CS-E5875 High-Throughput Bioinformatics 2026

Exercise 1: Sequence analysis

In this exercise, you will work with the genomic sequence of SARS-CoV-2, a strain of coronavirus responsible for COVID-19. You will learn how to analyze sequence composition, identify open reading frames (ORFs), apply probabilistic sequence models, and perform sequence alignments to explore similarities between SARS-CoV-2 and other coronaviruses.

To assist you, we have provided an R script template (`Exercise1_template.R`). You are free to use this template as a guide to complete the exercise. Ensure that you have installed and loaded the following R packages: `ape` (for fetching data from GenBank), `Biostrings` (for sequence manipulation), `ORFik` (for ORF analyses), and `pwalign` (for sequence alignments).

You are required to submit a short report in PDF format that clearly explains the steps of your analysis. The report should include any required plots, results, and discussion of your findings. Additionally, you must upload an R script or notebook containing the code you used to complete the exercise. **Please note that submitting only the R script or notebook is insufficient; the report is an essential part of your submission**.

The deadline for submitting both the report and the code is on 19th of January at 23:55.

Analyzing sequence composition and open reading frames

Retrieve the genomic sequence of SARS-CoV-2 (accession number NC_045512) from GenBank using the `read.GenBank` function in R.

1. (1 point) Compute the GC skew of the SARS-CoV-2 genome in sliding windows. Set a window size of 500 base pairs and a step size of 50 base pairs. Plot the calculated GC skew values against the base pair position. Ensure your plot includes properly labeled axes.
2. (1 point) Define an open reading frame (ORF) as a continuous sequence

of codons starting with a start codon (`ATG`) and terminating at the first in-frame stop codon (`TAA`, `TAG`, or `TGA`). Extract all ORFs from the sequence and compute their lengths. Identify and report the length and location of the longest ORF.

3. (1.5 point) To evaluate the statistical significance of the identified ORFs, implement a permutation test with 1000 iterations. In each iteration, shuffle the nucleotides of the original sequence independently to generate a randomized sequence, then identify ORFs in this randomized sequence. For each observed ORF, calculate the empirical p-value as the proportion of random ORFs that are at least as long. Report the number of ORFs with $p < 0.01$.

   Visualize the results by plotting the distribution of ORF lengths from the randomized sequences and overlaying the lengths of the observed ORFs from the original sequence.

4. (1 point) Apply multiple testing correction to control for false discoveries. Use both the Bonferroni and Benjamini-Hochberg (BH) correction methods to adjust the p-values obtained from the permutation test. Report the number of significant ORFs at $p < 0.01$ after each correction. Discuss the differences between the two methods and which one is more conservative.

Identifying the ORF coding for the spike protein

Coronaviruses are characterized by distinctive spike proteins that protrude from their viral envelope, giving the virus its crown-like appearance. These spike proteins are critical for viral entry into host cells, making them important targets for therapeutics.

Your next task is to identify the SARS-CoV-2 ORF that codes for the spike protein using a probabilistic model trained on spike protein coding nucleotide sequences from other human coronaviruses (HCoV-229E, HCoV-NL63, HCoV-OC43, HCoV-HKU1, SARS-CoV, and MERS-CoV).

Begin by loading the provided nucleotide sequences of spike protein coding regions from the six other human coronaviruses (`data/coronaviruses_spike.fasta`) using the `readDNAStringSet` function from the `Biostrings` package.

5. (1 point) Train a first-order Markov model by estimating the nucleotide transition probabilities from the sequences of spike protein coding regions from the other human coronaviruses. Report the 4×4 nucleotide transition probability matrix.

6. (1 point) Extract the nucleotide sequences of significant ORFs identified using the Bonferroni correction in Question 3. Compute the log-likelihood of each candidate ORF under the first-order Markov derived from the spike protein coding nucleotide sequences of the other human coronaviruses. You can assume that the first nucleotide is given (i.e., $P(x_0) = 1.$). Which ORF is most likely to be the one coding for the spike protein in SARS-CoV-2?

   Note:

   - Log-likelihood is used to prevent numerical underflow when multiplying small probabilities:

     $$\text{Log-likelihood} = \log(P(x_0) \prod_i P(x_i|x_{i-i})) = \log P(x_0) + \sum_i \log(P(x_i|x_{i-1}))$$

   - To account for differences in ORF lengths, normalize the log-likelihoods by dividing by the number of transitions in each ORF. The normalized log-likelihood represents the average log-likelihood per transition, enabling a fair comparison across ORFs of varying lengths.

Comparing SARS-CoV-2 to other coronaviruses

In the previous section, you identified the ORF coding the spike protein. If you did not complete this step, use the sequence provided in `data/SARS-CoV-2_spike.fasta`.

Your next task is to explore how SARS-CoV-2 relates to other human coronaviruses by comparing their spike protein sequences. To keep the computational complexity reasonable, we focus on the spike proteins rather than the entire viral genome.

Begin by loading the nucleotide sequences coding the spike proteins from `data/coronaviruses_spike.fasta` and `data/SARS-CoV-2_spike.fasta`.

7. (0.5 points) Translate the nucleotide sequences into amino acid sequences.

Verify that the translated amino acid sequences begin with methionine (M), which is indicative of a correct start codon (`ATG`).

8. (1 point) Perform a global alignment of the SARS-CoV-2 spike protein against each spike protein sequence from other coronaviruses. Use the Needleman-Wunsch algorithm with the BLOSUM62 substitution matrix. Use gap opening penalty of 10 and gap extension penalty of 2. Report the alignment scores. Identify and report the human coronavirus whose spike protein sequence is globally most similar to that of SARS-CoV-2.

9. (1 point) Perform a local alignment of the SARS-CoV-2 spike protein against the spike proteins of other human coronaviruses. Use the Smith-Waterman algorithm with the BLOSUM62 substitution matrix. Use gap opening penalty of 5 and gap extension penalty of 1. Report the alignment scores. Identify and report the human coronavirus spike protein that shows the highest local similarity to SARS-CoV-2.

10. (1 point) Bats are considered natural reservoirs for many coronaviruses, including relatives of SARS-CoV-2. Conduct a BLAST search online (`https://blast.ncbi.nlm.nih.gov/Blast.cgi`) by uploading the SARS-CoV-2 spike protein sequence and searching against a database of bat coronaviruses. Identify and report the top bat coronavirus with the highest similarity to SARS-CoV-2 based on the spike protein sequence. Include the accession code, alignment score, E-value, percentage identity and the host species associated with the virus.

    Hint: From the BLAST homepage, choose "Protein BLAST" and use the "Non-redundant protein sequences (nr)" database. Select "Bat coronavirus (taxid: 1508220)" as the organism and "blastp (protein-protein BLAST)" as the algorithm. By expanding the "Algorithm Parameters" box at the bottom, set the word size to 6. To access more details about a result, click on the accession code in the BLAST output.