VCU Bioinformatics and Bioengineering Summer Institute

PSSMs to search for repeated sequences - Problem Set

Questions relating to **Repeated sequences**

A1. You run across the following intergenic region between the convergent genes *alr0787* and *purN* (highlighted in yellow), taken from the genome of *Anabaena*/*Nostoc* PCC 7120. Two sets of repeated sequences pop out at you (the first set is shown in red/magenta and underlined and the second in blue/green and double underlined). Are the two sets mostly complementary to each other? Could they form a stem/loop structure?

```
                     alr0787 → |
TTGTTTGTGGAAACTGCGCCAGTCGTAGAACACAACACTTCC
GTTAAATTAGCTAGGACTTACGCATACAGGTTGTCTGTTGAG
ACCGGGTGTAAGGATGTAAGGGTGTAAGGGTTTCAAGCATTT
ATACCCCTACACCCCCATACCCCTATACCCTTCTCCAAACCC
TTGATCTTTCGTTTTTATGCGTAAGTCCTATTAGTTTGAGTC
              | ← purN
```

A2. You run across a region of a chromosome that appears to be a long tandem repeat. A stretch of 200 nucleotides is immediately repeated by the same 200 nucleotides… sort of. Actually, there are a number of defects in the repeat. Well, actually, there are a ***lot*** of defects, 100 to be exact. Maybe you're fooling yourself. What's the probability that two adjacent sequences 200 bases long would be 50% identical? Presume that all nucleotides are equally likely.

Hint #1: Recall from high school math the following:

Probability = (number of ways a pattern can occur) * (probability of one specific pattern)

In this case, that works out to:

$$\text{Probability} = ({}_{200}C_{100}) \quad * \ (1/4)^{\text{matches}} \cdot (3/4)^{\text{mismatches}}$$

$$= \quad 200! \,/\, [100! \cdot (200\text{-}100)!] \quad * \ (1/4)^{100} \cdot (3/4)^{100}$$

Hint #2: If your calculator isn't up to the task, try writing a program to do the job.

Questions relating to **Construction of PSSMs**

B1. Use Excel to construct a PSSM for the NtcA sequence data given in Table 3 of Monday's notes).

B2. PSSMs can be used to find motifs in protein just as well as motifs in DNA. You make this realization while lolling on the outer banks (which might not be a bad idea sometime this summer, especially for those of you generally confined to the middle of the continent). True, there's a lot more amino acids than nucleotides,… will that complicate the analysis? You wonder, how much more uncertainty there is in protein alignments than in DNA alignments. You recall that $\mathbf{H_{max}}$ (the maximum uncertainty for a given position in an alignment) for DNA is 2. What is $\mathbf{H_{max}}$ for protein? Far away from any electronic aid, you'll need to estimate an answer, which can be in the form $x < \mathbf{H_{max}} < y$)

Questions relating to **Arrays**

C1. Write a program that will calculate and store the square roots of numbers 1 to 100. To do this:

a. Define an array called square_root
b. Write a loop that will go through the numbers 1 to 100
c. Within the loop, calculate the square root of the number under consideration
d. Within the loop, assign that number to the appropriate element of the array

C2. Complete the following program:

```
! Prints input sentences backwards
! Not the most direct way, I admit
! Strategy:
!   Read in a sentence from the keyboard
!   Split the sentence up into letters, stored in the array letters$
!   Print the array backwards
! Input:
!   Sentences from keyboard
!   Program asks for more until user inputs a bare period

[Put statement here that defines the array called letters$, with an extent of 1000]

DO
   INPUT PROMPT “Type in a sentence or just a period to quit: “: sentence$
   IF sentence$ = “.” THEN EXIT DO
   LET length_of_sentence = Len(sentence$)

   FOR position = 1 TO Len(sentence$)
      LET letter$ = sentence$[position:position]
  [Assign letter$ to an element in the array letters$]
   NEXT position

   FOR [fill in this part] STEP -1
      LET letter$ = [fill in this part]
      PRINT letter$;
   NEXT [fill in this part]

   PRINT
LOOP

END
```

Questions relating to **Implementation of PSSMs**

D1. Produce a program that prints a table of nucleotide frequencies, given a file of aligned sequences. Note that I’m not asking you to write anything from scratch, since PSSM.tru already does what you want (except print out the table). So write the program by ***subtraction***, throwing away from PSSM.tru all that is not necessary for the desired task.

Here’s a subroutine (to be placed ***after*** the END statement) to print two-dimensional arrays:

```
SUB Print_table_integers(array(,)) ! The internal parentheses warns the compiler
                                   !   what kind of array to expect
! Prints a two-dimensional array
! Presumes the first dimension gives rows and the second dimension gives columns
! Presumes array holds positive integers no bigger than 999
! Presumes nothing about the extent of the arrays:
!     Lbound(array,n) gives the lower bound of the nth dimension
!     Ubound(array,n) gives the upper bound of the nth dimension

   FOR j = Lbound(array,2) TO Ubound(array,2)
      FOR i = Lbound(array,1) TO Ubound(array,1)
         PRINT Using$(“####”, array(i,j));
      NEXT I
      PRINT
   NEXT j
END SUB
```

D2. Find all sequences in the *E. coli* genome (see file on web) that is close to the CRP binding site given by Li et al (2002) (see Table 2). The symbol “W” stands for “A” or “T” (“W” is taken from “weak”, because A and T pairs by only two hydrogen bonds).

D3. You want to find all *Nostoc* ***proteins*** that may be response regulators (a type of protein that often regulates the transcription of genes). Revise the logic of the Main Program of PSSM.tru so that it outlined a procedure to do what you want.

D4. (Not for the faint of heart) Alter PSSM.tru so that it will find all *Nostoc* proteins that may be response regulators, using the file `response_regulators.txt` on the web.

Questions relating to **Applications of PSSMs**

E1. Cyclic AMP (cAMP) is a small molecule used as an internal signal to represent a variety of states. For example, the concentration changes in the liver in response to sugar levels and determines whether glycogen is broken down to release stores of glucose. You’re interested in understanding how cAMP is sensed by protein, i.e. the structural component of protein responsible for its binding to cAMP. To this end, you’ve collected the amino acid sequences of a variety of protein related to cAMP (each protein is preceded by its GenBank accession number):

P07278	Yeast, cAMP-dependent protein kinase regulatory chain
P03020	E. coli, cAMP receptor protein (CRP)
Q64359	rat, Cyclic-nucleotide-gated olfactory channel OCNC2 subunit
P29747	fly, Cyclic-AMP response element binding protein A
P18847	human, Cyclic-AMP-dependent transcription factor ATF-3
P34122	Dictyostelium, Cyclic-AMP-Binding protein CABP1
P27925	cow, cAMP-response element protein 2 CREB2
Q9NP56	human, cAMP-specific phosphodiesterase
Q04758	mouse, cAMP-dependent protein kinase inhibitor

The amino acid sequences for these proteins can be found online on Thursday's web site.

E1a. Run *Meme*, a pattern-finding program (link available on the web site), asking it to find any significant sequence motifs within this collection of protein.

E1b. Rerun *Meme*. Demand that *Meme* require that any motif identified by found in <u>each</u> protein in the list

E1c. Run Pfam or BlastP (from web) over one of the sequences to check how the motifs you found match the opinion of the world at large.

E1d. Consider: Did you get what you expected? How many of the nine proteins have motifs found by Meme? Do any of the motifs found correspond to cAMP-binding motifs? Rationalize your results.