Giri Narasimhan
ECS 254; Phone: x3748
giri@cis.fiu.edu
www.cis.fiu.edu/~giri/teach/BioinfS13.html
Sequence Alignment
BLAST Variants

- **Nucleotide BLAST**
  - **Standard blastn**
  - **MEGABLAST** (Compare large sets, Near-exact searches)
  - **Short Sequences** (higher E-value threshold, smaller word size, no low-complexity filtering)

- **Protein BLAST**
  - **Standard blastp**
  - **PSI-BLAST** (Position Specific Iterated BLAST)
  - **PHI-BLAST** (Pattern Hit Initiated BLAST; reg expr. Or Motif search)
  - **Short Sequences** (higher E-value threshold, smaller word size, no low-complexity filtering, PAM-30)

- **Translating BLAST**
  - **Blastx**: Search nucleotide sequence in protein database (6 reading frames)
  - **Tblastn**: Search protein sequence in nucleotide dB
  - **Tblastx**: Search nucleotide seq (6 frames) in nucleotide DB (6 frames)
BLAST Cont’d

- **RPS BLAST**
  - Compare protein sequence against Conserved Domain DB; Helps in predicting rough structure and function

- **Pairwise BLAST**
  - blastp (2 Proteins), blastn (2 nucleotides), tblastn (protein-nucleotide w/ 6 frames), blastx (nucleotide-protein), tblastx (nucleotide w/6 frames-nucleotide w/ 6 frames)

- **Specialized BLAST**
  - Human & Other finished/unfinished genomes
  - *P. falciparum*: Search ESTs, STSs, GSSs, HTGs
  - VecScreen: screen for contamination while sequencing
  - IgBLAST: Immunoglobulin sequence database
BLAST Parameters and Output

- Type of sequence, nucleotide/protein
- Word size, \( w \)
- Gap penalties, \( p_1 \) and \( p_2 \)
- Neighborhood Threshold Score, \( T \)
- Score Threshold, \( S \)
- E-value Cutoff, \( E \)
- Number of hits to display, \( H \)
- Database to search, \( D \)
- Scoring Matrix, \( M \)
- Score \( s \) and E-value \( e \)

\( E \)-value \( e \) is the expected number of sequences that would have an alignment score greater than the current score \( s \).
How to score mismatches?

![Matrix Diagram](image-url)
## Scoring Matrix to Use

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAM 40</td>
<td>Short alignments with high similarity (70-90%)</td>
</tr>
<tr>
<td>PAM 160</td>
<td>Members of a protein family (50-60%)</td>
</tr>
<tr>
<td>PAM 250</td>
<td>Longer alignments (divergent sequences) (~30%)</td>
</tr>
<tr>
<td>BLOSUM90</td>
<td>Short alignments with high similarity (70-90%)</td>
</tr>
<tr>
<td>BLOSUM80</td>
<td>Members of a protein family (50-60%)</td>
</tr>
<tr>
<td>BLOSUM62</td>
<td>Finding all potential hits (30-40%)</td>
</tr>
<tr>
<td>BLOSUM30</td>
<td>Longer alignments (divergent sequences) (&lt;30%)</td>
</tr>
</tbody>
</table>
BLAST algorithm: Phase 1

Phase 1: get list of word pairs (w=3) above threshold T

Example: for a human RBP query

...FSGTWYA...

GTW is a word in this query sequence

A list of words (w=3) is:

FSG  SGT  GTW  TWY  WYA
YSG  TGT  ATW  SWY  WFA
FTG  SVT  GSW  TWF  WYS
Use BLOSUM to score word hits

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>R</th>
<th>N</th>
<th>D</th>
<th>C</th>
<th>Q</th>
<th>E</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>L</th>
<th>K</th>
<th>M</th>
<th>F</th>
<th>P</th>
<th>S</th>
<th>T</th>
<th>W</th>
<th>Y</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>4</td>
<td>-1</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R</td>
<td>4</td>
<td>-2</td>
<td>0</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>5</td>
<td>-2</td>
<td>1</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
</tr>
<tr>
<td>Q</td>
<td>5</td>
<td>0</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
</tr>
<tr>
<td>E</td>
<td>2</td>
<td>0</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
</tr>
<tr>
<td>G</td>
<td>6</td>
<td>0</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
</tr>
<tr>
<td>H</td>
<td>8</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>I</td>
<td>4</td>
<td>-2</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
<td>-3</td>
</tr>
<tr>
<td>L</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>K</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>F</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>S</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>T</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>W</td>
<td>11</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>V</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

1/28/13
Phase 1: Find list of similar words

- Find list of words of length \( w \) (here \( w = 3 \)) and distance at least \( T \) (here \( T = 11 \))

- GTW 22
- GSW 18
- ATW 16
- NTW 16
- GTY 13
- GNW 10
- GAW 9
BLAST: Phases 2 & 3

- Phase 2: Scan database for exact hits of similar words list and find HotSpots
- Phase 3:
  - Extend good hit in either direction.
  - Keep track of the score (use a scoring matrix)
  - Stop when the score drops below some cutoff.

KENFDKARFSGTWYAMAKKDPEG 50 RBP (query)
MKGLDIQKVAGTWYSŁAMAASD. 44 lactoglobulin (hit)

extend Hit! extend
BLAST: Threshold vs # Hits & Extensions
Word Size

- **Blastn**: $w = 7, 11, \text{ or } 15$.
  - $w=15$ gives fewer matches and is faster than $w=11$ or $w=7$.

- **Megablast**: $w = 28 \text{ to } 64$.
  - Megablast is VERY fast for finding closely related DNA sequences!
Scores: Follow Extreme Value Distribution

\[ E = Kmn \, e^{-\lambda S} \]

\( m, n = \text{seq length} \)

\( S = \text{Raw Score} \)

\( K \approx \text{Search space} \)

\[ S' = (\lambda S - \ln K) / \ln 2 \]

\( S' = \text{Bit Score} \)

\[ p = 1 - e^{-E} \]

\( p = \text{p-value} \)

1/28/13
CAP5510/CGS5166
## E-value versus P-value

<table>
<thead>
<tr>
<th>E-value</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.9999546</td>
</tr>
<tr>
<td>5</td>
<td>0.99326205</td>
</tr>
<tr>
<td>2</td>
<td>0.86466472</td>
</tr>
<tr>
<td>1</td>
<td>0.63212056</td>
</tr>
<tr>
<td>0.1</td>
<td>0.09516258</td>
</tr>
<tr>
<td>0.05</td>
<td>0.04877058</td>
</tr>
<tr>
<td>0.001</td>
<td>0.00099950</td>
</tr>
<tr>
<td>0.0001</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

E-values are easier to interpret; If query is short aa sequence, then use very large E-value; Sometimes even meaningful hits have large E-values.
BLAST: Steps

- Choose your sequence
- Choose your tool
- Choose your database
- Select parameters, if needed
- Interpret your results
NCBI Handbook, Eds. McEntyre, Ostell
Graphical Overview of BLAST Results

Distribution of 41 Blast Hits on the Query Sequence

Mouse-over to show defline and scores. Click to show alignments

Color Key for Alignment Scores

- <40
- 40-50
- 50-80
- 80-200
- >=200

1/28/13
List of hits with one line descriptions

<table>
<thead>
<tr>
<th>Sequence ID</th>
<th>Description</th>
<th>Score</th>
<th>E Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>gi</td>
<td>116365</td>
<td>sp</td>
<td>P26374</td>
</tr>
<tr>
<td>gi</td>
<td>121431807</td>
<td>sp</td>
<td>P24386</td>
</tr>
<tr>
<td>gi</td>
<td>585775</td>
<td>sp</td>
<td>P37727</td>
</tr>
<tr>
<td>gi</td>
<td>13626886</td>
<td>sp</td>
<td>Q61598</td>
</tr>
<tr>
<td>gi</td>
<td>729566</td>
<td>sp</td>
<td>P39958</td>
</tr>
<tr>
<td>gi</td>
<td>13626813</td>
<td>sp</td>
<td>O97556</td>
</tr>
<tr>
<td>gi</td>
<td>13638229</td>
<td>sp</td>
<td>P50397</td>
</tr>
<tr>
<td>gi</td>
<td>1707888</td>
<td>sp</td>
<td>P50398</td>
</tr>
<tr>
<td>gi</td>
<td>121108</td>
<td>sp</td>
<td>P21856</td>
</tr>
<tr>
<td>gi</td>
<td>21903424</td>
<td>sp</td>
<td>P50396</td>
</tr>
<tr>
<td>gi</td>
<td>13626812</td>
<td>sp</td>
<td>O97555</td>
</tr>
<tr>
<td>gi</td>
<td>1707886</td>
<td>sp</td>
<td>P31150</td>
</tr>
<tr>
<td>gi</td>
<td>13638228</td>
<td>sp</td>
<td>P50395</td>
</tr>
<tr>
<td>gi</td>
<td>1707891</td>
<td>sp</td>
<td>P50399</td>
</tr>
<tr>
<td>gi</td>
<td>1723467</td>
<td>sp</td>
<td>Q10305</td>
</tr>
<tr>
<td>gi</td>
<td>585776</td>
<td>sp</td>
<td>P32864</td>
</tr>
<tr>
<td>gi</td>
<td>10720243</td>
<td>sp</td>
<td>O93831</td>
</tr>
<tr>
<td>gi</td>
<td>2498411</td>
<td>sp</td>
<td>Q49398</td>
</tr>
<tr>
<td>gi</td>
<td>11135401</td>
<td>sp</td>
<td>Q9X8Q9</td>
</tr>
<tr>
<td>gi</td>
<td>11135075</td>
<td>sp</td>
<td>O05139</td>
</tr>
<tr>
<td>gi</td>
<td>11135195</td>
<td>sp</td>
<td>P57112</td>
</tr>
<tr>
<td>gi</td>
<td>122257022</td>
<td>sp</td>
<td>Q8TZJ8</td>
</tr>
<tr>
<td>gi</td>
<td>3915516</td>
<td>sp</td>
<td>P94488</td>
</tr>
<tr>
<td>gi</td>
<td>231788</td>
<td>sp</td>
<td>P30599</td>
</tr>
<tr>
<td>gi</td>
<td>2498412</td>
<td>sp</td>
<td>P75499</td>
</tr>
<tr>
<td>gi</td>
<td>547891</td>
<td>sp</td>
<td>P36225</td>
</tr>
<tr>
<td>gi</td>
<td>586602</td>
<td>sp</td>
<td>P37747</td>
</tr>
</tbody>
</table>
List of alignments
Pairwise alignment result of human beta globin and myoglobin

| Query 4 | LTPEEKSAVTALWGKVNVDVVG--GEALGRLLVVYYPWTQRELSFGLSTPDAVMGNPKV 61 |
| Sbjct 3 | LSDGEWQLVMLVNGKVEDLPGHGEVQLIRLFKGHPGTEKFDKFKHKLKSEDEMKASEDL 62 |
| Query 62 | KAHGKKVLGAFSDGLAHLDNLKCTFATLSELHCDKLHVPENFRLLLGVVLACVLAHHFGK 121 |
| Sbjct 63 | KKHGATVLTLGILKKGHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKHPG 122 |
| Query 122 | EFPPVQAAYQKVVAGVANALAHKY 146 |
| Sbjct 123 | DFGADAQGAMNKAELFRKDMASNY 147 |

Information about this alignment: score, expect value, identities, positives, gaps...

Myoglobin RefSeq

Slide: Courtesy J. Pevsner
Pairwise alignment result of human beta globin and myoglobin: the score is a sum of match, mismatch, gap creation, and gap extension scores.

Score = 18.1 bits (35), Expect = 0.015, Method: Composition-based stats. Identities = 11/24 (45%), Positives = 12/24 (50%), Gaps = 2/24 (8%)

Query 12 | VTALWGKVNVD--EVGGEALGRIL | 33
Sbjct 11 | VLNVGKVEADIPGQEVLRRL | 34

match | 4 11 5 6 4 6 5 4 5 | sum of matches: +60
mismatch | -1 1 0 -2 -2 -4 0 | sum of mismatches: -13
gap open | | sum of gap penalties: -12
gap extend | -11 -1 |

total raw score: 60 - 13 - 12 = 35
Pairwise alignment result of human beta globin and myoglobin: the score is a sum of match, mismatch, gap creation, and gap extension scores

Score = 18.1 bits (35), Expect = 0.015, Method: Composition-based stats.
Identities = 11/24 (45%), Positives = 12/24 (50%), Gaps = 2/24 (8%)

Query 12: VTALWGKVNVD---EVGGEALGRLL 33
Sbjct 11: VLNVWGKVEADIPGHHQEVFLRLF 34

<table>
<thead>
<tr>
<th>Match</th>
<th>4</th>
<th>11</th>
<th>5</th>
<th>6</th>
<th>4</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>sum of matches: +60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mismatch</td>
<td>-1</td>
<td>1</td>
<td>0</td>
<td>-2</td>
<td>-2</td>
<td>-2</td>
<td>-4</td>
<td>0</td>
<td>sum of mismatches: -13</td>
</tr>
<tr>
<td>Gap open</td>
<td>-2</td>
<td>-2</td>
<td>0</td>
<td>-3</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>sum of gap penalties: -12</td>
</tr>
<tr>
<td>Gap extend</td>
<td>-11</td>
<td>-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>total raw score: 60 - 13 - 12 = 35</td>
</tr>
</tbody>
</table>

V matching V earns +4
T matching L earns -1

These scores come from a “scoring matrix”!

Slide: Courtesy J. Pevsner
If $S$ is the (raw) score for a local alignment, the normalized score $S'$ (in bits) is given by

$$S' = \frac{\lambda S - \ln(K)}{\ln(2)}$$

The parameters $K$ and $\lambda$ depend on the scoring system.
E-value is **not a probability**, but describes strength of random background noise.

E-value describes **number of hits** one can “**expect**” to see by chance when searching a database of a particular size.

It decreases exponentially with the score (S).

**E-value = 1** means “in a database of current size, one might expect to see **one** match with a similar score simply by chance. Lower E-value mean more “**significant**” match.

**WARNING:** Short sequences can be virtually identical and have relatively high E-values.

- Calculation of E-value takes into account length of query sequence. Since shorter sequences have a high probability of occurring in the database purely by chance, E-values can be high.
Most sequences with significant similarity over their entire lengths are homologous.

Matches that are > 50% identical in a 20-40 aa region occur frequently by chance.

Distantly related homologs may lack significant similarity. Homologous sequences may have few absolutely conserved residues.

A homologous to B & B to C $\Rightarrow$ A homologous to C.

Low complexity regions, transmembrane regions and coiled-coil regions frequently display significant similarity without homology.

Greater evolutionary distance implies that length of a local alignment required to achieve a statistically significant score also increases.
Rules of Thumb

- Results of searches using different scoring systems may be compared directly using normalized scores.
- If $S$ is the (raw) score for a local alignment, the normalized score $S'$ (in bits) is given by
  
  $$S' = \frac{\lambda S - \ln(K)}{\ln(2)}$$

  The parameters depend on the scoring system.
- **Statistically significant normalized score,**

  $$S' > \log\left(\frac{N}{E}\right)$$

  where $E$-value $= E$, and $N$ = size of search space.
Assessing whether proteins are homologous

>gi|4505583|ref|NP_002562.1| progestagen-associated endometrial protein (placental protein 14, pregnancy-associated endometrial alpha-2-globulin, alpha uterine protein); Progestagen-associated endometrial protein (placental protein 14) [Homo sapiens]
>gi|190215|gb|AAA60147.1| (J04129) placental protein 14 [Homo sapiens]

Score = 32.0 bits (71), Expect = 0.49
Identities = 26/107 (24%), Positives = 48/107 (44%), Gaps = 11/107 (10%)

Query: 26  RVKENFDKARFSGTUYAMAKKDPEGLFLQDNIVAEFSVDETGQMSTAKGRVRLNNWD— 84
+ K++ + + +GTU++MA + L + A + V T + +L+ W+

Sbjct: 5  QTKQDLELPKLAGTUHSMAMAT---NNISLMATLKAPLRVHITSLLP TEDNLEIHLHRWEN 63

Query: 85  —VCADMVGTFTTDTEDPAKFKMKYWGVASFLQKGNDDHUIVDTDYDT 130
+ C + T +P KFK+ Y V A + + ++DTDYD +

Sbjct: 64  NSCVEKKVLGEKTGNPPKKFKINY—TVA------NEATL DTDYDNF 102

RBP4 and PAEP:
Low bit score, E value 0.49, 24% identity ("twilight zone"). But they are indeed homologous. Try a BLAST search with PAEP as a query, and find many other lipocalins.
Difficulties with BLAST

- Use human beta globin as a query against human RefSeq proteins, and blastp does not “find” human myoglobin. This is because the two proteins are too distantly related. PSI-BLAST at NCBI as well as hidden Markov models easily solve this problem.

- How can we search using 10,000 base pairs as a query, or even millions of base pairs? Many BLAST-like tools for genomic DNA are available such as PatternHunter, Megablast, BLAT, and BLASTZ.
Related Tools

- **Megablast**
  - For long, closely-related sequences
  - Uses large $w$ and is very fast

- **BLAT**
  - UCSC tool
  - DB broken into words; query is searched

- **PatternHunter**
  - Generalized seeds used instead of words

- **BLASTZ, Lagan, SSAHA**
Global Alignment: An example

Given
\[ \delta[I, J] = \text{Score of Matching the } I^{\text{th}} \text{ character of sequence } V & \text{ the } J^{\text{th}} \text{ character of sequence } W \]

Compute
\[ S[I, J] = \text{Score of Matching First } I \text{ characters of sequence } V \& \text{ First } J \text{ characters of sequence } W \]

Match/Mismatch score
Recurrence Relation
\[ S[I, J] = \text{MAXIMUM} \{ \]
\[ S[I-1, J-1] + \delta(V[I], W[J]), \]
\[ S[I-1, J] + \delta(V[I], -), \]
\[ S[I, J-1] + \delta(-, W[J]) \}\]

Gap Penalty
What happens with last character(s)?

1. Last characters MATCH
   
   \[
   \begin{array}{c}
   V \\
   W
   \end{array}
   \]

2. Last characters MISMATCH
   
   \[
   \begin{array}{c}
   V \\
   W
   \end{array}
   \]

3. Last character of W aligned with GAP
   
   \[
   \begin{array}{c}
   V \\
   W
   \end{array}
   \]

4. Last character of V aligned with GAP
   
   \[
   \begin{array}{c}
   V \\
   W
   \end{array}
   \]
How to fill in the matrix?

Add Match or Mismatch Score

Add gap penalty for gap in seq 1

Add gap penalty for gap in seq 2
Global Alignment: An example

V: G A A T T C A G T T A
W: G G A T C G A

S[I, J] = MAXIMUM {
S[I-1, J-1] + δ(V[I], W[J]),
S[I-1, J] + δ(V[I], ⎯),
S[I, J-1] + δ(⎯, W[J])}

Match score = 1; Mismatch = Gap = -1
Traceback

\[ S[i, j] = \text{MAXIMUM}\{ S[i-1, j-1] + \delta(V[i], W[j]), S[i-1, j] + \delta(V[i], \_\_\_), S[i, j-1] + \delta(\_\_, W[j]) \} \]
Alternative Traceback

**V:** G - A A T T T C A G T T A

**W:** G G - A - T C - G - - A

**V:** G A A T T T C A G T T A

**W:** G G A - T C - G - - A

Previous
### Improved Traceback

<table>
<thead>
<tr>
<th></th>
<th>G</th>
<th>A</th>
<th>A</th>
<th>T</th>
<th>T</th>
<th>C</th>
<th>A</th>
<th>G</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</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>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>×1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>×1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>×1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>×2</td>
<td>←2</td>
<td>←2</td>
<td>←2</td>
<td>←2</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>↑1</td>
<td>↑1</td>
<td>×2</td>
<td>←2</td>
<td>←2</td>
<td>←2</td>
<td>×2</td>
<td>↑2</td>
<td>↑2</td>
<td>↑2</td>
<td>×3</td>
</tr>
<tr>
<td>T</td>
<td>0</td>
<td>↑1</td>
<td>←2</td>
<td>↑2</td>
<td>×3</td>
<td>×3</td>
<td>←3</td>
<td>←3</td>
<td>×3</td>
<td>×3</td>
<td>↑3</td>
<td>↑3</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>↑2</td>
<td>↑3</td>
<td>↑3</td>
<td>×4</td>
<td>←4</td>
<td>←4</td>
<td>←4</td>
<td>←4</td>
<td>←4</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>↑2</td>
<td>↑3</td>
<td>↑3</td>
<td>↑4</td>
<td>↑4</td>
<td>×5</td>
<td>←5</td>
<td>←5</td>
<td>←5</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>×3</td>
<td>↑3</td>
<td>↑3</td>
<td>↑4</td>
<td>×5</td>
<td>↑5</td>
<td>↑5</td>
<td>↑5</td>
<td>×6</td>
</tr>
</tbody>
</table>
Improved Traceback

<table>
<thead>
<tr>
<th></th>
<th>G</th>
<th>A</th>
<th>A</th>
<th>T</th>
<th>T</th>
<th>C</th>
<th>A</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</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>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>×1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>×1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>×1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>×2</td>
<td>←2</td>
<td>←2</td>
<td>←2</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>↑1</td>
<td>↑1</td>
<td>×2</td>
<td>←2</td>
<td>←2</td>
<td>←2</td>
<td>×2</td>
<td>↑2</td>
<td>↑2</td>
<td>↑2</td>
<td>×3</td>
</tr>
<tr>
<td>T</td>
<td>0</td>
<td>↑1</td>
<td>←2</td>
<td>↑2</td>
<td>×3</td>
<td>×3</td>
<td>←3</td>
<td>←3</td>
<td>×3</td>
<td>×3</td>
<td>×3</td>
<td>↑3</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>↑2</td>
<td>↑3</td>
<td>↑3</td>
<td>×4</td>
<td>←4</td>
<td>←4</td>
<td>←4</td>
<td>←4</td>
<td>←4</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>↑2</td>
<td>↑3</td>
<td>↑3</td>
<td>↑4</td>
<td>↑4</td>
<td>×5</td>
<td>←5</td>
<td>←5</td>
<td>←5</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>×3</td>
<td>↑3</td>
<td>↑3</td>
<td>↑4</td>
<td>×5</td>
<td>↑5</td>
<td>↑5</td>
<td>↑5</td>
<td>×6</td>
</tr>
</tbody>
</table>
## Improved Traceback

<table>
<thead>
<tr>
<th></th>
<th>G</th>
<th>A</th>
<th>A</th>
<th>T</th>
<th>T</th>
<th>C</th>
<th>A</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>V:</td>
<td>G</td>
<td>A</td>
<td>-</td>
<td>A</td>
<td>T</td>
<td>T</td>
<td>C</td>
<td>A</td>
<td>G</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>W:</td>
<td>G</td>
<td>-</td>
<td>G</td>
<td>A</td>
<td>-</td>
<td>T</td>
<td>C</td>
<td>-</td>
<td>G</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>0</td>
<td>×1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>×1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
<td>←1</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>×1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>↑1</td>
<td>×2</td>
<td>←2</td>
<td>←2</td>
<td>←2</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>↑1</td>
<td>↑1</td>
<td>×2</td>
<td>←2</td>
<td>←2</td>
<td>×2</td>
<td>↑2</td>
<td>↑2</td>
<td>↑2</td>
<td>×3</td>
</tr>
<tr>
<td>T</td>
<td>0</td>
<td>↑1</td>
<td>←2</td>
<td>↑2</td>
<td>×3</td>
<td>×3</td>
<td>←3</td>
<td>←3</td>
<td>×3</td>
<td>×3</td>
<td>↑3</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>↑2</td>
<td>↑3</td>
<td>↑3</td>
<td>×4</td>
<td>←4</td>
<td>←4</td>
<td>←4</td>
<td>←4</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>↑2</td>
<td>↑3</td>
<td>↑3</td>
<td>↑4</td>
<td>↑4</td>
<td>×5</td>
<td>←5</td>
<td>←5</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>↑1</td>
<td>↑2</td>
<td>×3</td>
<td>↑3</td>
<td>↑3</td>
<td>↑4</td>
<td>×5</td>
<td>↑5</td>
<td>↑5</td>
<td>×6</td>
</tr>
</tbody>
</table>
Subproblems

- Optimally align $V[1..I]$ and $W[1..J]$ for every possible values of $I$ and $J$.

- Having optimally aligned
  - $V[1..I-1]$ and $W[1..J-1]$
  - $V[1..I]$ and $W[1..J-1]$
  - $V[1..I-1]$ and $W[1, J]$

  it is possible to optimally align $V[1..I]$ and $W[1..J]$.

- $O(mn)$,
  where $m = \text{length of } V$,
  and $n = \text{length of } W$. 

1/28/13
CAP5510/CGS5166
Generalizations of Similarity Function

- **Mismatch Penalty** = $\alpha$
- **Spaces (Insertions/Deletions, InDels)** = $\beta$
- **Affine Gap Penalties:**
  - (Gap open, Gap extension) = ($\gamma$, $\delta$)
- **Weighted Mismatch** = $\Phi(a,b)$
- **Weighted Matches** = $\Omega(a)$
**Alternative Scoring Schemes**

<table>
<thead>
<tr>
<th></th>
<th>G</th>
<th>A</th>
<th>A</th>
<th>T</th>
<th>T</th>
<th>C</th>
<th>A</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-2</td>
<td>-3</td>
<td>-4</td>
<td>-5</td>
<td>-6</td>
<td>-7</td>
<td>-8</td>
<td>-9</td>
<td>-10</td>
<td>-11</td>
<td>-12</td>
</tr>
<tr>
<td>G</td>
<td>-2</td>
<td>×1</td>
<td>←-1</td>
<td>←-2</td>
<td>←-3</td>
<td>←-4</td>
<td>←-5</td>
<td>←-6</td>
<td>←-7</td>
<td>←-8</td>
<td>←-9</td>
</tr>
<tr>
<td>G</td>
<td>-3</td>
<td>↑-1</td>
<td>×-1</td>
<td>←-3</td>
<td>←-4</td>
<td>←-5</td>
<td>←-6</td>
<td>←-7</td>
<td>×-5</td>
<td>←-7</td>
<td>←-8</td>
</tr>
<tr>
<td>A</td>
<td>-4</td>
<td>↑-2</td>
<td>×0</td>
<td>×0</td>
<td>←-2</td>
<td>←-3</td>
<td>←-4</td>
<td>←-5</td>
<td>←-6</td>
<td>←-7</td>
<td>×-7</td>
</tr>
<tr>
<td>T</td>
<td>-5</td>
<td>↑-3</td>
<td>↑-2</td>
<td>↑-2</td>
<td>×1</td>
<td>←-1</td>
<td>←-2</td>
<td>←-3</td>
<td>←-4</td>
<td>←-5</td>
<td>←-6</td>
</tr>
<tr>
<td>C</td>
<td>-6</td>
<td>↑-4</td>
<td>↑-3</td>
<td>↑-3</td>
<td>↑-1</td>
<td>×-1</td>
<td>×0</td>
<td>←-2</td>
<td>←-3</td>
<td>←-4</td>
<td>←-5</td>
</tr>
<tr>
<td>G</td>
<td>-7</td>
<td>↑-5</td>
<td>↑-4</td>
<td>↑-4</td>
<td>↑-2</td>
<td>↑-3</td>
<td>↑-2</td>
<td>×-2</td>
<td>×-1</td>
<td>←-3</td>
<td>←-4</td>
</tr>
<tr>
<td>A</td>
<td>-8</td>
<td>↑-6</td>
<td>↑-5</td>
<td>↑-5</td>
<td>↑-3</td>
<td>↑-4</td>
<td>↑-3</td>
<td>×-1</td>
<td>↑-3</td>
<td>×-3</td>
<td>×-3</td>
</tr>
</tbody>
</table>

- **Match +1**
- **Mismatch -2**
- **Gap (-2, -1)**

<table>
<thead>
<tr>
<th>V:</th>
<th>G</th>
<th>A</th>
<th>A</th>
<th>T</th>
<th>T</th>
<th>C</th>
<th>A</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| W:  | G   | G   | A   | T   | -   | C   | -   | G   | -   | -   | A   |

1/28/13

CAP5510/CGS5166 43
Local Sequence Alignment

Example: comparing long stretches of anonymous DNA; aligning proteins that share only some motifs or domains.

Smith-Waterman Algorithm
Recurrence Relations
(Global vs Local Alignments)

- \( S[I, J] = \text{MAXIMUM} \{ \)
  \( S[I-1, J-1] + \delta(V[I], W[J]), \)
  \( S[I-1, J] + \delta(V[I], -), \)
  \( S[I, J-1] + \delta(-, W[J]) \} \)

- \( S[I, J] = \text{MAXIMUM} \{ 0, \)
  \( S[I-1, J-1] + \delta(V[I], W[J]), \)
  \( S[I-1, J] + \delta(V[I], -), \)
  \( S[I, J-1] + \delta(-, W[J]) \} \)
Local Alignment: Example

Match +1
Mismatch −1
Gap (-1, -1)
Properties of Smith-Waterman Algorithm

- **How to find all regions of “high similarity”?**
  - Find all entries above a threshold score and traceback.

- **What if: Matches = 1 & Mismatches/spaces = 0?**
  - Longest Common Subsequence Problem

- **What if: Matches = 1 & Mismatches/spaces = -∞?**
  - Longest Common Substring Problem

- **What if the average entry is positive?**
  - Global Alignment
Calculation of an alignment score

\[ S = \sum \text{(identities, mismatches)} - \sum \text{(gap penalties)} \]

\[ \text{Score} = \text{Max}(S) \]

How to score mismatches?

![Blosum62 scoring matrix](image)

**Slide: Courtesy J. Pevsner**
For each amino acid pair a, b

For each BLOCK

- Align all proteins in the BLOCK
- Eliminate proteins that are more than n% identical
- Count $F(a)$, $F(b)$, $F(a,b)$
- Compute Log-odds Ratio

$$\log\left(\frac{F(a,b)}{F(a)F(b)}\right)$$
### Scoring Matrix to Use

- **PAM 40**: Short alignments with high similarity (70-90%)
- **PAM 160**: Members of a protein family (50-60%)
- **PAM 250**: Longer alignments (divergent sequences) (~30%)
- **BLOSUM90**: Short alignments with high similarity (70-90%)
- **BLOSUM80**: Members of a protein family (50-60%)
- **BLOSUM62**: Finding all potential hits (30-40%)
- **BLOSUM30**: Longer alignments (divergent sequences) (<30%)
Rat versus mouse globin

Rat versus bacterial globin

More conserved

Less conserved

Slide: Courtesy J. Pevsner
Local/Standalone BLAST

- Right click on a desired archive and select "Save link as..." from the popup menu
- In the prompt, switch to a desired directory (folder) and click the "Save" button to save the archive to a desired location on the local disk
- Installation details are at:
- With the help of this installation, you can run BLAST with preformatted databases or format your own database before you run BLAST queries.
Multiple Sequence Alignment
Multiple Alignments

- **Global**
  - ClustalW, ClustalX
  - MSA
  - T-Coffee

- **Local**
  - BLOCKS
  - eMOTIF
  - GIBBS
  - HMMER
  - MACAW
  - MEME

- **Other**
  - Profile Analysis from msa (UCSD)
  - SAM HMM (from msa)
<table>
<thead>
<tr>
<th></th>
<th>fly</th>
<th>human</th>
<th>plant</th>
<th>bacterium</th>
<th>yeast</th>
<th>archaeon</th>
</tr>
</thead>
<tbody>
<tr>
<td>amino acid sequence</td>
<td>GAKKVIISAP SAD...F VCGVNLDAVK PDMKVSNAS CTTNCLAPLA</td>
<td>GAKRVIISAP SAD...F VMGVNHEKYD NSSLKIISNAS CTTNCLAPLA</td>
<td>GAKKVIISAP SAD...F VGVGNEHTYQ PNMDIVSNAS CTTNCLAPLA</td>
<td>GAKKVVMTGP SKDNTPM...F VKGANFDKY. AGQDIVSNAS CTTNCLAPLA</td>
<td>GAKKVVITAP SS...F VMGVNEEKYT SDLKIVSNAS CTTNCLAPLA</td>
<td>GADKVLISAP PKGDEPVKQL VYGVNHDEYD GE.DVSNAS CTTNTSITPVA</td>
</tr>
<tr>
<td></td>
<td>fly</td>
<td>human</td>
<td>plant</td>
<td>bacterium</td>
<td>yeast</td>
<td>archaeon</td>
</tr>
<tr>
<td></td>
<td>KVINDNFEIV EGLMTTVHAT TATQKTVDGP SGKLWRDGRG AAQNIIPAST</td>
<td>KVIHDNFGIV EGLMTTVHAI TATQKTVDGP SGKLWRDGRG ALQNIIPAST</td>
<td>KVVHEEFGIL EGLMTTVHAT TATQKTVDGP SMKDWRGGRG ASQNIIPSST</td>
<td>KVINDNFGII EGLMTTVHAT TATQKTVDGP SHKDWRGGGRG ASQNIIPSST</td>
<td>KVINDAFGIE EGLMTTVHSL TATQKTVDGP SHKDWRGGRT ASQNIIPSST</td>
<td>KVLDEEFGIN AGQLTTVHAY TGSQNLMDGP NGKP.RRRRA AAENIIPTST</td>
</tr>
<tr>
<td></td>
<td>fly</td>
<td>human</td>
<td>plant</td>
<td>bacterium</td>
<td>yeast</td>
<td>archaeon</td>
</tr>
<tr>
<td></td>
<td>GAAKAVGKVI PALNGKLTGM AFRVPTPNVS VVDLTVRLGK GASYDEIKAK</td>
<td>GAAKAVGKVI PELNGKLTGM AFRVPTANVS VVDLTCRLEK PAKYDDIKKV</td>
<td>GAAKAVGKVL PELNGKLTGM AFRVPTSNVS VVDLTCRLEK GASYEDVKAA</td>
<td>GAAKAVGKVL PELNGKLTGM AFRVPTPNVS VVDLTVRLEK AATYEQIKAA</td>
<td>GAAKAVGKVL PELQGKLTGM AFRVPTVDVS VVDLTVKLNK ETYDEIKKV</td>
<td>GAAQAATEVL PELEGKLDGM AIRVPVPNVS ITEFVVDLDD DVTESTDNAA</td>
</tr>
</tbody>
</table>
Multiple Alignments: CLUSTALW

* identical
: conserved substitutions
. semi-conserved substitutions

<table>
<thead>
<tr>
<th>Accession</th>
<th>Sequence 1</th>
<th>Sequence 2</th>
<th>Sequence 3</th>
<th>Sequence 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>gi</td>
<td>2213819</td>
<td>CDN-ELKSEAIIEHLCASEFALR------------MKIKEVKKENGDKK 223</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gi</td>
<td>12656123</td>
<td>------ELKSEAIIEHLCASEFALR------------MKIKEVKKENG- 31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gi</td>
<td>7512442</td>
<td>CKNKNDNDNDIMETLCKNDFALK-------------IKVKEITYNRDTK 211</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gi</td>
<td>1344282</td>
<td>QDECKFDYVEYYETSSGAFSLLGRFCGAEPPLVSSSHHELAVLFRTDH 400</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Red: AVFPMLW (Small & hydrophobic)
Blue: DE (Acidic)
Magenta: RHK (Basic)
Green: STYHCNGQ (Hydroxyl, Amine, Basic)
Gray: Others
A. Estimate the amino acid frequencies in the motif columns of all but one sequence. Also obtain background.

Random start positions chosen

Location of motif in each sequence provides first estimate of motif composition
How to Score Multiple Alignments?

- **Sum of Pairs Score (SP)**
  - Optimal alignment: $O(d^N)$ [Dynamic Prog]
  - Approximate Algorithm: **Approx Ratio 2**
    - Locate Center: $O(d^2N^2)$
    - Locate Consensus: $O(d^2N^2)$

**Consensus char**: char with min distance sum

**Consensus string**: string of consensus char

**Center**: input string with min distance sum
Multiple Alignment Methods

- **Phylogenetic Tree Alignment (NP-Complete)**
  - Given tree, task is to label leaves with strings

- **Iterative Method(s)**
  - Build a MST using the distance function

- **Clustering Methods**
  - Hierarchical Clustering
  - K-Means Clustering
Multiple Alignment Methods (Cont’d)

- **Gibbs Sampling Method**

- **Hidden Markov Model**
Multiple Sequence Alignments (MSA)

- **Choice of Scoring Function**
  - Global vs local
  - Gap penalties
  - Substitution matrices
  - Incorporating other information
  - Statistical Significance

- **Computational Issues**
  - Exact/heuristic/approximate algorithms for optimal MSA
  - Progressive/Iterative/DP
  - Iterative: Stochastic/Non-stochastic/Consistency-based

- **Evaluating MSAs**
  - Choice of good test sets or benchmarks (BAliBASE)
  - How to decide thresholds for good/bad alignments
Progressive MSA: CLUSTALW

This example shows how a progressive alignment strategy can be misled. In the initial alignment of sequences 1 and 2, ClustalW has a choice between aligning CAT with CAT and making an internal gap or making a mismatch between C and F and having a terminal gap. Since terminal gaps are much cheaper than internals, the ClustalW scoring schemes prefers the former. In the next stage, when the extra sequence is added, it turns out that properly aligning the two CATs in the previous stage would have led to a better scoring sums-of-pairs multiple alignment.

### Table 1. Some recent and less recent available methods for MSAs.

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>Website</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSA</td>
<td>Exact</td>
<td><a href="http://www.ibc.wustl.edu/ibc/msa.html">http://www.ibc.wustl.edu/ibc/msa.html</a></td>
<td>[28]</td>
</tr>
<tr>
<td>OMA</td>
<td>Iterative DCA</td>
<td><a href="http://bibiserv.techfak.uni-bielefeld.de/oma">http://bibiserv.techfak.uni-bielefeld.de/oma</a></td>
<td>[61]</td>
</tr>
<tr>
<td>MultAln</td>
<td>Progressive</td>
<td><a href="http://www.toulouse.inra.fr/multalin.html">http://www.toulouse.inra.fr/multalin.html</a></td>
<td>[41]</td>
</tr>
<tr>
<td>ComAlign</td>
<td>Consistency-based</td>
<td><a href="http://www.daimi.au.dk/~ocaprani">http://www.daimi.au.dk/~ocaprani</a></td>
<td>[75]</td>
</tr>
<tr>
<td>Praline</td>
<td>Iterative/progressive</td>
<td><a href="mailto:jhering@imr.mrc.ac.uk">jhering@imr.mrc.ac.uk</a></td>
<td>[48]</td>
</tr>
<tr>
<td>HMMER</td>
<td>Iterative/Stochastic/HMM</td>
<td><a href="http://hmmer.wustl.edu/">http://hmmer.wustl.edu/</a></td>
<td>[68]</td>
</tr>
<tr>
<td>GA</td>
<td>Iterative/Stochastic/GA</td>
<td><a href="mailto:czhang@watnow.uwaterloo.ca">czhang@watnow.uwaterloo.ca</a></td>
<td>[52]</td>
</tr>
</tbody>
</table>

**C. Notredame, Pharmacogenomics, 3(1), 2002.**
MSA: Conclusions

- Very important
  - Phylogenetic analyses
  - Identify members of a family
  - Protein structure prediction
- No perfect methods
- Popular
  - Progressive methods: CLUSTALW
  - Recent interesting ones: Prrp, SAGA, DiAlign, T-Coffee
- Review of Methods [C. Notredame, Pharmacogenomics, 3(1), 2002]
  - CLUSTALW works reasonably well, in general
  - DiAlign is better for sequences with long insertions & deletions (indels)
  - T-Coffee is best available method