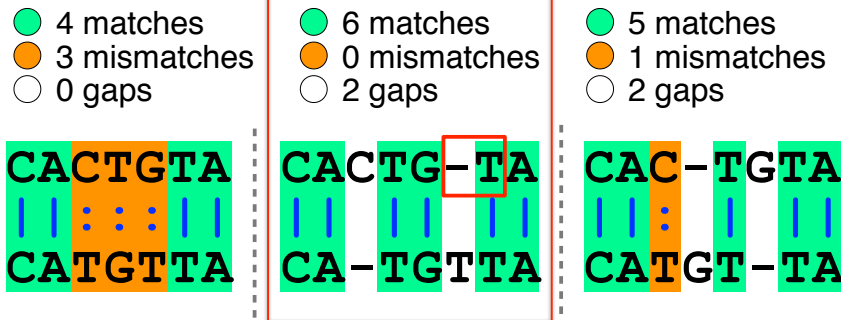
## Optimal alignments

- Biologists often prefer **parsimonious alignments**, where the number of postulated sequence changes is minimized.



## Optimal alignments

- Biologists often prefer **parsimonious alignments**, where the number of postulated sequence changes is minimized.



## Optimal alignments

- Biologists often prefer **parsimonious alignments**, where the number of postulated sequence changes is minimized.
- 4 matches  
● 3 mismatches  
○ 0 gaps
- 6 matches  
● 0 mismatches  
○ 2 gaps
- 5 matches  
● 1 mismatches  
○ 2 gaps
- CACTGTA  
 |||:  
 CATGTTA
- CACTGT-A  
 ||| |||  
 CA-TGTTA
- CAC-TGTA  
 |||:  
 CATGT-TA
- Warning:** There may be more than one optimal alignment and these may not reflect the true evolutionary history of our sequences!

## ALIGNMENT FOUNDATIONS

- Why...**
  - Why compare biological sequences?
- What...**
  - Alignment view of sequence changes during evolution (matches, mismatches and gaps)

- How...**
  - Dot matrices
  - Dynamic programming
    - Global alignment
    - Local alignment
  - BLAST heuristic approach

## ALIGNMENT FOUNDATIONS

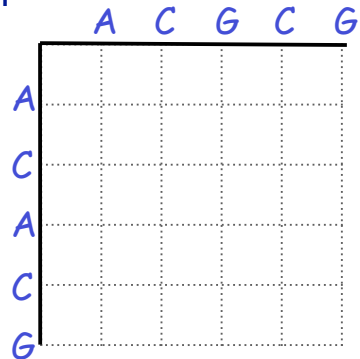
- Why...**
  - Why compare biological sequences?
- What...**
  - Alignment view of sequence changes during evolution (matches, mismatches and gaps)

- How...**
  - Dot matrices
  - Dynamic programming
    - Global alignment
    - Local alignment
  - BLAST heuristic approach

How do we compute the optimal alignment between two sequences?

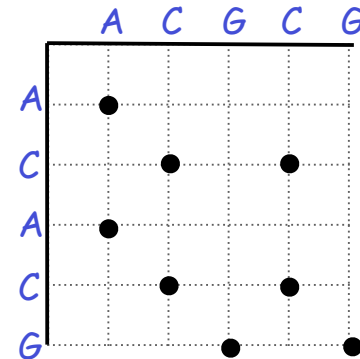
## Dot plots: simple graphical approach

- Place one sequence on the vertical axis of a 2D grid (or matrix) and the other on the horizontal



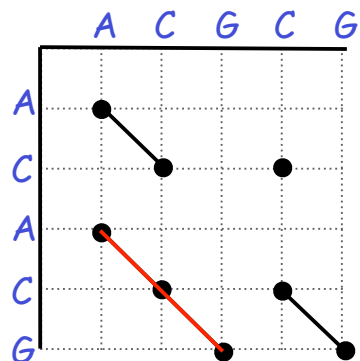
## Dot plots: simple graphical approach

- Now simply put dots where the horizontal and vertical sequence values match



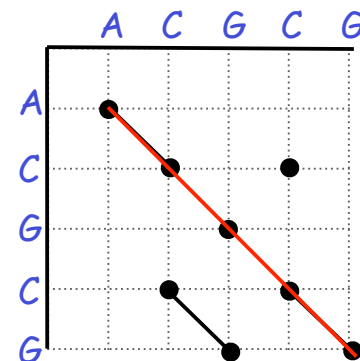
## Dot plots: simple graphical approach

- Diagonal runs of dots indicate matched segments of sequence



## Dot plots: simple graphical approach

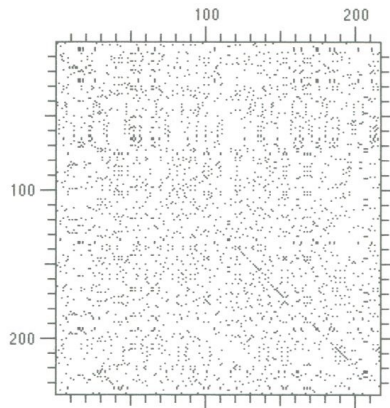
- Q.** What would the dot matrix of a two identical sequences look like?





## Dot plots: simple graphical approach

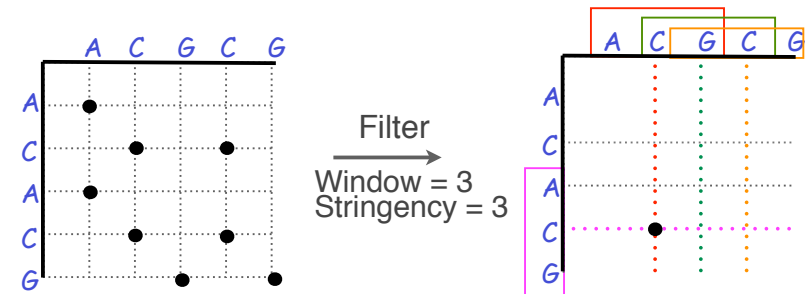
- Dot matrices for long sequences can be noisy



## Dot plots: window size and match stringency

**Solution:** use a window and a threshold

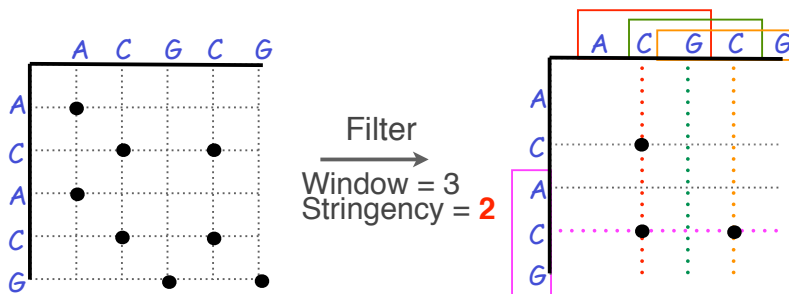
- compare character by character within a window
- require certain fraction of matches within window in order to display it with a dot.
- You have to choose window size and stringency



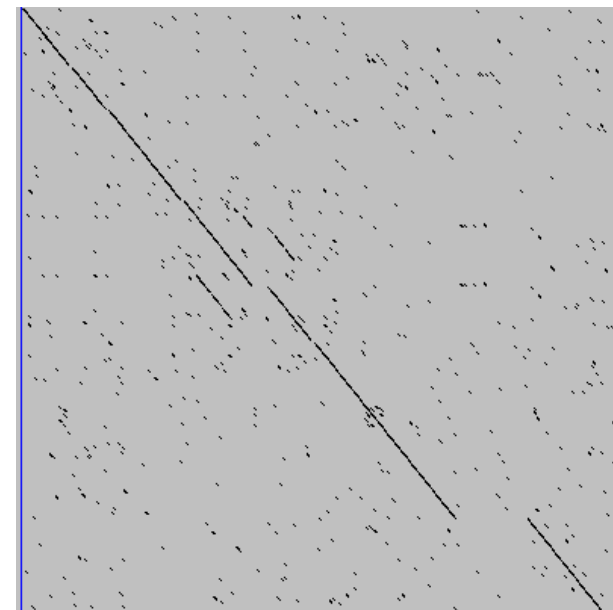
## Dot plots: window size and match stringency

**Solution:** use a window and a threshold

- compare character by character within a window
- require certain fraction of matches within window in order to display it with a dot.
- You have to choose window size and stringency



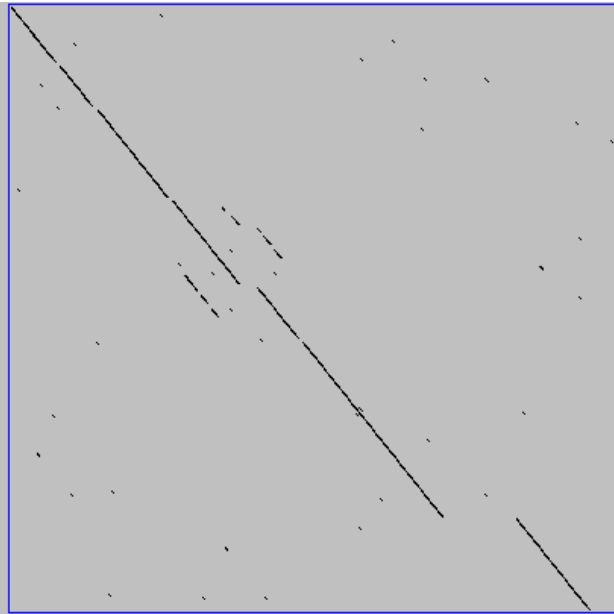
## Window size = 5 bases



A dot plot simply puts a dot where two sequences match. In this example, dots are placed in the plot if 5 bases in a row match perfectly. Requiring a 5 base perfect match is a **heuristic** – only look at regions that have a certain degree of identity.

Do you expect evolutionarily related sequences to have more word matches (matches in a row over a certain length) than random or unrelated sequences?

## Window size = 7 bases



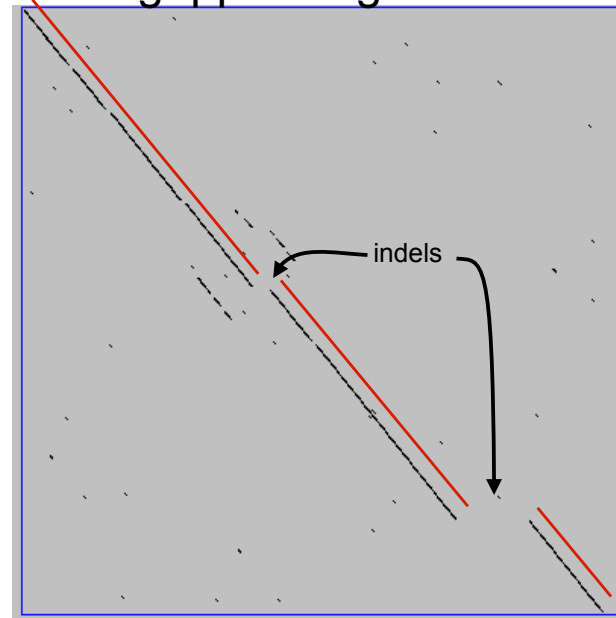
This is a dot plot of the same sequence pair. Now 7 bases in a row must match for a dot to be placed. Noise is reduced.

Using windows of a certain length is very similar to using words (kmers) of N characters in the heuristic alignment search tools

Bigger window (kmer)  
fewer matches to consider

Web site used: <http://www.vivo.colostate.edu/molkit/dnadot/>

## Ungapped alignments



Only **diagonals** can be followed.

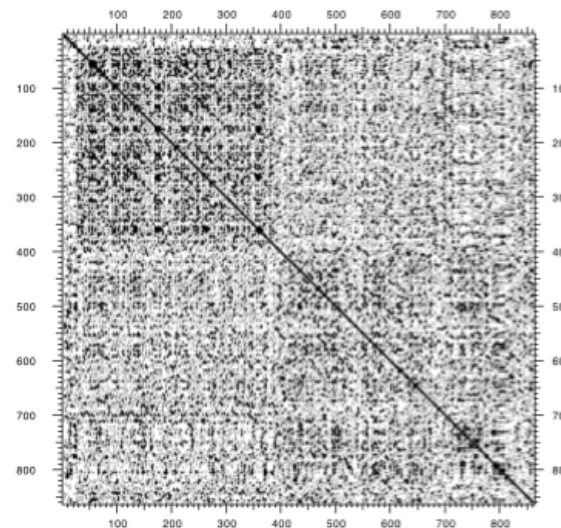
Downward or rightward paths represent **insertion** or **deletions** (gaps in one sequence or the other).

Web site used: <http://www.vivo.colostate.edu/molkit/dnadot/>

## Uses for dot matrices

- Visually assessing the similarity of two protein or two nucleic acid sequences
- Finding local repeat sequences within a larger sequence by comparing a sequence to itself
  - Repeats appear as a set of diagonal runs stacked vertically and/or horizontally

## Repeats

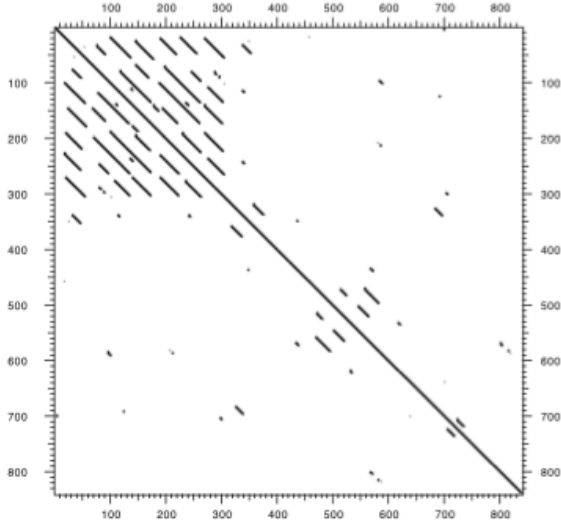


Human LDL receptor  
protein sequence  
(Genbank P01130)

W = 1  
S = 1

(Figure from Mount, "Bioinformatics sequence and genome analysis")

## Repeats



Human LDL receptor  
protein sequence  
(Genbank P01130)

$W = 23$   
 $S = 7$

(Figure from Mount, "Bioinformatics sequence and genome analysis")

# Your Turn!

Exploration of dot plot parameters (hands-on worksheet **Section 1**)

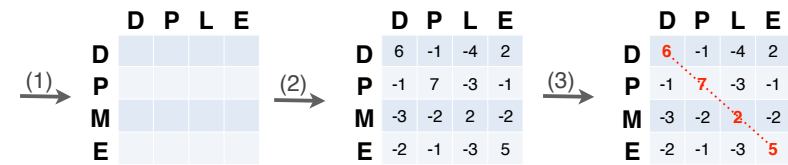
<http://bio3d.ucsd.edu/dotplot/> <https://bioboot.shinyapps.io/dotplot/>

## ALIGNMENT FOUNDATIONS

- **Why...**
  - Why compare biological sequences?
- **What...**
  - Alignment view of sequence changes during evolution (matches, mismatches and gaps)
- **How...**
  - ▶ Dot matrices
  - ▶ Dynamic programming
    - Global alignment
    - Local alignment
  - ▶ BLAST heuristic approach

## The Dynamic Programming Algorithm

- The dynamic programming algorithm can be thought of an extension to the dot plot approach
  - One sequence is placed down the side of a grid and another across the top
  - Instead of placing a dot in the grid, we **compute a score** for each position
  - Finding the optimal alignment corresponds to finding the path through the grid with the **best possible score**



Needleman, S.B. & Wunsch, C.D. (1970) "A general method applicable to the search for similarities in the amino acid sequences of two proteins." J. Mol. Biol. 48:443-453.

## Algorithm of Needleman and Wunsch

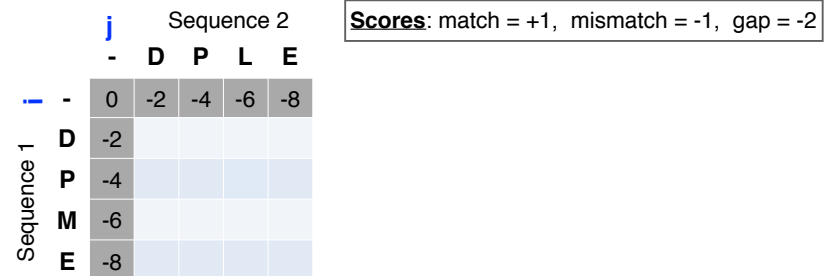
- The Needleman–Wunsch approach to global sequence alignment has three basic steps:
  - (1) setting up a 2D-grid (or **alignment matrix**),
  - (2) **scoring the matrix**, and
  - (3) identifying the **optimal path** through the matrix



Needleman, S.B. & Wunsch, C.D. (1970) "A general method applicable to the search for similarities in the amino acid sequences of two proteins." J. Mol. Biol. 48:443-453.

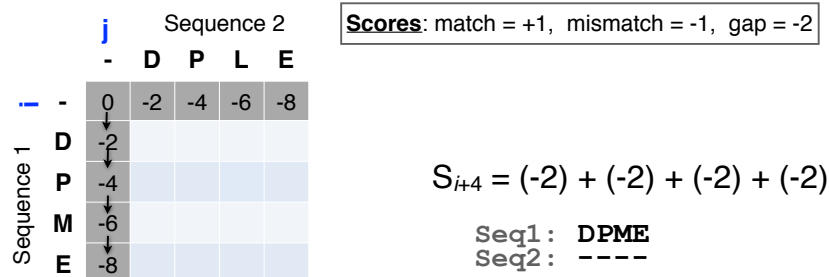
## Scoring the alignment matrix

- Start by filling in the first row and column – these are all indels (gaps).
  - Each step you take you will add the **gap penalty** to the score ( $S_{i,j}$ ) accumulated in the previous cell



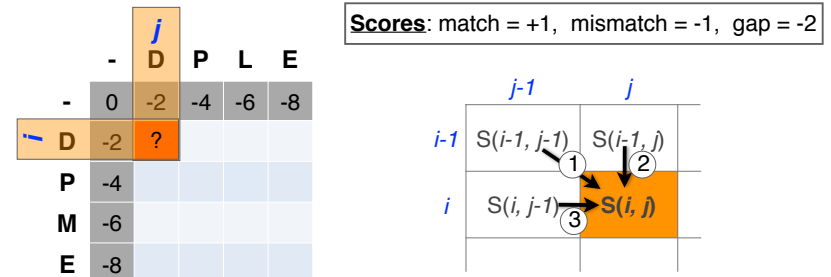
## Scoring the alignment matrix

- Start by filling in the first row and column – these are all indels (gaps).
  - Each step you take you will add the **gap penalty** to the score ( $S_{i,j}$ ) accumulated in the previous cell



## Scoring the alignment matrix

- Then go to the empty corner cell (upper left). It has filled in values in up, left and diagonal directions
  - Now can ask which of the three directions gives the highest score?
  - keep track of this score and direction



## Scoring the alignment matrix

- Then go to the empty corner cell (upper left). It has filled in values in up, left and diagonal directions
  - Now can ask which of the three directions gives the highest score?
  - keep track of this score and direction

			<i>j</i>			
	-	D	P	L	E	
-	0	-2	-4	-6	-8	
<i>i</i>	D	-2	?			
	P	-4				
	M	-6				
	E	-8				

Scores: match = +1, mismatch = -1, gap = -2

$$S(i, j) = \text{Max} \begin{cases} S(i-1, j-1) + (\text{mis})\text{match} & \rightarrow \textcircled{1} \\ S(i-1, j) + \text{gap penalty} & \rightarrow \textcircled{2} \\ S(i, j-1) + \text{gap penalty} & \rightarrow \textcircled{3} \end{cases}$$

## Scoring the alignment matrix

- Then go to the empty corner cell (upper left). It has filled in values in up, left and diagonal directions
  - Now can ask which direction gives the highest score
  - keep track of direction and score

			<i>j</i>			
	-	D	P	L	E	
-	0	-2	-4	-6	-8	
<i>i</i>	D	-2	1			
	P	-4				
	M	-6				
	E	-8				

Scores: match = +1, mismatch = -1, gap = -2

- $\rightarrow \textcircled{1} (0) + (+1) = +1 \leq (D-D) \text{ match!}$
  - $\downarrow \textcircled{2} (-2) + (-2) = -4$
  - $\rightarrow \textcircled{3} (-2) + (-2) = -4$
- Alignment
- D  
D

## Scoring the alignment matrix

- At each step, the score in the current cell is determined by the scores in the neighboring cells
  - The maximal score and the direction that gave that score is stored (we will use these later to determine the optimal alignment)

			<i>j</i>			
	-	D	P	L	E	
-	0	-2	-4	-6	-8	
<i>i</i>	D	-2	1	-1		
	P	-4				
	M	-6				
	E	-8				

Scores: match = +1, mismatch = -1, gap = -2

- $\rightarrow \textcircled{1} (-2) + (-1) = -3 \leq (D-P) \text{ mismatch!}$
  - $\downarrow \textcircled{2} (-4) + (-2) = -6$
  - $\rightarrow \textcircled{3} (1) + (-2) = -1$
- Alignment
- D-  
DP

## Scoring the alignment matrix

- We will continue to store the alignment score ( $S_{i,j}$ ) for all possible alignments in the alignment matrix.

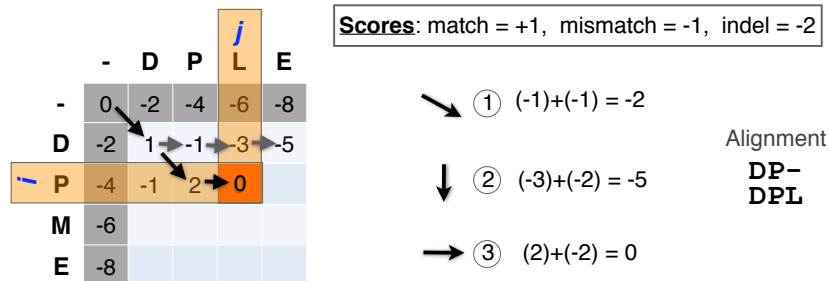
				<i>j</i>		
	-	D	P	L	E	
-	0	-2	-4	-6	-8	
<i>i</i>	D	-2	1	-1	-3	
	P	-4				
	M	-6				
	E	-8				

Scores: match = +1, mismatch = -1, gap = -2

- $\rightarrow \textcircled{1} (-4) + (-1) = -5 \leq (D-L) \text{ mismatch}$
  - $\downarrow \textcircled{2} (-6) + (-2) = -8$
  - $\rightarrow \textcircled{3} (-1) + (-2) = -3$
- Alignment
- D--  
DPL

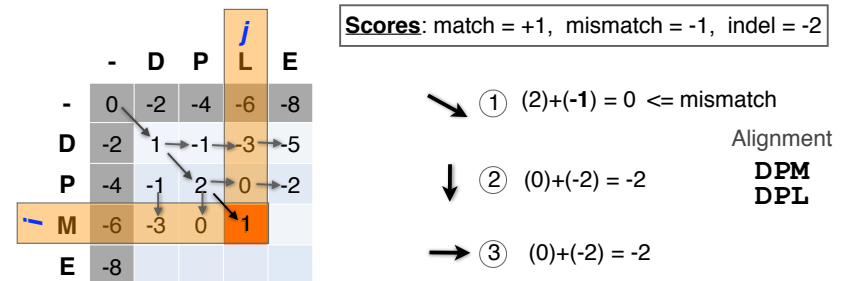
## Scoring the alignment matrix

- For the highlighted cell, the corresponding score ( $S_{i,j}$ ) refers to the score of the optimal alignment of the first  $i$  characters from sequence1, and the first  $j$  characters from sequence2.



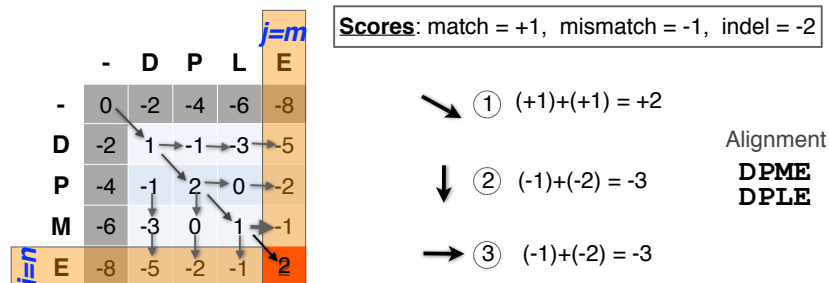
## Scoring the alignment matrix

- At each step, the score in the current cell is determined by the scores in the neighboring cells
  - The maximal score and the direction that gave that score is stored



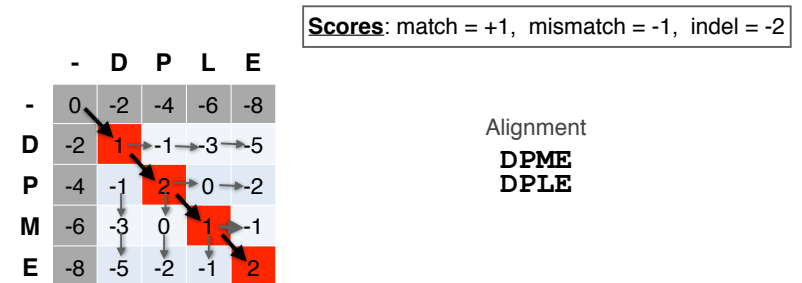
## Scoring the alignment matrix

- The score of the best alignment of the entire sequences corresponds to  $S_{n,m}$ 
  - (where  $n$  and  $m$  are the length of the sequences)



## Scoring the alignment matrix

- To find the best alignment, we retrace the arrows starting from the bottom right cell
  - N.B. The optimal alignment score and alignment are dependent on the chosen scoring system



## Questions:

- What is the optimal score for the alignment of these sequences and how do we find the optimal alignment?

	-	C	A	T	G	T	T	A
-	0	-2	-4	-6	-8	-10	-12	-14
C	-2	1	-1	-3	-5	-7	-9	-11
A	-4	-1	2	0	-2	-4	-6	-8
C	-6	-3	0	1	-1	-3	-5	-7
T	-8	-5	-2	1	0	0	-2	-4
G	-10	-7	-4	-1	2	0	-1	-3
T	-12	-9	-6	-3	0	3	1	-1
A	-14	-11	-8	-5	-2	1	2	2

## Questions:

- What is the optimal score for the alignment of these sequences and how do we find the optimal alignment?

	-	C	A	T	G	T	T	A
-	0	-2	-4	-6	-8	-10	-12	-14
C	-2	1	-1	-3	-5	-7	-9	-11
A	-4	-1	2	0	-2	-4	-6	-8
C	-6	-3	0	1	-1	-3	-5	-7
T	-8	-5	-2	1	0	0	-2	-4
G	-10	-7	-4	-1	2	0	-1	-3
T	-12	-9	-6	-3	0	3	1	-1
A	-14	-11	-8	-5	-2	1	2	2

## Questions:

- To find the best alignment we retrace the arrows starting from the bottom right cell

	-	C	A	T	G	T	T	A
-	0	-2	-4	-6	-8	-10	-12	-14
C	-2	1	-1	-3	-5	-7	-9	-11
A	-4	-1	2	0	-2	-4	-6	-8
C	-6	-3	0	1	-1	-3	-5	-7
T	-8	-5	-2	1	0	0	-2	-4
G	-10	-7	-4	-1	2	0	-1	-3
T	-12	-9	-6	-3	0	3	1	-1
A	-14	-11	-8	-5	-2	1	2	2

## More than one alignment possible

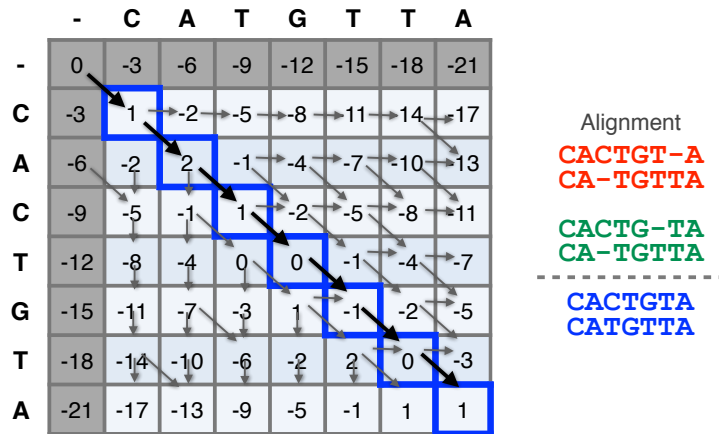
- Sometimes more than one alignment can result in the same optimal score

	-	C	A	T	G	T	T	A
-	0	-2	-4	-6	-8	-10	-12	-14
C	-2	1	-1	-3	-5	-7	-9	-11
A	-4	-1	2	0	-2	-4	-6	-8
C	-6	-3	0	1	-1	-3	-5	-7
T	-8	-5	-2	1	0	0	-2	-4
G	-10	-7	-4	-1	2	0	-1	-3
T	-12	-9	-6	-3	0	3	1	-1
A	-14	-11	-8	-5	-2	1	2	2

Alignment  
**CACTGT-A**  
**CA-TGTTA**  
  
**CACTG-TA**  
**CA-TGTTA**

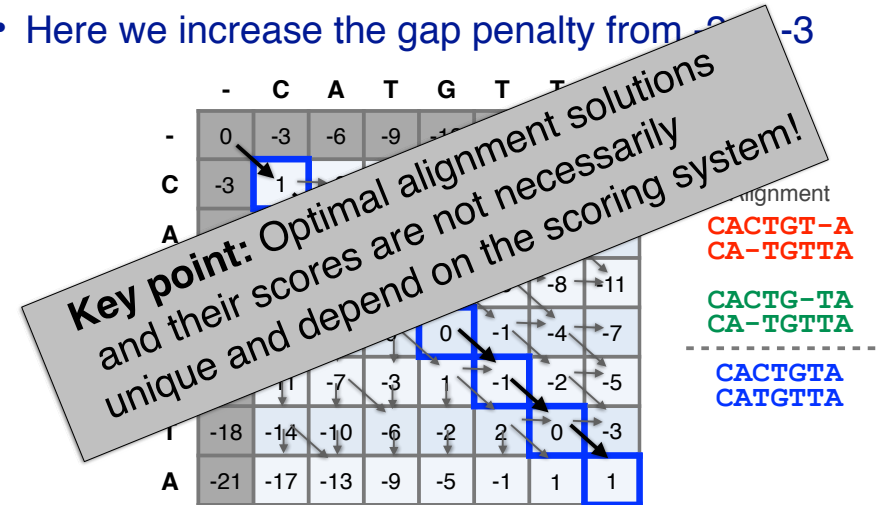
The alignment and score are dependent on the scoring system

- Here we increase the gap penalty from -2 to -3



The alignment and score are dependent on the scoring system

- Here we increase the gap penalty from -2 to -3



**Key point:** Optimal alignment solutions and their scores are not necessarily unique and depend on the scoring system!

# Your Turn!

Hands-on worksheet **Sections 2 & 3**

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

		A	G	T	T	C
	0					
A						
T						
T						
G						
C						

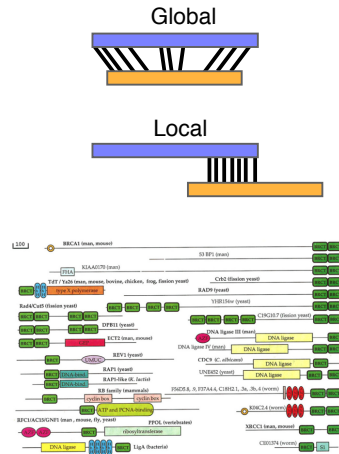
## ALIGNMENT FOUNDATIONS

- Why...
  - Why compare biological sequences?
- What...
  - Alignment view of sequence changes during evolution (matches, mismatches and gaps)
- How...
  - Dot matrices
  - Dynamic programming
    - Global alignment
    - Local alignment
  - BLAST heuristic approach



# Global vs local alignments

- Needleman-Wunsch is a **global alignment** algorithm
  - Resulting alignment spans the complete sequences end to end
  - This is appropriate for closely related sequences that are similar in length
- For many practical applications we require **local alignments**
  - Local alignments highlight sub-regions (e.g. protein domains) in the two sequences that align well



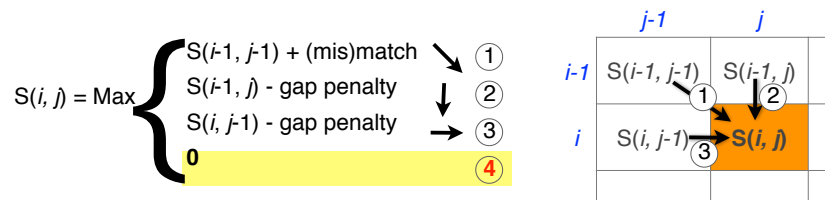
# Local alignment: Definition

- Smith & Waterman proposed simply that a local alignment of two sequences allow arbitrary-length segments of each sequence to be aligned, with no penalty for the unaligned portions of the sequences. Otherwise, the score for a local alignment is calculated the same way as that for a global alignment

Smith, T.F. & Waterman, M.S. (1981) "Identification of common molecular subsequences." J. Mol. Biol. 147:195-197.

# The Smith-Waterman algorithm

- Three main modifications to Needleman-Wunsch:
  - Allow a node to start at 0
  - The score for a particular cell cannot be negative
    - if all other score options produce a negative value, then a zero must be inserted in the cell
  - Record the highest- scoring node, and trace back from there

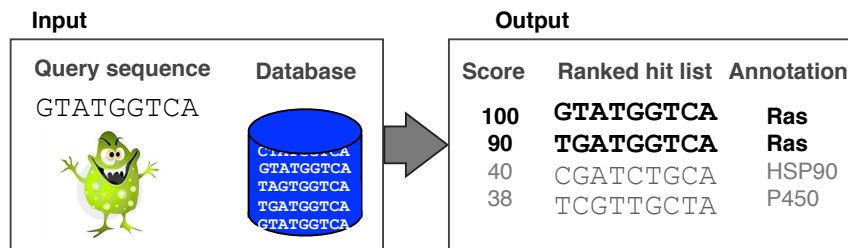


		Sequence 1														
		-	C	A	G	C	C	U	C	G	C	U	U	A	G	
Sequence 2	-	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
	A	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
	U	0.0	0.0	1.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.7
	U	0.0	0.0	0.0	0.7	0.3	0.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.7
	G	0.0	0.0	0.0	1.0	0.3	0.0	0.0	0.7	1.0	0.0	0.0	0.7	0.7	1.0	1.0
	C	0.0	1.0	0.0	0.0	2.0	1.3	0.3	1.0	0.3	2.0	0.7	0.3	0.3	0.3	0.3
	C	0.0	1.0	0.7	0.0	1.0	3.0	1.7	1.3	1.0	1.3	1.7	0.3	0.0	0.0	0.0
	A	0.0	0.0	2.0	0.7	0.3	1.7	2.7	1.3	1.0	0.7	1.0	1.3	1.3	0.0	0.0
	U	0.0	0.0	0.7	1.7	0.3	1.3	2.7	2.3	1.0	0.7	1.7	2.0	1.0	1.0	1.0
	U	0.0	0.0	0.3	0.3	1.3	1.0	2.3	2.3	2.0	0.7	1.7	2.7	1.7	1.0	1.0
G	0.0	0.0	0.0	1.3	0.0	1.0	1.0	2.0	3.3	2.0	1.7	1.3	2.3	2.7	2.7	
A	0.0	0.0	1.0	0.0	1.0	0.3	0.7	0.7	2.0	3.0	1.7	1.3	2.3	2.0	2.0	
C	0.0	1.0	0.0	0.7	1.0	2.0	0.7	1.7	1.7	3.0	2.7	1.3	1.0	2.0	2.0	
G	0.0	0.0	0.7	1.0	0.3	0.7	1.7	0.3	2.7	1.7	2.7	2.3	1.0	2.0	2.0	
G	0.0	0.0	0.0	1.7	0.7	0.3	0.3	1.3	1.3	2.3	1.3	2.3	2.0	2.0	2.0	

Local alignment  
**GCC-AUG**  
**GCCUCG**

## Local alignments can be used for database searching

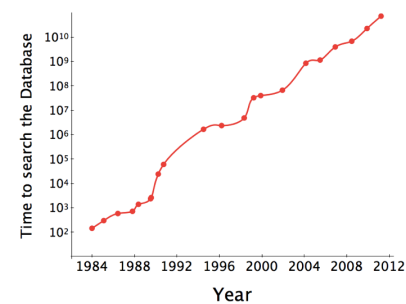
- **Goal:** Given a query sequence (Q) and a sequence database (D), find a list of sequences from D that are most similar to Q
  - **Input:** Q, D and scoring scheme
  - **Output:** Ranked list of hits



69

## The database search problem

- Due to the rapid growth of sequence databases, search algorithms have to be both efficient and sensitive
  - Time to search with SW is proportional to  $m \times n$  ( $m$  is length of query,  $n$  is length of database), **too slow for large databases!**

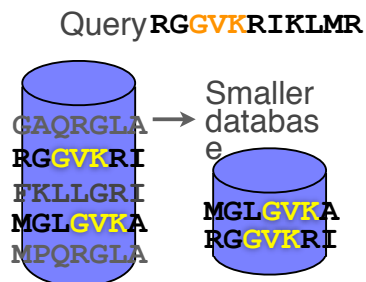


To reduce search time **heuristic algorithms**, such as BLAST, first remove database sequences without a strong local similarity to the query sequence in a quick initial scan.

70

## The database search problem

- Due to the rapid growth of sequence databases, search algorithms have to be both efficient and sensitive
  - Time to search with SW is proportional to  $m \times n$  ( $m$  is length of query,  $n$  is length of database), **too slow for large databases!**



To reduce search time **heuristic algorithms**, such as BLAST, first remove database sequences without a strong local similarity to the query sequence in a quick initial scan.

71

## ALIGNMENT FOUNDATIONS

- **Why...**
  - Why compare biological sequences?
- **What...**
  - Alignment view of sequence changes during evolution (matches, mismatches and gaps)
- **How...**
  - ▶ Dot matrices
  - ▶ Dynamic programming
    - Global alignment
    - Local alignment
  - ▶ **BLAST heuristic approach**

## Rapid, heuristic versions of Smith–Waterman: **BLAST**

- BLAST (Basic Local Alignment Search Tool) is a simplified form of Smith-Waterman (SW) alignment that is popular because it is **fast** and **easily accessible**
  - BLAST is a heuristic approximation to SW - It examines only part of the search space
  - BLAST saves time by restricting the search by scanning database sequences for likely matches before performing more rigorous alignments
  - Sacrifices some sensitivity in exchange for speed
  - In contrast to SW, BLAST is not guaranteed to find optimal alignments

73

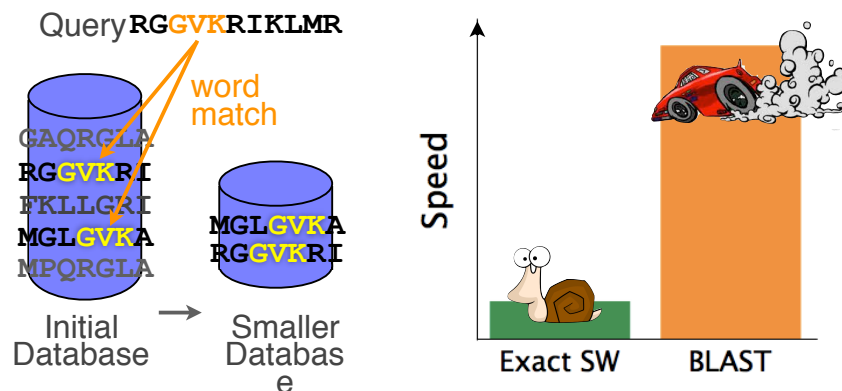
## Rapid, heuristic versions of Smith–Waterman: **BLAST**

- BLAST (Basic Local Alignment Search Tool) is a simplified form of Smith-Waterman (SW) alignment that is popular because it is **fast** and **easily accessible**
  - BLAST finds regions of local similarity between sequences
  - BLAST saves time by restricting the search by scanning database sequences for likely matches before performing more rigorous alignments
  - Sacrifices some sensitivity in exchange for speed
  - In contrast to SW, BLAST is not guaranteed to find optimal alignments

“The central idea of the BLAST algorithm is to confine attention to sequence pairs that contain an initial **word pair match**”  
Altschul et al. (1990)

74

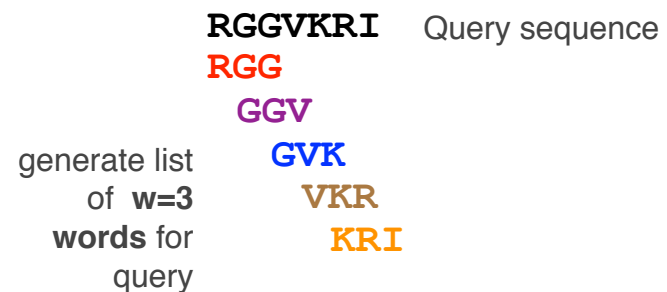
- BLAST uses this pre-screening heuristic approximation resulting in an approach that is about 50 times faster than the Smith-Waterman



75

## How BLAST works

- Four basic phases
  - **Phase 1:** compile a list of query word pairs ( $w=3$ )



76

## Blast

- **Phase 2:** expand word pairs to include those similar to query (defined as those above a similarity threshold to original word, i.e. match scores in substitution matrix)

**RGGVKRI** Query sequence  
**RGG RAG RIG RLG ...**  
**GGV GAV GTV GCV ...**  
**GVK GAK GIK GGK ...**  
**VKR VRR VHR VER ...**  
**KRI KKI KHI KDI ...**

extend list of words similar to query

77

## Blast

- **Phase 3:** a database is scanned to find sequence entries that match the compiled word list

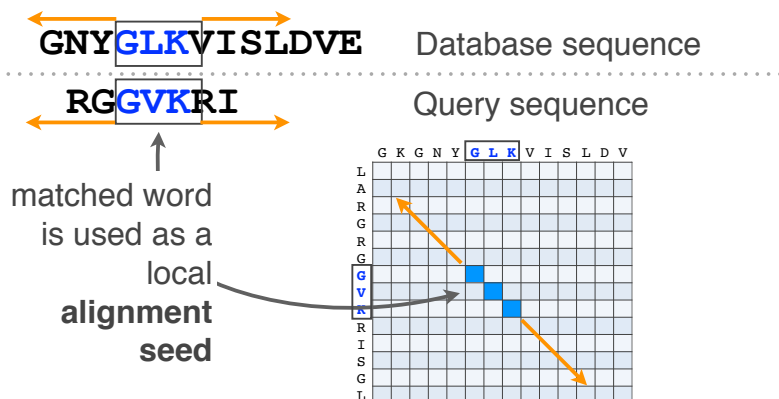
**GNYGLKVISLDVE** Database sequence  
**RGGVKRI** Query sequence  
**RGG RAG RIG RLG ...**  
**GGV GAV GTV GCV ...**  
**GVK GLK GIK GGK ...**  
**VKR VRR VHR VER ...**  
**KRI KKI KHI KDI ...**

search for perfect matches in the database sequence

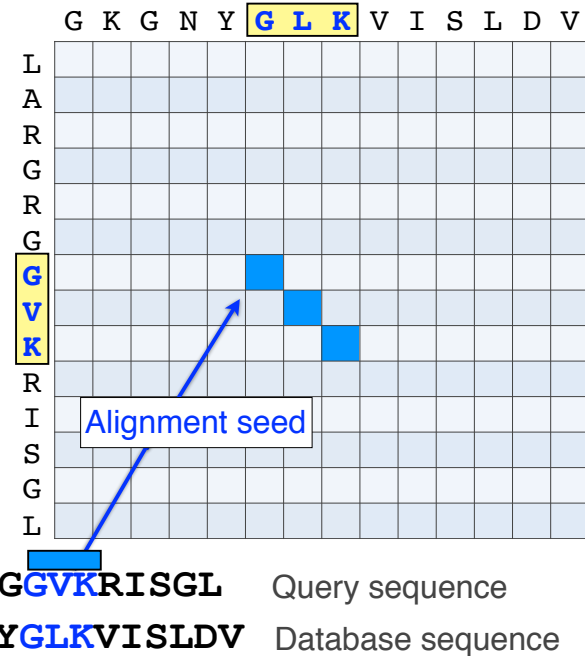
78

## Blast

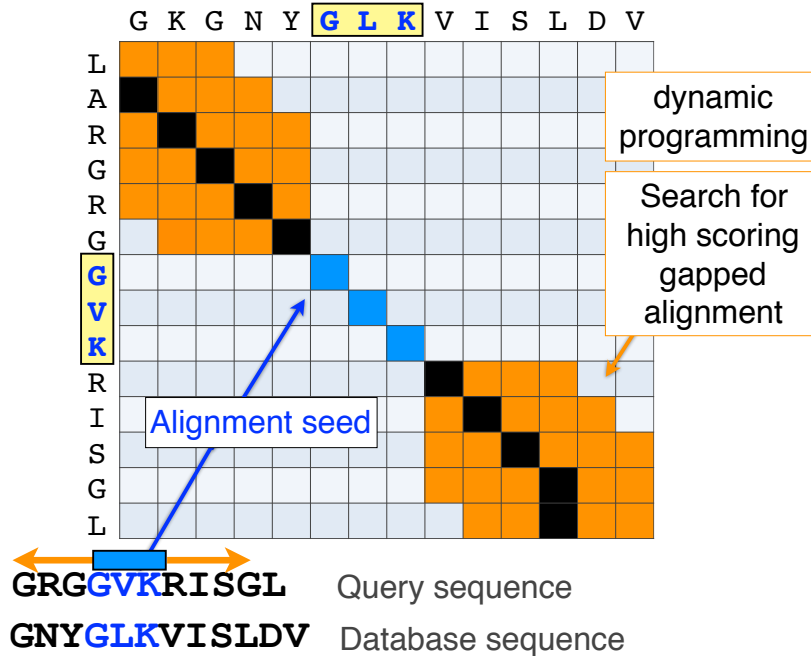
- **Phase 4:** the initial database hits are extended in both directions using dynamic programming



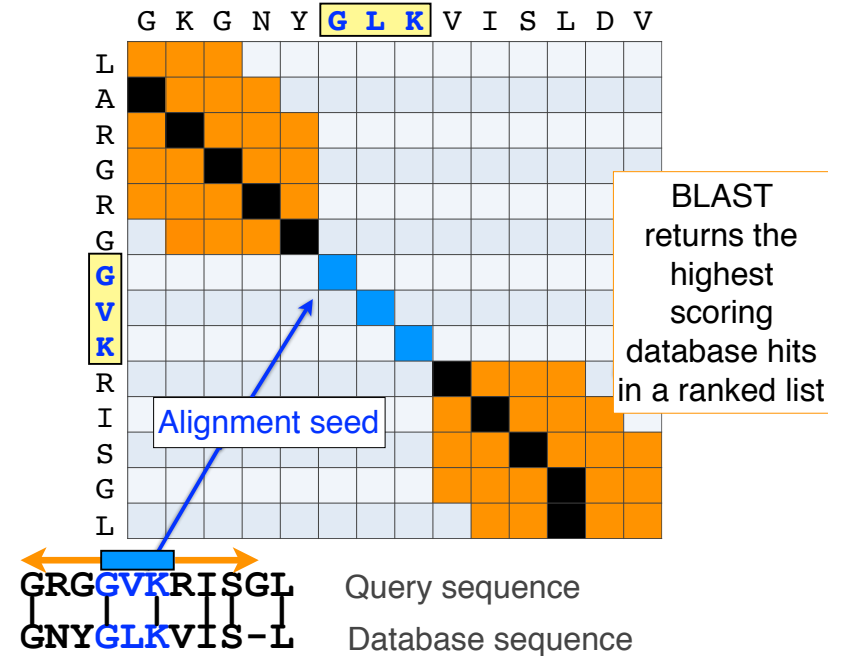
79



80



81



82

## BLAST output

- BLAST returns the highest scoring database hits in a ranked list along with details about the target sequence and alignment statistics

Description	Max score	Query cover	E value	Max ident	Accession
kinesin-1 heavy chain [Homo sapiens]	677	100%	0	100%	NP_004512.1
Kif5b protein [Mus musculus]	676	100%	0	98%	AAA20133.1
Kinesin-14 heavy chain [Danio rerio]	595	88%	0	78%	XP_00320703
hypothetical protein EGK_18589	48.2	40%	0.03	32%	ELK35081.1
mKIAA4102 protein [Mus musculus]	42.7	38%	3.02	24%	EHH28205.1

83

## Statistical significance of results

- An important feature of BLAST is the computation of statistical significance for each hit. This is described by the **E value** (expect value)

Description	Max score	Query cover	E value	Max ident	Accession
kinesin-1 heavy chain [Homo sapiens]	677	100%	0	100%	NP_004512.1
Kif5b protein [Mus musculus]	676	100%	0	98%	AAA20133.1
Kinesin-14 heavy chain [Danio rerio]	595	88%	0	78%	XP_00320703
hypothetical protein EGK_18589	48.2	40%	0.03	32%	ELK35081.1
mKIAA4102 protein [Mus musculus]	42.7	38%	3.02	24%	EHH28205.1

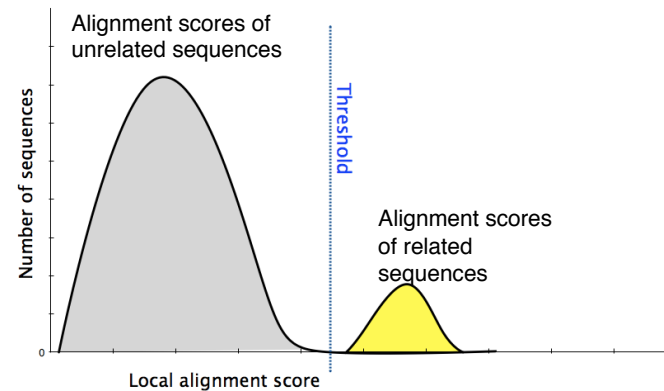
84

## BLAST scores and E-values

- The **E value** is the **expected** number of hits that are as good or better than the observed local alignment score (with this score or better) if the query and database are **random** with respect to each other
  - *i.e.* the number of alignments expected to occur by chance with equivalent or better scores
- Typically, only hits with E value **below** a significance threshold are reported
  - This is equivalent to selecting alignments with score above a certain score threshold

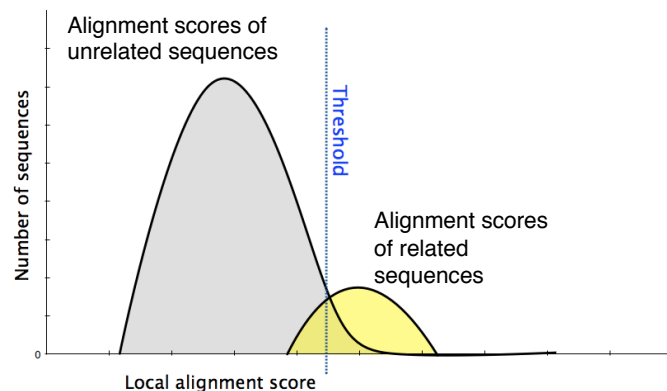
85

- Ideally, a threshold separates all query related sequences (yellow) from all unrelated sequences (gray)



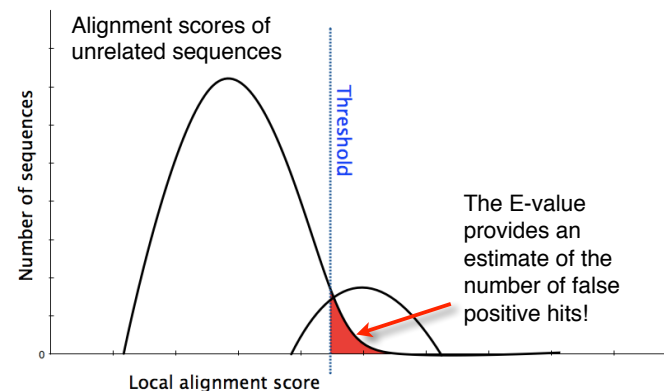
86

- Unfortunately, often both score distributions overlap
  - The E value describes the expected number of hits with a score above the threshold if the query and database are unrelated



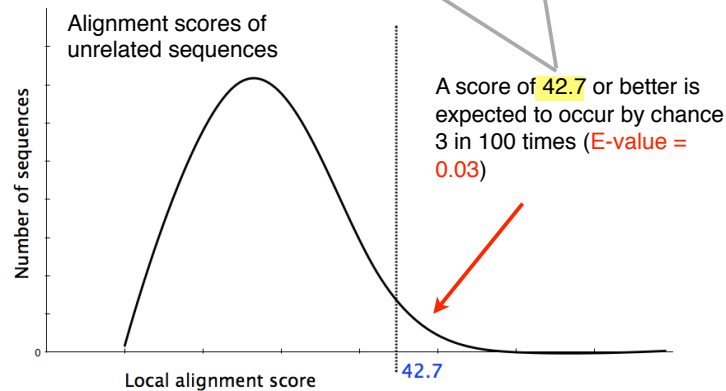
87

- Unfortunately, often both score distributions overlap
  - The E value describes the expected number of hits with a score above the threshold if the query and database are unrelated



88

Description	Max score	Query cover	E value	Max ident	Accession
kinesin-1 heavy chain [Homo sapiens]	677	100%	0	100%	NP_004512.1
Kif5b protein [Mus musculus]	676	100%	0	98%	AAA20133.1
Kinesin-14 heavy chain [Danio rerio]	595	88%	0	78%	XP_00320703
hypothetical protein EGK_18589	42.7	40%	0.03	32%	ELK35081.1



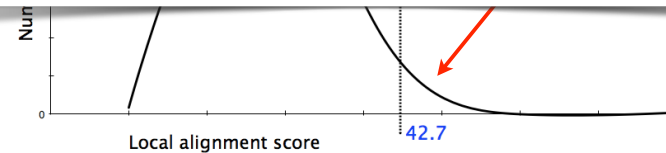
89

Description	Max score	Total score	Query cover	E value	Max ident	Accession
kinesin-1 heavy chain [Homo sapiens]	677	677	100%	0	100%	NP_004512.1
Kif5b protein [Mus musculus]	676	676	100%	0	98%	AAA20133.1

In general  $E$  values  $< 0.005$  are usually significant.

To find out more about  $E$  values see: "*The Statistics of Sequence Similarity Scores*" available in the help section of the NCBI BLAST site:

<http://www.ncbi.nlm.nih.gov/blast/tutorial/Altschul-1.html>



90

## Your Turn!

Hands-on worksheet **Sections 4 & 5**

- ▶ Please do answer the last lab review question (Q19).
- ▶ We encourage discussion and exploration!

## Practical database searching with BLAST

NCBI BLAST Home Page

<http://blast.ncbi.nlm.nih.gov/Blast.cgi>

92

## Practical database searching with BLAST

- There are four basic components to a traditional BLAST search
  - (1) Choose the sequence (query)
  - (2) Select the BLAST program
  - (3) Choose the database to search
  - (4) Choose optional parameters
- Then click “BLAST”

93

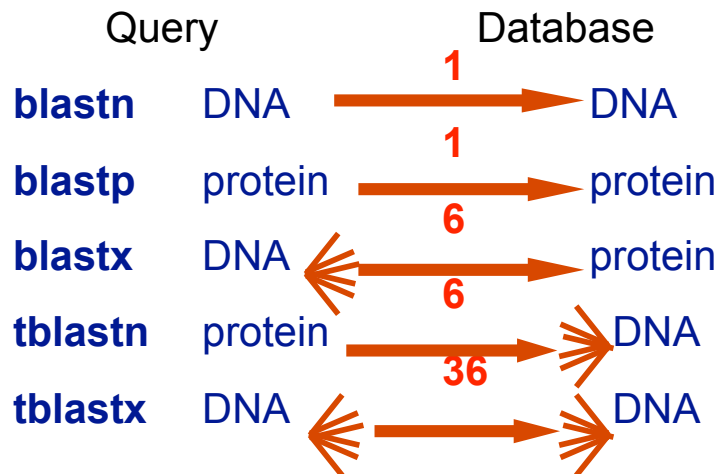
## Step 1: Choose your sequence

- Sequence can be input in FASTA format or as accession number

NCBI Protein search results for 'hemoglobin subunit beta [Homo sapiens]' (NCBI Reference Sequence: NP\_000509.1). The sequence is: >gi|4504349|ref|NP\_000509.1| hemoglobin subunit beta [Homo sapiens] MVHLTPEEKSAVTALWGRVNVDEVGGEALGRLLVVPHTQRFESFODLSTPDAVMGNPVRKAHGKRVLG AFSDGLAHLNLIKGTFA TLSELHCDKLRVDPENFRLLGNVLCVLAHHPGKEFPPVQAAAYQKVVAGVAN ALAHKYH

94

## Step 2: Choose the BLAST program



95

## DNA potentially encodes six proteins

```

5' CAT CAA
5' ATC AAC
5' TCA ACT

5' CATCAACTACAAC TCCAAAGACACCCTTACACATCAACAAACCTACCCAC 3'
3' GTAGTTGATGTTGAGGTTTCTGTGGGAATGTGTAGTTGTTTGGATGGGTG 5'

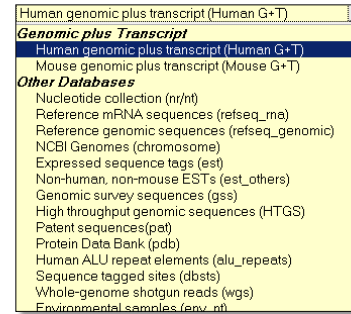
5' GTG GGT
5' TGG GTA
5' GGG TAG
    
```

96

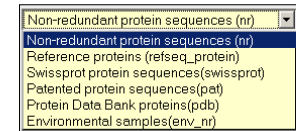


### Step 3: Choose the database

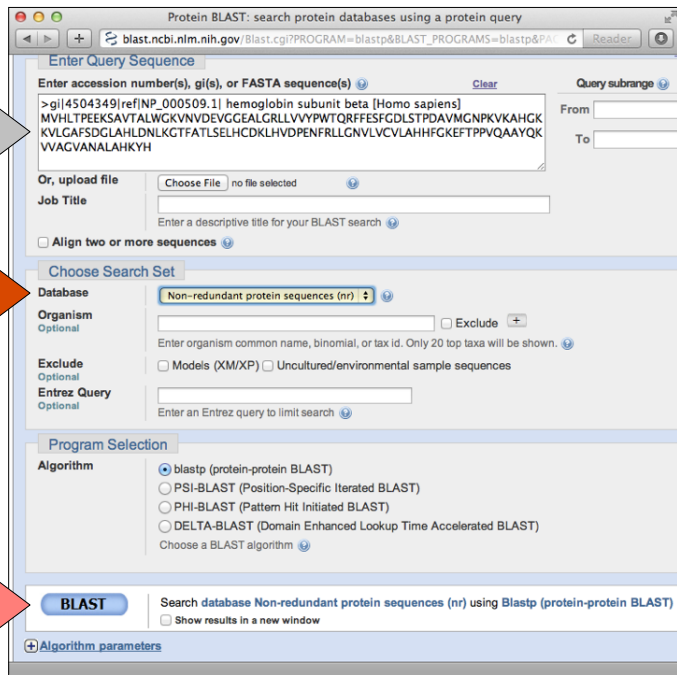
- nr = non-redundant (most general database)
- dbest = database of expressed sequence tags
- dbsts = database of sequence tag sites
- gss = genomic survey sequences



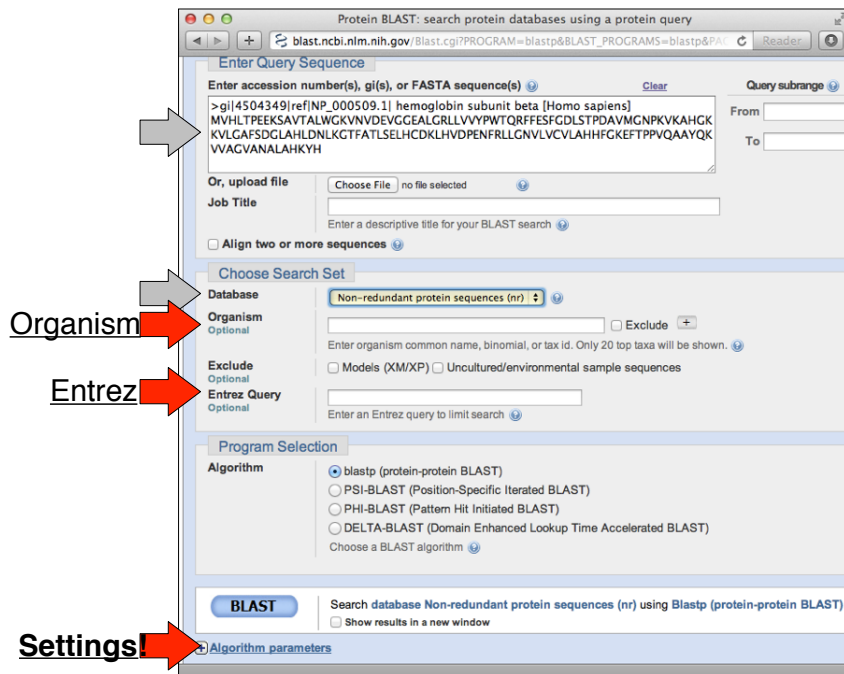
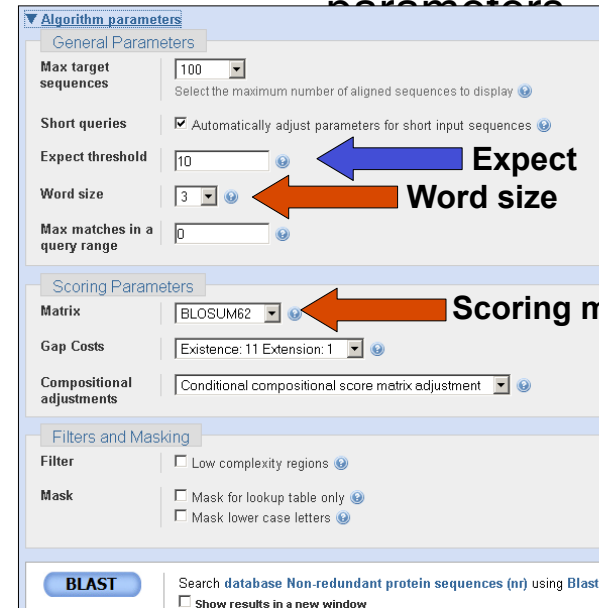
nucleotide databases



protein databases



### Step 4a: Select optional search parameters



## Step 4: Optional parameters

- You can...
  - choose the organism to search
  - change the substitution matrix
  - change the expect (E) value
  - change the word size
  - change the output format

101

## Results page

NCBI Blast:gi|4504349|ref|NP\_000509.1| hemoglobin

BLAST® Basic Local Alignment Search Tool

Home Recent Results Saved Strategies Help

NCBI/BLAST/blastp suite/ Formatting Results - FVGUTMR2013

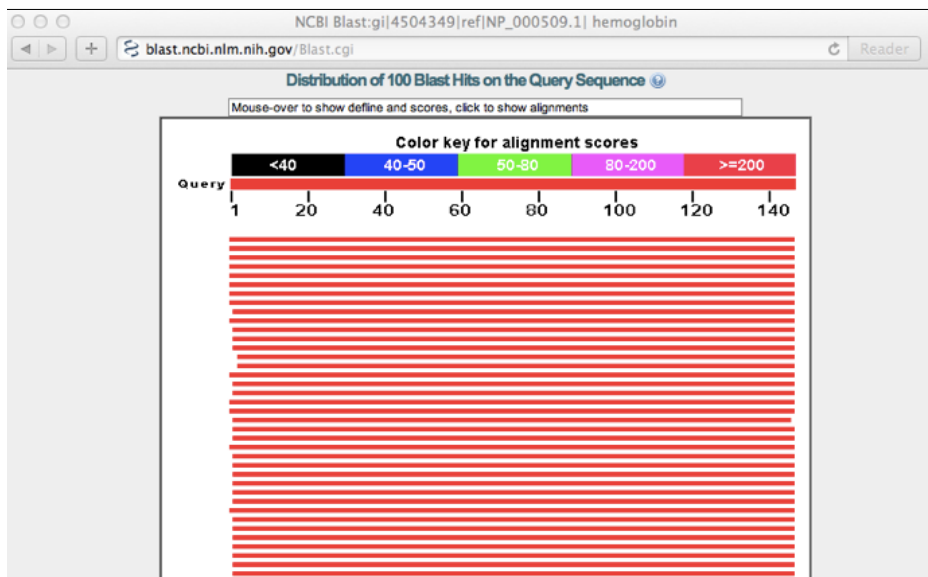
g|4504349|ref|NP\_000509.1| hemoglobin

Query ID: gi|4504349|ref|NP\_000509.1| hemoglobin subunit beta [Homo sapiens]  
 Description: beta [Homo sapiens]  
 Molecule type: amino acid  
 Query Length: 147

Database Name: nr  
 Description: All non-redundant GenBank CDS translations+PDB+SwissProt+PIR+PRF excluding environmental samples from WGS projects  
 Program: BLASTP 2.2.27+

Graphic Summary: Putative conserved domains have been detected, click on the image below for detailed results. Query seq. Specific hits: globin Superfamilies: globin\_like superfamily

## Further down the results page...



## Further down the results page...

NCBI Blast:gi|4504349|ref|NP\_000509.1| hemoglobin

Sequences producing significant alignments:

Select: All None Selected:0

Alignments

Description	Max score	Total score	Query cover	E value	Max ident	Accession
hemoglobin beta [synthetic construct]	301	301	100%	9e-103	100%	AAX37051.1
hemoglobin beta [synthetic construct]	301	301	100%	1e-102	100%	AAX29557.1
hemoglobin subunit beta [Homo sapiens] >ref XP_508242.1  PREDICTED: hemoglobin s	301	301	100%	1e-102	100%	NP_000509.1
RecName: Full=Hemoglobin subunit beta; AltName: Full=Beta-globin; AllName: Full=He	300	300	100%	4e-102	99%	P02024.2
beta globin chain variant [Homo sapiens]	299	299	100%	5e-102	99%	AAN84548.1
beta globin [Homo sapiens] >qbj AAZ39781.1  beta globin [Homo sapiens] >qbj AAZ3978	299	299	100%	5e-102	99%	AAZ39780.1
beta-globin [Homo sapiens]	299	299	100%	5e-102	99%	ACU56984.1
hemoglobin beta chain [Homo sapiens]	299	299	100%	6e-102	99%	AAD19696.1
Chain B. Structure Of Haemoglobin In The Deoxy Quaternary State With Ligand Bound A	298	298	99%	9e-102	100%	1CQH_B
hemoglobin beta subunit variant [Homo sapiens] >qbj AAA88054.1  beta-globin [Homo sa	298	298	100%	1e-101	99%	AAF00489.1
Chain B. Human Hemoglobin D Los Angeles: Crystal Structure >pdb 2YRS D.Chain D. H	298	298	99%	2e-101	99%	2YRS_B
Chain B. High-Resolution X-Ray Study Of Deoxy Recombinant Human Hemoglobins Syn	297	297	99%	3e-101	99%	1DXU_B
Chain B. Analysis Of The Crystal Structure. Molecular Modeling And Infrared Soectroscoc	297	297	99%	3e-101	99%	1HDB_B

## Further down the results page...

NCBI Blast:gil4504349[ref|NP\_000509.1| hemoglobin

blast.ncbi.nlm.nih.gov/Blast.cgi

hemoglobin subunit beta [Homo sapiens]  
Sequence ID: [ref|NP\\_000509.1|](#) Length: 147 Number of Matches: 1  
See 84 more title(s)

Range 1: 1 to 147	Expect	Method	Identities	Positives	Gaps
301 bits(770)	1e-102	Compositional matrix adjust.	147/147(100%)	147/147(100%)	0/147(0%)

Query 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60  
MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60  
Sbjct 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

Query 61 VKAHGKVLGAFSDGLAHLNLRKGTFTATLSELHCDKLVDPENFRLLGNLVCLLAHFHFG 120  
VKAHKKVLAGFSDGLAHLNLRKGTFTATLSELHCDKLVDPENFRLLGNLVCLLAHFHFG 120  
Sbjct 61 VKAHGKVLGAFSDGLAHLNLRKGTFTATLSELHCDKLVDPENFRLLGNLVCLLAHFHFG 120

Query 121 KEFTPPVQAAYQKVVAGVANALAHKYH 147  
KEFTPPVQAAYQKVVAGVANALAHKYH 147  
Sbjct 121 KEFTPPVQAAYQKVVAGVANALAHKYH 147

RecName: Full=Hemoglobin subunit beta; AltName: Full=Beta-globin; AltName: Full=Hemoglobin beta chain  
Sequence ID: [spi|P02024.2|HBB\\_GORGO](#) Length: 147 Number of Matches: 1

Range 1: 1 to 147	Expect	Method	Identities	Positives	Gaps
300 bits(767)	4e-102	Compositional matrix adjust.	146/147(99%)	147/147(100%)	0/147(0%)

## Different output formats are available

NCBI Blast:gil4504349[ref|NP\_000509.1| hemoglobin

blast.ncbi.nlm.nih.gov/Blast.cgi

BLAST® Basic Local Alignment Search Tool

Home Recent Results Saved Strategies Help

NCBI BLAST/ blastp suite/ Formatting Results - FVGUTMZR2015

Edit and Resubmit Save Search Strategies **Formatting options** Download

Change the result display ba You Tube Learn about the enhanced report B

Formatting options

Show Alignment as HTML Old View Reset form to defaults

Alignment View Query-anchored with letters for identities

Display  Graphical Overview  Sequence Retrieval  NCBI-gi

Masking Character: Lower Case Color: Grey

Limit results Descriptions: 50 Graphical overview: 50 Alignments: 50

Organism Type common name, binomial, taxid, or group name. Only 20 top taxa will be shown.  
Enter organism name or id-completions will be suggested Exclude

Entrez query:

Expect Min:  Expect Max:

Percent Identity Min:  Percent Identity Max:

Format for  PSI-BLAST with inclusion threshold:

gil4504349[ref|NP\_000509.1| hemoglobin

## E.g. Query anchored alignments

NCBI Blast:gil4504349[ref|NP\_000509.1| hemoglobin

blast.ncbi.nlm.nih.gov/Blast.cgi

Query 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

AAK37051 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

AAK29557 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

NP\_000509 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

P02024 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

AAK84548 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

AAZ39780 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

ACU56984 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

AAD19696 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

ICOH\_B 1 VHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

AAF00489 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

2YRS\_B 1 VHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1DXU\_B 1 MHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1HDB\_B 1 VHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1DXV\_B 2 HLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

3KMF\_C 2 HLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

AAI68978 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

1NOP\_B 1 VHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1K1K\_B 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

AAK11320 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

XP\_002822173 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

1Y85\_B 1 VHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1YE0\_B 1 MHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1O10\_B 1 MHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

CAA23759 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

1YE2\_B 1 MHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1Y5F\_B 1 MHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1A00\_B 1 MHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1H8S\_B 1 VHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1A8Y\_B 1 MHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

1Y8V\_B 1 VHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 59

## ... and alignments with dots for identities

NCBI Blast:gil4504349[ref|NP\_000509.1| hemoglobin

blast.ncbi.nlm.nih.gov/Blast.cgi

Query 1 MVHLTPEEKSAVTALWGKLVNDEVGGEALGRLLVVPWTRQRFESFGDLSTPDAVGNPK 60

AAK37051 1 ..... 60

AAK29557 1 ..... 60

NP\_000509 1 ..... 60

P02024 1 ..... 60

AAK84548 1 ..... 60

AAZ39780 1 .....K..... 60

ACU56984 1 .....K..... 60

AAD19696 1 .....L..... 60

ICOH\_B 1 ..... 59

AAF00489 1 ..... 60

2YRS\_B 1 ..... 59

1DXU\_B 1 M..... 59

1HDB\_B 1 ..... 59

1DXV\_B 2 ..... 59

3KMF\_C 2 ..... 59

AAI68978 1 .....K..... 60

1NOP\_B 1 ..... 59

1K1K\_B 1 .....V..... 59

AAK11320 1 .....X..... 60

XP\_002822173 1 ..... 60

1Y85\_B 1 ..... 59

1YE0\_B 1 M..... 59

1O10\_B 1 M.....A..... 59

CAA23759 1 .....V.....X..... 60

1YE2\_B 1 M.....F..... 59

1Y5F\_B 1 M..... 59

1A00\_B 1 M.....Y..... 59

## Common problems

- Selecting the wrong version of BLAST
- Selecting the wrong database
- Too many hits returned
- Too few hits returned
- Unclear about the significance of a particular result - are these sequences homologous?

109

## How to handle too many results

- Focus on the question you are trying to answer
  - select “refseq” database to eliminate redundant matches from “nr”
  - Limit hits by organism
  - Use just a portion of the query sequence, when appropriate
  - Adjust the expect value; lowering  $E$  will reduce the number of matches returned

110

## How to handle too few results

- Many genes and proteins have no significant database matches
  - remove Entrez limits
  - raise E-value threshold
  - search different databases
  - try scoring matrices with lower BLOSUM values (or higher PAM values)
  - use a search algorithm that is more sensitive than BLAST (*e.g.* PSI-BLAST or HMMer)

111

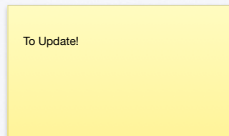
## Summary of key points

- Sequence alignment is a fundamental operation underlying much of bioinformatics.
- Even when optimal solutions can be obtained they are not necessarily unique or reflective of the biologically correct alignment.
- Dynamic programming is a classic approach for solving the pairwise alignment problem.
- Global and local alignment, and their major application areas.
- Heuristic approaches are necessary for large database searches and many genomic applications.

## FOR NEXT CLASS...

Check out the online:

- ✓ **Reading:** Sean Eddy's "What is dynamic programming?"
- ✓ **Homework:** (1) **Quiz**, (2) **Alignment Exercise**.



## Homework Grading

Both (1) quiz questions and (2) alignment exercise carry equal weights (*i.e.* 50% each).

(Homework 2) Assessment Criteria	Points	
Setup labeled <b>alignment matrix</b>	1	
Include initial column and row for <b>GAPs</b>	1	
All alignment matrix elements <b>scored</b> ( <i>i.e.</i> filled in)	1	
Evidence for correct use of <b>scoring scheme</b>	1	
<b>Direction arrows</b> drawn between all cells	1	
Evidence of multiple arrows to a given cell if appropriate	1	D
Correct <b>optimal score</b> position in matrix used	1	C
Correct optimal score obtained for given scoring scheme	1	B
<b>Traceback path(s)</b> clearly highlighted	1	A
Correct <b>alignment(s)</b> yielding optimal score listed	1	A+