

# BCB BCB/GDCB/STAT/COM S 568 Spring 2010

## Homework 1

January 19, 2010

**Due one week later. Answers to selected problems will be posted.**

- 1) A global alignment of two sequences  $\mathbf{A}=a_1 a_2 \dots a_M$  and  $\mathbf{B}=b_1 b_2 \dots b_N$  can be represented by the set of index pairs  $P = \{(i_1, j_1), (i_2, j_2), \dots, (i_k, j_k)\}$ ,  $1 \leq i_1 < i_2 < \dots < i_k \leq M$ ,  $1 \leq j_1 < j_2 < \dots < j_k \leq N$ , where the index pairs  $(i_x, j_x)$  indicate that  $a_{i_x}$  is aligned with  $b_{j_x}$ .
  - a. The Needleman-Wunsch algorithm imposes the restriction  $i_x - i_{x-1} = 1$  and/or  $j_x - j_{x-1} = 1$  for  $x=1, 2, \dots, k+1$  where  $i_0 = j_0 = 0$ ,  $i_{k+1} = M+1$ , and  $j_{k+1} = N+1$  (avoidance of “double gaps”). Prove that the optimal score of a global alignment with end-gap penalties can be calculated as  $S_{MN}$ , where  $S_{ij}$  is derived recursively at each step as the maximum of  $S_{i-1, j-1} + \sigma(a_i, b_j)$ ,  $S_{i-1, j-1-p} + \sigma(a_i, b_{j-p}) + w(p)$ ,  $p=1, 2, \dots, j-1$ , and  $S_{i-1-q, j-1} + \sigma(a_{i-q}, b_j) + w(q)$ ,  $q=1, 2, \dots, i-1$ , provided one specifies correct initial values of  $S_{00}$ ,  $S_{0j}$ ,  $j=1, 2, \dots, N$ , and  $S_{i0}$ ,  $i=1, 2, \dots, M$  (here  $\sigma(a_i, b_i)$  is the score for aligning  $a_i, b_j$ , and  $w(x)$  is the gap penalty for a gap of size  $x$ ).
    - i. Indicate to what values  $S_{00}$ ,  $S_{0j}$ , and  $S_{i0}$  should be set for the recursion to work.
    - ii. How would you change the algorithm to calculate the optimal score for a global alignment without end-gap penalties?
    - iii. Give an algorithm to derive the number of all possible alignments for sequences of lengths  $M$  and  $N$ .
  - b. Derive the algorithm to calculate the optimal score as in (a) but without the restriction of avoidance of double gaps.
    - i. Determine the complexity of the algorithm: how many operations are required to calculate the optimal score?
- 2) In class we showed that for an alignment of length  $N$  with per site match probability  $p$ , the length  $k$  of the longest common word is approximately  $(\ln N - \ln \ln 1/\alpha) / \ln (1/p)$ , where  $\alpha$  is the probability of no occurrence of a  $k$ -word.
  - a. Re-derive the approximation.
  - b. Set  $p=0.25$  and plot  $k$  as a function of  $\alpha$  for various choices of  $N$ . Interpret your graphs.