

CS612 - Algorithms in Bioinformatics

Sequence Alignment

February 12, 2019

Searching for Sequence Similarity

- **Problem:** Determine possible biological function associated with a decoded gene sequence
- **Approach/Process:**
 - Treat given gene sequence as a query sequence
 - Search over a database of functionally-annotated gene sequences
 - Gene sequences for which the function is determined and deposited
 - If the query sequence x is similar to a sequence y in the database
 - Then we add $\text{function}(y)$ to the list of possible functions of x
- **Assumption:** similar sequences have similar functions
 - In other words, sequence is the main determinant of function

Searching for Sequence Similarity

- **Problem:** Determine possible biological function associated with a decoded gene sequence
- **Subproblems** (of general interest to computer scientists):
 - How do we measure sequence similarity?
 - How do we align two sequences? Do they have to match exactly or as long as they overlap significantly, we can make the same prediction?
 - Over what threshold of similarity does the assumption hold?
 - Can we associate a confidence as a function of similarity?
 - What if we want to compare more than two sequences?

Database Search and Sequence Alignment

```
AAB24882      TYHMCQFHCRIYVNNHSGEKLYECNEFSKAFSCPSHLQCHKRRQIGEKTHEHNQCGKAFPT 60
AAB24881      -----YECNQCGKAFAQHSSLKCHYRTHIGEKPYECNQCGKAFSK 40
                ****: ,***: * *:*** * :*****,:* *****,.

AAB24882      PSHLQYHERTHIGEKPYECHQCGQAFKKCSLLQRHKRTHIGEKPYE-CNQCGKAFAQ- 116
AAB24881      HSHLQCHKRTHIGEKPYECNQCGKAFSQHGLLQRHKRTHIGEKPYMNVINMVKPLHNS 98
                **** *:*****:***:***: ,*****:*****: *.::
```

Why do We Want to Compare Sequences?

- Evolutionary relationships
 - Phylogenetic trees can be constructed based on comparison of the sequences of a molecule (example: 16S rRNA) taken from different species
 - Residues conserved during evolution play an important role
- Prediction of protein structure and function
 - Proteins which are very similar in sequence generally have similar 3D structure and function as well
 - By searching a sequence of unknown structure against a database of known proteins the structure and/or function can in many cases be predicted

Database Search and Sequence Alignment

Definition (Sequence alignment/comparison)

The arrangement of two or more amino acid or nucleotide sequences in such a way as to maximize their similarity under some scoring function. Alternatively – we want to minimize the *edit distance* between the sequences

Definition (Edit distance)

The minimum number of **substitutions**, **deletions** or **insertions** required to convert one string into another

Database Search and Sequence Alignment

Example: How do we align "kitten" and "sitting"?

- 1 kitten → **s**itten (substitution)
- 2 **s**itten → sitt**i**n (substitution)
- 3 sittin → sitt**ing** (insertion)

KITTEN-
SITTING

Things to Keep in Mind

- How do we determine the score?
 - What is the reward for a match? Same for all matches?
 - What is the penalty for a mismatch? Are all mismatches the same? (Usually not. We use substitution matrices to estimate this)
 - Gap penalty – Same penalty for opening a gap vs. extending it?
- How do we perform the alignment? (Dynamic programming or variants)
- How do we statistically evaluate the significance of our results?

Things to Keep in Mind When Working With Alignments

- Pairwise alignment programs always find the optimal alignment of two sequences
 - They do so even if it does not make any sense at all to align the two sequences
 - "Optimal" means optimal according to the **substitution matrix** and **gap penalties** you choose – also if you choose the wrong ones
- Generally the underlying assumptions are wrong
 - The frequency of substitution is not the same at all positions
 - Nor is the frequencies of insertions and deletions the same
 - Affine gap penalties do not properly model ins/del events

Using Sequence Alignment to Search Databases

- The most common usage of pairwise sequence alignment is searching databases for related sequences
- Although the alignments themselves may be unreliable the alignment scores gives a lot of information about which sequences are related and which are not
- Having a set of related sequences is a lot more informative than just one sequence – even if nothing is known about the related sequences

Requirements for Sequence Alignment

- A very fast method to find potentially related sequences
 - Systematically searching through the databases with the alignment methods take too long even though dynamic programming is fast
 - Some method to initially identify possible matches is therefore needed to speed up the search
- A method to evaluate which matches to trust
 - Statistics on the alignment score distributions can be used to calculate the significance of an alignment
 - This way we can not only rank which matches are better than others but also tell if any of them are good at all

Local or Global Alignment

- Global alignment “forces” the alignment of the entire sequence.
- Generally local alignment is used for performing database searches
 - For most cases you would be interested in knowing if any parts of your sequences look like something else
 - The protein sequence databases have not been split into domains
- It is not always the optimal thing to do but ...
 - In the case where the complete sequence should match the local alignment score will be almost identical to the global one
 - If you really want a global alignment you can make it afterwards

```
Global  FTFTALILLAVAV
        F--TAL-LLA-AV
```

```
Local   FTFTALILL-AVAV
        --FTAL-LLAAV--
```

Local or Global Alignment

- Because you can start a new alignment anywhere dynamic programming scores cannot become negative
- The trace-back is started at the highest values rather than the lower right corner
- The trace-back is stopped as soon as a zero is encountered

Global Alignment – Generic Example

- Here we use the basic edit distance for demonstration purposes.
- We allocate an $(m+1) \times (n+1)$ table, where m and n are the sizes of the sequences, plus a 0^{th} row and a 0^{th} column.
- The dynamic programming equation below tells us how to fill the table, from top to bottom and left to right.
- In global alignment, $C[m, n]$ is the final result.

j	0	1	2	3	4	5	6
$i \backslash y_j$		B	D	C	A	B	A
0 x_i	0	0	0	0	0	0	0
1 A	0	↑	↑	↑	↖	←	↖
2 B	0	↖	↑	←	↑	↖	←
3 C	0	↑	↑	↖	←	↑	↑
4 B	0	↖	↑	↑	↑	↖	←
5 D	0	↑	↖	↑	↑	↖	↑
6 A	0	↑	↑	↑	↖	↑	↖
7 B	0	↖	↑	↑	↑	↖	↑

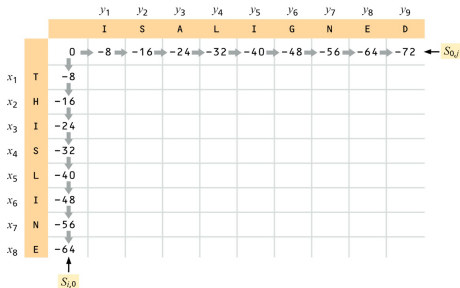
Global Alignment – Generic Example

We fill the table top to bottom, left to right, as:

$$c[i, j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ c[i-1, j-1] + 1 & \text{if } i, j > 0 \text{ and } x_i = y_j \\ \max\{c[i-1, j], c[i, j-1]\} & \text{if } i, j > 0 \text{ and } x_i \neq y_j \end{cases}$$

- $c[i, j]$ represents the match score between $x[1 \dots i]$ and $y[1 \dots j]$.
- If any of the indices is 0, this is a match with an empty string, which is by definition 0.
- Our final score is $c[m, n]$.

Global Alignment: Needleman-Wunsch



x = THISLINE

y = ISALIGNED

$S_{i,j}$ stores the score of the optimal alignment of all characters/residues up to x_i of x with all residues up to y_j of y .

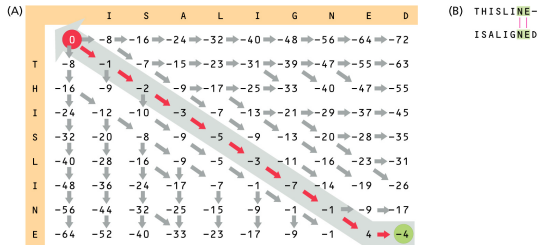
The first row and columns are gaps.

Optimal alignment:

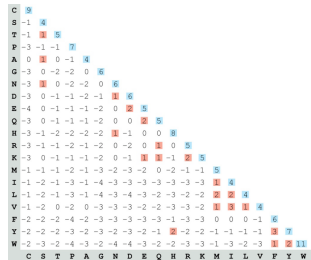
THIS-LI-NE-
--ISALIGNED

$$S_{i,j} = \max \begin{cases} S_{i-1,j-1} + s(x_i, y_j) \\ S_{i-1,j} + g \\ S_{i,j-1} + g \end{cases}$$

Global Alignment: Needleman-Wunsch



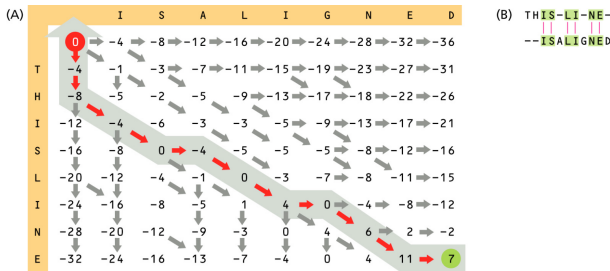
$$S_{i,j} = \max \begin{cases} S_{i-1,j-1} + s(x_i, y_j) \\ S_{i-1,j} + g \\ S_{i,j-1} + g \end{cases}$$



Scoring matrix used:
BLOSUM-62

The gap penalty is so high (-8) that there is no incentive to add gaps rather than allow mismatches (the most severe of which has a penalty of -4) The "fault" is with the scoring matrix used the alignment is optimal within the scoring matrix used.

Global Alignment: Needleman-Wunsch



$$S_{i,j} = \max \begin{cases} S_{i-1,j-1} + s(x_i, y_j) \\ S_{i-1,j} + g \\ S_{i,j-1} + g \end{cases}$$

Scoring matrix used matches the gap penalty (-4) to the most severe mismatch (-4).

From Global to Local Alignment

Main differences over Needleman-Wunsch:

- Whenever the score of the optimal sub-alignment is less than zero, it is rejected (the matrix element is set to 0)
- Traceback starts from the highest-scoring element:

$$S_{i,j} = \max \begin{cases} S_{i-1,j-1} + s(x_i, y_j) \\ S_{i-1,j} + g(n_{gap1})_{1 \leq n_{gap1} \leq i} \\ S_{i,j-1} + g(n_{gap2})_{1 \leq n_{gap2} \leq j} \\ 0 \end{cases}$$

What does the rejection of a negative optimal sub-alignment mean?

Hint: many mini global alignments not worth to continue at some point

Note that the score given takes into account affine gap penalties (penalizing more for opening a gap, less for extending a gap)

The Smith-Waterman algorithm (local alignment)

		H	E	A	G	A	W	G	H	E	E
P	0	0	0	0	0	0	0	0	0	0	0
A	0	0	0	5	0	5	0	0	0	0	0
W	0	0	0	0	2	0	20	12	4	0	0
H	0	10	2	0	0	0	12	18	22	14	6
E	0	2	16	8	0	0	4	10	18	28	20
A	0	0	8	21	13	5	0	4	10	20	27
E	0	0	6	13	18	12	4	0	4	16	26

AWGHE

AW-HE

Substitution Matrices

What is a substitution matrix?

	A	G	C	T
A	+1	-3	-3	-3
G	-3	+1	-3	-3
C	-3	-3	+1	-3
T	-3	-3	-3	+1

An Example of a Substitution Matrices

	A	G	C	T
A	+1	-3	-3	-3
G	-3	+1	-3	-3
C	-3	-3	+1	-3
T	-3	-3	-3	+1

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

$$\text{Score} = 19 - 9 = 10$$

Why Use Substitution Matrices?

- Determine likelihood of homology between two sequences.
- Substitutions that are more likely should get a higher score,
- Substitutions that are less likely should get a lower score.

Scoring Matrices

- Log-odds matrix where each cell gives the probability of aligning those two residues
- Score of alignment = Sum of log-odds scores of residues
- Score for each residue given by:

$$s(a, b) = \frac{1}{\lambda} \log\left(\frac{p_{ab}}{f_a f_b}\right)$$

Types of Matrices

- **Percent Identity** – Standard scoring matrix to align DNA sequences
- **PAM** – Estimates the rate at which each possible residue in a sequence changes to each other residue over time
- **BLOSUM-X** – Identifies sequences that are X% similar to the query sequence

Nucleotide Scoring Matrix

Approximate ratios used on the web page:

Percent identity	Match/Mismatch
99%	1/-3
98%	2/-5
95%	1/-2
90%	2/-3
85%	3/-4
80%	4/-5
75%	1/-1
70%	11/-10
65%	5/-4
60%	7/-5
50%	3/-2

Amino Acid PAM Matrices

- **P**ercent **A**ccepted **M**utation
- Dayhoff (1978), 1572 changes in 71 families of proteins, at least 85% similar
- For each amino acid, count 20 numbers
- For example, how many F (phenylalanine) stay the same, how many change to the other 19 amino acids
- Normalize: divide each of these 20 numbers by (sum of 20 numbers)
- PAM1: 1% probability of change

The Column/Row of F in PAM1

F to A: 0.0002	F to L: 0.0013
F to R: 0.0001	F to K: 0.0000
F to N: 0.0001	F to M: 0.0001
F to D: 0.0000	F to F: 0.9946
F to C: 0.0000	F to P: 0.0001
F to Q: 0.0000	F to S: 0.0003
F to E: 0.0000	F to T: 0.0001
F to G: 0.0001	F to W: 0.0001
F to H: 0.0002	F to Y: 0.0021
F to I: 0.0007	F to V: 0.0001

$$PAM_2 = PAM_1 * PAM_1 = (PAM_1)^2$$

$$PAM_{250} = (PAM_1)^{250}$$

Example - PAM120

C	9																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
---	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

- **BLO**cks of amino acid **SU**bstitution **M**atrices
- Start with highly-conserved patterns (blocks) in a large set of closely related proteins
- Use the likelihood of substitutions found in those sequences to create a substitution probability matrix
- BLOSUM-n means that the sequences used were n% alike
- BLOSUM62 is “standard”
- Nature Biotechnology: <http://www.nature.com/nbt/journal/v22/n8/abs/nbt0804-1035.html>

Example of BLOSUM62

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

Example of BLOSUM62

Figure 1 shows a 20x20 matrix of log-odds scores for amino acid substitutions. The matrix is symmetric, with the diagonal elements all equal to 4. The scores range from -3 (Serine to Aspartate) to 11 (Tryptophan to Tryptophan). Annotations highlight that negative scores indicate less likely substitutions, positive scores indicate more likely substitutions, common amino acids have low weights, and rare amino acids have high weights.

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

Which Scoring Matrix to Use?

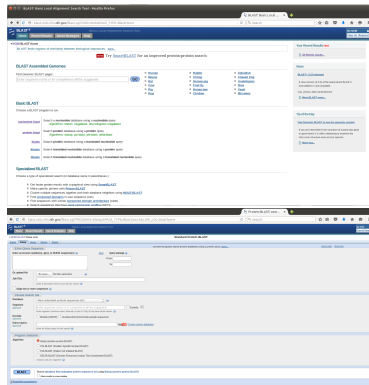
- How can one decide whether to use BLOSUM or PAM when comparing and aligning sequences?
- This decision is also more difficult when the evolutionary distance between the sequences is not known
- What to do: try different ones and compare results
- Different studies have concluded that for the PAM matrices it is generally best to try PAM40, PAM120, and PAM250
- When used for local alignments
 - Lower PAM matrices find short local alignments
 - Higher PAM matrices find longer but weaker local alignments
- Several different matrices should be used, and the alignment that is judged to be evolutionarily the most accurate is the one chosen
 - Question: how can one judge which one is the most accurate?
 - Judgment on a control set where the evolutionary relationship is known

Heuristic Search Algorithms

- FASTA (Pearson 1995)
- Uses heuristics to avoid calculating the full dynamic programming matrix
- Speed up searches by an order of magnitude compared to full Smith-Waterman
- The statistical side of FASTA is still stronger than BLAST
- BLAST (Altschul 1990, 1997)
- Uses rapid word lookup methods to completely skip most of the database entries
- Extremely fast
- Almost as sensitive as FASTA

<http://www.ncbi.nlm.nih.gov/BLAST/>

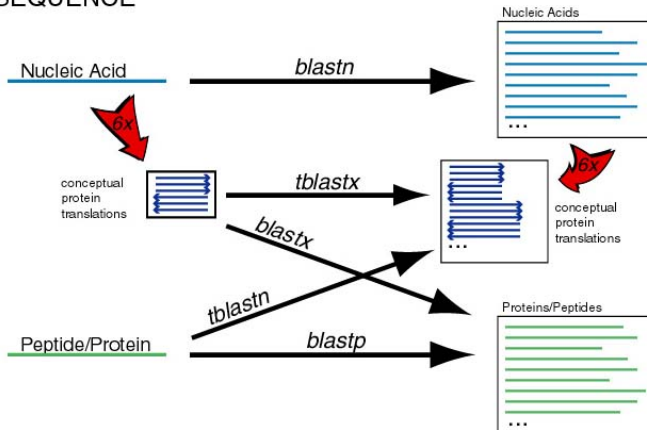
- Very fast computer dedicated to running BLAST searches
- Many databases that are always up to date
- Nice simple web interface
- But you still need to knowledge about BLAST to use it properly



Different BLAST Programs

QUERY
SEQUENCE

DATABASE



Pairwise alignment of hemoglobin α chain and myoglobin

```
24.7% identity;          Global alignment score: 130

      10      20      30      40      50
HBA_HU VLSPADKTNVKAAWGKVGAAHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKG---
      ::: ..  :  ::::: : .. :. : : : : : : : : : : : : : : : : : : : : :
MYG_PH VLSEGEWQLVLHVWAKVEADVAGHGQDILIRLFKSHFETLEKFDRFKHLKTEAEMKASED
      10      20      30      40      50      60

      60      70      80      90      100     110
HBA_HU ---HGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHL
      :: : : : : . . : : : : : : : : : : : : : : : : : : : : : : : :
MYG_PH LKKHGVTVLTAIGAILKKKGHAEELKPLAQSHATKHKI-PIKYLEFISEAIIHVLHSRH
      70      80      90      100     110

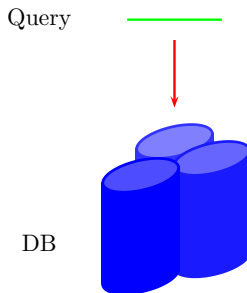
      120     130     140
HBA_HU PAEFTPAVHASLDKFLASVSTVLTSKYR-----
      :.. : : : : : : : : : : : : : : : : : : : : : : : : : : :
MYG_PH PGDFGADAQGAMNKALELFRKDIAAKYKELGYQG
      120     130     140     150
```

How BLAST Works

Basic Local Alignment Search Tool

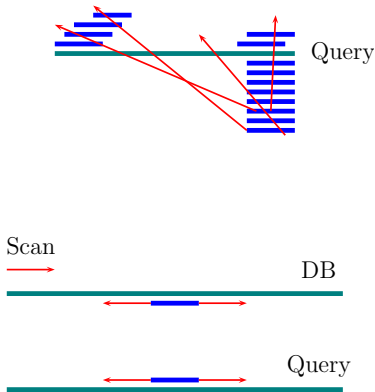
Main idea:

- Construct a dictionary of all the words in the query
- Initiate a local alignment for each word match between query and DB
- Running Time: $O(MN)$
- However, orders of magnitude faster than Smith-Waterman



Blast – Original Version

- **Dictionary:** All words of length k (approx. 11)
Alignment initiated between words of alignment score approx. T (typically $T = k$)
- **Alignment:** Ungapped extensions until score below statistical threshold
- **Output:** All local alignments with score more than statistical threshold



How BLAST works

- The search is accelerated by indexing the sequence databases in a so-called suffix array
 - Three letter subsequences are used as keys to the sequences
 - Closely related substitutions are also included
 - This gives approx. 150 index keys for each sequence
- This is used in two ways
 - To quickly discard sequences that are not similar at all before even beginning to align them
 - To constrain the alignment and thereby speed up the alignment procedure itself

Evaluating the Significance of an Alignment

- **Score and bit-score:** depend on scoring method.
- **Z-score** = $\frac{\text{score} - \text{mean}}{\text{stddev}}$
- **E-value (Expect value):** number of unrelated database sequences expected to yield same or higher score by pure chance
- **P-value (Probability):** probability that a database yields by pure chance at least one alignment with same or higher score

Evaluating the Significance of an Alignment

- The E-value describes the number of hits one can "expect" to see by chance when searching a database of a particular size.
- It decreases exponentially with the Score (S) that is assigned to a match between two sequences.
- It essentially describes the random background noise that exists for matches between sequences.
- The E-value is used as a convenient way to create a significance threshold for reporting results.
- When increased from the default value of 10, a larger list with more low-scoring hits can be reported.
- E-value approaching zero \rightarrow significant alignment. Less than 0.01 = almost always homologous; $1e-10$ for nucleotide searches of $1e-4$ for protein searches = frequently related

- In BLAST 2.0, the E-value is also used instead of the P-value (probability) to report the significance of matches. For example, an E value of 1 assigned to a hit can be interpreted as meaning that in a database of the current size one might expect to see 1 match with a similar score simply by chance.
- Be careful when comparing E- or P-values from different searches.
- Comparison is only meaningful for different query sequences searched against the same database with the same BLAST parameters.

How to Interpret Log-odds Matrix

- If you know the scores in a matrix, how do determine what kind of alignments it will find?
- You need to determine the frequencies implied by the scores
- Works backwards of course:

$$s(a, b) = \frac{1}{\lambda} \log\left(\frac{p_{ab}}{f_a f_b}\right) \Rightarrow f_a f_b e^{\lambda s(a,b)} = p_{ab}$$

How to Interpret Log-odds Matrix

In order to find p_{ab} , you need to find λ .

$$f_a f_b e^{\lambda s(a,b)} = p_{ab}$$

All probabilities must add up to 1, to set it to 1 and solve for lambda

$$\sum_{a,b} f_a f_b e^{\lambda s(a,b)} = 1$$

The E-value

- In the limit of sufficiently large sequence lengths m and n , the statistics of High Scoring Segment Pairs (HSP) scores are characterized by two parameters, K and λ .
- The expected number of HSPs with score at least S is given by the formula:

$$E = K m n e^{-\lambda S}$$

- This formula makes eminently intuitive sense: Doubling the length of either sequence should double the number of HSPs attaining a given score.
- The value also decreases exponentially with the score.
- The parameters K and λ can be thought of simply as natural scales for the search space size and the scoring system respectively.

A Curse or a Blessing?

- Large databases are a blessing ...
 - They are more likely to contain something similar to the query
- ... and a curse
 - Increasing the size of the database decreases the significance of the hits you get
 - Searching huge databases requires fast computer
- What requirements this puts on software development
 - The programs must be speeded up or database searches will take longer and longer
 - The false positive rate must be reduced to not lose specificity

Multiple Sequence Alignment (MSA)

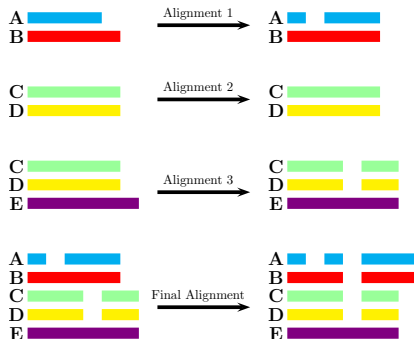
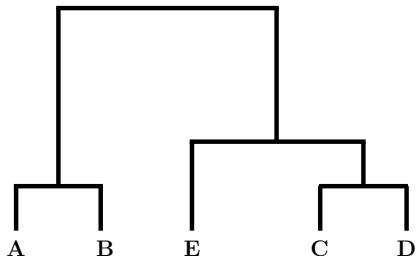
```
HBB_HUMAN  -----VHLTPEEKSAVTALWGKVN--VDEVGGEALGRLLVVYPWTQRFFESFGDLST
HBB_HORSE  -----VQLSGEEKAAVLALWDKVN--EEEVGGEALGRLLVVYPWTQRFFDSFGDLSN
HBA_HUMAN  -----VLSPADKTNVKAAGKVGAGHAGEYGAEALERMFSLFPTTKTYFPHF-DLS-
HBA_HORSE  -----VLSAADKTNVKAAWSKVGGHAGEYGAEALERMFSLFPTTKTYFPHF-DLS-
MYG_PHYCA  -----VLSGEWQLVLHVWAKVEADVAGHGQDILIRLFKSHPETLEKFFDRFKHLKT
GLB5_PETMA PIVDTGSVAPLSAAEKTIRSAWAPVYSTYETSGVDILVKFFTSTPAAQEFPKFKGLTT
LGB2_LUPLU -----GALTESQAALVKSSWEEFNANIPKHTHRFFILVLEIAPAAKDLFSFLKGTSE
          *:  :  :  *  .           :  .:  *  :  *  :  .

HBB_HUMAN  PDAVMGNPKVKAHGKKVLGAFSDGLAHLDN-----LKGTFATLSELHCDKLHVDPENFRL
HBB_HORSE  PGAVMGNPKVKAHGKKVLHLSFGEGVHHLDN-----LKGTFATLSELHCDKLHVDPENFRL
HBA_HUMAN  ----HGSAQVKGHGKKVADALTNVAHVDD-----MPNALSLSLDLHAHKLRVDPVNFKL
HBA_HORSE  ----HGSAQVKAHGKKVGDAITLAVGHLD-----LPGALSLSLDLHAHKLRVDPVNFKL
MYG_PHYCA  EAEMKASEDLKKHGVTVLTAIGAILKKKGH-----HEAELKPLAQSHATKHKIPIKYLEF
GLB5_PETMA ADQLKKSADVRWHAERIINAVNDASMDDT--EKMSMKLRDLGSKHAKSFQVDPQYFKV
LGB2_LUPLU VP--QNNFELQAHAGKVFKLVEAAIQLQVTGVVTDATLKNLGSVHVSKGVAD-AHFPV
          . .:: * .  :  .           :  * .  *  .           :  .
```

Why MSA is Better?

- More sequences contain more information
- Multiple sequence alignment allows us to compare all related proteins simultaneously
- It allows us to identify features that are conserved among the sequences
- Using a multiple sequence alignment (a profile) one can find more related sequences than by simple pairwise comparison

Building a Phylogenetic Tree



Assembling the Tree of Life



M. Madigan and B. Mairs, 1997

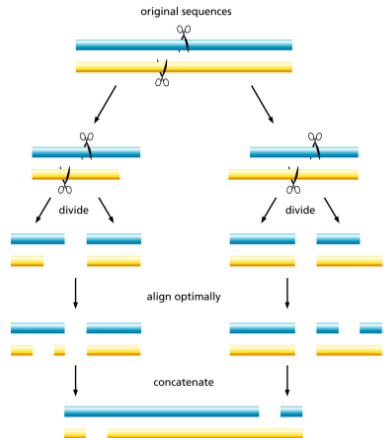
Assembled from aligned sequences of ribosomal RNA

Multiple Sequence Alignment

- Multiple sequence alignment is NP-hard.
- The most practical and widely used method in multiple sequence alignment is the hierarchical extensions of pairwise alignment methods.
- The principal is that multiple alignments is achieved by successive application of pairwise methods.

Divide and Conquer

- Divide the sequences near their midpoint.
- Repeat until length falls below threshold.
- Feed sequences to MSA.
- Merge sequences.



Multiple Sequence Alignment – Summary of Steps

- Compare all sequences pairwise.
- Perform cluster analysis on the pairwise data to generate a hierarchy for alignment. This may be in the form of a binary tree (guide tree).
- Build the multiple alignment by first aligning the most similar pair of sequences, then the next most similar pair and so on.
- Once an alignment of two sequences has been made, then this is fixed.
- Thus for a set of sequences A, B, C, D having aligned A with C and B with D the alignment of A, B, C, D is obtained by comparing the alignments of A and C with that of B and D using averaged scores at each aligned position.

	v_1	v_2	v_3	v_4
v_1	–			
v_2	.17	–		
v_3	.87	.28	–	
v_4	.59	.33	.62	–

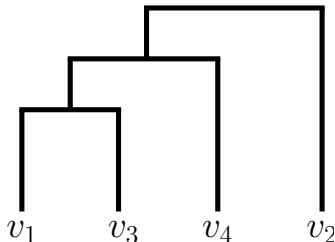
.17 means 17% identical.

Calculate:

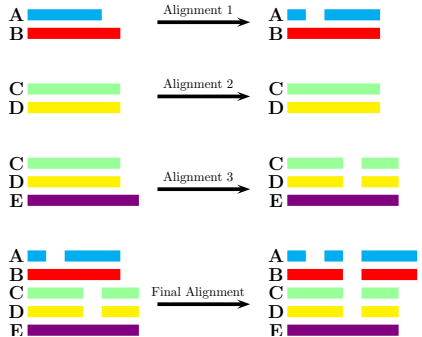
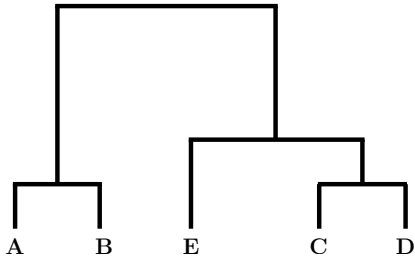
$$v_{1,3} = \text{alignment}(v_1, v_3)$$

$$v_{1,3,4} = \text{alignment}((v_{1,3}), v_4)$$

$$v_{1,2,3,4} = \text{alignment}((v_{1,3,4}), v_2)$$



Example



Building a Consensus Sequence

- Concatenation of all the sequences can give a consensus sequence
- The consensus character for column i is the character that minimizes the summed distance to it from all the characters in column i
- Distance is measured using the substitution matrix
- A very simple method, but doesn't account for variability.
- Useful for highly conserved sequences.

A	B	A
A	B	–
–	B	A
C	A	–
<hr/>		
A	B	A

Sequence Patterns

Patterns are known as regular expressions.

- The Prosite syntax for patterns:
 - Uses one-letter codes for amino acids (G=Gly, P=Pro, ...)
 - Each element in a pattern is separated from its neighbor by a `'_'`
 - The symbol `'X'` is used where any amino acid is accepted
 - Ambiguities are indicated by square parentheses `'[]'` ([AG] means Ala or Gly)
 - Amino acids that are not accepted at a given position are listed between a pair of curly brackets `'{}'` ({AG} means any amino acid except Ala and Gly),
 - Repetitions are indicated between parentheses `'()'` ([AG](2,4) means Ala or Gly between 2 and 4 times, X(2) means any amino acid twice).
 - A pattern is anchored to the first and last positions in the protein by the symbols `'<'` and `'>'` respectively.

Sequence Patterns – Example

The following pattern: $< A - x - [ST](2) - x(0, 1) - \{V\}$ means:

- An Alanine (A) in the first position
- Followed by any amino acid,
- Followed by a Serine (S) or Threonine (T) twice.
- Followed or not by any amino acid.
- Followed by any amino acid except Valine (V).

How to Build a Pattern

GHEGVGKVVKLGAGA
GHEKKGYFEDRGPSA
GHEGYGGRSRGGGYS
GHEFEGPKGCGALYI
GHELRGTTFMPALEC

↓

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
G	H	E	G	V	G	K	V	V	K	L	G	A	G	A
			K	K		Y	F	E	D	R	A	P	S	S
			F	Y		G	R	S	R	G		G	Y	I
			L	E		P	K	G	C	P		L	E	C
				R		T	T	F	M					

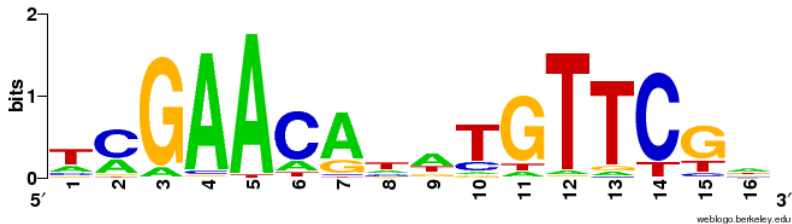
↓

Pattern: $G - H - E - X(2) - G - X(5) - [GA] - X(3)$

↓

Search databases

Sequence Logo



Pros and Cons of Profiles

- Fast and easy to implement and understand.
- Unlike a consensus sequence – can accommodate alternative amino acids per position.
- Not sensitive to insertions/deletions.
- Small patterns find a lot of false positives. Long patterns are very difficult to design.

Searching Similar Sequences Using PSI-BLAST

PSI-Blast = Position Specific Iterated BLAST.

- A standard BLAST search is performed against a database using a substitution matrix (e.g. BLOSUM62).
- A position-specific scoring matrix (PSSM) is constructed automatically from a multiple alignment of the highest scoring hits of the initial BLAST search. High conserved positions receive high scores and weakly conserved positions receive low scores.
- The PSSM replaces the initial matrix to perform a second BLAST search.
- The former steps can be repeated and the new found sequences included to build a new PSSM.
- We say that the PSI-BLAST has converged if no new sequences are included in the last cycle.

PSI-BLAST dangers

- Avoid too close sequences \rightarrow overfit!
- Can include false homologous! Therefore check the matches carefully: include or exclude sequences based on biological knowledge.
- The E-value reflects the significance of the match to the previous training set, not to the original sequence!
- Choose carefully your query sequence.
- Try reverse experiment to certify.

ClustalW for Multiple Sequence Alignment

- ClustalW can create multiple alignments, manipulate existing alignments, do profile analysis and create phylogenetic trees.
- Scoring alignments by calculating all the pairwise scores and progressively build a tree using a neighbor joining algorithm.
- Alignment can be done by 2 methods: slow/accurate or fast/approximate.

- MUSCLE – MUltiple Sequence Comparison by Log-Expectation. Significantly faster than ClustalW and often gives better results.
- T-Coffee (Tree-based Consistency Objective Function For alignment Evaluation).
- MAFFT (Multiple Alignment using Fast Fourier Transform).

General Considerations for MSA

- The more sequences to align the better.
- Don't include similar ($> 80\%$) sequences.
- Sub-groups should be pre-aligned separately, and one member of each subgroup should be included in the final multiple alignment.

Sources Cited

- Debra Goldberg, Algorithms for Molecular Biology, Fall 2008 www.bioalgorithms.info (lectures for students and faculty).
- Daniel Sam, “Greedy Algorithm” presentation.
- Glenn Tesler, “Genome Rearrangements in Mammalian Evolution: Lessons from Human and Mouse Genomes” presentation.
- Ernst Mayr, “What evolution is”.
- Neil C. Jones, Pavel A. Pevzner, “An Introduction to Bioinformatics Algorithms” .
- Alberts, Bruce, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, Peter Walter. Molecular Biology of the Cell. New York: Garland Science. 2002.
- Mount, Ellis, Barbara A. List. Milestones in Science & Technology. Phoenix: The Oryx Press. 1994.
- Voet, Donald, Judith Voet, Charlotte Pratt. Fundamentals of Biochemistry. New Jersey: John Wiley & Sons, Inc. 2002.
- Campbell, Neil. Biology, Third Edition. The Benjamin/Cummings Publishing Company, Inc., 1993.
- Snustad, Peter and Simmons, Michael. Principles of Genetics. John Wiley & Sons, Inc, 2003.