# CHAPTER 4: PRODUCING AND ANALYZING SEQUENCE ALIGNMENTS

Dr. Garrett Dancik

### Motivation

- You have recently sequenced a gene and its CDS begins with
  - GGCGGAGCCAGGCCGGCCTAGAGTCACTTCTCC
- You have isolated a protein and its amino acid sequence is
  - MGKEIPTDAPWEAQHADKWDKMTMKELIDKICWTKTA
- Questions:
  - What does this protein do?
  - What are the important functional regions?
  - Do other organisms have similar genes or proteins?
- To answer these questions we can find similar sequences, identified through sequence alignments, using tools such as BLAST

### Sequence alignment

- Two sequences should be aligned in such a way that maximizes their *similarity*
  - If they derive from a common ancestor, characters (bases or amino acids) derived from the same ancestral base should be aligned
  - Shared domains in proteins (and important regions in nucleotide sequences) should align, even if the sequences are not similar overall
- Alignment should take into account biological mutations and other events
  - Point mutations
  - Insertions or deletions (indels)
  - Gene duplications and pseudogenes (a gene copy that does not produce a functional protein)
    - The human genome has up to 20,000 pseudogenes!

#### Sequence alignment example

Consider the alignment of two hypothetical protein sequences:

THISSEQUENCE and THATSEQUENCE



#### Sequence alignment example (different lengths)

• Now consider the alignment of two hypothetical protein sequences:

#### THATSEQUENCE and THISISASEQUENCE,

where the amino acids I, S, and A were inserted into one of the original sequences

Т	H	A	$\mathbf{T}$	S	Ε	Q	U	Е	Ν	С	Е			
т	H	I	S	I	S	Α	S	Ε	Q	U	Ε	N	С	Ε

- When aligning both sequences from the beginning
  - similarity which is obvious to us is lost
  - false matches are created because of differences in length

#### Sequence alignment example (different lengths)

 The solution is to introduce a gap, which corresponds to an insertion or a deletion and is usually indicated by a dash (-) in an alignment

Т	H	Ι	S	Ι	S	A	—	S	Ε	Q	U	Ε	N	С	E
L															
T	H	-	-	-	-	Α	$\mathbf{T}$	S	Е	Q	U	Е	N	С	Е

- There are always multiple possible alignments, and the best alignment is not always obvious
- The alignment must be selected using a quantitative scoring measure

# Sequence homology

- **Similarity** is a descriptive term indicating that two or more sequences have a certain degree of identity or likeness
- Homologous sequences (or homologues) are sequences that are descended from a common ancestor
- Homologous genes will accumulate different mutations (divergent evolution) during the course of evolution and their sequences are often not identical.
- **Convergent evolution** is when sequences with high similarity are not homologous
- Sequences with high similarity are *inferred* to be homologous
  - But homologous sequences may not have high similarity





# Homology is more easily detected from protein sequences

- Number of possible characters in nucleotides vs. proteins?
- Matches in nucleotide sequences are more likely due to chance than matches in protein sequences
- The genetic code is redundant
  - Identical amino acid sequences can be encoded by different nucleotide sequences
  - Nucleotide sequences are more likely to change over time
- Structure and function of a protein is determined by its amino acid sequence (although this is determined by the nucleotide sequence)

#### Scoring alignments

- Since multiple alignments are always possible, the best possible alignment is determined based on an alignment score
  - The **optimal alignment** is the alignment with the best score (multiple optimal alignments are possible)
  - Suboptimal alignments have slightly less scores than the best one
- The percentage or percent identity of an alignment is equal to the number of identical matches in an alignment divided by the length of the alignment (including gaps)



 The above alignment is optimal and has a percent identity of 11/16 = 68.75%

# **Dot-plots**

- A dot-plot displays the alignment of two sequences and visualizes sequence similarity graphically
- A dot indicates identity between characters of each sequence
- Interruptions along the diagonal indicate a gap
- In addition to visualizing overall similarity, dot-plots can indicate intrasequence repeats



#### Dot-plots and background noise

- A. Dot-plot of an SH2 domain with itself
- B. The same dot-plot but with background noise removed, based on a window of 10 residues and a minimum identity score within each window of 3



#### Dot-plots showing BRCA2 repeat domain

Background is removed using a window of 30 and a minimum score of 5



# Similarity versus identity

- Genuine matches do not have to be identical
- Certain non-identical amino acids may have
  - Similar physical and chemical properties
  - May be more likely to be present at the same region than others in related sequences
- Percent similarity is calculated in the same way as percent identity but both identical and similar matches are considered



- Isoleucine (I) and alanine (A) are hydrophobic; serine (S) and threonine (T) are polar
- Percent similarity is 12/15 = 80%

#### Substitution matrices

- For protein sequences, the score for each aligned pair of amino acids is determined by a substitution matrix, which has values for all possible pairs of residues.
- Example using BLOSUM-62 matrix:

Seq1:	Т	Η	Ι	S	S	Ε	Q	U	Ε	Ν	С	Ε
Seq2:	т	Η	A	т	S	Ε	Q	U	Ε	Ν	С	Ε
Score:	5	8	-1	1	4	5	5	0	5	6	9	5

#### This alignment has an overall score (S) of 52

\/ <del>~</del> /	_												RI (	OS	UJN	1-6	2 m	natr	ix		
С	9		small and polar residues																		
S	-1	4																			
Т	-1	1	5																		
Ρ	-3	-1	-1	7																	
А	0	1	0	-1	4	4 small and nonpolar															
G	-3	0	-2	-2	0	6															
Ν	-3	1	0	-2	-2	0	6			nola	ar or	acid	lic re	npise	165						
D	-3	0	-1	-1	-2	-1	1	6		pore		aora		5100							
Е	-4	0	-1	-1	-1	-2	0	2	5												
Q	-3	0	-1	-1	-1	-2	0	0	2	5											
Н	-3	-1	-2	-2	-2	-2	1	-1	0	0	8		b	asic							
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5									
К	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5								
М	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5		la	rge a	ind			
Т	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4	hy	ydrop	ohob	İC		
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	a	romat	tic
Υ	-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	
	С	S	Т	Ρ	А	G	Ν	D	Е	Q	Н	R	Κ	Μ	I	L	V	F	Υ	W	

### Substitution matrices

#### BLOSUM matrices

- BLOck SUbstitution Matrix
- Based on local alignments to detect conserved short regions
- Sequences grouped based on percent identity, where the percent identify threshold for grouping determines the specific BLOSUM matrix
  - BLOSUM-62 is based on grouping aligned sequences with no more than 62% identity
- Substitution frequencies are then calculated
- Positive scores indicate conservative (more likely) substitutions
- Negative scores indicate non-conservative (less likely) substitutions
- All BLOSUM matrices are based on observed alignments

#### Substitution matrices

- Point Accepted Mutation (PAM) matrices
  - Based on amino acid frequencies in alignment of similar (>85% identical) and homologous protein sequences
  - Probabilities were calculated for whether a given amino acid mutates to any other over a given period of time
  - The logarithm of this probability gives the substitution score
  - Based on number of changes from each amino acid and total number of occurrences
  - There are multiple PAM matrices and the PAM # corresponds to the number of accepted point mutations per 100 residues.
  - All PAM matrices are based on PAM1; others are inferred.
  - For example, the PAM250 contains scores based on an expected evolutionary distance corresponding to 250 point accepted mutations for every 100 amino acid residues

#### PAM vs. BLOSUM Substitution matrices

- Choice depends on evolutionary distance
- For closely related sequences
  - Use higher BLOSUM number and lower PAM number
- For distantly related sequences
  - Use lower BLOSUM number or higher PAM number

## **Inserting Gaps**

- A **gap** in a sequence alignment indicates an insertion or deletion in the sequence
- When a gap is introduced, a gap opening penalty is added to the score
  - Insertions and deletions are not likely to occur in regions of structural importance
- Insertions tend to be several residues long
  - A smaller gap extension penalty is added each time a gap is extended
- Gaps cannot be aligned with each other

#### **Gap Penalties**

- A "gap" (composed of a sequence of gap characters in the alignment, e.g., ---) has a penalty composed of a gap opening penalty for the initial character of each gap and a gap extension penalty for each subsequent character. Typically gaps are not penalized if they occur at the beginning or end of the alignment (this is known as a semi-global alignment)
- Here we use a gap opening penalty of 10 and a gap extension penalty of 1



This semi-global alignment has an overall score (S) of -5

## Types of alignments

- A (**semi**) **global alignment** aligns two sequences across their entire lengths
  - Appropriate for homologous sequences
- A local alignment detects shared regions (e.g., domains) which may be missed in global alignments
- A pairwise alignment is the alignment of two sequences
- A **multiple alignment** is the simultaneous alignment of more than two sequences

### Many proteins have multiple domains



(A) local

PI3-kinase	DRHNSNIMVKDDGQLFHIDFG
cAMP PK	DLKPENLLIDQQGYIQVTDFG

# Local and global alignments

global (B) PI3-kinase HQLGNLR--LEECRI---MSSAKRPLWLNWENPDIMSELLFQNNEIIFKNGDDLRQDMLT CAMP PK GNAAAAKKGXEQESVKEFLAKAKEDFLKKWENPAQNTAHLDQFERIKTLGTGSFGRVML-PI3-kinase LQIIRIME--NIWQNQGLDLRMLPYGCLSIGDCVGLIEVVRNSHTIMQ-IQCKGGLKGAL CAMP PK ---VKHMETGNHYAMKILDKQKVVK-----LKQIEHTLNEKRILQAVNFPFLVKLEF PI3-kinase QFNSHT-LHQWLKDKNKGEIYDAA--IDLFTRSCAGYCVATFILGIGDRHNSNIMVKD-D CAMP PK SFKDNSNLYMVMEYVPGGEMFSHLRRIGRFSEPHARFYAAQIVLTFEYLHSLDLIYRDLK PI3-kinase GQLFHIDFGHFLDHKKKKFGYKRERVP----FVLTQDFL---IVISKGAQECTKTREFE CAMP PK PENLLIDQQGYI--QVTDFGFAK-RVKGRTWXLCGTPEYLAPEIILSKGYNKAVDWWALG 

# Alignment algorithms (preview)

- Needleman-Wunsch (1970) and variations:
  - for aligning two sequences
  - uses dynamic programming to "consider" all possible alignments (there are 10<sup>600</sup> possible alignments for two sequences of length 1000; there are only 10<sup>80</sup> atoms in the known universe)
- FASTA: uses a heuristic method for efficient searches (though not guaranteed to find the optimal solution)
  - Creates dictionary of k-tuples for the query sequence which is checked against sequences in the database
  - A local alignment algorithm is used to complete the alignment
- BLAST (Basic Local Alignment Search Tool): also fast and uses a heuristic
  - Finds short matches (which do not have to be exact)
  - Then uses local alignment to complete the alignment