Tutorial 1 – Constructing a catalogue of tandem repeats and determining the repetitive content of a genome

1. Introduction

First of all, download the file `tutorial_data.tgz` (http://www.lbm.fmvz.usp.br /trap/tutorials/tutorial_data.tgz) to a directory of your choice. Decompress the file using the following command:

```
tar xzvf tutorial_data.tgz
```

or, alternatively…

```
gzip –d tutorial_data.tgz
```

and then… `tar xvf tutorial_data.tar`

This command will create the `tutorial_data` directory, which contains five subdirectories:

- `config_files` - this directory contains configuration files of three tutorials: `catalogue.cnf`, `annotation.cnf` and `markers.cnf`.
- `data` - this directory contains a multiple sequence FASTA file (`tutorial_data.fasta`) comprising five distinct sequences. Single sequence files of these five sequences are also provided in this directory:
    - `contig00000040.fasta`
    - `contig00000153.fasta`
    - `contig00000278.fasta`
    - `contig00000333.fasta`.
    - `contig00000382.fasta`
- `test` - this will be your working directory for the tutorial and is initially empty.
- `trf` - this directory contains output files previously generated by TRF, using the `tutorial_data.fasta` file as an input.
- `tutorials` – this directory contains three subdirectories (`catalogue_dir`, `annotation_dir` and `markers_dir`). Each one harbors the output files of the corresponding tutorial. We are providing these files just in case you have problems in running TRAP, and want to check how the output files should look for this example data set.

We have previously run TRF version 4.00 on a multiple sequence FASTA file (`tutorial_data.fasta`). As mentioned above, the output files are stored in the `/trf` directory. We used the Linux version of TRF. Command for invoking TRF may vary depending on the platform, version of TRF and configuration of the server. The following command was used in our case:

```
trf400.linux.exe tutorial_data.fasta 2 5 7 80 10 25 1000 –f
```

In this tutorial, we describe how to run TRAP in order to generate a comprehensive analysis of the tandem repeat content of a genome. We will create HTML files that can be visualized as web pages, as well as CSV files that can be opened using any spreadsheet program like MS Excel, KSpread, OpenOffice Calc, etc.

2. Running TRAP on the command line:

To run TRAP, first go to the `test` directory, and then type the following command:

```
trap.pl –i ../trf/tutorial_data.fasta –od catalogue_dir –of
cataloge_file –min 1 –cpmin 2 –id 70 –tbf html+csv –sort size
–rr -trf
```

If everything goes well, a new subdirectory will be created:

```
catalogue_dir
```

Inside the `catalogue_dir` directory, you will find the following files:

```
catalogue_file_redundant_regions.html
catalogue_file_TRAP_summary_table.csv
catalogue_file_TRAP_complete_table.csv
catalogue_file_TRAP_summary_table.html
catalogue_file_TRAP_complete_table_index.html
```

…and a newly created `html_data/` directory.

3. Understanding TRAP parameters:

A comprehensive explanation on all TRAP parameters is depicted in the *How to run TRAP* document. Please refer to it if you need more information.

Shortly, the command line used above specifies the following parameters:

- `-i tutorial_data.fasta` – uses files with names containing the prefix "`tutorial_data.fasta`" (see the file names in the `trf` directory) for input.
- `-od catalogue_dir` – specifies `catalogue_dir` as the directory that will store TRAP output files.
- `-of catalogue_file` – specifies `catalogue_file` as the prefix for all TRAP output file names.
- `-min 1` – selects repeats with period sizes ≥ 1.
- `-cpmin 2` – selects repeat *loci* with copy number ≥ 2.
- `-id 70` – selects repeat *loci* with percentage of matches ≥ 70% between adjacent repeat units overall.
- `-tbf html+csv` – generates summary and complete tables on both HTML and CSV file formats.
- `-sort size` – sorts out the output tables according to the period size of the repeat units.
- `-rr` – generates a redundancy report of the selected repeats.
- `-trf` – creates output files that resemble the original TRF output HTML files, but displaying only the repeat *loci* selected by TRAP.

4. Understanding TRAP output files:

The following files are created:

- `catalogue_file_TRAP_summary_table.csv`
- `catalogue_file_TRAP_summary_table.html`

These HTML (for web browsers) and CSV (for spreadsheet programs) files list all the selected repeats, classified in increasing order, according to the period size. In addition, regions containing redundant repeats (see parameter `-rr`) are identified and the result of the calculation of the number of repeat bases excluding redundancy is also displayed. The figure below is a screenshot of the HTML visualized on a browser.

Period size	Total number of repeat *loci*	Total number of repeat units	Average number of repeat units/*locus*
1	1	17	17
3	53	943.4	17.8
4	3	18.2	6.06
5	5	16.2	3.24
6	8	28.3	3.53
7	21	95.5	4.54
8	2	6.19	3.09
9	9	30.9	3.43
10	4	9.19	2.29
11	1	2.5	2.5
12	4	13.7	3.42
13	1	3.9	3.9
15	2	10.7	5.35
16	1	2.1	2.1
17	1	3.4	3.4
20	1	6.8	6.8
21	3	36.8	12.26
24	1	2.9	2.9
26	1	2.29	2.29
28	1	6.8	6.8
30	1	4.4	4.4
32	1	3.6	3.6
38	1	6.7	6.7
45	1	4.59	4.59
84	1	2.1	2.1
85	1	2.5	2.5

Total number of repeat *loci*: 129
Total number of repeat bases: 7224
Total number of repeat copies: 1280.69
Total number of repeat bases excluding base redundancy : 4695

- `catalogue_file_TRAP_complete_table.csv`
- `catalogue_file_TRAP_complete_table_index.html`

These HTML (for web browsers) and CSV (for spreadsheet programs) files list all repeat *loci* grouped according to the repeat unit sequence. In this example, where the parameter `-sort size` was employed, the tables are sorted out according to the period size of the repeat units, in increasing order (see figure below).

Sorted by the period size of the repeats (increasing order)
[1 to 9]
[9 to 30]
[32 to 85]

next page

Total number of bases	Total number of repeat *loci*	Total number of repeat units	Average number of repeat units per *locus*	Period size	Sequence
17	1	17.0	17	1	C
2892	52	936.7	18.01	3	AGC
20	1	6.7	6.7	3	AGA
32	2	8	4	4	GCCC
40	1	10.2	10.2	4	ATTT
35	2	7.2	3.6	5	TTTTA
16	1	3.2	3.2	5	AAATT
17	1	3.2	3.2	5	CGCTG
13	1	2.6	2.6	5	CAAAG
16	1	2.8	2.8	6	TTAAAA
33	2	5.5	2.75	6	CTGCTT
16	1	2.7	2.7	6	TTTTCT
13	1	2.2	2.2	6	ACAGCG
15	1	2.5	2.5	6	CCTCCC
35	1	5.8	5.8	6	GCTGCA
44	1	6.8	6.8	6	AAACCC
648	18	87.1	4.83	7	TAGGGTT
17	1	2.4	2.4	7	TTTCCTT
26	1	3.7	3.7	7	GCTTAGG
16	1	2.3	2.3	7	AAGAAAG
33	1	3.8	3.8	8	TAATTTTT
19	1	2.4	2.4	8	CCCTTTAT
18	1	2.0	2	9	TTTATTTTA
23	1	2.7	2.7	9	AATAAATTG

All repeats are displayed with links, so that clicking on any repeat sequence will open up a new window with a table displaying all loci presenting repeat units with this motif. Relevant information such as the coordinates of each locus and copy number is also displayed (see figure below). The links will not be functional if the

option `-trf` is not used.

Indices	Period Size	Copy Number	Consensus Size	Percent Matches	Percent Indels	Score	A	C	G	T	Consensus_sequence	Prefix Sequence Name
700--719	10	2.0	10	100	0	40	30	30	10	30	ACGCTCTTAA	contig00000333
776--795	10	2.0	10	100	0	40	30	30	10	30	ACGCTCTTAA	contig00000333
816--843	10	2.8	10	100	0	56	29	29	11	32	GCTCTTAAAC	contig00000333

The left column of the table presents the coordinates of each repeat locus and a corresponding link to the output generated by TRF. Thus, clicking on any link will open up a new window displaying the repeat consensus sequence, left and right flanking region sequences (if TRF has been run with parameter `-f`), period size, copy number, sequence of the repeat locus, etc. (see figure below).

```
Alignment explanation


    Indices: 700--719  Score: 40
    Period size: 10  Copynumber: 2.0  Consensus size: 10

        690 TCTGCGGCAG


        700 ACGCTCTTAA
          1 ACGCTCTTAA


        710 ACGCTCTTAA
          1 ACGCTCTTAA

        720 GGGGTTTTTC


Statistics
Matches: 10,  Mismatches: 0, Indels: 0
        1.00            0.00        0.00

Matches are distributed among these distances:
  10   10  1.00

ACGTcount: A:0.30, C:0.30, G:0.10, T:0.30


Consensus pattern (10 bp):
ACGCTCTTAA

Left flanking sequence: Indices 200 -- 699
GGAGCAGCAGCTGAGCCCCCCCGCGCCCCCNAGCAGCAGCAGCAGCAGCAGCTGAGCCCCCC
CGCCCGCCCCCAGCAGCAGCAGCAGCAGCAGCAGGAGCAGCAGCANGCAGCAGCACCGGTCT
CGACGCCGCCGCCGAGGAGCGCGTCGAGGTGGGCGCTGCCAGTGGTGAAGCGGATGAGGTTGCGC
CTGGCGGCGAAGACCTCGTGGGCGGANGCAAANAGCCGAANGCTGCANGAGCTGCCNTGGNCGGC
```

If the period size is longer than 20 bp, a sequence button will be displayed on the table, instead of the nucleotide sequence itself (see figure below).

Indices	Period Size	Copy Number	Consensus Size	Percent Matches	Percent Indels	Score	A	C	G	T	Consensus_sequence	Prefix Sequence Name
4597--5223	21	29.6	21	71	10	483	1	33	21	43	Sequence	contig00000333

In this case, clicking on the sequence button will open up a window presenting the sequence (see figure below).

- `catalogue_file_redundant_regions.html`

This HTML file reports the coordinates of the redundant regions and their respective nested repeats. A screenshot of this file opened on a web browser can be seen below. Notice that a list of the sequences presenting redundant repeats is presented.

Sequence Index	Sequence Description
2	contig00000333
3	contig00000278
4	contig00000040
5	contig00000153

Clicking on any link will open up a window displaying all the redundant (nested) repeats, as shown in the figure below.

Positions of the repeat *loci*

74 to 322
581 to 1022
2557 to 2695
3274 to 3292
3793 to 3811
4382 to 4439
4554 to 5231
5295 to 5334

Redundant region: 74 to 322

Coordinates	Period Size	Copy Number	Consensus Size	Percent Matches	Percent Indels	Score
74 to 322	3	84.3	3	72	9	98
94 to 299	46	4.6	45	91	3	338
122 to 137	4	4.0	4	83	0	25
258 to 273	4	4.0	4	83	0	25

Redundant region: 581 to 1022

Coordinates	Period Size	Copy Number	Consensus Size	Percent Matches	Percent Indels	Score
581 to 833	38	6.7	38	97	0	422
700 to 719	10	2.0	10	100	0	40
776 to 795	10	2.0	10	100	0	40
816 to 843	10	2.8	10	100	0	56
832 to 1022	28	6.8	28	85	2	259
832 to 1008	86	2.1	84	84	2	245

5. Running TRAP with parameters specified in a configuration file:

The same data set and parameters of this tutorial can also be analyzed by TRAP using a configuration file. In this case, you should use only the command line parameter `-c`, which specifies the configuration file name. If you are in the `test` directory, you should use the following command:

```
trap.pl -c ../config_files/catalogue.cnf
```

The `catalogue.cnf` configuration file specifies exactly the same parameters used in the command line described above (see item 2). The configuration file content is listed below:

```
input_prefix = ../trf/tutorial_data.fasta
output_directory = catalogue_dir
output_file_prefix = catalogue_file
minimum_period_size = 1
maximum_period_size =
minimum_copy_number = 2
maximum_copy_number =
define_repeat_sequence =
minimum_flanking_region_size =
minimum_match_percentage = 70
table_format = html+csv
sort_field = size
create_consensus_file = no
create_feature_table = no
create_gff = no
create_flanking_region_file = no
create_trf_like_file = yes
create_redundancy_report = yes
```