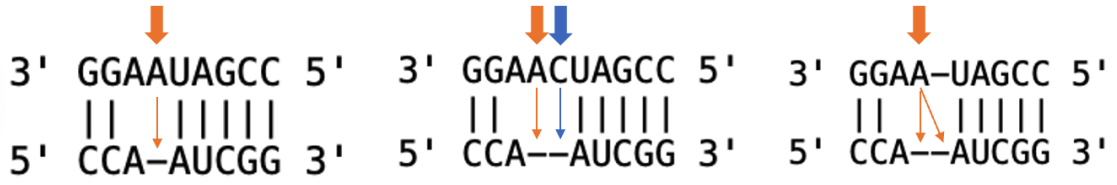
MMG1344H: Foundational Computational Biology (FCB) I **Winter 2022**

Assignment 2:

Part 1

Use Python or R to adapt the Smith-Waterman algorithm and find local alignments between two RNAs that optimize hybridization. Note that in the original Smith-Waterman application two DNA sequences were aligned in the same 5'-3' orientation, while here one 5'-3' RNA sequence is being aligned with another 3'-5' RNA sequence. To score the hybridization of one base from one strand to a base from the other strand, we will use the following scheme:

- G:C base pairs get 3 points
- A:U base pairs get 2 points
- G:U base pairs get 2 points (NOTE: the non-Watson-Crick base-pairing G:U is more favoured in RNA:RNA hybridization than is G:T in DNA:DNA hybridization (Vendeix, Munoz, and Agris 2009))
- All other base pairings get -1 point
- For simplicity, while neighbouring gaps are allowed, please consider only gaps of length 1, using a gap penalty of 1 point (i.e. a gap will receive -1 point). This requirement is illustrated as follows:



Single gap (length = 1)	Neighbouring gaps (length = 1)	Single gap (length > 1)
✅ *Allowed*	✅ *Allowed*	❌ *Not allowed*

Input

Your program will read all test cases from one input FASTA file placed in the working directory (i.e. the same directory in which your program file lives). **All sequences are provided in 5' to 3' order.**

A sample test set (*sample_input.fasta*) and the full test set (*test_input.fasta*) are provided to you.

Output

Your program should generate optimal local alignments for each test case in the full test set and saved alignments into a file named *output.txt*. **If there are multiple optimal alignments for a given test case, your program should output ALL of them.**

The output for the sample test set (*sample_output.txt)* is provided to you. Please use it as a reference when you format and validate your alignments.

Part 2

Here we use data from a 2014 study of the earliest cell-fate decision in mammalian development (Biase, Cao, and Zhong 2014). This study looked at gene expression in sister blastomeres (each cleaved from the same zygote) by using single-cell RNA sequencing (scRNA-seq). They reported highly reproducible between-blastomere differences among 10 samples of 2-cell stage mouse embryos and 5 samples of 4-cell stage embryos. Between-blastomere gene expression differences appeared to dominate between-embryo differences, and these differences were sufficient to cluster sister blastomeres into distinct groups. For numerous protein-coding genes, reproducibly bimodal expression in sister blastomeres was observed, and this could not be explained by random fluctuations.

You will focus on the scRNA-seq gene expression data (in units of "fragments per kilobase of transcript per million mapped reads"; or FPKM) which covers 6812 genes from 4-cell mouse embryos (20 samples of 4-cell blastomeres) and 2-cell mouse embryos (20 samples of 2-cell blastomeres).

1. Implement the *k*-means algorithm (do NOT use pre-existing Python or R implementations or packages for this) with ***k* = 4** and Euclidean distance with standardized data. Test and run your code on the scRNA-seq data provided (*Biase_2014.csv*). You will need to "standardize" your data, i.e., rescale the values such that the mean for all genes (over all 40 samples in the data set) is equal (or very close) to 0, and so that the variance for each sample is equal or very close to 1.
2. The *k*-means algorithm tends to converge quickly but it may be stuck on local optima, so please run your algorithm 10 times with different initialization conditions (controlled by a random seed) to see if this behaviour exists on your data. For each run, please report:
    a. The random seed.
    b. The cluster size and the identity of the objects (cell samples) in every cluster.
3. Using the best clustering run from Step 2, for each cluster, find all the genes that show enriched or depleted expression in that cluster. To estimate significance (using a *P*-value threshold of 0.05), use a two-sided Mann-Whitney *U* test (aka Wilcoxon rank-sum test). For example, to test each gene in Cluster 1, apply the test to examine whether the expression distribution for that gene is different between Cluster 1 samples and samples in all other clusters. Don't forget to use a Bonferroni approach to correct your P-values for multiple hypothesis testing (and think carefully about how many tests for which you have to correct). Please report significant genes in each cluster with their Bonferroni corrected P-value.
4. Based on the gene lists, can you identify which 4-cell-enriched cluster is similar to which 2-cell-enriched cluster? Please explain why.

Submission

Please submit a zip file (first_lastname.zip) with:

1. Your source code (in .R or .py format).
2. Your program's output for the full test set (in .txt format) in Part I.
3. A README.txt file on how to run your program.
4. A document (in .doc or .pdf format) with answers to questions in Part II.

References

Biase, Fernando H., Xiaoyi Cao, and Sheng Zhong. 2014. "Cell Fate Inclination within 2-Cell and 4-Cell Mouse Embryos Revealed by Single-Cell RNA Sequencing." *Genome Research* 24 (11): 1787–96.

Vendeix, Franck A. P., Antonio M. Munoz, and Paul F. Agris. 2009. "Free Energy Calculation of Modified Base-Pair Formation in Explicit Solvent: A Predictive Model." *RNA* 15 (12): 2278–87.