Algorithm Design and Analysis Project

PREPARED BY

Niranjan Joshi : 160001026

Kanishkar J : 160001028

CS 254: Design and Analysis of Algorithms Lab

Introduction

A sequence alignment is a way of arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. Aligned sequences of nucleotide or amino acid residues are typically represented as rows within a matrix. Gaps are inserted between the residues so that identical or similar characters are aligned in successive columns.

If two sequences in an alignment share a common ancestor, mismatches can be interpreted as point mutations and gaps as indels (that is, insertion or deletion mutations) introduced in one or both lineages in the time since they diverged from one another.

Need for Sequence Alignment

In recent years, genome projects conducted on a variety of organisms generated massive amounts of sequence data for genes and proteins, which requires computational analysis. Sequence alignment shows the relations between genes or between proteins, leading to a better understanding of their homology and functionality. Sequence alignment can also reveal conserved domains and motifs.

There are two approaches to this problem :

1. Dynamic Programming: Smith-Waterman Algorithm for alignment of two sequence Alignment.
2. Heuristic approach: Progressive Alignment construction for Multiple Sequence Alignment.

Smith-Waterman Algorithm

Smith-Waterman Algorithm is used for local alignment of two sequences. It gives the best possible alignment of two sequences based on substitution matrix and gap-scoring scheme. Dynamic Programming is used to solve the recurrence relation in bottom up approach. This is done by filling the scoring matrix.

The DNA sequence contains bases denoted by A,T,G,C. Substitution matrix gives the score for matches and mismatches. Matches have positive score while mismatches have negative scores. Gap scoring scheme gives the penalty W(k) for gap of length k. The scoring matrix is filled by the algorithm based on above two criteria. Following is an example of sequence alignment:

```
TACGGGCCCGCTA
| |       | | |         | | |
TA___GCC___CTA
```

Progressive Alignment construction

The dynamic approach to multiple sequence alignment has been proven to be a NP-complete problem. For n individual sequences, the naive method requires constructing the n-dimensional equivalent of the matrix formed in standard pairwise sequence alignment. The search space thus increases exponentially with increasing n and is also strongly dependent on sequence length.

Hence, we go for a heuristic approach. In this approach we compromise on accuracy for performance. The most popular heuristic approach to this class of problems is known as known as progressive technique. Progressive alignment builds up a final Multi sequence alignment by combining pairwise alignments beginning with the most similar pair and progressing to the most distantly related. All progressive alignment methods require two stages: a first stage in which the relationships between the sequences are represented as a tree, called a guide tree, and a second step in which the MSA is built by adding the sequences sequentially to the growing MSA according to the guide tree.

References

https://en.wikipedia.org/wiki/Multiple_sequence_alignment

https://courses.cs.washington.edu/courses/cse427/10wi/progressive.html

http://www.springer.com/cda/content/document/cda_downloaddocument/9780387713366-c2.pdf?SGWID=0-0-45-387705-p173729311

https://en.wikipedia.org/wiki/Sequence_alignment

https://en.wikipedia.org/wiki/Smith%E2%80%93Waterman_algorithm