

Lecture 07-09

Longest common subsequence problem

Dynamic Programming

Example,

S = ABAZDC

T = BACBAD

$M_{i,j} = \text{MAXIMUM} [ M_{i-1,j-1} + S, M_{i-1,j}, M_{i,j-1} ]$

Solution???

		B	A	C	B	A	D
		0	0	0	0	0	0
A	0	0	1	1	1	1	1
B	0	1	1	1	2	2	2
A	0	1	2	2	2	3	3
Z	0	1	2	2	2	3	3
D	0	1	2	2	2	3	4
C	0	1	2	3	3	3	4

Initialization  
Scoring  
Alignment

Multiple  
Possibilities

RECAP

## Longest common subsequence problem

### Dynamic Programming

Example,

$S = ABAZDC$

$T = BACBAD$

Space complexity:  $O(N^2)$

Time complexity:  $O(N^2)$

		B	A	C	B	A	D
		0	0	0	0	0	0
A	0	0	1	1	1	1	1
B	0	1	1	1	2	2	2
A	0	1	2	2	2	3	3
Z	0	1	2	2	2	3	3
D	0	1	2	2	2	3	4
C	0	1	2	3	3	3	4

RECAP

## Dynamic Programming

$S = BACBAD$

$T = ABAZDC$

Initialization

Matrix Fill

Score function

$$M_{i,j} = \text{Max} [ M_{i-1,j-1} + S_{i,j}; M_{i-1,j}; M_{i,j-1} ]$$

Traceback

$$Tb_{i,j} = \{ \nearrow; \uparrow; \leftarrow \}$$

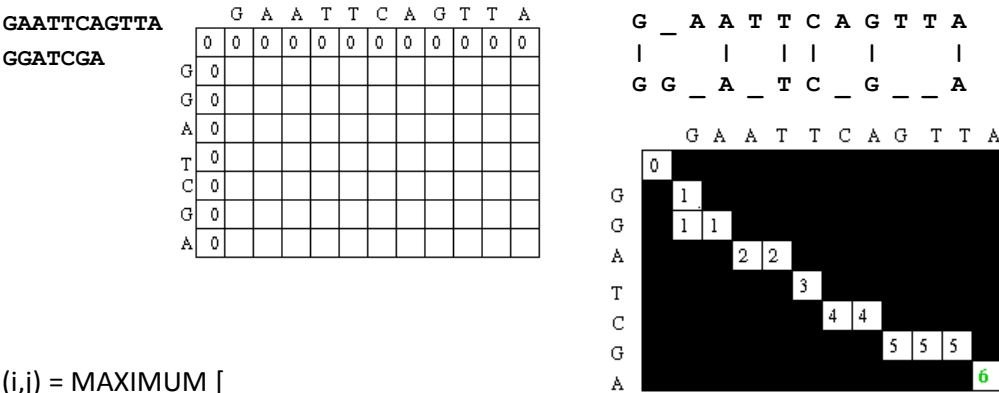
Gap Penalty:

Opening

Extension

Combined

# Pairwise Sequence Alignment



$M(i,j) = \text{MAXIMUM [}$   
     $M_{i-1,j-1} + S_{ij}$  (match/mismatch in the diagonal),  
     $M_{i,j-1} + w$  (gap in sequence #1),  
     $M_{i-1,j} + w$  (gap in sequence #2) ]

Workout

Source: <http://www.cse.cmu.edu/~burt/603/4spring/lec04.html>

# Pairwise Sequence Alignment

- Problem Statement
  - Input: Two sequences.  
    >Seq1 | Dummy | SEQUENCE  
    GAATTCAGTTA  
  
    >Seq2 | Dummy | SEQUENCE  
    GGATCGA
  - Output: Their alignment subject to optimum alignment score.

G \_ A A T T C A G T T A  
| \_ | | | | |  
G G \_ A \_ T C \_ G \_ \_ A

## Alignment Score Function

$$W(k) = c_{\text{open}} + c_{\text{length}} * k$$

$$M_{i,j} = \text{MAXIMUM} [ \\ M_{i-1,j-1} + S_{ij}, \\ M_{i-1,j-k} + w(k) \text{ (k = 1, ..., j-1)}, \\ M_{i-k,j-1} + w(k) \text{ (k = 1, ..., i-1)} ]$$

$$M_{i,j} = \text{MAXIMUM} [ \\ 0 \\ M_{i-1,j-1} + S(a_i, b_j), \\ M_{i-1,j} + w(a_i, -), \\ M_{i-k,j-1} + w(-, b_j) ]$$

## Global Pairwise Sequence Alignment

- Needleman-Wunch

$$W(k) = c_{\text{open}} + c_{\text{length}} * k$$

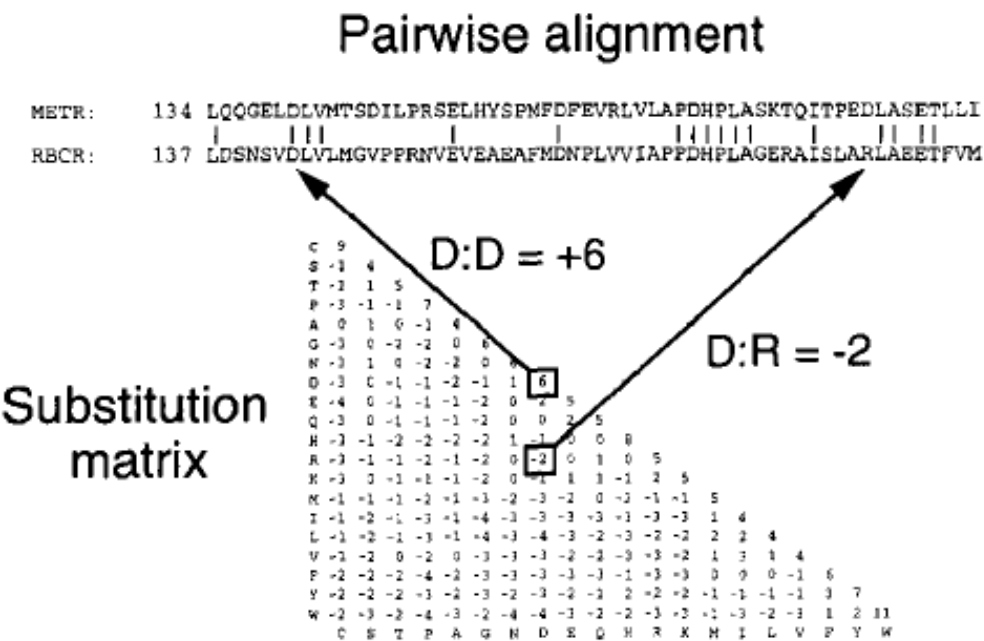
$$M(i,j) = \text{MAXIMUM} [ \\ M_{i-1,j-1} + S_{ij} \text{ (substitution matrix)}, \\ M_{i-1,j-k} + w(k) \text{ (k = 1, ..., j-1)}, \\ M_{i-k,j-1} + w(k) \text{ (k = 1, ..., i-1)} ]$$

# Local Pairwise Sequence Alignment

- Smith-Waterman

$$W(k) = C_{\text{open}} + C_{\text{length}} * k$$

$$M(i,j) = \text{MAXIMUM} [ \begin{aligned} &0 \\ &M_{i-1,j-1} + S(a_i,b_j) \text{ (match/mismatch),} \\ &M_{i-1,j} + w(a_i,-), \\ &M_{i-k,j-1} + w(-,b_j) \end{aligned} ]$$



# Substitution Matrix

	<b>BLOSUM62</b>																			
Ala	-																			
Arg	-1	5																		
Asn	-2	0	6																	
Asp	-2	-2	1	6																
Cys	0	-3	-3	-3	9															
Gln	-1	1	0	0	-3	5														
Glu	-1	0	0	2	-4	2	5													
Gly	0	-2	0	-1	-3	-2	-2	6												
His	-2	0	1	-1	-3	0	0	-2	8											
Ile	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	2	7		
Val	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4
	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val

## BLOSUM62

(**B**LOcks **S**Ubstitution **M**atrix)

$$S_{ij} = \left(\frac{1}{\lambda}\right) \log\left(\frac{p_{ij}}{q_i \times q_j}\right)$$

$p_{ij}$  is the probability of two amino acids  $i$  and  $j$  replacing each other in a homologous sequence,

$q_i$  and  $q_j$  are the background probabilities of finding the amino acids  $i$  and  $j$  in any protein sequence,

$\lambda$  is a scaling factor.

## (**B**locks **S**ubstitution **M**atrix)

Henikoff, S.; Henikoff, J.G. (1992). Amino Acid Substitution Matrices from Protein Blocks. *PNAS* **89** (22):10915-10919.

Eddy, S. R. (2004) Where did the BLOSUM62 alignment score matrix come from? *Nature Biotechnology* **22**:1035-1036.

Styczynski, M. P.; Jensen, K. L.; Rigoutsos, I.; Stephanopoulos, G. (2008). BLOSUM62 miscalculations improve search performance. *Nature Biotechnology* **26**:274 - 275.

## (**B**locks **S**ubstitution **M**atrix)

- BLOSUM matrices with high numbers are designed for comparing closely related sequences, while those with low numbers are designed for comparing distant related sequences.
- BLOSUM80 is used for less divergent alignments, and BLOSUM45 is used for more divergent alignments.
- The matrices were created by merging (clustering) all sequences that were more similar than a given percentage into one single sequence and then comparing those sequences (that were all more divergent than the given percentage value) only; thus reducing the contribution of closely related sequences.
- The percentage used was appended to the name, giving BLOSUM80 for example where sequences that were more than 80% identical were clustered.

# Point Accepted Mutation

- PAM is the replacement of a single amino acid in the primary structure of a protein with another single amino acid, which is accepted by the processes of natural selection.
- In particular, silent mutations are not point accepted mutations, nor are mutations which are lethal or which are rejected by natural selection in other ways.
- The calculation of PAM substitution matrix were based on 1572 observed mutations in the phylogenetic trees of 71 families of closely related proteins. The proteins to be studied were selected on the basis of having high similarity with their predecessors. The protein alignments included were required to display at least 85% identity.

## Dissimilar matrices can provide comparable performance with optimized gap penalties

	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W
	1	-1	-1	-1	-1	-1	0	-1	0	-1	-2	-2	0	-1	-1	0	-1	-1	-3	-4
		3	0	-1	0	0	0	0	-1	0	-1	-1	0	-1	-1	-1	-1	0	0	-1
C	13		3	-1	-1	-1	0	-1	-1	-1	-2	-1	-1	0	0	0	0	0	0	1
S	-1	5		2	-1	0	-1	0	-1	-1	-1	-2	0	-1	0	-2	-1	0	0	1
T	-1	2	5		3	0	-1	-2	-1	-1	-1	-1	-1	0	0	-1	0	-1	0	1
P	-4	-1	-1	10		1	0	-1	-2	-1	-1	-2	-1	1	0	0	-1	1	1	1
A	-1	1	0	-1	5		3	0	-1	-1	0	-1	-1	0	0	-1	-1	-1	-1	0
G	-3	0	-2	-2	0	8		3	-1	-1	-1	-2	-1	-1	0	0	-1	-1	0	0
N	-2	1	0	-2	-1	0	7		2	0	0	0	0	0	-1	0	-1	1	1	1
D	-4	0	-1	-1	-2	-1	2	8		4	0	-1	0	1	-1	0	-1	-1	1	2
E	-3	-1	-1	-1	-1	-3	0	2	6		4	-1	-1	0	-2	-1	-2	-1	0	-2
Q	-3	0	-1	-1	-1	-2	0	0	2	7		2	0	0	-2	-1	-1	0	1	-1
H	-3	-1	-2	-2	-2	-2	1	-1	0	1	10		3	-1	-1	-1	-1	-1	0	1
R	-4	-1	-1	-3	-2	-3	-1	-2	0	1	0	7		3	0	0	-1	-2	0	0
K	-3	0	-1	-1	-1	-2	0	-1	1	2	0	3	6		1	-1	1	-1	0	-1
M	-2	-2	-1	-3	-1	-3	-2	-4	-2	0	-1	-2	-2	7		1	-1	-1	-1	-1
I	-2	-3	-1	-3	-1	-4	-3	-4	-4	-3	-4	-4	-3	2	5		2	-1	0	0
L	-2	-3	-1	-4	-2	-4	-4	-4	-4	-3	-2	-3	-3	3	2	5		1	-1	-3
V	-1	-2	0	-3	0	-4	-3	-4	-3	-3	-4	-3	-3	1	4	1	5		0	-2
F	-2	-3	-2	-4	-3	-4	-4	-5	-3	-4	-1	-3	-4	0	0	1	-1	8		1
Y	-3	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-1	-2	0	-1	-1	-1	4	8	
W	-5	-4	-3	-4	-3	-3	-4	-5	-3	-1	-3	-3	-3	-1	-3	-2	-3	1	2	15
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W