ITO 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 2023

Assignment 10 (20 points)

Hand out: Thursday, July 6 18:00
Hand in due: Thursday, July 13 18:00
Direct inquiries via the ILIAS forum or to your respective tutor at:
caroline.jachmann@uni-tuebingen.de
meret.haeusler@student.uni-tuebingen.de
simon.hackl@uni-tuebingen.de
carolin.schwitalla@qbic.uni-tuebingen.de

Theoretical Assignments

1. **Supervised training** (2P)

   Given a hidden Markov model $M = \{\Sigma, Q, P, E\}$, for which the parameters of the transition matrix P and emission probability matrix E need to be trained. For the approach of supervised training we sometimes have the problem of overfitting. To avoid this we first count the number of times each particular transition or emission occurs in the training sequences

   P_{kl}: Number of transitions from state k to l
   $E_k(b)$: Number of emissions of b in state k

   We then obtain the maximum likelihood estimators (MLE) for (P, e) by adding plus 1 to each observation and normalize appropriately (this is also called adding a pseudo count with the *Laplace* rule). How do the final MLE p_{kl} and $e_k(b)$ look like? Write down the corresponding general formulas.

Practical Assignments

2. **PSWM of binding sites of a transcription factor** (4P)

   For this task you only need an alignment tool of your choice to conduct a multiple sequence alignment. The other sub-tasks should be computed by hand. Cite the tool you used for the MSA.

   Assume a transcription factor TF has been measured to bind a set of 10 sequences:
   `'AGTAGCCA'`, `'CGTTCCTACA'`, `'GTTGGTACC'`, `'GTTGCCA'`, `'TGTCGCCATG'`, `'CGTTGTCAT'`, `'AGTTACCA'`, `'GTTAGCACA'`, `'GTTTTTATG'`, `'GGTTGGTA'`.

   (a) Use a multiple sequence alignment tool of your choice to align these 10 sequences.

   (b) Identify the core 7 residue part of the alignment that can be aligned without gaps.

(c) Convert this 7-long multiple alignment into a position specific weight matrix PSWM, as introduced in the lecture using the maximum-likelihood method (i.e., normalized counts for each column of the matrix).

(d) From this compute the PSWM with logarithmic propensity values using $q(A) = q(T) = q(G) = q(C) = 1/4$ as background values (use also Laplace pseudocounts, see task 1).

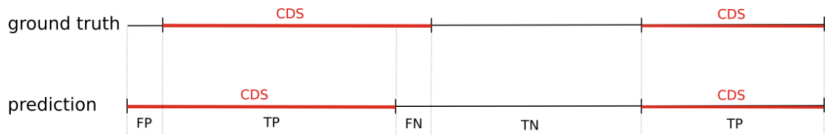
Hand in the MLE PSWM and the PSWM with logarithmic propensities.

3. Gene prediction (14P)

In this task we ask you to compare two gene prediction methods of your choice from the following set: `PROKKA`, `GeneMark`, and `Bakta`, three gene prediction software tools that use either a dynamic programming approach (`Prokka`), hidden Markov models (`GeneMark`) or linear combination (`Bakta`) to identify coding regions in prokaryotic DNA.

All three programs output their feature predictions in a `.gff`-file. The GFF file format is used to describe predicted features like CDS (short for coding sequence, so the ORF) or exon of DNA sequences. Inform yourself about the file format, especially what each line and what the nine different columns represent.

Your task is to run **two of the three** gene prediction methods on the genome of *Legionella pneumophila* and compare their respective output to the given `.gff`-file (`legionella_pneumophila_Philadelphia1_groundtruth_annotation.gff`), which we consider as the ground truth. In more detail, we only want to examine the coding sequences (CDS) on the nucleotide resolution level. Therefore, we count nucleotides/positions that are predicted correctly (true positives, true negatives) and not correctly (false positives, false negatives) as either being 'coding' or 'not coding', which is indicated in the following scheme:



Note: The sum of true positives, true negatives, false positives and false negatives equals the length of the genome. Make sure that when calculating the sum of true positives, true negatives, false positives and false negatives you consider that the predictions are made for both strands of the genome.

(a) Run two of the three methods:

- In order to run `PROKKA` go to the website `https://usegalaxy.org/` and create an account there. After confirming your email address search for the `PROKKA` tool and upload the input genome file for analysis. Select only the `.gff`-file as an output.
- Run `GeneMark` using the web service `http://exon.gatech.edu/GeneMark/gmhmmp.cgi`. Make sure you select the correct species (strain) and output format (gff).
- Run `Bakta` using the web service `https://bakta.computational.bio`
- Save the respective outputs on your computer. Include the two gff files in your hand in and name them accordingly.

(b) Comparison of results:

- Get familiar with the provided GFF parser[1] to include it in your program. You might want to read the class documentation for instructions (See Bonus task for an alternative). Make sure that you include the provided .jars as dependencies in your IDE.
- Write a Java program which compares the `.gff`-files of the two programs to the provided ground truth. Focus on the CDS feature. Compute true positives, true negatives, false positives and false negatives as described in the introduction by comparing the CDS regions of the respective files. From these compute the sensitivity, specificity and accuracy of `PROKKA`, `Bakta` and/or `GeneMark`, respectively. Hint: you may need to delete the fasta appendix and lines containing question marks in the .gff outputs from `Bakta` and `PROKKA` in order to parse them with the GFFReader.
- Report the output in your PDF and cite the tools that you compared in this task. Which of the two prediction tools would you use based on the calculated statistics? Justify your answer in 1-2 sentences.
- **Bonus (+2P):** Write your own `.gff` parser. Indicate in the PDF if you have implemented your own GFF parser.

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.

[1] `https://biojava.org/docs/api/org/biojava/nbio/genome/parsers/gff/GFF3Reader.html`