Week 8 Homework BCH/BIO519

Files can be found at mellencamp at /ifs/courses/bch519/spring13/assignments/Week8/

Homework is due by midnight on Mon. 3/31. As always, make sure that your assignment reflects your own individual work.

Exercise 1: Create a simple ORF finder

For the first exercise, you'll write a program to find open reading frames in a DNA sequence. Your program should take as input the provided sequences ("sequence_A" and "sequence_B") and supply as output:

(1) The sizes of the potential ORFs greater than 15 amino acids from all 3 forward reading frames (you're welcome to do all 6 reading frames, but not required. It's pretty easy though, Hint: to reverse a string *dna* in Python, write *dna[::-1]*).
(2) The translations into protein of the ORFs.
(3) Note: As defined for this exercise, an ORF does not have to begin with an ATG, but should be any sequence of nucleotides that encodes a polypeptide of >15 amino acids.
(4) Output a peptide each line with this format:
frame #: length_of_peptide sequence_of_peptide

To save you some effort in looking up and entering the genetic code, I've supplied a code snippet to set up a Python dictionary of codons in the file "codondictionary.py" which you can copy into your program (open *URL: https://gist.github.com/taoliu/9799831*).

You should write your program in Python and name it "*your_last_name*.ORFfinder.py". Choose sensible names for your variables and comment your code extensively so I will be able to follow what you've done. Include your full name and email address in a comment line at the top of the script.

When you're ready to submit your assignment, do the following. *Make sure you follow these instructions or your assignment may not be graded!*
Use the command --submit hw=8. Submit each of your program files. In addition, like you did previously, create a terminal session *script*. Name this session *last_name.HWwk8.ex1.txt*.

1. At the command line, enter "script *your_last_name*.wk8.ex1.txt". Enter "cat *name_of_your_program*", e.g. "*your_last_name*.ORFfinder.py".
2. Run your program
3. If your output was saved to a file, *cat* the output file.
4. exit

Hints: There are a lot of different ways to do this. Some will be more efficient than others.

• Do not assume that there are no header lines or newlines in the sequence

• You can loop through 3 times, one for each reading frame. But a quicker way to do it would be to go through once, and assign each reading frame to its own string or array.

• This might be a good place to check out the modulus arithmetic operator "%"!

• Be careful about the end of the sequence! Remember, the last codon begins 3 bases from the end.

• Remember that there might be more than one ORF in each reading frame.

Exercise 2: Look at the protein sequences you obtained from Exercise 1. Predict which are real protein sequences; record your guesses. Check *all* of your results using BLASTP. How did you do with your predictions? Are your matches significant (real) matches? Assuming that you find a real protein sequence, provide the name of the protein encoded by your DNA sequence and the species to which it belongs (for both sequences A and B). Now check the nucleotide sequences using BLASTN. How do the results compare to your protein results?

Exercise 3: The following sequences are all confirmed binding sites for the *Drosophila* transcription factor Tinman.

```
NCACTTGAN
CCACTTGAG
CCACTTGAG
ACAATTAAA
TCACTTCAC
GCACTTAAG
CCACTTGGA
GCACTTGAG
CCACTTAGG
GCCCTTGAG
CCACTTGAG
GAACTTGAC
GCACTTGAA
CCACTTGAN
CCACTTGAN
```

a) Derive a consensus sequence for Tin binding. Use the rules of thumb provided in the D'haeseleer paper (Assigned Reading paper) to decide when to use a single base or a degenerate symbol:
"Conventionally, a single base is shown if it occurs in more than half the sites and at least twice as often as the second most frequent base. Otherwise, a double degenerate symbol is used if two bases occur in more than 75% of the sites, or a triple degenerate symbol when one base does not occur at all."

IUPAC nucleotide code

A	Adenine
C	Cytosine
G	Guanine
T (or U)	Thymine (or Uracil)
R	A or G
Y	C or T
S	G or C
W	A or T
K	G or T
M	A or C
B	C or G or T
D	A or G or T
H	A or C or T
V	A or C or G
N	any base
. or -	gap

b) Make a logo for this motif using the tools at http://weblogo.berkeley.edu//logo.cgi (default settings are fine). Make sure you include a picture of the logo when you submit the assignment.
c) How does the logo compare with your consensus site?
d) Tinman is orthologous to the mouse transcription factor Nkx2.5, listed in JASPAR as Nkx2-5. Find the entry for Nkx2-5 in JASPAR (http://jaspar.genereg.net). How do the two motifs compare (what's similar/different)?

Answers to Exercises 2 and 3 can be submitted as a simple text file along with the programming assignment, or e-mailed to me <tliu4@buffalo.edu> as an MS-Word document (this might be easier). If using the latter option, please make sure the subject line of the email reads "BCH519 HW Wk8".