Sequence alignment

Sergio Peignier

23 novembre 2015

A sequence is defined as an ordered list of elements belonging to a given alphabet (e.g., the DNA sequence *attcgccggat* defined by the alphabet $\{a, t, g, c\}$). Sequence alignment consist in arranging two (or more) sequences to identify regions of similarity. In order to compare two sequences we compare in pairs letters from both sequences. Letters may match, mismatch, be deleted or inserted :

1. Match : The two letters are the same
2. Mismatch : The two letters are differential
3. Indel (INsertion or DELetion) : One letter aligns to a gap in the other sequence.

1 Needleman-Wunsch algorithm

Needleman et Wunsch algorithm performs global sequence alignement, i.e., it finds the best alignment over the entire length of two sequences s and s'. Intuitively the algorithm seeks an alignment that maximizes the number of element-to-element matches. Before describing the algorithm we need to choose a scoring system, i.e., the score associated to matches, mismatches and indels. The scoring system defined by Needleman and Wunsch is very simple, the respective scores are : $S(x, x') = 1$ for matches (if $x = x'$), $S(x, x') = -1$ for mismatches (if $x \neq n'$) and $d = -1$ for indels.

The algorithm has tree major steps :

— Initialize the comparison matrix$F$ with the following with procedure :

		s'_1	s'_2	...	s'_m
	0	-1	-2		-m
s_1	-1				
s_2	-2				
...					
s_n	-n				

— Fill in the Matrix : Use the following equation to fill the matrix :

$$F_{ij} = max(F_{i-1,j-1} + S(s_i, s'_j), F_{i,j-1} + d, F_{i-1,j} + d)$$

Try to understand the logic of this equation.
For each element you can record the decision made ($F_{i-1,j-1}$, $F_{i,j-1}$ or $F_{i-1,j}$).

— Find the best alignement : Finally we start from the element $F_{n,m}$ of the matrix and we move backward until we reach the element $F_{0,0}$. If we consider the element $F_{i-1,j}$, we should check wich decision ($F_{i-1,j}$, $F_{i,j-1}$ or $F_{i-1,j-1}$) leaded to the actual solution (many solutions are sometimes possible, in this case you can keep all of them). If the decision was $F_{i-1,j}$ then it means that s_i is aligned with a gap, if it was $F_{i,j-1}$ then s'_j is aligned with a gap and if it was $F_{i-1,j-1}$ then s_i and s'_j are aligned. You can use the records you made in the previous step or recompute the possibilities ... both are possible.

2 Smith-Waterman algorithm

Smith-Waterman algorithm performs local sequence alignement, i.e., it finds alignements shorter than the entire sequences. Look to this algorithm principles on the internet and then modify the previous algorithm to perform local alignment.

3 Hierarchical clustering

Let us define the distance between two sequences as the number of mismatches and indels in the global alignment of both sequences. Using this definition it is possible to clusterize sequences. We will use a bottom-up hirarchical clustering approach. Initially each object (each sequence in this case) is a class. Then the number of classes is iterativelly reduced by merging clusters. At each iteration we will merge the two closest elements. There are several ways to compute the distance between clusters containing more than one point :

— We can consider the maximum distance between elements of each cluster (complete-linkage) :

$$\max\{\, d(x,y) : x \in \mathcal{C}_1,\, y \in \mathcal{C}_2 \,\}$$

.

— The minimum distance between elements of each cluster (single-linkage clustering) :

$$\min\{\, d(x,y) : x \in \mathcal{C}_1,\, y \in \mathcal{C}_2 \,\}$$

— The mean distance between elements of each cluster (average linkage clustering) :

$$\frac{1}{|\mathcal{C}_1| \cdot |\mathcal{C}_2|} \sum_{x \in \mathcal{C}_1} \sum_{y \in \mathcal{C}_2} d(x,y)$$

It could be usefull to compute an initial pairwise distance matrix and then merge the columns and the rows of this matrix while we merge clusters. Download some micro-RNA sequences from http ://www.mirbase.org/ftp.shtml, implement the hierarchical clustering algorithm and use your Needleman-Wunsch algorithm to clusterize sequences.