Lecture 5,6
Local sequence alignment

Chapter 6 in Jones and Pevzner

Spring 2018
February 1, 6, 2018
Evolution as a tool for biological insight

- “Nothing in biology makes sense except in the light of evolution” - Theodosius Dobzhansky.

- The functionality of many genes is virtually the same among many organisms: Can understand biology in simpler organisms than ourselves (“model organisms”).
Local alignment: rationale

- Proteins are often multi-functional, and are composed of regions (domains), each of which contributes a particular function.

- Example:
  - Homeobox genes have a short region called the *homeodomain* that is highly conserved between species.
  - A global alignment might not find the homeodomain because it would try to align the ENTIRE sequence.

Drosophila homeodomain PDB: 1ZQ3
Local vs. Global Alignment (cont’d)

Global Alignment

--T--CC--C--AGT--TATGT--CAGGGGACACG--A--GCATGCAGA--GAC  
|    |    |    |    |    |    |    |    |    |    |    |    |    |    |  |
AATTGCCGCC--GTCGT--T--TTCAG----CA--GTTATG--T--CAGAT--C

Local Alignment—better for finding a conserved segment

  tccCAGTTATGTCAAGgggacacgagcatgcagagac  
|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
aattgccgcccgtcggtttttcagCAGTTATGTCAGatc
The Local Alignment Problem

- **Goal**: Find the best local alignment between two strings
- **Input**: Strings $v$, $w$ and scoring matrix $\delta$
- **Output**: Alignment of substrings of $v$ and $w$ whose alignment score is maximum among all possible alignment of all possible substrings
Local vs. global alignment

Compute a “mini” global alignment to get a local alignment
Local vs. Global Alignment

- The **Global Alignment Problem** tries to find the longest path between vertices \((0,0)\) and \((n,m)\) in the edit graph.
- The **Local Alignment Problem** tries to find the longest path among paths between arbitrary vertices \((i,j)\) and \((i',j')\) in the edit graph.
- In an edit graph with negatively-scored edges, a local alignment may score higher than a global alignment.
The Problem with this Problem

- Naïve method (run time $O(n^4)$):
  - In a grid of size $n \times n$ there are $n^2$ nodes $(i,j)$ that may serve as a source.
  - For each such node computing alignments from $(i,j)$ to $(i',j')$ takes $O(n^2)$ time.
Local Alignment: Example
Local Alignment: Example
Local Alignment: Example
Local Alignment: Example
Local Alignment: Example
Local Alignment: Free Rides

The dashed edges represent the free rides from (0,0) to every other node.
Local Alignment: Recurrence

\[ s_{i,j} = \max \begin{cases} 
    0 \\
    s_{i-1,j-1} + \delta(v_i, w_j) \\
    s_{i-1,j} + \delta(v_i, -) \\
    s_{i,j-1} + \delta(-, w_j) 
\end{cases} \]

**Power of ZERO**: this is the only change from the original recurrence of a global alignment, representing the “free ride” edge.
Local Alignment: Backtrace

- Score of best local alignment is the maximum entry of $s_{ij}$
- The alignment is found by a backtrace from the maximum node, to a node for which the score is 0.
Local Alignment: SW

- This local alignment algorithm is known as the “Smith-Waterman” algorithm\(^1\).

\(^1\)The Smith-Waterman algorithm considers a more sophisticated gap penalty scheme
Scoring Indels: Naive Approach

- A fixed penalty $\sigma$ is given to every indel:
  - $-\sigma$ for 1 indel,
  - $-2\sigma$ for 2 consecutive indels
  - $-3\sigma$ for 3 consecutive indels, etc.

Can be too severe penalty for a series of 100 consecutive indels
Affine Gap Penalties

- In nature, a series of $k$ indels often comes as a single event rather than a series of $k$ single nucleotide events:

```
ATA--GC
ATATTGC
```

```
ATAG--GC
AT--GTGC
```

This is more likely.

Normal scoring would give the same score for both alignments.

This is less likely.
Affine gap penalty

- Score for a gap of length $x$ is:
  $$-(\rho + \sigma x)$$
  where: $\rho > 0$ is the gap opening penalty
  $\sigma > 0$ is the gap extension penalty

- $\rho$ is large relative to $\sigma$ because you do not want to add too much of a penalty for extending the gap.
To reflect affine gap penalties we have to add “long” horizontal and vertical edges to the edit graph. Each such edge of length $x$ should have weight $-\rho - x \cdot \sigma$.
Adding “Affine Penalty” Edges to the Edit Graph

• There are many such edges!
• Adding them to the graph increases the running time of the alignment algorithm by a factor of $n$.
• The complexity increases from $O(n^2)$ to $O(n^3)$
The 3-leveled Manhattan Grid

gaps in $w$

matches/mismatches

gaps in $v$
Manhattan in 3 Layers
Affine Gap Penalties and 3 Layer Manhattan Grid

- We’ll have three recurrences in a 3-layered graph.
- The top level creates/extends gaps in the sequence $w$.
- The bottom level creates/extends gaps in sequence $v$.
- The middle level extends matches and mismatches.
Switching Between the Layers

- Levels:
  - The **main level** is for diagonal edges
  - The **lower level** is for horizontal edges
  - The **upper level** is for vertical edges
- A jumping penalty is assigned to moving from the main level to either the upper level or the lower level (-\(\rho\) - \(\sigma\))
- There is a gap extension penalty for each continuation on a level other than the main level (-\(\sigma\))
Affine Gap Penalty Recurrences

\[
\begin{align*}
    s_{i,j} & = \max \begin{cases} 
        s_{i-1,j} - \sigma \\
        s_{i-1,j} - (\rho + \sigma) 
    \end{cases} & \text{Continue gap in } w \\
    s_{i,j} & = \max \begin{cases} 
        s_{i,j-1} - \sigma \\
        s_{i,j-1} - (\rho + \sigma) 
    \end{cases} & \text{Continue gap in } v \\
    s_{i,j} & = \max \begin{cases} 
        s_{i-1,j-1} + \delta(v_i, w_j) \\
        s_{i,j} \\
        s_{i,j} 
    \end{cases} & \text{Match or Mismatch}
\end{align*}
\]

Continue gap in \( w \) : from middle
Start gap in \( w \) : from middle
Continue gap in \( v \)
Start gap in \( v \) : from middle
Match or Mismatch
End gap: from top
End gap: from bottom
Should we compare DNA or protein sequences?

- DNA sequence is less conserved than protein sequence
- The protein sequence contains more information than the DNA sequence

⇒ Less effective to compare coding regions at the nucleotide level
Scoring Matrices
Making a Scoring Matrix

- Scoring matrices are created based on the intuition that some mutations have a smaller effect on the function of a protein
  ⇒ Such mismatch penalties should be less harsh than others.
Although R and K are different amino acids, they have a positive score.

Why? They are both positively charged amino acids, therefore substitution will not greatly change the function of the protein.
Substitutions of Amino Acids

Mutation rates between amino acids have dramatic differences!

<table>
<thead>
<tr>
<th>HYDROPHOBIC AMINO ACIDS</th>
<th>HYDROPHILIC AMINO ACIDS</th>
<th>ACIDIC AMINO ACIDS</th>
<th>POLAR AMINO ACIDS WITH UNCHARGED R GROUPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alanine (Ala or A)</td>
<td>Lysine (Lys or K)</td>
<td>Aspartate (Asp or D)</td>
<td>Serine (Ser or S)</td>
</tr>
<tr>
<td>Valine (Val or V)</td>
<td>Arginine (Arg or R)</td>
<td>Glutamate (Glu or E)</td>
<td>Threonine (Thr or T)</td>
</tr>
<tr>
<td>Isoleucine (Ile or I)</td>
<td>Histidine (His or H)</td>
<td>Asparagine (Asn or N)</td>
<td>Glutamine (Gln or Q)</td>
</tr>
<tr>
<td>Leucine (Leu or L)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SPECIAL AMINO ACIDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methionine (Met or M)</td>
</tr>
<tr>
<td>Phenylalanine (Phe or F)</td>
</tr>
<tr>
<td>Tyrosine (Tyr or Y)</td>
</tr>
<tr>
<td>Tryptophan (Trp or W)</td>
</tr>
</tbody>
</table>

| Cysteine (Cys or C) |
| Glycine (Gly or G)  |
| Proline (Pro or P)  |
Conservation

- Amino acid changes that preserve the physicochemical properties of the original residue should receive higher scores
  - polar to polar
    - aspartate $\rightarrow$ glutamate
  - hydrophobic to hydrophobic
    - alanine $\rightarrow$ valine
Percent Sequence Identity

- A measure of the extent to which two nucleotide or amino acid sequences are similar

70% identical

mismatch

indel
**BLOSUM**

- **Blocks Substitution Matrix**
- Scores derived from *observations* of the frequencies of substitutions in alignments of related proteins
- Matrix name indicates evolutionary distance:
  - BLOSUM62 was created using sequences sharing no more than 62% identity

Amino acid substitution matrices from protein blocks.
An entry from the BLOCKS database

Block PR00851A

ID   XRODRMPGMNTB; BLOCK
AC   PR00851A; distance from previous block=(52,131)
DE   Xeroderma pigmentosum group B protein signature
BL   adapted; width=21; seqs=8; 99.5%=985; strength=1287

XPB_HUMAN|P19447    ( 74) RPLWVAPDGHIIFLEAFSPVYK 54
XPB_MOUSE|P49135   ( 74) RPLWVAPDGHIIFLEAFSPVYK 54
P91579     ( 80) RPLYLAPDGHIIFLESFSPVYK 67
XPB_DROME|Q02870   ( 84) RPLWVAPNGHVIFLESFSPVYK 79
RA25_YEAST|Q00578  (131) PLWISPSDGRIILLESFSPLAE 100
Q38861     ( 52) RPLWACADGRIFLETFSPLYK 71
O13768     ( 90) PLWINPIDGRIILEAFSPLAE 100
O00835     ( 79) RPIWVCVPDGHIIFLETFSAIYK 86

The BLOCKS database is at: http://blocks.fhcrc.org/
# The Blosum50 Scoring Matrix

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

*Note: The table represents the Blosum50 scoring matrix for amino acid alignments.*