Journal of Statistical Software

April 2011, Volume 40, Issue 7. *http://www.jstatsoft.org/*

R Package wgaim: QTL Analysis in Bi-Parental Populations Using Linear Mixed Models

Julian Taylor
CSIRO

Arunas Verbyla
CSIRO

Abstract

The **wgaim** (whole genome average interval mapping) package developed in the R system for statistical computing (R Development Core Team 2011) builds on linear mixed modelling techniques by incorporating a whole genome approach to detecting significant quantitative trait loci (QTL) in bi-parental populations. Much of the sophistication is inherited through the well established linear mixed modelling package **ASReml-R** (Butler *et al.* 2009). As **wgaim** uses an extension of interval mapping to incorporate the whole genome into the analysis, functions are provided which allow conversion of genetic data objects created with the **qtl** package of Broman and Wu (2010) available in R. Results of QTL analyses are available using summary and print methods as well as diagnostic summaries of the selection method. In addition, the package features a flexible linkage map plotting function that can be easily manipulated to provide an aesthetic viewable genetic map. As a visual summary, QTL obtained from one or more models can also be added to the linkage map.

Keywords: interval mapping, mixed models, quantitative trait loci, R.

1. Introduction

Whole genome analysis is receiving wide attention in the statistical genetics community. In the context of plant breeding experiments the focus is on quantitative trait loci (QTL) which attempt to explain the link between a trait of interest and the underlying genetics of the plant. Many approaches of QTL analysis are available such as marker regression methods (Hayley and Knott 1992; Martinez and Curnow 1992) and interval mapping (Zeng 1994; Whittaker *et al.* 1996). These methods are common place in QTL software and are available for use in R packages such as the **qtl** package of Broman and Wu (2010). This particular suite of software is also complemented with a book (Broman and Sen 2009) which has been favourably reviewed (Zhou 2010).

There has also been some focus on the use of numerical integration techniques for the analysis of QTL. Xu (2003) and Zhang *et al.* (2008) suggest the use of Bayesian variable shrinkage and utilise Markov chain Monte Carlo (MCMC) to perform the analysis. An MCMC approach is also adopted in the R package **qtlbim** (Yandell *et al.* 2005). The package builds on the **qtl** package and the Bayesian paradigm allows an extensible list of trait types to be analysed. The package also makes use of the new model selection technique, the Deviance Information Criterion (Shriner and Yi 2009), to aid in identifying the correct QTL model. Similarly, a non-MCMC approach is adopted in the **BayesQTLBIC** package (Ball 2010) where the QTL analysis involves the use of the Bayesian Information Criterion (Schwarz 1978) as a QTL model selection tool.

Unfortunately many of the aformentioned methods and their software lack the ability to account for complex extraneous variation usually associated with plant or animal based QTL studies. Limited covariate additions are possible in R package **qtlbim** and through the inventive online **GridQTL** software which uses the ideas of Seaton *et al.* (2002). Kang *et al.* (2008) uses linear mixed models in the R package **EMMA** but it does not allow for extraneous random effects and possible complex variance structures that may be needed to capture environmental processes, such as spatial layouts, existing in the experiment. In this paper we discuss the R package **wgaim** which implements the genetic and inferential derivations of the whole genome average interval mapping (WGAIM) approach of Verbyla *et al.* (2007). The package is available from the Comprehensive R Archive Network (CRAN) at `http://CRAN.R-project.org/package=wgaim`. This approach allows the simultaneous modelling of genetic and non-genetic variation through extensions of the linear mixed model. The extended model allows complex extraneous variation to be captured as well as simultaneously incorporating a whole genome analysis to detection and selection of QTL using a linkage map. The underlying linear mixed modelling analysis is achieved computationally using the R package **ASReml-R**. The simulation results and examples in Verbyla *et al.* (2007) show that WGAIM is a powerful tool for QTL detection and outperforms more rudimentary methods such as composite interval mapping. As it incorporates the whole genome into the analysis it eliminates the necessity for piecemeal model fitting along the genome which in turn avoids the use of model selection criteria to control the number of false positive QTL. It must be noted Huang and George (2009) also use **ASReml-R** as their core engine for whole genome QTL analysis in the R package **dlmap**. In this package, the backward elimination model fitting procedure and multiple testing corrections used to control false positive QTL suggest this package differs markedly from **wgaim**. In **wgaim** the false positives are controlled naturally by assuming a background level of QTL variation through a single variance component associated with a contiguous set of QTL across the whole genome. This parameter can then be tested to determine the presence of QTL somewhere on the genome. As a result, a less cumbersome approach to detecting and selecting QTL is ensured.

The WGAIM method uses an extension of interval mapping to perform its analysis. Thus, for convenience and flexibility, the **wgaim** package provides the ability to convert genetic data objects created in the **qtl** package to objects for further use in **wgaim**. The converted objects retain a similar structure to objects created in **qtl** and therefore can still be used with functions within the package. Users of **wgaim** need to be aware that it is a software package intended for the analysis and summary of QTL and only contains minimal tools for exploratory linkage map manipulation. Much of the exploratory work can be handled with functions supplied in the **qtl** package and users should consult its documentation if required. In addition, the

interval mapping approach of Verbyla *et al.* (2007) and its implementation in **wgaim** is also restricted to populations with only two distinct genotypes. Some of these populations include, double haploid (DH), back-crosses and recombinant inbred lines (RIL). To ensure this rule is adhered to, error trapping has been placed in the appropriate functions of **wgaim**.

Throughout the WGAIM procedure the underlying linear mixed model analysis is achieved using the highly flexible R software package **ASReml-R**, built as a front end wrapper for the more sophisticated stand alone version, **ASReml** (Gilmour *et al.* 2009). This software allows the user the ability to flexibly model spatial or environmental variation as well as possible variation that may arise from additional components associated with the experimental design. It uses an average information algorithm developed in Gilmour *et al.* (1995) that allows efficient computing of residual maximum likelihood (REML) (Patterson and Thompson 1971) estimates for the variance parameters. The use of REML estimation in the linear mixed model context becomes increasingly necessary in situations where the data is unbalanced. Much of its sophistication has been influenced from its common use in the analysis of crop variety trials (Smith *et al.* 2001, 2005, 2006) where complex additional components such as spatial correlation structures or multiplicative factor analytic models need to be incorporated into the mixed model. If available, the software also allows complex pedigree information to be included (Oakey *et al.* 2006). Many of these additional flexibilities in ASReml have also established it as a valuable software tool in the livestock industries. In more recent years it has been used as a core engine for more complex genetic analyses as in Gilmour (2007), Verbyla *et al.* (2007) and Huang and George (2009). The stand alone software and the R package **ASReml-R** is only commercially available through `http://www.vsni.co.uk/` but trial licenses are also available.

The paper is arranged as follows. Section 2 briefly describes the theory of the WGAIM algorithm that is implemented in **wgaim**. Section 3 presents a walk through a typical QTL analysis using the functions of **qtl** and **wgaim**. QTL analyses from two plant breeding experiments are provided in Section 4. The second example shows some of the enhanced features of **wgaim** including the ability to plot an aesthetic genetic map. For visualization QTL can also be placed on the map post analysis. Post analysis diagnostics are also available which present features of the forward selection procedure used to determine the QTL.

2. WGAIM theoretical method

Before discussing the functions of the **wgaim** package it is necessary to provide a theoretical overview of the methodology used in its implementation. The WGAIM approach is a forward selection method that uses a whole genome approach to genetic analysis at each step. Following Verbyla *et al.* (2007), initially a working model is developed that assumes a QTL in every interval. Thus for a given set of trait observations $\boldsymbol{y} = (y_1, \ldots, y_n)$ consider the model

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\tau} + \boldsymbol{Z}_e\boldsymbol{u}_e + \boldsymbol{Z}_g\boldsymbol{g} + \boldsymbol{e}, \tag{1}$$

where $\boldsymbol{\tau}$ is a t length vector of fixed effects with an associated $n \times t$ explanatory design matrix $\boldsymbol{X}$ and $\boldsymbol{u}_e$ is a $b \times 1$ length vector of random effects with an associated $n \times b$ design matrix $\boldsymbol{Z}_e$. Typically, the distribution of $\boldsymbol{u}_e \sim N(\boldsymbol{0}, \sigma^2\boldsymbol{G}(\boldsymbol{\varphi}))$ and is assumed mutually independent to the residual vector $\boldsymbol{e} \sim N(\boldsymbol{0}, \sigma^2\boldsymbol{R}(\boldsymbol{\phi}))$ with $\boldsymbol{\varphi}$ and $\boldsymbol{\phi}$ being vectors of variance ratios.

The vector $\boldsymbol{g}$ in (1) represents a r length vector of genotypic random effects with its associated design matrix $\boldsymbol{Z}_g$. Let c be the number of chromosomes and m_k be the number of markers

on chromosome k, $(k = 1, \ldots, c)$, and $q_{i,k:j}$ represent the parental allele type for line i in interval j on chromosome k. In WGAIM, $q_{i,k:j} = \pm 1$, reflecting two possible genotypes AA, BB for DH and RIL and AB, BB for back-cross populations. The ith genetic component of this model is then given by

$$g_i = \sum_{k=1}^{c} \sum_{j=1}^{m_k - 1} q_{i,k:j} a_{k:j} + p_i,$$

where $a_{k:j}$ is QTL effect size assumed to have distribution $a_{k:j} \sim N(0, \sigma^2 \gamma_a)$ and $p_i \sim N(0, \sigma^2 \gamma_p)$ represents a polygenic or residual genetic effect not captured by the QTL effects.

As in interval mapping the vector of QTL allele types are replaced by the expectation of the QTL genotype given the flanking markers. Let $\boldsymbol{m}_{k:j}$ be the jth marker on the kth chromosome and applying a parameter reduction technique from Verbyla *et al.* (2007) produces a vector of genotypic effects of the form

$$\begin{aligned} \boldsymbol{g} &= \sum_{k=1}^{c} \sum_{j=1}^{m_k - 1} (\boldsymbol{m}_{k:j} + \boldsymbol{m}_{k:j,j+1}) \psi_{k:j} a_{k:j} + \boldsymbol{p} \\ &= \boldsymbol{M}_\psi \boldsymbol{a} + \boldsymbol{p}, \end{aligned} \tag{2}$$

where $\psi_{k:j} = \theta_{k:j,j+1} / 2 d_{k:j,j+1} (1 - \theta_{k:j,j+1})$ and $\theta_{k:j,j+1}, d_{k:j,j+1}$ are the recombination fraction and Haldane's genetic distance between marker j and $j+1$ respectively on the kth chromosome. Thus $\boldsymbol{M}_\psi$ is a fully specified known matrix of pseudo-markers spanning the whole genome. A more detailed overview of this decomposition and its derivation can be found in Verbyla *et al.* (2007). Let $\boldsymbol{Z}_a = \boldsymbol{Z}_g \boldsymbol{M}_\psi$ then the full working statistical model for analysis is then

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\tau} + \boldsymbol{Z}_e \boldsymbol{u}_e + \boldsymbol{Z}_a \boldsymbol{a} + \boldsymbol{Z}_g \boldsymbol{p} + \boldsymbol{e}. \tag{3}$$

After the fitting of (3) the simple hypothesis $H_0 : \gamma_a = 0$ is tested based on the statistic $-2 \log \boldsymbol{\Lambda} = -2(\log L - \log L_0)$ where L and L_0 is the residual likelihood of the working model (3) with and without the random regression QTL effects, $\boldsymbol{Z}_a \boldsymbol{a}$. Stram and Lee (1994) suggest that under H_0, $-2 \log \boldsymbol{\Lambda}$ is distributed as the mixture $\frac{1}{2}(\chi^2_0 + \chi^2_1)$ due to the necessity of testing whether the variance ratio is on the boundary on the parameter space.

If γ_a is found to be significant a putative QTL is determined using an outlier detection method based on the alternative outlier model (AOM) for linear mixed models from Gogel (1997) and formalised in Gogel *et al.* (2001). Verbyla *et al.* (2007) uses the AOM to develop a score statistic for each of the chromosomes. For example, for the kth chromosome let $\boldsymbol{a}_{k0} = \boldsymbol{a}_k + \boldsymbol{\delta}_k$ where $\boldsymbol{\delta}_k$ is a vector of random effects such that $\boldsymbol{\delta}_k \sim N(0, \sigma^2 \gamma_{a,k} \boldsymbol{I}_{m_k - 1})$. The full outlier model is

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\tau} + \boldsymbol{Z}_e \boldsymbol{u}_e + \boldsymbol{Z}_a \boldsymbol{a} + \boldsymbol{Z}_{a,k} \boldsymbol{\delta}_k + \boldsymbol{Z}_g \boldsymbol{p} + \boldsymbol{e}, \tag{4}$$

where $\boldsymbol{Z}_{a,k}$ is the matrix $\boldsymbol{Z}_a$ appropriately subsetted to chromosome k. The REML score is then derived for $\gamma_{a,k}$ and evaluated at $\gamma_{a,k} = 0$, namely

$$U_k(0) = -\frac{1}{2} \left(\mathrm{tr}(\boldsymbol{C}_{k,k}) - \frac{1}{\sigma^2 \gamma_a^2} \tilde{\boldsymbol{a}}_k^T \tilde{\boldsymbol{a}}_k \right), \tag{5}$$

where $\boldsymbol{C}_{k,k} = \boldsymbol{Z}_{a,k}\boldsymbol{P}\boldsymbol{Z}_{a,k}$ with $\boldsymbol{P} = \boldsymbol{H}^{-1} - \boldsymbol{H}^{-1}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{H}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{H}^{-1}$, $\boldsymbol{H} = \sigma^2(\boldsymbol{R} + \boldsymbol{Z}\boldsymbol{G}\boldsymbol{Z}^T + \gamma_a\boldsymbol{Z}_a\boldsymbol{Z}_a^T + \gamma_p\boldsymbol{Z}_p\boldsymbol{Z}_p^T$ and best linear unbiased predictors (BLUPS) $\tilde{\boldsymbol{a}}_k = \gamma_a\boldsymbol{Z}_{a,k}^T\boldsymbol{P}\boldsymbol{y}$. This score has mean zero and this will occur exactly when the terms in the parentheses of (5) are equal. Scores that depart from zero suggest a departure from $\gamma_{a,k} = 0$. A simple statistic that reflects this departure can be based on the "outlier" statistic

$$t_k^2 = \frac{\tilde{\boldsymbol{a}}_k^T\tilde{\boldsymbol{a}}_k}{\sigma^2\gamma_a^2\mathrm{tr}(\boldsymbol{C}_{k,k})} = \frac{\sum_{j=1}^{m_k-1}\tilde{a}_{k:j}^2}{\sum_{j=1}^{m_k-1}\mathrm{var}(\tilde{a}_{k:j})}.$$

This statistic can therefore be calculated from the BLUPS of the QTL sizes and their prediction error variances arising from the working model. In most cases mixed model software, including **ASReml-R** used in **wgaim**, provide the ability to extract these components for this use.

In a similar manner to the above once the chromosome with the largest outlier statistic is identified, the individual intervals within that chromosome are checked. For example if the largest t_k^2 is from the kth chromosome, a similar derivation can be followed for the outlier statistic of the jth interval, namely

$$t_{k:j}^2 = \frac{\tilde{a}_{k:j}^2}{\mathrm{var}(\tilde{a}_{k:j})}.$$

A putative QTL is then determined by choosing the largest $t_{k:j}^2$ within that chromosome. It must be stated at this point that although (4) is formulated to derive the theory for QTL outlier detection there is no requirement to fit this model as the chromosome and interval outlier statistics only contain components obtainable from a fit of the working model proposed in (3). Thus there is only a minimal computational cost to determine an appropriate QTL interval using this method.

Once a QTL interval is selected it is moved into the fixed effects of the working model (3) and the process is repeated until γ_a is not significant. After the selection process is complete the selected QTL intervals appear as fixed effects and the final model is

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\tau} + \sum_{i=1}^{S}\boldsymbol{z}_{a,i}a_i + \boldsymbol{Z}_e\boldsymbol{u}_e + \boldsymbol{Z}_g\boldsymbol{p} + \boldsymbol{e},$$

where $\boldsymbol{z}_{a,i}$ is the appropriate column of $\boldsymbol{Z}_a$ for the ith QTL. This complete approach is known as the WGAIM algorithm.

3. A casual walk through

A typical QTL analysis with **wgaim** can be viewed as series of steps with the appropriate functions

1. Fit a base `asreml()` (see the **ASReml-R** package) model as in (3) but without the added marker/interval genetic information term $\boldsymbol{Z}_a\boldsymbol{a}$. The `asreml()` call allows very complex structures for the variance matrices $\boldsymbol{G}(\boldsymbol{\varphi})$ and $\boldsymbol{R}(\boldsymbol{\phi})$ through its `random` and `rcov` arguments. This makes it an ideal modelling tool for capturing non-genetic experimental variation, such as design components and/or extraneous environmental variation. From

a plant breeding context Verbyla *et al.* (2007) also suggests including the polygenic or residual genetic term $\boldsymbol{Z}_g\boldsymbol{p}$ in the base model as a simple random effect. Examples of base models can be found in Section 4 of this article.

For a comprehensive overview of the **ASReml-R** package, including thorough examples of its flexibility, users should, in the first instance, consult the documentation that is included with the package. On any operating system that has **ASReml-R** installed, the documentation can be found using the simple command `asreml.man()` in R.

2. Read in genetic data using `read.cross()` (see the **qtl** package). This function allows the reading in of genetic information in a number of formats including files generated from commonly used genetic software programs such as Mapmaker and QTL Cartographer. For the exact requirements of all available file types and their nomenclature users should consult the **qtl** documentation.

   The `read.cross()` function can also process more advanced genetic crosses. However, in **wgaim** the QTL analysis is restricted to populations with two genotypes. Thus users should be aware that the class of the returned object from `read.cross()` needs to have the structure `c("bc", "cross")`. The `"bc"` is an abbreviated form for "back-cross". It is this class structure that is checked in the proceeding steps.

3. Convert genetic `"cross"` data to an `"interval"` object using

   ```
   cross2int(fullgeno, missgeno = "MartinezCurnow", rem.mark = TRUE,
     id = "id", subset = NULL)
   ```

   The function contains a number of arguments that allow some linkage map manipulation before calculation of the interval information for each chromosome. If `missgeno = "MartinezCurnow"`, missing values within a chromosome are calculated using the rules of Martinez and Curnow (1992). If `missgeno = "Broman"` the they are calculated using the default values of `argmax.geno()` in the **qtl** package. If `rem.mark = TRUE`, coincident markers across the genome are removed from the marker set. The correlated markers and how they are connected is returned as part of the final object. The `id` is a required argument that determines the names of the unique rows of `fullgeno` and is used for matching names with phenotypic data in the next step. There is also an option to `subset` your map if desired.

   The final genetic data object returned retains the `c("bc", "cross")` class for backward compatibility with other functions in the **qtl** package as well as inherits the class `"interval"` for functionality within the **wgaim** package.

4. Merge the genetic `"interval"` data with the base model phenotypic data using

   ```
   wmerge(geno, pheno, by = NULL, ...)
   ```

   All named arguments of this function are required for a successful merging of genotypic and phenotypic data. The `geno` argument must be a genetic data object inheriting the class `"interval"` from a call to `cross2int()`. The `pheno` can be the usual data frame or a file. If it is a file then it is read in by the wrapper function `asreml.read.table()` which conveniently converts column names with capital letters to factors. The `...` argument can be used as additional arguments to `asreml.read.table()`. The `by` argument is

used for merging `geno` and `pheno` and should be a column name that is present in `geno$pheno` as well as present in one of the columns of `pheno`. There is error trapping in the function if these rules are not adhered to.

By default this function initially merges interval information from different chromosomes or linkage groups to form the fully specified interval matrix $\boldsymbol{M}_\psi$ in (2) with column names "`Chr.<chr>.<int>`". The full genetic matrix, $\boldsymbol{M}_\psi$ is then merged with the phenotypic data which is equivalent to expanding the genetic information using $\boldsymbol{Z}_a = \boldsymbol{Z}_g \boldsymbol{M}_\psi$, ensuring replicates of the same line will have the same genetic structure. It should be noted that unmatched elements of `by` are handled differently depending on whether they are from the `geno` or `pheno` data. If elements of `by` exist in `pheno` and are unmatched with elements in `geno` then they are kept to ensure completeness of the phenotypic data. If elements of `by` exist in `geno` and not in `pheno` they are dropped as there will be no phenotypic information available for that genetic line.

The merged object retains all the same components as the `"interval"` data object with the addition of named components `pheno.dat` representing the phenotypic data only and `full.data` representing the fully merged phenotypic and genotypic data.

5. Perform QTL analysis with `wgaim()`

```
wgaim(baseModel, parentData, TypeI = 0.05, attempts = 5,
  trace = TRUE, ...)
```

The `baseModel` argument must be an `asreml.object` and therefore have `"asreml"` as its class attribute. Thus a call to `wgaim()` is actually a call to `wgaim.asreml()`. This stipulation ensures that an `asreml()` call has been used to form the base model in step 2 before attempting QTL analysis. An error trapping function, `wgaim.default()` is called if the class of the base model is not `"asreml"`. The second argument `parentData` is a data object formed from a call to `wmerge()`. `parentData` must be of class `"interval"` and contain the named component `full.data`. There are initial checks in `wgaim.asreml()` to ensure that `parentData` contains the `baseModel` data. The `TypeI` argument allows users to change the significance level for the testing of QTL effects variance component γ_a. As `asreml()` calls output components of the fit to the screen there is an option to `trace` this to a file if desired.

The fitting of the working model (3) is achieved through added functionality to the `asreml()` call. The merged interval matrix $\boldsymbol{Z}_a$ is added to the base `asreml` model as a contiguous block of random effects with a single variance component, γ_a. If this variance component is significant then the algorithm searches for a QTL. Once a QTL is found it places the appropriate interval column of $\boldsymbol{Z}_a$ into the fixed component of the base model and reiterates this process. Thus the process of finding and selecting QTL using `wgaim()` is automated and may require several model fits. For this reason users must be patient if they are analysing a dataset with a large number of observations or a large number of markers. Upon completion of the algorithm, `summary()` and `print()` methods are available to summarize the QTL.

4. Worked examples

4.1. The zinc data

The zinc data is available in **wgaim** as a usable date set to display the functionality of the package. The data consists of 200 observations of zinc concentration and shoot length for a DH population of wheat. There are two replicates of 90 double haploid lines from a crossing of the wheat varieties Cascades and Rac875-2 and ten each of the parents in an `id` variable. The data also includes a `Type` variable to distinguish the parents from the DH lines. The experiment also contained two blocks in a variable called `Block`.

A suitable base model for shoot length is explored by considering (3) without the random regression effects, $\boldsymbol{Z}_a\boldsymbol{a}$, attributed to genetic markers/intervals. As we are interested in the genetic variance associated with the DH lines, `Type` is modelled as a fixed effect ($\boldsymbol{\tau}$) to ensure the removal of the genetic effect associated with the parents. The `Block` as a random effect of non-interest ($\boldsymbol{u}_e$) and `id` as a set of polygenic random effects $\boldsymbol{p}$.

```
R> data("zinc", package = "wgaim")
R> sh.fm <- asreml(shoot ~ Type, random = ~Block + id, data = zinc)
```

A simple summary of the variance parameters in the model can be achieved with

```
R> summary(sh.fm)$varcomp
```

```
                    gamma  component  std.error   z.ratio constraint
Block!Block.var 0.1902258 0.03721561 0.05539823 0.6717835   Positive
id!id.var       9.1030840 1.78091965 0.28195227 6.3163871   Positive
R!variance      1.0000000 0.19563915 0.02674723 7.3143694   Positive
```

The summary reveals there is only a small difference between Blocks. However, the genetic variance is more than nine times the residual variance of the model making the response an ideal candidate for QTL analysis.

wgaim is prepackaged with a genetic map associated with the zinc data. The data has already been read in using `read.cross()` and can be loaded using

```
R> data("raccas", package = "wgaim")
```

Alternatively to illustrate the use of `read.cross()` in conjunction with this package the same data is available from the `extdata` directory of the package library as a CSV file. A subset of the data from the CSV file is given in Table 1. This reveals that the CSV file is in the rotated CSV format (see `read.cross()` from the **qtl** package). The genotypes are set as `AA` or `AB` and missing values are `"-"`. Thus a call to `read.cross()` is

```
R> wgpath <- system.file("extdata", package = "wgaim")
R> raccas <- read.cross("csvr", file = "raccas.csv",
+    genotypes = c("AA", "AB"), dir = wgpath, na.strings = c("-", "NA"))
R> class(raccas)
```

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
1	id			DH01	DH02	DH03	DH04	DH05	DH06	DH07
2	wmc469	1A1	0.00	AB	AA	AB	-	AB	AB	AB
3	wPt.5914	1A1	7.26	AB	AA	AB	AB	AA	AB	AB
4	wPt.0751	1A1	8.88	-	AA	AB	AB	AA	AB	AB
5	P42.M49.235	1A1	20.11	AA	AA	AB	AB	AA	AB	AB
6	bcd304C	1A2	0.00	AA	AB	AB	AA	AA	AA	AA
7	P31.M55.148	1A2	31.52	AA	-	AB	AA	AA	AA	AA
8	barc213	1A2	42.60	AA	AB	AB	AA	AA	AA	AA
9	P34.M48.83	1A2	43.84	AA	AB	AB	AA	AA	AA	AA
10	gwm99	1A2	54.30	AA	AA	AA	AA	AA	AA	AA

Table 1: A subset of the genetic data from the comma delimited file `raccas.csv`.

```
[1] "bc"    "cross"
```

The returned object has the required class structure and is converted to an `"interval"` object using

```
R> raccas <- cross2int(raccas, missgeno = "Mart", id = "id", rem.mark = TRUE)
R> summary(raccas)
R> class(raccas)
```

```
[1] "bc"    "cross"  "interval"
```

As coincident markers are omitted from the map it is written to a file, `"dummy.csv"`, and read back in using using `read.cross()` to allow a re-estimation of genetic information for the reduced map. The summary shows there is total of 468 markers across 40 linkage groups. It also reveals that just over 5% of markers were missing and imputed using the rules of Martinez and Curnow (1992). The classes of `raccas` and their ordering is retained and it now also inherits the class `"interval"` for use with functions in **wgaim**.

The genetic `"interval"` data can now be merged with the phenotypic zinc data using

```
R> raccasM <- wmerge(raccas, zinc, by = "id")
R> names(raccasM)
```

```
[1] "geno"      "pheno"     "rf"        "pheno.dat" "full.data"
```

This newly merged data retains the same classes as `raccas` and adds a named component `"pheno.dat"` containing the phenotypic data only and `"full.data"` containing the essential merging of the phenotypic `zinc` data with all the chromosomal `"intval"` components of the genotypic `raccas` data.

With this newly merged data a QTL analysis is simply

```
R> zn.qtl <- wgaim(sh.fm, parentData = raccasM, na.method.X = "include")
R> summary(zn.qtl, raccasM)
```

```
  Chromosome Left Marker dist(cM) Right Marker dist(cM)   Size z.ratio  Pr(z)
1        3D2         gdm8    31.51        gdm136    32.64  0.436    4.13      0
2         4B      Rht1mut     54.8          gwm6    70.12   0.54    4.86      0
3        4D1      barc098        0    P42.M49.70     1.13  0.422    3.63  3e-04
4        4D2     wPt.2573        0    Rht2W.type    23.48 -0.383   -2.82 0.0048
```

The analysis reveals four significant QTL in four linkage groups. Verbyla *et al.* (2007) recommends the use of p-values, rather than the commonly used LOD scores, as the overall test of significance for each of the QTL. The argument `LOD = TRUE` can be given to `summary.wgaim()` if LOD scores are necessary.

4.2. Sunco-Tasman data

This example stresses the importance of modelling extraneous variation to a ensure a more appropriate QTL analysis. The Sunco-Tasman data is available in the `data` directory of **wgaim** and contains the results of a field trial conducted in the year 2000 with 175 double haploid lines from a crossing of wheat varieties Sunco and Tasman. The original field trial was arranged in a 31 rows by 12 columns with two replicates of each line. A milling experiment was then performed which replicated 23% of the field samples producing 456 samples milled over 38 mill days with 12 samples per day. The focus is on the trait milling yield.

```
R> stpheno <- asreml.read.table(paste(wgpath, "\\stpheno.csv", sep = ""),
+    header = TRUE, sep = ",")
R> names(stpheno)
```

```
 [1] "X"        "Expt"     "Type"     "id"       "Range"    "Row"
 [7] "Rep"      "Millday"  "Milldate" "Millord"  "myield"   "lord"
[13] "lrow"
```

Smith *et al.* (2006) provides a phenotypic analysis of the data. They give a base model of the form

```
R> st.fmF <- asreml(myield ~ Type + lord + lrow, random = ~ id + Rep +
+    Range:Row + Millday, rcov = ~ Millday:ar1(Millord),
+    data = stpheno, na.method.X = "include")
R> summary(st.fmF)$varcomp
```

```
                   gamma  component  std.error   z.ratio    constraint
id             7.0925458 1.92573995 0.23965934 8.0353220      Positive
Rep            0.2843737 0.07721201 0.15604795 0.4947967      Positive
Range:Row      1.4973306 0.40654927 0.06206771 6.5500926      Positive
Millday        1.7795039 0.48316385 0.15646257 3.0880476      Positive
R!variance     1.0000000 0.27151604 0.08035809 3.3788264      Positive
R!Millord.cor 0.7109431 0.71094307 0.12682697 5.6056142 Unconstrained
```

The model realistically accounts for extraneous plot variation occurring in the field as well as variation due to the design components of the milling experiment. The `lord` and `lrow`

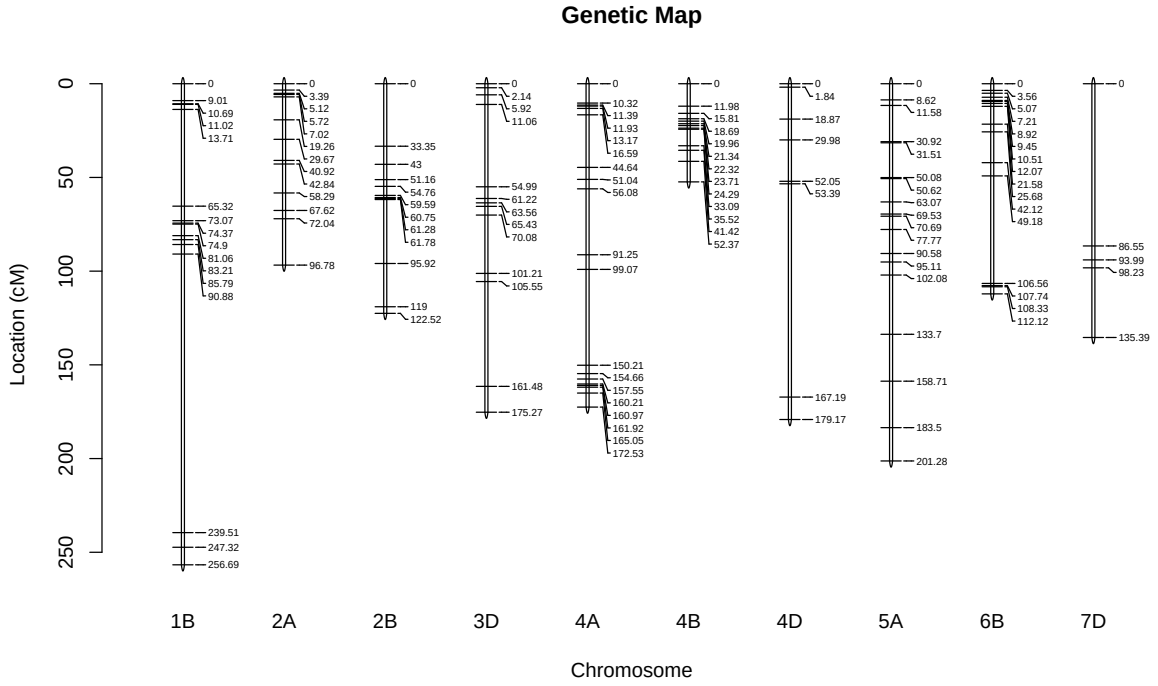

Figure 1: A subset of the genetic map for the Sunco-Tasman data. Names of chromosomes are given at the bottom and genetic distances between markers are placed alongside each of the chromosomes.

components of the fixed model are mean centred covariates of Millord and Row that capture the natural linear trends that occur in the samples across milling order on any given day and across rows in the field. The summary reveals a large genetic variance component. For comparison a NULL model (no extraneous effects) is also fitted.

```
R> st.fmN <- asreml(myield ~ 1, random = ~ id, data = stpheno,
+    na.method.X = "include")
```

The genetic map consists of 287 unique markers across 21 chromosomes and can be read in and converted using

```
R> stmap <- read.cross("csv", file="stgenomap.csv", genotypes=c("A", "B"),
+    dir = wgpath, na.strings = c("-", "NA"))
R> stmap <- cross2int(stmap, missgeno="Bro", id = "id")
R> names(stmap$geno)

 [1] "1A" "1B" "1D" "2A" "2B" "2D" "3A" "3B" "3D" "4A" "4B" "4D" "5A" "5B"
[15] "5D" "6A" "6B" "6D" "7A" "7B" "7D"
```

It is possible to view the genetic map using `link.map()`. The function allows sub-setting according to distance (cM) and/or chromosome. Figure 1 shows the genetic map resulting from

```
R> link.map(stmap, marker.names = "dist", cex = 0.5,
+    chr = c("1B", "2A", "2B", "3D", "4A", "4B", "4D", "5A", "6B", "7D"))
```

For larger maps a more aesthetic plot is reached by adjusting the character expansion (`cex`) parameter and increasing the plotting window width manually.

Merging `stpheno` and `stmap` and performing QTL analysis for the full model `st.fmF` and the null model `st.fmN`

```
R> stmerge <- wmerge(stmap, stpheno, by = "id")
R> st.qtlN <- wgaim(st.fmN, stmerge, na.method.X = "include",
+    trace = "nullmodel.txt")
R> st.qtlF <- wgaim(st.fmF, stmerge, na.method.X = "include",
+    trace = "fullmodel.txt")
```

The process of selecting QTL is determined from the outlier statistics calculations in Section 2. These are saved for each QTL selection and can be viewed using the `out.stat()` command. For the first two iterations of the process the chromosome and interval outliers statistics given in Figure 2 are produced with

```
R> out.stat(st.qtlF, stmerge, int = FALSE, iter = 1:2, cex = 0.6,
+    ylim = c(0, 6.5))
R> out.stat(st.qtlF, stmerge, int = TRUE, iter = 1:2, cex = 0.6)
```

There is also an additional argument that allows the user to subset the genetic map to specific chromosomes which is only available when `int = TRUE`.

Each of these QTL models can be summarised visually using `link.map()`. In this case it calls the method `link.map.wgaim()` to plot the QTL on the genetic map. Multiple models or traits can be handled through `link.map.default()`. For example, Figure 3 is produced with

```
R> link.map.default(list(st.qtlF, st.qtlN), stmerge, marker.names = "dist",
+    cex = 0.6, clist = list(qcol = c("red", "light blue"), mcol = "red",
+    tcol = c("red", "light blue")), trait.labels = c("Full", "Null"))
```

This QTL plotting procedure is highly flexible to user colour changes. Through an argument `clist` it allows the user to specify the QTL colour between markers, the colour of the flanking QTL marker names, the colour of the trait names and the rest of the marker names. If no colours are chosen `qcol` and `tcol` defaults to `rainbow(n)` where n is the number of traits. The QTL map reveals that an extra six QTL were detected in the full model compared to the null model, highlighting the importance of modelling extraneous variation appropriately in QTL analyses.

From a statistical standpoint the QTL selected across the genome cannot be expected to be orthogonal. Thus the introduction of the next QTL in the forward selection process will inevitably affect the significance of the previously selected QTL. A post diagnostic evaluation of the QTL p-values in the forward selection process can be displayed using

```
R> tr(st.qtlF, iter = 1:10, digits = 3)
```

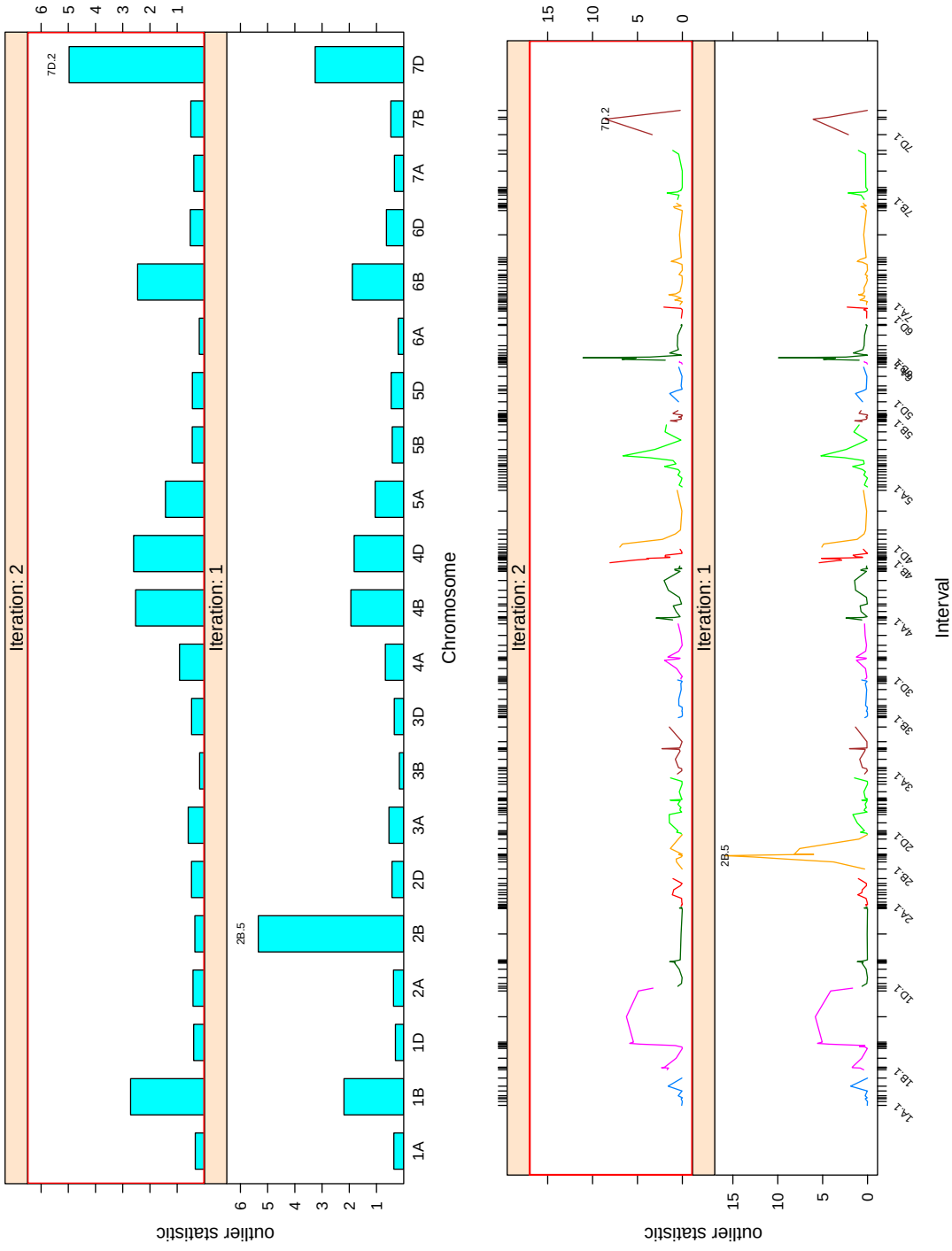

Figure 2: Chromosome and interval outlier statistics for the first two iterations of the wgaim fit for the full model.

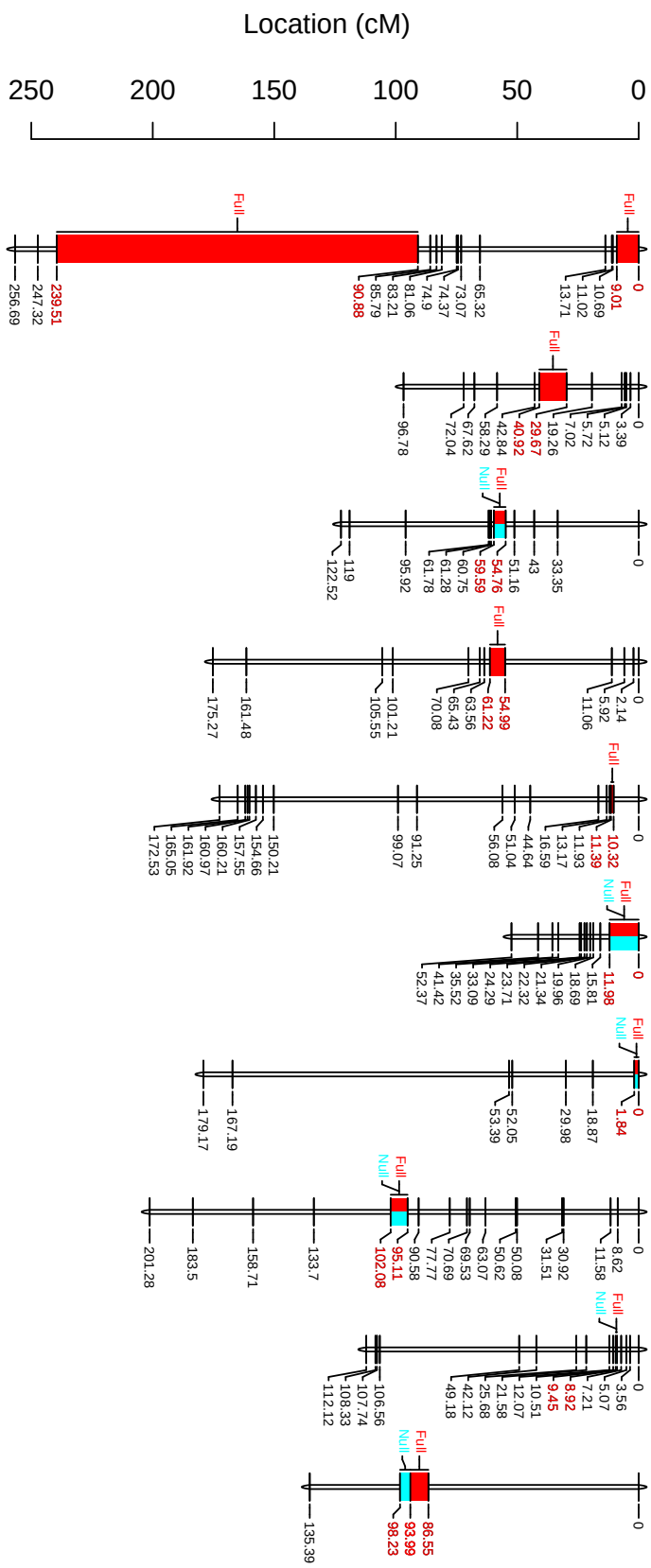

Figure 3: Genetic map with QTL for the Full and Null models obtained from an analysis of the Sunco-Tasman data. Markers and intervals for the QTL are highlighted and trait names are placed on the left hand side of the chromosomes.

```
Incremental QTL P-value Matrix.
===============================
          2B.5   7D.2   4D.1   4B.1  1B.13   6B.5  5A.13   1B.1   4A.2  3D.5
Iter.1  <0.001
Iter.2  <0.001 <0.001
Iter.3  <0.001  0.001 <0.001
Iter.4  <0.001 <0.001 <0.001 <0.001
Iter.5  <0.001 <0.001 <0.001 <0.001 <0.001
Iter.6   0.002 <0.001 <0.001 <0.001 <0.001 <0.001
Iter.7  <0.001 <0.001 <0.001 <0.001 <0.001 <0.001 <0.001
Iter.8  <0.001 <0.001 <0.001 <0.001 <0.001 <0.001  0.002 <0.001
Iter.9   0.001 <0.001 <0.001  0.016 <0.001 <0.001  0.002 <0.001 <0.001
Iter.10 <0.001 <0.001 <0.001 <0.001  0.004 <0.001  0.012 <0.001 <0.001 0.008

Outlier Detection Diagnostic.
=============================
              L0       L1 Statistic Pvalue
Iter.1  -309.563 -250.669   117.787 <0.001
Iter.2  -279.483 -243.213    72.541 <0.001
Iter.3  -272.049 -240.762    62.574 <0.001
Iter.4  -269.746 -238.879    61.734 <0.001
Iter.5  -256.599 -235.993    41.211 <0.001
Iter.6  -251.322 -232.332     37.98 <0.001
Iter.7  -236.172 -225.886    20.572 <0.001
Iter.8  -225.186   -221.5     7.372  0.003
Iter.9  -225.029 -221.577     6.904  0.004
Iter.10  -223.37 -221.047     4.647  0.016
Iter.11 -223.008  -220.78     4.456  0.017
Iter.12 -221.138 -220.029     2.219  0.068
```

The first of these displays shows the p-values of the selected QTL for the first ten iterations occurring in the WGAIM process. An example of the dynamic changes in significance can be seen for the selected QTL interval 4B.1. The introduction of 4A.2 decreases the significance of 4B.1, whereas the introduction of 3D.5 increases it significance. The second display presents the likelihood ratio tests, $-2 \log \mathbf{\Lambda}$, for the significance of the QTL variance parameter, γ_a, in (3), with the inclusion of the last hypothesis test where the null model is retained.

5. Summary

This paper shows the implementation of whole genome average interval mapping algorithm of Verbyla *et al.* (2007) in the R package **wgaim**. The interval mapping approach adopted in **wgaim** requires the conversion of genetic data objects created from the **qtl** package. The package also uses the sophisticated linear mixed modelling software **ASReml-R** for QTL analysis thus allowing users with added flexibility to simultaneously model sources of genetic and non-genetic variation through the addition of highly structured random effects and/or possible correlation between observations. Selected QTL can be easily summarized and checked

for their significance as well as plotted on a linkage map for visual inspection of their location on the genome.

Currently only QTL analysis of univariate traits is possible with **wgaim**. However, a multivariate version of the WGAIM algorithm is being researched and preliminary papers have been submitted (Verbyla and Cullis 2011; Verbyla *et al.* 2011). The software implementation of this multivariate approach is currently being tested. Research is currently being conducted to determine the inclusion of higher order effects such as epistatic interactions into the WGAIM approach. We are hopeful that these new approaches will be implemented in future releases of **wgaim**.

References

Ball R (2010). ***BayesQTLBIC**: Bayesian Multi-Locus QTL Analysis Based on the BIC Criterion*. R package version 1.0-1, URL http://CRAN.R-project.org/package=BayesQTLBIC.

Broman KW, Sen S (2009). *A Guide to QTL Mapping with R/**qtl***. Springer-Verlag, New York.

Broman KW, Wu H (2010). ***qtl**: Tools for Analayzing QTL Experiments*. R package version 1.15-15, URL http://CRAN.R-project.org/package=qtl.

Butler DG, Cullis BR, Gilmour AR, Gogel BJ (2009). "**ASReml-R** Reference Manual." *Technical report*, Queensland Department of Primary Industries. URL http://www.vsni.co.uk/software/asreml/.

Gilmour AR (2007). "Mixed Model Regression Mapping for QTL Detection in Experimental Crosses." *Computational Statistics & Data Analysis*, **51**, 3749–3764.

Gilmour AR, Gogel BJ, Cullis BR, Thompson R (2009). ***ASReml** User Guide*. Release 3.0, URL http://www.vsni.co.uk/software/asreml/.

Gilmour AR, Thompson R, Cullis BR (1995). "Average Information REML: An Efficient Algorithm for Variance Parameter Estimation in Linear Mixed Models." *Biometrics*, **51**, 1440–1450.

Gogel BJ (1997). *Spatial Analysis of Multi-Environment Variety Trials*. Ph.D. thesis, Department of Statistics, University of Adelaide.

Gogel BJ, Welham SJ, Verbyla AP, Cullis BR (2001). "Outlier Detection in Linear Mixed Effects; Summary of Research. Report P106." *Technical report*, University of Adelaide, Biometrics.

Hayley CS, Knott SA (1992). "A Simple Regression Method for Mapping Quantitative Trait Loci in Line Crosses Using Flanking Markers." *Heredity*, **69**, 315–324.

Huang B, George A (2009). "Look Before You Leap: A New Approach to Mapping QTL." *Theoretical and Applied Genetics*, **119**, 899–911.

Kang HM, Zaitlen NA, Wade CM, Kirby A, Heckerman D, Daly MJ, Eskin E (2008). "Efficient Control of Population Structure in Model Organism Association Mapping." *Genetics*, **178**, 1709–1723.

Martinez O, Curnow RN (1992). "Estimating the Locations and Sizes of the Effects of Quantitative Trait Loci Using Flanking Markers." *Theoretical and Applied Genetics*, **85**, 480–488.

Oakey H, Verbyla AP, S PW, Cullis BR, Kuchel H (2006). "Joint Modelling of Additive and Non-Additive Genetic Line Effects in Single Field Trials." *Theoretical and Applied Genetics*, **113**, 809–819.

Patterson HD, Thompson R (1971). "Recovery of Interblock Information When Block Sizes Are Unequal." *Biometrika*, **58**, 545–554.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistcial Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Schwarz G (1978). "Estimating the Dimension of a Model." *The Annals of Statistics*, **6**, 461–464.

Seaton G, Haley CS, Knott SA, Kearsey M, Visscher PM (2002). "**QTL Express**: Mapping Quantitative Trait Loci in Simple and Complex Pedigrees." *Bioinformatics*, **18**, 339–340.

Shriner D, Yi N (2009). "Deviance Information Criterion (DIC) in Bayesian Multiple QTL Mapping." *Computational Statistics & Data Analysis*, **53**, 1850–1860.

Smith A, Cullis BR, Thompson R (2001). "Analysing Variety by Environment Data Using Multiplicative Mixed Models." *Biometrics*, **57**, 1138–1147.

Smith A, Cullis BR, Thompson R (2005). "The Analysis of Crop Cultivar Breeding and Evaluation Trials: An Overview of Current Mixed Mode Approaches." *Journal of Agricultural Science*, **143**, 449–462.

Smith AB, Lim P, Cullis BR (2006). "The Design and Analysis of Multi-Phase Plant Breeding Programs." *Journal of Agricultural Science*, **144**, 393–409.

Stram DO, Lee JW (1994). "Variance Components Testing in the Longitudinal Mixed Effects Model." *Biometrics*, **50**, 1171–1177.

Verbyla AP, Cullis BR, Thompson R (2007). "The Analysis of QTL by Simultaneous Use of the Full Linkage Map." *Theoretical and Applied Genetics*, **116**, 95–111.

Verbyla AP, Cullis BR (2011). "Multivariate Whole Genome Average Interval Mapping: QTL Analysis for Multiple Traits and/or Multiple Environments." *Theoretical and Applied Genetics, Submitted*.

Verbyla AP, Hackett CA, Newton AM, Taylor WBT, Cullis BR (2011). "Multi-Treatment QTL Analysis Using Whole Genome Average Interval Mapping." *Theoretical and Applied Genetics, Submitted*.

Whittaker JC, Thompson R, Visscher PM (1996). "On the Mapping of QTL by Regression of Phenotype on Marker-Type." *Heredity*, **77**, 22–32.

Xu S (2003). "Estimating Polygenic Effects using Markers of the Entire Genome." *Genetics*, **164**, 789–801.

Yandell BS, Mehta T, Banerjee S, Shriner D, Venkataraman R, Moon JY, Neely WW, Wu H, Smith R, Yi N (2005). "R/**qtlbim**: QTL with Bayesian Interval Mapping in Experimental Crosses." *Bioinformatics*, **23**, 641–643.

Zeng ZB (1994). "Precision Mapping of Quantitative Trait Loci." *Genetics*, **136**, 1457–1468.

Zhang M, Zhang D, Wells M (2008). "Variable Selection for Large p Small n Regression Models with Incomplete Data: Mapping QTL with Epistases." *Bioinformatics*, **9**.

Zhou Q (2010). "Review of 'A Guide to QTL Mapping with R/**qtl**'." *Journal of Statistical Software, Book Reviews*, **32**(5), 1–3. URL http://www.jstatsoft.org/v32/b05/.

Affiliation:

Julian Taylor
Mathematics, Informatics & Statistics
CSIRO
PMB 2, Glen Osmond, SA, 5064, Australia
E-mail: julian.taylor@csiro.au

Arunas Verbyla
Mathematics, Informatics & Statistics
CSIRO
PMB 2, Glen Osmond, SA, 5064, Australia
and
School of Agriculture, Food and Wine
The University of Adelaide
PMB 1, Glen Osmond, SA, 5064, Australia
E-mail: ari.verbyla@csiro.au

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 40, Issue 7 *Submitted:* 2010-07-13
April 2011 *Accepted:* 2011-02-25