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 11   (20 points)

Hand out: Thursday, July 13 18:00
Hand in due: Thursday, July 20 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. **RNA Secondary Structure Theory**   (4P)

   Non-coding RNAs (ncRNAs) fulfill different molecular functions. One of them is the inhibition of the translation of protein coding genes via riboswitches. Describe the secondary structure elements of a typical riboswitch and explain how the secondary structure of this class of ncRNAs allows the regulation of protein translation (write at most 100-150 words). In addition, please also hand in a sketch of this process that you refer to within your description. Be creative. Don't forget to cite sources when you use them.

2. **Dot-Bracket Notation**   (2P)

   Considering the following RNA secondary structure in dot-bracket notation

   $$(((((......)))))...(((((.....)))))$$

   , we ask you to think of a sequence whose secondary structure matches the given dot-bracket notation. Not only submit your sequence, but also show that it matches the given structure. Feel free to use a web-server to answer the last part of this task (e.g. `http://rna.tbi.univie.ac.at//cgi-bin/RNAWebSuite/RNAfold.cgi`)

3. **Modified Nussinov**   (4P)

   The Nussinov algorithm in its basic form as presented in the lecture may lead to infeasible structures: For example, neighboring bases can be paired, as in the solution `.((..()))` for the sequence GGGAAAUCC.

   In this task, we ask you to modify the Nussinov folding algorithm so that hairpin loops must have a minimum length of l. Provide the new recursion equations as well as the initialization of the modified algorithm. Does the traceback also differ?

Practical Assignments

4. **Dot-Plot Visualization for RNA Transcripts** (10P)

Background:

Originally, dot-plots were introduced in bioinformatics to compare biological sequences, i.e., to identify regions of high similarity between two sequences. To this end, the usual approach for dot-plots of sequences is to plot the sequence positions of two sequences against each other and fill each cell that has the same (nucleotide-)base on both axis.

Dot-plots have since been extended to represent interactions between biological sequences. Dots/filled cells may vary in size or color depending on desired features. For RNA sequences, the positions of a single sequence can be plotted against itself, and cells corresponding to bases that form hydrogen bonds are filled. You can see an example dot plot in Figure 1.
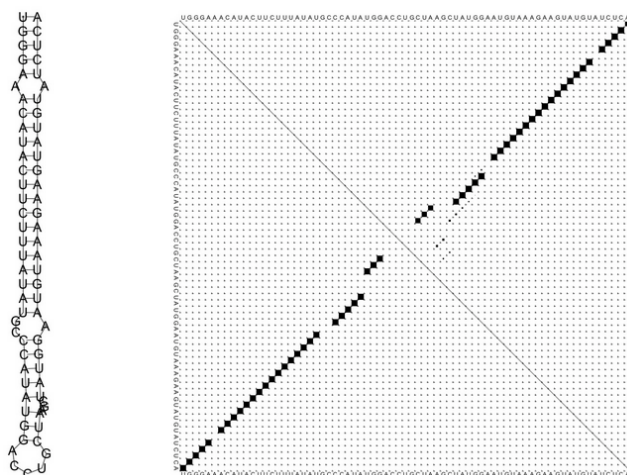

Abbildung 1: RNA Sec. Struc. dot plot, *Applied Bioinformatics, Chap. 7* by D. A. Hendrix

Task:

In this task we ask you to implement a program that uses a dot-plot to visualize RNA structures by highlighting base pairs. For the actual visualization we will use the `Java XChart` library. You will find a basic tutorial on how to use `Java XChart` (especially to generate scatter plots) in the code skeleton of this assignment.

Your program should receive an RNA sequence and a secondary structure in dot-bracket notation as input and generate a dot-plot image as output. The respective data is not parsed from a file this time but directly passed to your program, i.e., it should be executable as follows

```
java -jar <name1>_<name2>_<assignment>_11.jar -seq <RNA_SEQUENCE> -struc <DOT_BRACKET_STRING>
```

(a) Implement the method `parseRnaDbString(String rnaDbString)` of the `RNADotPlot` class to parse a RNA secondary structure in dot-bracket notation into two lists of coordinates, i.e., the x- and y-axis coordinates which are about to be filled in the dot-plot. The lists are to be stored in the `RNADotPlot` class and the i-th entries of the lists, x_i and y_i, indicate that the bases at position x_i and y_i form a bond.

(b) Implement the method `getDotPlot()` of the `RNADotPlot` class to generate a corresponding scatter plot `XYChart` instance that represents a dot-plot of the stored RNA sequence and structure. Use the information of the `Java XChart` tutorial in the code skeleton for this task. Visualize and interpret the given tRNA sequence and its structure in the file `resources/tRNA.txt`. Hand-in the resulting visualisation in your `pdf` report.

(b′) Extend your visualization, e.g., by highlighting of G-U base pairs (by adding distinc series) or adjusting the axis tick labels of the plot with the base contents of the sequence. Feel free to be creative.

(c) Explain briefly how the dot-plot allows the user to find hairpins and distinguish them from interior loops or multi-loops. If you did not manage to solve subtask (b) you can use the provided dot-plot in the file `resources/tRNA-dot-plot.png`.

Note: Common dot-plot visualizations label their x- and y-axis with the respective RNA sequence and start their coordinates at the upper-left corner. For your implementation it is sufficient when the ticks are labeled by the positions of the sequence and start at the lower-left corner of the grid.

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.