Integrative Transcriptomics
Prof. K. Nieselt,
Institute for Bioinformatics and Medical Informatics Tübingen
Prof. S. Nahnsen,
Institute for Bioinformatics and Medical Informatics Tübingen

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Lecture: Grundlagen der Bioinformatik — SoSe 2022

Assignment 2 (20 points)

Hand out: Thursday, May 5
Hand in due: Thursday, May 12, 18:00
Direct inquiries via the ILIAS forum or to your respective tutor at:
Mathias Witte Paz: iizwi01@uni-tuebingen.de
theresa-anisja.harbig@uni-tuebingen.de
meret.haeusler@student.uni-tuebingen.de
jules.kreuer@student.uni-tuebingen.de
simon.heumos@qbic.uni-tuebingen.de

Theoretical Assignments

1. **Global and local alignment by hand** (6P)

   For the following two sequences, $X =$`TGATTCAT`, $Y =$`GAGAT`, compute both an optimal global and local alignment, using a linear gap penalty and the following scoring parameters:

   $s(a, b) = -2$ if $a \neq b$ and $s(a, a) = +2$ and $d = 3$.

   Hand in the DP matrix, as well as one alignment via traceback for each problem.

   Hint: You can use `https://www.tablesgenerator.com/` to create a table in LaTeX. If you want to include pictures, please include only **good quality** pictures or scans.

2. **BLAST - theoretical considerations** (3P)

   (a) For a protein sequence of length 175 and a database of length $8 \cdot 10^8$, what normalized score S' is needed for an E-value of 0.02?

   (b) The E-value (E) gives a measure of the number of false results (false positives) you would expect to see if you used a given alignment score threshold.

   i. How would you expect the E-value E to change if we double the length of our query sequence?

   ii. How would you expect the E-value E to change if we cut the size of our database in half?

   iii. In general, would you expect the E-value E to change if we use a different scoring matrix?

Practical Assignments

3. **Using BLAST** (4P)

   Identify the sequence provided in the file *unknown_protein.fasta* using BLAST. What BLAST program is suited for this task?

   From the results, discuss briefly (around 250-300 words) the following question:

   - What is the function of the protein?
   - From which organism does the protein probably come from?
   - How trustworthy are the results of your search? Argument using the different search values (such as E-value, percentage identity, etc.)
   - How about the other hits? Do they confirm the result of your search?

4. **Needleman-Wunsch Algorithm** (7p)

   Implement the Needleman-Wunsch Algorithm with linear gap-penalties. To do so, implement a function `compute` that fills the dynamic programming matrix as explained in the lecture, and a function `traceback` that provides **one** possible alignment with the optimal score. The parameters for the scoring of matches and mismatches as well as gap penalty should be read from the command line. Run the alignment for the files *yersenia_1.fasta* and *yersenia_2.fasta* with the following parameters:

   $s(a, b) = -2$ if $a \neq b$ and $s(a, a) = +2$ and $d = 4$.

   Use your FASTA parser from the previous tasks to read the files or the FASTA reader provided by `BioPython`. Your program should write the aligned sequences to a file and the optimal alignment score to the console.

5. **Gain Extra Points: Enrich your Output with Additional Information** (+3P)

   Write a more verbose output of your Needleman-Wunsch algorithm to a file. The output should contain (but is not limited to) the following:

   (a) A visual alignment showing the matches, mismatches and gaps between both sequences

   (b) The used algorithm parameters

   (c) The amount of matches/mismatches/gaps

Please read the questions carefully. If there are any questions, you may ask them during the tutorial session or in the forum of ILIAS. You will usually get an answer in time, but late e-mails (e.g. the evening of the hand-in) might not be answered in time. Please upload all your solutions to ILIAS. Don't forget to put your names on every sheet **and** in your source code files. Please pack both your source code as well as the theoretical part into one single archive file and give it a name using this scheme: `<name1>_<name2>_<Assignment>_<#>.zip`. The program should run without any modification needed.