Tools and Algorithms in Bioinformatics
GCBA815, Fall 2015

Week-4
BLAST Algorithm Continued
Multiple Sequence Alignment

Babu Guda, Ph.D.
Department of Genetics, Cell Biology & Anatomy
Bioinformatics and Systems Biology Core
University of Nebraska Medical Center

Basic Elements in Searching Biological Databases

- Sensitivity versus Specificity/selectivity
- Scoring Scheme, Gap penalties
- Distance/Substitution Matrices (PAM, BLOSSUM Series)
- Search Parameters (E-value, Bit score)
- Handling Data Quality Issues (Filtering, Clustering)
- Type of Algorithm (Smith-Waterman, Needleman-Wunsch)
Choice of the Searching Algorithm

An ideal algorithm should have
- Good specificity and sensitivity
- Should be fast running
- Should not use too much memory

Greedy algorithms are very sensitive, but very slow. Heuristic algorithms are relatively fast, but loose some sensitivity. It’s always a challenge for a programmer to develop algorithms that fulfill both of these requirements.

- Very greedy algorithm, so very sensitive
- Implements Dynamic programming
- Provides global alignment between the two sequences

- A set of heuristics were applied to the above algorithm to make it less greedy, so it is less sensitive but runs faster
- Implements Dynamic programming
- Provide local alignment between two sequences
- Both BLAST and FASTA use this algorithm with varying heuristics applied in each case
FASTA (FAST Algorithm)

- The first step is application of heuristics and the second step is using dynamic programming.
  - First, the query sequence and the database sequence are cut into defined length words and a word matching is performed in all-to-all combinations.
  - Word size is 2 for proteins and 6 for nucleic acids.
  - If the initial score is above a threshold, the second score is computed by joining fragments and using gaps of less than some maximum length.
  - If this second score is above some threshold, Smith-Waterman alignment is performed within the regions of high identities (known as high-scoring pairs).

Protein and nucleotide substitution matrices
BLAST (Basic Local Alignment Search Tool)

• The first step is application of heuristics and the second step is using dynamic programming.

• First, the query sequence and the database sequence are cut into defined length words and a word matching is performed in all combinations.

• Words that score above a threshold are used to extend the word list.

Expanded list - 47 words

<table>
<thead>
<tr>
<th>Word</th>
<th>Expanded List</th>
</tr>
</thead>
<tbody>
<tr>
<td>ql</td>
<td>ql, qm, hl</td>
</tr>
<tr>
<td>ln</td>
<td>ln</td>
</tr>
<tr>
<td>nf</td>
<td>nf, af, ny, df, qf, ef, gf, hf, kf, sf, tf</td>
</tr>
<tr>
<td>fs</td>
<td>fs, fa, fn, fd, fg, fp, ft, ys</td>
</tr>
<tr>
<td>gw</td>
<td>gw, aw, rw, nw, dw, qw, ew, hw, iw, kw, mw, pw, sw, tw, vw</td>
</tr>
</tbody>
</table>

BLAST continued ...

• BLAST is a local alignment algorithm.

• Several High Scoring Segments are found, with the maximum scoring segment used to define a band in the path graph.

• Smith-Waterman algorithm is performed on several possible segments to obtain optimal alignment.

• The word size for Protein is 3 and for Nucleic acid is 11.
Finding High Scoring Pairs (HSPs)

Comparison of BLAST and FASTA

- BLAST uses an expanded list to compensate for the loss of sensitivity from increased word size
- BLAST is more sensitive than FASTA for protein searches while FASTA is more sensitive than BLAST for nucleic acid searches
- Both BLAST and FASTA run faster than the original Needleman-Wauch algorithm at the cost of loss of sensitivity
- Both algorithms fail to find optimal alignments that fall outside of the defined band width
Essential Elements of an Alignment Algorithm

- Defining the problem (Global, semi-global, local alignment)
- Scoring scheme (Gap penalties)
- Distance Matrix (PAM, BLOSUM series)
- Scoring/Target function (How scores are calculated)
- Good programming language to test the algorithm

Types of Alignments

- **Global** - When two sequences are of approximately equal length. Here, the goal is to obtain maximum score by completely aligning them

- **Semi-global** - When one sequence matches towards one end of the other.
  - Ex. Searches for 5’ or 3’ regulatory sequences

- **Local** - When one sequence is a sub-string of the other or the goal is to get maximum local score
  - Protein motif searches in a database
Local vs Global alignments

Pair-wise sequence comparison
Different Alignment Paths

Sequence T

Whole genome sequence comparison

(A)  (B)

(C)  (D)
Scoring System for Alignments

**Scoring Weights**

- **Matches**: +10 - These are arbitrary values, but the real values come from distance matrices (PAM, BLOSUM etc.)
- **Mismatches**: +4

**Gap Penalties**

- **Gap Initiation**: -10 - Arbitrarily chosen but, optimized
- **Gap Extension**: -2 - for a particular Distance matrix

**Rules of Thumb for Affine Gap Penalties**

- Gap Initiation Penalty should be 2 to 3 times the largest negative score in a distance matrix table
- Gap Extension Penalty should be 0.3 to 0.1 times the Initiation Penalty

---

**Similarity Score**

- Similarity score is the sum of pair-wise scores for matches/mismatches and gap penalties

**Scoring Weights**

- **Matches**: +10
- **Mismatches**: +4
- **Gap Initiation**: -10
- **Gap Extension**: -2

Similarity Score = (Match score + Mismatch score) + Gap penalty

**Global Alignment**

EPSGFPAMVSTVHGQEIQI
E------PAMVST------QI
Score: (9 x 10) + (0 x 4) + (2 x -10 + 7 x -2)
Score = 90 + (-34) = 56
**Scoring/Target function**

• The scoring function calculates the similarity score. The goal of the algorithm is to maximize similarity score.

  **Global Alignment**
  
  EPSGFPANVSTVGQEQIQI  
  E----PANVST-----QI 
  
  Score: \((9 \times 10) + (2 \times -10 + 7 \times -2)\)  
  Score = 90 + (-34) = 56

  **Scoring Weights**
  
  - Matches +10
  - Mismatches +4
  - Gap Initiation -10
  - Gap Extension -2

  **Local Alignment**
  
  EPSGFPANVSTVGQEQIQI  
  E----PANVST-----QI 
  
  Score: \((6 \times 10) + (2 \times 0 + 10 \times 0)\)  
  Score = 60 + 0 = 60

  **Scoring Weights**
  
  - Matches +10
  - Mismatches +4
  - Gap Initiation -10
  - Gap Extension -2
  - No terminal gap penalty

**Different types of BLAST programs**

<table>
<thead>
<tr>
<th>Program</th>
<th>Query</th>
<th>Database</th>
<th>Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td>blastn</td>
<td>nucleotide</td>
<td>nucleotide</td>
<td>nucleotide</td>
</tr>
<tr>
<td>blastp</td>
<td>protein</td>
<td>protein</td>
<td>protein</td>
</tr>
<tr>
<td>blastx</td>
<td>nucleotide</td>
<td>protein</td>
<td>protein</td>
</tr>
<tr>
<td>tblastn</td>
<td>protein</td>
<td>nucleotide</td>
<td>protein</td>
</tr>
<tr>
<td>tblastx</td>
<td>nucleotide</td>
<td>nucleotide</td>
<td>protein</td>
</tr>
<tr>
<td>megablast</td>
<td>nucleotide</td>
<td>nucleotide</td>
<td>nucleotide</td>
</tr>
<tr>
<td>PHI-Blast</td>
<td>protein</td>
<td>protein</td>
<td>protein</td>
</tr>
<tr>
<td>PSI-Blast</td>
<td>Protein-Profile</td>
<td>protein</td>
<td>protein</td>
</tr>
<tr>
<td>RPS-Blast</td>
<td>protein</td>
<td>Profiles</td>
<td>protein</td>
</tr>
</tbody>
</table>
Multiple Sequence Alignment

Terminology

- **Homologous**: Similar
- **Paralogous**: Present in the same species, diverged after gene duplication.
- **Orthologous**: Present in different species, diverged after speciation.
- **Xenologous**: Genes acquired by horizontal gene transfer
- **Analogous**: Similarity observed by convergent evolution, but not by common evolutionary origin.
Evolution of Genomes and Genetic Variation

- Variation between species
  - Recombination, Cis-regulatory elements
  - Point mutations/insertions/deletions
  - Evolutionary selection \( \rightarrow \) speciation
- Variation within species:
  - Phenotype = Genotype + Environment
  - SNPs, haplotypes, aneuploidy, etc.
- Variation at cellular level:
  - Spatial state (Tissue/Subcellular location)
  - Temporal state (Stages in life cycle)
  - Physiological state (normal/disease)
  - External stimuli
Multiple Sequence Alignments

• Aligning homologous residues among a set of sequences, together in columns

• Multiple alignments exhibit structural and evolutionary information among closely related species i.e., families
  • A column of aligned residues in a conserved region is likely to occupy similar 3-D positions in protein structure

• Patterns or motifs common to a set of sequences may only be apparent from multiple alignments

• Necessary for building family profiles, phylogenetic trees and extracting evolutionary relationships

• Useful for homology modeling if at least one member of the family has a known 3-D structure
Multiple alignments from different VGC protein families

Sodium VGC

Potassium VGC

Calcium VGC

Voltage-gated ion channel proteins

Fig. 1. Conserved motifs in the voltage-sensing module of calcium, sodium, and potassium ion channel proteins.

Fig. 2. Conserved motifs in the pore-forming module of calcium, sodium, and potassium ion channel proteins.
Scoring a Multiple Alignment

- Some positions are more conserved than others
- Sequences are not independent, but are related by a phylogeny

$$S = \sum_i S(m_i) + G$$

where, $S(m_i)$ is the score for column $i$, $G$ is gap penalty
Basic progressive alignment procedure

• Calculate alignment score for all pair-wise combinations using Dynamic Programming

• Determine distances from scores for all pairs of sequences and build a distance matrix

• Use a distance-based method to construct a ‘guide-tree’

• Add sequences to the growing alignment using the order given by the tree

• Most of the multiple alignment methods differ in the way the guide tree is constructed
Feng-Doolittle Method

Get a distance matrix

Calculate pair-wise scores with DP method, convert raw scores into pair-wise distances using the following formula

\[ D = -\ln S_{eff} \]

\[ S_{eff} = \frac{S_{obs} - S_{rand}}{S_{max} - S_{rand}} \]

where,

- \( S_{eff} \) is the effective score
- \( S_{obs} \) is the similarity score (DP score) between a pair
- \( S_{max} \) is the max score, average of aligning either sequence to itself
- \( S_{rand} \) is the background noise, obtained by aligning two random sequences of equal length and composition

From pair-wise distances, build a matrix for all sequence combinations as follows

<table>
<thead>
<tr>
<th></th>
<th>Human</th>
<th>Chimp</th>
<th>Gorilla</th>
<th>Orang</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human</td>
<td>0</td>
<td>88</td>
<td>103</td>
<td>160</td>
</tr>
<tr>
<td>Chimp</td>
<td>0</td>
<td>106</td>
<td>170</td>
<td></td>
</tr>
<tr>
<td>Gorilla</td>
<td>0</td>
<td>166</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Orang</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- The number of pair-wise distances are \( N(N-1)/2 \)
Alignment of sequences

- The alignment order is determined from the order sequences were added to the guide tree
- First 2 sequences from the node are added first. In this case, C-H are aligned according to the standard DP algorithm
- Next, G is aligned to CH as the best of G(CH) and (CH)G alignments
- Assuming that of the above two, G(CH) has the best score, O is aligned to G(CH) as the best of O(G(CH)) and G(O(CH))
- Again, higher similarity score determines which one is the best. This process is repeated iteratively until all sequences are aligned.
- As you notice, the order of sequences in the output are not the same as you see in the guide tree.

Multiple sequence alignment (MSA)

- MSA: tools– Clustal, Mafft, Muscle, TCoffee, Mview
- Accessible from Ebi website: [http://www.ebi.ac.uk/Tools/msa/](http://www.ebi.ac.uk/Tools/msa/)
- Obtains a distance matrix by using scores from pair-wise DP method
- Guide tree is built using ‘neighbor joining’ method
- Sequences are weighed to compensate for biased representation in larger families
- Substitution matrices change on the fly as the alignment progresses; closely related sequences are aligned with ‘hard’ matrices (e.g. BLOSUM80) and distant sequences are aligned with ‘soft’ matrices (e.g. BLOSUM40)
- Position specific gap penalties are used similar to profiles
- Guide tree may be adjusted on the fly to defer the alignment of low-scoring sequences