Pairwise Sequence Alignment

Manu Madhavan

Lecture 5

Outline

- What is Sequence Alignment
- Importance of Sequence Alignment
- Global and Local Alignment
- Algorithms for Global Alignment

Sequence Alignment

Biological Problem

Sequence alignment is a way of arranging protein (or DNA) sequences to identify regions of similarity that may be a consequence of evolutionary relationships between the sequences.

- Genome sequencing allows comparison of organisms at DNA and protein levels
- Comparisons can be used to
  - Find evolutionary relationships between organisms
  - Identify functionally conserved sequences
  - Identify corresponding genes in human and model organisms: develop models for human diseases

Sequence Homology

- **Homology**: genes that derive from a common ancestor-gene are called homologs
- **Orthologous** genes are homologous genes in different organisms
- **Paralogous** genes are homologous genes in one organism that derive from gene duplication
- **Gene duplication**: one gene is duplicated in multiple copies that therefore free to evolve and assume new functions

Sequence similarity

- Intuitively, similarity of two sequences refers to the degree of match between corresponding positions in sequence
- Sequence similarity is not sequence homology
- Homology is more difficult to detect over greater evolutionary distances

Causes of Gene (dis) similarity

- **mutation**: a nucleotide at a certain location is replaced by another nucleotide ATA $\rightarrow$ AGA
- **insertion**: at a certain location one new nucleotide is inserted in between two existing nucleotides (e.g.: AA $\rightarrow$ AGA)
- **deletion**: at a certain location one existing nucleotide is deleted (e.g.: ACTG $\rightarrow$ AC-G)
- **indel**: an insertion or a deletion

Sequence Alignment

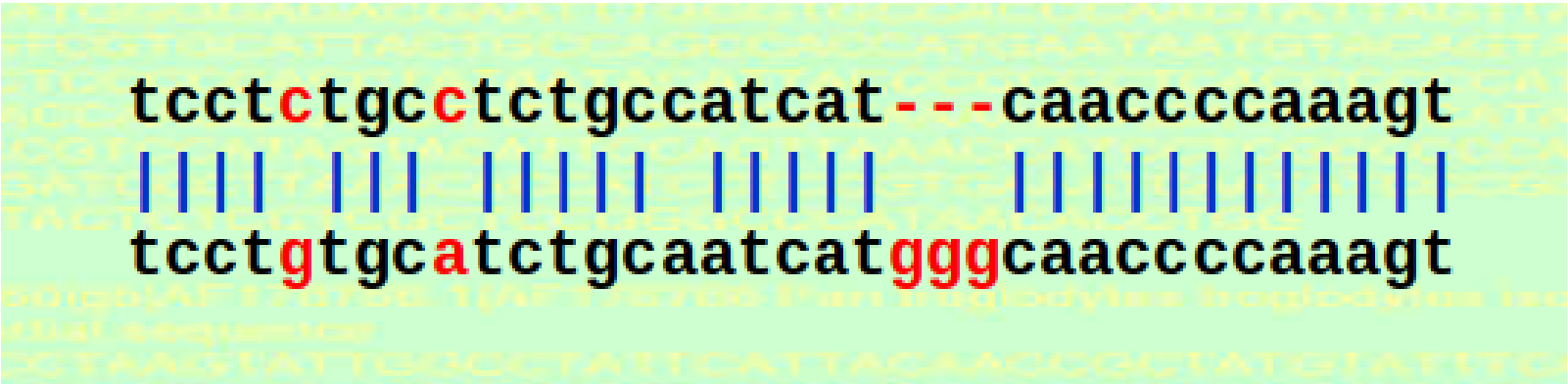
- Find the similarity between two (or more) DNA-sequences by finding a good alignment between them
- Alignment specifies which positions in two sequences match

```
acgtctag      acgtctag      acgtctag
||              |||||       || |||||
actctag-      -actctag      ac-tctag
```

2 matches	5 matches	7 matches
5 mismatches	2 mismatches	0 mismatches
1 not aligned	1 not aligned	1 not aligned

Sequence Alignment

- Sequence alignment is an arrangement of two or more sequences, highlighting their similarity.
- The sequences are padded with gaps (dashes) so that wherever possible, columns contain identical characters from the sequences involved



Sequence Alignment

- **Pairwise Sequence Alignment:** methods are concerned with finding the best-matching piece-wise local or global alignments of protein (amino acid) or DNA (nucleic acid) sequences.
- Global Alignment: an alignment in which all the characters in both sequences participate in the alignment.
- Local Alignment: a matching two sequence from regions which have more similar with each other
- Multiple Alignment

Algorithms

- **Needleman-Wunsch**
  Pairwise global alignment only.
- **Smith-Waterman**
  Pairwise, local (or global) alignment.
- **BLAST**
  Pairwise heuristic local alignment

The Needleman-Wunsch algorithm

- The Needleman-Wunsch algorithm (1970, J Mol Biol. 48(3):443-53) performs a global alignment on two sequences (s and t) and is applied to align protein or nucleotide sequences.
- The Needleman-Wunsch algorithm is an example of dynamic programming, and is guaranteed to find the alignment with the maximum score.

The Needleman-Wunsch algorithm

Alignment Scoring Function

The cost of aligning two symbols x_i and y_j is the scoring function $\sigma(x_i, y_j)$

$$\sigma(x_i, y_j) = \begin{cases} -1 & \text{if } x_i \neq y_j \text{ or } (x_i, \_) \text{ or } (\_, y_j) \\ 1 & \text{if } x_i = y_j \end{cases}$$

More better scores can be calculated by PAM (Point Accepted Mutation) matrix (Margaret Dayhoff), BLOSUM (BLOck SUbstitution Matrix) (Henikoff and Henikoff)

Alignment cost

$$M = \sum_{i=1}^{c} \sigma(x_i, y_j)$$

Dynamic Programming

- A matrix D(i,j)indexed by residues of each sequence is built recursively,such that
- A gap from left or above cell
- A match/mismatch from diagonal element

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

where $s(x_i, y_j)$ is the substitution cost g is the gap penalty

Dynamic Programming-steps

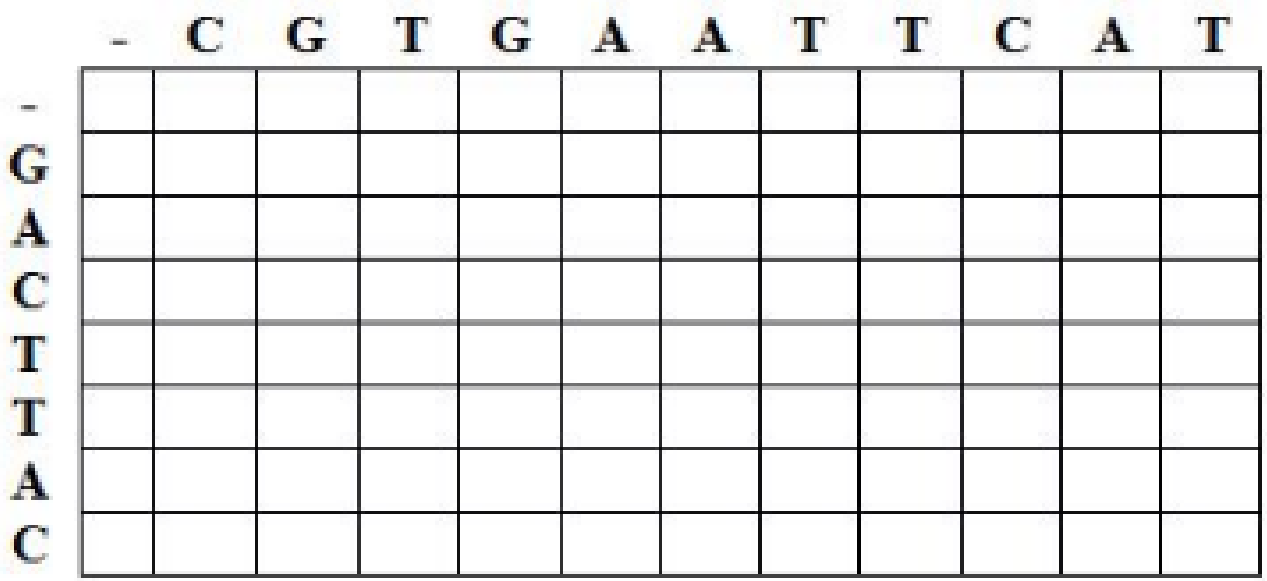
- Initialization of the score matrix
- Calculation of scores and filling the traceback matrix
- Deducing the alignment from the traceback matrix

Let's work on this simple example

- Input: `GACTT` (sequence #1) , `ATT` (sequence #2)

Steps

- Input: `CGTGAATTCAT` (sequence #1) , `GACTTAC` (sequence #2)



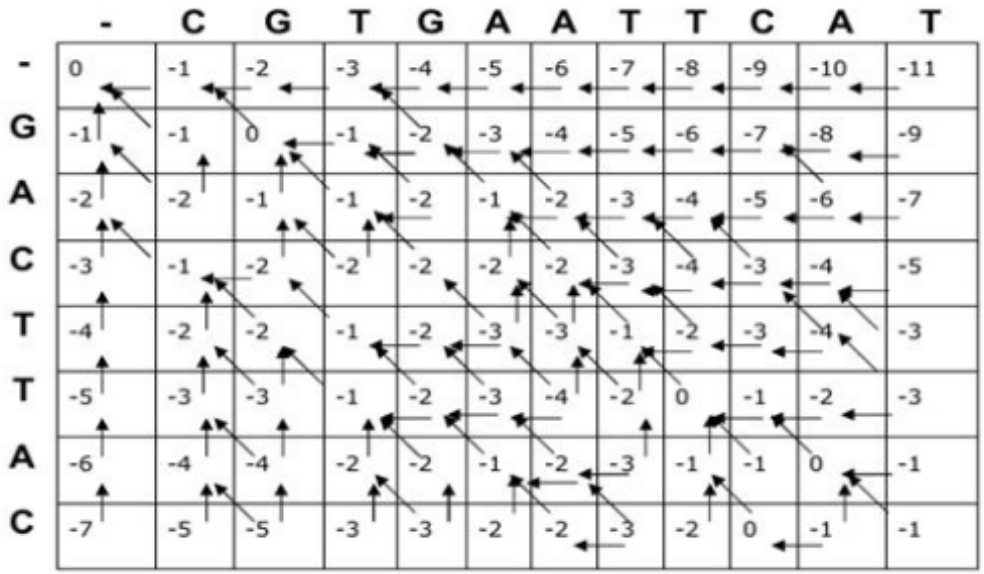
Steps: Initialization

- Input: `CGTGAATTCAT` (sequence #1) , `GACTTAC` (sequence #2)

		-	C	G	T	G	A	A	T	T	C	A	T
	-	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11
G		-1											
A		-2											
C		-3											
T		-4											
T		-5											
A		-6											
C		-7											

Steps: Filling the matrix

- Input: `CGTGAATTCAT` (sequence #1) , `GACTTAC` (sequence #2)



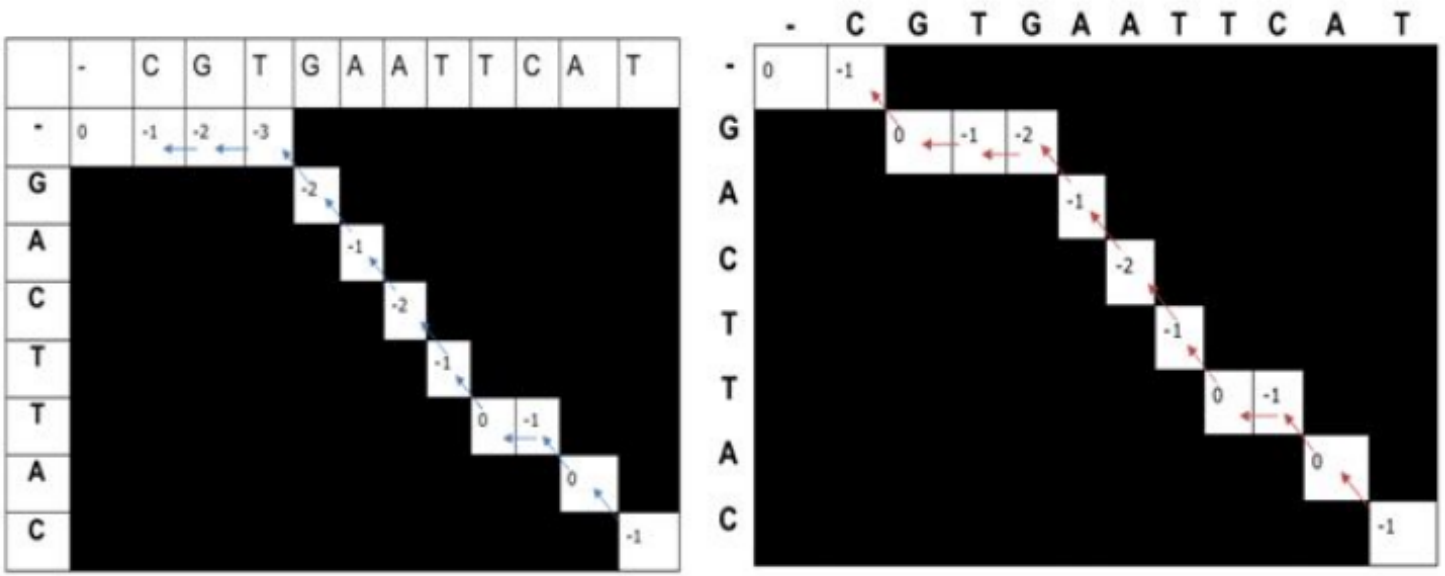
	-	C	G	T	G	A	A	T	T	C	A	T
-	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11
G	-1	-1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
A	-2	-2	-1	-1	-2	-1	-2	-3	-4	-5	-6	-7
C	-3	-1	-2	-2	-2	-2	-2	-3	-4	-3	-4	-5
T	-4	-2	-2	-1	-2	-3	-3	-1	-2	-3	-4	-3
T	-5	-3	-3	-1	-2	-3	-4	-2	0	-1	-2	-3
A	-6	-4	-4	-2	-2	-1	-2	-3	-1	-1	0	-1
C	-7	-5	-5	-3	-3	-2	-2	-3	-2	0	-1	-1

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

where $s(x_i, y_j)$ is the substitution cost g is the gap penalty

Steps: Traceback

- Input: `CGTGAATTCAT` (sequence #1) , `GACTTAC` (sequence #2)



Exercise

- Input: `ACAGTAG` (sequence #1) , `ACTCG` (sequence #2)

Next

Reading

Chapter-2, Krane and Raymer, Fundamentals of Bioinformatics

- Global Sequence Alignment
- BLAST
- MSA