R/QTL mapping manual (version 1.7)
for internal use at the JIC

Luzie Wingen

Simon Griffiths group

Department of Crop Genetics
John Innes Centre
Norwich Research Park
Norwich
NR4 7UH
UK
Tel. 0044-(0)1603-450508
luzie.wingen@jic.ac.uk

15-Feb-2023

QTL mapping in R (using package qtl and Rstudio)

The statistical software R (see: http://cran.r-project.org/) is used to conduct QTL analysis using the library (R-package) qtl. Tutorials and latest updates are available from http://www.rqtl.org/.

Installation of qtl

This tutorial assumes that you have installed R followed by Rstudio. The qtl package can be downloaded and installed from CRAN (The Comprehensive R Archive Network) in a single step. Follow instructions in Rstudio how to install a missing package. Here a summary of the steps: go to tab Tools, click on Install Packages ..., type qtl into the empty field for packages and click Install. Packages can also be installed by typing install.packages('qtl') into the R console. Depending on how R is set up, a CRAN repository or mirror has to be set. e.g. in R-studio this is done under Tools -> Global options. Most likely, a useful mirror location is pre-set and the installation of the qtl package should run smoothly.

Installation of the jic scripts

After the successful qtl installation copy the zipped directory rqtl_jic_vs1.7.zip to a location on your computer you are likely to find again. Unzip the directory to rqtl_jic_vs1.7. The directory contains two subfolders: rqtl_data and rqtl_out. In the rqtl_data directory the example files ParW246_rqtl_genotypes.csv and ParW246_rqtl_phenotypes.csv will be present.

Preparing data: a genotype and a phenotype file

The data files for the R/qtl analysis have to be in a specific format. The following requirements have to be fulfilled for the preparation of the phenotype and the genotype file.

- Phenotype data:
  - save data in a comma-separated-value spread sheet (csv).
  - for the rqtl_jic scripts, the file name has to contain the keywords '_rqtl_phenotypes.csv' and be otherwise identical to the genotype file name. ParW246_rqtl_phenotypes.csv, for instance, is a valid name.
  - the first row of the data file should start with the key word id (no quotes), followed by the trait names separated by commas (no quotes). The trait names can contain treat or environment additions, separated from the trait by an underscore (e.g. 'Ht_CF2010').
  - the following rows start with variety names in the first column. The names of varieties and the complete set has to be identical to varity names and set in the genotype file.
  - check with the example file in the data directory.
- Genotype and mapping data:
  - save data in a comma-separated-value spread sheet (csv).
  - for the rqtl_jic scripts, the name has to contain the keywords '_rqtl_genotypes.csv' and be otherwise identical to the phenotype file name, e.g. ParW246_rqtl_genotypes.csv would be a valid name.
  - the first row starts with the keyword id (no quotes), followed by the marker names, separated by commas (no quotes).

  - the second row starts with and empty field, followed by linkage group names. Each linkage group have to have a different name. It is a good idea to include the chromosome name in the linkage group name, if possible (e.g. 1Aa, 1Ab, 1Ac, 1B). Linkage groups need to be alphanumerically ordered.
  - the third row starts with an empty field, followed by position numbers. They have to be in numerical order within their linkage group.
  - following rows start with the variety names (identical to those in the phenotypes), followed by the allele status. The genotype coding is: 'A','B','H','-', lower case letters are allowed, but do not mix upper and lower case.
  - check with the example file in the data directory.

- **One file for phenotype, genotype and mapping data together (not well implemented)**
  - save data in a comma-separated-value spread sheet (`csv`).
  - for the `rqtl_jic` scripts the file name has to contain the keywords `'_rqtl_pg.csv'`.
  - orientation is the `R/qtl` format of `'csv'`, which is quite similar to the single files above, but joined up to one file.
  - First column(s) are traits, starting with traitname, followed by two empty fields - for the chromosome and position information further below. At least on phenotype column is needed, more are possible.
  - Following columns are genotype columns, they start with markername, chromosome, position and then the genotype scoring ('A','B','H','-' allowed). Each chromosome/linkage group needs a different name. Linkage groups need to be in alpha-numerical order. Postions need to be numerically ordered within each linkage groupl

Where to put the data files?

The prepared genotype and phenotype files are copied to the subdirectory `rqtl_data`. The analysis pipeline will work on all files present in that folder that match the pattern (e.g. containing `_rqtl_genotypes.csv` in the name). So, all files from previous analyses should be moved to another folder, it can be a folder underneath the data folder, e.g. `rqtl_data/ParW246/` would work. The automatic pipeline only detects files in the data directory, not in sub-directories.

Controlled trait vocabulary

A file containing a list of useful trait abbreviations and trait descriptions, a controlled trait vocabulary, (`field_trial_phenotypes_or_traits.csv`) is also present in the `rqtl_jic` directory together with the `R scripts`. If the phenotype file uses the trait abbreviations from this controlled vocabulary, the more explanatory, longer trait names will be included in the final qtl summary table. The vocabulary file will be updated to contain crop-ontology approved vocabulary in the next version.

Analysis parameters

To allow for many different cross types and other variables, the analysis parameter can be adjusted to individual needs. Some parameters are pre-set, but can be changed by the user. In some cases the programme makes guesses what is best. Parameters can be set in the file `start_qtl_analysis.R`, which can be either opened in `Rstudio` or in a simple editor but should not be edited in `Word`.
Currently, the parameter section looks like this (lines too long for this document carry on in the next line with white space at the beginning) :

```
###################################
### Analysis Parameter settings ###
###################################
fnames <- dir(path=datadir,pattern=paste("^.*","_rqtl_.*",".csv",sep=""),
   full.names=TRUE)
ptindex <- grep(’phenotype’,fnames)
gtindex <- grep(’genotype’,fnames)
gpindex <- grep(’_pg’,fnames) ### joint genotype and phenotype files.
pmethod="pdf"  ###  pmethod="jpeg"
epistasis=F ### epistasis is computational expensive! FALSE by default.
crosstype=NULL ### can be ’dh’,’bc’,’riself’ or ’f2’.
          Set to NULL to use F.gen and BC.gen.
BC.gen=0 ### Back-cross generation. Set BC.gen=NA to use crosstype (above).
F.gen=4 ### F generation. Set F.gen=NA to use crosstype (above).
map=F ### analysis of the map quality - not needed but maybe interesting
redMap=F ### reduces the marker number - for large maps, e.g. iSelect/axiom data
genotypeCodes=c(’A’,’H’,’B’,’-’)### genotype codes in order AA, AB, BB, missing
alpha=0.4 ### analysis significance level 0.4 = 40% to identify CIM co-factors.
alphaCIM=0.05 ### analysis significance level 0.05 = 5%
```

By changing

- `fnames`: change expected filenames for the input data
- `ptindex`: pattern in the names of phenotype files
- `gtindex`: pattern in the names of genotype files
- `pmethod`: set printing method can
- `epistasis`: analysis with epistatic QTL interaction. This needs a lot of computer power. It is by default set to `FALSE`.
- `crosstype`: the type of cross to generate the population. Options for this important parameter are: `’dh’`, `’bc’`, `’f2’` or `’riself’`. For complicated crosses use the parameters below.
- `BC.gen` and `F.gen`: used to define an intermediate or more complicated cross type. E.g. for a F4 generation: `BC.gen=0` and `F.gen=4`. Set both to `NA` for the simple crosstypes (e.g. `’dh’`).
- `map`: switch for a map quality analysis. Normally set to `FALSE`.
- `redMap`: a reduction of the numbers of markers in a map can be switched on. This is very useful for large maps (more than 1000 markers). Normally set to `FALSE`.
- `genotypeCodes`: the encoding of the genotypes can be changed to those present in the genotype file. Normally set to `’A’`, `’B’`, `’H’`, and `’-’`.
- `alpha`: initial and generous significant level for QTL detection (alpha=0.4). The scan will find co-factors for the following CIM.
- `alphaCIM`: stringent significant level for composite interval mapping (CIM) (alphaCIM=0.05). This is also the level for non-parametric QTL scans as no second scan is conducted.

More parameters will be added in the future. Please feed back if a specific parameter would be useful.

Start analysis

The analysis can be started interactively from a R console, e.g. in Rstudio. Make sure your working directory is the rqtl_jic_vs1.7 directory. If you are unsure, ask with getwd() for your current working directory. You can set the directory to a different directory with the command setwd(), e.g. setwd("C:/Users/username/Desktop/rqtl_jic_vs1.7/". Check with getwd() if the directory is now correct. Start the analysis by typing source("start_qtl_analysis.R").

Run as a batch command

Alternatively, under Linux or in the Windows command window, the QTL analysis can be started using the R CMD BATCH command line option. If the batch option is chosen, the short script qtl_batch_analysis.R can be called. This script contains the above R command to start the analysis. It will write the output to file, which is useful to document your analysis.
To start this script from the Linux command line, type into a terminal window: R CMD BATCH -no-save -no-restore-data qtl_batch_analysis.R The analysis should run through and the log should be written into the qtl_batch_analysis.Rout file.

Output

Output will be written into the directory rqtl_out/plots and rqtl_out/csv.
The analysis will produce pdf files (rqtl_out/plots) with:

- plots of phenotype histograms (filename contains: _phenotype_histograms_)
- plots of the whole genome QTL scan (filename contains: _qtl_genome_overview_)
  If more than one QTL was found, the initial scan is repeated with the identified QTL as co-factors. In this case, the second plot for that trait will show two LOD traces: the trace from the the first scan (using Haley-Knott regression, thus labeled as 'hk') and the second trace using composite interval mapping (keyword CIM).
- plots of individual chromsomes with QTLs as CIM scan (filename contains: _qtls_)
- plots of QTL positions shown on a simple map (filename contains: _qtl_map_location_)
- plots of QTL effects (filename contains: _qtl_additive_effects_)

A QTL summary table will be written as csv spread sheet into directory rqtl_out/csv (filename contains qtl_table_). Moreover, in subdirectory lodfiles the QTL traces for each trait and for both QTL scans (first hk scan, second CIM scan) are written into comma-seperated-spread sheets.

Feedback

I'm happy to receive feedback. Please, acknowlege the use of these scripts.
Luzie Wingen (luzie.wingen@jic.ac.uk)