

Homework Assignment 2, Due Monday Sept 22, 2008

CSCI 6490/4490 Algorithms for Computational Biology

September 12, 2008

1. Implement the branch-and-bound algorithm `BRANCHANDBOUNDMOTIFSEARCH` to solve the motif finding problem. This algorithm is given on page 111 and it uses two additional algorithms: one is `NEXTVERTEX` on page 107 and another `BYPASS` on page 108. The implementation should follow exactly the same idea used in the algorithm. You may choose to use one of the following languages: Java, C, C++, or Python.

Other requirements:

- You also need to implement a main program to take input data, to call function `BRANCHANDBOUNDMOTIFSEARCH`, and to output results.
- Input: (1) DNA containing t sequences, each with length n , and (2) integer l , the length of motifs. You may assume $t \leq 50$, $n \leq 100$, and $l \leq 20$ to be used in testing you code. But your program should allow larger ranges of these parameters for the program to be useful.
- Output: (1) $S = \{s_1, s_2, \dots, s_t\}$, the beginning positions of found motifs from DNA, and (2) a motif profile P constructed from S .

Note that you may not want to pass the DNA as a parameter to functions since it could be large.

2. Implement the Gibbs sampling algorithm `GIBBSAMPLINGMOTIFFINDING` for the motif finding problem. The idea of the algorithm is on page 413 and algorithm is in lecture note 2. The input and output requirements for this program are the same as for the program in 1.

In addition, you need to design and implement the condition for convergence. You cannot simply use a counter to stop the program arbitrarily.

Since we will test to see if your program works well on DNA with biased nucleotide frequencies, you may want to find a way (use an existing program online, download one from the web, or code one by yourself) to generate such biased sequence samples. This sample generation code is not a part of the implementation assignment.

Programs are submitted to the teaching assistant (TA) to grade. Programs are developed on your own laptop and you need to demo (compile and execute) your programs in the TA's office. No late homework answers will be accepted (with exception in health or family emergency related cases).