Global Alignment of Protein Sequences
(NW, SW, PAM, BLOSUM)
Topic 1 Info

• Overview slide has blue background - readings for upcoming lectures are listed at bottom of overview slide

• Review slides will have purple background

• Send your background/interests to TA for posting if reg’d for grad version

• PS1 is posted. BLAST tutorial may be helpful

• PS2 is posted. Look at the programming problem
Local Alignment (BLAST) and Statistics

- Sequencing
  - Conventional
  - 2nd generation

- Local Alignment:
  - a simple BLAST-like algorithm
  - Statistics of matching
  - Target frequencies and mismatch penalties for nucleotide alignments

Background for 2/7, 2/12 lectures: Z&B Ch. 4 & 5, BLAST tutorial
Questions: Chemistry / Library Prep

Dye terminator chemistry: dye is attached to base

How to put different adapters on the two ends?
At least three ways:

1) RNA ligation

2) polyA tailing/polyTVN-ad2priming/circularization (PMID 19213877)

3) ligation of Y-shaped adapters
DNA Sequence Alignment I: Motivation

You are studying a recently discovered human non-coding RNA. You search it against the mouse genome using BLASTN (N for nucleotide) and obtain the following alignment:

Q: 1   ttgacctagatgagatgtcgttcacttttactcaggtacagaaaa 45
      ||||  |||||||||||| | |||||||||||| || |||||||||
S: 403 ttgatctagatgagatgcattcacttttactgagctacagaaaa 447

Is this alignment significant? Is this likely to represent a homologous RNA? How to find alignments?
DNA Sequence Alignment II

Identify high scoring segments whose score $S$ exceeds a cutoff $x$ using a local alignment algorithm (e.g., BLAST)

Scores follow an extreme value (aka Gumbel) distribution:

$$P(S > x) = 1 - \exp[-KMN e^{-\lambda x}]$$

For sequences/databases of length $M, N$ where $K, \lambda$ are positive parameters that depend on the score matrix and the composition of the sequences being compared

Conditions: expected score is negative, but positive scores possible

Alternate algorithm

Karlin & Altschul 1990
Computational Efficiency

Measure efficiency in cpu run time and memory

$O(\cdot) = \text{“big-oh” notation (computational Order of problem)}$

Consider the number of individual computations required to run algorithm as a function of the number of ‘units’ in the problem (e.g., base pairs, amino acid residues)

Analyze the asymptotic worst-case running time or sometimes just do the experiment and measure run time

If problem scales as square of the number of units it is

$O(n^2)$  “order n-squared”
DNA Sequence Alignment III

How is $\lambda$ related to the score matrix?

$\lambda$ is the unique positive solution to the equation*:

$$\sum_{i,j} p_i r_j e^{\lambda S_{ij}} = 1$$

$p_i = \text{freq. of nt } i \text{ in query}$, $r_j = \text{freq. of nt } j \text{ in subject}$

$S_{ij} = \text{score for aligning an } i,j \text{ pair}$

“Target frequencies”* : $q_{ij} = p_i r_j e^{\lambda S_{ij}}$

*Karlin & Altschul, 1990
Optimal mismatch penalty $m$ for given target identity fraction $r$

$$m = \frac{\ln(4(1-r)/3)}{\ln(4r)}$$

Examples:

<table>
<thead>
<tr>
<th>$r$</th>
<th>0.75</th>
<th>0.95</th>
<th>0.99</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m$</td>
<td>-1</td>
<td>-2</td>
<td>-3</td>
</tr>
</tbody>
</table>

$r$ = expected fraction of identities in high-scoring BLAST hits
Meaning of mismatch penalty equation

\[ m = \frac{\ln(4(1-r)/3)}{\ln(4r)} \]

So why is \( m = -3 \) better for finding matches with 99% identity?

Does it mean that you can only find 99% identical matches with a mismatch score of -3?

Answer: No. It’s also possible to find 99% matches with \( m = -1 \) or -2.

But \( m \) changes the match length required to achieve statistical significance.

\( \lambda \) is the unique positive solution to the equation

\[ \sum p_i p_j e^{\lambda s_{ij}} = 1 \]

\( p_i \) = frequency of nt i, \( s_{ij} \) = score for aligning an i,j pair

and \( P(S > x) = 1 - \exp[-KMN e^{-\lambda x}] \)

If we change the mismatch score from -1 to -3, \( \lambda \) will increase. Therefore, the score required to achieve a given level of significance will decrease, i.e. shorter hits will be significant.

So why would you ever want to use \( m = -1 \)?
Google: blastn

Courtesy of National Library of Medicine. In the public domain.
DNA Sequence Alignment VIII

Translating searches:
  translate in all possible reading frames
  search peptides against protein database (BLASTP)

```
ttgacctagatgagatgtcgttcacttttactgagctacagaaaa
```

```
ttg  acc  tag  atg  aga  tgt  cgt  tca  ctt  tta  ctg  agc  tac  aga  aaa
   L  T  x  M  R  C  R  S  L  L  L  S  Y  R  K
```

```
t  tga  cct  aga  tga  gat  gtc  gtt  cac  ttt  tac  tga  gct  aca  gaa  aa
   x  P  R  x  D  V  V  H  F  Y  x  S  T  E
```

```
t  gac  cta  gat  gag  atg  tcg  ttc  act  ttt  act  gag  cta  cag  aaa  a
   D  L  D  E  M  S  F  T  F  T  E  L  Q  K
```

Also consider reading frames on complementary DNA strand
DNA Sequence Alignment IX

Common flavors of BLAST:

<table>
<thead>
<tr>
<th>Program</th>
<th>Query</th>
<th>Database</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLASTP</td>
<td>aa</td>
<td>aa</td>
</tr>
<tr>
<td>BLASTN</td>
<td>nt</td>
<td>nt</td>
</tr>
<tr>
<td>BLASTX</td>
<td>nt (⇒ aa)</td>
<td>aa</td>
</tr>
<tr>
<td>TBLASTN</td>
<td>aa</td>
<td>nt (⇒ aa)</td>
</tr>
<tr>
<td>TBLASTX</td>
<td>nt (⇒ aa)</td>
<td>nt (⇒ aa)</td>
</tr>
<tr>
<td>PsiBLAST</td>
<td>aa (aa msa)</td>
<td>aa</td>
</tr>
</tbody>
</table>

*msa = multiple sequence alignment*

Which would be best for searching ESTs against a genome?
Global Alignment of Protein Sequences (NW, SW, PAM, BLOSUM)

- Global sequence alignment (Needleman-Wunch-Sellers)
- Gapped local sequence alignment (Smith-Waterman)
- Substitution matrices for protein comparison

Background for today: Z&B Chapters 4,5 (esp. pp. 119-125)
Why align protein sequences?

- Functional predictions based on identifying homologous proteins or protein domains

Assumes

Sequence similarity $\rightarrow$ Similarity in function (and/or structure)

- almost always true for similarity $> 30$
- $20-30\%$ similarity is “the twilight zone”

BUT: Function carried out at level of folded protein, i.e. 3-D structure
Sequence conservation occurs at level of 1-D sequence

Converse is not true

Structural similarity $\not\rightarrow$ Sequence similarity (or even homology)
Convergent Evolution

Last common ancestor lived > 500 Mya and lacked wings (and probably legs and eyes)

Same idea for proteins - can result in similar structures with no significant similarity in sequence
Convergent Evolution of Fe$^{3+}$-binding Proteins

Last common ancestor occurred > 2Bya and bound anions

Courtesy of Nature Publishing Group. Used with permission.

Convergent Evolution of a Protein and an RNA

Unlikely to have ever had a common molecular ancestor

© American Association for the Advancement of Science. All rights reserved. This content is excluded from our Creative Commons license. For more information, see [http://ocw.mit.edu/help/faq-fair-use/](http://ocw.mit.edu/help/faq-fair-use/).


T. maritima ribosome recycling factor (RRF)
Types of Alignments

Scope:
- Local
- Global
- Semiglobal

Scoring system:
- Ungapped
- Gapped
  - linear
  - affine
Dot Matrix Alignment Example

What type of alignment would be most appropriate for this pair of sequences?

Global
What type of alignment would be most appropriate for this pair of sequences? Local
Gaps (aka “Indels”)

- Linear Gap Penalty
  \[ \gamma(n) = nA, \quad n = \text{no. of gaps}, \quad A = \text{gap penalty} \]

- “Affine” gap penalty
  \[ W_n = G + n\gamma, \]
  \[ n = \text{no. of gaps}, \quad \gamma = \text{gap extension penalty}, \]
  \[ \text{and } G = \text{gap opening penalty} \]
  Or:
  \[ W_n = G + (n-1)\gamma \]
  with alternative definition of gap opening penalty
Obtain optimal global alignment using *Dynamic Programming*:

First write one sequence across the top, and one down along the side

<table>
<thead>
<tr>
<th>Gap</th>
<th>V</th>
<th>D</th>
<th>S</th>
<th>C</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gap</td>
<td>0</td>
<td>1 gap</td>
<td>2 gaps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>V</td>
<td></td>
<td>1 gap</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td>2 gaps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Note – linear gap penalty:* $\gamma(n) = nA$, where $A =$ gap penalty *a negative number*
Dynamic Programming:

Initialize the alignment matrix

<table>
<thead>
<tr>
<th>i =0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>j =</td>
<td>Gap</td>
<td>V</td>
<td>D</td>
<td>S</td>
<td>C</td>
</tr>
<tr>
<td>0</td>
<td>Gap</td>
<td>0</td>
<td>-8</td>
<td>-16</td>
<td>-24</td>
</tr>
<tr>
<td>1</td>
<td>V</td>
<td>-8</td>
<td>S</td>
<td>C</td>
<td>Y</td>
</tr>
</tbody>
</table>

\[ S_{ij} \]

**S\_ij** = score of optimal alignment ending at position i in seq 1 and j in seq 2. Requires that we know **S**(i-1, j-1), **S**(i, j-1), **S**(i-1, j)...

4 L   -32

Recursive: Solution to larger problem is built up from solutions to smaller problems

5 C   -40

Store **S\_ij** and how we arrived at **S\_ij** in a matrix

Often called ‘dynamic programming’ or more generally ‘recursive optimization’

What is the gap penalty in this example?
Dynamic Programming: Recursion

Sequence 1

i = 0 1 2 3 4 5

Sequence 2

j = Gap V D S C Y

<table>
<thead>
<tr>
<th>j</th>
<th>0</th>
<th>V</th>
<th>E</th>
<th>S</th>
<th>L</th>
<th>C</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Gap</td>
<td>0</td>
<td>-8</td>
<td>-16</td>
<td>-24</td>
<td>-32</td>
<td>-40</td>
</tr>
<tr>
<td>1</td>
<td>V</td>
<td>-8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>E</td>
<td>-16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>S</td>
<td>-24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>L</td>
<td>-32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>C</td>
<td>-40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Y</td>
<td>-48</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Global alignments: Needleman-Wunsch-Sellers

\[ S_{ij} = \text{max of:} \begin{cases} 
S_{i-1, j-1} + \sigma(x_i, y_j) \text{ (diagonal)} \\
S_{i-1, j} + A \text{ (from left to right)} \\
S_{i, j-1} + A \text{ (from top to bottom)} 
\end{cases} \]

Computational complexity? \( O(mn) \) with linear gap penalty
PAM250 Scoring Matrix

|     | C   | S   | T   | P   | A   | G   | N   | D   | E   | O   | H   | R   | K   | M   | I   | L   | V   | F   | Y   | W   |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| C   | 12  | 0   | 2   | 3   |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| S   | 0   | 12  | 1   | 3   |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| T   | -2  | 12  | 12  | 3   |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| P   | -3  | 10  | 5   | 6   |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| A   | -2  | 11  | 1   | 2   |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| G   | -3  | 10  | 11  | 1   |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| N   | -4  | 10  | 1   | 1   | 0   |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| D   | -5  | 0   | -1  | 1   | 0   | 1   |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| E   | -5  | 0   | -1  | 1   | 0   | 1   | 1   |     |     |     |     |     |     |     |     |     |     |     |     |
| Q   | -5  | -1  | -1  | 0   | -1  | -2  | -1  | 1   |     |     |     |     |     |     |     |     |     |     |     |     |
| H   | -3  | -1  | -1  | 0   | -1  | -2  | 2   | 1   | 1   | 2   | 3   | 4   |     |     |     |     |     |     |     |     |
| R   | -4  | 0   | -1  | 1   | 0   | -1  | 0   | 1   | 0   | 2   | 3   | 5   |     |     |     |     |     |     |     |     |
| K   | -5  | 0   | 0   | -1  | -1  | -2  | 1   | 0   | 0   | 2   | 3   | 5   |     |     |     |     |     |     |     |     |
| M   | -5  | -2  | -1  | -2  | -1  | -3  | -2  | -3  | -4  | -2  | -3  | -3  | 6   |     |     |     |     |     |     |     |
| I   | -2  | -1  | 1   | -1  | 0   | -3  | -2  | -2  | -3  | -2  | -2  | -2  | 2   | 5   |     |     |     |     |     |     |
| L   | -6  | -3  | -2  | -3  | -2  | -4  | -3  | -4  | -3  | -2  | -3  | -3  | 4   | 2   | 6   |     |     |     |     |     |
| V   | -2  | -1  | 0   | 1   | 0   | -1  | 2   | -2  | -2  | -2  | -2  | -2  | 2   | 2   | 4   | 4   |     |     |     |     |
| F   | -4  | -3  | -3  | -5  | -4  | -5  | -4  | -6  | -5  | -5  | -4  | -5  | 0   | 1   | 2   | 1   | 2   | 1   | 1   |
| Y   | -3  | -3  | -5  | -3  | -5  | -2  | -4  | -4  | -4  | 0   | -4  | -4  | -2  | -1  | -1  | -2  | 7   | 10  | 1    |
| W   | -8  | -2  | -5  | -6  | -7  | -4  | -7  | -7  | -5  | -3  | -3  | -4  | -5  | -2  | -6  | 0   | 0   | 17  |     |

© source unknown. All rights reserved. This content is excluded from our Creative Commons license. For more information, see [http://ocw.mit.edu/help/faq-fair-use/](http://ocw.mit.edu/help/faq-fair-use/).
### Dynamic Programming: filling in matrix

\[ S_{ij} = \max \{ S_{i-1, j-1} + \sigma(x_i, y_j) \text{ (diagonal)}, \ S_{i-1, j} + A \text{ (from left to right)}, \ S_{i, j-1} + A \text{ (from top to bottom)} \} \]
Sequence 1

Sequence 2

<table>
<thead>
<tr>
<th>i = 0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>j =</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>Gap</td>
<td>0</td>
<td>-8</td>
<td>-16</td>
<td>-24</td>
</tr>
<tr>
<td>1</td>
<td>V</td>
<td>-8</td>
<td>4</td>
<td>S</td>
<td>S _ ij = max of: S_{i-1, j-1} + \sigma(x_i, y_j) (diagonal)</td>
</tr>
<tr>
<td>2</td>
<td>E</td>
<td></td>
<td></td>
<td>-16</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>S</td>
<td></td>
<td></td>
<td>-24</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>L</td>
<td></td>
<td></td>
<td>-32</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>C</td>
<td></td>
<td></td>
<td>-40</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Y</td>
<td></td>
<td></td>
<td>-48</td>
<td></td>
</tr>
</tbody>
</table>
### Sequence 1

<table>
<thead>
<tr>
<th>i=0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Sequence 2

<table>
<thead>
<tr>
<th>j</th>
<th>i=0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gap</td>
<td>0</td>
<td>-8</td>
<td>-16</td>
<td>-24</td>
<td>-32</td>
<td>-40</td>
</tr>
</tbody>
</table>

- **0**
  - Gap: 0
  - Score: -8, -16, -24, -32, -40
- **1**
  - V: -8
  - Score: 4
- **2**
  - E
- **3**
  - S
- **4**
  - L
- **5**
  - C
- **6**
  - Y
### Completed Dynamic Programming Matrix

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>i = 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>j = 0</td>
<td>Gap</td>
<td>V</td>
<td>D</td>
<td>S</td>
<td>C</td>
<td>Y</td>
</tr>
<tr>
<td>0</td>
<td>Gap</td>
<td>0</td>
<td>-8</td>
<td>-16</td>
<td>-24</td>
<td>-32</td>
</tr>
<tr>
<td>1</td>
<td>V</td>
<td>-8</td>
<td>4</td>
<td>-8</td>
<td>-4</td>
<td>-12</td>
</tr>
<tr>
<td>2</td>
<td>E</td>
<td>-16</td>
<td>-6</td>
<td>7</td>
<td>-1</td>
<td>-9</td>
</tr>
<tr>
<td>3</td>
<td>S</td>
<td>-24</td>
<td>-14</td>
<td>-6</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>L</td>
<td>-32</td>
<td>-22</td>
<td>-14</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>5</td>
<td>C</td>
<td>-40</td>
<td>-30</td>
<td>-22</td>
<td>-7</td>
<td>13</td>
</tr>
<tr>
<td>6</td>
<td>Y</td>
<td>-48</td>
<td>-38</td>
<td>-30</td>
<td>-15</td>
<td>5</td>
</tr>
</tbody>
</table>

Keep track of scores AND how we got them → “traceback matrix”
The Traceback:

After the alignment square is finished, start at the lower right and work backwards following the arrows to see how you got there...

\[
\begin{array}{cccccc}
  & i = 0 & 1 & 2 & 3 & 4 & 5 \\
 0 & \text{Gap} & 0 & 4 & -8 & -16 & -24 & -32 & -40 \\
1 & V & -8 & \text{4} & -12 & -20 & -28 \\
2 & E & -16 & -6 & 7 & 2 & 1 & 11 & 0 \\
3 & S & -24 & -14 & -6 & 9 & 1 & 1 & 0 \\
4 & L & -32 & -22 & -14 & 1 & 3 & 0 & 0 \\
5 & C & -40 & -30 & -22 & -7 & 13 & 3 & 3 \\
6 & Y & -48 & -38 & -30 & -15 & 5 & 23 \\
\end{array}
\]
The Traceback gives the alignment:

```
  0  Gap  0  4  -8  -16  -24  -32  -40
  1  V    -8  4   -8  -12  -20  -28
  2  E    -16 -6   7   -1  -9  -17
  3  S    -24 -14 -6   9   1   -7
  4  L    -32 -22 -14  1   3    0
  5  C    -40 -30 -22 -7  13    3
  6  Y    -48 -38 -30 -15  5   23
```

“Life must be lived forwards and understood backwards.”
- Søren Kierkegaard
Semiglobal Alignment

Allow sequences to overhang at either end without penalty
-usually gives better alignments of homologous sequences of
different lengths

Same algorithm as before except

- initialize edges of DP matrix $S_{i,0}$ and $S_{0,j}$ to 0

- instead of requiring traceback to begin at $S_{m,n}$, allow it to
  begin at highest score in bottom row or rightmost column
Gapped Local Alignment

Temple Smith and Michael Waterman, 1981 – modified Needleman-Wunsch-Sellers

Local alignment is the best scoring alignment of a substring in sequence x to a substring in sequence y.

Key idea is not to force the alignment to extend to the ends of the sequences

Photograph of scientists removed due to copyright restrictions.
Smith-Waterman Local Alignment

Again, use dynamic programming

Same basic scheme as before except

• similarity matrix MUST include negative values for mismatches

and

• when the value calculated for a position in the scoring matrix is negative, the value is set to zero - **this terminates the alignment**
**Smith-Waterman:**

Write one sequence across the top, and one down along the side

<table>
<thead>
<tr>
<th></th>
<th>i = 0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>j =</td>
<td>Gap</td>
<td>V</td>
<td>D</td>
<td>S</td>
<td>C</td>
<td>Y</td>
</tr>
</tbody>
</table>

0  Gap
1  V  0
2  E  0
3  S  0
4  L  0
5  C  0
6  Y  0

Local alignments: Smith-Waterman

\[ S_{ij} = \max \{ S_{i-1, j-1} + \sigma(x_i, y_j) \text{ (diagonal)}, S_{i-1, j} - A \text{ (from left to right)}, S_{i, j-1} - A \text{ (from top to bottom)} \} \]
Need a metric of similarity between amino acid pairs

**Simplest metric – identity matrix**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>K</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>H</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>I</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>K</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

OK for nucleic acids, but for proteins can do substantially better

What properties should an amino acid similarity matrix have?

Refer to Z&B pp. 119-125
Scoring system should favor matching identical or related amino acids and penalize for poor matches and for gaps

Need to know how often a particular amino acid pair is found in related proteins compared with its occurrence by chance, and also how often gaps (insertions/deletions) are found in related proteins relative to dissimilar amino acid pairs
Scores and Evolution

Any alignment scoring system brings with it an implicit evolutionary model
Amino Acid Substitution Matrices

Margaret Dayhoff, 1978, PAM Matrices

**Explicit evolutionary model**
Assumes symmetry: $A \rightarrow B = B \rightarrow A$
Assumes amino acid substitutions observed over short periods of time can be extrapolated to long periods of time

71 groups of protein sequences, 85% similar
1572 amino acid changes.

Functional proteins $\rightarrow$ mutations “accepted” by natural selection

PAM1 matrix means 1% divergence between proteins - i.e.
1 amino acid change per 100 residues. Some texts re-state this as the probability of each amino acid changing into another is $\sim 1\%$ and probability of not changing is $\sim 99\%$
Construction of a Dayhoff Matrix: PAM1

Step 1: *Measure pairwise substitution frequencies* for each amino acid within families of related proteins that can be confidently aligned

... GDSFHYFVSHG ... .
... GDSFHYYVSGG ... .
... GDSYHYFVSFG ... .
... GDSHYFVSFG ... .
... GDSFHFFVSFG ... .

900 Phe (F) remained F
100 Phe (F) → 80 Tyr (Y), 3 Trp (W), 2 His (H)...

Gives $n_{ab}$, i.e. $n_{YF}=80$

$n_{WF}=3$

...in evolution

$n$ indicates raw count of events
DNA Sequence Evolution

**Generation n-1** (grandparent)

```
5’ TGGCATGCACCCTGTAAGTCAATATAAATGGCTACGCCTAGCCCATGCGA 3’
3’ ACCGTACGTGGGACATTCAGTTATATTTACCAGATCGGATCGGGGTACGCT 5’
```

**Generation n** (parent)

```
5’ TGGCATGCACCCTGTAAGTCAATATAAATGGCTATGCCTAGCCCATGCGA 3’
3’ ACCGTACGTGGGACATTCAGTTATATTTACCAGATACGGATCGGGGTACGCT 5’
```

**Generation n+1** (child)

```
5’ TGGCATGCACCCTGTAAGTCAATATAAATGGCTATGCCTAGCCCATGCGA 3’
3’ ACCGTACGTGGGACATTCAGTTATATTTACCAGATACGGATCGGGCAGACGCT 5’
```

Images of The Simpsons © FOX. All rights reserved. This content is excluded from our Creative Commons license. For more information, see [http://ocw.mit.edu/help/faq-fair-use/](http://ocw.mit.edu/help/faq-fair-use/).
Markov Model (aka Markov Chain)

Classical Definition

A discrete stochastic process $X_1, X_2, X_3, \ldots$ which has the Markov property:

$$P(X_{n+1} = j \mid X_1=x_1, X_2=x_2, \ldots X_n=x_n) = P(X_{n+1} = j \mid X_n=x_n)$$

(for all $x_i$, all $j$, all $n$)

In words:

A random process which has the property that the future (next state) is conditionally independent of the past given the present (current state)

Andrey Markov, a Russian mathematician (1856 - 1922)
Spring 2014

For information about citing these materials or our Terms of Use, visit: http://ocw.mit.edu/terms.