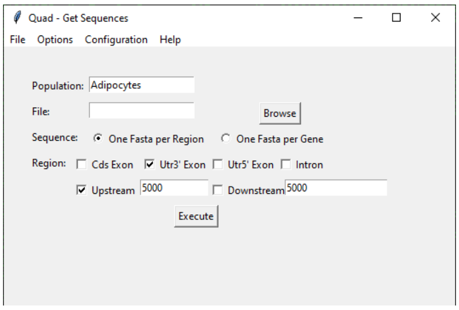
Quad Manual

Quad is a software created in order to find G-quadruplexes forming regions within a set of transcript sequences. It uses transcript IDs as in RefSeq annotation and it searches for their sequence from *Genome Browser* Database (https://genome.ucsc.edu). The program selects portions of these sequences, which can possibly form a G-quadruplex structure, giving them a stability score.

Two different preexisting algorithms were included in this software: *QGRS* and *PQS*. The first one, created by Ramapo group, is a software which has two versions: the first one working locally on computer (https://github.com/freezer333/qgrs-cpp), accessible by terminal, and the second being a web-based one, whose link is http://bioinformatics.ramapo.edu/QGRS/index.php. PQS is a R-library distributed by Bioconductor, which implements the previous algorithm and is imperfection tolerant. In order to be used from *Quad* window, a R compiler needs to be present on Computer and the library must be discharged as explained at the following link: https://bioconductor.org/packages/release/bioc/html/pqsfinder.html. By using *Quad*, it is possible to choose between QGRS local version, QGRS web version (QGRS Mapper) or PQS (pqsfinder). The scoring depends on the chosen algorithm, so that values are not comparable one to another. In this regard, the same sequence with strong stability should return high values in both scales, in case the research is done under similar options. At the time of *Quad* creation, QGRS Web-based tool did not work with multiple sequences in a batch analysis. This led to the programming of a new software which was able to easily solve this problem. Since both algorithms work on a single input sequence, a function, which does a batch research of sequences corresponding to a set of transcripts, was added to the software. Some *Python* libraries were included: *Pandas* (https://pandas.pydata.org), which manages large tables, *Matplotlib* (https://matplotlib.org) to plot collected data and *Numpy* (https://numpy.org) for statistical calculations.

The *Get Sequences* function takes transcript names from the first column of a *csv*-file and submits them to *Genome Browser*. This is a tool which manages accession to a Sequence Database from NCBI and allows discharging separate *fasta*-files for different regions of the searched transcript. This possibility was maintained in Quad. This website is accessible by a software under two conditions: a maximum of one hit every 15 seconds and no more than 5,000 hits per day. The *Get Sequences* function uses a *http request* to database and saves sequences in a local data structure. The information on sequence from Genome Browser can be found using RNA ID and other three missing data: the corresponding chromosome, the start position of the sequence and its end position. The function automatically navigates the websites, analyzes *html* pages using a regular expression method in order to find such missing information and it returns the *fasta*-files, following the chosen conditions. A html page can be thought as a segment, whose points correspond to characters: the method finds the position of *Ncbi Ref Seq* annotation and re-elaborates the following *points* to select only the data of interest. These include a cipher after the dot in transcript ID, chromosome name, start and end position of the sequence. *Get Sequences* is implemented to get also alias Chromosomes. By including this information and using search option it constructs the *url* which redirects directly to *fasta* files. The research can be done by searching a single *fasta* file per gene or a single *fasta* file per region, as in the original database tool. It is also possible to search only particular regions (i.e. Exons, Introns) and to set Upstream and Downstream length. The regions of interest can be selected by checking their corresponding boxes, while Upstream and Downstream length must be inserted in a specific text box.



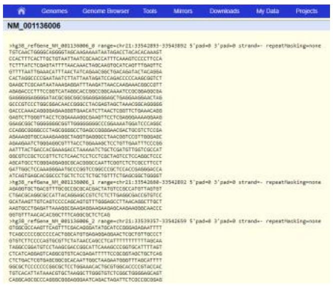
Population: Adipocytes

File:

Sequence: ● One Fasta per Region ○ One Fasta per Gene

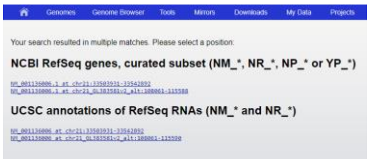
Region: □ Cds Exon ☑ Utr3' Exon □ Utr5' Exon □ Intron

☑ Upstream 5000 □ Downstream 5000



NM_001136006

```
>hg38_refGene_NM_001136006_0 range=chr21:33542893-33543892 5'pad=0 3'pad=0 strand=- repeatMasking=none
[illegible]
>hg38_refGene_NM_001136006_1 range=chr21:33542680-33542892 5'pad=0 3'pad=0 strand=- repeatMasking=none
[illegible]
>hg38_refGene_NM_001136006_2 range=chr21:33539357-33542659 5'pad=0 3'pad=0 strand=- repeatMasking=none
[illegible]
```



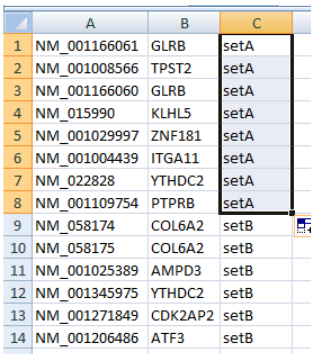
Your search resulted in multiple matches. Please select a position:

NCBI RefSeq genes, curated subset (NM_*, NR_*, NP_* or YP_*)

NM_001136006.1 at chr21:33503931-33542892
NM_001136006.1 at chr21_GL383581v2_alt:108061-115580

UCSC annotations of RefSeq RNAs (NM_* and NR_*)

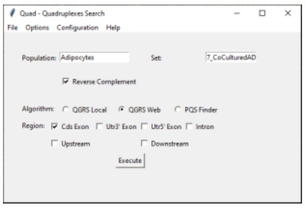
NM_001136006 at chr21:33503931-33542892
NM_001136006 at chr21_GL383581v2_alt:108061-115580



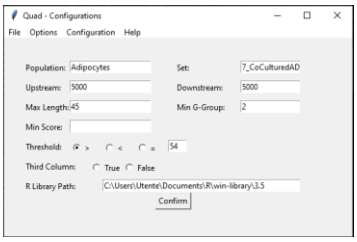
	A	B	C
1	NM_001166061	GLRB	setA
2	NM_001008566	TPST2	setA
3	NM_001166060	GLRB	setA
4	NM_015990	KLHL5	setA
5	NM_001029997	ZNF181	setA
6	NM_001004439	ITGA11	setA
7	NM_022828	YTHDC2	setA
8	NM_001109754	PTPRB	setA
9	NM_058174	COL6A2	setB
10	NM_058175	COL6A2	setB
11	NM_001025389	AMPD3	setB
12	NM_001345975	YTHDC2	setB
13	NM_001271849	CDK2AP2	setB
14	NM_001206486	ATF3	setB

Every transcript list is labeled as *Set* by the program, while the ensemble of sets related one to another are called *Population* group. Input IDs must be saved in a csv file with two or three columns. The first one contains transcript codes (e.g. NM_0015990), the second one its corresponding gene-name. The optional third field is useful when genes belong to different gene sets and a single run for all sets is wanted. Otherwise, *csv*-filename will be regarded as *set* name by the program. The input file structure must be declared in the configuration window by selecting a boolean value for the variable *third column*: True, in case sets are wired in the file, False, if not. The *Get Sequences* function creates a data structure and saves all *fasta* files in the *Fasta* directory. It also transforms them into text files, containing only the sequence, and stores these into the *Text* folder. In case some sequences are missing, it is possible to generate a *csv* file with their IDs, by running the *Get Lost Sequences* option. It creates, in the *data* directory, a new folder called *Lost*, whose subdirectories represent *set* names. The option stores these files, identified by a prefix ,*"lost_"*, followed by the set name. If interested in RNA sequence, which is the reverse complement to its corresponding DNA discharged from database, the option *Reverse Complement* calculates them.

The *Quadruplex search* function opens all the *text* files in the *region* directory and submits them to the selected algorithm, returning a *csv* file for each *text* file in input. All outputs are saved into a directory homonymous with the algorithm name. Algorithm and regions can be chosen in specific text boxes, such as *population* and *set*. In order to search putative G-structures in the RNA sequence, there is a specific check box named *Reverse Complement*. Original algorithms make it possible to choose minimum number of Guanines in the G-repeat, minimum length and other parameters. *Quad*, as a choice, maintained default options except for the maximum length and the minimal number of Guanines in each group, for QGRS, and the minimal score, for PQS. They are settable in configuration windows.



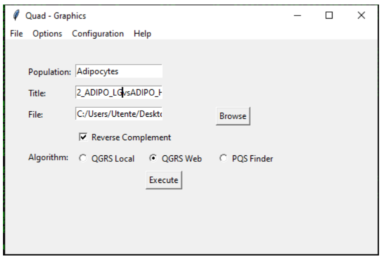
Population: Adipocytes
Set: 7_CoCulturedAD
☑ Reverse Complement
Algorithm: ○ QGRS Local ● QGRS Web ○ PQS Finder
Region: ☑ Cds Exon □ Utr3' Exon □ Utr5' Exon □ Intron
□ Upstream □ Downstream



Population: Adipocytes
Set: 7_CoCulturedAD
Upstream: 5000
Downstream: 5000
Max Length: 45
Min G-Group: 2
Min Score:
Threshold: ● > ○ < ○ = 54
Third Column: ○ True ○ False
R Library Path: C:\Users\Utente\Documents\R\win-library\3.5

Indeed, before running any options, it is suggested checking configuration window to see if some data are missing. In this frame, it is also possible to set *population* name, *set* name, Upstream and Downstream length, G-quadruplexes *Max length* and *minimal* G repeats in G-Group, if using Qgrs algorithm, or minimal score when executing PQS algorithm. Since PQS is a R-library, it must be discharged from the official Bioconductor website and its path inserted in the corresponding box, because the path may change, being based on the operating system.

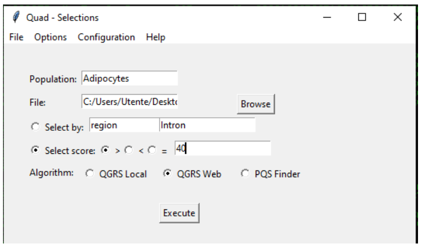
QGRS Web, during the execution, could provide a dimension error over some sequences. The corresponding *text* filename and the date of the research are automatically written down onto the *log* file named "*qgrsweb_error_yyyymmdd.log*". This problem can be handled by the *Errors Management* option, which consists of two functions: *Split Large Files* and *Union Splitted Files*. The *Split Large Files* function divides large sequences into half and submits them to the searching tools. Not to lose possible G-quadruplexes over the split, an overlap is maintained. Its length is one half of the maximum length searched, as specified in the configuration frame. All results are saved in *QGRSWebSplitted* folder, while split sequences are stored in *TextSplitted* directory. In case the same error appears in split sequences, it is sufficient to execute the function again, until no error is generated. A message box will declare whether all sequences are submitted to the method. The *Union Splitted Files* option reunites all *csv* created by the last function into a single file cleaning the overlaps and it recalculates the correct quadruplex position in the original sequence. All reunion files are saved in the basic QGRS Web folder.



Population: Adipocytes
Title: 2_ADIPO_LGvsADIPO_H
File: C:/Users/Utente/Deskt
☑ Reverse Complement
Algorithm: ○ QGRS Local ● QGRS Web ○ PQS Finder

Quadruplex Distribution enables a single view over the whole transcript analyzed: it saves all partial *csv* from all genes into one file for each set and puts all set distributions, belonging to the same population, into a single larger file. The files will have *dist_<Algorithm name>_* as prefix and the date as suffix. In order to see a graph representing the frequency of each score found during the analysis, there is *Graphs* option which requires only the file selection and the algorithm declaration. The *Graph* title is then generated automatically, based on filename, but it can be changed.

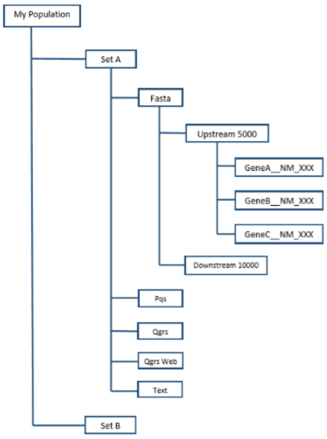
The distribution files may be too large, so that common spreadsheets might not be able to open them. For this reason, the software includes *Select* option, which makes it possible to choose by field or by score. For instance, it is possible to create a file containing only G-quadruplexes with scores greater than a selected threshold or G-structures found only in one type of region. In case some genes are present in different sets of the same population, *Get Intersection* creates a distribution file containing Quadruplex data referred only to the intersection transcripts and to a file which specifies the genes present in multiple sets.



Population: Adipocytes
File: C:/Users/Utente/Deskt
○ Select by: region Intron
● Select score: ● > ○ < ○ = 40
Algorithm: ○ QGRS Local ● QGRS Web ○ PQS Finder

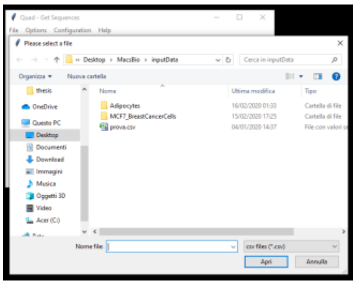
Quad database.

The Database is wired in the file system. Quad organizes data hierarchically. The top of hierarchy is the population directory. The next level maintains different gene sets in different subdirectories. Some data are stored in *csv*-files, while others are saved in *fasta* or *text* files. Files containing information about the same nucleotidic sequence are divided into separate nodes, according to their file type. For example, *fasta* files regarding set A is in “population1 > setA > Fasta”. The next level represents genomic regions. The last folder relates to a single transcript and provides information on its corresponding gene name. Such filenames contain gene identifier and transcript ID, followed by some letters and a number which specifies the region. Cds Exons are labeled as *c*, Introns as *i*, *f* stands for Utr Exon 5’ and *t* for Utr Exon 3’. Upstream and Downstream are named respectively as *u* and *d*. In case multiple region types are selected before execution, a combination of these six letters is added to the filename. Instead, when all sequences are discharged in a single run, the resulting directory is labeled as *FullSeq* and *full* suffix is added to the filename. The numbers refer to a specific piece of sequence and they are counted in a zero-based system. As an example the second intron sequence for NM_016228 transcript is stored in *../Text/Intron/AADAT__NM_016228/AADAT__NM_016228___i_1.txt*.



Fasta files are discharged directly from *Genome Browser*, while *Text* files are created by Quad in order to contain only the nucleotidic sequence. The three algorithms generate a csv file for each input sequence, saving the output in the last node of hierarchical database. When the *Quadruplex Distribution* function is executed, all *csv* files regarding all transcripts in a particular set are joined in the same table. This is saved in a single *csv* file stored in the *set* folder. Additionally, the function creates a complete *csv* file in the top level folder, with all the information about a specific population.

The tables in the first two levels contain the majority of information which the user could be interested in. Anyway if a specific sequence needs to be checked, it is stored in the last node of *fasta* path or *text* path.



Nome	Ultima modifica	Tipo
Adipocytes	16/02/2020 01:33	Cartella di file
MCF7_BreastCancerCells	15/02/2020 17:25	Cartella di file
prova.csv	04/01/2020 14:37	File con valori s

Each filename, preceded by its own path, declares any information on its content. For instance, the file containing first Cds Exon RNA sequence, from transcript NM_016228 is named as follows: *“AADAT__NM_016228___c_0.txt”*. Its relative path is *“../data/<Population>/<Set>/TextReverseComplement/CdsExon/AADAT__NM_016228/”*. Instead, the distribution files and their subsets do not need their path to specify content. Indeed, the filename of a distribution of putative G-structure in RNA, calculated by QgrsWeb algorithm, is: *“dist_QgrsWebReverseComplement_<Population>_<set>_yyyymmdd.csv”*. After running *Selection* option on this file, it adds a prefix to its name, which indicates how the selection has been executed. Some options require a file selection: by pressing, the *Browse button* directly opens the suited folder for that option. The file menu has an *Open* option which facilitates accession to database. At any moment, this guide can be opened by clicking *Help* and selecting *Contents*.