Dept. of Computer Science and Engineering
University of Rajshahi
www.ru.ac.bd

Dr. Shamim Ahmad

"It's a Fact"

Sequence comparisons, which are based on evolutionary theory, are the foundation of bioinformatics

Alignments tell us about...

- Function or activity of a new gene/protein
- Structure or shape of a new protein
- Location or preferred location of a protein
- Stability of a gene or protein
- Origin of a gene, protein, organelle, organism…

Similarity versus Homology

- Similarity refers to the likeness or % similarity between 2 sequences
- Similarity of sequences usually means sharing a statistically measured number of bases or amino acids
- Similarity does not necessarily imply homology

- Homology refers to shared ancestry
- Two sequences are homologous if they are derived from a common ancestral sequence
- Homology often implies similarity
  (note that structural, but not sequence, similarity may occur)

Similarity versus Homology

- Similarity can be quantified
- It is correct to say that two sequences are X% identical
- It is correct to say that two sequences have a similarity score of Z
- It is correct to say that two sequences are X% similar, *as long as the criteria for similarity is clear.*

Similarity versus Homology

- Homology cannot be quantified
  "Its homologous or it isn't"
- If two sequences have a high % identity it is OK to say they are homologous
- It is incorrect to say two sequences have a homology score of Z
- It is incorrect to say two sequences are X% homologous or have a homology of X %

Similarity by chance – the impact of sequence complexity

MCDEFGHIKLAN…. High Complexity

ACTGTCACTGAT…. Mid Complexity

NNNNTTTTTNNN…. Low Complexity

Low complexity sequences are more likely to appear similar by chance

Can you think of examples of low complexity sequences that in Nature? Perhaps encoding certain structural features?

Example of *homology* but *little sequence similarity*: The N-terminal domain of OprF and OmpA share only 15% identity but are homologous

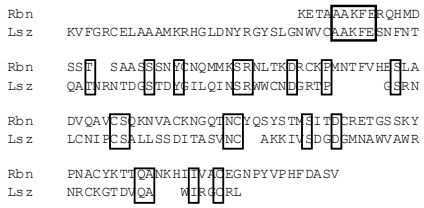
```
OprF   1 -QGQNSVEIEAFGKRYFTDSVRNMKN-------ADLYGGSIGYFLTDDVELALSYGEYH
OmpA   1 APKDNTWYTGAKLGWSQYHDTGLINNNGPTHENKLGAGAFGGYQVNPYVGFEMGYDWLG
             *    *              *           *   **     *    *

OprF  52 DVRGTYETGNKKVHGNLTSLDAIYHFGTPGVGLRPYVSAGLA-HQNITNINSDSQGRQQ
OmpA  60 RMPYKGSVENGAYKAQGVQLTAKLGYPIT-DDLDIYTRLGGMVWRADTYSNVYGKNHDT
                 *         * *          *  *  *       *  *

OprF 110 MTMANIGAGLKYYFTENFFAKASLDGQYGLEKRDNGHQG--EWMAGLGVGFNFG
OmpA 118 GVSPVFAGGVEYAITPEIATRLEYQWTNNIGDAHTIGTRPDNGMLSLGVSYRFG
                 *  *  *                          *  ***   **
```

Some Simple (but not Hardfast) Guiding Rules

After low complexity sequences are considered…

- If two sequence are > 200 residues and > 25% identical, they are likely related
- If two sequences are 15-25% identical they may be related, but more tests are needed
- If two sequences are < 15% identical they are most likely not related (but not always!)
- If you need more than 1 gap for every 20 residues the alignment is suspicious

Assessing Sequence Similarity



is this alignment significant?

Sequence Alignment - Methods

- **Dot Plots**
- **Dynamic Programming**
- **Heuristic (Approx. but Fast) Local Alignment – FASTA and BLAST**
- **Multiple Sequence Alignment**

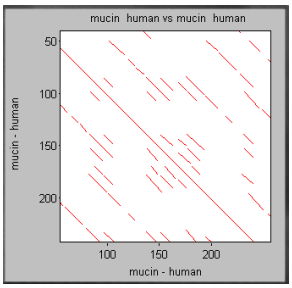
Dot Plots

- "Invented" in 1970 by Gibbs & McIntyre
- Good for quick graphical overview – any size of sequence
- Simplest method for sequence comparison
- Inter-sequence comparison
- Intra-sequence comparison
  - ✓ Identifies internal repeats
  - ✓ Identifies domains or "modules"

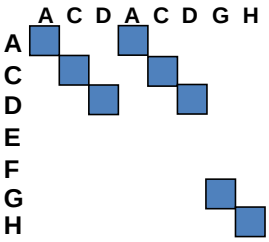
Dot Plot Algorithm

- *Take two sequences (A & B), write sequence A out as a row (length=m) and sequence B as a column (length =n)*
- *Create a table or "matrix" of "m" columns and "n" rows*
- *Compare each letter of sequence A with every letter in sequence B. If there's a match mark it with a dot, if not, leave blank*

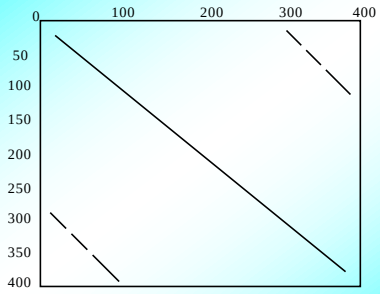
Dot Plots & Internal Repeats



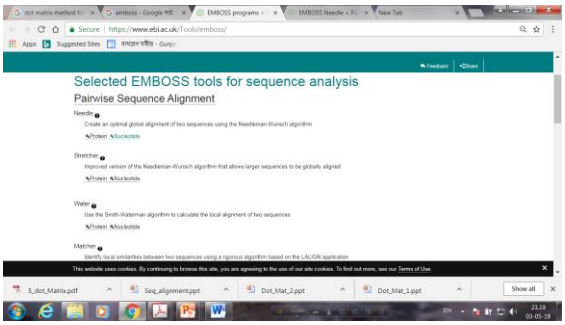
Dot Plot Algorithm
Direct repeats

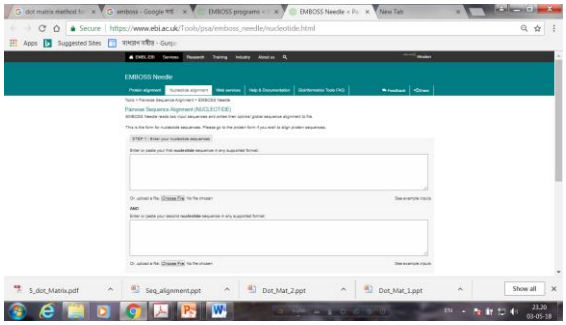


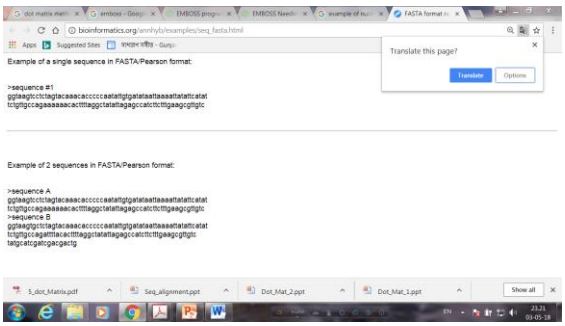
PLEK_HUMAN (horizontal) vs. PLEK_HUMAN (vertical)

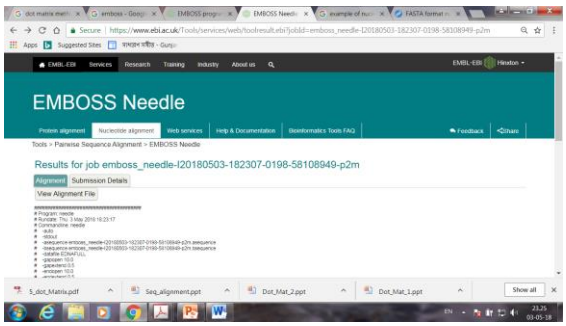


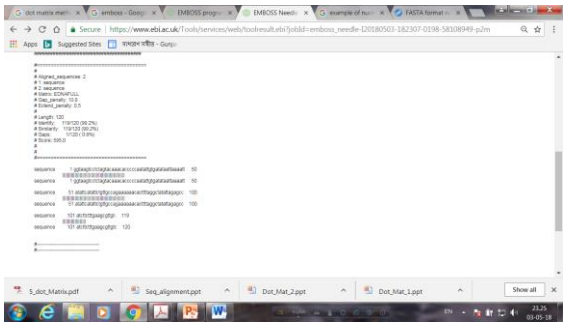
A dot plot of human pleckstrin sequence against itself produced with Erik Sonnhammer's 'dotter' program. The sequence is plotted from N- to C- terminus along horizontal and vertical axes between residues 1 and approximately 350.

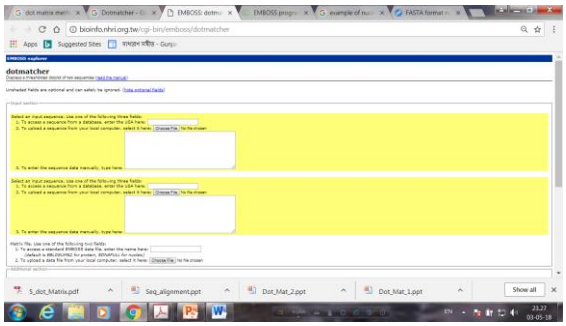

Selected EMBOSS tools for sequence analysis
Pairwise Sequence Alignment
Needle
Stretcher
Water
Matcher

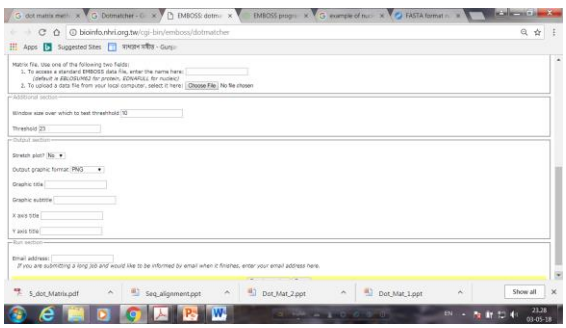

EMBOSS Needle

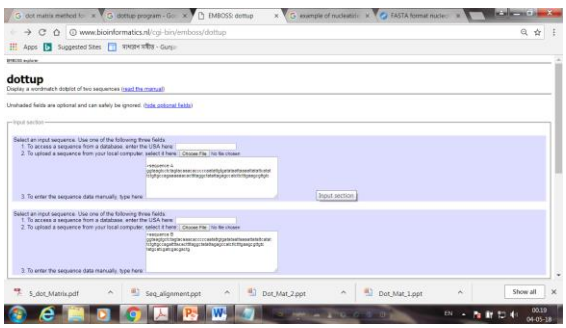

Translate this page?
Translate
Options
Example of a single sequence in FASTA/Pearson format:
Example of 2 sequences in FASTA/Pearson format:

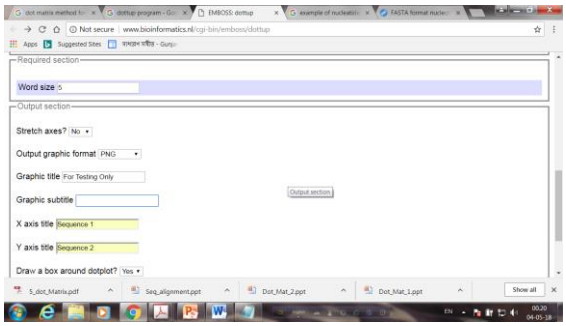

EMBOSS Needle
Results for job emboss_needle-I20180503-182307-0198-58108949-p2m
Alignment
Submission Details
View Alignment File

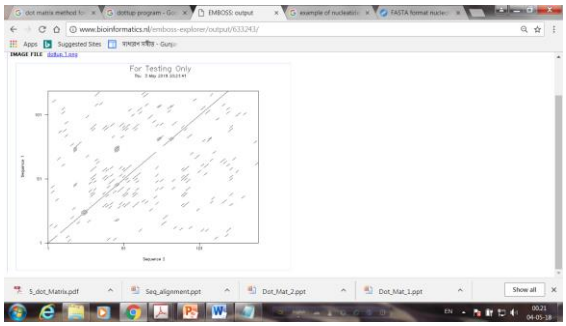


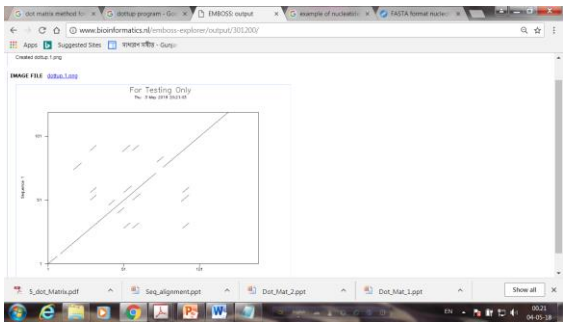


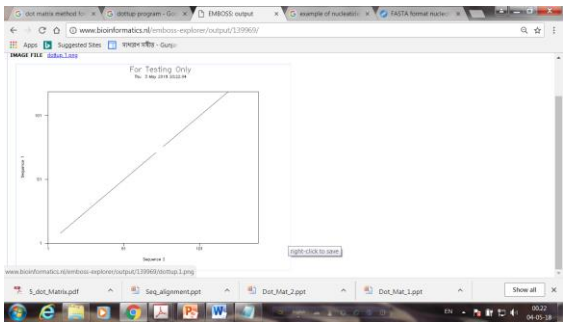





Required section
Word size 5
Output section
Stretch axes? No
Output graphic format PNG
Graphic title For Testing Only
Graphic subtitle
X axis title Sequence 1
Y axis title Sequence 2
Draw a box around dotplot? Yes
Output section


For Testing Only


For Testing Only

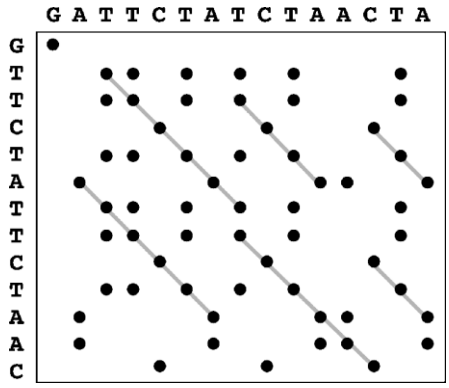

For Testing Only
right-click to save

```
seq1   EARDF-NQYYSSIKRSGSIQ
         . : .:::::::::. . .
seq2   LPKLFIDQYYSSIKRTMG-H
```

global sequence alignment

```
seq1   NQYYSSIKRS
       .:::::::::.
seq2   DQYYSSIKRT
```

local sequence alignment



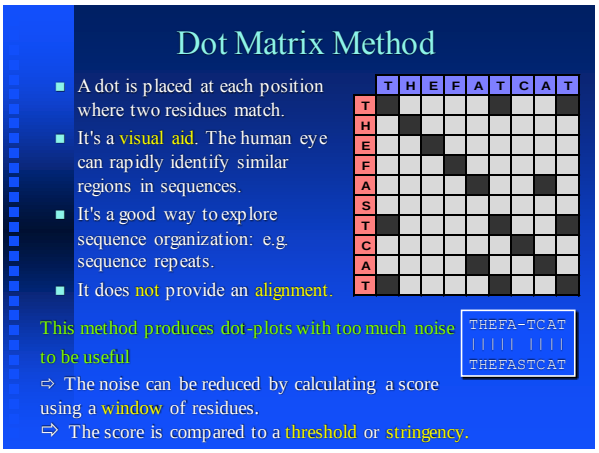
Identity

Identity defines the percentage of amino acids (or nucleotides) with a direct match in the alignment.

Similarity

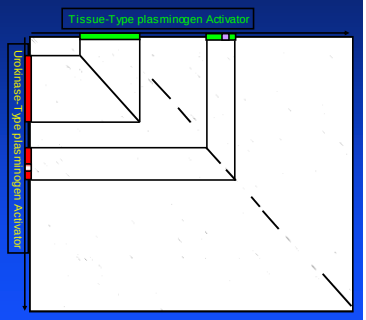
When one amino acid is mutated to a similar residue such that the physiochemical properties are preserved, a conservative substitution is said to have occurred.

- For example, a change from a**rginine** to l**ysine** maintains the +1 positive charge



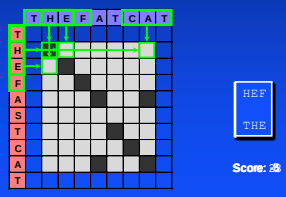
Dot Matrix Representation

- Produces a graphical representation of similarity regions
- The horizontal and vertical dimensions correspond to the compared sequences
- A region of similarity stands out as a diagonal



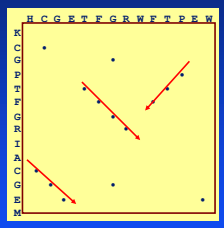
Dot Matrix or Dot-plot

- Each window of the first sequence is aligned (without gaps) to each window of the 2nd sequence
- A colour is set into a rectangular array according to the score of the aligned windows

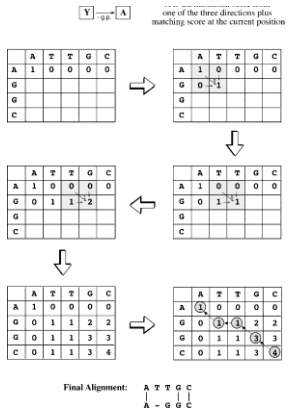


Dot Matrix Display

- Diagonal rows (↘) of dots reveal sequence similarity or repeats.
- Anti-diagonal rows (↙) of dots represent inverted repeats.
- Isolated dots represent random similarity.



Dynamic Programming Method



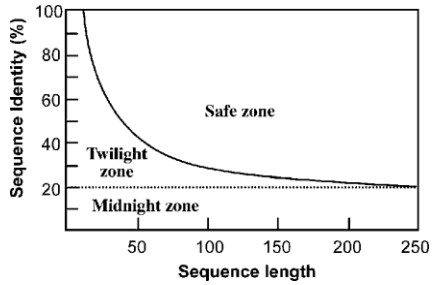


Table 2 - 3. IUB letter codes of amino acids.
(1) assuming the standard genetic code.

1-letter	3-letter	Meaning	Codon(1)
A	Ala	Alanine	GCT,GCC,GCA,GCG
B		Asp or Asn	GAT,GAC,AAT,AAC
C	Cys	Cysteine	TGT,TGC
D	Asp	Aspartic	GAT,GAC
E	Glu	Glutamic	GAA,GAG
F	Phe	Phenylalanine	TTT,TTC
G	Gly	Glycine	GGT,GGC,GGA,GGG
H	His	Histidine	CAT,CAC
I	Ile	Isoleucine	ATT,ATC,ATA
K	Lys	Lysine	AAA,AAG
L	Leu	Leucine	TTG,TTA,CTT,CTC,CTA,CTG
M	Met	Methionine	ATG
N	Asn	Asparagine	AAT,AAC
P	Pro	Proline	CCT,CCC,CCA,CCG
Q	Gln	Glutamine	CAA,CAG
R	Arg	Arginine	CGT,CGC,CGA,CGG,AGA,AGG
S	Ser	Serine	TCT,TCC,TCA,TCG,AGT,AGC
T	Thr	Threonine	ACT,ACC,ACA,ACG
V	Val	Valine	GTT,GTC,GTA,GTG
W	Trp	Tryptophan	TGG
X	Xxx	Unknown	
Y	Tyr	Tyrosine	TAT,TAC
Z		Glu or Gln	GAA,GAG,CAA,CAG
*	End	Terminator	TAA,TAG,TGA

The Principle of Parsimony in Phylogeny

Infer relationships between species. It states that the tree with the fewest common ancestors is the most likely.

Example

- Four species
  - All of which have wings
  - But only three of which can hover while flying. The most parsimonious possible model
- All four species have one ancestor
- The second trait,
  - Three species that hover have a common ancestor
  - Two different evolutionary paths.

Construction of Substitution matrices

- **BLOSUM**
  - **BLO**CKS **SU**BSTITUTION **M**ATRIX
- PAM
  - **P**OINT **A**CCEPTED **M**UTATIONS

Substitution matrices

- Substitution matrix contains values proportional to the probability that amino acid A mutates into amino acid B for all pairs of amino acids through a period of evolution

- Substitution matrices are constructed from a large and diverse sample of sequence alignments

How to construct substitution matrices ?

- Tabulate substitutions
  - A to A: 9867 times
  - A to R: 2 times
  - A to N: 9 times
  - etc.…

How to construct substitution matrices ?

MUTATION RATES

	A	R	N	D	C	Q	E	G	H
A	9867	2	9	10	3	8	17	21	2
R	1	9913	1	0	1	10	0	0	10
N	4	1	9822	36	0	4	6	6	21
D	6	0	42	9859	0	6	53	6	4
C	1	1	0	0	9973	0	0	0	1
Q	3	9	4	5	0	9876	27	1	23
E	10	0	7	56	0	35	9865	4	2
G	21	1	12	11	1	3	7	9935	1
H	1	8	18	3	1	20	1	0	9912
I	2	2	3	1	2	1	2	0	0
...	...	...	...	...	...	...	...	...	...

How to construct substitution matrices ?

Substitution matrix score =

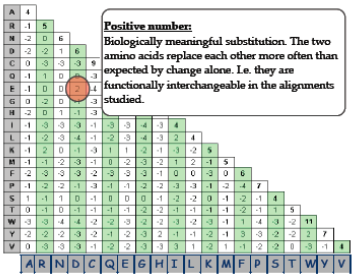
Log $\frac{\text{Observed mutation rate in alignment}}{\text{Expected random mutation rate}}$

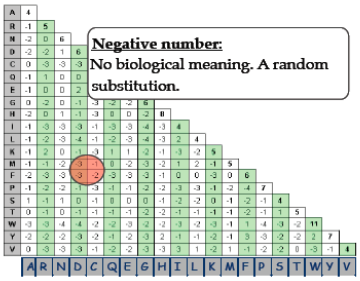
The random mutation rate

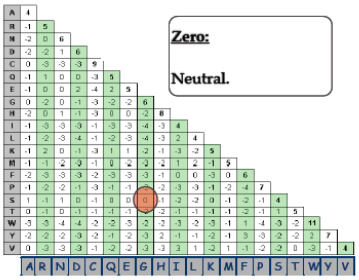
Example:

Expected random mutation rate is 1 in 10000 and observed mutation rate of W to R is 1 in 10

Score = log (0.1/0.0001) = log (1000) = +3







Substitution matrices

- Protein sequences are more complex
  - matrices = collection of scoring rules
- Matrices over events such as
  - mismatch and perfect match
- Need to define gap penalty separately
- E.g. **BLO**cks **SU**bstitution **M**atrix **(BLOSUM)**

BLOSUM-x matrices

- Constructed from aligned sequences with specific **x% similarity**
  - matrix built using sequences with no more then 50% similarity is called **BLOSUM-50**
- For highly mutating / dissimilar sequences use
  - BLOSUM-45 and lower
- For highly conserved / similar sequences use
  - BLOSUM -62 and higher

BLOSUM 62

	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	-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	

- What diagonal represents? **perfect match between a.a.**
- What is the score for substitution E→D (acid a.a.)? **Score = 2**
- More drastic substitution K→I (basic to non-polar)? **Score = -3**

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

	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W
C	12																			
S	0	2																		
T	-2	1	3																	
P	-3	1	0	6																
A	-2	1	1	1	2															
G	-3	1	0	-1	1	5														
N	-4	1	0	-1	0	0	2													
D	-5	0	0	-1	0	1	2	4												
E	-5	0	0	-1	0	0	1	3	4											
Q	-5	-1	-1	0	0	-1	1	2	2	4										
H	-3	-1	-1	0	-1	-2	2	1	1	3	6									
R	-4	0	-1	0	-2	-3	0	-1	-1	1	2	6								
K	-5	0	0	-1	-1	-2	1	0	0	1	0	3	5							
M	-5	-2	-1	-2	-1	-3	-2	-3	-2	-1	-2	0	0	6						
I	-2	-1	0	-2	-1	-3	-2	-2	-2	-2	-2	-2	-2	2	5					
L	-6	-3	-2	-3	-2	-4	-3	-4	-3	-2	-2	-3	-2	4	2	6				
V	-2	-1	0	-1	0	-1	-2	-2	-2	-2	-2	-2	-2	2	4	2	4			
F	-4	-3	-3	-5	-4	-5	-4	-6	-5	-5	-2	-4	-5	0	1	2	-1	9		
Y	0	-3	-3	-5	-3	-5	-2	-4	-4	-4	0	-4	-4	-2	-1	-1	-2	7	10	
W	-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

Simple Scoring Rule

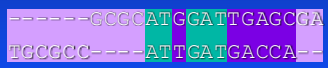
Score each position independently:

- Match: +1
- Mismatch: -1
- Indel: -2

Score of an alignment is sum of position scores

Example

Score: $(+1\times13) + (-1\times2) + (-2\times4) = 3$



Score: $(+1\times5) + (-1\times6) + (-2\times11) = -23$

More General Scores

- The choice of +1,-1, and -2 scores is quite arbitrary
- Depending on the context, some changes are more plausible than others
  - Exchange of an amino-acid by one with similar properties (size, charge, etc.) vs.
  - Exchange of an amino-acid by one with opposite properties
- Probabilistic interpretation: How likely is one alignment versus another ?

Global alignment vs Local alignment

- Global alignment is attempting to match as much of the sequence as possible.
  The tool for Global alignment is based on Needleman-Wunsch algorithm.
- Local alignment is to try to find the regions with highest density of matches. The tool for local alignment is based on Smith-Waterman.
- Both algorithms are derivates from the basic dynamic programming algorithm.

```
L G P S S K Q T G K G S - S R I W D N

                                    Global alignment
L N - I T K S A G K G A I M R L G D A


- - - - - - - T G K G - - - - - - - -

                                    Local alignment
- - - - - - - A G K G - - - - - - - -
```


Local alignment illustration (2 of 2)

		G	G	C	T	C	A	A	T	C	A
	0	0	0	0	0	0	0	0	0	0	0
A	0	0	0	0	0	0	2	2	0	0	2
C	0	0	0	2	0	2	0	1	1	2	0
C	0	0	0	2	1	2	1	0	0	3	1
T	0	0	0	0	4	2	1	0	2	1	2
A	0	0	0	0	2	3	4	3	1	1	3
A	0	0	0	0	0	1	5	6	4	2	3
G	0	2	2	0	0	0	3	4	5	3	1
G	0	2	4	2	0	0	1	2	3	4	2

Local alignment illustration (3 of 3)

		G	G	C	T	C	A	A	T	C	A
	0	0	0	0	0	0	0	0	0	0	0
A	0	0	0	0	0	0	2	2	0	0	2
C	0	0	0	2	0	2	0	1	1	2	0
C	0	0	0	1	1	2	1	0	0	3	1
T	0	0	0	0	4	2	1	0	2	1	1
A	0	0	0	0	2	3	4	3	1	1	3
A	0	0	0	0	0	1	5	6	4	2	3
G	0	2	2	0	0	0	3	4	5	3	1
G	0	2	4	2	0	0	1	2	3	4	2

Best score: 6

CTCAA
CT-AA
locally

GGCTCAATCA
ACCT-AAGG
in the whole seq. context (globally)

An example of aligning text strings

Raw Data ???

T C A T G
C A T T G

2 matches, 0 gaps

T C A T G
| |
C A T T G

3 matches (2 end gaps)

T C A T G .
| | |
. C A T T G

4 matches, 1 insertion

T C A - T G
| | | |
. C A T T G

4 matches, 1 insertion

T C A T - G
| | | |
. C A T T G

An example: scoring a sequence alignment with a gap penalty

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

Note: 1. it is likely to have non-identical amino acids placed in the corresponding positions.

2. Scores gained by each match are not always the same, for instance two rare amino acids will score more than two common.

3. The alignment gap(s) may be introduced for optimising the score. Introduction of gaps causes penalties.

Steps for the dynamic programming algorithm

1. Score of new alignment = Score of previous alignment (A) + Score of new aligned pair

V	D	S	-	C	Y		V	D	S	-	C		Y	
V	E	S	L	C	Y		V	E	S	L	C		Y	
					15	=					8	+		7

2. Score of alignment (A) = Score of previous alignment (B) + Score of new aligned pair

V	D	S	-	C			V	D	S	-			C	
V	E	S	L	C			V	E	S	L			C	
					8	=					-1	+		9

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

Are these proteins homologs?

```
SEQ 1: RVVNLVPS--FWVLDATYKNYAINYNCDVTYKLY
          L P    W L   Y N    Y C     L
SEQ 2: QFFPLMPPAPYWILATDYENLPLVYSCTTFFWLF
```

NO (score = 9)

```
SEQ 1: RVVNLVPS--FWVLDATYKNYAINYNCDVTYKLY
          L P    W LDATYKNYA  Y C     L
SEQ 2: QFFPLMPPAPYWILDATYKNYALVYSCTTFFWLF
```

MAYBE (score = 15)

```
SEQ 1: RVVNLVPS--FWVLDATYKNYAINYNCDVTYKLY
       RVV L PS   W LDATYKNYA  Y CDVTYKL
SEQ 2: RVVPLMPSAPYWILDATYKNYALVYSCDVTYKLF
```

YES (score = 24)