FINAL PROJECT – COMP 113, 266 / BIOL 265, 266 SPRING 2022
(due date: Tuesday 10 May, 9 AM)

OVERVIEW

The last assignment of the year is to do a project that has a significant programming component. This does not mean that the program is necessarily long but that it uses many of the skills you have learned over the term. It might involve writing a program from scratch or it may be that you find code on the web that you can build upon or improve. I expect that you will hand in two documents. The first will be a .py or .ipynb file that if I run in IDLE or Jupyter Notebook will give me an idea of what your project is capable of. The second document would be a write up that describes your project in words. Depending upon the project, it may include background on the project including any special terms you use or theories its based upon, a short "user guide" to give me an idea of how to use your program as well as any bugs that may still exist, a description of the general design of your program along with any issues you encountered while developing it, and possible plans for extensions or fixes. I don't expect this second document to be more than about 5 pages in length. Again, depending upon the project you may need to submit other files such as input data files. If there are any special instructions as to how to handle these files (or concerning any other aspect of using your code) please include these in your write up as well as in comments in your Python script.

Below is a list of potential topics for your final project. The topics involve the analysis of biological data and will require you do some background reading on the topic. Some potential resources for these topics are also provided below. Each of the topics requires that you write one or more Python scripts in order to complete the task. Most of the topics are somewhat open-ended in that you can always find ways to enhance your program or find more data to analyze.

You are encouraged to work in groups of two or three for the final project. However, each member of the group is required to submit an *individually written* final report and Python code. If different group members did different parts of the project, please be sure to describe your roles.

POTENTIAL RESOURCES

(A) Wesleyan Microarray Database
(B) Wesleyan RNA Database
(C) Wesleyan Drosophila Splice Site Database
(D) Wesleyan Drosophila cDNA Database
(E) Wesleyan Yeast Database
(F) Wesleyan Information Theoretic Analysis Tool

All these data resources can be accessed from https://numana.wesleyan.edu/~mweir/. Resource A is the database that we use in BIOL 265. It has a web interface that can be used to select and download subsets of various microarray data sets for analysis with desktop software tools such as Genesis. Resources B, C, D and E have web interfaces that can be used to execute data analysis tools (stored procedures) in the respective databases. The databases can also be accessed using Azure Data Studio.

To access Resources D and E, you can click on the link for WesQL and then click the Login tab. Select the IGS-server and login with the same username and password you use in the academic computer labs. Click the Stored Procedures tab and then select the devCDNA_350 database or devYeast_350.

The Wesleyan Information Theoretic Analysis Tool (Resource F) allows you to carry out information-theoretic analysis of sets of aligned sequences.

PROJECT TOPICS

1. Translation Consensus Sequences (Resources D, E and F)

Write Python code to calculate information content from a set of aligned sequences. You may find it useful to refer to https://wesmoodle.wesleyan.edu/pluginfile.php/412381/mod_resource/content/6/info_theory.htm. You will need to include pseudocounts so that the code does not try to calculate log 0. Test your code by using Resource F above (e.g. using some of the sequence sets discussed below).

For the remainder of the project, you may use your Python code and/or Resource F. Please choose a subset of options a-e below. (Options e and f may be particularly interesting.)

a. Use information theory to analyze the consensus sequences surrounding ATGs in *Drosophila* cDNA transcripts or Yeast annotated genes. First, compute the information and nucleotide frequencies for selected sets of the annotated translation start sites (e.g. using all transcripts, transcripts with short 5' UTRs, or transcripts with no ATGs upstream of the start site). Sample sets of sequences (and corresponding SQL code) are available at https://wesmoodle.wesleyan.edu/mod/resource/view.php?id=302129. Use your results to predict a consensus sequence for positions -6**..**-1 and 4**..**6 (where ATG is found at positions 1, 2, 3, and there is no 0 coordinate). Test your consensus sequences by designing and computing a consensus score for each cDNA transcript and summarizing the results in a histogram showing the distribution of scores. Using resource F, compare your consensus scores with individual information scores.

b. Compare the information and nucleotide frequencies at the annotated translation start sites with the same measurements at upstream ATGs using selected sets of transcripts. Do your results suggest that the start sites are, generally speaking, annotated correctly? You might investigate information at annotated start sites for groups of mRNA with increasing numbers of upstream ATGs.

c. The authors Kozak (for many organisms) and Cavener (for Drosophila) have examined the frequencies of different bases at positions near the known start (ATG) codons of large numbers of proteins. In particular, Cavener collected the sequences that occur in the 10 positions upstream of the AUG initiating translation for many fly genes. (A partial listing is found at https://wesmoodle.wesleyan.edu/mod/resource/view.php?id=302131) Compute the information and nucleotide frequencies at positions -6**..**-1 using the (partial) data set. How do these results compare with the above analyses? In general, compare your results with those found in Cavener (1987 - Nucl. Acid Res. 15:1353-1361; see http://www.ncbi.nlm.nih.gov/pmc/articles/PMC340553/.)

d. Some annotated translation start codons are preceded by ATGs in their 5'UTR. In some cases, these upstream ATGs initiate a theoretical ORF that overlaps the annotated translation start codon (annAUG). Investigate the information content of annAUGs with overlapping upstream ORFs.

e. Compare the frequencies of G in yeast ORF sequences with the frequencies of GNN codons. Are GNN codons found at higher-than-expected frequencies? Is this also observed in ORFs in other species such as Drosophila?

f. Protein Open Reading Frames (ORFs) have a 3-nucleotide periodicity characterized by a tendency towards GNN codons. Investigate this periodicity in the vicinity of translation start sites. Compare the strength of this periodicity in sets of genes with high and low protein expression (e.g. top 10% cf bottom 10%). Investigate whether the periodicity is stronger near the beginning of ORFs.

g. Investigate the 3-nucleotide periodicity discussed in (e) in groups of genes up- or down-regulated under stress conditions. Refer to ribosome profile resource from Gerashchenko and Gladyshev 2014 https://wesmoodle.wesleyan.edu/mod/resource/view.php?id=622389

2. Splicing Consensus Sequences (Resources A, C)

Write Python code to calculate relative individual information of a sequence based on a weight matrix derived from a reference set of aligned sequences. You may find it useful to refer to https://wesmoodle.wesleyan.edu/pluginfile.php/412381/mod_resource/content/6/info_theory.htm and https://wesmoodle.wesleyan.edu/mod/resource/view.php?id=302147. You will need to include pseudocounts so that the code does not try to calculate log 0. Test your code by using Resource F above (e.g. using some of the sequence sets discussed below).

For the remainder of the project, you may use your Python code and/or Resource F. Please choose a subset of options a-d below.

a. The term 'consensus sequence' is misleading because few if any sequences conform to a given consensus sequence. Assess this statement for sets of donor and acceptor splice sites based on various criteria such as differing intron lengths, etc. In particular, by using different threshold values for information at each nucleotide position, determine potential consensus sequences for donor and acceptor splice sites.

b. As discussed in https://wesmoodle.wesleyan.edu/pluginfile.php/412381/mod_resource/content/6/info_theory.htm, the weight matrix used to calculate individual information can be used to establish consensus and 'excluded' consensus sequences. Construct consensus sequences using different values of τ threshold weights (see Example 6 in BIOL265 Handout https://wesmoodle.wesleyan.edu/mod/resource/view.php?id=252795). Instead of using random background frequencies for A, C, G and T, use as background frequencies the following values – these are the average values for Drosophila cDNAs:
A = 0.288
C = 0.212
G = 0.212
T = 0.288
How do background frequencies affect your calculations of consensus sequences?

c. Markov models can be used to model relative individual information. Based on the discussion and code in https://wesmoodle.wesleyan.edu/mod/folder/view.php?id=252813 compose an HMM version of your above code for relative individual information.

d. Use the sequence walker in Resource F, or write an equivalent sequence walker in Python, to assess whether the spliceosome tends to use donor and acceptor sites with the highest local individual information values. It may be useful to graph individual information scores from the sequence walker to test this idea.

3. RNA Folding (Resource B)

Implement using Python a version of the Nussinov algorithm that predicts the folding of RNA sequences. For a given sequence, your version should output the predicted structure(s) in both base-pair and dot-paren

formats. It should also allow certain natural parameters to be set (e.g. minimum size of a hairpin loop or stacked basepairs). The Nussinov algorithm is discussed in BIOL265 Handout https://wesmoodle.wesleyan.edu/mod/resource/view.php?id=252805.

Use your program and the Mfold program to compare the results of the Nussinov and Zuker algorithms for sequences that represent at least four well known classes of RNAs (e.g. tRNA, 5sRNA). Also, compare your results to the standard secondary structures for the sequences.

Optional: Compare the outputs both qualitatively (by comparing secondary structure diagrams) and quantitatively (by computing appropriate distances between the base-pair and dot-paren formats).

Optional: Try modifying your implementation of the Nussinov algorithm by giving different scores to GC and AU pairing. Compare your results with the results obtained using the standard Nussinov and Mfold algorithms.

4. Genetic Nets

Genetic nets are introduced at https://wesmoodle.wesleyan.edu/mod/resource/view.php?id=302175. The section "Further Analysis" suggests possible initial directions for a project.

Think of approaches to extend the analysis further.
For example:

- What happens if you try multiple trials with the same networks, but starting with different randomly-assigned start states? (or randomly-assigned states not previously visited)
- What happens if two well connected subnetworks have limited connection between each other?
- Do almost identical networks behave similarly -- for example if only one gene has a different Boolean function?
- Other ideas?

5. Proposed Topic

Choose a project topic that is not discussed above. Before investing significant time on it, please write a short paragraph that summarizes the main ideas and submit it to Professor Weir.