postQTL: a QTL mapping R workflow to improve the accuracy of true positive loci identification

Prashant Bhandari¹ and Tong Geon Lee¹,²,³,⁴*

Abstract

Objective: The determination of the location of quantitative trait loci (QTL) (i.e., QTL mapping) is essential for identifying new genes. Various statistical methods are being incorporated into different QTL mapping functions. However, statistical errors and limitations may often occur in QTL mapping, implying the risk of false positive errors and/or failing to detect a true positive QTL effect. We simulated the power to detect four simulated QTL in tomato using cim() and stepwiseqtl(), widely adopted QTL mapping functions, and QTL.gCIMapping(), a derivative of the composite interval mapping method. While there is general agreement that those three functions identified simulated QTL, missing or false positive QTL were observed, which were prevalent when more realistic data (such as smaller population size) were provided.

Results: To address this issue, we developed postQTL, a QTL mapping R workflow that incorporates (i) both cim() and stepwiseqtl(), (ii) widely used R packages developed for model selection, and (iii) automation to increase the accuracy, efficiency, and accessibility of QTL mapping. QTL mapping experiments on tomato F₂ populations in which QTL effects were simulated or calculated showed advantages of postQTL in QTL detection.

Keywords: QTL, Mapping, Model search, Regularization, R workflow

Introduction

Quantitative trait loci (QTL) affect the phenotypic variation observed in quantitative characters. Quantitative traits (character) display a continuous distribution of phenotypes, and many agronomically important traits (e.g., yield) belong to quantitative traits. Therefore, QTL mapping, the determination of the location of QTL, is one of the first steps to build a scientific basis for plant genetics and breeding.

Various statistical methods have been used for QTL mapping (e.g., Analysis of variance (ANOVA) [1], interval mapping (IM) [2], composite interval mapping (CIM) [3–6], multiple interval mapping (MIM) [7], Bayesian methods [8], and genetic algorithms [9]). Among these, the CIM method, which uses genetic marker covariates to detect QTL, is widely adopted in plants for QTL models [10]. Several more recent derivatives of the original CIM are also available (e.g., inclusive composite interval mapping (ICIM) [11] and genome-wide composite interval mapping (GCIM) [12, 13]).

Broman and Speed [14] stated that the key problem with CIM is the choice of the set of markers to use as regressors, inevitably affecting the power for QTL detection (e.g., increasing the variance of the logarithm of the odds (LOD) score). Likewise, QTL mapping is a model selection problem closely concordant with controlling the inclusion of extraneous loci while identifying as many true QTL as possible [14, 15]. While an exhaustive model search would be the ideal solution to the problem,
dramatically increasing computational demands by consolidating different models/large genotypic datasets are less clearly advantageous to non-specialist researchers such as plant breeders. In QTL mapping, forward selection or some version of stepwise selection is commonly used because of its computational efficiency and a close approximation of the best subset selection [14–16]. Recently, an R/qtl function stepewiseqtl() with a forward/backward stepwise search algorithm was implemented in the R/qtl package (note stepwiseqtl() was added to R/qtl version 1.09 and later [15]).

Regularization methods (e.g., minimax concave penalty (MCP) [17], least absolute shrinkage and selection operator (LASSO) [18]), have been used in statistics and machine learning for feature selection (e.g., adopted in genomic prediction [19, 20]). Practical implications of such methods for QTL mapping would help us to select a true QTL by distinguishing a true QTL effect from extraneous loci-driven noise (e.g., regression coefficients for a given marker shrink to near 0 where a model fits).

Given the above information, a comprehensive approach integrating the results from both mapping functions, cim() (note that a version of the CIM method is implemented into the R/qtl and stepewiseqtl(), and regularization method(s) may be as good as a single function-driven QTL mapping, or an optimal strategy to improve the accuracy of true positive QTL identification. Here, we present postQTL, a QTL mapping R workflow that incorporates (i) both cim() and stepewiseqtl(), (ii) widely used R packages developed for model selection, and (iii) automation to increase the accuracy, efficiency, and accessibility of QTL mapping. postQTL is an R script that executes several R function(s) and R packages connected by the R language. This package has been developed to be straightforward for use in a typical R environment on Linux/PC hardware. We focus on the tomato, a model fruit crop with relatively large QTL mapping efforts.

Main text

Methods

Simulating QTL mapping data

In this study, we used the R environment (version 4.1.1; [21]) with the R/qtl package (version 1.50; [22]) on both a stand-alone Windows operating PC and the UF/Research Computing Linux server, HiPerGator 3.0 [23]. A genetic map created for tomato (Solanum lycopersicum) [24] was used to simulate two different sized F₂ populations using the R/qtl function sim.cross(): F₂ individuals with 1000 (hereafter referred to as population #1) and 100 (hereafter referred to as population #2) to mimic mapping populations or breeding studies. Four simulated QTL were positioned at 5 cM on each of chromosomes 1, 5, 7, and 12, and simulated to have additive effect sizes of 1.0, 0.5, 0.2, and 0.2, respectively (Additional file 1). The missing data percentage was set at 50% and an error rate of $10^{-4}$ was used. Three QTL mapping functions, cim(), QTL.gCIMapping() (version 3.3.1; an R function implemented into the GCIM method [25]), and stepwiseqtl() of the R/qtl package, were used to map simulated QTL in both populations #1 and #2. For cim(), covariates 3 and 11 were chosen. For each mapping function, 100 iterations were performed and the identified QTL were reported.

postQTL

The postQTL R workflow developed for this project consists of six steps (Fig. 1; Additional file 2). The first step of the workflow loads the postQTL R script and R packages. Recent versions of commonly used R packages were used for postQTL (the full list of R packages required is available at Additional file 3). The postQTL R script executes four function(s), map_qtl(), model_qtl(), regularize_qtl(), and model_chromosome(). Before executing the script, the user loads an input dataset (a single CSV file that carries the genotypic and phenotypic datasets is available at Additional file 1). After loading, postQTL simultaneously uses cim() with a covariate 11 and stepwiseqtl() to identify QTL (here performed by a function map_qtl()). postQTL performs 10 iterations for cim() and a single run for stepwiseqtl(), and then merges the identified QTL. After identifying QTL, postQTL performs an exhaustive model search for markers residing in the identified QTL using R function regsubsets() implemented in the R package ‘leaps’ [26] (performed by model_qtl()), providing information about whether or not identified QTL are included in the best model. Such a model search restriction over the identified QTL reduces computational demands. For the QTL identified by stepwiseqtl(), a function find.marker() was used to find the nearest marker to the QTL. For the QTL by cim(), both a marker at the QTL peak position and one of two markers flanking the QTL peak were used. The user determines the best model(s) once a list of models with ranks is created by postQTL [e.g., choose marker(s) with top Cp value(s); note that postQTL outputs BIC and adjusted R² using an R package ‘leaps’ as optional additional resources for users]. Next, postQTL calculates the regression coefficient ($\beta$) for the representative marker(s) in the identified QTL using regularization methods LASSO and MCP (performed by regularize_qtl()). LASSO and MCP were implemented using the R packages ‘glmnet’ (version 4.1.3; [27]) and ‘picasso’ (version 1.3.1; [28]), respectively. Finally, postQTL performs an exhaustive model search with a fixed model size (a default value of 3 was set) for each chromosome (e.g., all markers reside on chromosome 1) to identify best predictors (performed by
model_chromosome()). Theoretically, a true QTL is more likely to be physically located near or at those predictors. The user can modify arguments listed in postQTL (e.g., chromosome, numberofpredictors) for their QTL mapping project (e.g., a large number of accurate markers vs. a few selected markers). postQTL was tested on a stand-alone Windows operating PC via an RStudio [29] and Linux environment [23] via a terminal emulator.

**QTL mapping of tomato height**

All mapping methods (cim(), QTL.gCIMapping(), and stepwiseqtl()) and postQTL were applied to continuous phenotypic data for the tomato seedling trait. 116 F2 plants randomly selected from a segregating progeny in a population, which is known to show at least two QTL (genes) including the tomato *BRACHYTIC* locus [30] for height based on field evaluations, were chosen. The genotypic (F2 generation genotyped by the tomato Illumina Infinium array and a molecular marker for the tomato *BRACHYTIC* locus; Additional file 4) and phenotypic (F2, F2:3, and F2:4; Additional file 5) datasets were prepared as described in our previous study [30].

**Results**

We simulated four QTL with different effect sizes on four different chromosomes. We compared the QTL mapping results among cim(), QTL.gCIMapping(), and stepwiseqtl() on two differentially sized populations. In simulated population #1 (1000 F2 individuals), cim() exhibited the highest frequency of the identification of all four simulated QTL followed by stepwiseqtl() (LOD score > 3.0; Fig. 2A). However, in simulated population #2 (100 F2 individuals), a lower accuracy of the identification of simulated QTL was observed for all three mapping functions (Fig. 2B). Both QTL.gCIMapping() and stepwiseqtl() missed at least one of simulated QTL, while cim() identified all four simulated QTL in at least 8% of 100 iterations. Expectedly, the overall frequency of identifying potential false positive QTL in population #2 was higher than in population #1 (Fig. 2C and D). Therefore, in the following sections, we demonstrate how postQTL performed a QTL mapping and its post-analysis on population #2.

In simulated population #2, the first function of postQTL, map_qtl(), identified seven QTL on chromosomes 1, 2, 3, 4, 5, 7, and 12 (column named ‘Mapping’ in Table 1).

```
library(postQTL)
# Load data.
myqtldata <- qtl::read.cross (format=c("csvr"), file="qtl_data.csv", genotypes=c("a","b","H"))
# Run a QTL mapping.
map_qtl(myqtldata)
# Run a model search for markers residing in the identified QTL.
model_qtl("inputformodelqtl.csv")
# Calculate the regression coefficient for marker(s) in the identified QTL using regularization methods.
regularize_qtl("inputforregularizeqtl.csv")
# Run a model search for each chromosome.
model_chromosome(myqtldata, chromosome, numberofpredictors)
```

**Fig. 1 An overview of the postQTL R workflow**

...
that chromosome. Additionally, the estimated regression coefficient for the marker SL4.0ch03.3639459 in Q3 was close to 0.00, which gives further information to exclude Q3 from a list of potential true positive QTL.

Finally, postQTL identified the best prediction markers per chromosome by running a model search with a fixed model size of 3 (column named `Marker prediction` in Table 1). Two markers, SL4.0ch01.2691495 and SL4.0ch12.7161, were located in the simulated QTL, Q1 and Q12, respectively. Notably, detection of the marker SL4.0ch12.7161, located in one of the small effect QTL, provided further information to control the inclusion rate of small effect QTL.

To validate that postQTL can be useful to identify potential QTL associated with the plant height trait, we performed genetic mapping analyses using postQTL and previously developed mapping functions, cim(), QTL.gCIMapping(), and stepwiseqtl(). Three QTL signals with significant effects (LOD > 4) were detected from at least two filial generations by at least two analyses: a position near the `BRACHYTIC` fine-mapped on chromosome 1 [30], another position (close to 5.0 Mbp) on chromosome 1 and an approximate 60.0 Mbp on chromosome 7 (Additional file 6). Importantly, postQTL exhibited the QTL signal on chromosome 7 from all three filial generations (F2, F2.3, and F2.4), which indicates this location can be used to investigate desirable alleles that might explain the phenotypic segregation in the tomato population used in this study.

**Discussion**

Despite the availability of multiple methods for QTL mapping, a need exists for a comprehensive approach integrating the results from multiple QTL mapping methods, which may be the optimal strategy to most accurately identify QTL. We developed postQTL, an R workflow that implements two widely adopted QTL mapping functions. We used postQTL primarily in a tomato community, as tested on F2 and other advanced populations. However, postQTL should apply to any (plant) species as long as the QTL mapping functions, cim() and stepwiseqtl(), fit the species of user interest. In the mapping of low genetic effect QTL, missing such QTL is likely to be observed when researchers repeat the mapping analysis with independent imputations. To address this, postQTL has the default number of iterations for cim() as 10.

A critical element of any QTL mapping workflow is its ease of use. postQTL is suitable for both R environment novices and experienced R users. postQTL automates the entire QTL mapping process by requiring only one R workflow. Further, postQTL only requires commonly used R packages in the R program, not requiring additional processing steps outside of the workflow.

Lastly, postQTL includes regularization methods which could be useful supplements to the researcher
when the conclusions on QTL determination can be subject to considerable uncertainty.

**Limitations**

postQTL limits an exhaustive model search for markers residing in the identified QTL and for markers for each chromosome. Clearly, an exhaustive model search...
for all genetic markers incorporated into QTL mapping data should be optimized to maximize computational efficiency.

Abbreviations

ANOVA: Analysis of variance; BIC: Bayesian information criterion; CIM: Composite interval mapping; GCIM: Genome-wide composite interval mapping; ICM: Inclusive composite interval mapping; IM: Interval mapping; LASSO: Least absolute shrinkage and selection operator; LOD: Logarithm of odds; MCP: Minimax concave penalty; MIM: Multiple interval mapping; QTL: Quantitative trait loci.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13104-022-08617-z.

Acknowledgements

Not applicable.

Author contributions

PB has contributed to the conception and design of the work, the acquisition, analysis, and interpretation of data, drafted the work, and approved the submitted version. TGL has contributed to the design of the work, the interpretation, analysis, and interpretation of data, drafted the work, and approved the submitted version. All authors read and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

All data generated or analyzed during this study are included in this published article and its supplementary information files.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

1 Horticultural Sciences Department, University of Florida, Gainesville, FL 32611, USA. 2 Gulf Coast Research and Education Center, University of Florida, Wimauma, FL 33598, USA. 3 Plant Breeders Working Group, University of Florida, Gainesville, FL 32611, USA. 4 Plant Molecular and Cellular Biology Graduate Program, University of Florida, Gainesville, FL 32611, USA.

Received: 22 January 2022 Accepted: 24 March 2022 Published online: 04 May 2022

References

1. Soller M, Brody T, Genizi A. On the power of experimental designs for the detection of linkage between marker loci and quantitative loci in crosses between inbred lines. Theor Appl Genet. 1976;47:35–9.
2. Lander ES, Botstein D. Mapping mendelian factors underlying quantitative traits using RFLP linkage maps. Genetics. 1989;121:185–99.
3. Jansen RC. Interval mapping of multiple quantitative trait loci. Genetics. 1993;135:205–11.
4. Jansen RC, Stam P. High resolution of quantitative traits into multiple loci via interval mapping. Genetics. 1994;136:1447–55.
5. Zeng Z-B. Theoretical basis for separation of multiple linked gene effects in mapping quantitative trait loci. Proc Natl Acad Sci USA. 1993;90:10972–6.
6. Zeng Z-B, Kao C-H, Basten CJ. Estimating the genetic architecture of quantitative traits. Genet Res. 1999;74:279–89.
7. Kao C-H, Zeng Z-B, Teasdale RD. Multiple interval mapping for quantitative trait loci. Genetics. 1999;152:1203–16.
8. Satagopan JM, Yandell BS, Newton MA, Oxborn TC. A Bayesian approach to detect quantitative trait loci using Markov chain Monte Carlo. Genetics. 1996;144:805–16.
9. Carlborg O, Andersson L, Kinghorn B. The use of a genetic algorithm for simultaneous mapping of multiple interacting quantitative trait loci. Genetics. 2000;153:2003–10.
10. Doerge RW. Mapping and analysis of quantitative trait loci in experimental populations. Nat Rev Genet. 2002;2:43–52.
11. Li H, Ye G, Wang J. A modified algorithm for the improvement of composite interval mapping. Genetics. 2007;175:361–74.
12. Wang SB, Wen YJ, Ren WL, Ni YL, Zhang J, Feng JY, Zhang YM. Mapping small-effect and linked quantitative trait loci for complex traits in backcross or DH populations via a multi-locus GWAS methodology. Sci Rep. 2016;6:29951.
13. Wen YJ, Zhang YW, Zhang J, Feng JY, Dunwell JM, Zhang YM. An efficient multi-locus mixed model framework for the detection of small and linked QTLs in F2. Brief Bioinform. 2019;20:1913–24.
14. Broman KW, Speed TP. A model selection approach for the identification of quantitative trait loci in experimental crosses. J R Stat Soc Ser B Stat Methodol. 2002;64:641–56.
15. Broman KW, Wu H, Sen S, Churchill GA. R/qtl: QTL mapping in experimental crosses. Bioinformatics. 2003;19:889–90.
16. Sohild F, Sohali MU, Shabbir J. An introduction to statistical learning with applications in R: by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. New York: Springer; 2013.
17. Zhang C-H. Penalized linear unbiased selection. Technical Report 2007. Dept. Statistics, Rutgers Univ.
18. Tibshirani R. Regression shrinkage and selection via the lasso. J R Statist Soc B. 1996;58:267–88.
19. Hayes BJ, Bowman PJ, Chamberlain AJ, Goddard ME. Invited review: genomic selection in dairy cattle: progress and challenges. J Dairy Sci. 2009;92:433–43.
20. Cossa J, Campos Gde L, Pérez P, Gianola D, Burgueño J, et al. Prediction of genetic values of quantitative traits in plant breeding using pedigree and molecular markers. Genetics. 2010;186:713–24.
21. R Core Team. R: A language and environment for statistical computing. 2020. R Foundation for Statistical Computing, Vienna, Austria.
22. Arends D, Prins P, Jansen RC, Broman KW. R/qtl: high-throughput multiple QTL mapping. Bioinformatics. 2010;26:2990–2.
23. University of Florida. 2022. HiPerGator. https://www.rc.ufl.edu/services/hipergator.
24. Bhandari P, Lee TG. A genetic map and linkage panel for the large-fruited fresh-market tomato. J Am Soc Hortic Sci. 2021;146:125–31.
25. Zhang YW, Wen YJ, Dunwell JM, Zhang YM. QTL.qCIMapping GUI v2.0: an R software for detecting small-effect and linked QTLs for quantitative traits in bi-parental segregation populations. Comput Struct Biotechnol J. 2019;18:59–65.
26. Lumley T. 2020. Leaps: R package version 3.1.
27. Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. J Stat Softw. 2010;33:1–22.
28. Ge J, Li X, Jiang H, Liu H, Zhang T, Wang M, et al. Picasso: a sparse learning library for high dimensional data analysis in R and Python. J Mach Learn Res. 2018;19:1–5.
29. RStudio Team. RStudio: Integrated Development for R. RStudio, PBC, Boston, MA URL http://www.rstudio.com/. 2020.
30. Lee TG, Hutton SF, Shekasteband R. Fine mapping of the brachytic locus on the tomato genome. J Amer Soc Hort Sci. 2018;143:239–47.
31. Fernandez-Pozo N, Menda N, Edwards JD, Saha S, Tecle IY, Strickler SR, Bombarely A, Fisher-York T, Pujar A, Foerster H, Yan A, Mueller LA. The Sol Genomics Network (SGN)—from genotype to phenotype to breeding. Nucleic Acids Res. 2015;43:D1036–41.

Publisher’s Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:
• fast, convenient online submission
• thorough peer review by experienced researchers in your field
• rapid publication on acceptance
• support for research data, including large and complex data types
• gold Open Access which fosters wider collaboration and increased citations
• maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.
Learn more biomedcentral.com/submissions