Quadron: predictor of sequence-driven genomic DNA quadruplex formation propensity

The Quadron methodology, developed via gradient boosting machines trained on the experimental G4-seq data (**Balasubramanian Laboratory**), is described and validated in detail at http://dx.doi.org/ToBePublished.

Installation

1. Install the latest version of R programming language or skip to the next step. Step-wise instructions on R installation and upgrade can be found through the following links: 1, 2.

2. Launch R from the command line and install the following R packages, including *shiny* (required for the graphical user interface), *doMC*, *foreach* and *itertools* (required for a parallel execution of the program) from within R.

```
$ R
```

```
> install.packages("plyr")
> install.packages("data.table")
> install.packages("caret")
> install.packages("shiny")
> install.packages("doMC")
> install.packages("foreach")
> install.packages("itertools")
```

3. Download the Quadron source code from the GitHub repository or its interfacing home page. You can also do that via a Linux/Unix/OSX command line, given that git is installed, by typing the following:

```
$ git clone https://github.com/aleksahak/Quadron
```

The downloaded folder has the following content:

- **lib/** - the subfolder containing all the source files,
- **QuadronGUI/** - the subfolder containing the graphical user interface,
- **Quadron.R** - the interfacing R script used to execute Quadron from command line.

4. Install the specific, 0.4-4, version of *xgboost* R library. This specific version is required since there has been a compatibility breach in mapping the model tree structures in subsequent versions. The easiest way to install this version for R on Linux operating system, with all the necessary compilers available, is through the source code distributed through the native R CRAN archive. To do so, launch R and type the following command:

```
$ R
```

```
# For Linux:
> install.packages(
    "http://cran.r-project.org/src/contrib/Archive/xgboost/xgboost_0.4-4.tar.gz",
    repos=NULL,
    type="source")
```

Theoretically, the above command should work with MacOSX as well, however, some versions of the clang compiler in MacOSX do not support the fopenmp compiler keyword, hence installation from binaries is a safer routh. The pre-compiled binaries of *xgboost_0.4-4* are distributed with Quadron (also accessible from Revolution Analytics MRAN repository). They are located in the **lib/xgboost_0.4_4** subdirectory. For the installation, execute the following commands for MacOSX and Windows operating systems respectively (from command prompt, while located in the **lib/xgboost_0.4_4** subdirectory of the downloaded Quadron folder):

```
# For MacOSX:
$ cd lib/xgboost_0.4_4
$ R CMD INSTALL macosx_bin/xgboost_0.4-4.tgz

# For Windows, first launch R:
$ R
```

```
# Next, from within R (for Windows):
> install.packages(
    "windows_bin/xgboost_0.4-4.zip",
    repos=NULL,
    type="binary")
```

5. Finally, you need to bit compile the Quadron package by going into the **lib/** subfolder and executing **bitcompile.R** code from within R.

```
$ cd lib/
$ R
```

```
> source("bitcompile.R")
```

This generates a single file, **Quadron.lib**, which encapsulates the main Quadron code and all its dependencies. At this stage, the subfolder **lib/** can be safely removed. You might, however, want to copy the **test.fasta** file from inside **lib/**, in order to test the Quadron installation.

At this point, the Quadron installation folder should contain:

- **Quadron.lib** - the bit-compiled Quadron program,
- **QuadronGUI/** - the subfolder containing the graphical user interface,
- **Quadron.R** - the interfacing R script used to execute Quadron from command line,

and, if the test sequence file is preserved,

- **test.fasta** - the example DNA sequence fasta file.

Running Quadron from R

Quadron can be executed as an R program, either from within R, or from the Linux/Unix/OSX command line through R CMD BATCH or Rscript execution. The latter two options allow the usage of Quadron from the scripts written via programming languages other than R.

In R, as exemplified in the **Quadron.R** interfacing script, one should load the **Quadron.lib** bit-compiled file, then execute Quadron via the R function *Quadron()*. The latter accepts three arguments:

- *FastaFile* - the relative or absolute path to the fasta file to analyse,
- *OutFile* - the relative or absolute path to the output file to be saved,
- *nCPU* - the number of CPUs to be used for the calculation, where the larger values can markedly speed up the PQS mapping stage of Quadron predictions for large genomes.

Alternatively, the **Quadron.R** file can be edited to set the desired arguments, and executed from the command line via R CMD BATCH or Rscript.

Running Quadron from GUI (Graphical User Interface)

Quadron features a browser-based graphical user interface (GUI), written with Shiny that can be executed locally on as many CPUs as desired. To launch the GUI, enter the **QuadronGUI/** subfolder and double

click on **QuadronGUI** file. If the file fails to open a browser, make sure that the permissions are correctly set for the file (as executable):

```
$ cd QuadronGUI/
$ chmod 700 QuadronGUI # For setting the permissions to open. May require sudo rights.
```

If double clicking on **QuadronGUI** does not launch your browser after the above step, then, most probably, your Rscript utility of R is not installed on the default */usr/bin/Rscript* path. To correct the QuadronGUI setup, first find out the installed path for Rscript by typing the following from the command line:

```
$ which Rscript
```

then open the **QuadronGUI** executable file via a usual plain text editor and correct the Rscript path stated at the first line.

Now double click on **QuadronGUI**. As soon as the browser opens, the rest is self explanatory.

License

You may redistribute the Quadron source code (or its components) and/or modify/translate it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 of the License (GPL2). You can find the details of GPL2 in the following link: http://www.gnu.org/licenses/gpl-2.0.html.

Questions and bug reports to Aleksandr B. Sahakyan via [aleksahak *at* cantab *dot* net].