

# **Module: Sequence Alignment Theory and Applications**

## **Session: Pair-wise Sequence Alignment**

# Learning Objectives



- Basic principles of pair-wise alignments:
  - Algorithms ( & use of scoring matrices & gap penalties) to produce an alignment
- Concepts of the dynamic programming approach for pair-wise sequence alignment (global and local)
  - Steps performed by the dynamic programming algorithm
- Probability and statistical analysis of Sequence Alignment

# Learning Outcomes



- Align two sets of sequences using both a global and local alignment approach
- Understand the output of the pair-wise alignment  
Describe steps performed by the dynamic programming algorithm during alignment
- Understand probability and statistical analysis of sequence alignment

# Sequence Alignment Algorithms:



- **An algorithm** is a sequence of actions to be performed to arrive at a solution
- Rigorous algorithms (optimal alignments)- Dynamic programming
  - Needleman-Wunsch ('70) used for **global alignments**
  - Smith-Waterman ('81) for **local alignments**; provides one or more alignments of the sequences
- Heuristic algorithms (faster but only just approximate alignments...**see session 3**)
  - BLAST ('90)
  - FASTA ('85)

# Dynamic Programming Algorithm-1



## The Challenge:

- Sequence Alignment for **optimal** (*Highest scoring*) **output** gets cumbersome and nearly unmanageable for long and or a large number of sequences
- Alignment **Algorithms** can be compiled to help

# Dynamic Programming Algorithm-2



- Strategy:
  - Break alignment down into series of sequential sub-alignment that can be readily computed
  - Align each sub-solution optimally
  - Avoid recalculating the scores already considered
  - Piece the sub-solutions back together (i.e. if sub-solution are optimal..then entire solution must be optimal)
- Dynamic programming algorithms : Guaranteed to return the highest scoring or optimal alignment between two sequences for a given scoring system

# Scoring a sequence alignment



Sequence 1 V D S - C Y

Sequence 2 V E S L C Y

Score 4 2 4 -11 9 7

**Score = sum of amino acid pair scores (26)**

**minus single gap penalty (11) = 15**

1. Individual alignment scores are taken from an amino acid substitution matrix
2. Non-identical amino acids can be placed in corresponding positions.
3. Scores gained by each match are not always the same, for instance two rare amino acids will score more than two common.
4. Alignment gap(s) may be introduced for optimising the score. Gaps cause penalties.

# Dynamic Programming Algorithm-3



- Programs, including web hosted ones (e.g. LALIGN <http://www.ebi.ac.uk/Tools/psa/lalign/>, EMBOSS Water [www.ebi.ac.uk/Tools/psa/emboss\\_water/](http://www.ebi.ac.uk/Tools/psa/emboss_water/)), performing this type of analysis are readily available
- Method requires careful consideration for choice of variable settings in the programs:
  - Scoring matrix
  - Gap penalties
- Method is highly computationally demanding



# The Steps



$$\begin{array}{rcl}
 \text{1. Score of new} & = & \text{Score of previous} + \text{Score of new} \\
 \text{alignment} & & \text{alignment (A)} \quad \text{aligned pair} \\
 \begin{array}{c} \mathbf{V \ D \ S \ - \ C \ Y} \\ \mathbf{V \ E \ S \ L \ C \ Y} \end{array} & & \begin{array}{c} \mathbf{V \ D \ S \ - \ C} \\ \mathbf{V \ E \ S \ L \ C} \end{array} \quad \begin{array}{c} \mathbf{Y} \\ \mathbf{Y} \end{array} \\
 15 & = & 8 + 7
 \end{array}$$

$$\begin{array}{rcl}
 \text{2. Score of} & = & \text{Score of previous} + \text{Score of new} \\
 \text{alignment (A)} & & \text{alignment (B)} \quad \text{aligned pair} \\
 \begin{array}{c} \mathbf{V \ D \ S \ - \ C} \\ \mathbf{V \ E \ S \ L \ C} \end{array} & & \begin{array}{c} \mathbf{V \ D \ S \ -} \\ \mathbf{V \ E \ S \ L} \end{array} \quad \begin{array}{c} \mathbf{C} \\ \mathbf{C} \end{array} \\
 8 & = & -1 + 9
 \end{array}$$

3. Repeat removing aligned pairs until end of alignments is reached

# Example Global alignment by dynamic programming



Aligning sequences: a1a2a3a4 and b1b2b3b4

# Step 1



|     | gap    | a1    | a2     | a3     | a4     |
|-----|--------|-------|--------|--------|--------|
| gap | 0      | 1 gap | 2 gaps | 3 gaps | 4 gaps |
| b1  | 1 gap  |       |        |        |        |
| b2  | 2 gaps |       |        |        |        |
| b3  | 3 gaps |       |        |        |        |
| b4  | 4 gaps |       |        |        |        |

Credit: David Mount, 2004

2a.

|     | gap    | a1    | a2     | a3     | a4     |
|-----|--------|-------|--------|--------|--------|
| gap | 0      | 1 gap | 2 gaps | 3 gaps | 4 gaps |
| b1  | 1 gap  | s11   |        |        |        |
| b2  | 2 gaps |       |        |        |        |
| b3  | 3 gaps |       |        |        |        |
| b4  | 4 gaps |       |        |        |        |

# 2b.

|     | gap    | a1    | a2     | a3     | a4     |
|-----|--------|-------|--------|--------|--------|
| gap | 0      | 1 gap | 2 gaps | 3 gaps | 4 gaps |
| b1  | 1 gap  | s11   | 4      |        |        |
| b2  | 2 gaps | s12   |        |        |        |
| b3  | 3 gaps |       |        |        |        |
| b4  | 4 gaps |       |        |        |        |

**Trial1:** a1-b2 + 1 gap: - a1 X X X  
b1b2 X X -

**Trial2:** s11 + 1 gap: a1 - X X  
b1b2 X X

2c.

|     | gap    | a1    | a2     | a3     | a4     |
|-----|--------|-------|--------|--------|--------|
| gap | 0      | 1 gap | 2 gaps | 3 gaps | 4 gaps |
| b1  | 1 gap  | s11   | s21    |        |        |
| b2  | 2 gaps | s12   |        |        |        |
| b3  | 3 gaps |       |        |        |        |
| b4  | 4 gaps |       |        |        |        |

The diagram illustrates a grid structure with labels **a1**, **a2**, **a3**, **a4** in the top row and **b1**, **b2**, **b3**, **b4** in the left column. The grid cells contain text indicating the number of gaps or specific symbols. Arrows highlight specific relationships: a vertical arrow points from '2 gaps' in the cell (gap, a2) to 's21' in the cell (b1, a2); a horizontal arrow points from 's11' in the cell (b1, a1) to 's21' in the cell (b1, a2); and a diagonal arrow points from '1 gap' in the cell (b1, gap) to 's21' in the cell (b1, a2).

2d.

|     | gap    | a1    | a2     | a3     | a4     |
|-----|--------|-------|--------|--------|--------|
| gap | 0      | 1 gap | 2 gaps | 3 gaps | 4 gaps |
| b1  | 1 gap  | s11   | s21    |        |        |
| b2  | 2 gaps | s12   | s22    |        |        |
| b3  | 3 gaps |       |        |        |        |
| b4  | 4 gaps |       |        |        |        |

The diagram illustrates sequence alignment between two sequences, a1 and a2, across four rows (b1 to b4). The alignment is shown in the table above. Arrows indicate the following transitions:

- A diagonal arrow from s11 to s22.
- A vertical arrow from s21 to s22.
- A vertical arrow from the '2 gaps' cell in the first row to s22.
- A horizontal arrow from s12 to s22.
- A horizontal arrow from the '2 gaps' cell in the second row to s22.

### 3.Part of trace back matrix

|     | gap    | a1        | a2     | a3     | a4     |
|-----|--------|-----------|--------|--------|--------|
| gap | 0      | 1 gap     | 2 gaps | 3 gaps | 4 gaps |
| b1  | 1 gap  | s11 ← s21 | s31    | s41    |        |
| b2  | 2 gaps | s12       | s22    | s32    | s42    |
| b3  | 3 gaps | s13       | s23    | s33    | s43    |
| b4  | 4 gaps | s14       | s24    | s34    | s44    |



## 4. Trace back matrix

|     | gap    | a1        | a2        | a3        | a4     |
|-----|--------|-----------|-----------|-----------|--------|
| gap | 0      | 1 gap     | 2 gaps    | 3 gaps    | 4 gaps |
| b1  | 1 gap  | s11 ← s21 | s31       | s41       |        |
| b2  | 2 gaps | s12       | s22 ← s32 | s42       |        |
| b3  | 3 gaps | s13       | s23       | s33 ← s43 |        |
| b4  | 4 gaps | s14       | s24       | s34       | s44    |

Alignment A:

|    |    |    |    |
|----|----|----|----|
| a1 | a2 | a3 | a4 |
| b1 | b2 | b3 | b4 |

Credit: David Mount, 2004

Alignment B:

|    |    |    |    |    |
|----|----|----|----|----|
| a1 | a2 | a3 | a4 | -- |
| b1 | -- | b2 | b3 | b4 |

# Dynamic programming can provide Global or Local Sequence alignments



- **Modifications for Local alignments (Smith Waterman Algorithm):**
  - Scoring system must include –ve scores for mismatches
  - When matrix position value becomes –ve, converted to zero, terminating alignment up to that point.
  - Trace back formulation modified to have zero as the minimum value at any position

# Dynamic Programming: Algorithm

## procedure- Gap scores



- Gaps in sequence alignments are given a big penalty to reflect fact that they are not expected to occur very often
- Gap Opening penalty- defines the cost for opening a gap in one of the sequences
  - No simple rule to predict optimal value for the gap penalty but if much higher than the default, local alignments containing gaps may split into shorter alignments
- Gap extension penalty – an extra penalty proportional to length of the gap. Usually <<< than Opening penalty.
- Gaps should be penalized more on their existence than their length

# Global alignment: Needleman-Wunsch algorithm



- Dynamic programming method can be used to give global alignments as described by Needleman and Wunsch (1970)
- The optimal score at each matrix position is calculated by adding the current match score to previously scored positions and subtracting gap penalties if applicable
- Each matrix position may have a **+ve**, or **–ve** score or Zero (**0**)

# Global alignment: Needleman-Wunsch algorithm cont..



- Needleman-Wunsch algorithm maximizes the number of matches between the sequences along the their entire length
- To produce a global sequence alignment from the scoring matrix, a second matrix called a **trace-back matrix** is produced
- Trace-back matrix keeps track of all the moves (residue alignment comparisons) in the scoring matrix
- Alignment produced by stringing together pairings from the optimal scores at each matrix position

# Advantages and Disadvantages of Needleman-Wunsch algorithm



- Suitable for global alignment of two closely related sequences
- Can miss best local similarities
- Unsuitable for aligning:
  - very divergent sequences,
  - sequences with different domain structures

# Local Alignment: Smith-Waterman Algorithm



- A modification of the dynamic programming algorithm for sequence alignment enables ability to create local sequence alignments (Smith and Waterman 1981a,b)
- Rules for calculating scoring matrix varies slightly from Needleman-Wunsch algorithm:
  - Scoring system includes –Ve scores for mismatches
  - When dynamic prog scoring matrix value becomes –ve, it is set to zero, which terminates alignment up to that point, and a new one can begin
- Trace-back matrix keeps track of all the trial moves
- Alignment produced by starting at the highest scoring positions in the matrix following a path up to Zero

# Advantages and disadvantages of Smith-Waterman Algorithm



- Local alignments more meaningful than global matches because:
  - Can identify conserved local sequence domains present in both sequences
  - Can match two sequences with different lengths of overlap
- Smith-Waterman algorithm struggles finding highest scoring alignment when sequences include regions that align locally separated by other poorly aligning regions



# Dynamic Programming: Algorithm procedure- Substitution Matrix



- Substitution matrix controls the cost of mutations:
  - Substitutions that are more likely should get a higher score
  - Substitutions that are less likely should get a lower score
- Appropriate substitution matrix can help determine likelihood of homology between two sequences

# Dynamic Programming: Algorithm

## procedure- Types of Substitution Matrices



- Percent Identity:
  - Standard scoring matrix for aligning nucleotide (DNA) sequences
- Percent Accepted Mutation (PAM)
  - Protein sequence alignment
  - Estimates the rate at which each possible residue in a sequence changes to each other residue over time
- BLOSUM
  - For proteins
  - Derived from alignments found in BLOCKS database
  - BLOSUM-X: Identifies sequences that are X% similar to the query sequence

# DNA Substitution Matrices



Sequence 1 ACTACCAGTTCATTGATACTTCTCAAA

Sequence 2 TACCATTACCGTGTAACTGAAAGGACTTAAAGACT

|   | A | G | C | T |
|---|---|---|---|---|
| A | 1 | 0 | 0 | 0 |
| G | 0 | 1 | 0 | 0 |
| C | 0 | 0 | 1 | 0 |
| T | 0 | 0 | 0 | 1 |

Match: 1

Mismatch: 0

**Score: 6**

# Protein Scoring Systems

- Matrices reflect:
  - # of mutations to convert one A.a. residue to another
  - Chemical similarity
  - Observed mutation frequencies
  - Probability of occurrence of each A.a. residue
- Widely used scoring matrices:
  - PAM
  - BLOSUM

# Percent Accepted Mutation (PAM)

- Family of matrices listing likelihood of change from one A.a. to another in homologous (i.e. Similar) protein sequences during evolution
  - Tracks evolutionary origin of proteins
  - Each gives changes expected for a given period of evolutionary time
- Predicted changes used to produce optimal alignments between two proteins and to score it
  - Assumption: A.a. substitutions observed over a shorter period of evolutionary history can be extrapolated to longer periods for higher PAMs

# Percent Accepted Mutation (PAM) cont..

- Dayhoff, 1978 calculated, PAM1 matrix reflecting average change of 1% of all A.a. positions
- The common PAM250 matrix represents extrapolation to a level of 250% change expected in 2500 million years
  - NB: sequences at this level of divergence still show 20% similarity
- Higher PAMs yield better alignments than lower numbered PAMs for distantly related proteins and vice versa. For example could use:
  - PAM120: aligning sequences with 40% similarity
  - PAM80: ~50% similarity
  - PAM60: ~60% similarity

# The PAM250 Matrix

|   |     |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
|---|-----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|----|----|--|--|
| C | Cys | 12 |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
| S | Ser | 0  | 2  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
| T | Thr | -2 | 1  | 3  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
| P | Pro | -3 | 1  | 0  | 6  |    |    |    |    |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
| A | Ala | -2 | 1  | 1  | 1  | 2  |    |    |    |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
| G | Gly | -3 | 1  | 0  | -1 | 1  | 5  |    |    |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
| N | Asn | -4 | 1  | 0  | -1 | 0  | 0  | 2  |    |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
| D | Asp | -5 | 0  | 0  | -1 | 0  | 1  | 2  | 4  |    |    |    |    |    |    |    |    |    |   |    |    |  |  |
| E | Glu | -5 | 0  | 0  | -1 | 0  | 0  | 1  | 3  | 4  |    |    |    |    |    |    |    |    |   |    |    |  |  |
| Q | Gln | -5 | -1 | -1 | 0  | 0  | -1 | 1  | 2  | 2  | 4  |    |    |    |    |    |    |    |   |    |    |  |  |
| H | His | -3 | -1 | -1 | 0  | -1 | -2 | 2  | 1  | 1  | 3  | 6  |    |    |    |    |    |    |   |    |    |  |  |
| R | Arg | -4 | 0  | -1 | 0  | -2 | -3 | 0  | -1 | -1 | 1  | 2  | 6  |    |    |    |    |    |   |    |    |  |  |
| K | Lys | -5 | 0  | 0  | -1 | -1 | -2 | 1  | 0  | 0  | 1  | 0  | 3  | 5  |    |    |    |    |   |    |    |  |  |
| M | Met | -5 | -2 | -1 | -2 | -1 | -3 | -2 | -3 | -2 | -1 | -2 | 0  | 0  | 6  |    |    |    |   |    |    |  |  |
| I | Ile | -2 | -1 | 0  | -2 | -1 | -3 | -2 | -2 | -2 | -2 | -2 | -2 | -2 | 2  | 5  |    |    |   |    |    |  |  |
| L | Leu | -6 | -3 | -2 | -3 | -2 | -4 | -3 | -4 | -3 | -2 | -2 | -3 | -3 | 4  | 2  | 6  |    |   |    |    |  |  |
| V | Val | -2 | -1 | 0  | -1 | 0  | -1 | -2 | -2 | -2 | -2 | -2 | -2 | -2 | 2  | 4  | 2  | 4  |   |    |    |  |  |
| F | Phe | -4 | -3 | -3 | -5 | -5 | -5 | -4 | -6 | -5 | -5 | -2 | -4 | -5 | 0  | 1  | 2  | -1 | 9 |    |    |  |  |
| Y | Tyr | 0  | -3 | -3 | -5 | -3 | -5 | -2 | -4 | -4 | -4 | 0  | -4 | -4 | -2 | -1 | -1 | -2 | 7 | 10 |    |  |  |
| W | Trp | -8 | -2 | -5 | -6 | -6 | -7 | -4 | -7 | -7 | -5 | -3 | 2  | -3 | -4 | -5 | -2 | -6 | 0 | 0  | 17 |  |  |
|   | C   | S  | T  | P  | A  | G  | N  | D  | E  | Q  | H  | R  | K  | M  | I  | L  | V  | F  | Y | W  |    |  |  |

the side group: (C) sulfhydryl (NDEQ) acid, acid amide and small hydrophobic, and (FYW) Log odds values (X 10 ) calculated (probability A.a pair will be found in alignment of two homologous proteins divided by the probability that ancestor probability is given, if the two probabilities are equal, -4 means (more by chance). Thus the probability is 10+10=20, whereas YY/TP is 10 between homologous sequences.

Amino acids are grouped according to to the chemistry of the side group: (C) sulfhydryl, (STPAG)-small hydrophilic, (NDEQ) acid, acid amide and hydrophilic, (HRK) basic, (MILV) small hydrophobic, and (FYW) aromatic. Each matrix value is Log odds values ( $\times 10$ ) calculated from odds scores: (probability A.a pair will be found in alignments of homologous proteins divided by probability that the pair will be found in alignment of unrelated proteins) : +10 means that ancestor probability is greater, 0 means that the probability are equal, -4 means that the change is random (more by chance). Thus the probability of alignment YY/YY is  $10+10=20$ , whereas YY/TP is  $-3-5=-8$ , a rare and unexpected between homologous sequences.

<sup>[1]</sup> M.O. Dayhoff: *Survey of new data and computer methods of analysis* (1978), Atlas of protein sequence and structure, **5:3**

# Blocks Amino Acid Substitution Matrices (BLOSUM)

- Use different strategy to estimate target frequencies
- Derived from alignments of domains of more diverse proteins (Henikoff & Henikoff, 1992)
- Occurrences of each amino acid pair in each column of each block alignment is counted
- Numbers derived from all blocks were used to compute the BLOSUM matrices



# The BLOSUM Matrix

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

BLOSUM is based on local alignments. Scans for A.a. in very conserved regions of protein families (that do not have gaps in the sequence alignment) /blocks are made and the relative frequencies of amino acids and their substitution probabilities counted. Then, log-odds score for each of the 210 possible substitutions of the 20 standard amino acids are calculated. Zero (0) score means freq of A.a. pair in the database is as expected by chance; **+ve** score means pair found more often than by chance; **-ve** means pair found less often than by chance

[2] S. Henikoff and J.G. Henikoff: *Amino acid substitution matrices from protein blocks* (1992), Proc. Natl. Acad. Sci., **89**:10915–10919s

# Comparing PAM and BLOSUM


## **PAM Approach :**

- Uses an evolutionary model
- Uses closely related sequences
- Extrapolates to greater distances

## **BLOSUM Approach:**

- Looks at more distantly related sequences
- Observes actual mutations in motifs (not extrapolations)
- Uses sets of different overall identity

# Comparing PAM and BLOSUM cont..



| PAM    | BLOSUM   |
|--------|----------|
| PAM100 | BLOSUM90 |
| PAM120 | BLOSUM80 |
| PAM160 | BLOSUM60 |
| PAM200 | BLOSUM52 |
| PAM250 | BLOSUM45 |

More distant sequences

- PAM 120: for general use
  - PAM 60: for close relations
  - PAM 250: for distant relations
  - BLOSUM 62: for general use
  - BLOSUM 80: for close relations
  - BLOSUM 45: for distant relations
- **BLOSUM** looks at mutations directly in motifs of more diverse sequences, whereas, **PAM** extrapolate evolutionary information based on a small set of related sequences

# Probability and Statistical analysis of Sequence Alignments

Evaluating Significance of an Alignment

# Alignment Scores

- **Odds scores:**
  - Chance of an aligned pair occurring in alignment of related sequences compared to the chance of that pair being found in random alignments of unrelated sequences.
  - **Evaluates alignment quality-** Whether most probable/reasonable matches have been attained.
  - **Limitation:** Can't address issue:
    - Whether or not the amino acids should actually align/ or could equally align that way by chance
    - Whether has a biological meaning

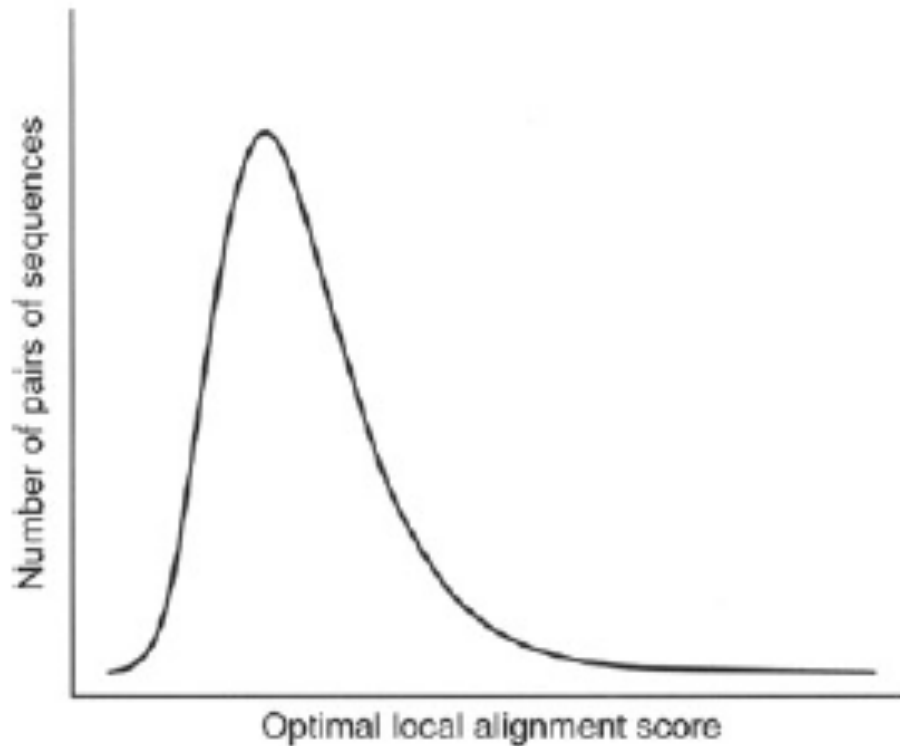
# Alignment Scores-2

- **Optimal alignment:**
  - Picked from the highest scoring.
  - **Confounder-** As an alignment between two protein sequences grows to include alignments of more related pairs, the score for the alignment increases, making the alignment more likely to be picked.

# Significance of Alignment.

- **Score distribution patterns:**
  - For simple phenomena (e.g. tossing a coin or rolling dice)- easy to calculate exactly the expected distribution of results and likelihood of any particular result.
  - **Challenge:** For sequences hard to define population distribution from which alignment is selected
  - **Practical approach:** If the score of the alignment observed is no better than might be expected from a random permutation (reshuffling) of the sequence, then likely it is due to chance

# Optimal score distribution



Optimal local alignment scores for pairs of random amino acid sequences of the same length

Scores follow not normal but an **extreme value distribution**. For any score  $X$  the probability of observing a score  $\geq X$  is:  $P(\text{score} \geq X) = 1 - \exp(-Ke^{-\lambda x})$ , where  $K$  and  $\lambda$  are parameters related to the position of maximum and width of the distribution. **Note long tail to the right**-implies that a score several SD above the mean has a higher probability of arising by chance (i.e. being as high, and yet less significant) than if the scores followed a normal distribution.

**Significance Test:** Take means and SD of the scores of randomized alignments, and ask whether score of the original sequence is unusually high (statistically different)

If randomized sequences score as well as the original one, the alignment is unlikely to be significant.



# Measures of Statistical Significance

## 1.) Z-score:

Extent to which original result is an outlier from the population

- $Z \text{ score} = \frac{\text{Score} - \text{Mean}}{\text{SD}}$  (if zero means observed similarity no better than random)

SD

## 2.) P-Value:

Probability that observed match is by chance

- Probability should be **very small e.g.  $P \ll 0.05$** , but the smaller the better

## 3.) E-value (for databases-see later:

The number of matches as good as the observed one, expected to appear by chance in a database of the size probed.

# Conclusion

- Dynamic programming algorithm can provide alignment of DNA or protein sequences
  - Entire length of sequence (Needleman-Wunsch)
  - Localized regions (Smith-Waterman)
- Finding best alignment depends on appropriate choices for scoring matrix and gap penalties
- Important to validate alignment by showing that there are no alternative alignments almost as good (in terms of both score and probability)
- List of some Sequence Alignment resources can be found at:  
[http://en.wikipedia.org/wiki/  
List of sequence alignment software](http://en.wikipedia.org/wiki/List_of_sequence_alignment_software)