COSC 348: Computing for Bioinformatics

Lecture 6:
Sequence Alignment – Local Alignment

Lubica Benuskova

http://www.cs.otago.ac.nz/cosc348/

Local sequence alignment

- By contrast to the global alignment, local alignments identify local regions of similarity between sequences of different lengths:

```
Global  FTFTALILLAVAV
        F--TAL-LLA-AV

Local   FTFTALILL-AVAV
        --FTAL-LLAAV--
```

- We distinguish two main approaches to the local alignment:
  - The **Smith-Waterman algorithm;**
  - **Word methods**, also known as **k-tuple methods**, implemented in the well-known families of programs FASTA and BLAST.

Smith-Waterman algorithm (SSEARCH)

- Variation of the Needleman-Wunsch algorithm. Thus, it is guaranteed to find the optimal local alignment (with respect to the scoring system being used).

- The difference to the Needleman-Wunsch algorithm is that *negative scoring matrix cells are set to zero*, which renders the local alignments visible. Backtracing ***starts at the highest scoring matrix cell and proceeds until a cell with score zero*** is encountered, yielding the highest scoring local alignment. We proceed with the second highest score, etc.

- The Smith-Waterman algorithm is costly: in order to align two sequences of lengths m and n, $O(mn)$ time and space are required.

Word (*k*-tuple) methods

- Word methods, also known as *k*-tuple methods, are heuristic methods that are not guaranteed to find an optimal alignment solution, but are significantly more efficient than Smith-Waterman algorithm.

- Word methods are especially useful in large-scale database searches where a large proportion of stored sequences will have essentially **no** significant match with the query sequence.

- Word methods are best known for their implementation in the database search tools **FASTA** and the **BLAST** family.

FASTA

- FASTA (pronounced “fast A”) is a sequence alignment software package.

- The current FASTA package contains programs for protein:protein, DNA:DNA, protein:translated DNA (with frameshifts), and ordered or unordered peptide searches, etc.

- FASTA is one of the bioinformatics services of the The European Bioinformatics Institute (EBI) located in U.K., which is part of European Molecular Biology Laboratory (EMBL) (centered in Germany).

FASTA: how it works

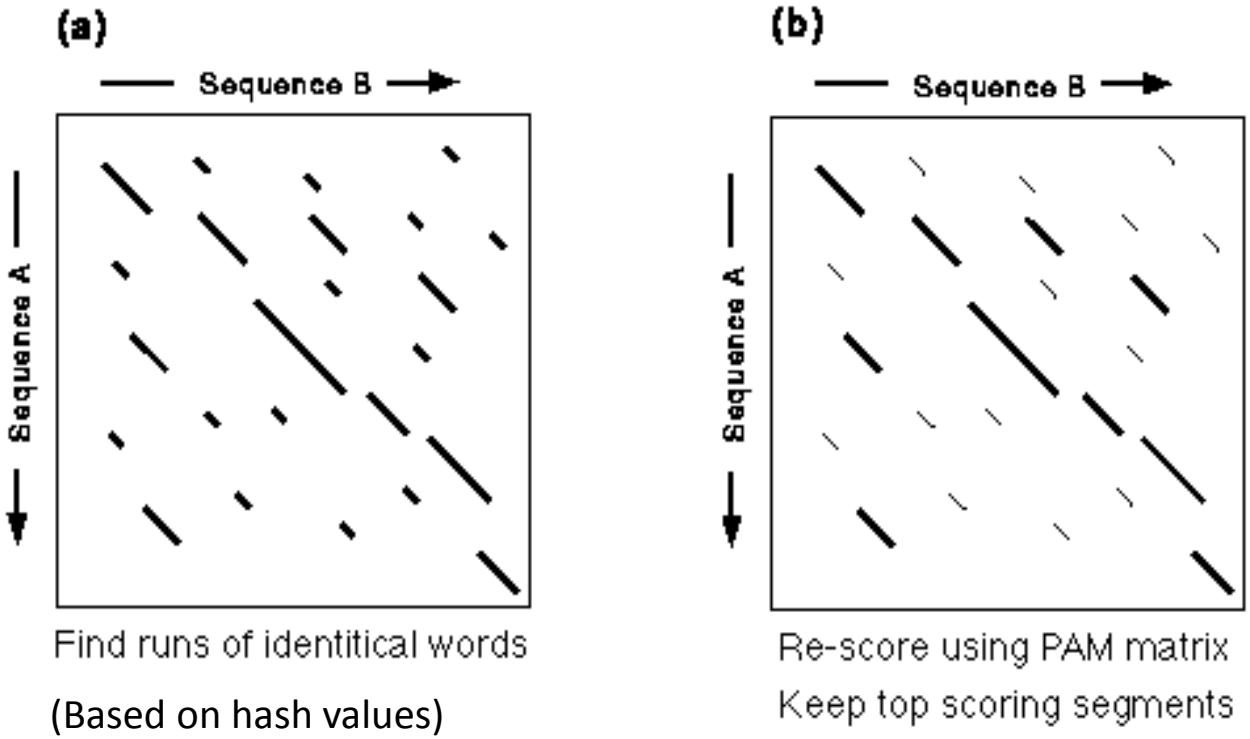
- Let us have a query sequence and a stored sequence.

- Identify a set of short non-overlapping strings (words, *k*-tuples) in the query sequence that will be matched against a stored sequence in the database.

- **Step1:** Initially the program stores word-to-word matches of a length *k* using a pattern search by the hash table. From the word hits that are returned, the program looks for segments that contain a cluster of nearby word hits. We have to define how many non-hits is allowed between nearby matching words so they form a cluster. *N* longest segments are stored.

FASTA – continuation

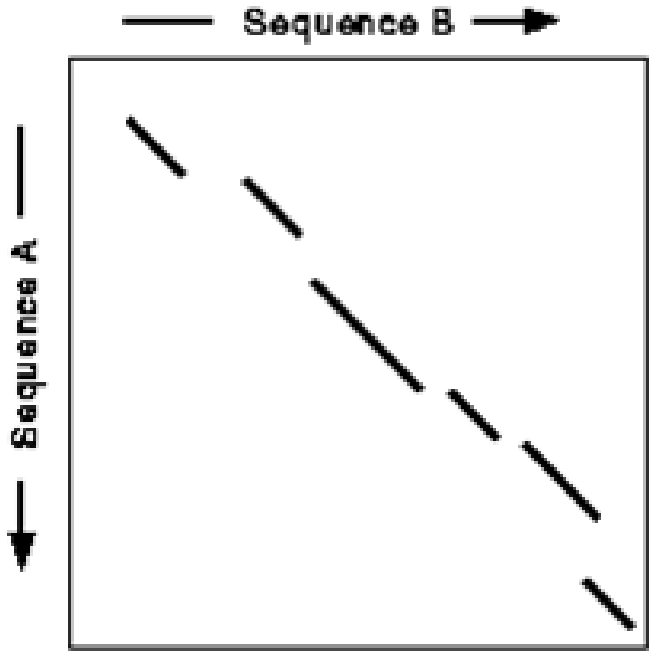
- **Step2:** Rescan the segments taken using the scoring matrix, while trimming the ends of the segments to include only those portions of segments that contribute highest to the segment score. A segment with the maximum score is identified. The highest score is referred to as **init1** score.



FASTA – continuation

Step3:

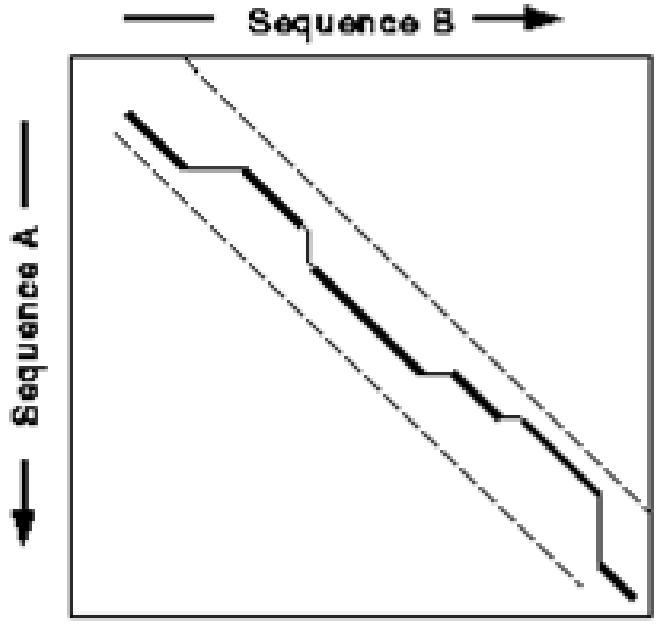
- Store segments with scores greater than a CUTOFF value. (This value is approximately one standard deviation above the average score expected from unrelated sequences in the database).



FASTA – continuation

Step3 (cont):

- Join these segments to form an approximate (global) alignment with gaps.
- Calculate the global alignment score that is the sum of the joined regions minus the penalties for gaps.



FASTA – continuation

Step4:

- This step uses a Smith-Waterman algorithm to create an optimised score (opt) for local alignment of query sequence to a each database sequence.

- It takes a band of 32 letters centered on the **init1** segment for calculating the optimal local alignment.

- After all sequences in the database are searched the program plots the scores of each database sequence in a histogram, and calculates the statistical significance of each.

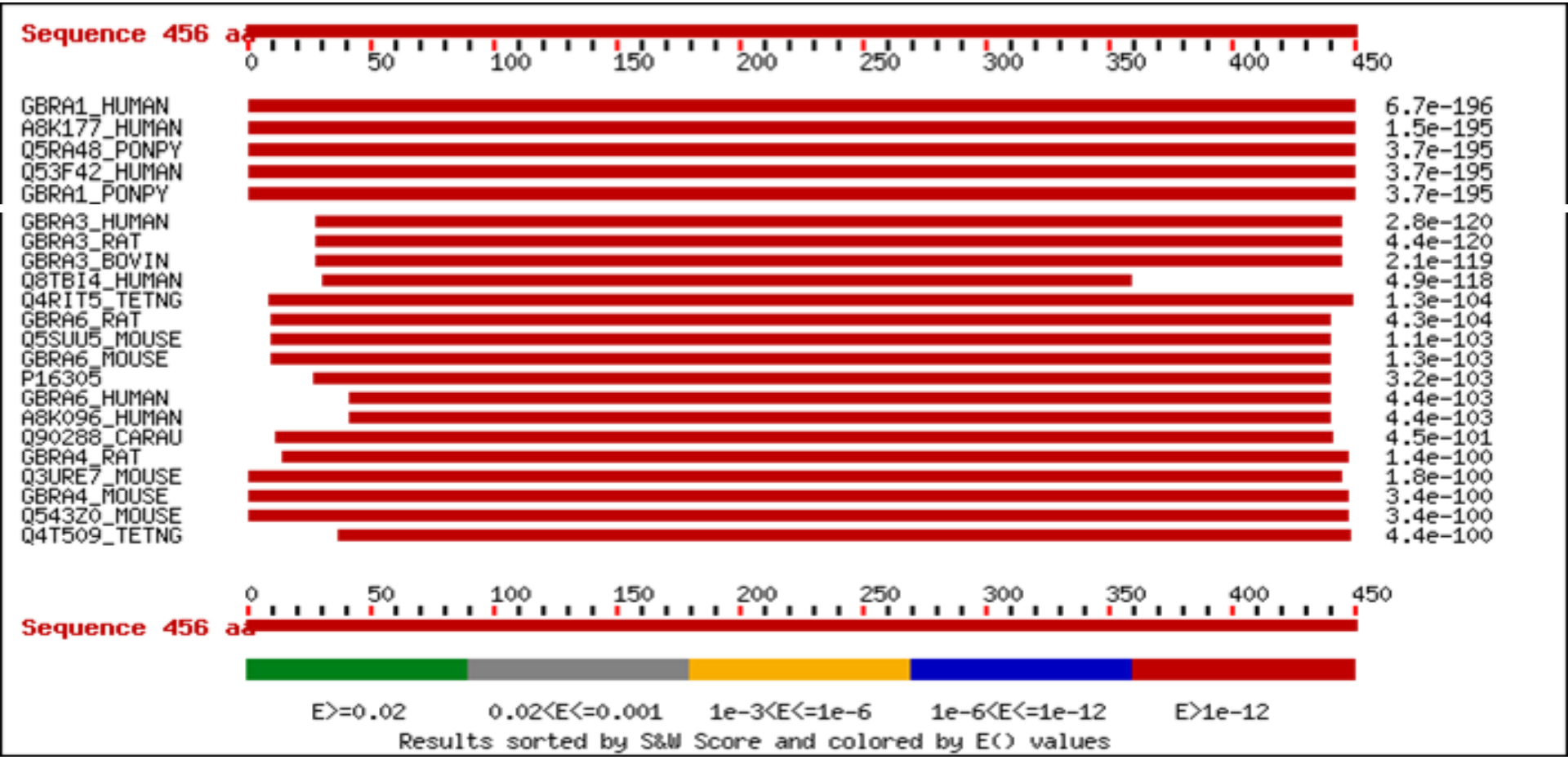
- The so-called E-value represents the likelihood that the observed alignment is due to chance alone. It has to be < 0.05.

Interpretation of results

- very low E(.) values (~ E-100) are *homologues* (homologs)
- Homology is an evolutionary statement which means "similarity from common ancestry"
- long list of gradually declining E(.) values indicates a large sequence (gene, protein, RNA) family
- long regions of moderate similarity are more significant than short regions of high identity

Example of result from FASTA

Query sequence is GBR1_HUMAN and the list of the most similar ones:



BLAST
(**B**asic **L**ocal **A**lignment **S**earch **T**ool

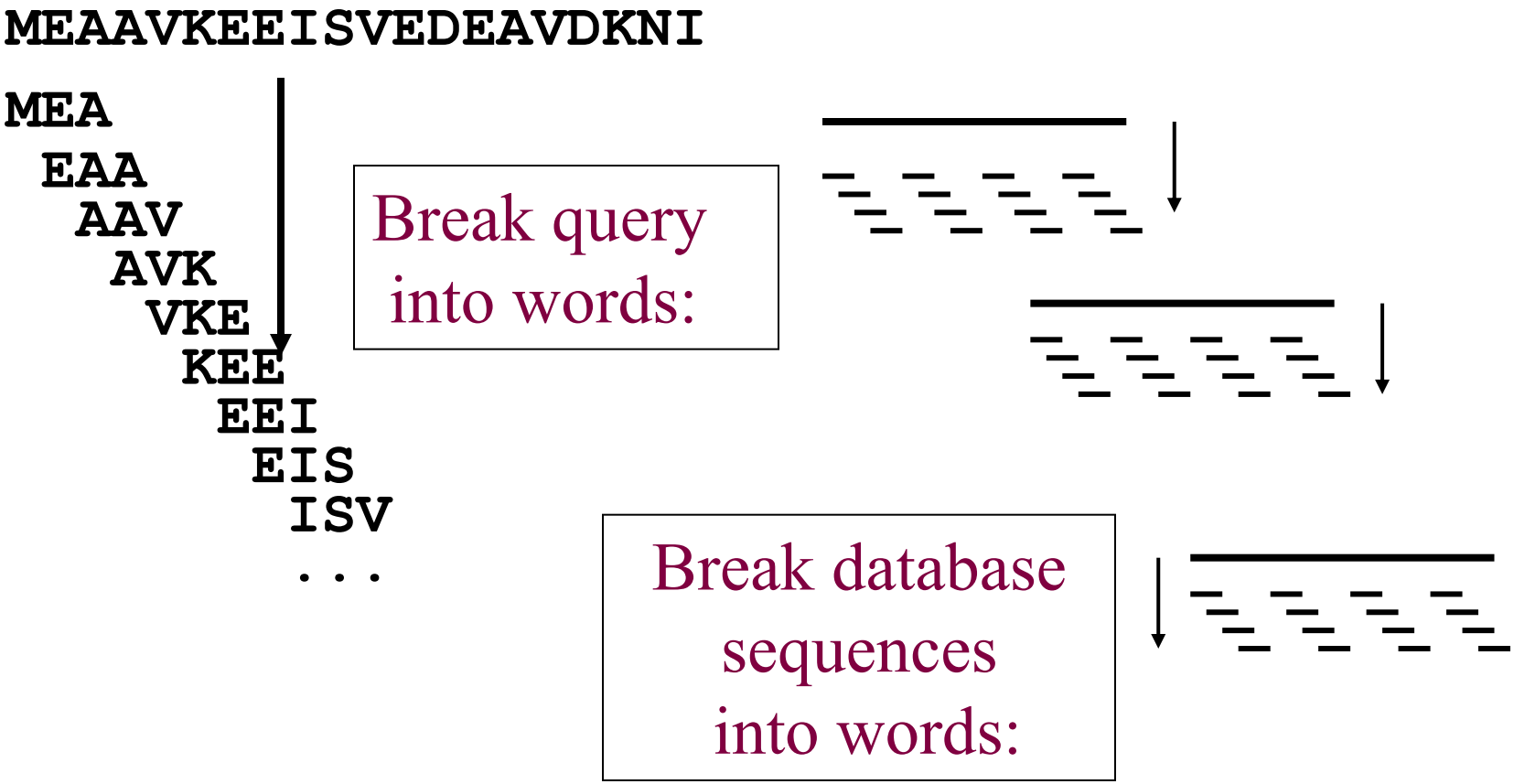
- One of the tools of the NCBI - The U.S. **N**ational **C**enter for **B**iotechnology **I**nformation.

- Uses word matching like FASTA

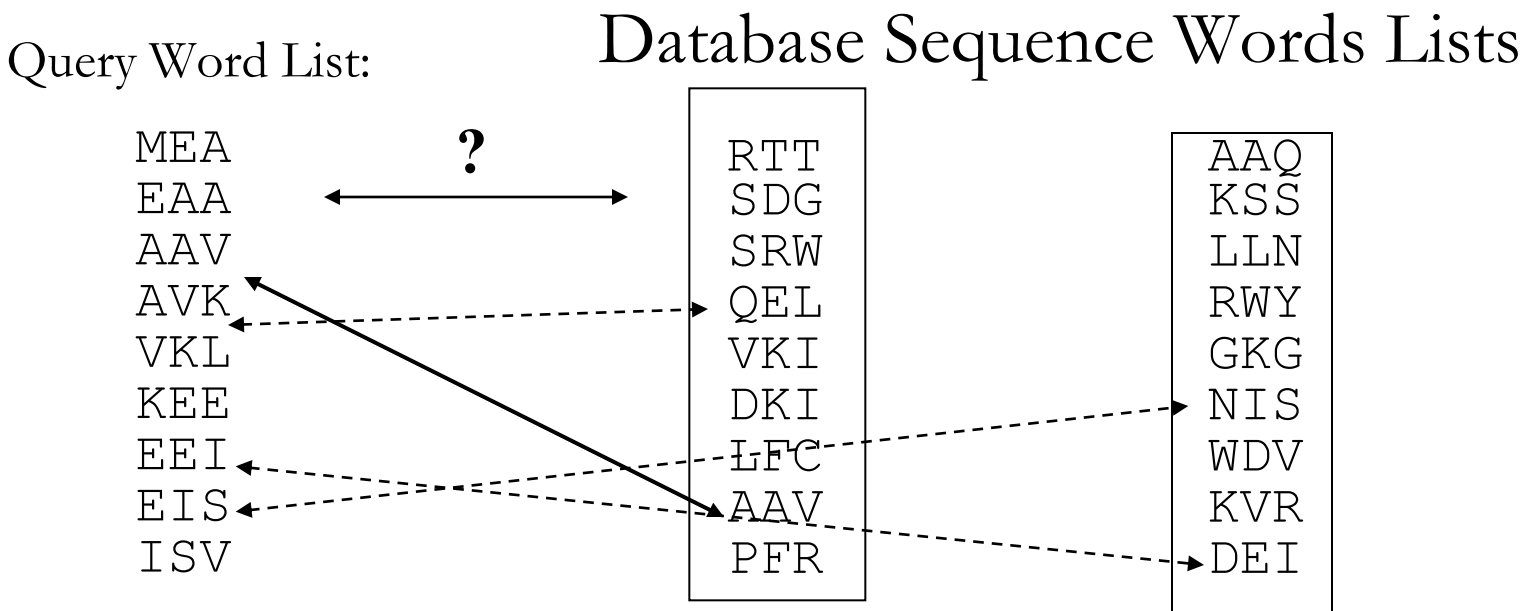
- Similarity matching of words (3 AA's, 11 bases/nucleotides)
  - does not require identical words.

- If no words are similar, then there is no alignment
  - won't find matches for very short sequences

BLAST word matching

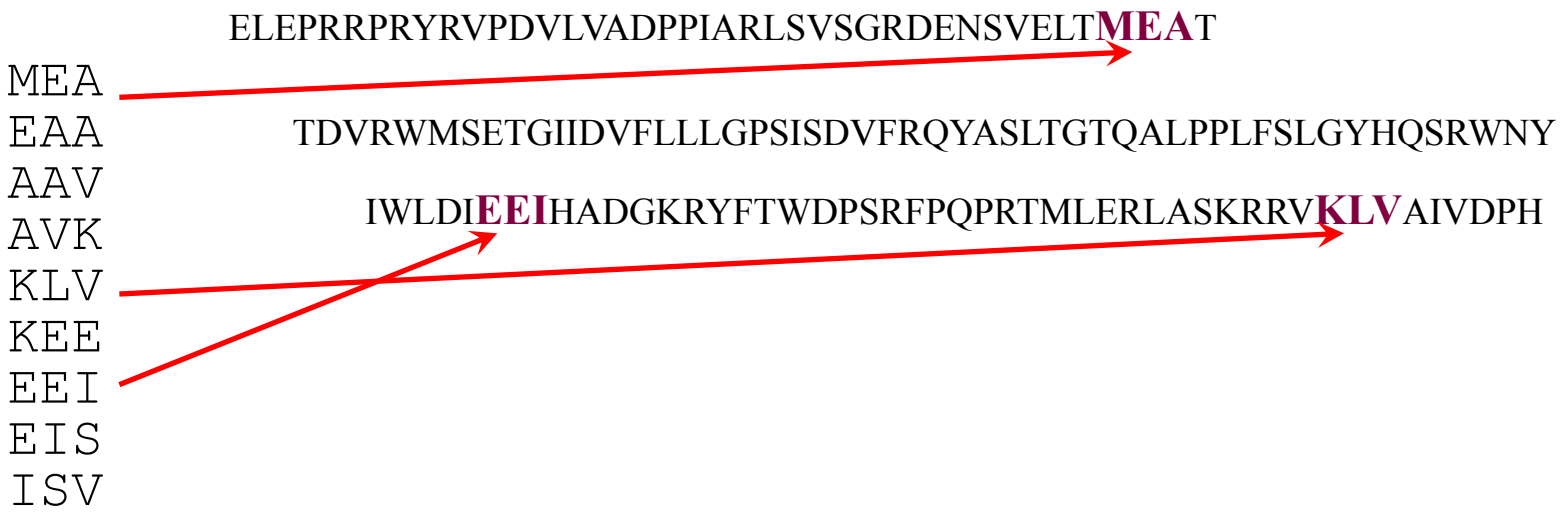


Compare word lists



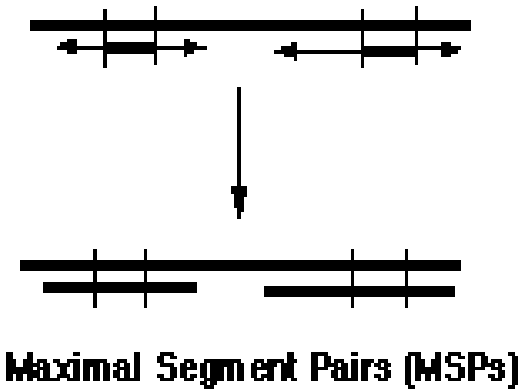
Compare word lists by Hashing & allow near matches!

Find locations of matching words in all sequences



Extend hits one base at a time

- Then BLAST extends the matches in both directions, starting at the seed. The un-gapped alignment process extends the initial seed match of length W in each direction in an order to boost the alignment score. Indels are not considered during this stage.



- In the last stage, BLAST performs a gapped alignment between the query sequence and the database sequence using a variation of the *Smith-Waterman algorithm*. Statistically significant alignments are then displayed to the user.

BLAST : example of result

- Job Title: P14867|GBRA1_HUMAN Gamma-aminobutyric acid...
  Show Conserved Domains
  Putative conserved domains have been detected, click on the image below for detailed results.
  *
  BLASTP 2.2.18 (Mar-02-2008) protein-protein BLAST

  Database: Non-redundant SwissProt sequences
  309,621 sequences; 115,465,120 total letters

  Query= P14867|GBRA1_HUMAN Gamma-aminobutyric acid receptor subunit alpha-1 - Homo sapiens (Human).
  Length=456

  Sequences producing significant alignments: (Bits) Value

  sp|P14867.3|GBRA1_HUMAN Gamma-aminobutyric acid receptor subu... 948 0.0 Gene info
  sp|Q5R6B2.1|GBRA1_PONPY Gamma-aminobutyric acid receptor subu... 944 0.0
  sp|Q4R534.1|GBRA1_MACFA Gamma-aminobutyric acid receptor subu... 944 0.0
  sp|P08219.1|GBRA1_BOVIN Gamma-aminobutyric acid receptor subu... 939 0.0 Gene info
  sp|P62813.1|GBRA1_RAT Gamma-aminobutyric acid receptor subuni... 908 0.0 Gene info
  sp|P19150.1|GBRA1_CHICK Gamma-aminobutyric acid receptor subu... 882 0.0 Gene info
  sp|P47869.2|GBRA2_HUMAN Gamma-aminobutyric acid receptor subu... 670 0.0 Gene info
  sp|P26048.1|GBRA2_MOUSE Gamma-aminobutyric acid receptor subu... 669 0.0 Gene info
  sp|P23576.1|GBRA2_RAT Gamma-aminobutyric acid receptor subuni... 669 0.0 Gene info
  sp|P10063.1|GBRA2_BOVIN Gamma-aminobutyric acid receptor subu... 667 0.0 Gene info
- sp|Q08E50.1|GBRA5_BOVIN Gamma-aminobutyric acid receptor subu... 641 0.0 Gene info
  sp|Q8BHJ7.1|GBRA5_MOUSE Gamma-aminobutyric acid receptor subu... 640 0.0 Gene info
  sp|P31644.1|GBRA5_HUMAN Gamma-aminobutyric acid receptor subu... 638 0.0 Gene info
  sp|P19969.1|GBRA5_RAT Gamma-aminobutyric acid receptor subuni... 636 0.0 Gene info
  sp|P34903.1|GBRA3_HUMAN Gamma-aminobutyric acid receptor subu... 632 0.0 Gene info
  sp|P26049.1|GBRA3_MOUSE Gamma-aminobutyric acid receptor subu... 630 6e-180 Gene info
  sp|P10064.1|GBRA3_BOVIN Gamma-aminobutyric acid receptor subu... 628 2e-179 Gene info
  sp|P20236.1|GBRA3_RAT Gamma-aminobutyric acid receptor subuni... 627 3e-179 Gene info
  sp|P30191.1|GBRA6_RAT Gamma-aminobutyric acid receptor subuni... 520 6e-147 Gene info
  sp|P16305.2|GBRA6_MOUSE Gamma-aminobutyric acid receptor subu... 518 2e-146 Gene info
  sp|Q90845.1|GBRA6_CHICK Gamma-aminobutyric acid receptor subu... 518 3e-146 Gene info

BLAST is approximate but fast

- BLAST makes similarity searches very quickly, but also makes errors
  - misses some important similarities
  - makes many incorrect matches

- The NCBI **BLAST** web server lets you compare your query sequence to various sequences stored in the GenBank;

- This is a VERY fast and powerful computer.

- The speed and relatively good accuracy of BLAST are the key why the tool is the most popular bioinformatics search tool.

What program to use for alignment?

1) **BLAST** is the fastest
   - limited sets of databases
   - nice translation tools, i.e. BLASTX (automatic translation of DNA query sequence to compare with protein databanks)
   - TBLASTN (automatic translation of an entire DNA database to compare with your protein query sequence)

2) **FASTA** works best
   - precise choice of databases
   - more sensitive for DNA-DNA comparisons
   - **FASTX** and **TFASTX** can find similarities in sequences with frameshifts

3) Smith-Waterman is slower, but even more sensitive
   - **SSEARCH** in **FASTA**

Multiple sequence alignment (MSA)

- Multiple sequence alignment (MSA) is an alignment of **> 2** sequences at a time; usually a query sequence and the database (library of sequences).

- MSA is used to identify conserved sequence regions across a group of sequences. Such conserved *sequence motifs* can be used for instance, to locate the catalytic sites of enzymes, promoter regions in DNA, etc.

- MSA is also used to find evolutionary relationships by constructing *phylogenetic trees* based on similarity of sequences.

- MSA is computationally difficult to produce and rigorous formulations of the problem lead to *NP-complete* combinatorial optimisation problems.

Dynamic programming methods

- Programs first perform pair-wise alignment on each pair of sequences (*using any of the pair-wise alignment methods*).

- Then, they perform local re-arrangements on these results, in order to optimise overlaps between multiple sequences. **The goal is to optimise *multiple* local alignments.**

- The so-called **"sum of pairs"** method has been implemented as a **scoring** method to evaluate these multiple alignments.

- The sum-of-pairs criterion means that the score of a multiple alignment of *N* sequences *is the sum of the N created pair-wise alignments*.

Progressive methods (ClustalW)

- Progressive, also known as hierarchical or tree methods, generate MSA by first aligning pair-wise the most similar sequences and then adding successively less related sequences.

- The initial tree describing the sequence relatedness is based on pair-wise comparisons for instance by FASTA or BLAST.

- Local re-arrangements are performed in order to optimise multiple overlaps. Scoring is based on sum of pairs.

- Progressive techniques automatically construct a phylogenetic tree as well as MSA (ClustalW).

Example of MSA by ClustalW

- Colours denote different chemical groups of amino acids, i.e. hydrophobic, acidic, etc. Symbols: "*" means identical character, ":" means conserved substitutions, "." means semi-conserved substitution, and blank means a non-conserved substitution: