MODULE OVERVIEW

Objective: Provide an introduction to the practice of bioinformatics as well as a practical guide to using common bioinformatics databases and algorithms

1.1. Introduction to Bioinformatics

1.2. Sequence Alignment and Database Searching

1.3. Structural Bioinformatics

1.4. Genome Informatics: High Throughput Sequencing Applications and Analytical Methods
WEEK ONE REVIEW

✔️ Answers to last weeks homework (19/20):
  Answers week 1

✔️ Muddy Point Assessment (14/20):
  Responses
  - NCBI BLAST frustrations
  - Need for FASTA header lines “>example1”
  - More on protein structure viewing and finding
  - “Nice Assignment”.
Check out the “Background Reading” material online:

- Dynamic Programming
- Database Searching

Complete the lecture 1.2 homework questions:

http://tinyurl.com/bioinf525-quiz2
TODAYS MENU

• Alignment basics
  ‣ Why compare biological sequences?

• Homologue detection
  ‣ Orthologs, paralogs, similarity and identity
  ‣ Sequence changes during evolution
  ‣ Alignment view: matches, mismatches and gaps

• Pairwise sequence alignment methods
  ‣ Brute force alignment
  ‣ Dot matrices
  ‣ Dynamic programing
    (global vs local alignment)

• Rapid heuristic approaches
  ‣ BLAST

• Practical database searching
  ‣ PSI-BLAST and HMM approaches
Basic Idea: Display one sequence above another with spaces (termed gaps) inserted in both to reveal similarity of nucleotides or amino acids.

Seq1: C A T T C A C

Seq2: C T C G C A G C
Basic Idea: Display one sequence above another with spaces (termed gaps) inserted in both to reveal similarity of nucleotides or amino acids.

Seq1: C A T T C A C  
     |     |     |
Seq2: C T C G C A G C

Two types of character correspondence

match
mismatch
**Basic Idea:** Display one sequence above another with spaces (termed **gaps**) inserted in both to reveal **similarity** of nucleotides or amino acids.

**Seq1:** C A T – T C A – C

**Seq2:** C – T C G C A G C

Add gaps to increase number of matches
Basic Idea: Display one sequence above another with spaces (termed **gaps**) inserted in both to reveal **similarity** of nucleotides or amino acids.

Seq1: C A T – T C A – C  

Seq2: C – T C G C A G C

Gaps represent ‘indels’
mismatch represent mutations

match
mismatch \} **mutation**
insertion \} **indels**
deletion
Why compare biological sequences?

• To obtain **functional or mechanistic insight** about a sequence by inference from another potentially better characterized sequence
• To find whether two (or more) genes or proteins are **evolutionarily related**
• To find **structurally or functionally similar regions** within sequences (e.g. catalytic sites, binding sites for other molecules, etc.)
• Many practical bioinformatics applications...
Practical applications of sequence alignment include...

- **Similarity searching of databases**
  - Protein structure prediction, annotation, etc...

- **Assembly of sequence reads** into a longer construct such as a genomic sequence

- **Mapping sequencing reads to a known genome**
  - "Resequencing", looking for differences from reference genome - SNPs, indels (insertions or deletions)
  - Mapping transcription factor binding sites via ChIP-Seq (chromatin immuno-precipitation sequencing)
  - Pretty much all next-gen sequencing data analysis
Practical applications of sequence alignment include...

- **Similarity searching of databases**
  - Protein structure prediction
- **Assembly of sequence reads into a longer construct** such as a bacterial genome
- **Mapping sequencing reads to a known genome**
  - "Resequencing", looking for differences from reference SNPs, indels (insertions or deletions)
  - Mapping transcription factor binding sites via ChIP-Seq (chromatin immuno-precipitation sequencing)
  - Pretty much all next-gen sequencing data analysis

N.B. Pairwise sequence alignment is arguably the most fundamental operation of bioinformatics!
Outline for today

• Alignment basics
  ‣ Why compare biological sequences?

• Homologue detection
  ‣ Orthologs, paralogs, similarity and identity
  ‣ Sequence changes during evolution
  ‣ Alignment view: matches, mismatches and gaps

• Pairwise sequence alignment methods
  ‣ Brute force alignment
  ‣ Dot matrices
  ‣ Dynamic programing
    (global vs local alignment)

• Rapid heuristic approaches
  ‣ BLAST

• Practical database searching
  ‣ PSI-BLAST and HMM approaches
Sequence comparison is most informative when it detects **homologs**

**Homologs** are sequences that have common origins *i.e.* they share a **common ancestor**

- They may or may not have common activity

Can be used to establish evolutionary relationships
Key terms

When we talk about related sequences we use specific terminology.

*Homologous sequences* may be either:

– **Orthologs** or **Paralogs**

(Note. these are all or nothing relationships!)

*Any pair of sequences* may share a certain level of:

– **Identity** and/or **Similarity**

(Note. if these metrics are above a certain level we often infer homology)
Orthologs tend to have similar function

Orthologs: are homologs produced by speciation that have diverged due to divergence of the organisms they are associated with.

– Ortho = [greek: straight] ... implies direct descent
Paralogs tend to have slightly different functions

**Paralogs**: are homologs produced by *gene duplication*. They represent genes derived from a common ancestral gene that *duplicated within an organism* and then subsequently *diverged by accumulated mutation*.

— Para = [greek: along side of]
Orthologs vs Paralogs

• In practice, determining ortholog vs paralog can be a complex problem:
  – gene loss after duplication,
  – lack of knowledge of evolutionary history,
  – weak similarity because of evolutionary distance

• Homology does not necessarily imply exact same function
  – may have similar function at very crude level but play a different physiological role
Sequence changes during evolution

There are three major types of sequence change that can occur during evolution.

– Mutations/Substitutions
– Deletions
– Insertions

**Diagram**

- **CTCGTTA**
- **CATGTGA**
- **CACTGTA**
- **(B)**
- **(A)**
- **(C)**
Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

- **Mutations/Substitutions**
- Deletions
- Insertions

Likely occurred prior to speciation
Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

- Mutations/Substitutions
- Deletions
- Insertions

CTCGTTA → CACGT TA

CACGT TA

Mutation

CATGT TA  CACTG TA

(speciation)
Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

– Mutations/Substitutions
  - CTCGTTA → CACGTTA
– Deletions
  - CACGTTA → CACTTA
– Insertions
Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

– Mutations/Substitutions
  CTCGTTA $\rightarrow$ CACGTTA

– Deletions
  CACGTTA $\rightarrow$ CACTTA
  CACTTA $\rightarrow$ CACTGTA

– Insertions
  CACTTA $\rightarrow$ CACTGTA
Mutations, deletions and insertions

There are three major types of sequence change that can occur during evolution.

- **Mutations/Substitutions**
  - CTCGTTA → CACGTTA
  - CACGTTA → CATGTTA

- Deletions

- Insertions
Alignments are great tools to visualize sequence similarity and evolutionary changes in homologous sequences.

- **Mismatches** represent mutations/substitutions
- **Gaps** represent insertions and deletions (indels)

**Alignment view**
Alternative alignments

• Unfortunately, finding the correct alignment is difficult if we do not know the evolutionary history of the two sequences
  – There are many possible alignments
  – Which alignment is best?
Alternative alignments

- One way to judge alignments is to compare their number of matches, insertions, deletions and mutations.

<table>
<thead>
<tr>
<th></th>
<th>4 matches</th>
<th>3 mismatches</th>
<th>0 gaps</th>
<th>6 matches</th>
<th>0 mismatches</th>
<th>2 gaps</th>
<th>5 matches</th>
<th>1 mismatches</th>
<th>2 gaps</th>
</tr>
</thead>
<tbody>
<tr>
<td>CATGTTA</td>
<td>CACTGTA</td>
<td>CACTGT-A</td>
<td>CAC-TGTA</td>
<td>CACTTGTA</td>
<td>CATGTTA</td>
<td>CATGT-TA</td>
<td>CATGTTA</td>
<td>CATGT-TA</td>
<td></td>
</tr>
</tbody>
</table>
Scoring alignments

- We can assign a score for each match (+3), mismatch (+1) and indel (-1) to identify the optimal alignment for this scoring scheme.

<table>
<thead>
<tr>
<th>Alignment</th>
<th>Matches</th>
<th>Mismatches</th>
<th>Indels</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>CATGTTA</td>
<td>4</td>
<td>3</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>CATGTTA</td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>16</td>
</tr>
<tr>
<td>CATGTTA</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>14</td>
</tr>
</tbody>
</table>
Optimal alignments

- Biologists often prefer **parsimonious alignments**, where the number of postulated sequence changes is minimized.
Optimal alignments

- Biologists often prefer **parsimonious alignments**, where the number of sequence changes is minimized.

**Warning:** There may be more than one optimal alignment and these may not reflect the true evolutionary history of our sequences!
Side note: sequence identity and similarity

• Two commonly quoted metrics for pairs of aligned sequences.
  – **Sequence identity**: typically quotes the percent of identical characters in the aligned region of two sequences
  – **Sequence similarity**: typically the score resulting from optimal pair-wise alignment (note dependence on parameters used: *i.e.* scoring scheme)

• N.B. In contrast, **homology is an all or nothing relationship**, you can not have a percent homology!
Side note: sequence identity and similarity

• High sequence similarity is frequently used as an indicator of homology
  – Use to find genes and/or proteins with potentially similar or identical function
  – Can query a database of sequences by performing a series of pair-wise alignments

• Knowledge of the difference between sequences can also yield valuable functional and mechanistic insights
  – A gene from a normal and an affected subject – possible cause of a heritable disease
  – Similar proteins with different substrate specificities – what amino acid changes might be responsible for this?
Outline for today

- **Alignment basics**
  - Why compare biological sequences?

- **Homologue detection**
  - Orthologs, paralogs, similarity and identity
  - Sequence changes during evolution
  - Alignment view: matches, mismatches and gaps

- **Pairwise sequence alignment methods**
  - Brute force alignment
  - Dot matrices
  - Dynamic programming
    (global vs local alignment)

- **Rapid heuristic approaches**
  - BLAST

- **Practical database searching**
  - PSI-BLAST and HMM approaches
Outline for today

• Alignment basics
  ‣ Why compare biological sequences?

• Homologue detection
  ‣ Orthologs, paralogs, similarity and identity
  ‣ Sequence changes during evolution
  ‣ Alignment view: matches, mismatches and gaps

• Pairwise sequence alignment methods
  (global vs local alignment)

• Rapid heuristic approaches
  ‣ BLAST

• Practical database searching
  ‣ PSI-BLAST and HMM approaches

How do we compute the optimal alignment between two sequences?

Quiz questions:
http://tinyurl.com/bioinf525-quiz2
**Pair-wise Sequence Alignment**

- **Objective**: arrange two sequences in such a fashion that pairs of matching characters between the two sequences are maximized
  - Match does not have to be identity, can be defined by a function that ranks or scores the characters being compared (often termed a **substitution matrix**)
  - Ungapped alignment example – bars indicate matching characters

<table>
<thead>
<tr>
<th>Seq1</th>
<th>GTAATCTG-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq2</td>
<td>-TAAGCTGA</td>
</tr>
</tbody>
</table>
Simplest case – brute force alignments

• In the simplest case we can simply slide one sequence across the other and count matching characters for each possible alignment
  – Chose a scoring scheme and do not allow internal gaps within sequences
  – Algorithmic complexity is linear
    \[ N + M \text{ alignments to consider} \]
    (where \( N \) and \( M \) are the length of each sequence)
Brute Force Alignment, No Gaps

Etc...

Slide from Jeffery de Wet
Gaps make the brute force method unusable for all but the shortest sequences

- Pairs of related sequences often have insertions or deletions relative to one-another, we therefore require **gapped pair-wise alignment**
  - Need to generate all the possible gap lengths and combinations of gaps at all possible positions in both sequences
  - For two sequences of equal length, the formula is:

\[
\binom{2N}{N} = \frac{(2N)!}{(N!)^2} \approx \frac{2^{2N}}{\sqrt{\pi N}}
\]

- N = 10: 184756
- N = 50: \(\sim 1.00E29\)
- N = 250: \(\sim 1.17E149\)

Slide from Jeffery de Wet
Three general solutions to the alignment problem

- The **dot plot** or **dot matrix** approach
  - A simple graphical method for pair-wise alignment
  - No scoring, so difficult to compare alternative alignments
  - Can give visual clues to sequence structure but requires human interaction

- **Dynamic programming** algorithms
  - Provides Optimal solutions (but not necessarily unique solutions)

- **Heuristic word** or **k-tuple** approaches
  - Much faster (e.g. **BLAST** and **FASTA**)
  - Widely used for database searches
  - May miss some pairs with low similarity
Three general solutions to the alignment problem

• **The dot plot or dot matrix approach**
  – A simple graphical method for pair-wise alignment
  – No scoring, so difficult to compare alternative alignments
  – Can give visual clues to sequence structure but requires human interaction

• **Dynamic programming algorithms**
  – Provides Optimal solutions (but not necessarily unique solutions)

• **Heuristic word or k-tuple approaches**
  – Much faster (e.g. BLAST and FASTA)
  – Widely used for database searches
  – May miss some pairs with low similarity
Dot plots: simple graphical approach

• Place one sequence on the vertical axis of a 2D grid (or matrix) and the other on the horizontal
Dot plots: simple graphical approach

• Now simply put dots where the horizontal and vertical sequence values match
Dot plots: simple graphical approach

- Diagonal runs of dots indicate matched segments of sequence
Dot plots: simple graphical approach

**Q.** What would the dot matrix of a two identical sequences look like?
Dot plots: simple graphical approach

• Dot matrices for long sequences can be noisy
Dot plots: window size and match stringency

**Solution:** use a window and a threshold
– compare character by character within a window
– require certain fraction of matches within window in order to display it with a dot.

- You have to choose window size and stringency

![Diagram showing dot plot with window size and stringency example]
Dot plots: window size and match stringency

**Solution:** use a *window* and a *threshold*
– compare character by character within a window
– require certain fraction of matches within window in order to display it with a dot.
  - You have to choose window size and stringency

```
        A   C   G   C   G
|____|____|____|____|____|
  A   | |   | |   | |   |
  C   | |   | |   | |   |
  A   | |   | |   | |   |
  C   | |   | |   | |   |
  G   | |   | |   | |   |
        A   C   G   C   G
```

Filter
Window = 3
Stringency = 2
Window size = 5 bases

A dot plot simply puts a dot where two sequences match. In this example, dots are placed in the plot if 5 bases in a row match perfectly. Requiring a 5 base perfect match is a heuristic – only look at regions that have a certain degree of identity.

Do you expect evolutionarily related sequences to have more word matches (matches in a row over a certain length) than random or unrelated sequences?
This is a dot plot of the same sequence pair. Now 7 bases in a row must match for a dot to be placed. Noise is reduced.

Using windows of a certain length is very similar to using words (kmers) of N characters in the heuristic alignment search tools.

Bigger window (kmer) fewer matches to consider.

Web site used: [http://www.vivo.colostate.edu/molkit/dnadot/](http://www.vivo.colostate.edu/molkit/dnadot/)
Only **diagonals** can be followed.

Downward or rightward paths represent **insertion** or **deletions** (gaps in one sequence or the other).
Global alignments go from end to end, i.e. from the upper left corner to the lower right corner.

Global alignments do not have good statistical characterization and are not used for database searches.
Uses for dot matrices

• Visually assessing the similarity of two protein or two nucleic acid sequences

• Finding local repeat sequences within a larger sequence by comparing a sequence to itself
  – Repeats appear as a set of diagonal runs stacked vertically and/or horizontally
Repeats

Human LDL receptor protein sequence (Genbank P01130)

W = 1
S = 1

(Figure from Mount, “Bioinformatics sequence and genome analysis”)}
Repeats

Human LDL receptor protein sequence (Genbank P01130)

W = 23
S = 7

(Figure from Mount, “Bioinformatics sequence and genome analysis”)
Side note: dots can have “weights”

- Some matches can be rewarded more than others, depending on likelihood
- Use PAM or BLOSUM substitution matrix
  - (more on these later)
- Put a dot only if a minimum total or average weight is achieved
  - See chapter 3 in Mount, “Bioinformatics sequence and genome analysis”.
Three general solutions to the alignment problem

• The dot plot or dot matrix approach
  – A simple graphical method for pair-wise alignment
  – No scoring, so difficult to compare alternative alignments
  – Can give visual clues to sequence structure but requires human interaction

• Dynamic programming algorithms
  – Provides Optimal solutions (but not necessarily unique solutions)

• Heuristic word or k-tuple approaches
  – Much faster (e.g. BLAST and FASTA)
  – Widely used for database searches
  – May miss some pairs with low similarity
The Dynamic Programming Algorithm

- The dynamic programming algorithm can be thought of an extension to the dot plot approach
  - One sequence is placed down the side of a grid and another across the top
  - Instead of placing a dot in the grid, we **compute a score** for each position
  - Finding the optimal alignment corresponds to finding the path through the grid with the **highest possible score**
Different paths represent different alignments

Matches are represented by diagonal paths and indels with horizontal or vertical path segments
Algorithm of Needleman and Wunsch

• The Needleman–Wunsch approach to global sequence alignment has three basic steps:
  (1) setting up a 2D-grid (or alignment matrix),
  (2) scoring the matrix, and
  (3) identifying the optimal path through the matrix

Scoring the alignment matrix

• Start by filling in the first row and column – these are all indels (gaps).
  – Each step you take you will add the gap penalty to the score \( S_{i,j} \) accumulated in the previous cell

<table>
<thead>
<tr>
<th></th>
<th>D</th>
<th>P</th>
<th>L</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
</tr>
<tr>
<td>D</td>
<td>-2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>-4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>-6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>-8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Scores:** match = +1, mismatch = -1, gap = -2
Scoring the alignment matrix

- Start by filling in the first row and column – these are all indels (gaps).
  
  Each step you take you will add the **gap penalty** to the score ($S_{i,j}$) accumulated in the previous cell.

\[
\begin{array}{cccc}
& - & D & P & L & E \\
- & 0 & -2 & -4 & -6 & -8 \\
D & -2 & & & & \\
P & -4 & & & & \\
M & -6 & & & & \\
E & -8 & & & & \\
\end{array}
\]

**Scores:** match = +1, mismatch = -1, gap = -2

\[
S_{i+4} = (-2) + (-2) + (-2) + (-2)
\]

Seq1: DPME
Seq2: ----

62
Scoring the alignment matrix

• Then go to the empty corner cell (upper left). It has filled in values in up, left and diagonal directions
  – Now can ask which of the three directions gives the highest score?
  – keep track of this score and direction

Scores: match = +1, mismatch = -1, gap = -2
Scoring the alignment matrix

• Then go to the empty corner cell (upper left). It has filled in values in up, left and diagonal directions
  – Now can ask which of the three directions gives the highest score?
  – keep track of this score and direction

Scores: match = +1, mismatch = -1, gap = -2

\[
S(i, j) = \text{Max} \left\{ \begin{array}{l}
S(i-1, j-1) + \text{(mis)match} \\
S(i-1, j) - \text{gap penalty} \\
S(i, j-1) - \text{gap penalty}
\end{array} \right. 
\]
Scoring the alignment matrix

- Then go to the empty corner cell (upper left). It has filled in values in up, left and diagonal directions.
  - Now can ask which direction gives the highest score.
  - Keep track of direction and score.

<table>
<thead>
<tr>
<th></th>
<th>D</th>
<th>P</th>
<th>L</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
</tr>
<tr>
<td>D</td>
<td>-2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>-4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>-6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>-8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Scores:** match = +1, mismatch = -1, gap = -2

1. (0) + (+1) = +1 <= (D-D) match!
2. (-2) + (-2) = -4
3. (-2) + (-2) = -4
Scoring the alignment matrix

- At each step, the score in the current cell is determined by the scores in the neighboring cells
  - The maximal score and the direction that gave that score is stored (we will use these later to determine the optimal alignment)

Scores: match = +1, mismatch = -1, gap = -2

1. \((-2) + (-1) = -3\) \(\leq (D-P)\) mismatch!
2. \((-4) + (-2) = -6\)
3. \((1) + (-2) = -1\)

Alignment: \(D-P\)
Scoring the alignment matrix

- We will continue to store the alignment score \( S_{i,j} \) for all possible alignments in the alignment matrix.

**Scores:** match = +1, mismatch = -1, gap = -2

1. \((-4)+(-1) = -5\)  \(\leq (D-L)\) mismatch
2. \((-6)+(-2) = -8\)
3. \((-1)+(-2) = -3\)

Alignment: \(D--\) \(DPL\)
Scoring the alignment matrix

- For the highlighted cell, the corresponding score ($S_{i,j}$) refers to the score of the optimal alignment of the first $i$ characters from sequence1, and the first $j$ characters from sequence2.

Scores: match = +1, mismatch = -1, indel = -2

1. $(-1)+(-1) = -2$
2. $(-3)+(-2) = -5$
3. $(2)+(-2) = 0$

Alignment DP- DPL
Scoring the alignment matrix

- At each step, the score in the current cell is determined by the scores in the neighboring cells.
  - The maximal score and the direction that gave that score is stored.

Scores: match = +1, mismatch = -1, indel = -2

1. (2)+(-1) = 0  <= mismatch
2. (0)+(-2) = -2
3. (0)+(-2) = -2
## Scoring the alignment matrix

- The score of the best alignment of the entire sequences corresponds to $S_{n,m}$
  - (where $n$ and $m$ are the length of the sequences)

### Scores: match = +1, mismatch = -1, indel = -2

<table>
<thead>
<tr>
<th></th>
<th>-</th>
<th>D</th>
<th>P</th>
<th>L</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
<td>-8</td>
</tr>
<tr>
<td>D</td>
<td>-2</td>
<td>1</td>
<td>-1</td>
<td>-3</td>
<td>-5</td>
</tr>
<tr>
<td>P</td>
<td>-4</td>
<td>-1</td>
<td>2</td>
<td>0</td>
<td>-2</td>
</tr>
<tr>
<td>M</td>
<td>-6</td>
<td>-3</td>
<td>0</td>
<td>1</td>
<td>-1</td>
</tr>
</tbody>
</table>

- **Alignment:**
  1. $(+1)+(+1) = +2$
  2. $(-1)+(-2) = -3$
  3. $(-1)+(-2) = -3$
Scoring the alignment matrix

• To find the best alignment, we retrace the arrows starting from the bottom right cell
  – N.B. The optimal alignment score and alignment are dependent on the chosen scoring system

<table>
<thead>
<tr>
<th></th>
<th>-</th>
<th>D</th>
<th>P</th>
<th>L</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
<td>-8</td>
</tr>
<tr>
<td>D</td>
<td>-2</td>
<td>1</td>
<td>-1</td>
<td>-3</td>
<td>-5</td>
</tr>
<tr>
<td>P</td>
<td>-4</td>
<td>-1</td>
<td>2</td>
<td>0</td>
<td>-2</td>
</tr>
<tr>
<td>M</td>
<td>-6</td>
<td>-3</td>
<td>0</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>E</td>
<td>-8</td>
<td>-5</td>
<td>-2</td>
<td>-1</td>
<td>2</td>
</tr>
</tbody>
</table>

**Scores**: match = +1, mismatch = -1, indel = -2

Alignment

**Alignment**

DPME

DPLE
Questions:

- What is the optimal score for the alignment of these sequences and how do we find the optimal alignment?

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>T</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
<td>-8</td>
<td>-10</td>
</tr>
<tr>
<td>C</td>
<td>-2</td>
<td>1</td>
<td>-1</td>
<td>-3</td>
<td>-5</td>
<td>-7</td>
</tr>
<tr>
<td>A</td>
<td>-4</td>
<td>-1</td>
<td>2</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
</tr>
<tr>
<td>C</td>
<td>-6</td>
<td>-3</td>
<td>0</td>
<td>1</td>
<td>-1</td>
<td>-3</td>
</tr>
<tr>
<td>T</td>
<td>-8</td>
<td>-5</td>
<td>-2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>-10</td>
<td>-7</td>
<td>-4</td>
<td>-1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>T</td>
<td>-12</td>
<td>-9</td>
<td>-6</td>
<td>-3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>A</td>
<td>-14</td>
<td>-11</td>
<td>-8</td>
<td>-5</td>
<td>-2</td>
<td>1</td>
</tr>
</tbody>
</table>

The optimal score is found at the bottom right corner of the table, which is 2.
Questions:

- What is the optimal score for the alignment of these sequences and how do we find the optimal alignment?

<table>
<thead>
<tr>
<th></th>
<th>-</th>
<th>C</th>
<th>A</th>
<th>T</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
<td>-8</td>
<td>-10</td>
<td>-12</td>
<td>-14</td>
</tr>
<tr>
<td>C</td>
<td>-2</td>
<td>1</td>
<td>-1</td>
<td>-3</td>
<td>-5</td>
<td>-7</td>
<td>-9</td>
<td>-11</td>
</tr>
<tr>
<td>A</td>
<td>-4</td>
<td>-1</td>
<td>2</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
<td>-8</td>
</tr>
<tr>
<td>C</td>
<td>-6</td>
<td>-3</td>
<td>0</td>
<td>1</td>
<td>-1</td>
<td>-3</td>
<td>-5</td>
<td>-7</td>
</tr>
<tr>
<td>T</td>
<td>-8</td>
<td>-5</td>
<td>-2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
</tr>
<tr>
<td>G</td>
<td>-10</td>
<td>-7</td>
<td>-4</td>
<td>-1</td>
<td>2</td>
<td>0</td>
<td>-1</td>
<td>-3</td>
</tr>
<tr>
<td>T</td>
<td>-12</td>
<td>-9</td>
<td>-6</td>
<td>-3</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>A</td>
<td>-14</td>
<td>-11</td>
<td>-8</td>
<td>-5</td>
<td>-2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
Questions:

- To find the best alignment we retrace the arrows starting from the bottom right cell.

<table>
<thead>
<tr>
<th></th>
<th>-</th>
<th>C</th>
<th>A</th>
<th>T</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
<td>-8</td>
<td>-10</td>
<td>-12</td>
<td>-14</td>
</tr>
<tr>
<td>C</td>
<td>-2</td>
<td>1</td>
<td>-1</td>
<td>-3</td>
<td>-5</td>
<td>-7</td>
<td>-9</td>
<td>-11</td>
</tr>
<tr>
<td>A</td>
<td>-4</td>
<td>-1</td>
<td>2</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
<td>-6</td>
<td>-8</td>
</tr>
<tr>
<td>C</td>
<td>-6</td>
<td>-3</td>
<td>0</td>
<td>1</td>
<td>-1</td>
<td>-3</td>
<td>-5</td>
<td>-7</td>
</tr>
<tr>
<td>T</td>
<td>-8</td>
<td>-5</td>
<td>-2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-2</td>
<td>-4</td>
</tr>
<tr>
<td>G</td>
<td>-10</td>
<td>-7</td>
<td>-4</td>
<td>-1</td>
<td>2</td>
<td>0</td>
<td>-1</td>
<td>-3</td>
</tr>
<tr>
<td>T</td>
<td>-12</td>
<td>-9</td>
<td>-6</td>
<td>-3</td>
<td>0</td>
<td>3</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>A</td>
<td>-14</td>
<td>-11</td>
<td>-8</td>
<td>-5</td>
<td>-2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
More than one alignment possible

- Sometimes more than one alignment can result in the same optimal score

Alignment

CACTGT−A
CA−TGTTA

CACTG−TA
CA−TGTTA
The alignment and score are dependent on the scoring system

- Here we increase the gap penalty from -2 to -3

<table>
<thead>
<tr>
<th></th>
<th>-</th>
<th>C</th>
<th>A</th>
<th>T</th>
<th>G</th>
<th>T</th>
<th>T</th>
<th>T</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>-3</td>
<td>-6</td>
<td>-9</td>
<td>-12</td>
<td>-15</td>
<td>-18</td>
<td>-21</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>-3</td>
<td>1</td>
<td>-2</td>
<td>-5</td>
<td>-8</td>
<td>-11</td>
<td>-14</td>
<td>-17</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>-6</td>
<td>-2</td>
<td>2</td>
<td>-1</td>
<td>-4</td>
<td>-7</td>
<td>-10</td>
<td>-13</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>-9</td>
<td>-5</td>
<td>-1</td>
<td>1</td>
<td>-2</td>
<td>-5</td>
<td>-8</td>
<td>-11</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>-12</td>
<td>-8</td>
<td>-4</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>-4</td>
<td>-7</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>-15</td>
<td>-11</td>
<td>-7</td>
<td>-3</td>
<td>1</td>
<td>-1</td>
<td>-2</td>
<td>-5</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>-18</td>
<td>-14</td>
<td>-10</td>
<td>-6</td>
<td>-2</td>
<td>2</td>
<td>0</td>
<td>-3</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>-21</td>
<td>-17</td>
<td>-13</td>
<td>-9</td>
<td>-5</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Alignment

CACTGT-A
CA-TGTTA

CACTG-TA
CA-TGTTA

CACTGTA
CATGTTA
Global vs local alignments

• Needleman-Wunsch is a **global alignment** algorithm
  – Resulting alignment spans the complete sequences end to end
  – This is appropriate for closely related sequences that are similar in length

• For many practical applications we require **local alignments**
  – Local alignments highlight sub-regions (*e.g.* protein domains) in the two sequences that align well
Local alignment: Definition

- Smith & Waterman proposed simply that a local alignment of two sequences allow arbitrary-length segments of each sequence to be aligned, with no penalty for the unaligned portions of the sequences. Otherwise, the score for a local alignment is calculated the same way as that for a global alignment.

The Smith-Waterman algorithm

• Three main modifications to Needleman-Wunsch:
  – Allow a node to start at 0
  – The score for a particular cell cannot be negative
    • if all other score options produce a negative value, then a zero must be inserted in the cell
  – Record the highest-scoring node, and trace back from there

\[ S(i, j) = \begin{cases} 
  S(i-1, j-1) + \text{(mis)match} \\
  S(i-1, j) - \text{gap penalty} \\
  S(i, j-1) - \text{gap penalty} \\
  0 
\end{cases} \]
### Sequence 1

<table>
<thead>
<tr>
<th></th>
<th>C</th>
<th>A</th>
<th>G</th>
<th>C</th>
<th>C</th>
<th>U</th>
<th>C</th>
<th>G</th>
<th>C</th>
<th>U</th>
<th>U</th>
<th>A</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>-</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td><strong>A</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
</tr>
<tr>
<td><strong>A</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>0.7</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>0.7</td>
</tr>
<tr>
<td><strong>U</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.7</td>
<td>0.3</td>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
</tr>
<tr>
<td><strong>G</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.3</td>
<td>0.0</td>
<td>0.7</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.7</td>
<td>0.7</td>
<td>1.0</td>
</tr>
<tr>
<td><strong>C</strong></td>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>2.0</td>
<td>1.3</td>
<td>0.3</td>
<td>1.0</td>
<td>0.3</td>
<td>2.0</td>
<td>0.7</td>
<td>0.3</td>
</tr>
<tr>
<td><strong>C</strong></td>
<td>0.0</td>
<td>1.0</td>
<td>0.7</td>
<td>0.0</td>
<td>1.0</td>
<td>3.0</td>
<td>1.7</td>
<td>1.3</td>
<td>1.0</td>
<td>1.3</td>
<td>1.7</td>
<td>0.3</td>
<td>0.0</td>
</tr>
<tr>
<td><strong>A</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>2.0</td>
<td>0.7</td>
<td>0.3</td>
<td>1.7</td>
<td>2.7</td>
<td>1.3</td>
<td>1.0</td>
<td>0.7</td>
<td>1.0</td>
<td>1.3</td>
<td>1.3</td>
</tr>
<tr>
<td><strong>A</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>0.7</td>
<td>0.3</td>
<td>1.3</td>
<td>1.0</td>
<td>2.3</td>
<td>1.3</td>
<td>2.3</td>
<td>0.7</td>
<td>1.7</td>
<td>2.7</td>
</tr>
<tr>
<td><strong>U</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>0.3</td>
<td>0.3</td>
<td>1.3</td>
<td>1.0</td>
<td>2.3</td>
<td>2.0</td>
<td>0.7</td>
<td>1.7</td>
<td>2.7</td>
<td>1.7</td>
<td>1.0</td>
</tr>
<tr>
<td><strong>U</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.3</td>
<td>0.0</td>
<td>1.0</td>
<td>1.0</td>
<td>2.0</td>
<td>1.3</td>
<td>2.3</td>
<td>2.0</td>
<td>1.7</td>
<td>1.3</td>
</tr>
<tr>
<td><strong>G</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
<td>1.0</td>
<td>0.3</td>
<td>0.7</td>
<td>2.0</td>
<td>3.0</td>
<td>1.7</td>
<td>1.3</td>
<td>2.3</td>
<td>2.0</td>
</tr>
<tr>
<td><strong>A</strong></td>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.7</td>
<td>1.0</td>
<td>2.0</td>
<td>0.7</td>
<td>1.7</td>
<td>1.7</td>
<td>3.0</td>
<td>2.7</td>
<td>1.3</td>
<td>1.0</td>
</tr>
<tr>
<td><strong>C</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>0.7</td>
<td>1.0</td>
<td>0.3</td>
<td>0.7</td>
<td>1.7</td>
<td>0.3</td>
<td>2.7</td>
<td>1.7</td>
<td>2.7</td>
<td>2.3</td>
<td>1.0</td>
</tr>
<tr>
<td><strong>G</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>0.7</td>
<td>1.0</td>
<td>0.3</td>
<td>0.3</td>
<td>2.7</td>
<td>1.7</td>
<td>1.3</td>
<td>1.3</td>
<td>2.3</td>
<td>1.3</td>
<td>2.0</td>
</tr>
<tr>
<td><strong>G</strong></td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.7</td>
<td>0.3</td>
<td>0.3</td>
<td>1.3</td>
<td>1.3</td>
<td>2.3</td>
<td>1.3</td>
<td>2.3</td>
<td>1.3</td>
<td>1.0</td>
</tr>
</tbody>
</table>

### Sequence 2

#### Local alignment

- **GCC–AUG**
- **GCCUCGC**
Local alignments can be used for database searching

- **Goal**: Given a query sequence (Q) and a sequence database (D), find a list of sequences from D that are most similar to Q
  - **Input**: Q, D and scoring scheme
  - **Output**: Ranked list of hits

Input

<table>
<thead>
<tr>
<th>Query sequence</th>
<th>Database</th>
</tr>
</thead>
<tbody>
<tr>
<td>GTATGGTCA</td>
<td></td>
</tr>
</tbody>
</table>

Output

<table>
<thead>
<tr>
<th>Score</th>
<th>Ranked hit list</th>
<th>Annotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>GTATGGTCA</td>
<td>Ras</td>
</tr>
<tr>
<td>90</td>
<td>TGATGGTCA</td>
<td>Ras</td>
</tr>
<tr>
<td>40</td>
<td>CGATCTGCA</td>
<td>HSP90</td>
</tr>
<tr>
<td>38</td>
<td>TCGTTGCTA</td>
<td>P450</td>
</tr>
</tbody>
</table>
The database search problem

• Due to the rapid growth of sequence databases, search algorithms have to be both efficient and sensitive
  – Time to search with SW is proportional to $m \times n$ ($m$ is length of query, $n$ is length of database), too slow for large databases!

To reduce search time heuristic algorithms, such as BLAST, first remove database sequences without a strong local similarity to the query sequence in a quick initial scan.
The database search problem

- Due to the rapid growth of sequence databases, search algorithms have to be both efficient and sensitive.
  - Time to search with SW is proportional to $m \times n$ ($m$ is length of query, $n$ is length of database), too slow for large databases!

To reduce search time heuristic algorithms, such as BLAST, first remove database sequences without a strong local similarity to the query sequence in a quick initial scan.
Outline for today

• Alignment basics
  ‣ Why compare biological sequences?

• Homologue detection
  ‣ Orthologs, paralogs, similarity and identity
  ‣ Sequence changes during evolution
  ‣ Alignment view: matches, mismatches and gaps

• Pairwise sequence alignment methods
  ‣ Brute force alignment
  ‣ Dot matrices
  ‣ Dynamic programing
    (global vs local alignment)

• Rapid heuristic approaches
  ‣ BLAST

• Practical database searching
  ‣ PSI-BLAST and HMM approaches
Rapid, heuristic versions of Smith–Waterman: **BLAST**

- **BLAST** (Basic Local Alignment Search Tool) is a simplified form of Smith-Waterman (SW) alignment that is popular because it is **fast** and **easily accessible**
  - BLAST is a heuristic approximation to SW - It examines only part of the search space
  - BLAST saves time by restricting the search by scanning database sequences for likely matches before performing more rigorous alignments
  - Sacrifices some sensitivity in exchange for speed
  - In contrast to SW, BLAST is not guaranteed to find optimal alignments
Rapid, heuristic versions of Smith–Waterman: **BLAST**

- **BLAST** (Basic Local Alignment Search Tool) is a simplified form of Smith-Waterman (SW) alignment, popular because it is **fast** and **easily accessible**.
  - BLAST finds regions of similarity between biological sequences.
  - BLAST saves time by restricting the search by scanning database sequences for likely matches before performing more rigorous alignments.
  - Sacrifices some sensitivity in exchange for speed.
  - In contrast to SW, BLAST is not guaranteed to find optimal alignments.

"The central idea of the BLAST algorithm is to confine attention to sequence pairs that contain an initial word pair match."

Altschul et al. (1990)
• BLAST uses this pre-screening heuristic approximation resulting in an approach that is about 50 times faster than the Smith-Waterman algorithm.
How BLAST works

• Four basic phases
  – **Phase 1**: compile a list of query word pairs \((w=3)\)

```
RGGVKRI  Query sequence
RGG
GGV
GVK
VKR
KRI
```

generate list of \(w=3\) words for query
Blast

– **Phase 2**: expand word pairs to include those similar to query (defined as those above a similarity threshold to original word, i.e. match scores in substitution matrix)

**RGGVKRI**  Query sequence

**RGG RAG RIG RLG ...**

**GGV GAV GTV GCV ...**

**GVK GAK GIK GGK ...**

**VKR VRR VHR VER ...**

**KRI KKI KHI KDI ...**
Blast

– **Phase 3**: a database is scanned to find sequence entries that match the compiled word list

GNYGLKVISLDEVE Database sequence

RGGVKRI Query sequence

search for perfect matches in the database sequence

GGV GAV GTV GCV ...

GVK GLK GIK GGK ...

VKR VRR VHR VER ...

KRI KKI KHI KDI ...
Blast

– **Phase 4**: the initial database hits are extended in both directions using dynamic programming

---

**Query sequence**

**Database sequence**

matched word is used as a local alignment seed
<table>
<thead>
<tr>
<th>G</th>
<th>K</th>
<th>G</th>
<th>N</th>
<th>Y</th>
<th><strong>GLK</strong></th>
<th>V</th>
<th>I</th>
<th>S</th>
<th>L</th>
<th>D</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</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>A</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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</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>G</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>G</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>V</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>K</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>G</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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</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>S</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>G</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>L</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>

**Alignment seed**

**Query sequence**: GRYGLKVISL

**Database sequence**: GNYGLKVISLDV
Dynamic programming

Search for high scoring gapped alignment

Alignment seed

Query sequence

Database sequence

GNYGLKVISLDV

GRG

GVKRISGL
BLAST returns the highest scoring database hits in a ranked list.

Alignment seed

Query sequence: GRGGVKGVRISGL

Database sequence: GNYGLKVIS-L
### BLAST output

- BLAST returns the highest scoring database hits in a ranked list along with details about the target sequence and alignment statistics

<table>
<thead>
<tr>
<th>Description</th>
<th>Max score</th>
<th>Total score</th>
<th>Query cover</th>
<th>E value</th>
<th>Max ident</th>
<th>Accession</th>
</tr>
</thead>
<tbody>
<tr>
<td>kinesin-1 heavy chain [Homo sapiens]</td>
<td>677</td>
<td>677</td>
<td>100%</td>
<td>0</td>
<td>100%</td>
<td>NP_004512.1</td>
</tr>
<tr>
<td>Kif5b protein [Mus musculus]</td>
<td>676</td>
<td>676</td>
<td>100%</td>
<td>0</td>
<td>98%</td>
<td>AAA20133.1</td>
</tr>
<tr>
<td>Kinesin-14 heavy chain [Danio rerio]</td>
<td>595</td>
<td>595</td>
<td>88%</td>
<td>0</td>
<td>78%</td>
<td>XP_00320703</td>
</tr>
<tr>
<td>hypothetical protein EGK_18589</td>
<td>48.2</td>
<td>52</td>
<td>40%</td>
<td>0.03</td>
<td>32%</td>
<td>ELK35081.1</td>
</tr>
<tr>
<td>mKIAA4102 protein [Mus musculus]</td>
<td>42.7</td>
<td>42.7</td>
<td>38%</td>
<td>3.02</td>
<td>24%</td>
<td>EHH28205.1</td>
</tr>
</tbody>
</table>
Statistical significance of results

- An important feature of BLAST is the computation of statistical significance for each hit. This is described by the **E value** (expect value)

<table>
<thead>
<tr>
<th>Description</th>
<th>Max score</th>
<th>Total score</th>
<th>Query cover</th>
<th>E value</th>
<th>Max ident</th>
<th>Accession</th>
</tr>
</thead>
<tbody>
<tr>
<td>kinesin-1 heavy chain [Homo sapiens]</td>
<td>677</td>
<td>677</td>
<td>100%</td>
<td>0</td>
<td>100%</td>
<td>NP_004512.1</td>
</tr>
<tr>
<td>Kif5b protein [Mus musculus]</td>
<td>676</td>
<td>676</td>
<td>100%</td>
<td>0</td>
<td>98%</td>
<td>AAA20133.1</td>
</tr>
<tr>
<td>Kinesin-14 heavy chain [Danio rerio]</td>
<td>595</td>
<td>595</td>
<td>88%</td>
<td>0</td>
<td>78%</td>
<td>XP_00320703</td>
</tr>
<tr>
<td>hypothetical protein EGK_18589</td>
<td>48.2</td>
<td>52</td>
<td>40%</td>
<td>0.03</td>
<td>32%</td>
<td>ELK35081.1</td>
</tr>
<tr>
<td>mKIAA4102 protein [Mus musculus]</td>
<td>42.7</td>
<td>42.7</td>
<td>38%</td>
<td>3.02</td>
<td>24%</td>
<td>EHH28205.1</td>
</tr>
</tbody>
</table>
The **E value** is the **expected** number of hits that are as good or better than the observed local alignment score (with this score or better) if the query and database are **random** with respect to each other

- *i.e.* the number of alignments expected to occur by chance with equivalent or better scores

Typically, only hits with E value **below** a significance threshold are reported

- This is equivalent to selecting alignments with score above a certain score threshold
• Ideally, a threshold separates all query related sequences (yellow) from all unrelated sequences (gray)
• Unfortunately, often both score distributions overlap
  – The E value describes the expected number of hits with a score above the threshold if the query and database are unrelated
• Unfortunately, often both score distributions overlap
  – The E value describes the expected number of hits with a score above the threshold if the query and database are unrelated

Alignment scores of unrelated sequences

The E-value provides an estimate of the number of false positive hits!
<table>
<thead>
<tr>
<th>Description</th>
<th>Max score</th>
<th>Total score</th>
<th>Query cover</th>
<th>E value</th>
<th>Max ident</th>
<th>Accession</th>
</tr>
</thead>
<tbody>
<tr>
<td>kinesin-1 heavy chain [Homo sapiens]</td>
<td>677</td>
<td>677</td>
<td>100%</td>
<td>0</td>
<td>100%</td>
<td>NP_004512.1</td>
</tr>
<tr>
<td>Kif5b protein [Mus musculus]</td>
<td>676</td>
<td>676</td>
<td>100%</td>
<td>0</td>
<td>98%</td>
<td>AAA20133.1</td>
</tr>
<tr>
<td>Kinesin-14 heavy chain [Danio rerio]</td>
<td>595</td>
<td>595</td>
<td>88%</td>
<td>0</td>
<td>78%</td>
<td>XP_00320703</td>
</tr>
<tr>
<td>hypothetical protein EGK_18589</td>
<td>42.7</td>
<td>52</td>
<td>40%</td>
<td>0.03</td>
<td>32%</td>
<td>ELK35081.1</td>
</tr>
</tbody>
</table>

**Alignment scores of unrelated sequences**

A score of 42.7 or better is expected to occur by chance 3 in 100 times (E-value = 0.03)
<table>
<thead>
<tr>
<th>Description</th>
<th>Max score</th>
<th>Total score</th>
<th>Query cover</th>
<th>E value</th>
<th>Max ident</th>
<th>Accession</th>
</tr>
</thead>
<tbody>
<tr>
<td>kinesin-1 heavy chain [Homo sapiens]</td>
<td>677</td>
<td>677</td>
<td>100%</td>
<td>0</td>
<td>100%</td>
<td>NP_004512.1</td>
</tr>
<tr>
<td>Kif5b protein [Mus musculus]</td>
<td>676</td>
<td>676</td>
<td>100%</td>
<td>0</td>
<td>98%</td>
<td>AAA20133.1</td>
</tr>
<tr>
<td>KinesinV14 heavy chain [Danio rerio]</td>
<td>595</td>
<td>595</td>
<td>88%</td>
<td>0</td>
<td>78%</td>
<td>XP_00320703</td>
</tr>
<tr>
<td>Hypothetical protein EGK_18589</td>
<td>42.7</td>
<td>52</td>
<td>40%</td>
<td>0.03</td>
<td>32%</td>
<td>ELK35081.1</td>
</tr>
</tbody>
</table>

In general $E$ values < 0.005 are usually significant.

To find out more about $E$ values see: “The Statistics of Sequence Similarity Scores” available in the help section of the NCBI BLAST site:

Outline for today

• Alignment basics
  ‣ Why compare biological sequences?

• Homologue detection
  ‣ Orthologs, paralogs, similarity and identity
  ‣ Sequence changes during evolution
  ‣ Alignment view: matches, mismatches and gaps

• Pairwise sequence alignment methods
  ‣ Brute force alignment
  ‣ Dot matrices
  ‣ Dynamic programing
    (global vs local alignment)

• Rapid heuristic approaches
  ‣ BLAST

• Practical database searching
  ‣ BLAST, PSI-BLAST and HMM approaches
Practical database searching with BLAST

NCBI BLAST Home Page
Practical database searching with BLAST

• There are four basic components to a traditional BLAST search
  – (1) Choose the sequence (query)
  – (2) Select the BLAST program
  – (3) Choose the database to search
  – (4) Choose optional parameters

• Then click “BLAST”
Step 1: Choose your sequence

- Sequence can be input in FASTA format or as accession number
Step 2: Choose the BLAST program

- **Rat**
  - Arabidopsis thaliana
- **Danio rerio**
  - Drosophila melanogaster
- **Microbes**
  - Apis mellifera

**Basic BLAST**

Choose a BLAST program to run.

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
</table>
| nucleotide blast | Search a nucleotide database using a nucleotide query
| protein blast  | Search protein database using a protein query
| blastx         | Search protein database using a translated nucleotide query
| tblastn        | Search translated nucleotide database using a protein query
| tblastx        | Search translated nucleotide database using a translated nucleotide query

**Specialized BLAST**

Choose a type of specialized search (or database name in parentheses.)
Step 2: Choose the BLAST program

<table>
<thead>
<tr>
<th>Query</th>
<th>Database</th>
</tr>
</thead>
<tbody>
<tr>
<td>blastn</td>
<td>DNA</td>
</tr>
<tr>
<td>blastp</td>
<td>protein</td>
</tr>
<tr>
<td>blastx</td>
<td>DNA</td>
</tr>
<tr>
<td>tblastn</td>
<td>protein</td>
</tr>
<tr>
<td>tblastx</td>
<td>DNA</td>
</tr>
</tbody>
</table>

1. blastn DNA → DNA
2. blastp protein → protein
3. blastx DNA → protein
4. blastx DNA → protein
5. tblastn protein → DNA
6. tblastx DNA → DNA
7. tblastx DNA → DNA
DNA potentially encodes six proteins

5’ CAT CAA
5’ ATC AAC
5’ TCA ACT

5’ CATCAACTACAACCTCCAAAGACACCCCTTACACATCAACAAACCTACCCAC 3’
3’ GTAGTTGATGGTTGAGGTTTCTGTGGGAATGTGTAGTTGTTTGGATGGGTG 5’

5’ GTG GGT
5’ TGG GTA
5’ GGG TAG
Protein BLAST: search protein databases using a protein query

Enter Query Sequence

Enter accession number(s), gi(s), or FASTA sequence(s) [Clear]

Or, upload file [Choose File] no file selected

Job Title

[Align two or more sequences]

Choose Search Set

Database [Non-redundant protein sequences (nr)]

Organism [Optional]

Exclude [Optional]

Entreze Query [Optional]

Program Selection

Algorithm [blastp (protein-protein BLAST)]

Search database Non-redundant protein sequences (nr) using Blastp (protein-protein BLAST)

Show results in a new window
Step 3: Choose the database

nr = non-redundant (most general database)
dbest = database of expressed sequence tags
dbsts = database of sequence tag sites
gss = genomic survey sequences

protein databases
nucleotide databases
Step 4a: Select optional search parameters

- Scoring matrix: BLOSUM62
- Expect: 10
- Word size: 3
- Max target sequences: 100
Step 4: Optional parameters

• You can...
  – choose the organism to search
  – change the substitution matrix
  – change the expect (E) value
  – change the word size
  – change the output format
Results page

NCBI Blast: gi|4504349|ref|NP_000509.1| hemoglobin

Query ID: id|84677
Description: gi|4504349|ref|NP_000509.1| hemoglobin subunit beta [Homo sapiens]
Molecule type: amino acid
Query Length: 147

Database Name: nr
Description: All non-redundant GenBank CDS translations+PDB+SwissProt+PIR+PRF excluding environmental samples from WGS projects
Program: BLASTP 2.2.27+

DELTA-BLAST, a more sensitive protein-protein search

Putative conserved domains have been detected, click on the image below for detailed results.

Query seq.
Specific hits
Superfamilies

Distribution of 100 Blast Hits on the Query superfamily

Graphical Summary

Show Conserved Domains
Further down the results page...
Further down the results page...

<table>
<thead>
<tr>
<th>Description</th>
<th>Max score</th>
<th>Total score</th>
<th>Query cover</th>
<th>E value</th>
<th>Max ident</th>
<th>Accession</th>
</tr>
</thead>
<tbody>
<tr>
<td>hemoglobin beta [synthetic construct]</td>
<td>301</td>
<td>301</td>
<td>100%</td>
<td>9e-103</td>
<td>100%</td>
<td>AAX37051.1</td>
</tr>
<tr>
<td>hemoglobin beta [synthetic construct]</td>
<td>301</td>
<td>301</td>
<td>100%</td>
<td>1e-102</td>
<td>100%</td>
<td>AAX29557.1</td>
</tr>
<tr>
<td>hemoglobin subunit beta [Homo sapiens] &gt; ref</td>
<td>XP_508242.1</td>
<td>PREDICTED: hemoglobin s</td>
<td>301</td>
<td>301</td>
<td>100%</td>
<td>9e-103</td>
</tr>
<tr>
<td>RecName: Full=Hemoglobin subunit beta; AltName: Full=Beta-globin; AltName: Full=Hemoglobin s</td>
<td>300</td>
<td>300</td>
<td>100%</td>
<td>9e-103</td>
<td>100%</td>
<td>AAX37051.1</td>
</tr>
<tr>
<td>beta globin chain variant [Homo sapiens]</td>
<td>299</td>
<td>299</td>
<td>100%</td>
<td>5e-102</td>
<td>99%</td>
<td>AAN84548.1</td>
</tr>
<tr>
<td>beta globin [Homo sapiens] &gt; gb</td>
<td>AAZ39781.1</td>
<td>beta globin [Homo sapiens] &gt; gb</td>
<td>AAZ39781</td>
<td>299</td>
<td>299</td>
<td>100%</td>
</tr>
<tr>
<td>beta-globin [Homo sapiens]</td>
<td>299</td>
<td>299</td>
<td>100%</td>
<td>5e-102</td>
<td>99%</td>
<td>AAN84548.1</td>
</tr>
<tr>
<td>hemoglobin beta chain [Homo sapiens]</td>
<td>299</td>
<td>299</td>
<td>100%</td>
<td>6e-102</td>
<td>99%</td>
<td>AAD19696.1</td>
</tr>
<tr>
<td>Chain B, Structure Of Haemoglobin In The Deoxy Quaternary State With Ligand Bound Ai</td>
<td>298</td>
<td>298</td>
<td>99%</td>
<td>9e-102</td>
<td>100%</td>
<td>1COH_B</td>
</tr>
<tr>
<td>hemoglobin beta subunit variant [Homo sapiens] &gt; gb</td>
<td>AAA88054.1</td>
<td>beta-globin [Homo sapiens] &gt; gb</td>
<td>AAZ39781</td>
<td>298</td>
<td>298</td>
<td>100%</td>
</tr>
<tr>
<td>Chain B, High-Resolution X-Ray Study Of Deoxy Recombinant Human Hemoglobins Syn</td>
<td>297</td>
<td>297</td>
<td>99%</td>
<td>3e-101</td>
<td>99%</td>
<td>1DXU_B</td>
</tr>
<tr>
<td>Chain B, Analysis Of The Crystal Structure, Molecular Modeling And Infrared Spectrosc</td>
<td>297</td>
<td>297</td>
<td>99%</td>
<td>3e-101</td>
<td>99%</td>
<td>1HDB_B</td>
</tr>
</tbody>
</table>
Further down the results page...

### Hemoglobin subunit beta [Homo sapiens]
- **Sequence ID:** ref[NP_000509.1]
- **Length:** 147
- **Number of Matches:** 1

<table>
<thead>
<tr>
<th>Range 1: 1 to 147</th>
<th>GenPept</th>
<th>Graphics</th>
<th>Score</th>
<th>Expect</th>
<th>Method</th>
<th>Identities</th>
<th>Positives</th>
<th>Gaps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>301 bits(770)</td>
<td>1e-102</td>
<td>Compositional matrix adjust.</td>
<td>147/147(100%)</td>
<td>147/147(100%)</td>
<td>0/147(0%)</td>
</tr>
<tr>
<td>Query 1</td>
<td>MVHTLPEKSAVTALNGWKNVDEVGGEALRGLVVYPWTQRFLESGDSLTPDAVMNFK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sbjct 1</td>
<td>MVHTLPEKSAVTALNGWKNVDEVGGEALRGLVVYPWTQRFLESGDSLTPDAVMNFK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Query 61</td>
<td>VKAHKGGKLASSGDSNLHDNLKGGTATLSELCDKLHVDPEFFRLLGNVLVCVLAHFFG</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sbjct 61</td>
<td>VKAHKGGKLASSGDSNLHDNLKGGTATLSELCDKLHVDPEFFRLLGNVLVCVLAHFFG</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Query 121</td>
<td>KEFTPVQAAYKVVAGVANALAHKYH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sbjct 121</td>
<td>KEFTPVQAAYKVVAGVANALAHKYH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Related Information
- **Gene** - associated gene details
- **UniGene** - clustered expressed sequence tags
- **Map Viewer** - aligned genomic context
- **Structure** - 3D structure displays
- **PubChem Bio**
- **Assay** - bioactivity screening

### RecName: Full=Hemoglobin subunit beta; AltName: Full=Beta-globin; AltName: Full=Hemoglobin beta chain
- **Sequence ID:** sp|P02024.2|HBB_GORGO
- **Length:** 147
- **Number of Matches:** 1

<table>
<thead>
<tr>
<th>Range 1: 1 to 147</th>
<th>GenPept</th>
<th>Graphics</th>
<th>Score</th>
<th>Expect</th>
<th>Method</th>
<th>Identities</th>
<th>Positives</th>
<th>Gaps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>300 bits(767)</td>
<td>4e-102</td>
<td>Compositional matrix adjust.</td>
<td>146/147(99%)</td>
<td>147/147(100%)</td>
<td>0/147(0%)</td>
</tr>
<tr>
<td>Query 1</td>
<td>MVHTLPEKSAVTALNGWKNVDEVGGEALRGLVVYPWTQRFLESGDSLTPDAVMNFK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sbjct 1</td>
<td>MVHTLPEKSAVTALNGWKNVDEVGGEALRGLVVYPWTQRFLESGDSLTPDAVMNFK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Query 61</td>
<td>VKAHKGGKLASSGDSNLHDNLKGGTATLSELCDKLHVDPEFFRLLGNVLVCVLAHFFG</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sbjct 61</td>
<td>VKAHKGGKLASSGDSNLHDNLKGGTATLSELCDKLHVDPEFFRLLGNVLVCVLAHFFG</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Query 121</td>
<td>KEFTPVQAAYKVVAGVANALAHKYH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sbjct 121</td>
<td>KEFTPVQAAYKVVAGVANALAHKYH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Different output formats are available
E.g. Query anchored alignments

<table>
<thead>
<tr>
<th>Query</th>
<th>1</th>
<th>MVHLTPKSAVTALGWKVDEVGEALGRLLVYYQRTFESGFGLSTPDAGMNPK</th>
<th>60</th>
</tr>
</thead>
</table>
... and alignments with dots for identities
Common problems

• Selecting the wrong version of BLAST
• Selecting the wrong database
• Too many hits returned
• Too few hits returned
• Unclear about the significance of a particular result - are these sequences homologous?
How to handle too many results

• Focus on the question you are trying to answer
  – select “refseq” database to eliminate redundant matches from “nr”
  – Limit hits by organism
  – Use just a portion of the query sequence, when appropriate
  – Adjust the expect value; lowering $E$ will reduce the number of matches returned
How to handle too few results

• Many genes and proteins have no significant database matches
  – remove Entrez limits
  – raise E-value threshold
  – search different databases
  – try scoring matrices with lower BLOSUM values (or higher PAM values)
  – use a search algorithm that is more sensitive than BLAST (e.g. PSI-BLAST or HMMer)
Side note: Scoring matrices

- A substitution matrix contains values proportional to the probability that amino acid \(i\) mutates into amino acid \(j\) for all pairs of amino acids.
- Substitution matrices are constructed by assembling a large and diverse sample of verified pairwise alignments (or multiple sequence alignments) of amino acids.
- Substitution matrices should reflect the probabilities of mutations occurring through a period of evolution.
- The two major types of substitution matrices are PAM and BLOSUM.
BLOSUM62 is the default BLASTp scoring matrix

• BLOSUM matrices are based on short, ungapped blocks of conserved amino acid sequences from multiple alignments
  – members of a block that have a most X percent sequence identity to each other are used to generate a BLOSUMX matrix
  – For example, using a cutoff of 62% identity will generate the BLOSUM62 matrix

• PAM matrices are similar but built from multiple alignments where amino acid substitutions are at rate of 1% (PAM 1)
  – Matrix multiplication is used generate higher PAM matrices
  – PAM3 = (PAM1 x PAM1 x PAM1) etc...
By default BLASTp Match scores come from the BLOSUM62 matrix

Note. Some amino acid mismatches have positive scores – highlighted in red.
Protein scoring matrices reflect the properties of amino acids.
Two problems standard BLAST cannot solve

• 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 (HMMs) 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 now available such as Megablast
PSI-BLAST: Position specific iterated BLAST

- The purpose of PSI-BLAST is to look deeper into the database for matches to your query protein sequence by employing a scoring matrix that is customized to your query.
  - PSI-BLAST constructs a multiple sequence alignment from the results of a first round BLAST search and then creates a “profile” or specialized position-specific scoring matrix (PSSM) for subsequent search rounds.
**PSI-BLAST**: Position-Specific Iterated BLAST

- Many proteins in a database are too distantly related to a query to be detected using standard BLAST. In many other cases matches are detected but are so distant that the inference of homology is unclear. Enter the more sensitive PSI-BLAST

1. BLAST input sequence to find significant alignments
2. Construct a multiple sequence alignment (MSA)
3. Construct a PSSM
4. BLAST PSSM profile to search for new hits
5. Iterate
Inspect the blastp output to identify empirical “rules” regarding amino acids tolerated at each position.
|   | A | R | N | D | C | Q | E | G | H | I | L | K | M | F | P | S | T | W | Y | V |
| 1 | M | -1 | -2 | -3 | -2 | -1 | -2 | -1 | -2 | -3 | 1 | 2 | -2 | 6 | 0 | 2 | 3 | 1 | 2 | 1 | 1 |
| 2 | K | -1 | 1 | 0 | -1 | -4 | 2 | 4 | -2 | 0 | -3 | -3 | 3 | -2 | -4 | -1 | 0 | -1 | -3 | -2 | 3 |
| 3 | W | -3 | -3 | -4 | -5 | -3 | -2 | -3 | -3 | -3 | -3 | -2 | -3 | -2 | -3 | -2 | -3 | 1 | -4 | -3 | -3 | 12 | 2 | -3 |
| 4 | V | 0 | -3 | -3 | -4 | -1 | -3 | -3 | -4 | -4 | 3 | 1 | -3 | 1 | -1 | -3 | -2 | 0 | -3 | -3 | -2 | 0 | 0 | 0 |
| 5 | W | -3 | -3 | -4 | -5 | -3 | -2 | -3 | -3 | -3 | -3 | -2 | -3 | -2 | -3 | -2 | -3 | 1 | -4 | -3 | -3 | 12 | 2 | -3 |
| 6 | A | 5 | -2 | -2 | -2 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 | 0 | 0 | 0 |
| 7 | L | -2 | -2 | -4 | -4 | -1 | -2 | -3 | -4 | -3 | 2 | 4 | -3 | 2 | 0 | -3 | -3 | -1 | -2 | -1 | 1 | 0 | 0 | 0 |
| 8 | L | -1 | -3 | -3 | -4 | -1 | -3 | -3 | -4 | -3 | 2 | 2 | -3 | 1 | 3 | -3 | -2 | -1 | -2 | -1 | 2 | 0 | 0 | 0 |
| 9 | L | -1 | -3 | -4 | -4 | -1 | -2 | -3 | -4 | -3 | 2 | 4 | -3 | 2 | 0 | -3 | -3 | -1 | -2 | -1 | 2 | 0 | 0 | 0 |
| 10 | L | -2 | -2 | -4 | -4 | -1 | -2 | -3 | -4 | -3 | 2 | 4 | -3 | 2 | 0 | -3 | -3 | -1 | -2 | -1 | 1 | 0 | 0 | 0 |
| 11 | A | 5 | -2 | -2 | -2 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 | 0 | 0 | 0 |
| 12 | A | 5 | -2 | -2 | -2 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 | 0 | 0 | 0 |
| 13 | W | -2 | -2 | -4 | -4 | -1 | -2 | -3 | -4 | -3 | 2 | 4 | -3 | 2 | 0 | -3 | -3 | -1 | -2 | -1 | 1 | 0 | 0 | 0 |
| 14 | A | 3 | -2 | -1 | -2 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 | 0 | 0 | 0 |
| 15 | A | 2 | -1 | 0 | -1 | -2 | 2 | 1 | 0 | 2 | -1 | -3 | -1 | 3 | 0 | -3 | -2 | -2 | 0 | 0 | 0 | 0 |
| 16 | A | 4 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | -1 | 1 | 0 | -3 | -2 | -2 | 0 | 0 | 0 | 0 | 0 | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 37 | S | 2 | -1 | -2 | -3 | -2 | -1 | -2 | -3 | -1 | 4 | 1 | -3 | -2 | -2 | 0 | 0 | -2 | -3 | -3 | -4 | 5 | -2 | 0 |
| 38 | G | 0 | -3 | -1 | -2 | -3 | 2 | 2 | 0 | 2 | 4 | -4 | -2 | -3 | -4 | -2 | 0 | -2 | -3 | -3 | -4 | 5 | -2 | 0 |
| 39 | T | 0 | -1 | 0 | -1 | -1 | -2 | -2 | -1 | -1 | -1 | -1 | -1 | -2 | -1 | 1 | 5 | -3 | -2 | 0 | 0 | 0 | 0 |
| 40 | W | -3 | -3 | -4 | -5 | -3 | -2 | -3 | -3 | -3 | -3 | -2 | -3 | -2 | 1 | -4 | -3 | -3 | 12 | 2 | -3 | 0 | 0 |
| 41 | Y | -2 | -2 | -2 | -3 | -3 | -2 | -2 | -3 | 2 | 2 | -2 | -1 | -2 | -1 | 1 | 3 | -3 | -2 | -2 | 2 | 7 | -1 |
| 42 | A | 4 | -2 | -2 | -2 | -1 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 | 0 | 0 | 0 |
|   | A | R | N | D | C | Q | E | G | H | I | L | K | M | F | P | S | T | W | Y | V |
| 1 | M | -1 | -2 | -2 | -3 | -2 | -1 | -2 | -3 | -2 | 1 | 2 | -2 | 6 | 0 | -3 | -2 | -1 | -2 | -1 | 1 |
| 2 | K | -1 | 1 | 0 | 1 | -4 | 2 | 4 | -2 | 0 | -3 | -3 | 3 | -2 | -4 | -1 | 0 | -1 | -3 | -2 | -3 |
| 3 | W | -3 | -3 | -4 | -5 | -3 | -2 | -3 | -3 | -3 | -2 | -2 | -3 | -2 | 1 | -4 | -3 | -3 | 12 | 2 | -3 |
| 4 | V | 0 | -3 | -3 | -4 | -1 | -3 | -3 | -4 | -4 | 3 | 1 | -3 | 1 | -1 | -3 | -2 | 0 | -3 | -1 | 4 |
| 5 | W | -3 | -3 | -4 | -5 | -3 | -2 | -3 | -3 | -3 | -2 | -3 | -2 | 1 | -4 | -3 | -3 | 12 | 2 | -3 |
| 6 | A | -2 | -2 | -2 | -1 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 |
| 7 | L | -2 | -2 | -4 | -4 | -1 | -2 | -3 | -4 | -3 | 2 | 4 | -3 | 2 | 0 | -3 | -3 | -1 | -2 | -1 | 1 |
| 8 | L | -1 | -3 | -3 | -4 | -1 | -3 | -3 | -4 | -3 | 2 | 2 | -3 | 1 | 3 | -3 | -2 | -1 | -2 | 0 | 3 |
| 9 | L | -1 | -3 | -4 | -4 | -1 | -3 | -3 | -4 | -3 | 2 | 2 | -3 | 1 | 3 | -3 | -2 | -1 | -2 | 0 | 3 |
| 10 | L | -2 | -2 | -4 | -4 | -1 | -2 | -3 | -4 | -3 | 2 | 4 | -3 | 2 | 0 | -3 | -3 | -1 | -2 | -1 | 1 |
| 11 | A | -2 | -2 | -2 | -2 | -1 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 |
| 12 | A | -2 | -2 | -2 | -2 | -1 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 |
| 13 | W | -2 | -3 | -4 | -4 | -1 | -2 | -3 | -4 | -3 | 2 | 4 | -3 | 2 | 0 | -3 | -3 | -1 | -2 | -1 | 1 |
| 14 | A | -2 | -1 | -2 | -2 | -1 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 |
| 15 | A | -1 | 0 | -1 | -1 | 0 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 |
| 16 | A | -2 | -1 | -2 | -2 | -1 | -1 | -1 | 0 | -2 | -2 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 |
| 37 | S | 2 | -1 | 0 | -1 | 0 | -1 | 0 | -1 | -2 | -3 | 0 | -2 | -3 | -2 | -2 | -3 | -2 | -2 | -3 |
| 38 | G | 0 | -3 | -1 | -2 | 0 | -1 | 0 | -1 | -2 | -3 | 0 | -2 | -3 | -2 | -3 | -4 | -3 | -4 |
| 39 | T | 0 | -1 | 0 | -1 | 0 | -1 | 0 | -1 | -2 | -3 | 0 | -2 | -3 | -2 | -3 | -4 | -3 | -4 |
| 40 | W | -3 | -3 | -4 | -5 | -3 | -2 | -3 | -4 | -3 | -2 | -3 | -2 | 2 | 1 | -4 | -3 | -3 | 12 | 2 | -3 |
| 41 | Y | -2 | -2 | -2 | -3 | -2 | -2 | -3 | -2 | -2 | -3 | -2 | -2 | 2 | 7 | -1 | 3 | -3 | -2 | -2 |

Note that a given amino acid (such as alanine) in your query protein can receive different scores for matching alanine—depending on the position in the protein.
The PSI-BLAST PSSM is essentially a query customized scoring matrix that is more sensitive than PAM or BLOSUM.

Note that a given amino acid (such as alanine) in your query protein can receive different scores for matching alanine—depending on the position in the protein.
Retinol-binding protein

Start search with single human RBD sequence

Odorant binding protein

Apolipoprotein D
Retinol-binding protein

Odorant binding protein

Apolipoprotein D

Result of initial blastp search
Retinol-binding protein

Odorant binding protein

Result of subsequent PSI-BLAST iteration (note, many more lipocalin hits returned!)

Apolipoprotein D
Retinol-binding protein

Odorant binding protein

Apolipoprotein D

Potential Lipocalins?

Result of later PSI-BLAST iteration (note, potential “corruption”!)
PSI-BLAST returns dramatically more hits

- The search process is continued iteratively, typically about five times, and at each step a new PSSM is built
  - You must decide how many iterations to perform and which sequences to include!
  - You can stop the search process at any point - typically whenever few new results are returned or when no new “sensible” results are found

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Hits with E &lt; 0.005</th>
<th>Hits with E &gt; 0.005</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>34</td>
<td>61</td>
</tr>
<tr>
<td>2</td>
<td>314</td>
<td>79</td>
</tr>
<tr>
<td>3</td>
<td>416</td>
<td>57</td>
</tr>
<tr>
<td>4</td>
<td>432</td>
<td>50</td>
</tr>
<tr>
<td>5</td>
<td>432</td>
<td>50</td>
</tr>
</tbody>
</table>

Human retinol-binding protein 4 (RBP4; P02753) was used as a query in a PSI-BLAST search of the RefSeq database.
HMMER3: a new generation of sequence homology search software

HMMER is used for searching sequence databases for homologs of protein sequences, and for making protein sequence alignments. It implements methods using probabilistic models called profile hidden Markov models (profile HMMs).

Compared to BLAST, FASTA, and other sequence alignment and database search tools based on older scoring methodology, HMMER aims to be significantly more accurate and more able to detect remote homologs because of the strength of its underlying mathematical models. In the past, this strength came at significant computational expense, but in the new HMMER3 project, HMMER is now essentially as fast as BLAST.

As part of this evolution in the HMMER software, we are committed to making the software available to as many scientists as possible. Earlier releases of HMMER were restricted to command line use. To make the software more accessible to the wide scientific community, we now provide servers that allow sequence searches to be performed interactively via the Web.

The current version is HMMER 3.0 (28 March 2010) and can be downloaded from the software section of the site. Previous versions of the HMMER software can be obtained from the archive section.

If you have used the HMMER website, please consider citing the following reference that describes this work:

HMMER web server: interactive sequence similarity searching
R.D. Finn, J. Clements, S.R. Eddy
protein sequence vs protein sequence database

Paste in your sequence or use the example

>sp|Q14807|KIF22_HUMAN
MAAGGSTQQRREMAAASAISSGAGRCLSLGATRPPPARRVAVRLRPFDPTAGA
SDPPCVRGMDSYCESLTNRQHKELKTQFQDFAFYGERSTQDDIYAGSVPILRHILRQGN
ASVLAYGPTGAKTHTMGLSPEQPGVIRPLRMLQLITRECEAEGRPWALSVTMSYELIY
QEKVVLIDLDPSGDLVIREDCRGNILIPQLSISIPSSSADFIREHFLPBRSSRTVGATRLN
QRRSSRSHAVLLKVDQQRERLAPFFQREQGKLYIDLGEDSNRRTGNKGLRLKESGAIINTS
LFVLGKSVDLNLQQLPRVYPSKRTLRQSLSLGSXASILIANAPIERRFFYLFDTVSAFLN
FAARSKVIVNRPTNESLQPHAPVZLLSQKELPPAEARSPEEEEGSPEMMAAPA
SASQKLSLPQKLSSMDFPPMLHELRLQDLRLASQGSGQPALLSPKRRMRVLMTKVEXEKL
ELERLTKQKKELEAKMLAQAEEKNCHTPMRPLRSLHTYGTAKPLKAVVMPLQIQEQ
AASPNAEILHIKKNKCRKLSELDALEPEEAQECWELQISPELAHGRQKILDLNENGS
ARDRILSICRGRKCAOLIVGRCWELHBRGSOVQELDERVEQCTKOMESLKLHIAAOGO

Submit  Reset

Comments or questions on the site? Send a mail to hmm@janelia.hhmi.org
Howard Hughes Medical Institute

Follow @hmm3r
### Pfam Domains

![Diagram of Pfam Domains](image)

Show hit details

### Distribution of Significant Hits

![Histogram of Distribution of Significant Hits](image)

### Query Matches (5100)

<table>
<thead>
<tr>
<th>Target</th>
<th>Description</th>
<th>Species</th>
<th>E-value</th>
<th>Alignments</th>
</tr>
</thead>
<tbody>
<tr>
<td>123979736</td>
<td>kinesin family member 22</td>
<td>synthetic construct</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>6453818</td>
<td>kinesin-like protein KIF22</td>
<td>Homo sapiens</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>30584615</td>
<td>Homo sapiens kinesin-like 4</td>
<td>synthetic construct</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>123994513</td>
<td>kinesin family member 22</td>
<td>synthetic construct</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>189053342</td>
<td>unnamed protein product</td>
<td>Homo sapiens</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>62898423</td>
<td>kinesin family member 22 variant</td>
<td>Homo sapiens</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>332845643</td>
<td>PREDICTED: kinesin family member 22 isoform 2</td>
<td>Pan troglodytes</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>75062021</td>
<td>RecName: Full=Kinesin-like protein KIF22</td>
<td>Homo sapiens</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>332266048</td>
<td>PREDICTED: kinesin-like protein KIF22-like isoform 1</td>
<td>Pongo abelii</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>297283748</td>
<td>PREDICTED: hypothetical protein LOC706401 isoform 3</td>
<td>Nomascus leucogenys</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>296219941</td>
<td>PREDICTED: LOW QUALITY PROTEIN: kinesin-like protein KIF22-like</td>
<td>Macaca mulatta</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>296196456</td>
<td>PREDICTED: kinesin-like protein KIF22-like</td>
<td>Callithrix jacchus</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>335284407</td>
<td>PREDICTED: kinesin-like protein KIF22-like</td>
<td>Callithrix jacchus</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>221046166</td>
<td>unnamed protein product</td>
<td>Sus scrofa</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>221045488</td>
<td>unnamed protein product</td>
<td>Homo sapiens</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
<tr>
<td>335284407</td>
<td>unnamed protein product</td>
<td>Homo sapiens</td>
<td>0.0e+00</td>
<td>show</td>
</tr>
</tbody>
</table>

Customize
HMMER
biosequence analysis using profile hidden Markov models

- **Job:** 9924F9AC-FEB5-11E0-A304-2B0C998A7913
- **Started:** 2011-10-24 23:01:15
- **Algorithm:** phmmer
- **HMMER Options:** -E 1 --domE 1 --incE 0.01 --incdomE 0.03 --mx BLOSUM62 --pextend 0.4 --popen 0.02 --seqdb nr

**Format**

- **FASTA**
  Download the significant hits from your search as a gzipped FASTA file.

- **Full length FASTA**
  A gzipped file containing the full length sequences for significant search hits.

- **Aligned FASTA**
  A gzipped file containing aligned significant search hits in FASTA format.

- **STOCKHOLM**
  Download an alignment of significant hits as a gzipped STOCKHOLM file.

- **Text**
  A plain text file containing the hit alignments and scores.

- **XML**
  An XML file formatted for machine parsing of the data.

- **JSON**
  All the results information encoded as a single JSON string.

- **HMM**
  Profile HMM downloads are not available.
Summary

- **Alignment basics**
  - Why compare biological sequences?

- **Homologue detection**
  - Orthologs, paralogs, similarity and identity
  - Sequence changes during evolution
  - Alignment view: matches, mismatches and gaps

- **Pairwise sequence alignment methods**
  - Brute force alignment
  - Dot matrices
  - Dynamic programming
    (global vs local alignment)

- **Rapid heuristic approaches**
  - BLAST

- **Practical database searching**
  - BLAST, PSI-BLAST and HMM approaches