

# 6.095/6.895 Fall 2005 - Problem Set 3

Due: October 24, 2005 at 8pm

Lecture readings: J4.4-4.9, J5.5, J12.2, J8.1-8.2, P1 and D7.1-7.5.

1. For this problem you will implement a simple Gibbs Sampling Routine. You should fill in the python template file `gibbs.py`. The Gibbs Sampling Algorithm can be summarized as follows:

```
1 GibbsSampler( $S_1, S_2, \dots, S_t, L$ ):
2   Inputs:
3      $S_1, S_2, \dots, S_t$ : strings that contain a common pattern
4      $L$ : the length of the probabilistic pattern to find
5
6   Choose a substrings of length  $L$  from each of  $S_1, S_2, \dots, S_t$ 
7
8   Repeat until convergence:
9     At random, choose one sequences  $S_c$ 
10
11    Create a pattern using all substrings except the one from  $S_c$ 
12
13    Score all substrings of length  $L$  in  $S_c$  using the pattern
14
15    Randomly choose a substring from  $S_c$  with probability proportional
16      to its score
17
18    Return the last computed pattern
```

More details of this algorithm (such as how to score a sequence) are found in lecture and J412. If you make any modifications to the algorithm or fill in ambiguities (there are several variations to Gibbs Sampling), state what they are and why you made them.

You should run your code on the four provided data sets. `data1` is an artificial data set with planted identical motifs, `data2` is an artificial data set with degenerate planted motifs, and `data3` and `data4` are real promoter sequences from yeast known to bind ACE2 and MBP1, respectively. You can use your list of known motifs from problem set 1 to verify your found motifs for `data3` and `data4`. The GC content of `data1` and `data2` is approximately 0.5 and the GC content of `data3` and `data4` is approximately 0.37. The pattern you find in `data1` should be very strong.

You should turn in a printout of the code you added in addition to what patterns of length 10 your program found consistently for each the four provided data sets. Repeat the procedure several times for each data set until you believe you have found the strongest patterns.

**Extra credit.** Download AlignACE, MEME and/or BioProspector, and compare your results to their performance. Describe and interpret your findings.

2. The expression of one gene may have a regulatory effect on another gene. In this problem we will represent gene regulation as a graph. Represent all the genes as vertices in a graph and have a directed edge from gene  $A$  to gene  $B$  if gene  $A$  directly regulates gene  $B$  (that is binds to its promoter region or inhibits its translation.
  - (a) From the graph described above, how can we detect if a gene directly regulates itself? This type of regulation is known as autoregulation.
  - (b) How can we detect if a gene indirectly regulates itself? This type of regulation is known as a multi-component loop.
  - (c) How can we augment this graph to represent the extent to which one gene directly regulates another? How do we represent a gene negatively regulating another one?
  - (d) What are some other features that we should represent with our graph to make it more accurately model gene regulation?

- (e) **Extra Credit.** Sketch an algorithm to discover the most abundant patterns of connectivity in a graph. These patterns are known as network motifs.

3. (a) Consider the following DNA sequences:

Position	1	2	3	4	5	6
Sequence 1	A	A	C	C	G	G
Sequence 2	A	C	T	C	A	G
Sequence 3	G	T	C	C	T	T
Sequence 4	G	G	T	T	C	G

Consider the three possible unrooted trees on four elements. For the sequences above, give the cost of each position for each tree. Also give the total cost of each tree and indicate the lowest cost tree. Assume the cost of a transition ( $A \leftrightarrow G, C \leftrightarrow T$ ) is 1, the cost of a transversion (any other mismatch) is 2 and there is no cost for a match.

- (b) Perform UPGMA clustering on sequences with distances:

	a	b	c	d	e
a	0	3	11	10	12
b	3	0	12	11	13
c	11	12	0	9	11
d	10	11	9	0	8
e	12	13	11	8	0

Include all intermediate trees and distance matrices resulting from the application of UPGMA as shown in class. Notice that the tree does not perfectly recreate the distances. How can you modify the tree, allowing for imbalanced lengths, so that it perfectly matches the distance metric? What algorithm seen in class would be able to do this? What feature of this algorithm would allow for this?

4. **6.895 Problem.** Many standard graph theoretic algorithms have applications to computational biology. Take some graph theory algorithm and indicate how it may be applied to biological problems. You should not use an application that was already discussed in class. Implementing and testing this application may make for a good semester project.