

R/qtl: An extensible QTL mapping environment

Karl W. Broman¹, Saunak Sen², Gary A. Churchill²

¹Dept. of Biostatistics, The Johns Hopkins University

²The Jackson Laboratory

<http://biosun01.biostat.jhsph.edu/~kbroman>

Abstract

We are developing a QTL mapping environment, [R/qtl](#), as an add-on package for the freely available and widely used statistical language/software [R](#) (see <http://www.R-project.org>). The development of this software as an add-on to R allows us to take advantage of the basic mathematical and statistical functions, and powerful graphics capabilities, that are provided with R. Further, the user will benefit by the seamless integration of the QTL mapping software into a general statistical analysis program. Our goal is to make complex QTL mapping methods widely accessible and allow users to focus on modeling rather than computing.

A key component of computational methods for QTL mapping is the hidden Markov model (HMM) technology for dealing with missing genotype data. We have implemented the main HMM algorithms, with allowance for the presence of genotyping errors, for backcrosses, intercrosses, and phase-known four-way crosses.

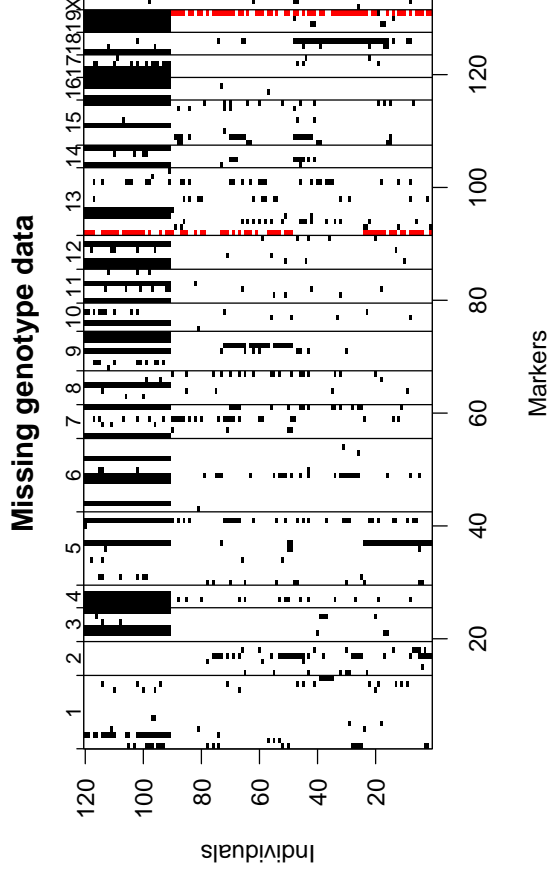
The current version of [R/qtl](#) includes facilities for estimating genetic maps, identifying genotyping errors, and performing single-QTL scans by interval mapping. The inclusion of covariates, analysis of dichotomous and ordinal traits, and a variety of multiple-QTL methods (MIM, MCMC and imputation) will be incorporated soon.

An example

Boyartchuk et al. (2001) Nat Genet
27:259-260

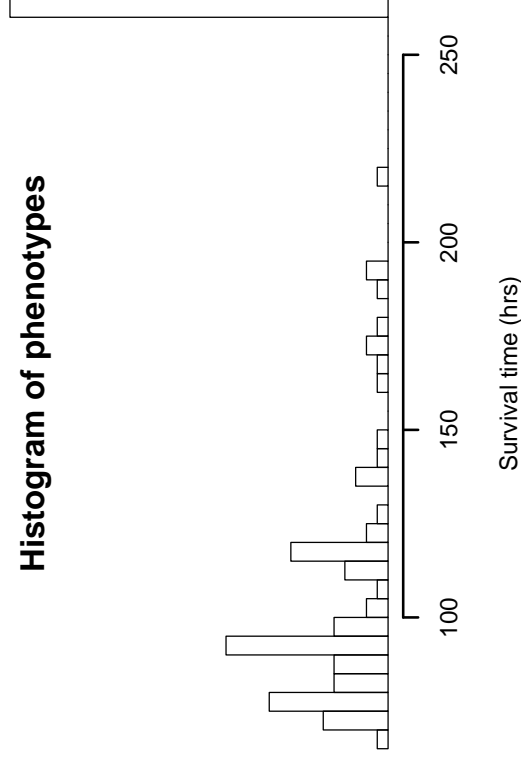
Pattern of missing genotype data.

Black pixels indicate a missing genotype. Red pixels indicate partially informative genotypes (e.g. A—).



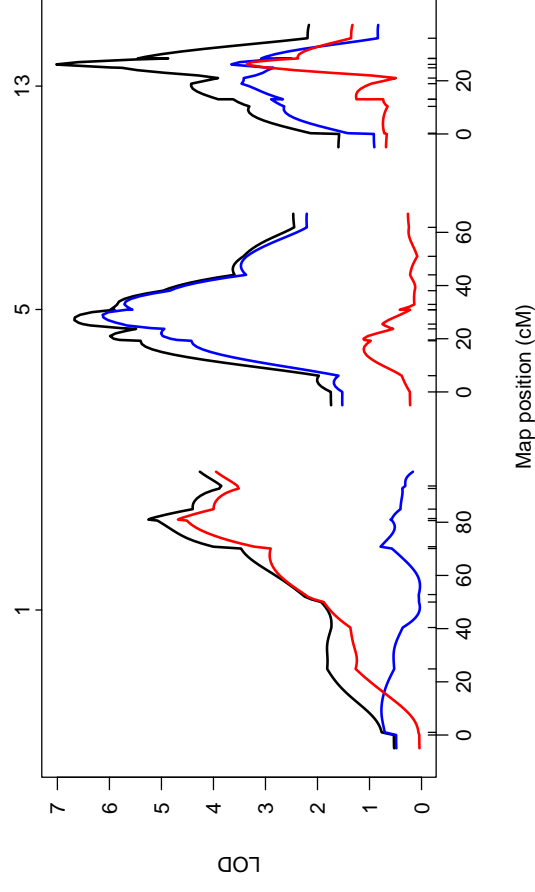
Histogram of survival time following bacterial infection.

Note that nearly 30% of these intercross mice recovered from the infection, surviving longer than 264 hours.



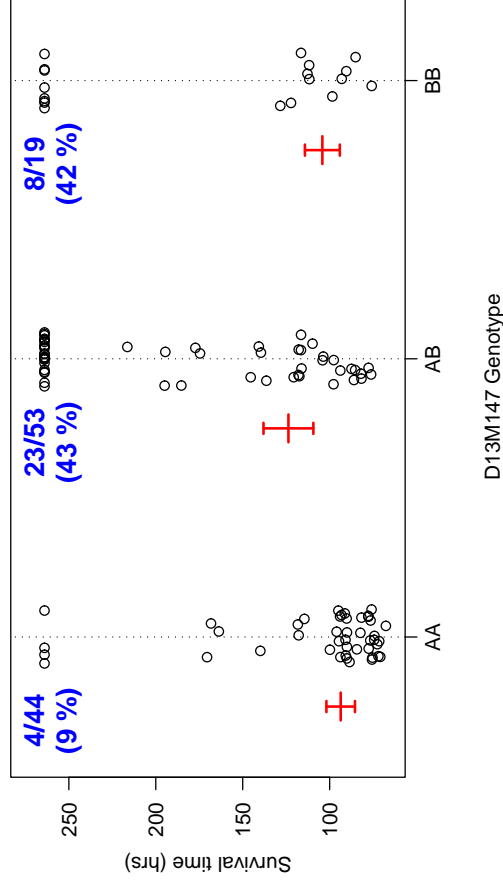
LOD curves using a two-part model (see panel 9).

Blue curves indicate differences in the probability of survival. **Red** curves indicate differences in the average time of death. **Black** curves indicate overall evidence for the presence of a QTL.



Phenotypes split according to genotype at D13Mit147.

Note the differences in both the probability of survival and the timing of death.



Why R/qtl?

- An interactive QTL mapping environment embedded within a general data analysis environment.
- Access to a variety of different approaches for QTL mapping, including sophisticated multiple-QTL methods.
- Includes functions for estimating genetic maps, identifying genotyping errors, and visualizing data.
- Easy extensibility for use with specialized crosses or specially-tailored models.
- Implemented in a combination of R and C.

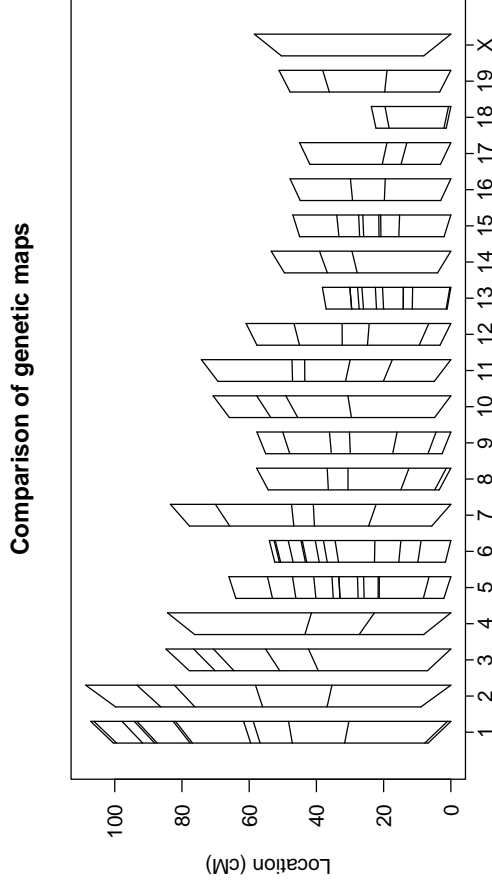
About R

- Open-source implementation of the S language. (Like S-PLUS, but free.)
- Language and environment for statistical computing and graphics.
- Provides a wide variety of statistical and graphical techniques (including linear and nonlinear modelling, statistical tests, time series analysis, classification, clustering).
- Available for UNIX, Windows and MacOS.

Genetic mapping

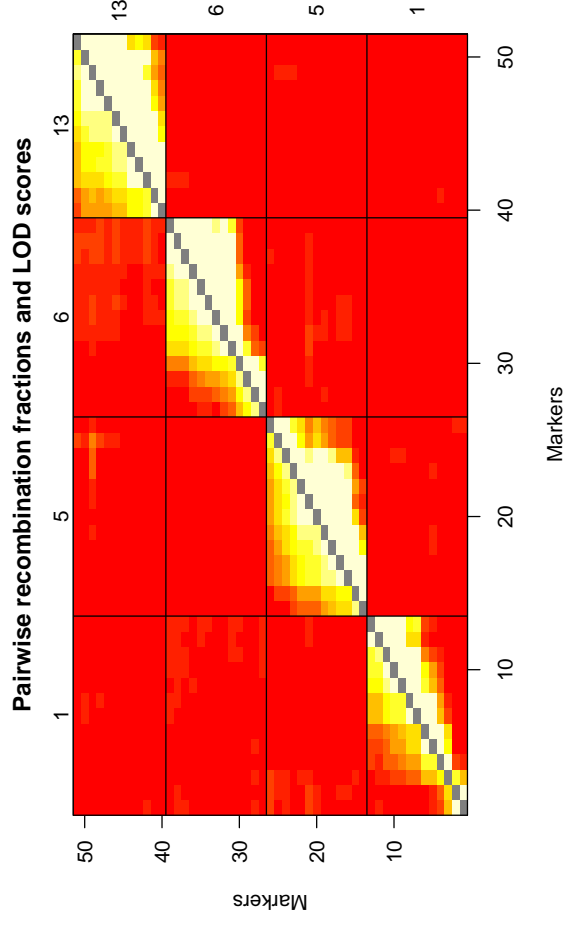
Estimated genetic maps.

R/qtl includes facilities for estimating genetic maps, with allowance for the presence of genotyping errors. A comparison of genetic maps can be a valuable diagnostic tool.



Pairwise recombination fractions.

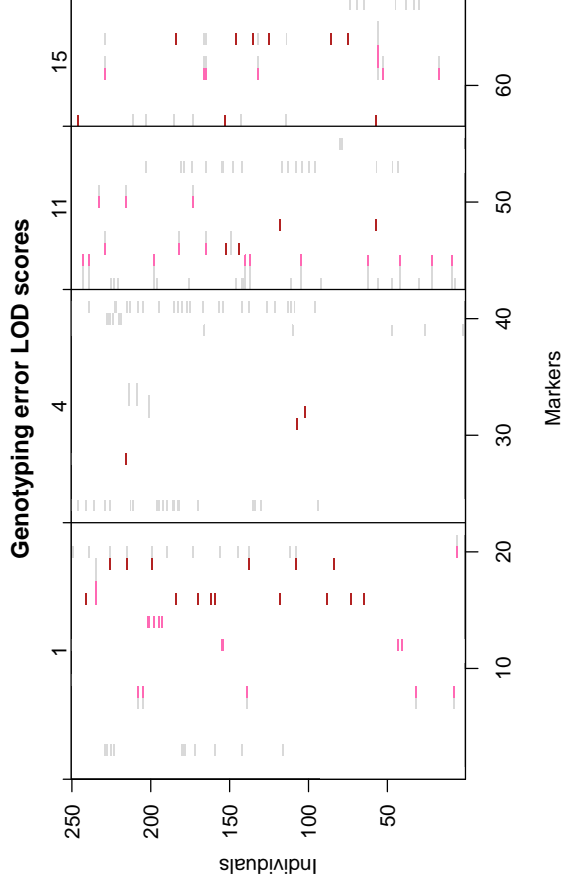
An inspection of estimated recombination fractions between all pairs of markers can assist in the identification of misplaced markers.



Genotyping errors

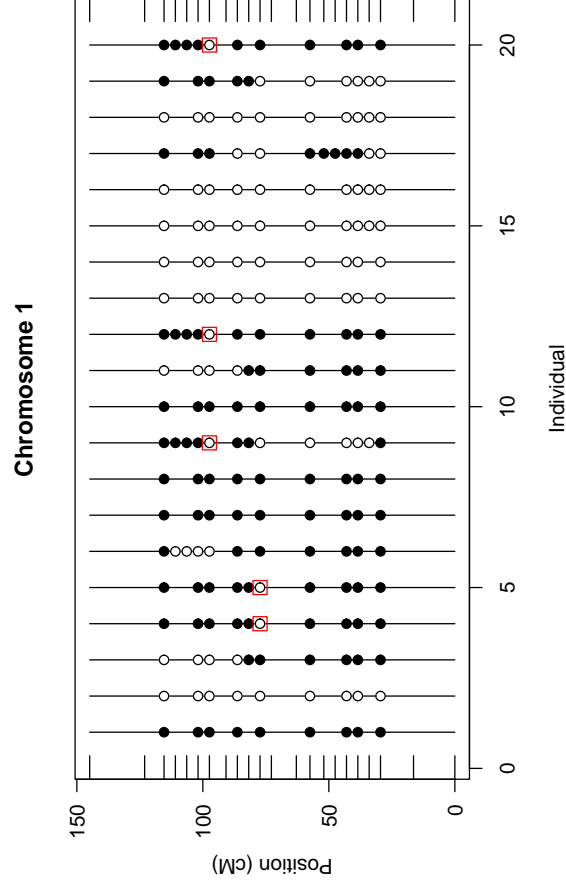
Error LOD scores.

The error LOD scores described by Lincoln and Lander (1992) assist in the identification of possible genotyping errors. Pink and red pixels at right indicate likely genotyping errors.

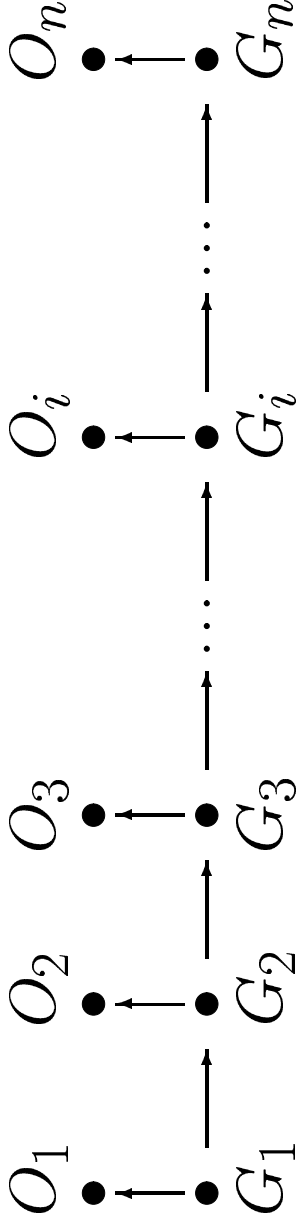


Visualization of genotype data

At right, marker genotype data for a backcross (open and closed circles correspond to the AA and AB genotypes, respectively). Likely genotyping errors are flagged.



The hidden Markov model engine



G_i = true genotype at marker i

O_i = observed genotype at i

The $\{G_i\}$ form a Markov chain.

O_i depends only on G_i .

Init: $\pi(g) = \Pr(G_1 = g)$

Step: $t_i(g, g') = \Pr(G_{i+1} = g' | G_i = g)$

Emit: $p_i(g) = \Pr(O_i | G_i = g)$

The above are specific to the type of cross. **Emit** may allow for genotyping errors.

Tasks:

- Calculate $\Pr(G_i = g | \mathbf{O})$.
- Draw from $\Pr(\mathbf{G} | \mathbf{O})$.
- Calculate $\arg \max_{\mathbf{G}} \Pr(\mathbf{G} | \mathbf{O})$.
- Estimate inter-marker distances.

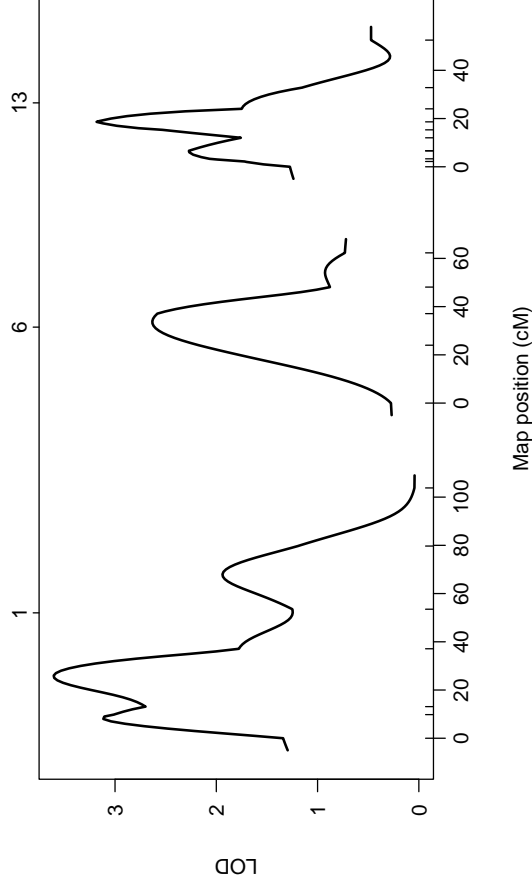
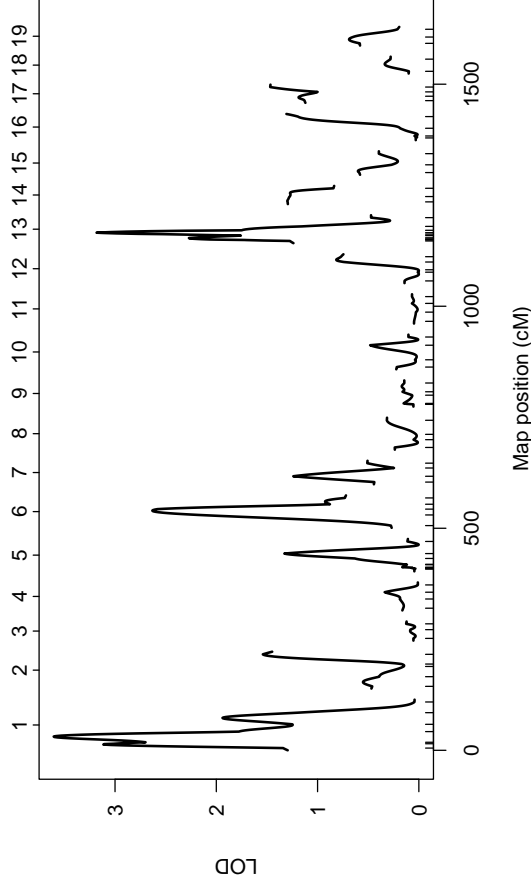
Single QTL methods

Currently:

- ANOVA at marker loci (aka marker regression)
- Interval mapping (IM)
- Haley-Knott approximation
- IM with a two-part model
- Permutation tests

Soon:

- IM with covariates (e.g. sex)
- IM with censored phenotypes
- Non-parametric IM



A two-part model for QTL mapping

A common departure from the assumption of normality: a large proportion of individuals have a phenotype that is either 0 or ∞ .

The two-part model:

For a mouse with QTL genotype g ,

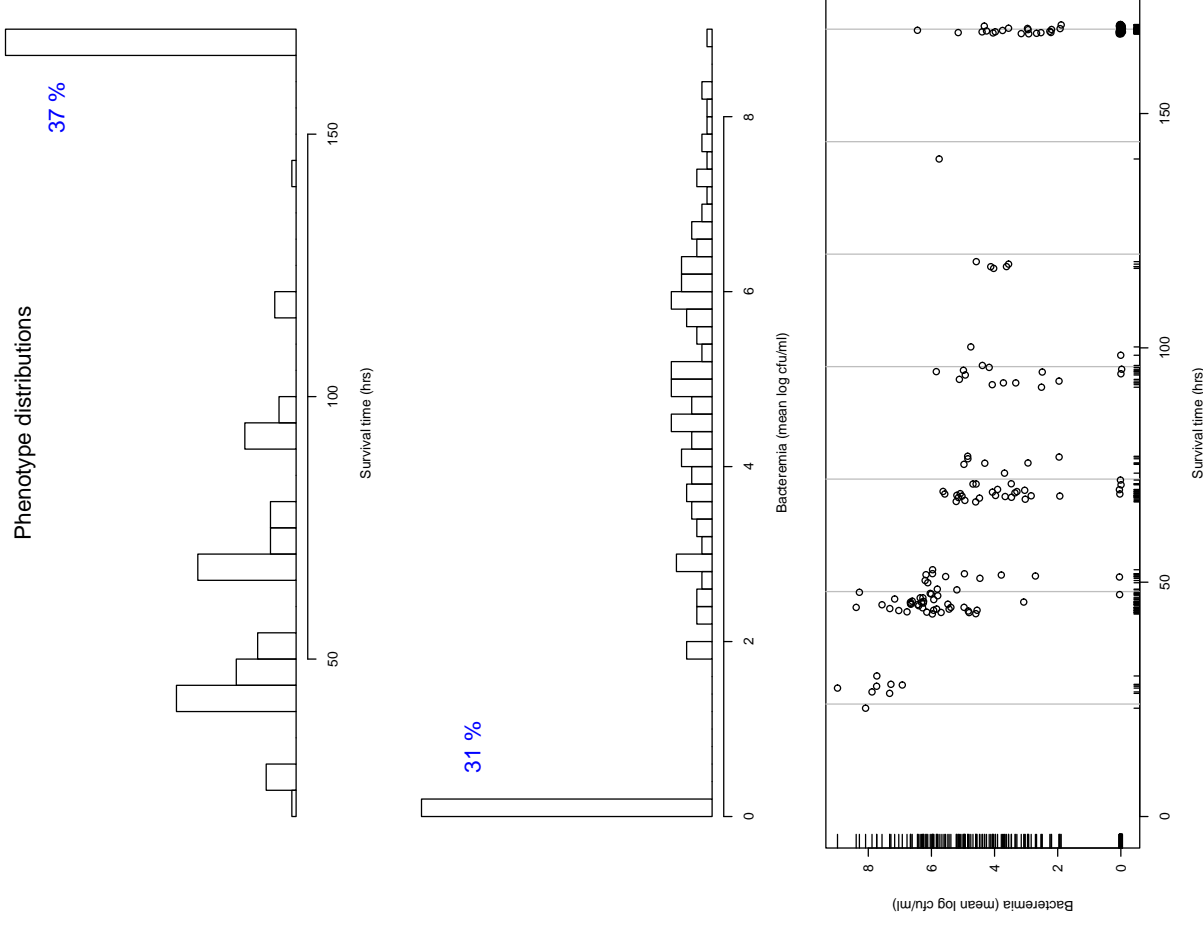
$$p_g = \Pr(\text{undefined phenotype} \mid g)$$

Given the phenotype (y) is defined,

$$y \mid g \sim \text{normal}(\mu_g, \sigma^2).$$

We calculate LOD scores measuring differences among the p_g , among the μ_g , and overall. (See panel 4.)

This illustrates the **extensibility** of R/qtl. While additional programming was required, we didn't need to start from scratch.



Multiple QTL methods

Why?

- Separate linked QTLs
- Examine epistasis (interactions between QTLs)
- Increase power to detect QTLs

Structure of the problem

- Class of models (e.g., linear with pairwise interactions)
- Model comparison (e.g., conditional LOD)
- Model search (e.g., stepwise selection)
- Expression of uncertainty
- Evaluation of the performance of a procedure

Currently:

Multiple regression

Soon:

Multiple interval mapping (MIM)

The pseudomarker algorithm (imputation)

Markov chain Monte Carlo (MCMC)

Eventually:

Tree-based models

Composite interval mapping

Advanced model search methods

Your favorite method