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 8 (20 points)

Hand out: Thursday, June 23
Hand in due : Thursday, June 30, 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. **Sequencing approaches (in a nutshell)** (4P)

   Inform yourselves on **one** of the following next-generation sequencing technologies:

   - Solexa/Illumina Sequencing
   - PacBio Sequencing
   - Oxford Nanopore Sequencing

   Summarize the most important characteristic of your chosen technology in a *nutshell*. For this, hand-in 3-4 bullet points (at most one sentence per bullet point) with the core information of the chosen technology (i.e. as you would do for a slide in a presentation).

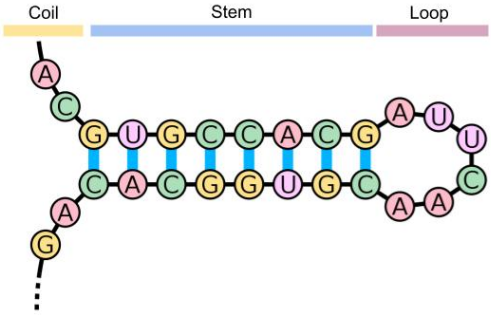
2. **HMM for RNA sequence structure prediction** (3P)

   We will discuss in a further chapter of the lecture (Chapter 11) that RNA sequences might fold into different structures, such as the hairpin loop structure. This structure consists of the following elements: the stem which is created by two complementary sequences that are not necessarily direct neighbours in the sequence, the loop which refers to the non-complementary subsequence between both paired sequences, and the coil referring to the rest of the bases. The figure[1] below shows a hairpin loop and labels its three structural elements.

   Your task is to draw the hidden and emissions states of the HMM that could be used to address the classification of an RNA sequence into a hairpin loop structure with these 3 structural elements[2]. You do not need to compute the transition or emission probabilities, but label each parameter correctly. What is the total number of parameters for this HMM?

[1] Adapted from: `https://en.wikipedia.org/wiki/Stem-loop`

[2] Disclaimer: HMMs are not a common approach for the classification of RNA bases into these three categories.



3. **Transition matrix computation by hand** (2P)

Given the following sequences of exonic regions, compute the transition matrix P_{exonic} by hand for the alphabet $\Sigma = \{b, e, A, T, C, G\}$.

```
seq1 = CTTCTTGTGT       seq2 = GTTGGACACTTTCGGG      seq3=TTGCTGTCGTA
seq4 = CAGACGTAAGTCG    seq5 = GCCCGTATAGGGC         seq6=CCTGTG
```

Practical Assignments

4. **Transition matrix and log-odds computation** (11P)

In a (hopefully far away) future, due to climate change and evolutionary pressure, the genomes of all organisms have been found to only contain the nucleotides guanine and cytosine, since the bonds between adenine and thymine were not strong enough to keep up with the high temperatures. Your colleagues have assembled the genome of one organism and they would like to know how likely it is that one specific contig (`contig.fasta`) encodes a protein. Since the organism lacks the known start and end codons, it is not possible to search for open-reading frames. Luckily, you have experimental evidence and have created two training sets: one with protein coding regions (`cds_set.fasta`) and one for non-protein coding regions (`notcds_set.fasta`). To help your colleagues you should proceed as following:

(a) **Training:** Create a program (`train_hmm.py`) that computes a transition matrix for a given FASTA file and exports it with a name of choice. Check the script (p. 158) to find the expected format for the transition matrix.

(b) **Testing:** Create a second program (`test_sequence.py`) that reads a sequence (FASTA file) and two transition matrices P_+ and P_- (with same format as above) to compute the probability of the given sequence under each model. This program then prints out into the console the probabilities of the sequence under the two given models and the respective log-odds ratio.

Apply your programs:

(c) Parse the file `cds_set.fasta`, compute the transition matrix P_{CDS} and export it using your program `train_hmm.py`.

(d) Repeat the same as above for the file `notcds_set.fasta` to compute and export P_{notCDS}

(e) Compute the probabilities for the provided sequence (`contig.fasta`) to be produced under each model and the respective log-odds ratio using your program `test_sequence.py`.

(f) What is your conclusion: is the sequence more likely to a CDS or not to be a CDS? Why?

Hand in the code for your two programs and the two transition matrices P_{CDS} and P_{notCDS}.

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.