Introduction to Bioinformatics

Jianlin Cheng, PhD

Department of Computer Science
Informatics Institute

Mizzou
University of Missouri

2011
Topics

• Introduction
• Biological Sequence Alignment and Database Search
• Analysis of gene expression data
What’s Bioinformatics?

An interdisciplinary science of developing and applying computational techniques to address problems in molecular biology

- Develop bioinformatics algorithms and tools
- Apply bioinformatics tools to address biological problems
History of Bioinformatics

How does a new interdisciplinary science emerge?

Natural Sciences

- Physics
- Biochemistry
- Genetics
- Molecular Biology

1950s

- Watson, Crick, Perutz, Kendrew, Pauling, Sanger...

1960s

1970s

1980s

1990s

2000s

Engineering/Math

- Electrical Engineering
- Mathematician

1960s

1970s

1980s

1990s

2000s

Computer and Information Science

- Von Newman, Shannon, Knuth, Dijkstra...

Experimental Science Information Science

Comp/Info Approach to Bio

Bioinformatics
Genome Sequencing

![Diagram showing the scale of genome sequencing across different species, from viruses to mammals.](image)
High-Throughput Sequencing

• Transcriptome (EST, RNA-Seq, Chip-Seq)
• Proteomics (Mass Spectrometry)
• Metabolomics
Growth of GenBank
(1982 - 2008)

Sequences (millions)

Base Pairs of DNA (billions)


Blue: Base Pairs
Red: Sequences
What can we do with these huge amount of data?

Find buried treasure - Doug Brutlag, 1999.
Typical Bioinformatics Problems

• What family does this gene / protein belong to?
• Are there other known homologous proteins?
• What is the function and structure of this protein?
• What biological pathway does this protein participate in?
• Is a mutation on a gene / protein related to a phenotype or disease?
• Is a gene differentially expressed in a biological condition?
Fundamental Problems: Sequence Comparison

• Why do we compare sequences?
• What’s similarity between two sequences?
• How to compare sequences?
• Is similarity significant?
Importance of Similarity Comparison

• Identify evolutionary relationship between genes and proteins
• Similar genes/proteins have similar function
• Similar proteins have similar structures
Global Pairwise Sequence Alignment

Alignment (similarity) score
Three Main Issues

1. Definition of alignment score
2. Algorithms of finding the optimal alignment
3. Evaluation of significance of alignment score
A simple scoring scheme

- Score of character pair: $S(\text{match}) = 1$, $S(\text{not\_match}) = -1$, $S(\text{gap\_char}) = -1$
- Score of an alignment = $\sum_{i=1}^{n} S_i$

ITAKPAKTPTSPKEQAIGLSVTFLSFLLPAGWVLYHL
ITAKPQWLSKTE--------SVTFLSFLLPQTQGLYHL

$5 - 7 - 7 + 10 - 4 + 4 = 1$
Optimization

• How can we find the best alignment to maximize alignment score?
• How many possible alignments exist for two sequences with length m and n?
Total Number of Possible Alignments

AGATCAGAAAT–G
--AT–AG–AATCC

m + n
Total Number of Alignments

Select $m$ positions out of $m+n$ possible positions:

\[
\binom{m+n}{m} = \frac{(m+n)!}{m!n!}
\]

Exponential!

If $m = 300$, $n = 300$, total $= 10^{37}$
Divide and Conquer

Goal: align prefix $P[1..i]$ and prefix $Q[1..j]$

\[ \begin{align*}
\text{Seq } P & : \textcolor{red}{\text{AGATCAGAAATGG}} \\
\text{Seq } Q & : \textcolor{red}{\text{ATAGAATCC}}
\end{align*} \]

Three possibilities assuming we know the optimal alignment of smaller prefixes:

- **Case 1**
  Use alignment of $P[1..i-1]$ and $Q[1..j-1]$, pair $P[i]$ and $Q[j]$.

  \[ \begin{array}{c}
  i \\
  \text{AGATCAG} \\
  \text{--AT--AG} \\
  j
  \end{array} \]

- **Case 2**
  Use alignment of $P[1..i]$ and $Q[1..j-1]$, pair $Q[j]$ with gap.

  \[ \begin{array}{c}
  i \\
  \text{AGATCAG} \\
  --AT-A-- \\
  j
  \end{array} \]

- **Case 3**
  Use alignment of $P[1..i-1]$ and $Q[1..j]$, pair $P[i]$ with gap.

  \[ \begin{array}{c}
  i \\
  \text{AGATCA--G} \\
  --AT--AG-- \\
  j
  \end{array} \]
Needleman and Wunsch Algorithm

• Given sequences P and Q, we use a matrix M to record the optimal alignment scores of all prefixes of P and Q. M[i,j] is the best alignment score for the prefixes P[1..i] and Q[1..j].
• \[ M[i,j] = \max \left[ M[i-1,j-1] + S(P[i],Q[j]), \right. \]
  \[ M[i,j-1] + S(-, Q[j]), \]
  \[ M[i-1,j] + S(P[i], -) \]
}

Dynamic Programming
Dynamic Programming Algorithm

Three-Step Algorithm:

• Initialization
• Matrix fill (scoring)
• Trace back (alignment)
1. Initialization of Matrix $M$

\[
\begin{array}{cccccccc}
  & A & T & A & G & A & A & T \\
- & 0 & -1 & -2 & -3 & -4 & -5 & -6 & -7 \\
A & -1 & & & & & & & \\
G & -2 & & & & & & & \\
A & -3 & & & & & & & \\
T & -4 & & & & & & & \\
C & -5 & & & & & & & \\
A & -6 & & & & & & & \\
G & -7 & & & & & & & \\
A & -8 & & & & & & & \\
A & -9 & & & & & & & \\
A & -10 & & & & & & & \\
T & -11 & & & & & & & \\
G & -12 & & & & & & & \\
\end{array}
\]
2. Fill Matrix

\[
\begin{array}{cccccccc}
\_ & A & T & A & G & A & A & T \\
\_ & 0 & -1 & -2 & -3 & -4 & -5 & -6 & -7 \\
A & -1 & 1 & & & & & & \\
G & & 0 & & & & & & \\
A & -2 & & -1 & & & & & \\
T & -3 & & & -1 & & & & \\
C & -4 & & & & & & & \\
A & -5 & & & & & & & \\
G & -6 & & & & & & & \\
A & -7 & & & & & & & \\
A & -8 & & & & & & & \\
A & -9 & & & & & & & \\
A & -10 & & & & & & & \\
T & -11 & & & & & & & \\
G & -12 & & & & & & & \\
\end{array}
\]

\( M[i,j] = \max \left[ M[i-1,j-1] + S(P[i],Q[j]), \right. \\
\left. M[i,j-1] + S(\_ , Q[j]), \right. \\
\left. M[i-1,j] + S(P[i], \_ ) \right] \)
## 2. Fill Matrix

![Fill Matrix Diagram]

The matrix is filled according to the following equation:

\[
M[i,j] = \max \left[ M[i-1,j-1] + S(P[i],Q[j]), M[i,j-1] + S(-, Q[j]), M[i-1,j] + S(P[i], -) \right]
\]

The table shows the matrix with the cells filled according to the given equation. The arrows indicate the direction of the dynamic programming calculation.
## 2. Fill Matrix

\[
M[i,j] = \max \left[ M[i-1,j-1] + S(P[i], Q[j]), M[i-1,j] + S(-, Q[j]), M[i,j-1] + S(P[i], -) \right]
\]

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>T</th>
<th>A</th>
<th>G</th>
<th>A</th>
<th>A</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>-1</td>
<td>-2</td>
<td>-3</td>
<td>-4</td>
<td>-5</td>
<td>-6</td>
</tr>
<tr>
<td>A</td>
<td>-1</td>
<td>1</td>
<td>0</td>
<td>-1</td>
<td>-2</td>
<td>-3</td>
<td>-4</td>
</tr>
<tr>
<td>G</td>
<td>-2</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-2</td>
</tr>
<tr>
<td>A</td>
<td>-3</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>-1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>T</td>
<td>-4</td>
<td>-2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>-5</td>
<td>-3</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>A</td>
<td>-6</td>
<td>-4</td>
<td>-2</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>-7</td>
<td>-5</td>
<td>-3</td>
<td>-1</td>
<td>1</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>A</td>
<td>-8</td>
<td>-6</td>
<td>-4</td>
<td>-2</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>A</td>
<td>-9</td>
<td>-7</td>
<td>-5</td>
<td>-3</td>
<td>-1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>A</td>
<td>-10</td>
<td>-8</td>
<td>-6</td>
<td>-4</td>
<td>-2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>T</td>
<td>-11</td>
<td>-9</td>
<td>-7</td>
<td>-5</td>
<td>-3</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>G</td>
<td>-12</td>
<td>-10</td>
<td>-8</td>
<td>-6</td>
<td>-4</td>
<td>-2</td>
<td>0</td>
</tr>
</tbody>
</table>
### 3. Trace Back

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

**AGATCAGAAATG**  
**--AT-AG-AAT--**
Local vs. Global Alignment

- **Global Alignment**
  
  ```
  --T--CC-C-AGT--TATGT-CAGGGGACACG--A-GCATGCAGA-GAC
  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
  AATTGCCGCCC-GTCGT-T-TTCAG----CA-GTTATG-T-CAGAT--C
  ```

- **Local Alignment**—better alignment to find conserved segment

  ```
  tccCAGTTATGTCAGgggacacgagcatgcagagac
  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
  aattgccgccccgtcgttttcagCAGTTATGTCAGatc
  ```

Transcription binding site
Smith-Waterman Algorithm

Same dynamic program algorithm as global alignment except for three differences.

1. All negative scores is converted to 0
2. Alignment can start from anywhere in the matrix
3. Alignment can end at anywhere in the matrix
Application Example (Alignment – Structure)

TARGET

ASILPKRLFGNCEQSTSDEGLK
IERTPLVPHISAQNVCLKIDD
VPERLIPERASFQWMNDK

TEMPLATE

ASILPKRLFGNCEQSTSDEGLKIERTPLVPHISAQNVCLKIDDVPERLIPE
MSVIPKRLYGNCEQTSSEAIRIEDSPIV---TADLVCLKIDEIFERLVGE

Source: A. Fisher, 2005
Global and Local Alignment Tools

• NEEDLE (global alignment)
  http://bioweb.pasteur.fr/seqanal/interfaces/needle.html

• WATER (local alignment)
  http://bioweb.pasteur.fr/seqanal/interfaces/water.html
How to accurately measure the similarity between amino acids (or nucleotides) is one key issue of sequence alignment.

For nucleotides, a simple identical / not identical scheme is mostly ok.

Due to various properties of amino acids, it is hard and also critical to measure the similarity between amino acids.
Evolutionary Substitution Approach

- During evolution, the substitution of similar (or dissimilar) amino acids is more (or less) likely to be selected within protein families than random substitutions (M. Dayhoff)
- The frequency / probability one residue substitutes another one is an indicator of their similarity.
PAM Scoring Matrices
(M. Dayhoff)

- Select a number of protein families.
- Align sequences in each family and count the frequency of amino acid substitution of each column. The frequency is used to compute the empirical substitution probability of which residue $i$ substitutes residue $j$ ($P_{ij}$).
- Similarity score is ratio of observed substitution probability over the random substitution probability. $S(i,j) = \log(P_{ij} / (P_i * P_j))$. $P_i$ is the observed probability of residue $i$ and $P_j$ is the observed probability of residue $j$.
- PAM: Point Accepted Mutation
A Simplified Example

<table>
<thead>
<tr>
<th>Chars</th>
<th>Prob.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>6 / 10</td>
</tr>
<tr>
<td>C</td>
<td>1 / 10</td>
</tr>
<tr>
<td>G</td>
<td>2 / 10</td>
</tr>
<tr>
<td>T</td>
<td>1 / 10</td>
</tr>
</tbody>
</table>

Substitution Frequency Table

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>G</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>30</td>
<td>6</td>
<td>12</td>
<td>6</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>G</td>
<td>12</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>T</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Total number of substitutions: 90

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>G</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>.33</td>
<td>.07</td>
<td>.14</td>
<td>.07</td>
</tr>
<tr>
<td>C</td>
<td>.07</td>
<td>0</td>
<td>.02</td>
<td>.01</td>
</tr>
<tr>
<td>G</td>
<td>.14</td>
<td>.02</td>
<td>.01</td>
<td>.02</td>
</tr>
<tr>
<td>T</td>
<td>.07</td>
<td>.01</td>
<td>.02</td>
<td>0</td>
</tr>
</tbody>
</table>

\[ P(A\leftarrow\rightarrow C) = 0.07 + 0.07 = 0.14 \]
A Simplified Example

<table>
<thead>
<tr>
<th>Chars</th>
<th>Prob.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>6 / 10</td>
</tr>
<tr>
<td>C</td>
<td>1 / 10</td>
</tr>
<tr>
<td>G</td>
<td>2 / 10</td>
</tr>
<tr>
<td>T</td>
<td>1 / 10</td>
</tr>
</tbody>
</table>

**Substitution Frequency Table**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>G</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>30</td>
<td>6</td>
<td>12</td>
<td>6</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>G</td>
<td>12</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>T</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Total number of substitutions: 90

P(A<->C) = 0.07 + 0.07 = 0.14

S(A,C) = log(0.14/(0.6*0.1)) = 0.36
|   | C  | S  | T  | P  | A  | G  | N  | D  | E  | Q  | H  | R  | K  | M  | I  | L  | V  | F  | Y  | W  |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| C | 12 |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| S | 0  | 2  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| T | -2 | 1  | 3  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| P | -3 | 1  | 0  | 6  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| A | -2 | 1  | 1  | 1  | 2  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| G | -3 | 1  | 0  | -1 | 1  | 5  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| N | -4 | 1  | 0  | -1 | 0  | 0  | 2  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| D | -5 | 0  | 0  | -1 | 0  | 1  | 2  | 4  |    |    |    |    |    |    |    |    |    |    |    |    |    |
| E | -5 | 0  | 0  | -1 | 0  | 0  | 1  | 3  | 4  |    |    |    |    |    |    |    |    |    |    |    |    |
| Q | -5 | -1 | -1 | 0  | 0  | -1 | 1  | 2  | 2  | 4  |    |    |    |    |    |    |    |    |    |    |    |
| H | -3 | -1 | -1 | 0  | -1 | -2 | 2  | 1  | 1  | 3  | 6  |    |    |    |    |    |    |    |    |    |    |
| R | -4 | 0  | -1 | 0  | -2 | -3 | 0  | -1 | -1 | 1  | 2  | 6  |    |    |    |    |    |    |    |    |    |
| K | -5 | 0  | 0  | -1 | -1 | -2 | 1  | 0  | 0  | 1  | 0  | 3  | 5  |    |    |    |    |    |    |    |    |
| M | -5 | -2 | -1 | -2 | -1 | -3 | -2 | -2 | -2 | -1 | -2 | 0  | 0  | 6  |    |    |    |    |    |    |    |
| I | -2 | -1 | 0  | -2 | -1 | -3 | -2 | -2 | -2 | -2 | -2 | 2  | 5  |    |    |    |    |    |    |    |    |
| L | -6 | -3 | -2 | -3 | -2 | -4 | -3 | -4 | -3 | -2 | -2 | -3 | 3  | 4  | 2  | 6  |    |    |    |    |    |
| V | -2 | -1 | 0  | -1 | 0  | -1 | -2 | -2 | -2 | -2 | -2 | 2  | 4  | 2  | 4  |    |    |    |    |    |    |
| F | -4 | -3 | -3 | -5 | -4 | -5 | -3 | -6 | -5 | -5 | -2 | -4 | -5 | 0  | 1  | 2  | -1 | 9  |    |    |
| Y | 0  | -3 | -3 | -5 | -3 | -5 | -2 | -4 | -4 | -4 | -2 | -1 | -1 | 2  | 7  | 10 |    |    |    |    |    |
| W | -8 | -2 | -5 | -6 | -6 | -7 | -4 | -7 | -7 | -5 | -3 | 2  | -3 | -4 | -5 | -2 | -6 | 0  | 0  | 17 |

PAM250 Matrix (log odds multiplied by 10)
BLOSUM Matrices
(Henikoff and Henikoff)

- PAM matrices don’t work well for aligning evolutionarily divergent sequences.
- BLOSUM: BLOcks SUbstitution Matrix
- PAM based on observed mutations throughout global alignment. BLOSUM based on highly conserved local regions /blocks without gaps.
- BLOSUMn is a matrix calculated from proteins share at most n% identity. BLOSUM62 is the most widely used matrix (BLAST, PSI-BLAST, CLUSTALW)
Significance of Sequence Alignment

• Why do we need significant test?
• Mathematical view: unusual versus “by chance”
• Biological view: evolutionary related or not?
Randomization Approach

- Randomization is a fundamental idea due to Fisher.
- Randomly permute chars within sequence P and Q to generate new sequences (P’ and Q’). Align new sequences and record alignment scores.
- Assuming these scores obey normal distribution, compute mean (μ) and standard derivation (σ) of alignment scores.
Normal distribution of alignment scores of two sequences

• If $S = u + 2\sigma$, the probability of observing the alignment score equal to or more extreme than this by chance is 2.5%, e.g., $P(S \geq u + 2\sigma) = 2.5%$. Thus we are 97.5% confident that the alignment score is significant (not by chance).
• For any score $x$, we can compute $P(S \geq x)$, which is called p-value.
Figure: Histogram of alignment scores
Model-Based Approach
(Karlin and Altschul)

http://www.people.virginia.edu/~wrp/cshl02/Altschul/Altschul-3.html

• Extreme Value Distribution

\[ P(S \geq x) = 1 - \exp(-Kmn e^{-\lambda x}) \]

K and lamda are statistical parameters depending on substitution matrix. For BLOSUM62, lamda=0.252, K=0.35
P-Value

• $P(S \geq x)$ is called p-value. It is the probability that random sequences has alignment score equal to or bigger than $x$.

• Smaller $\rightarrow$ more significant.
Problems of Using Dynamic Programming to Search Large Sequence Database

- Search homologs in DNA and protein database is often the first step of a bioinformatics study.
- DP is too slow for large sequence database search such as Genbank and UniProt. Each DP search can take hours.
- Most DP search time is wasted on unrelated sequences or dissimilar regions.
- Developing fast, practical sequence comparison methods for database search is important.
Fast Sequence Search Methods

• All successful, rapid sequence comparison methods are based on a simple fact: similar sequences /regions share some common words.
• First such method is FASTP (Pearson & Lipman, 1985).
• Most widely used methods are BLAST (Altschul et al., 1990) and PSI-BLAST (Altschul et al., 1997).
1. Compile a list of words for a query
2. Scan sequences in database for word hits
3. Extending hits
Compile Word List

- Words: w-mer with length w.
- Protein 4-mer and DNA 12-mer

Query:

DSRSKGEPRDSDGTLQSQEAKAVKKTSLFE

Words: DSRS, SRSK, RSKG, KGEF....
Example of extension

Query: DSRSKGEPRDSGTLQSQEAKAVKKKTSLFE

Words: DSRS, SRSK, RSKG, KGEP, ...

Database Sequence: PESRSKGEPRDSGGKQMDSOKPD

Maximum Segment Pair: ESRSKGEPRDSG
P-Value and E-Value

- P-value
- E-value = database size * p-value
- Common threshold: 0.01

\[ P\text{-value} = \text{Prob}(\text{score} \geq S) \]
Usage of BLAST

• Versions: BLASTP, BLASTN, BLASTX (translated)
• Sequence Databases: NR, PDB, SwissProt, Gene databases of organisms, or your own databases
• Expectation value
• Low complexity
• Similarity matrix (PAM or BLOSUM)
• Output format
The Basic Local Alignment Search Tool (BLAST) finds regions of local similarity between sequences. The program compares nucleotide or protein sequences to sequence databases and calculates the statistical significance of matches. BLAST can be used to infer functional and evolutionary relationships between sequences as well as help identify members of gene families.

### Nucleotide
- Quickly search for highly similar sequences (megablast)
- Quickly search for divergent sequences (discontiguous megablast)
- Nucleotide-nucleotide BLAST (blastn)
- Search for short, nearly exact matches
- Search trace archives with megablast or discontiguous megablast

### Protein
- Protein-protein BLAST (blastp)
- Position-specific iterated and pattern-hit initiated BLAST (PSI- and PHI-BLAST)
- Search for short, nearly exact matches
- Search the conserved domain database (rpsblast)
- Protein homology by domain architecture (cdart)

### Translated
- Translated query vs. protein database (blastx)
- Protein query vs. translated database (tblastn)
- Translated query vs. translated database (tblastx)

### Genomes
- Human, mouse, rat, chimp, cow, pig, dog, sheep, cat
- Chicken, puffer fish, zebrafish
- Fly, honey bee, other insects
- Microbes, environmental samples
- Plants, nematodes
- Fungi, protozoa, other eukaryotes
DNA Blast
The request ID is 1156545882-10456-164751611258.BLASTQ4

Sequences producing significant alignments:

<table>
<thead>
<tr>
<th>Score</th>
<th>E Value</th>
</tr>
</thead>
</table>
| gi|67876011|ref|ZP_00505069.1| Lipolytic enzyme, G-D-S-L:Clos... 344 1e-93
| gi|1210011|sp|F15329|SUNK CLOTH Putative endoglucanase X (EGX) ... 207 2e-56
| gi|3123333|dbj|PAC90705.1| q117764 [Gloeobacter violaceus PCC 7421 ... 103 5e-21
| gi|35214787|emb|CAJ01036.1| putative xylanase [Actinoplanes sp. 50.9 3e-17
| gi|6123721|ref|XP_386414.1| hypothetical protein EGO2383.1 [ ... 87.4 3e-16
| gi|11165736|gb|EF78610.1| hypothetical protein SNOS_14245 [Pha ... 53.2 7e-15
| gi|50254376|ref|ZP_0213270.1| hypothetical protein Bcep70_02 ... 82.1 1e-14
| gi|52209736|emb|CAL05705.1| putative exported oxidase [Burkho ... 81.3 2e-14
| gi|7657911|gb|ABA45588.1| galactose oxidase-like protein [Bu ... 81.3 3e-14
| gi|11225445|ref|YP_716293.1| putative Glycosyl hydrolase [Fr ... 79.3 9e-14

Matched sequences ranked by score and evalue

> [gi|35213333|dbj|PAC90705.1] 6 gi12764 [Gloeobacter violaceus PCC 7421]
| gi|35213333|ref|NP_925710.1| hypothetical protein gi12764 [Gloeobacter violaceus PCC 7421]
| Length=559

Score = 103 bits (256), Expect = 5e-21, Method: Composition-based stats.
Identities = 89/194 (45%), Positives = 115/194 (59%), Gaps = 12/194 (6%)

Query 7  KIMPGDSCCETGSGGEGSMYSYRTLYRLLLQACGLSIDFVGQARGSFSSPFDPKDEMHEGSGW 66
K+MP+SDS TEG G YRT+L+ L G + DFVGQ SGPSST DK+HEG G+
Sbjct 108 KVHPLGDSITEFTVSY--GGYRIIDWNLSVESEGNADFGVSQSDFGGSFLSUK+HEG+

Query 67  TTFIGASINNWLNTHMEDVVF1wigondillnon--1naroisln1IDCIFTVKNPVLFL 124
I QIA I+WL + F+ V L IG ND+ N + L S LIQIF + + V L+
Sbjct 166 QF1QIADGIDOWLPKYPFETVLLISTNIDKRNNDKFGAPRLSALIDQIFALRSSVLY 225

Query 125  VADYYPWF-EAIKQ-----YNAVPIGIVQKKXAGKKVYFYKLEIQDFRNDISWDGLHL 179
VA P + AI Q YNA IPIV G KKKV +V + D+ D +H
Sbjct 226 VASIPPDADSAIQRVLDYNAAIPIGVNGKIYTGQKVVVVDDYNAL--TTADLA-DTVHP 282

Query 100  SEIGYKIANWYK 193
GY KIA+ W++
Sbjct 283 DAEKYKIAADRWF 296

Significant local alignments
Database Search Using Sequence Profiles

• Multiple related sequences in protein family and super family (profile)
• More data, more robust, more sensitive
• Consider a group of related sequences (profile) is a **POWERFUL** idea
Why does a family of sequences help?

Step 1

Protein Universe
Why does a family of sequences help?

Step 1

Step 2

Protein Universe
Why does a family of sequences help?

Iterative search helps find remote homologs.

Protein Family

Protein Universe

Step 1

Step 2

Step 3
PSI-BLAST Algorithm

- Use BLAST to search database. Use significantly matched sequences to construct a profile / PSSM
- Repeat
  - Use PSSM to search database
  - Use significant matched sequences to construct a PSSM
- Until no new sequence is found or reach the maximum number of iterations.
Use PSI-BLAST Software

- **Command:**

  ```bash
  blastpgp -i seq_file -j iteration -h include_evalue_threshold -e report_evalue_threshold -d database -o output_file
  ```

  - **-i:** input sequence file in FASTA format
  - **-j:** number of iterations
  - **-d:** sequence database
  - **-h:** cut-off e-value of including a sequence into PSSM (profile)
  - **-e:** cut-off e-value of reporting a sequence
  - **-o:** output file