

Sequence Alignment

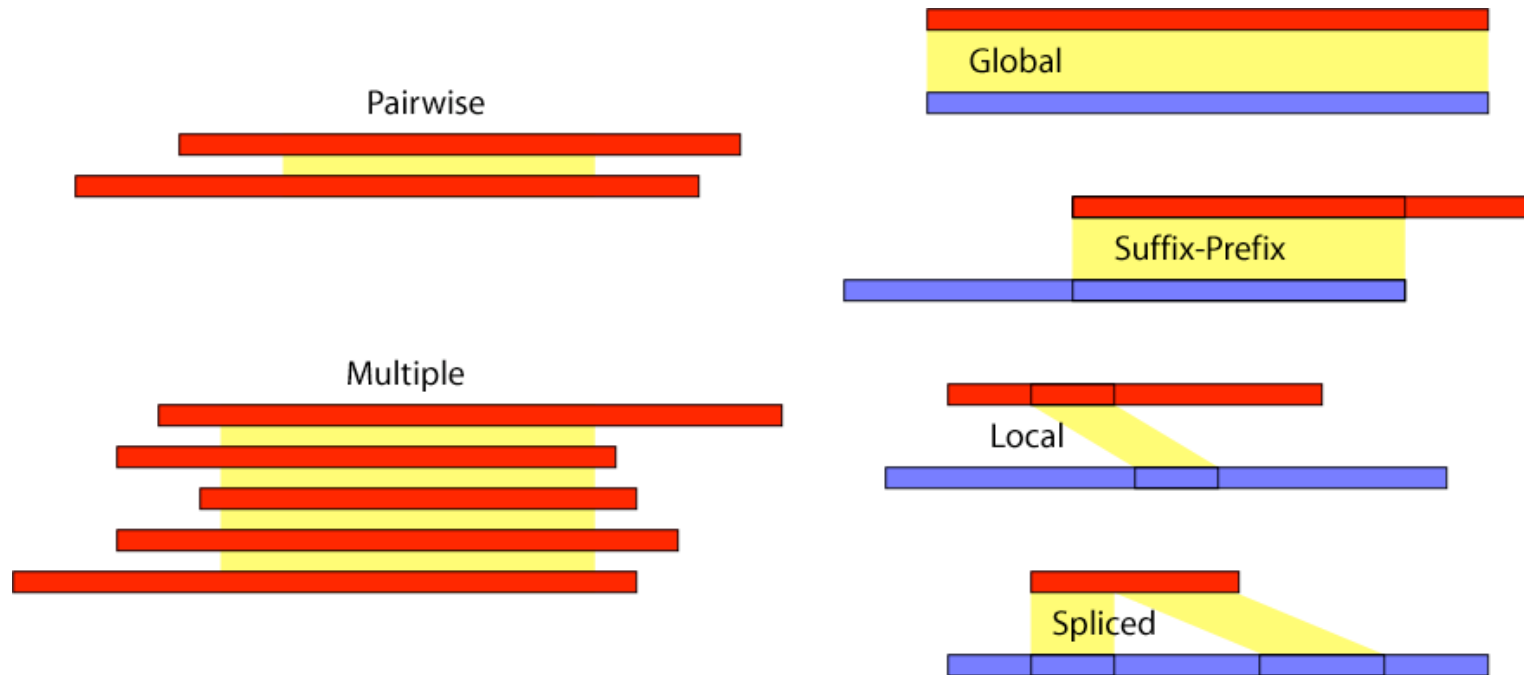
Introduction

- The next two weeks, we will discuss sequence alignment and all of its basic flavors.
- The ultimate goal of alignment is to describe sequence similarity, or how closely two sequences match each other.
 - Can be a score (number)
 - Can also be an “alignment” (visual)

Similarity vs. biology

- Similarity (also called identity) is the number of matches / alignment length
- Homology, on the other hand, implies sequences came from a common ancestor
- Two kinds:
 - Orthologous - speciation-based split
 - Paralogous - gene duplication-based split

Various types

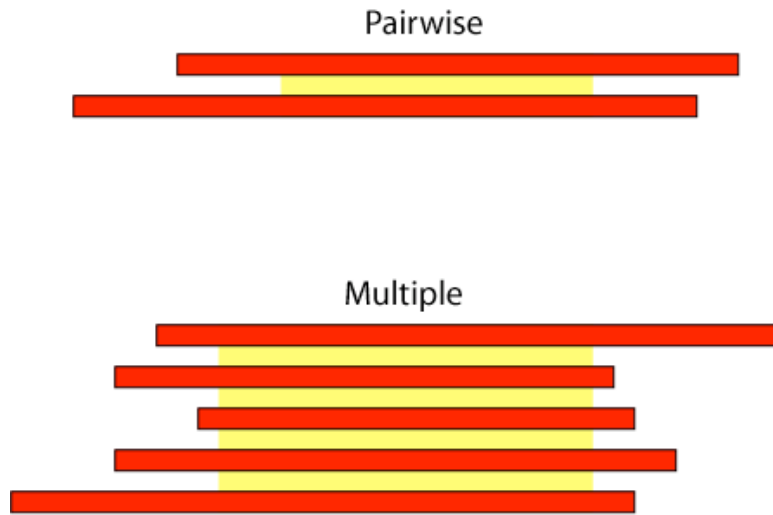


Applications

- Prediction on function
 - Commonalities among sequences can imply similar functions
- Database searching (BLAST)
 - Find interesting genes in a new genome
- Sequence divergence
 - Look at evolutionary relationships
- Sequence assembly
 - Making a big sequence from a bunch of small ones

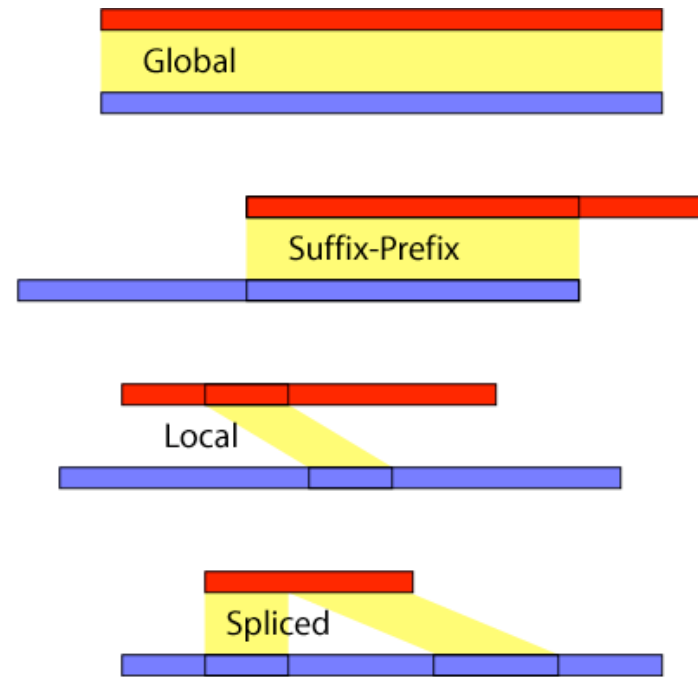
Examples

Database search



Phylogeny

Protein structure



Gene finding

Global alignment

- Also called a *pairwise alignment*
- Intuitive goal: related sequences will share many (most?) characters. To maximize this we introduce gaps represented by “-”

Two simple rules

- Rule #1:
 - A gap must be aligned to a nongap, i.e., “-” can not align to “-”
- Rule #2:
 - To distinguish good alignment from not so good ones, we introduce a scoring function E . Some functions have biological meaning, some are arbitrary.
- Consequence #1:
 - Alignment length can be no longer than sum of two sequences!

Alignments

- Show one sequence placed above another such that similarity is revealed

Example:

A: C A T - T C A - C
B: C - T C G C A G C

Improving readability

Example:

A:	C	A	T	-	T	C	A	-	C
B:	C	-	T	C	G	G	A	G	C

Scoring functions

- Here is a basic scoring function that rewards 1 for a match and -1 for a mismatch gap

$$E(-,a) = E(a,-) = E(a,b) = -1 \quad \forall a \neq b$$

$$E(a,b) = 1 \quad \forall a = b$$

- Can also be represented as a substitution matrix.

In class example

S: CATTCAC

T: CTCGCAGC

$$E(-,a) = E(a,-) = E(a,b) = -1 \quad \forall a \neq b$$

$$E(a,b) = 1 \quad \forall a = b$$

Measuring Similarity

Score: A measure of alignment quality

C	A	T	-	T	C	A	-	C
C	-	T	C	G	C	A	G	C

10	-5	10	-5	-2	10	10	-5	10

Total = 33

**Scored as $E(C,C)$ $E(A,-)$, $E(T,T)$, $E(-,C)$,
etc.**

What if we can do this?

- Suppose we have an optimum alignment of size L . Is the following true?
- $A^* = A^*(s_1 \dots s_i, t_1 \dots t_j) + A^*(s_{i+1} \dots s_n, t_{j+1} \dots t_m)$
 - Where $|s| = n$ and $|t| = m$
- Now, what would happen if $i = n - 1$ and $j = m - 1$?

Basic intuition

Case 1: Match $s[n]$ w/ $t[m]$

					$n - 1$	n	
s:	C	A	T	T	C	A	C
t:	C	-	T	T	C	A	G
					$m - 1$	m	

Case 2: Match $t[m]$ w/ gap

						n	
s:	C	A	T	T	C	A	C
t:	C	-	T	T	C	A	C
						$m - 1$	m

Case 3: Match $s[i]$ w/ gap

					$n - 1$	n	
s:	C	A	T	T	C	A	G
t:	C	-	T	T	C	A	G
					m		

Global alignment

- Dynamic programming (DP) will save the day!
- DP is a general technique used when a large problem can be broken into smaller, easier problems
- To solve sequence alignment, we will fix two substrings and find the best way to add the next character from at least one string.

Requirements

- We will need four things to compute a global alignment:
 1. Substitution matrix (parameters)
 2. Recurrence relation
 3. Filling up a table
 4. Traceback

Pairwise Global Alignment

$T[i,j]$ = Score of optimally aligning first i bases of s with first j bases of t .

$$T[i,j] = \max \begin{cases} T[i-1,j-1] + \text{score}(s[i], t[j]) \\ T[i-1,j] + g \\ T[i,j-1] + g \end{cases}$$

	λ	C	T	C	G	C	A	G	C
λ	0	-5	-10	-15	-20	-25	-30	-35	-40
C	-5	10	5						
A	-10								
T	-15								
T	-20								
C	-25								
A	-30								
C	-35								

+10 for match, -2 for mismatch, -5 for space (rowwise)

	λ	C	T	C	G	C	A	G	C
λ	0	-5	-10	-15	-20	-25	-30	-35	-40
C	-5	10	5	0	-5	-10	-15	-20	-25
A	-10	5	8	3	-2	-7	0	-5	-10
T	-15	0	15	10	5*	0	-5	-2	-7
T	-20	-5	10*	13	8	3	-2	-7	-4
C	-25	-10	5	20	15	18	13	8	3
A	-30	-15	0	15	18	13	28	23	18
C	-35	-20	-5	10	13	28	23	26	33

Traceback yields both optimal alignments in this example

Some Results

- Most pairwise sequence alignment problems can be solved in $O(mn)$ time.
- Space requirement can be reduced to $O(m+n)$, while keeping run-time fixed [Myers88].
- Two highly similar sequences can be aligned in $O(dn)$ time, where d is a measure of the distance between the sequences [Landau86].

Pairwise Sequence Alignment

Variations for next week:

- Given two sequences, find if parts of them are similar (local alignment).
- Given a large sequence and a short sequence, find if the short sequence is similar to a stretch of the long sequence.
- Cool fact is these are easy to do once we learn the basics of global alignment!