vqtl: An R package for QTL Mapping on Phenotypes with Heterogeneous Variance

Robert W. Corty and William Valdar

November 6, 2017

Abstract

Existing methods for QTL mapping in experimental crosses assume that the amount of residual variation is constant across all individuals. Many common situations violate this assumption. For example, female mice often have more variable phenotypes than males, some experimenters make more precise measurements than others, and specific genetic loci may influence the extent of variation between individuals. In all such cases, the heterogeneous variance modeling approach demonstrated here provides higher power, better protection against false positives, and allows for detection of QTL that influence phenotype variance, termed vQTL. The `R` package `vqtl` makes it easy for geneticists to apply the heterogeneous variance model, control family-wide error rate (FWER), and visualize and interpret their results. Because this package is interoperable with the popular `R/qtl` package and uses many of the same data structures and input patterns, it will be easy for geneticists to analyze the results of their experimental crosses with `vqtl`, possibly discovering new QTL. Here, we demonstrate typical usage.

Introduction

Experimental crosses of inbred organisms have been a cornerstone of forward genetics for over a century (**mendel1866**). They have provided important insights on nearly every trait of interest in human disease, agriculture, and livestock production. Studies across all these diverse fields were enabled by advances in methods for efficient breeding, phenotyping (Yang et al., 2014), genotyping (Williams et al., 1990), statistical methods (Lander and Botstein, 1989), and software tools (Broman et al., 2003).

One common assumption throughout is that the extent of residual variation is constant across all organisms in a study population. Said another way, it is typically assumed that nothing – neither environmental factors nor genetic factors – influences the extent of residual phenotype variation. In the companion piece of this article, we introduced a statistical modeling approach that accommodates heterogeneity in residual variation within a study population. We further demonstrated two critical benefits of using this "simultaneous mean-variance" modeling approach.

1. It allows for the detection of genetic loci that influence the extent of residual variation, which are likely to play central roles in the network of molecular interactions that gives rise to a complex trait.
2. It allows for accurate consideration of the quantity of information provided by each individual. Individuals with less residual variation provide more information about the mean of the groups to which they belong, as is evident in the formula for the standard error of the mean of a normally-distributed random variable: $\mathrm{SE}_\mu = \sigma/\sqrt{n}$.

Using the "simultaneous mean-variance" approach, we reanalyzed an F2 intercross of two inbred mouse strains that was carried out and published in 2008. This reanalysis uncovered some novel results. In support of other researchers who may be interested in reanalysing previously-conducted mapping studies using the simultaneous mean-variance approach, we have composed `R` package `vqtl`, which provides functions for conducting mean-variance genome scans, assessing the statistical significance of results, and visualizing and interpreting significant findings. `R` package `vqtl` uses the same `cross` data structure as the popular `qtl` package and is available on `CRAN`, so it is easy to get started.

Here, we demonstrate typical use of the `vqtl` package. The code used to simulate the phenotypes, calculate the statistics, estimate their significance, and visualize significant results is available at `github.com/rcorty`.

Simulate an Experimental Cross

We used `R/qtl` to simulate the experimental cross to be analyzed. We simulated a population of 200 male and 200 female F2 offspring, with 3 chromosomes of length 100 cM, each tagged by 11 equally-spaced markers and estimated genotype probabilities at 2cM intervals with `R/qtl`'s hidden markov model. We simulated four phenotypes:

1. `phenotype1` consists only of random noise and will serve as an example of negative results for all tests.
2. `phenotype2` is influenced by the 6th (middle) marker on chromosome one. The marker influences the mean of the phenotype, but not the variance, so it will serve as an example of a pure "mQTL".
3. `phenotype3` is influenced by the 6th (middle) marker on chromosome two. The marker influences the variance of the phenotype, but not the mean, so it will serve as an example of a pure "vQTL".
4. `phenotype4` is influenced by the 6th (middle) marker on chromosome three. The marker influences both the mean and the variance of the phenotype, so it will serve as an example of a joint "mvQTL".

We additionally consider `phenotype1x` through `phenotype4x`, which have the same type of genetic effects as `phenotype1` through `phenotype4`, but additionally have covariate effects on phenotype variance. All the same analyses and plots that are shown for `phenotype1` through `phenotype4` are shown for `phenotype1x` through `phenotype4x` in the appendix.

Conduct Genome Scans

The central function for genetic mapping in package `qtl` is `scanone`. Analogously, the central function for genetic mapping in package `vqtl` is `scanonevar`. `scanonevar` takes three required inputs:

1. `cross` contains the genetic and phenotypic information from an exerimental cross. This object can be the same `cross` object used in package `qtl`.
2. `mean.formula` specifies the phenotype to be mapped, the covariates to be corrected for, and the QTL terms to be fitted (`additive` and `dominance` components by default). The `mean.formula` uses the standard `R` formula notation.
3. `var.formula` specifies the covariates to be corrected for as well as the QTL terms to be fitted (`additive` and `dominance` components by default) in modeling the residual variance. The `var.formula` also uses the standard `R` formula notation.

Optional argument `chrs` is used to specify a subset of chromosomes to be scanned, defaulting to all chromosomes. Optional argument `return.covar.effects` is used to specify whether or not fitted effects of all covariates should be returned as part of the scan result, defaulting to `FALSE`.

Unlike `scanone`, which only tests for association between each locus and the phenotype mean, `scanonevar` computes three tests for each locus – association with phenotype mean, association with phenotype variance, and joint association with phenotype mean and variance. The statistic for each of these associations is a LOD score, the log of the ratio of the likelihood of the alternative model to the null model. The results of `scanonevar` on each of the four described phenotypes is shown in figure 1. The details of the null and alternative models used in each of the three tests can be found in the companion article.

The LOD score is the traditional test statistic in QTL mapping, but its interpretation can be difficult for two reasons: (1) LOD scores must be interpreted differently in autosomes and sex chromosomes. The sex chromosomes are typically fit with fewer parameters, depressing the expected value of the LOD scores under the null, and thus increasing the significance of any observed LOD score. (2) LOD scores from different tests must be interpreted differently. For example, the LOD score of the mvQTL test is always higher than the LOD score of the mQTL and vQTL tests. This relationship is due to the nested nature of the mvQTL test and the other two tests. All three tests use the same alternative model, but the null model in the mvQTL test imposes all the constraints of the null of the mQTL test and all the constraints of the null of the vQTL

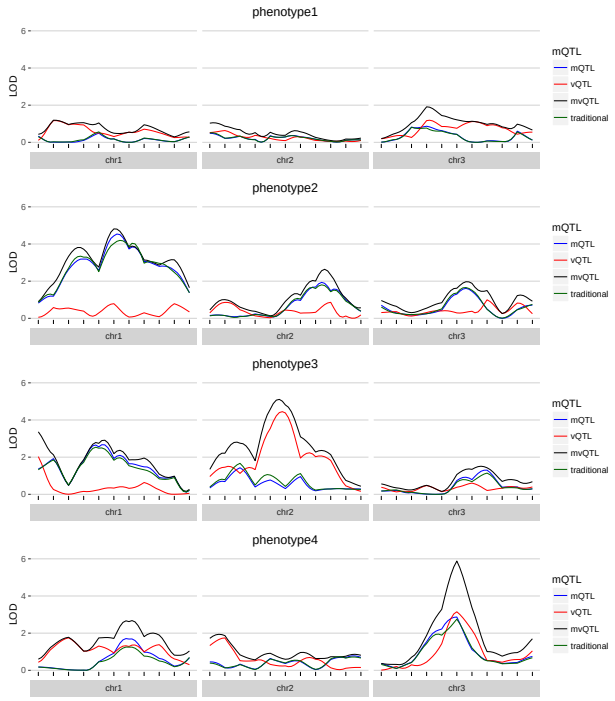


Figure 1: For each of the four simulated phenotypes, the genome scan shows the LOD score of each test – mean, variance, and joint – in blue, red, and black, respectively. The traditional test is in green and globally similar to the mean test.

test, so the LOD score of the mvQTL test must, by definition, be greater than the LOD score of the other tests.

These difficulties in interpreting the LOD score evident in figure 1. It seems that there are no important signials in the genome scan of `phenotype1` and it is visually clear that the most interesting signals for `phenotype2`, `phenotype3`, and `phenotype4` are on chromosomes one two and three, respectively. But important questions remain: (1) If we didn't have the genome scans from `phenotype2` - `phenotype4` available for comparison, would we be confident there are no statistically significant signals related to `phenotype1`? (2) How can we compare the results of the mvQTL test to the results of the other tests? (3) How could we compare the results of tests on autosomes to tests on sex chromosomes (if they were present)? (4) How often do we expect to observe results of this magnitude or greater when there is no true association, due simply to sampling variation and the multiplicity of tests conducted?

To put all genetic loci and all three tests on a level playing field, we consider two types of p-values: (1) Asymptotic p-values are calculated by `scanonevar` using the χ^2 distribution with the appropriate degrees of freedom for each locus. Though these p-values overcome the questions 1 through 3 of working with LOD scores described above, they leave problem 4 unresolved. (2) Empirical, family-wide error rate (FWER) -corrected p-values resolve all the above issues with interpreting LOD scores, thus they are the recommended means of assessing the significance of QTL mapping results. Methods for calculating them are described in the next section.

The object returned by the `scanonevar` function has class `scanonevar`. Calling `plot` on this object produces a publication-quality plot that shows the three association statistics at each locus. Calling `summary` on this object produces a summary of how the scan was conducted and what the results were. With this example dataset, it takes five seconds to run one genome scan on a Intel Core i5.

Assess the Significance of Results

The p-value assigned to each locus by calculating the position of its calculated LOD score in its null distribution answers the question, "How probable is it to observe this much or more evidence against the null at this locus, given that there is no true effect?" This question is not entirely relevant for genetic mapping, however, because we typically test for association at many loci, with no special interest in any individual locus *a priori*. Thus, a more appropriate p-value would answer the question, "How probable is it to observe this much or more evidence against the null at some locus in the genome, given that there are no true effects?" This is precisely the reasing behind the family-wide-error-rate (FWER) controlling procedure we illustrate below.

To calculate a FWER-controlling p-value, one typically must estimate the effective number of statistical tests conducted. The effective number of tests in a genome scan, however, is difficult to estimate. One lower bound is the number of chromosomes. Due to the randomization in meiosis no two non-syntenic loci are correlated in an experimental cross and therefore tests on different chromosomes are always independent. But there are many non-identical tests conducted on each chromosome, so the number of chromosomes is an under-estimate. One upper bound on the effective number of tests is the total number of loci. But, loci on the same chromosome are often in linkage disequilibrium, therefore the tests are not independent and the total number of loci is an over-estimate.

Our empirical approach sidesteps estimation of the effective number of tests. We conduct many permuted genomes scans, each executing a carefully-constructed model comparison to maintain all mean and variance effects of covariates and any non-focal genetic effects on mean and variance. The details of these model comparisons are provided in the companion article. For each test, we extract the highest observed value of the test statistic from each permutation scan and use those to model a generalized extreme value distribution. The observed LOD scores from the genome scan are then transformed by the cumulative distribution function of the extreme value distribution to estimate the FWER-controlling p-values. This approach is implemented in the function, `scanonevar.perm`. `scanonevar.perm` takes two required inputs:

1. `sov` is the `scanonevar` object, the statistical significance of which will be assessed through permutation.
2. `n.perms` is the number of permutations to conduct.

The object returned by `scanonevar.perm` is a `scanonevar` object with one important additional piece of information, an empirical p-value for each test at each locus. These p-values are FWER-corrected, so a value of 0.05 for a specific test at a specific locus implies that in 5% of similar experiments where there is no true genotype-phenotype association, we would expect to observe *some* locus with this much or more evidence of association. Additionally, the returned object contains a list of the per-genome-scan maximum observed LOD for each test and each chromosome type.

Accurate estimation of the FWER-controlled p-values requires many permutation scans. We recommend at least 100, and rarely more than 1000. These permutation scans can be run on multiple processors by specifying the optional `n.cores` argument. On an Intel Core i5, running 100 permutations takes about five minutes. When many phenotypes are studied, or the 5-50 minute runtime is problematic, these permutation scans can be broken into groups with different values for `random.seed`, run on separate computers, and combined with the `c` function. This function combines the permutations from all the inputted scans, re-evaluates the observed LOD scores in the context of all available permutations, and returns a new `scanonevar` object with more precisely estimated empirical p-values.

Investigate Significant Findings

Having identified some QTL, we want to visualize the allele effects at those loci. Because the `vqtl` package models both mean and variance effects, existing plotting utilities aren't able to display the entirety of the modeling results. To investigate the results of a `vqtl` scan at one particular locus, we developed the `mean_var_plot`. This plot shows information about the phenotype mean on the horizontal axis and information about the phenotype variance on the vertical axis. There are both "model-free" and "model-based" versions of this plotting utility, but here we show only the "model-based" version here.

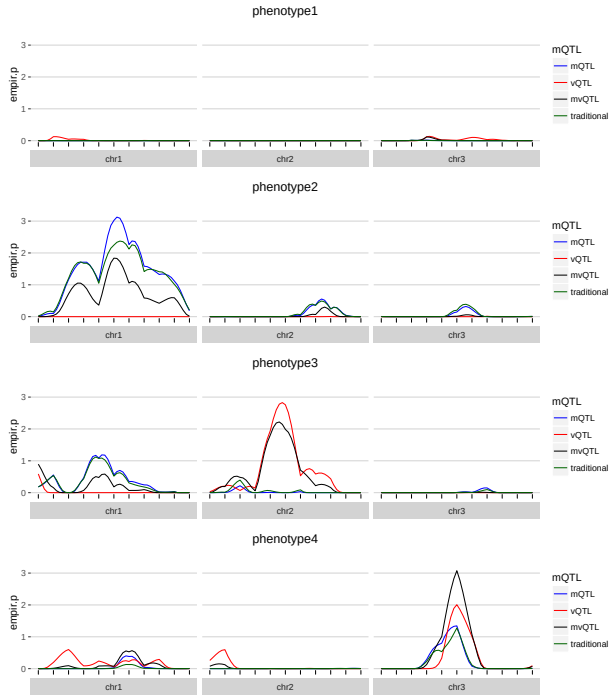


Figure 2: For each of the four simulated phenotypes, the genome scan shows the -log10 of the FWER-corrected p-value of each test – mean, variance, and joint – in blue, red, and black, respectively. Thus, a value of 3 implies that the quantity of evidence against the null is such that we expect to see this much or more evidence once per thousand genome scans when there is no true effect.

Figure 3: mean_var plots show the estimated genotype effects at a locus, with mean effects on the horizontal axis and variance effects on the vertical axis. Horizontal lines indicate standard errors for mean effects and vertical lines indicate standard errors for variance effects. More information on the interpretation of these plots is provided in the 'Investigate Significant Findings' section.

Figure 3 shows a model-based `mean_var_plot` for each of the four phenotypes. In each plot, the location of the dot shows the estimated mean and standard deviation of each genotype group, with the mean indicated by the horizontal position and the standard deviation indicated by the vertical position. The horizontal lines extending to the left and right from each dot show the standard error of the mean estimate, and the vertical lines extending up and down from each dot show the standard error of the standard deviation estimate. There are two types of grouping factors considered by the function `mean_var_plot_model_based`: (1) `focal.groups` are groups that are modeled and the prediction for each group is plotted. For example, a genetic marker is the `focal.group` in each plot in figure 3; `D1M1` in the top left, `D1M6` in the top right, etc. (2) `nuisance.groups` are groups that are modeled, but then averaged over before plotting. When there are many grouping factors thought to play a role in determining the mean and variance of an individual's phenotype, such as sex, treatment, and batch, we recommend putting just one or two in `focal.groups` and the others in `nuisance.groups` for clarity, cycling through which are displayed to gain a thorough understanding of the factors that determine the mean and variance of the phenotype.

For `phenotype1`, the `mean_var_plot` is shown at the first marker of the first chromosome. The estimates of the genotype effects on phenotype mean and variance are within one standard error of each other. This pattern is consistent with the fact that there are no genetic effects, and the p-value was not statistically significant at any locus.

For `phenotype2`, the `mean_var_plot` is shown at the most significant marker, the sixth marker on the first chromosome. The estimates of the genotype effects differ in the horizontal axis, but not the vertical axis. This pattern is consistent with the fact that there is a genetic effect on phenotype mean but none on phenotype variance and with the highly significant p-value for the mQTL test, but non-significant p-value for the vQTL test.

For `phenotype3`, the `mean_var_plot` is shown at the most significant marker, the sixth marker on the second chromosome. The estimates of the genotype effects differ in the vertical axis, but on on the horizontal axis. This pattern is consistent with the fact that there is a genetic effect on phenotype variance but none on phenotype mean and with the highly significant p-value for the vQTL test but non-significant p-value for the mQTL test.

For `phenotype4`, the `mean_var_plot` is shown at the most significant marker, the sixth marker on the third chromosome. The estimates fo the genotype effects differ somewhat in both the horizontal and vertical axes, but not as much on the horizontal axis as `phenotype2` and not as much on the vertical axis as `phenotype3`. This pattern is consistent with the fact that there is a genetic effect on both phenotype mean and variance and with the highly significant p-value for the mvQTL test and the marginally significant p-values for the other two tests.

Additional plotting utilities, `phenotype_plot`, `effects_plot` and `mean_var_plot_model_free` are described in the online vignette for the `vqtl` package, available at `github.com/rcorty`.

Real Data Example

We used the DGLM to reanalyze a reduced complexity cross between C57BL/6J and C57BL/6N mice in which cocaine response traits and circadian activity were measured retrieved from the mouse phenome database (**Bogue2007**). The SLM identified one QTL on chromosome 11 for the cocaine response traits but no QTL related to the circadian activity were reported (**Kumar2013**). The DGLM reanalysis identified the same QTL on chromosome 11 for the cocaine response traits and an additional QTL for a circadian activity trait on chromosome 6.

0.1 Summary of Experimental Methods

Kumar et. al intercrossed two "sister" strains of C57BL6 that diverged in 1951, approximately 330 generations ago. These strains were selected because they have a significant phenotypic difference in cocaine response, but only limited genetic differences owing to their recent coancestry. The intercross resulted in 244 F2 offfspring, 113 female and 131 male. These F2 offpsring were treated with 20mg/kg of cocaine and then tested in the open field during the day. At night time, their activity was measured on a running wheel. The effect of the cocaine was negligible, as its half life is approximately one hour, so at least five half lives had

passed by night time. After the initial study was completed, these strains were noted to have significantly different phenotypes related to circadian activity (**Banks2015**).

0.2 Results

QTL analysis using the SLM and the DGLM gave nearly identical results for cocaine-response traits (Fig. S1), replicating the published analysis (**Kumar2013**).

QTL analysis using the SLM identified no QTL for the "average wheel speed" trait (Fig. 4 in green), but QTL analysis using the DGLM identified one QTL on chromosome 6 (Fig. 4 in blue, black, and red). All three tests were statistically-significant, but the most significant was the mQTL test (blue), so we discuss it as an mQTL. Percent variance explained is somewhat harder to define in a model with variance heterogeneity, so we use the traditional linear model to assess the percent variance explained by the locus as 4.4%. As described above, the mQTL test tests for the same pattern that the SLM tests for – heterogeneity of genotype means.

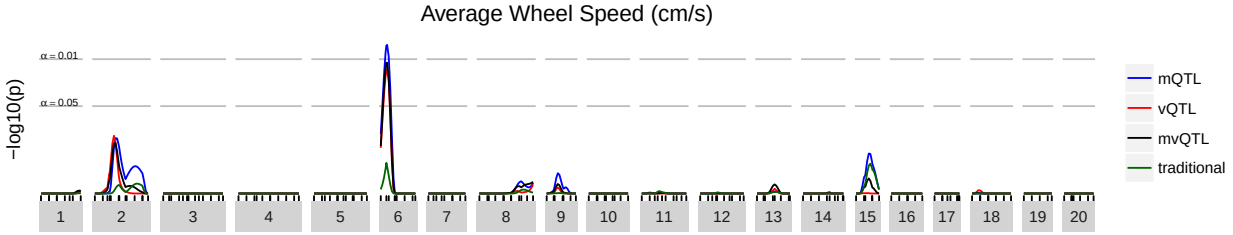


Figure 4: FWER-controlling p-values for the association between each genomic locus and the "average wheel speed" trait.

0.3 Understanding the Novel QTL

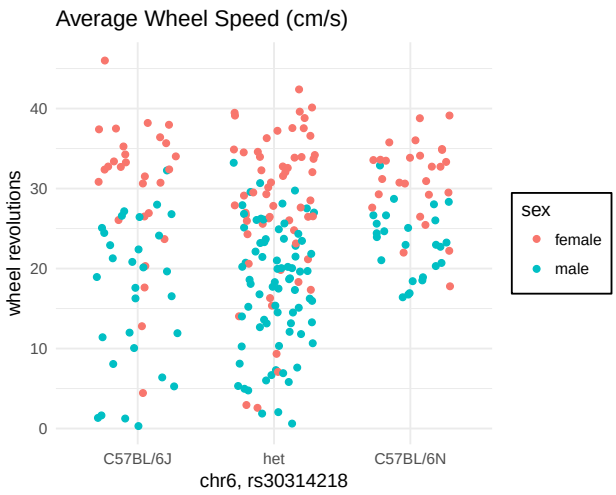
Though they test for the same pattern, the DGLM-based analysis identified an mQTL where the LM-based analysis did not. This situation may arise when there is variance heterogeneity due to the locus itself or due to a covariate. Figure 5a shows that the latter of these possibilities is true in this case.

Mice homozygous for the C57BL/6N allele at the QTL appear to have both higher average wheel speed and and less residual variance in average wheel speed than mice with other genotypes. The identification of this QTL by the DGLM but not the SLM can be understood by contrasting the model comparison involved in the DGLM_M test with the model comparison involved in the SLM test.

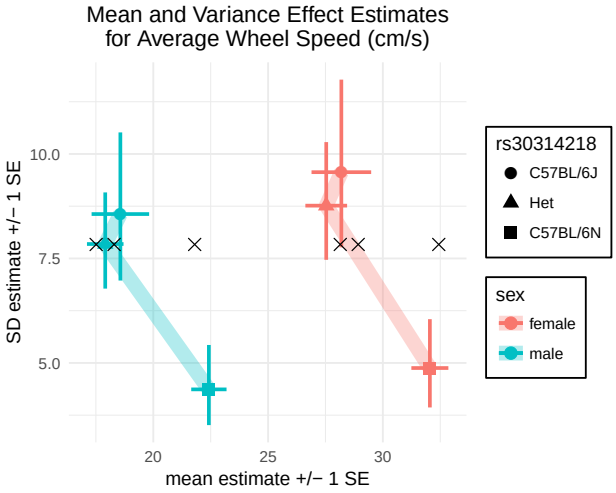
For the SLM, a single value of σ is estimated for the residual standard deviation of all mice. Approximately 25% of the mice are homozygous for the N allele, so σ is estimated mostly based on heterozygous and homozygous J mice. The SLM estimates $\hat{\sigma} = 7.83$, a slight underestimate for some genotype-sex combinations, and a drastic overestimate for the homozygous N of both sexes (Fig. 5b). With σ overestimated for the N homozygotes, the addition of a locus effect to the null model results in only a limited increase in the likelihood.

For the DGLM, six different values of σ are estimated, one for each genotype-sex combination (Fig. 5b). With a well-estimated σ for the 6N homozygotes, the addition of the locus effect to the null model results in a greater increase in the likelihood. Somewhat tautologically, the closer $\hat{\sigma}$ is to the maximum likelihood estimate of σ, the greater the likelihood of the alternative model and thus the greater the difference in likelihood between null and alternative models (Fig.).

A straightforward simulation, based on the estimated coefficients, shows that in a single-locus analysis, the DGLM has 30% power to reject the null at this locus and the SLM has 20% power (Fig. S2).



(a) Average wheel speed (cm/s) of all mice.



(b) Predicted mean and variance of mice according to sex and allele at the QTL.

Figure 5: (a) It is visually apparent that female mice had higher "average wheel speed" than male mice and that mice that were homozygous for C57BL/6N had higher "average wheel speed" and less intra-genotype variation in "average wheel speed". (b) What was visually apparent in subfigure (a) is captured by the DGLM. The estimated parameters relating to mice that are homozygous for the C57BL/6N allele imply a higher expected value and a lower residual variance than the other two genotype groups. Black x's indicate the estimates from the SLM, very similar to the DGLM estimates in the horizontal (mean) axis, but homogenous in the vertical (variance) axis.

Conclusion

We have demonstrated typical usage of the new `vqtl` `R` package for QTL mapping in experimental crosses. This package is most appropriate for crosses and phenotypes where nuisance covariates or genetic factors are known or suspected to influence phenotype variance. In the case of genetic factors, they can be mapped. In the case of nuisance covariates, they can be accommodated.

The central function of this package, `scanonevar`, carries out the initial genome scan. The permutation function, `scanonevar.perm`, executes a carefully-constructed set of permutations to empirically estimate a FWER-controlling p-value for each observed LOD score. A suite of plotting functions, e.g. the `mean_var_plot`, allows geneticists to evaluate the allelic and covariate effects that underlie detected signals.

Appendix

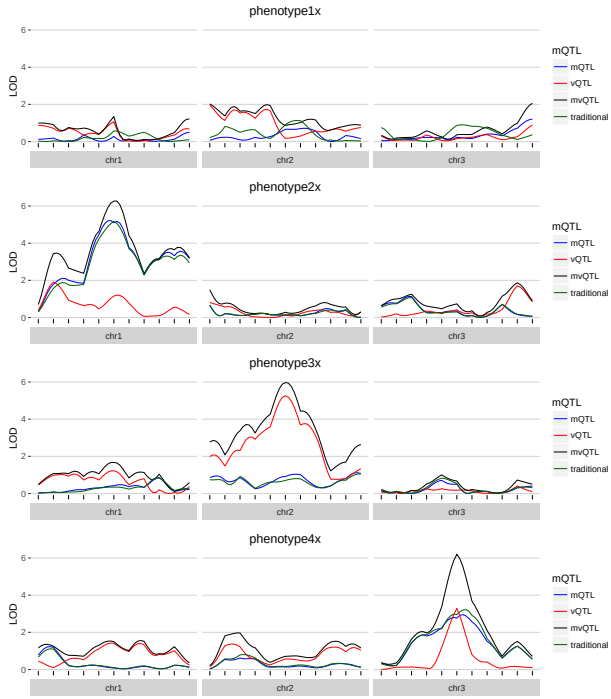


Figure A1: For each of the four simulated phenotypes, the genome scan shows the LOD score of each test – mean, variance, and joint – in blue, red, and black, respectively. The traditional test is in green and globally similar to the mean test.

Supplementary Materials

Supplemenat to Kumar Reanalysis

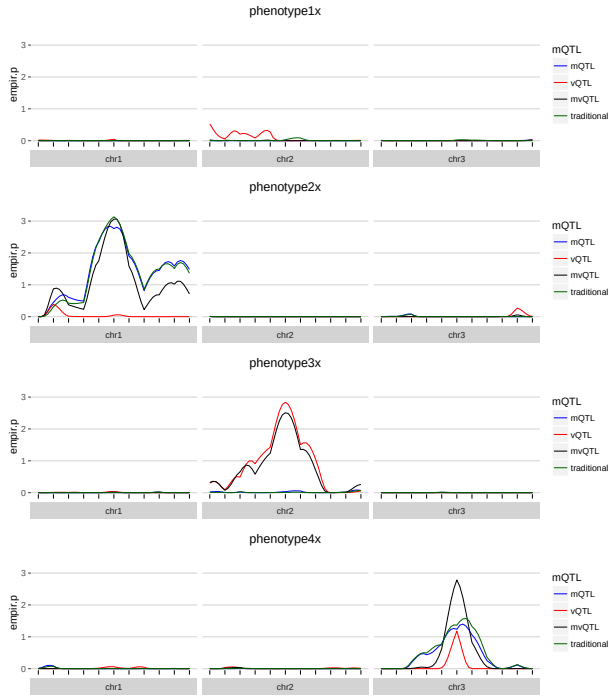


Figure A2: For each of the four simulated phenotypes, the genome scan shows the -log10 of the FWER-corrected p-value of each test – mean, variance, and joint – in blue, red, and black, respectively. Thus, a value of 3 implies that the quantity of evidence against the null is such that we expect to see this much or more evidence once per thousand genome scans when there is no true effect.

Figure A3: mean_var plots show the estimated genotype effects at a locus, with mean effects on the horizontal axis and variance effects on the vertical axis. Horizontal lines indicate standard errors for mean effects and vertical lines indicate standard errors for variance effects. More information on the interpretation of these plots is provided in the 'Investigate Significant Findings' section.

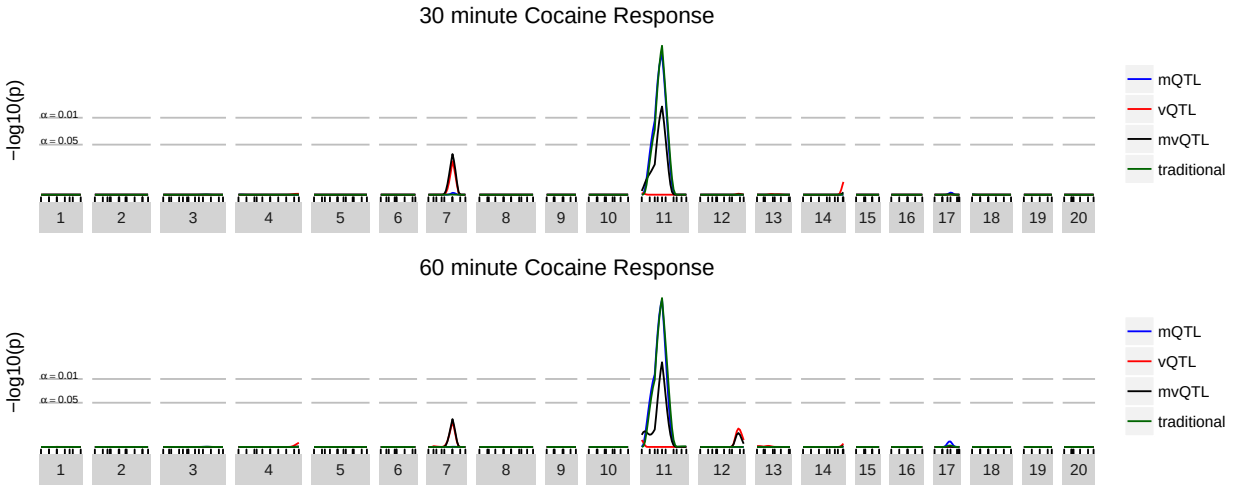


Figure S1: Replicated scans from Kumar2013

[possible cartoon about ML estimate of sigma giving higher likelihood than other estimates of sigma]

```
library(dglm)
set.seed(27599)

n <- 244
num_sims <- 1e4

sex <- rbinom(n = n, size = 1, prob = 0.5)
g <- rbinom(n = n, size = 2, prob = 0.5)

mean_mu <- 28.1915124
mean_sex <- -9.6256202
mean_het <- -0.6585349
mean_hom <- 0.8556717

var_mu <- 4.5160431
var_sex <- -0.2212854
var_het <- -0.1748949
var_hom <- -1.3465278
resid_scale <- 1.414214

mean_lp <- mean_mu + sex*mean_sex + mean_het*(g == 1) + mean_hom*(g == 2)
var_lp <- var_mu + sex*var_sex + var_het*(g == 1) + var_hom*(g == 2)
g <- factor(g)

lm_LRT <- dglm_LRT <- rep(NA, num_sims)
for (sim_idx in 1:num_sims) {
  y <- rnorm(n = n,
             mean = mean_lp,
             sd = exp(var_lp/resid_scale^2))

  lm_nul <- lm(formula = y ~ sex)
  lm_alt <- lm(formula = y ~ sex + g)

  dglm_nul <- dglm(formula = y ~ sex, dformula = ~ sex + g)
  dglm_alt <- dglm(formula = y ~ sex + g, dformula = ~ sex + g)

  lm_LRT[sim_idx] <- 2*(logLik(object = lm_alt) - logLik(object = lm_nul))
  dglm_LRT[sim_idx] <- dglm_nul$m2loglik - dglm_alt$m2loglik
}

par(mfrow = c(3, 1))
plot(x = sort(lm_LRT), y = sort(dglm_LRT)); abline(0, 1)
hist(x = lm_ps <- pchisq(q = lm_LRT, df = 2, lower.tail = FALSE), xlim = c(0, 1), main = 'SLM')
hist(x = dglm_ps <- pchisq(q = dglm_LRT, df = 2, lower.tail = FALSE), xlim = c(0, 1), main = 'DGLM')

mean(lm_ps < 0.05)
mean(dglm_ps < 0.05)
```

Figure S2: `R` code to simulate power of DGLM and LM in Kumar study at rs30314218.