Instructions for use of ddbRNA

Diego di Bernardo,Thomas Down, Tim Hubbard

Wellcome Trust Sanger Institute, CB10 1SA, Cambridge, UK

1 How to use ddbRNA

ddbRNA is an algorithm that allows to determine if in a given pairwise or multiple alignment there is a conserved secondary structure.

The input is one or more alignments in FASTA format with gaps included as $-$. The sequence can contain only the symbols A, C, G, T,N, - . Therefore sequences containing the symbol U, will be rejected. U must be replaced with T.

An example of input for one pairwise alignment is:

```
>L_clavatum
GTGGTGCGGTCATACCAGCACTACTAGACCGGATCCCATCAGAACTCCGAA
GTTAAGCGTGCTTGGGCCTGAATAGTACTGGGATGGGTGACCTCCCGGGAAGTCC
GGGTGCCGCACCCTC
>G_biloba
G-GGTGCGATCATACCAGCGTTAATGCACCGGATCCCATCAGAACTCCGCA
GTTAAGCACGCTTGGGCTGGAGTAGTACTAGGATGGGTGACCTCCTGGGAAGTCC
CAGTGTTGCACCCTC
```

To run the algorithm a JAVA 2 runtime environment, version 1.2 or later is needed. The command line is:

```
java -jar ddbRNA.jar [options] sequencefile.fasta
```

The options are:

-alignorder $< n >$ Number of sequences per alignment (default 2)

-window $< n >$ Window size (default 6)

-matches $< n >$ Required matches per window (default 5)

-shuffle $< n >$ Number of shufflings to calculate background signal (default 20)

-threshold $< n >$ Threshold for reporting an RNA (default 3.0)

-stripgaps Strip out all gaps from the alignment (default)

-nostripgaps Preserve gaps in the alignment

-verbose Prints location of compensatory mutations

The output with default options for the alignment in the previous example is:

```
L_clavatum    G_biloba     11.0    1.6 (+/- 1.855)    RNA
```

Where fields (1) and (2) are the identifiers of the two sequences; field (3) is the number of compensatory mutations found in the alignment; field (4) is the average number of compensatory mutations in the shuffled alignment; field (5) is the standard deviation; field (6) is the classification: 'RNA' if a conserved secondary structure is detected, 'OTH' otherwise.

As a default, an alignment is classified as RNA if the number of compensatory mutation (field 3 in the output) is greater than the average number of compensatory mutations in the shuffled alignment (field 4) plus three times the standard deviation (field 5). This threshold value can be changed with the option $-threshold$.

If more than one pairwise alignment is given as input, aReNA will read two sequences a time (i.e. one pairwise alignment) and report the results for all the alignments. An error will be reported if the number of sequences in not a multiple of the alignment order (pairwise, three way, etc. as specified by the $-alignorder$ option).

1.1 Three way alignment mode

The default options must be modified as follows to run the program in this mode:

```
java -jar ddbRNA.jar -alignorder 3 -threshold 4 sequencefile.fasta
```

An example of input for a three way alignment is:

```
>C_zeylanoi
--G-GTT-GCGGCCATATCTA-G-C-AG-AAAGCACCGTTTCCCGTCC-GATCAACTGTA
GTTAAGCTGCTAAGAGCGAG-ACCGAGTAGTGTAGTGGGAGACC-ATACGCGAAACTC--
-T---CGTGCTGCAATCT--
>P_irregula
--A-GCT-ACGGCCATACATA-G-A-TG-AAAATACCGGATCCCGTCC-GATCTCCGA-A
GTCAAGCATCTAATGGCGAC-GT-CAGTACTGTGATGGGGGACC-GCACGGGAATACG--
-T---CGTGCTGTAGTT---
>B_vorax
--G-TTA-TCGGCCATACTAA-G-C-CA-AAAGCACCGGATCCCATTC-GAACTCCGA-A
GTTAAGCGGCTTAAGGCATG-GT-TAGTACTAAGGTGGGGGACC-GCTTGGGAAGCCC--
-A---TGTGCTGATAGCTT-
```

The output for this example is:

```
C_zeylanoi   P_irregtla   B_vorax 26.0     0.6 (+/- 1.319) RNA
```

1.2 Verbose option

This option prints out the location of the compensatory mutations identified by the algorithm in the alignment.
For the pairwise alignment in the previous example the output with the $-verbose$ option is:

```
L_clavattm      G_biloba   11.0    1.5 (+/- 1.5) RNA
CMs: 11 23 25 67 68 91 108 118 122 129 130 | 140
Blocks: [1,120]
```

Where the CMs : line reports the location of compensatory mutations in the alignment. The $Blocks$ line reports the probable start and end of the secondary structure in the alignment using the information on the location of the CMs. To compute this boundaries a simple moving average algorithm is used. Please observe that the acciracy of this feature has not been tested.

1.3 Formatting the input

We included a Perl script written by Elena Rivas [BMC Bioinformatics. 2001;2(1):8] to convert blast hits to Fasta alignment.

The usage is:

```
blastn2qrna.pl min_id min_len max_eval <file_blast_result>
```

Where min_id is the minimum identity of blast hits allowed; min_len id the minimum length and max_eval the maximum evalue allowed.

This script uses two additional scripts (blast2qrna.pl and selectblasthits.pl), therefore in the main script (blast**n**2qrna.pl) the variable $script must be modified to specify the directory where the Perl scripts are stored.