AiB Project 3 - Multiple Alignment

Authors: Astrid Dahl (201806016), Ingeborg Bitsch Jensen (202005428), Henry Charlton (202303414), Carolina Jørgensen (201707243)

Introduction:

Both *sp_exact_3.py* and *sp_approx.py* are functional. For short sequences as e.g., testdata_short.txt, sp_exact_3.py can be used to compute both score and alignment in a reasonable amount of time. For longer sequences as e.g., testdata_long.txt, the score and alignment cannot be computed in a reasonable amount of time (computation time > 1 hour).

Methods:

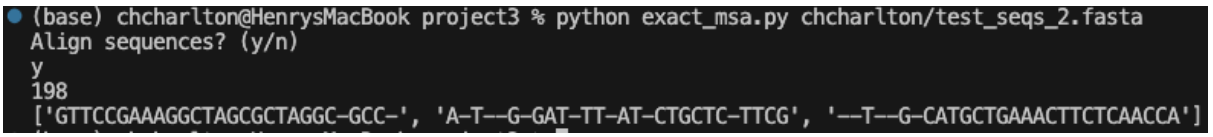
General

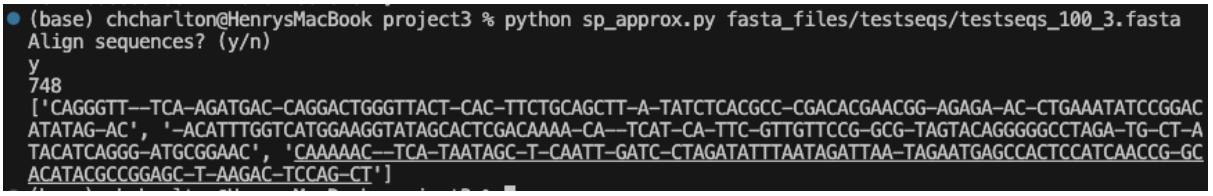
The code for sp_exact_3.py is implemented as described in "Multiple String Comparison - The Holy Grail". The code for sp_approx.py is implemented as given in the PowerPoint SP-MSA-Approx.pdf. *sp_exact_3.py* has been tested on the sequences in test_short.txt, and the correct score was found. The *sp_approx.py* has been tested using the example from the PowerPoint SP-MSA-Approx.pdf slide 11. The correct alignment was found and the same score was found as when the msa_sp_score_3k.py program was used on the computed alignment..

Value Checking

If the sequences provided include characters that are not included in the alphabet {A, T, G, C}, the character will be substituted by a randomly sampled character from the alphabet, as described in question 4.

Script usage examples:





Experiments:

Answer the following questions:

1. *What is the score of an optimal alignment of the first 3 sequences in brca1-testseqs.fasta (i.e. brca1_bos_taurus, brca1_canis_lupus and brca1_gallus_gallus) as computed by your program sp_exact_3? How does an optimal alignment look like?*

   Answer:

The following is the exact alignment of the first three BRCA1 test sequences, as computed by our exact SP msa script. It had a cost of 790 and took approximately 14 minutes to terminate.

ATGGATTTATCTGCGGATCATGTTGAAGAAGTACAAAATGTCCTCAATGCTATGCA-
GAAAATCTTAG--AGTGTCCAAT-ATGTCTGGAGTTGATCAAAGAG-CCT-GTCTCTAC
AAAGTGTGA-CCA-CA-TATTTTGCAAATTTTG-TATGCTGAA-AC-TTCTCAACCA-GA
AGAAAGGGCCTTCACAATGTCC--TTTGTGTAAGAATGA-

ATGGATTTATCTGCGGATCGTGTTGAAGAAGTACAAAATGTTCTTAATGCTATGCA-
GAAAATCTTAG--AGTGTCCAAT-ATGTCTGGAGTTGATCAAAGAG-CCT-GTTTCTAC
AAAGTGTGA-TCA-CA-TATTTTGCAAATTTTG-TATGCTGAA-AC-TTCTCAACCA-GA
GGAAGGGGCCTTCACAGTGTCC--TTTGTGTAAGAACGA-

GCGAA---ATGTA-ACA-CG-GTAGAGGTGAT-CGGGGTG-CGTT-ATAC-GTGCGTGGTG
ACCTCGGTCGGTGT-TGACGGTGCCTGGGGTTCCTCAGAGTGTTTTGGGGTCTGAA
GGATG-GACTTGTCAGTG-ATTGCCATTGGAGACGTGCAAAATGTGCTTTCAGCCAT
GCAGAA-GAA-CTT-GGAGTGTCCAGTCTGTTTAGATGTGAT

2. *What is the score of the alignment of the first 5 sequences in brca1-testseqs.fasta (i.e. brca1_bos_taurus, brca1_canis_lupus, brca1_gallus_gallus, brca1_homo_sapiens, and brca1_macaca_mulatta) as computed by your program sp_approx? Which of the 5 sequences is chosen as the 'center string'?*

   Answer:

   The center string selected is brca1_bos_taurus. The cost is 3310, as per the approximate MSA algorithm.

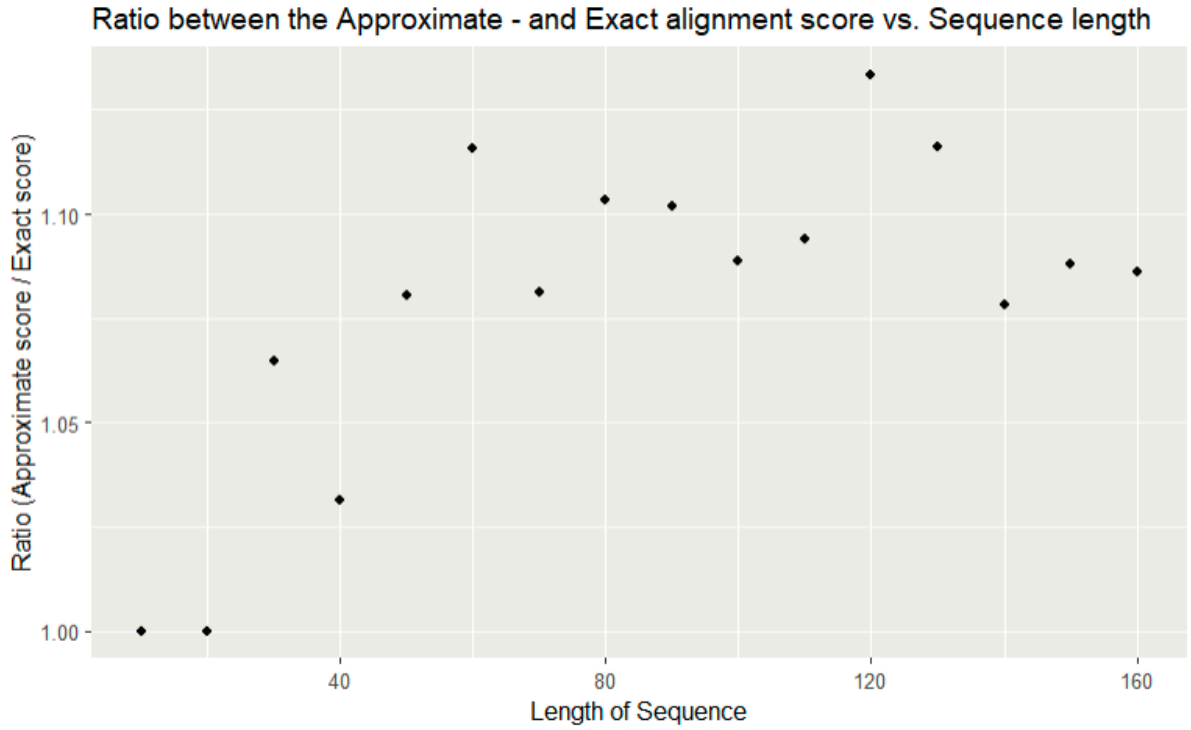
3. *Make an experiment comparing the scores of the alignments computed by sp_exact_3 and sp_approx that validates that the approximation ratio of sp_approx is 2(k-1)/k for k sequences. i.e 4/3 for three sequences.*
   You should use the testdata in testseqs.zip that contains 20 fasta files (testseqs_10_3.fasta, testseqs_20_3.fasta, ..., testseqs_200_3.fasta) each containing 3 sequences of lengths 10, 20, ..., 200.
   For each triplet of sequences (i.e. each fasta file), you should compute the optimal score of an MSA using sp_exact_3 and the score of the alignment produced by sp_approx. Make a graph in which you plot the ratio of the computed scores for each sequence length. Comment on what you observe.

   Answer:

   The question has been answered by using the first 16 test files (sequences of length 10 - 160). It was deemed unnecessary to use the last four files (170 - 200), since the pattern was revealed by plotting the first 16 files.

   The interesting thing seems to be that the ratio between the approximate and exact alignment score reaches a plateau. This plateau is reached when the sequences are 40-60 nucleotides of length. Hereafter it seems as if the ratio doesn't exceed a set value. This value seems to be

around 4/3. The equation 2(k-1)/k, where k is 3, equals 4/3. We can hereby validate that the approximation ratio can be computed by using the above formula.



4. *Finally, you must also compute a multiple alignment (using column-based sum-of-pairs score) with a score as close to optimum as possible of the 8 full length BRCA1 genes in brca1-full.fasta. Since these 8 sequences are 'real' dna data, it can actually contain other symbols than A, C, G and T due to sequencing, see fx this page on Wikipedia. Inspecting the sequences show that fx the rat sequence (rattus_norvegicus) contains two N's. You must decide how to handle this, e.g. substitute the N by any base you like.*

Answer:

Upon analysis we noticed that the BRCA1 sequences contained N's, R's, and S's. The Wikipedia page suggests that N's should be replaced by a random choice from ACGT, the R's from AG, and the S's from CG. We implemented this replacement as a part of our read_fasta function within the script for the approximate alignment. Note: this means that each time the script is run, there is a potential for a different alignment since the choices of replacement are random. The cost for the alignment using the SP approximate algorithm was 264007. The full alignment can be found in "brca1_alignment.fasta." The script took 26 minutes to run on a local machine.