Processing of Biological Sequences

In this class, we explore the central dogma of molecular biology. We will learn how to represent, manage, and operate on biological sequences. We will develop algorithms to perform sequence operations including finding the reverse complement and translation.

Learning Objectives

- Review the main concepts of the central dogma of molecular biology.
- Learn how to transcribe and translate DNA sequences.
- Describe sequence statistics.
- Develop code to obtain open reading frames.

1. Review the slides “Basic Processing of Biological Sequences”.
2. Take the Quiz.
3. Solve the exercises and implement the functions described in the programming section.

Find Open Reading Frames

An open reading frame (ORF) is a portion of a DNA molecule that spans from a start to a stop codon. When translated into an amino acid sequence, stop codons are not included. ORFs are therefore indicative of a possible gene location. Several tools have been developed to accomplish this task, among them the ORF Finder from NCBI: https://www.ncbi.nlm.nih.gov/orffinder/

The goal of this exercise is to:

1. explore the NCBI ORF FINDER tool.

- **Step 1:** Start at the NCBI homepage: www.ncbi.nlm.nih.gov
- **Step 2:** Use the All Databases drop down menu to select Gene.
- **Step 3:** Search the transcript sequence (mRNA) for the gene HBB. Search for the combination “HBB and HUMAN”. Select the top entry.
- **Step 4:** Select the "NCBI Reference Sequences (RefSeq)" entry. In the "mRNA and Protein(s)" section, select the NM_000518.5 (transcript identifier).
- **Step 5:** Retrieve the coding sequence in FASTA format.

- Step 6: Submit the sequence in the ORF finder tool.
- Check what is the longest ORF and its coordinates. What is the respective protein sequence?
- How many ORFs are found in this sequence?

Exercise 1. Translate Sequence

Consider the following sequence that codes for a gene:

>coding sequence 1

`accgcaatgtgggattcaacgacgcggaatagttaggtaaaa`

- Identify the start and the stop codons.
- Indicate how many codons exist between the start and stop codons.
- Indicate the translated protein sequence.

Exercise 2. Complete the code

Using the files provided for the class, complete the code for the functions in the script *sequence_functions.py*.

A. Write a function that writes a sequence to a text file: *write_seq_to_file()*

The input is a sequence and a filename.

B. Write a function that reads the standard genetic code from a file to a dictionary: *read_genetic_code_from_file()*

Use the provided file genetic_code.txt. Note that there may be undesirable characters to remove.

C. Implement *the reverse_complement()* function.

D. Write a test function *test_sequence_from_file()*, that reads from the input the name of a file containing a DNA sequence (e.g. example_Hinfluenzae.txt) and performs the following steps on the sequence:

1. Validates;
2. Translates;
3. Obtains the reverse complement;
4. Calculates the GC-content;
5. Performs the direct translation;

Exercise 3. Finding ORFs.

For this exercise, there is no need to import external modules! Use only the code provided for the class.

1. Complete the code for the functions *all_orfs()* and *all_orfs_ord()* in file *sequence_functions.py* and test them properly by using the different test functions in the main method.
2. Consider the DNA sequence *genomic_dna.fa* provided for the exercise. Write a function, that calls other functions from *sequence_functions.py*, in order to obtain the following information:

    2.1 Calculate the longest protein sequence that can be found in the DNA sequence.

    2.2 Indicate how many possible proteins are coded from the DNA sequence and the respective frame ((+1, +2, +3 or -1, -2, -3) from each these were coded. Write the proper code to implement this and add the result as a comment.

Exercise 4. Read fasta to dictionary

1. Write the function called *read_fasta_2dictionary*(). The function should load the dictionary where the keys are the sequence identifiers and the values are the sequences. The output should be dictionary. Use the file PS000727.fasta to test. For the identifier use only the word after the ">" and before the first blank space. For instance, in the following header the identifier should be sp|P51173|APEA_DICDI.

   >sp|P51173|APEA_DICDI DNA-(apurinic or apyrimidinic site) lyase OS=Dictyostelium discoideum OX=44689 GN=apeA PE=2 SV=2

Quiz

When reading a DNA frame which of the following sequences represent a possible protein:

1. ATGTATAGGGTATGA
2. ATGTGAAGGGTAACA
3. GAAGGGTAACATGA
4. TGAGAAGGGTAACA

When scanning for genes we can search directly in the DNA sequence because:

1. DNA is used to translate a sequence
2. The A nucleotide is equal to U
3. The sequence of the RNA transcript is similar to the gene (in the DNA) if we replace U by T

According to the IUPAC code the letter V represents:

1. AGT
2. ACG
3. ATC

What is the reverse complement of TATCCG

1. TATCCG
2. CGGATA
3. CGGTAT
4. TTTCCC

Run the function *test_frequency()* with the following input sequence
MIKFIKILLGCQINVPLNLQNICKNVINIPGFVMLWFLVILIAITMCIY

What is the frequency of N, V, W?

1. 4, 4, 4
2. 5, 4, 1
3. 5, 1, 4