Sequence analysis

G4HunterApps

Laurent Lacroix*

IBENS, ENS-CNRS UMR8197, Inserm U1024, Ecole Normale Supérieure, 75230 Paris Cedex 05, France

*To whom correspondence should be addressed.

Associate Editor: John Hancock

Received on September 25, 2018; revised on November 12, 2018; editorial decision on November 14, 2018; accepted on November 15, 2018

Abstract

Motivation: In order to help G4Hunter users and make it more accessible, I have developed a set of small applications within the Shiny/R framework.

Results: Each application fulfills simple tasks ranging from computing the G4Hunter score for a sequence or a list of sequences to extracting sequences with a G4Hunter score above a threshold for a sequence up to 5 Mb or a list of short sequences. The application can be installed either on the user computer within Rstudio or on a Rstudio server.

Availability and implementation: The source code for the ShinyApps is available on GitHub (https://github.com/LacroixLaurent).

Contact: laurent.lacroix@inserm.fr

1 Introduction

Nucleic acid sequences not only carry genetic information, but can also adopt various structures that go beyond double-helical DNA or stem/loop RNA combinations. These shapes and their dynamics could code for another level of genetic/genomic information. More and more light has been recently shed on alternative or unusual nucleic acid structures as sequences prone to ‘unorthodox’ conformations are proposed to have nucleic acid-related functions (Bacolla et al., 2004; Kouzine et al., 2008). Guanine quadruplexes (G4) are a family of alternative nucleic acid structures that have attracted attention because of their high structural stability under physiological conditions (Davis, 2004) and the widespread distribution of sequences compatible with G4 formation (Hansel-Hertsch et al., 2017; Maizels, 2012). Many recent papers also point towards biological effects that are, or could be, mediated through G4 formation (Maizels, 2015; Rhodes and Lipps, 2015). For a long time, pattern matching algorithms were used to search for genomic sequences able to form such structures (Hoppert and Balasubramanian, 2005; Todd et al., 2005), but other types of predicting algorithm have recently become available (Beaudoin et al., 2014; Garant et al., 2018; Sahakyan et al., 2017; Varizhuk et al., 2017). We have previously developed G4Hunter, one such algorithm (Bedrat et al., 2016). G4Hunter can re-evaluate the widespread occurrence of G4 forming sequences in various genomes in addition to mapping potential G4s that eluded other algorithms. In the original paper (Bedrat et al., 2016), we published the R-code for the algorithm allowing anyone to use G4Hunter. But I came to realize that many people are reluctant to code or might need G4Hunter for a simple task; such as calculating the G4Hunter score for an oligonucleotide or retrieving the G4Hunter predicted hits for their favourite gene or transcript. Therefore, I have developed small applications in the Shiny/R framework that allow such tasks with a user-friendly interface. For whole genomes scans, scripts are available within the original publication for G4Hunter (Bedrat et al., 2016).

2 Materials and methods

Fours applications have been published in Shiny/R starting from the scripts published (Bedrat et al., 2016). All apps require the Shiny library. Additional libraries required are: Xvector, Biostrings (Huber et al., 2015) and GenomicRanges (Lawrence et al., 2013).

All source code are available on GitHub (https://github.com/LacroixLaurent) and can be run on a personal computer via RStudio (https://www.rstudio.com/). Setting up the apps on an Rstudio server is out of the scope of this article.

3 Results

3.1 G4HunterScore

This app simply computes the G4Hunter score of a sequence using the published rule (Bedrat et al., 2016). Letters different from G or C are not translated and have a score of 0. Spaces are automatically removed. The app also reports the length of the sequence.
3.2 G4HunterTable
The G4Hunter table app provides a way to compute G4Hunter scores for a list of sequences either in a text format (one sequence per line) or in a multifasta format (one sequence per fasta entry). In the case of text-type entry, the app tolerates the presence of a header (first line) that can be removed with the ‘Remove the header’ option. Finally, results are reported in table form that can be exported as tab-separated values with three columns: the sequence, the G4Hunter scores and the lengths.

3.3 G4HunterSeeker
The top part of the application reuses the G4HunterScore app in order to allow the user to quickly check the G4Hunter score of a given sequence within the others apps without starting the G4HunterScore app.

Users can either type or paste a sequence (manual entry) or upload a fasta file containing the sequence (fasta file entry). Users also need to specify the sequence type (DNA or RNA alphabet) in order to properly import the sequence. The Threshold and Window size determine the parameters for the sequence search as described in the original publication. The higher the threshold, the more stringent the search: fewer G4 motifs will be found, but these will be the most stable/likely ones.

The procedure extracts sequences that have a G4Hunter score above the threshold (in absolute value) in a window, fuses the overlapping sequences and then refines theses sequences by removing bases at the extremities that are not G for sequences with a positive score (or C for the negative ones). It also looks at the first neighbouring base and adds it to the sequence if it is a G for sequences with a positive score (C for sequences with a negative score). Please see the original G4Hunter publication (Bedrat et al., 2016) and in particular Supplementary Figure S1B for more details.

The result is typically a table reporting: (i) the name of the input sequence (seqnames), (ii) the start, end and width of the hit (after refining the sequence as explained above), (iii) the strand of the hit (the strand is (+) if the proposed G4 forming sequence is in the Input Sequence and (−) if the G4 forming sequence is in the reverse complementary strand). The score is the G4Hunter score of the refined hit and max_score is the highest score in absolute value in a window of the chosen window size found during the search. The threshold and window are respectively the Threshold and Window size used for the search. The sequence corresponds to the refined sequence in the Input sequence. This field is sensitive to the Report G4 Hunter option (see below) and is not present if the Report sequences option is not selected.

The first line of the fasta file (after the > sign) imposes the sequence name (seqname) in the output. This can be changed by checking the Alternate Seqname option and entering the desired sequence name in the New Seqname option. The Report G4 Hunter option changes sequences with a negative score (G-rich sequences) into their reverse complement. Thus the output reports only G-rich sequences. The Number of Hits reports the number of sequences that match the settings. The Length of the Input Sequence corresponds to the length of the DNA sequence you enter with your fasta file or manually. The output can be exported to a text file (tab-separated values) that can be directly opened with Microsoft Excel. There is no limit for the size of the input sequence using the manual entry option and sequences up to several megabases are well tolerated by the app. For fasta file entry, there is a limit set by the default settings of the shiny environment: the file can be up to 5 Mb. However, this option can be modified (see the README file from the source code for details).

3.4 G4HunterMultiFastaSeeker
This app is very similar to the G4HunterSeeker app but allows the user to import multifasta files to perform the search on several input sequences. The limit in size for the fasta file is 5 Mb. If the number of entries in your multifasta file is large (>1000), the app will run significantly slower. You might need to use R outside of the Shiny interface. To avoid unwanted long computation times, you need to click on the button ‘Please click here to start the computation’ to start the G4Hunter process after uploading your file and checking the number of fasta entries. Up to 1000 entries result in a typical computing time below 1 min but for more than a few thousands entries, users might have to wait a few minutes. Options (Threshold, Window Size, Report sequences and Report G-sequences) are similar to the G4Hunter Seeker app. The output table has a unique name for each hit (hitnames) corresponding to the concatenation of the target entry name (from the fasta file) and the start of the hit in this target. The output can be exported to a text file (tab-separated values).

Acknowledgements
I thank J. L. Merigy, P. Martin, G. Micas, A. Sahakyan, D. Morgan and colleagues from Olivier Hyrien’s lab (IBENS, Paris) for discussion and V. Besic for English improvement. The app G4HunterSeeker was kindly made available through a web server by M. Dunning and A. Pajon (CRUK, Cambridge, UK): https://bioinformatics.cruk.cam.ac.uk/G4Hunter.

Funding
This work has been supported by Inserm. Funding for open access charge: The Agence Nationale de la Recherche [ANR-15-CE12-0011-01].

Conflict of Interest: none declared.

References
Bacolla, A. et al. (2004) Breakpoints of gross deletions coincide with non-B DNA conformations. Proc. Natl. Acad. Sci. USA, 101, 14162–14167.
Beaudoin, J.D. et al. (2014) New scoring system to identify RNA G-quadruplex folding. Nucleic Acids Res., 42, 1209–1223.
Bedrat, A. et al. (2016) Re-evaluation of G-quadruplex propensity with G4Hunter. Nucleic Acids Res., 44, 1746–1759.
Davis, J.T. (2004) G-quartets 40 years later: from 5’-GMP to molecular biology and supramolecular chemistry. Angew. Chem. Int. Ed. Engl., 43, 668–698.
Garant, J.M. et al. (2018) GHRNA screenr web server: user focused interface for RNA G-quadruplex prediction. Biochimie, 151, 115–118.
Hansel-Herrtsch, R. et al. (2017) DNA G-quadruplexes in the human genome: detection, functions and therapeutic potential. Nat. Rev. Mol. Cell Biol., 18, 279–284.
Huber, W. et al. (2015) Orchestrating high-throughput genomic analysis with Bioco conductor. Nat. Methods, 12, 115–121.
Huppert, J.L. and Balasubramanian, S. (2005) Prevalence of quadruplexes in the human genome. Nucleic Acids Res., 33, 2908–2916.
Kouzine, F. et al. (2008) The functional response of upstream DNA to dynamic supercoiling in vivo. Nat. Struct. Mol. Biol., 15, 146–154.
Lawrence, M. et al. (2013) Software for computing and annotating genomic ranges. PLoS Comput. Biol., 9, e1003118.
Mazels, N. (2012) G4 motifs in human genes. Ann. N. Y. Acad. Sci., 1267, 53–60.
Mazels, N. (2015) G4-associated human diseases. EMBO Rep., 16, 910–922.
Rhodes, D. and Lipps, H.J. (2015) G-quadruplexes and their regulatory roles in biology. Nucleic Acids Res., 43, 8627–8637.
Sahakyan, A.B. et al. (2017) Machine learning model for sequence-driven DNA G-quadruplex formation. Sci. Rep., 7, 14535.
Todd, A.K. et al. (2005) Highly prevalent putative quadruplex sequence motifs in human DNA. Nucleic Acids Res., 33, 2901–2907.
Varizhuk, A. et al. (2017) The expanding repertoire of G4 DNA structures. Biochimie, 135, 54–62.