Biosequence Algorithms, Spring 2005
Lecture 11: Introduction to Multiple Alignments

Pekka Kilpeläinen
University of Kuopio
Department of Computer Science
Multiple String Alignments

In this section:

(~ Sections 14.1–14.3 in Gusfield)

1. Definition of multiple alignment
2. Biological motivation
3. Aligning strings with a profile
A (global) multiple alignment (globaali monirinnastus) of $k > 2$ strings is an obvious generalization of two-string alignments:
Insert spaces into the strings to make them equally long (say, $l$ chars), and arrange them in $k$ rows and $l$ columns, each character or space in a unique column

Example: A multiple alignment of strings
\{abca, ababa, acbc, cbcc\}

```
a  b  c  _  a
a  b  a  b  a
a  c  c  b  _
c  b  _  b  c
```
Local Multiple Alignments

There are also *local* multiple alignments

A **local multiple alignment** of strings $S_1, \ldots, S_k$ consists of selecting a *substring* $S'_i$ of each $S_i$, and aligning these $k$ substrings globally.

We restrict to considering *global* multiple alignments only.
Multiple alignment is a specific formalization of *multiple string comparison*, which is one of the most important methodologies and active research areas in bio-sequence analysis; It is used for

- extracting and representing biologically important commonalities from a set of strings
  - which might go unnoticed if only two strings were compared
- inferring evolutionary history from DNA or protein sequences
Found commonalities are used to characterize (and to understand) families of proteins

- **family**: set of proteins related by structure, function, or evolutionary history, e.g., *globins* and *immunoglobulins*

“This classification is central to our understanding of how life has evolved, and makes elucidation and definition of such families one of the principal concerns of molecular biology” *(Cyrus Chothia, Nature, 1992)*
Usefulness of Compact Representations

Family representations are useful: It’s been estimated that the ~ 100,000 human proteins could be organized in about thousand families (or even a few hundred only)

A new sequence can be tested for potential membership in a family by comparing it with a family representation

Commonly used forms: profiles, consensus sequences, and signatures

all derived from multiple string comparison
Profiles as Family Representations

A profile of a multiple alignment $M$ with row-length $l$ is a $|\Sigma'| \times l$ matrix $p$, where $p(y, j)$ is the occurrence frequency that char $y$ occurs in column $j$ of $M$. ($\Sigma' = \Sigma \cup \{\_\}$)

Example: The profile of the previous multiple alignment:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>.75</td>
<td>.00</td>
<td>.25</td>
<td>.00</td>
<td>.50</td>
</tr>
<tr>
<td>$b$</td>
<td>.00</td>
<td>.75</td>
<td>.00</td>
<td>.75</td>
<td>.00</td>
</tr>
<tr>
<td>$c$</td>
<td>.25</td>
<td>.25</td>
<td>.50</td>
<td>.00</td>
<td>.25</td>
</tr>
<tr>
<td>_</td>
<td>.00</td>
<td>.00</td>
<td>.25</td>
<td>.25</td>
<td>.25</td>
</tr>
</tbody>
</table>

How to compare a string and a profile?
Aligning a String to a Profile

A profile $p$ is a sequence of columns $\sim \rightarrow$ we can align a string $S$ with $p$, by inserting spaces in them:

**Example:** A string/profile alignment:

<table>
<thead>
<tr>
<th>$p'$:</th>
<th>1</th>
<th>_</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>.75</td>
<td>.00</td>
<td>.25</td>
<td>.00</td>
<td>.50</td>
<td></td>
</tr>
<tr>
<td>$b$</td>
<td>.00</td>
<td>.75</td>
<td>.00</td>
<td>.75</td>
<td>.00</td>
<td></td>
</tr>
<tr>
<td>$c$</td>
<td>.25</td>
<td>.25</td>
<td>.50</td>
<td>.00</td>
<td>.25</td>
<td></td>
</tr>
<tr>
<td>_</td>
<td>.00</td>
<td>1.0</td>
<td>.00</td>
<td>.25</td>
<td>.25</td>
<td>.25</td>
</tr>
</tbody>
</table>

| $S'$: | a | a | b | _ | b | c |

How to score a string/profile alignment?
Scoring a String/Profile Alignment

Common approach: (1) The score $S(x, j)$ of a char $x$ aligned with a column $j$ is the average of the pair-wise character scores btw $x$ and any character at col $j$:

$$S(x, j) = \sum_{y \in \Sigma'} [s(x, y) \times p(y, j)]$$

(2) score of the full alignment = sum of column scores

**Example**: Assume character scores $s(a, a) = 2$, $s(a, b) = s(a, _) = -1$, and $s(a, c) = -3$

The first column of the previous alignment adds $S(a, 1) = 0.75 \times 2 + 0.25 \times (-3)$ to the total score, and the second $S(a, _) = 1.0 \times (-1)$
An optimal string/profile alignment can be computed as a straightforward extension of string/string alignments. Let $V(i, j)$ denote the value of an optimal alignment of prefix $S[1 \ldots i]$ with columns $1, \ldots, j$ of profile $p$.

**Recurrences:**

**Base cases:**

1. $V(0, j) = \sum_{k=1}^{j} S(_, k)$ (against $j$ first columns of $p$)
2. $V(i, 0) = \sum_{k=1}^{i} s(S_1[k], _) (S_1[1 \ldots i]$ against spaces)
Computing a String/Profile Alignment

**Inductive cases** for $i, j > 0$:

$$V(i, j) = \max \begin{cases} 
V(i - 1, j - 1) + S(S_1[i], j) \\
V(i - 1, j) + s(S_1[i], \_)
\end{cases}$$

With these recurrences an optimal string/profile alignment can be computed, similarly to a string/string alignment, in time $O(|\Sigma|nm)$.

Factor $|\Sigma|$ comes from considering all characters of column $j$ for computing the score of aligning a character at column $j$. 