Contact: coding.plantpath.ksu.edu/qgene. Source code is available on

Availability: map, population, and trait simulators; and is scriptable.

public QTL-mapping packages specialize in subsets of analysis methods and mating designs. They include MapMaker/QTL (Lander and Botstein, 1989), QTLMapper (Wang et al., 1999), Map Manager (Manly and Olson, 1999), pseudomarker (Sen and Churchill, 2001), QTL Express (Seaton et al., 2002), R/qtl (Broman et al., 2003), MCQTL (Jourjon et al., 2004), QTL Cartographer (Wang et al., 2007), GridQTL (Seaton et al., 2006), J/qtl (Wu et al., 2008) and QTLE Network (Yang et al., 2008). Aside from the diversity of input data formats, direct comparison of the outputs of multiple QTL-mapping methods is hard. Often two packages running nominally the same statistical tests on the same data do not produce identical results, while source code that might resolve the difference is closed or is difficult to understand or modify.

1 INTRODUCTION

Many statistical methods have been developed for mapping quantitative trait locus (QTLs) controlling polygenic traits. Best known are interval-mapping (IM) methods: single IM (SIM) (Haley and Knott, 1992; Lander and Botstein, 1989), composite IM (CIM) (Jansen, 1993; Zeng, 1994), multiple IM (MIM) (Kao et al., 1999) and Bayesian IM (BIM) (Satagopan et al., 1996; Sillanpää and Arjas, 1998). Multiple-trait (Jiang and Zeng, 1995; Korol et al., 1995), variance-model (Xu and Atchley, 1995), and shrinkage (penalized maximum-likelihood or PMLE) (Xu, 2003; Zhang and Xu, 2005) methods assume ever more relevance as expression QTL (eQTL) and association-mapping studies proliferate. But the literature is replete with QTL-mapping variants unavailable for practical use simply because they have not been implemented in usable software.

Public QTL-mapping packages specialize in subsets of analysis methods and mating designs. They include MapMaker/QTL (Lander and Botstein, 1989), QTLMapper (Wang et al., 1999), Map Manager (Manly and Olson, 1999), pseudomarker (Sen and Churchill, 2001), QTL Express (Seaton et al., 2002), R/qtl (Broman et al., 2003), MCQTL (Jourjon et al., 2004), QTL Cartographer (Wang et al., 2007), GridQTL (Seaton et al., 2006), J/qtl (Wu et al., 2008) and QTLE Network (Yang et al., 2008). Aside from the diversity of input data formats, direct comparison of the outputs of multiple QTL-mapping methods is hard. Often two packages running nominally the same statistical tests on the same data do not produce identical results, while source code that might resolve the difference is closed or is difficult to understand or modify.

2 DESCRIPTION

QGene 4.0, a full rewrite of an earlier package (Nelson, 1997) in the Java language, is a standalone software application equipped with a rich graphical user interface (GUI), which is intended to be installed locally on any computer supporting Java. For developers, QGene accommodates rapid QTL-analysis method implementation with a plug-in architecture. All plug-ins, besides using QGene’s built-in display machinery, may exploit QGene’s permutation-analysis module, which computes the threshold above which a QTL can be declared significant (Churchill and Doerge, 1994). Plug-ins relying on genotype probability distributions of QTLs and missing or dominant markers (Jiang and Zeng, 1997) can accommodate all mating designs known to QGene, while those employing linear models may exploit residual-normality checking.

QGene mating designs are specified by any sequence of the letters b, d, s and i, representing backcrossing, doubled-haploid creation, selfing and random intercrossing, starting at the F1 generation. Thus b specifies a BC1F1 design, s an F2, ss an F3. Unorthodox designs, such as sbibs are thus properly handled. The mating string is used for run-time construction of genotype probability-distribution matrices corresponding to the specified model. QGene at present operates only on line-cross designs.

Extension of conventional QTL software is practically limited to the original authors, of whom none can keep abreast with this busy research field. As a rule, no well-defined programming interface is presented to third-party developers who might share the load. Of the software listed above, R/qtl, essentially a function library written in the popular multi-platform statistical and graphical language R, is the most extensible. But use of object-oriented (OO) languages, such as Java simplifies third-party extension of software by exploiting the OO concept of inheritance to make unnecessary the modification of existing code when new functionality is desired.

Shared development can be promoted by use of a plug-in architecture. Plug-in facility is built into most Web browsers, spreadsheet and graphics applications, development frameworks such as Eclipse and bioinformatics tools such as MEGA (Kumar et al., 2008) and Cytoscape (Shannon et al., 2003). In this model, while the base application or ‘platform’ is ready for use without programming, it also accommodates add-on modules that are found and loaded at start time, and may then be run from the application’s interface like the built-in native features. While extension of the software with a new plug-in does require programming, the analyst need not write or even understand the supporting calculation, display and result-export machinery.

ABSTRACT

Summary: Of many statistical methods developed to date for quantitative trait locus (QTL) analysis, only a limited subset are available in public software allowing their exploration, comparison and practical application by researchers. We have developed QGene 4.0, a plug-in platform that allows execution and comparison of a variety of modern QTL-mapping methods and supports third-party addition of new ones. The software accommodates line-cross mating designs consisting of any arbitrary sequence of selfing, backcrossing, intercrossing and haploid-doubling steps that includes map, population, and trait simulators; and is scriptable.

Availability: Software and documentation are available at http://coding.plantpath.ksu.edu/qgene. Source code is available on request.

Contact: jcn@ksu.edu

To whom correspondence should be addressed.

© The Author 2008. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oxfordjournals.org
Analysts may wish to compare for themselves the behavior of different methods. QGene allows superimposed displays of QTL profiles for any desired number of analysis methods, traits and chromosomes. Plug-ins have been implemented for SIM and CIM by least squares and EM algorithms, MIM, BIM with diagnostic plots, PMLE and shrinkage IM (ShIM), an IM extension of PMLE to be described elsewhere. Multiple-trait methods include the EM-based approach of Jiang and Zeng (1995) and extensions of MIM, PMLE and ShIM. Methods common to other software packages run with comparable or higher speed, while several methods and features are available nowhere else. One example is manual (as well as automated) selection of CIM cofactors followed by dynamic recalculation of the QTL profile. QGene can also simulate maps, genotype data for all mating designs and multiple correlated traits from a variety of statistical distributions. Trait analysis of datasets includes segregation testing, assessment of trait normality and correlation, and transformation.

QGene supports internationalization via translation of a set of text resource files into any desired language. The software can be scripted, allowing execution of simulation experiments.

3 POTENTIAL EXTENSIONS

Some analyses and features that QGene (like most QTL software) currently lacks are: QTL × QTL interaction; non-genetic covariates; mixed models for analysis of multiple-environment trials; outcross, half-sib and multi-cross designs; autopolyploids; triploid endosperm, sex-linked and categorical traits; non-parametric QTL mapping; eQTL, functional and association mapping; and interaction with WWW databases. Some of these may be added by the authors, while others could be added as plug-ins by other developers. While the easiest plug-ins to develop will be for QTL mapping methods and file readers, GUI or database plug-ins are also possible for more ambitious developers. Plug-ins could even invoke external QTL-mapping or general statistical software programs and supply their results to QGene’s graphics machinery for comparative viewing.

4 CONCLUSIONS

Bioinformatics workflows increasingly rely on modular, distributed systems based on shared development, free of excessive dependence on central sources and dynamically adaptable to the suite of components available at any moment. While QGene falls well short of the ultimate extension of this trend (the Semantic Web and grid computing more closely approach it), we have tried to develop a QTL-mapping resource more compatible with it than most other QTL-mapping tools provided to date.

Funding: Support has been provided by National Science Foundation NSF-DBI grant 0109879 and USDA-NRI Applied Plant Genomics Program grant 2004-35317-14867 entitled ‘RiceCAP: A coordinated research, education, and extension project for the application of genomic discoveries to improve rice in the United States’. This is contribution 09-111-J from the Kansas Agriculture Experiment Station.

Conflict of Interest: none declared.

REFERENCES