mppR: An R Package for QTL Analysis in Multi-parent Populations

Vincent Garin
Wageningen University

Valentin Wimmer
KWS SAAT SE

Dietrich Borchardt
KWS SAAT SE

Fred van Eeuwijk
Wageningen University

Marcos Malosetti
Wageningen University

Abstract

mppR is an add-on package for the statistical software R for QTL analyses in multi-parent populations composed of genotypes from more than one cross like NAM populations, diallels or factorial designs. **mppR** contains functions to assist the user in a range of activities of QTL analysis such as: data processing, QTL detection, visualisation of results, and estimation of QTL effects. **mppR** workflow is structured along main functions allowing to: 1) perform preliminary data quality control; 2) organize data into a single data object; 3) cluster parental lines based on ancestry; 4) perform QTL detection; 5) evaluate QTL discoveries by cross-validation; and 6) determine multi-QTL effect models. The search of QTLs can be done by 4 different models that vary with respect to the way the QTL effects are modelled (cross-specific, parental, ancestral or bi-allelic).

Keywords: Multi-parent populations, quantitative trait loci, mixed models, R.

1. Introduction

Quantitative trait locus (QTL) analysis essentially consists in finding a relationship between DNA polymorphisms (e.g., SNPs) and phenotypic variation (Doerge 2002). QTL detection methods greatly depends on the genetic properties of the population that is used. Historically, QTL detection has been performed in designed experimental populations involving two parental lines (bi-parental crosses). Several methods and software packages have been developed for QTL analysis for such populations, for a review see Varshney *et al.* (2015). Multi-parent populations (MPPs) are an alternative type of population that can improve the chances of QTL detection while broadening the range of research questions that can be answered (Cavanagh *et al.* 2008). MPPs can be seen as a compromise between bi-parental crosses and association panels (Myles *et al.* 2009). Different types of MPPs have been developed including nested association mapping (NAM) populations (McMullen *et al.* 2009), diallels (Blanc *et al.* 2006) and factorial designs (Bardol *et al.* 2013). More complicated MPPs can be created by intercrossing multiple founders followed by inbreeding, like in multi-parent advanced generation inter-cross (MAGIC) populations (Cavanagh *et al.* 2008). Here, we consider MPPs as a collection of genotypes that are derived by crosses between at least three different parents. In this paper, a MPP QTL analysis is akin to the joint analysis of

such a population using a common marker map.

The development of an appropriate statistical methodology taking MPP properties into consideration is a *sine qua non* condition to fully exploit the potential of MPP genetic resources. The most critical question is how to account for genetic relatedness between the genotypes and how to integrate this information into the statistical model. A first simple option is to treat MPPs as an association mapping panel and to apply genome-wide association study (GWAS) QTL detection methods.

The use of a GWAS type of method presents several advantages. First, GWAS methods apply to almost any type of MPP design because the knowledge of population structure is not a prerequisite. The GWAS QTL detection is marker-based, that is, uses identity by state (IBS) information, and so generally only allows for two alleles at each tested position. A second advantage is the existence of powerful algorithms (e.g. EMMA - Kang *et al.* (2008)) that allow to scan in a reasonable amount of time large marker datasets. Finally, GWAS analyses are also based on sets of well-developed mixed models (Yu *et al.* 2006). These models allow to account for the population structure and the polygenic effect using a kinship matrix covering the relations between all genotypes from all populations (van Eeuwijk *et al.* 2010; Rincent *et al.* 2014). MPP GWAS type of QTL detection can be performed using packages like **TASSEL** (Bradbury *et al.* 2007) the R library **GAPIT** (Lipka *et al.* 2012) or the R packages **GenABEL** (Aulchenko *et al.* 2007) and **Sommer** (Covarrubias-Pazaran 2016).

However, a major limitation of GWAS methods is that they generally use bi-allelic marker models assuming two classes of effects at the QTL position. The bi-allelic assumption represents therefore a risk of failing to reflect the allele diversity potentially present at the QTLs within MPPs (Garin *et al.* 2017). Indeed, several factors like multiple alleles, cross-specific linkage phase between marker and QTL alleles, or interaction effects between the QTL and genetic background may cause complex allelic series.

Other approaches use available pedigree information to model or infer DNA transmission to the final lines starting from a set of parents or ancestors. This strategy make use of identity by descent (IBD) information and gives model with more than two alleles, which can be more appropriate to model complex allelic series (Blanc *et al.* 2006; Xavier *et al.* 2015). For example, the R package **NAM** (Xavier *et al.* 2015) proposes to take into consideration factors which can lead to complex allelic series like the difference of linkage phase association between marker and QTL in different crosses. **NAM** uses incidence matrices containing the number of alleles received per parent to estimate random QTL effects and to control for the polygenic background in the rest of the genome. The software package **MCQTL** (Jourjon *et al.* 2005) functioning in a Linux environment is also an option. **MCQTL** combined with the R package **clusthaplo** (Leroux *et al.* 2014) computes linear models with various assumptions about the origin and number of QTL alleles (cross-specific, parental, or ancestral).

We add a the bi-allelic model and incorporate a cross-validation strategy to evaluate the QTL detection performance of the different models. We also developed a method to build multi-QTL (MQE) models that allows QTLs with different types of effects at different loci in contrast to the more rigid approach in **MCQTL** (Jourjon *et al.* 2005) that assumes the same type of effect across the genome. In a nutshell, **mppR** fits a wide range of models with different assumptions about the QTL effects. The current version of **mppR** is based on the linear model. A more general version of **mppR** is available from the following GitHub repository `https://github.com/vincentgarin/mppR`. In a mixed model context, we can

allow for heterogeneity of variance present in MPPs, and other sources of random variation and/or dependence existing between genotypes. The **mppR** components that are based on mixed model technology depend on the **ASReml-R** package (Butler *et al.* 2009) and require a license.

This manual is organized as follows. Section 2 describes the statistical methodology for the proposed MPP QTL detection procedures. Section 3 illustrates in details the QTL detection procedure describing the different functions using a subset of the maize US-NAM population as example (Yu *et al.* 2008; McMullen *et al.* 2009).

2. Statistical methodology

2.1. Connectivity

mppR allows to analyse any type of MPP design with minimally two crosses between at least three different parents. In such designs, the possibility to estimate QTL parameters (identifiability) is linked to the notion of connectivity of the design. It is always possible to estimate one effect per cross. Therefore, the number crosses (n_c) constitutes the largest number of QTL effects that can be estimated. The estimation of parental and ancestral effects is linked to the connectivity of the MPP design (Rebaï and Goffinet 2000). Taking for example the parental alleles, it is possible to estimate $n_p - 1 \leq n_c$ parental alleles per connected part of the design. Design connectivity can be defined using graph theory (Weeks and Williams 1964). Following graph theory, an MPP design can be represented by a graph where parents (alleles) are vertices or nodes and crosses are edges or lines (Figure 1). A connected graph, is a graph where there exists a walk from any node i to any other node j. For example, the MPP design in Figure (1) is composed of two connected parts. Ideally, MPP QTL analysis should be run using only connected populations. The joint analysis of an MPP composed of several disjunct but internally connected parts is still possible. In that case, connectedness could follow from the sharing of a common ancestor by two parents of the design. For example if parents P_B and P_E of the MPP design of Figure (1) would receive their allele from the same ancestor, the MPP design would consist of a single connected part. For a bi-allelic model we assume that the design is fully connected. In any case, even if disconnected parts are analysed jointly, a minimal level of connection will be assumed by assuming that cofactors are shared in the whole population.

2.2. General model

We propose to describe the QTL detection model following the assumptions made on the form of the QTL effect and allele origin. Let us start by defining the following underlying single locus QTL detection model describing the relationship between the phenotypic values and genotypes coming from several crosses (Rebaï and Goffinet 1993):

$$y_{ijk} = \mu_{ij} + \alpha_i + \alpha_j + g_{ij} + e_{ijk} \tag{1}$$

where y_{ijk} represents the phenotypic value for the k^{th} individual from the cross between parents i and j. μ_{ij} is the cross mean and α_i and α_j represent the effects associated with the QTL alleles coming from parent i and j respectively. The QTL effects are assumed to

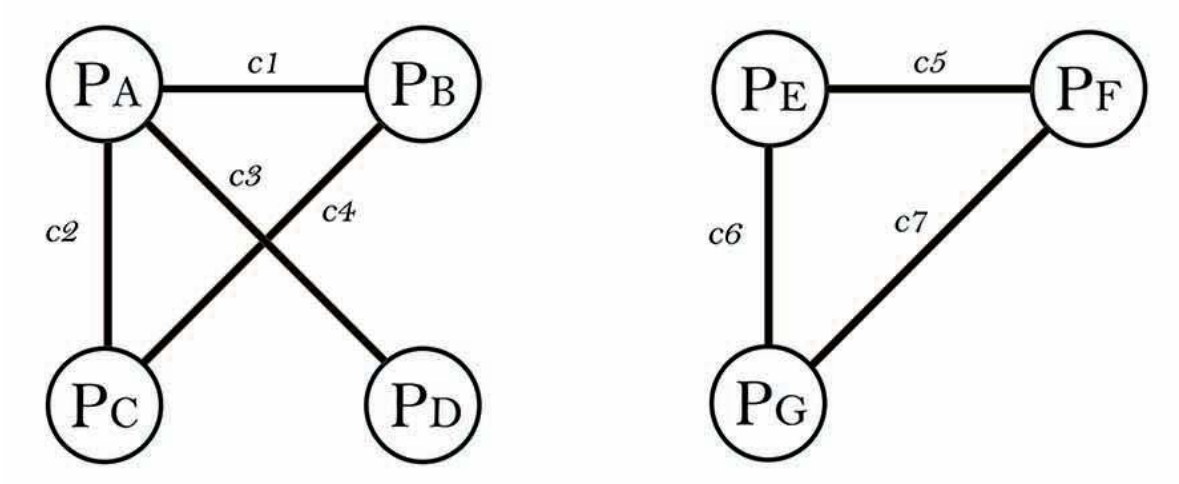

Figure 1: Example of a MPP design represented as a graph

be strictly additive (no dominance, no epistasis). g_{ij} is the random polygenic effect due to QTLs elsewhere in the genome with distribution $N(0, \sigma_g^2)$. Finally, e_{ijk} represents the random micro-environmental effect (plot error) having distribution $N(0, \sigma_e^2)$. In this model, σ_g^2 and σ_e^2 are unique meaning that the level of polygenic effect and environmental error is considered to be the same in each cross.

Model (1) can be rewritten in matrix notation:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{r} \tag{2}$$

where, $\boldsymbol{y}$ is the $[N \times 1]$ vector of phenotypic values. $N = \sum_{c=1}^{n_c} N_c$ where N_c is the number of genotypes coming from cross i. $\boldsymbol{X} = [\boldsymbol{X_c}|\boldsymbol{X_Q}]$ is the fixed effect incidence matrix and $\boldsymbol{\beta}' = [\boldsymbol{\beta}'_{\boldsymbol{c}}|\boldsymbol{\beta}'_{\boldsymbol{Q}}]$ the vectors of cross intercepts and QTL effects. $\boldsymbol{X}$ is composed of a part that links observations to the particular cross it belongs to ($\boldsymbol{X_c}$ an $[N \times n_c]$ matrix with n_c representing the number of crosses) and $\boldsymbol{X_Q}$ the part related with the QTL effect attached to the particular observation. $\boldsymbol{X_Q}$ is a matrix of dimensions $[N \times n_{al}]$ with n_{al} the number of QTL alleles that are assumed to segregate for the particular QTL locus. Several assumptions are possible concerning n_{al}. They correspond to different statistical models presented in the next section. The form of $\boldsymbol{X_Q}$ varies according to the type of QTL effect assumed. Finally, $\boldsymbol{r}$ represents the vector of random residual terms with distribution $N(0, \boldsymbol{R})$.

2.3. QTL effects

The QTL effect incidence matrix $\boldsymbol{X_Q}$ is the central term of the model. Assuming a diploid organism, the individual elements of $\boldsymbol{X_Q}$, x_{nl} take values between 0 and 2 and represent the expected number of copies of allele l with $l = 1, ..., n_{al}$ received by genotype n at the QTL position. The column number of $\boldsymbol{X_Q}$ (n_{al}) varies with the number of alleles assumed at the QTL position. We propose four models: cross-specific, parental, ancestral, and bi-allelic. These models correspond to different assumptions concerning the type of QTL effects and the allele origin. They are characterized by different ways to model genetic relatedness between genotypes using either IBD estimates, IBS information, or a combination of both.

Cross-specific model

The first model assumes that the QTL alleles that segregate within a particular cross are different from those that segregate in another cross. Cross-specific QTL effects can be seen as parental alleles interacting with the cross genetic background. Under this assumption, QTL alleles are nested within crosses and so QTL effects are estimated per cross. In the cross-specific model, $x_{nl} \in [0, 2]$ represents the expected number of allele copies received from one of the cross parents given the flanking markers. The expected number of parental allele copies is estimated using IBD probabilities computed by the package R/**qtl** (Broman *et al.* 2003). These probabilities are estimated with respect to the parents of each cross. For illustration purpose, let us take the following example of a MPP analysis combining material coming from two crosses: cross 1 ($P_A \times P_B$) and cross 2 ($P_A \times P_C$). In that case, we ignore the fact that the two crosses are connected since they share a common parent P_A. Therefore $\boldsymbol{X_Q}$ is a diagonal block structure with diagonal elements specifying the within cross allele origin. Model (2) can be re-written like that

$$\boldsymbol{y} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \boldsymbol{\beta_c} + \begin{array}{c} \begin{matrix} P_A\,P_B & \quad P_A\,P_C \end{matrix} \\ \begin{pmatrix} \begin{matrix} 2 & 0 \\ 1 & 1 \\ 0 & 2 \end{matrix} & \boldsymbol{0} \\ \boldsymbol{0} & \begin{matrix} 2 & 0 \\ 1 & 1 \\ 0 & 2 \end{matrix} \end{pmatrix} \end{array} \boldsymbol{\beta_Q} + \boldsymbol{r} \tag{3}$$

It is not possible to estimate two effects per cross since the parental scores are linearly dependent. The design matrix for the QTL effect in a cross-specific model is therefore constrained by redefining the parental information of a cross as half the difference between parent i and parent j. Therefore, for the cross-specific model, $\boldsymbol{X_Q}$ is of dimension $[N \times n_c]$ where n_c is the number of crosses and the vector $\boldsymbol{\beta_Q}$ is of dimension $[n_c \times 1]$. The cross-specific model contains the upper limit for the number of QTL effects that can be estimated. Indeed, in connected MPPs, the maximum number of effects that can be estimated is $n_c \geq n_p - 1$ where n_p is the number of parents (Rebaï and Goffinet 2000; Jansen *et al.* 2003). This model corresponds to the disconnected model described in Blanc *et al.* (2006).

Parental model

In the cross-specific model, all crosses are considered unrelated. A second option is the parental model that adds the connection between crosses via the parents shared between crosses. In that case, the parental QTL incidence matrix is simply obtained by re-arranging the columns of model (3) taking into consideration the connections created by the use of common parents. This model estimates one allele effect per parental line, which is considered to be independent of the genetic background. The QTL effect of parent p is assumed to be constant in all crosses where this parent has been used (Blanc *et al.* 2006).

In a connected MPP, if $(n_p - 1) < n_c$, one expects the parental model to be more powerful than the cross-specific model because the number of parameters to estimate is reduced (Blanc *et al.* 2006). The reduction in the number of parameters to estimate should also help to get better estimates of the QTL effects because the sample size used to estimate these effect

increases (Li *et al.* 2005). Full half diallels, with at least four parents, represent the most connected system where the number of crosses $n_c = (n_p * (1 - n_p))/2$ is maximised with respect to the number of parents (Jansen *et al.* 2003). Coming back to the previous example (3), we integrate in the QTL incidence matrix the fact that the two crosses are connected via the common parent P_A.

$$\boldsymbol{y} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \boldsymbol{\beta_c} + \begin{array}{c} \begin{matrix} P_A & P_B & P_C \end{matrix} \\ \begin{pmatrix} 2 & 0 & \\ 1 & 1 & 0 \\ 0 & 2 & \\ 2 & & 0 \\ 1 & 0 & 1 \\ 0 & & 2 \end{pmatrix} \end{array} \boldsymbol{\beta_Q} + \boldsymbol{r} \tag{4}$$

In the parental effect model, the matrix $\boldsymbol{X_Q}$ is of dimension $[N \times n_p]$ and $\boldsymbol{\beta_Q}$ is of dimension $[n_p \times 1]$. The parental model corresponds to the connected model described in Blanc *et al.* (2006).

Ancestral model

The third option, called ancestral model, goes one level up in the pedigree and uses relatedness between parents to cluster them into a reduced number of ancestral groups. We assume that parents belonging to the same cluster transmit the same allele (Jansen *et al.* 2003; Leroux *et al.* 2014). Different options can be used to cluster parental lines. One of them is the R package **clusthaplo** (Leroux *et al.* 2014). **clusthaplo** is an algorithm to cluster parental lines along the genome based on genetic similarity. **clusthaplo** uses a sliding window to define ancestral classes at each marker position based on local genetic similarities using marker scores within the window. If the local marker density is not large enough, **clusthaplo** uses the global genetic similarity defined by a kinship coefficient. **mppR** contains a function to call **clusthaplo**.

The ancestral QTL incidence matrix $\boldsymbol{X_Q^*}$ can be obtained by modifying the parental IBD QTL incidence matrix $\boldsymbol{X_Q}$ using **clusthaplo** results. The ancestral model uses therefore both IBD and parental relatedness IBS information. Continuing our example (4), let us assume that at the considered QTL position parents A and C belong to the same ancestral group A_1, and parent B falls apart in group A_2.

$$\boldsymbol{X_Q^*} = \boldsymbol{X_Q} \times \boldsymbol{A} = \begin{array}{c} \begin{matrix} P_A & P_B & P_C \end{matrix} \\ \begin{pmatrix} 2 & 0 & \\ 1 & 1 & 0 \\ 0 & 2 & \\ 2 & & 0 \\ 1 & 0 & 1 \\ 0 & & 2 \end{pmatrix} \end{array} \times \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{array}{c} \begin{matrix} A_1 & A_2 \end{matrix} \\ \begin{pmatrix} 2 & 0 \\ 1 & 1 \\ 0 & 2 \\ 2 & \\ 2 & 0 \\ 2 & \end{pmatrix} \end{array} \tag{5}$$

The matrix $\boldsymbol{X}_{\boldsymbol{Q}}^{*}$ of the ancestral model is of dimension $[N \times n_a]$ where n_a is the number of ancestral alleles. The corresponding vector $\boldsymbol{\beta}_{\boldsymbol{Q}}^{*}$ is of dimension $[n_a \times 1]$. The elements of $\boldsymbol{\beta}_{\boldsymbol{Q}}^{*}$ represent the estimates of the ancestral additive effects. The ancestral-effect model corresponds to the LDLA models used by Bardol *et al.* (2013) and Giraud *et al.* (2014).

Parental and ancestral model constraints

The estimation of parental (ancestral) QTL effect also requires the application of a constraint to the QTL incidence matrix. From a theoretical point of view, it is possible to estimate maximally $n_p - 1$ ($n_a - 1$) QTL effects per connected part of the design (Rebaï and Goffinet 2000; Weeks and Williams 1964). Therefore, the QTL effects are estimated setting to zero the most frequent parental (ancestral) allele within each connected part. For example in the example of Figure (1), we could have P_A set as reference of the first connected part and P_E set as reference of the second connected part. An alternative is to force the QTL effects to sum to zero. The sum to zero constraint will also take place within each connected part.

Bi-allelic model

The last possibility is the bi-allelic model. If the marker is at the QTL position, the bi-allelic model assumes that genotypes with the same SNP score transmit the same allele. Therefore, we assume that the same allele segregates in the whole population which connects all parts of the design that were not connected before. Genetic relatedness is therefore defined based on marker IBS information only. In this model, using the most frequent allele set as reference, $\boldsymbol{X}_{\boldsymbol{Q}}$ become a vector $[N \times 1]$ with values 0, 1 or 2 corresponding to the number of copies of the minor allele.

$$\boldsymbol{y} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \boldsymbol{\beta}_{\boldsymbol{c}} + \begin{matrix} SNP_1 \\ \begin{pmatrix} 2 \\ 1 \\ 0 \\ 1 \\ 2 \\ 0 \end{pmatrix} \end{matrix} \boldsymbol{\beta}_{\boldsymbol{Q}} + \boldsymbol{r} \tag{6}$$

This model corresponds to association mapping models (e.g., model B in Würschum *et al.* (2012)). In a connected MPP, if $(n_p - 1) < n_c$, the models can be ordered from less to more parsimonious models (cross-specific, parental, ancestral, bi-allelic).

2.4. Variance covariance structure

A second important part of the QTL detection problem is the data variance covariance structure (VCOV). Several assumptions are possible concerning the VCOV of model (2) $\boldsymbol{V} = Var(\boldsymbol{y}) = Var(\boldsymbol{r}) = \boldsymbol{R}$

Homogeneous residual term

The simplest form that can be assumed for $\boldsymbol{V}$ is a homogeneous residual term (HRT) variance. In this case, $\boldsymbol{R} = \boldsymbol{I_N}\sigma_r^2$. This corresponds to a linear model where residual terms are considered to be independent and to belong to the same distribution. In the HRT model, the variance of the polygenic term (σ_g^2) and the error variance (σ_e^2) of model (1) are both pooled in the unique variance residual term σ_r^2.

Other models are possible for the VCOV. You can find model with alternative VCOV in a more general version of **mppR** available from the following GitHub repository `https://github.com/vincentgarin/mppR`. You can find a description these models in the attached vignette.

2.5. Test statistics

The significance of the QTL effects $\hat{\boldsymbol{\beta}}_{\boldsymbol{Q}}$ can be estimated using the Wald test (Wald 1943). In the case of a HRT model, after simplification, the Wald test can be rewritten like that:

$$W(\hat{\boldsymbol{\beta}}) = \boldsymbol{y}'\hat{\boldsymbol{V}}^{-1}\hat{\boldsymbol{y}} \tag{7}$$

$W(\hat{\boldsymbol{\beta}})$ is distributed as a chi-squared distribution with the degrees of freedom being equal to the number of QTL alleles. Expression (7) shows that the test statistic depends on the correlation between the observed phenotypic values ($\boldsymbol{y}$) and the model fitted values ($\hat{\boldsymbol{y}}$) and on the estimated VCOV ($\hat{\boldsymbol{V}}$). For that reason, the choice of the QTL incidence matrix ($\boldsymbol{X_Q}$) should be such that the phenotypic variations is captured as accurately as possible. If these variations are due to parental or cross-specific effects, corresponding QTL effects should perform better at the price of a larger number of parameters to estimate. On the other hand, if the effects are similar through the MPP, a reduced number of parameters will capture this variability and allows gains in power. The VCOV structure should also be selected to reflect local patterns of variability. If heterogeneity is present between crosses, the CSRT model will give test statistics considering this heterogeneity.

2.6. QTL detection procedure

The QTL detection procedure proposed in **mppR** is based on the following steps: a) Optional significance threshold determination by permutation test (Churchill and Doerge 1994); b) cofactors selection by simple interval mapping (SIM); c) multi-QTL model search using composite interval mapping (CIM) (Zeng 1993, 1994); d) simultaneous evaluation of the selected candidate QTL positions after backward elimination.

Significance threshold determination

The QTL significance threshold can be determined by permutation. The use of permutation aims at reproducing the conditions of the null hypothesis (no QTL present or no association between the marker and the QTL) by breaking the link between the phenotype and the genetic markers (Churchill and Doerge 1994). Permutations allow to build a null hypothesis for the test statistic that reflects the characteristics of the experiment and should be valid for any distribution of the quantitative trait (Churchill and Doerge 1994). The number of permutations should be at least 1000. Alternatively, the user can also specify the significance threshold value.

Multi-QTL model determination

The determination of a multi-QTL model is done using CIM. Such a strategy is based on fitting the model using cofactors representing other QTLs than the tested QTL (Zeng 1993, 1994). The selection of cofactors is based on a SIM scan using the following model where X_c is a cross-specific intercept and X_Q model the QTL effect.

$$\boldsymbol{y} = \boldsymbol{X_c}\boldsymbol{\beta_c} + \boldsymbol{X_Q}\boldsymbol{\beta_Q} + \boldsymbol{r} \tag{8}$$

Cofactors can be selected with minimum distance in between based on the -log10(p) value SIM profile. CIM profile is computed based on the following model

$$\boldsymbol{y} = \boldsymbol{X_c}\boldsymbol{\beta_c} + \boldsymbol{X_q}\boldsymbol{\beta_q} + \boldsymbol{X_Q}\boldsymbol{\beta_Q} + \boldsymbol{r} \tag{9}$$

where $\boldsymbol{X_q}$ represents the selected cofactors. During the CIM scan, an exclusion window can be set around the tested QTL position to remove cofactors and avoid too strong collinearity between the cofactors and the tested position. Finally, the selected candidate QTLs can be simultaneously tested after a backward elimination. An optional confidence interval for each QTL position can be obtained using a -log10(p) value drop-off interval taking a CIM profile and excluding cofactors on the scanned chromosome.

2.7. Multi-QTL effect (MQE) model

A variation on the common QTL model with a single type of effect is the multi-QTL effect (MQE) model. In the MQE model, the QTLs present in the final model can have different types of effects (cross-specific, parental, ancestral or bi-allelic). To build an MQE model we use a forward selection procedure. For each QTL to be added, genome wide profiles with a consistent QTL effect are calculated for each types of QTL effects that have been specified by the user. The model that is fitted at each position within a profile is:

$$\boldsymbol{y} = \boldsymbol{X_c}\boldsymbol{\beta_c} + \boldsymbol{X_{Q1}}\boldsymbol{\beta_{Q1}} + \boldsymbol{r} \tag{10}$$

Where the (first) QTL ($\boldsymbol{X_{Q1}}\boldsymbol{\beta_Q}$) has an effect that is either cross-specific, parental, ancestral, or bi-allelic along the genome. From each of these profiles, the most significant position based on the -log10(p) value statistic is selected (e.g., $\boldsymbol{X_{Q1.cr}}$, $\boldsymbol{X_{Q1.par}}$, $\boldsymbol{X_{Q1.anc}}$, $\boldsymbol{X_{Q1.biall}}$). Note that the selected QTL positions for the different types of effects can be different. The QTL that increases most the R^2_{adj} (14) is selected, with it type of effect, and added to the model as a cofactor for the next set of genome wide scans. If at step 1 we selected a bi-allelic QTL, then at step 2 the QTL profiles will be based on the following models:

$$\boldsymbol{y} = \boldsymbol{X_c}\boldsymbol{\beta_c} + \boldsymbol{X_{q1.biall}}\boldsymbol{\beta_{q1}} + \boldsymbol{X_{Q2}}\boldsymbol{\beta_{Q2}} + \boldsymbol{r} \tag{11}$$

For the set of QTL effects specified by the user, genome wide scans are performed via a test for the QTL effect in the term $\boldsymbol{X_{Q2}}\boldsymbol{\beta_{Q2}}$. The forward selection process stops when no further significant QTLs can be identified. At this point, a final list of QTLs is compiled by a backward elimination. A final model with t QTLs could look like:

$$\boldsymbol{y} = \boldsymbol{X_c}\boldsymbol{\beta_c} + \boldsymbol{X_{Q1.biall}}\boldsymbol{\beta_{Q1}} + ... + \boldsymbol{X_{Q(t-1).par}}\boldsymbol{\beta_{Q(t-1)}} + \boldsymbol{X_{Qt.anc}}\boldsymbol{\beta_{Qt}} + \boldsymbol{r} \tag{12}$$

2.8. QTL effect estimation

Once a final list of QTLs is determined, the estimates for the regression coefficients in the corresponding multi-QTL model provide the QTL effects. For this model a global goodness of fit can also be calculated using R^2. Partial R^2 statistics are indicators of the contributions of each individual QTLs.

Genetic effect estimation and interpretation

In the cross-specific model (2.3.1), the genetic predictors for the additive effects represent half the difference between the second parent and the first parent for their conditional QTL genotype probabilities. The elements of $\boldsymbol{\beta_Q}$ represent the within cross allele substitution effect.

For the parental (2.3.2) and the ancestral (2.3.3) models, from a theoretical point of view, it is possible to estimate a maximum of $n_p - 1$ ($n_a - 1$) QTL effects per connected part of the design (Rebaï and Goffinet 2000). Within each connected part, the most frequent parental (ancestral) allele is used as reference. The estimated parental (ancestral) QTL effects must be interpreted as a deviation with respect to the connected part reference. Referring again to the example in Figure (1), if P_A is set as reference of the first connected part, its value will be zero and the other parent alleles effects (P_B, P_C and P_D) will represent deviations with respect to P_A allele. If the second part (P_E, P_F, and P_G) was analysed jointly with the first part, the effect of alleles P_F and P_G will represent deviation with respect to P_E and will be independent of the first part parental effects.

An alternative is to use a sum to zero constraint. In that case the parental (ancestral) QTL effects are forced to sum to zero. The individual QTL effects represent a deviation with respect to the central tendency. Here also, the constraints are defined within each connected part. For the bi-allelic model (2.3.5), the estimated genetic effect represents the additive effect of one allele copy of the minor allele with respect to the most frequent allele set as reference.

Global R^2

To compute the R^2 statistics, we compare the residual sums of squares of a full model with QTL(s), to the one of a reduced model without QTLs.

$$R^2 = 1 - \frac{\sum_{n=1}^{N} r^2_{n(full)}}{\sum_{n=1}^{N} r^2_{n(red)}} \tag{13}$$

The R^2 measurement can be adjusted to take into consideration the number of parameters used.

$$R^2_{adj} = 1 - \frac{\sum_{n=1}^{N} r^2_{n(full)}/df_{full}}{\sum_{n=1}^{N} r^2_{n(red)}/df_{red}} \tag{14}$$

Where df_{full} and df_{red} represent the degrees of freedom of the full and reduced model, respectively.

Partial R^2

For each single QTL, a partial R^2 can be calculated by the difference between the R^2 of the full model (all QTL positions) and the R^2 of the model that drops that particular QTL (difference R^2). **mppR** also calculates partial R^2 by comparing a model without QTLs with single locus QTL models (single R^2). These estimates can also be adjusted using formula (14). The partial R^2 is an estimation of the individual QTL contribution to the phenotypic variation. The difference and single R^2 give estimates of the lower and upper bound explained variance by individual QTLs.

2.9. Cross-validation

A cross-validation (CV) approach can be used to evaluate the performance of the QTL detection process and to assess the QTL effect in a pseudo-independent population (Utz *et al.* 2000). CV allows to assess predictability of QTL effects in data not used for model training. The proposed CV procedure is an adaptation of Utz *et al.* (2000) procedure to the MPP context. A single run of CV is composed of the following steps:

1. **Partitioning of the dataset**. The full dataset ($\boldsymbol{y_{DS}}, \boldsymbol{X_{DS}}$) is partitioned *within cross* into k subsets. Then for the k repetitions each k subset is successively used as validation set (VS) ($\boldsymbol{y_{VS}}, \boldsymbol{X_{VS}}$), the other $(k-1)$ subsets go into the training set (TS) ($\boldsymbol{y_{TS}}, \boldsymbol{X_{TS}}$).

2. **Explained genetic variance in the TS**. The training set is used to detect QTLs. These QTLs allow to evaluate the proportion of explained genetic variance in the TS using $\hat{p}_{TS} = \frac{R^2_{adj.TS}}{h^2}$ where $R^2_{adj.TS}$ is the adjusted R^2 (14) and h^2 the heritability to be specified by the user.

3. **Predicted genetic variance in the VS**. We can now use the estimated QTL effects in the TS ($\hat{\boldsymbol{\beta}}_{\boldsymbol{TS}}$) to predict phenotypic values in the VS ($\hat{\boldsymbol{y}}_{\boldsymbol{VS}} = \boldsymbol{X_{VS}}\hat{\boldsymbol{\beta}}_{\boldsymbol{TS}}$). The proportion of predicted genetic variance in the VS is $\hat{p}_{VS} = \frac{R^2_{VS}}{h^2}$, where R^2_{VS} is the squared Pearson correlation between the observed and predicted values. $\hat{p}_{VS}$ is computed within cross. A measurement at the whole MPP level is obtained by calculating the weighted average of the within cross values ($\bar{p}_{VS}$) accounting for the cross sizes. The relative bias between $\hat{p}_{TS}$ and $\bar{p}_{VS}$ is $1 - (\frac{\bar{p}_{VS}}{\hat{p}_{TS}})$.

3. Illustration: US-NAM population QTL analysis

3.1. Overview

mppR contains functions to perform all steps of an MPP QTL analysis, starting from the data processing to the visualisation of results. Figure 2 shows a schematic representation of the **mppR** workflow. The first part concerns the raw data processing to gather all required data in a single `mppData` object. The `mppData` object can be used in the functions: `mpp_proc()`, `mpp_CV()` and `MQE_proc()` which are generic functions to perform QTL analyses, cross-validation, and multi-QTL effect model computations respectively.

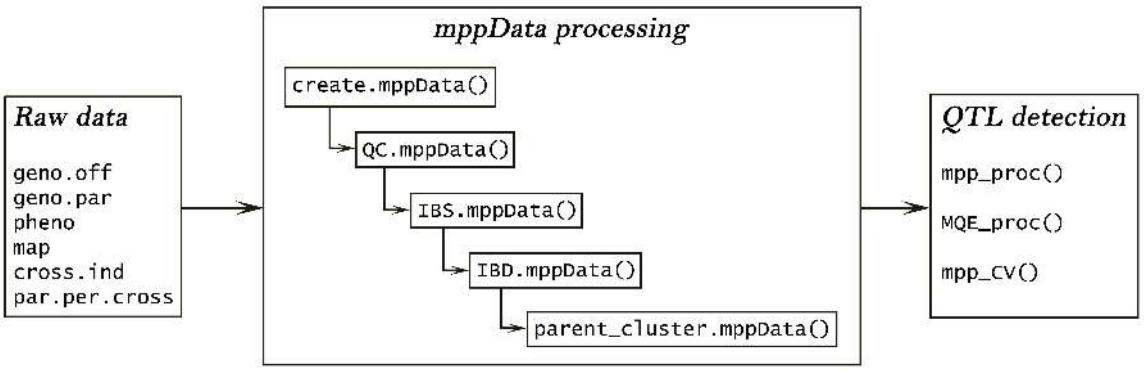

Figure 2: mppR workflow.

3.2. US-NAM data and raw data format

We included a subset of the maize US-NAM population (McMullen *et al.* 2009) within the **mppR** package as example dataset to illustrate the different functions. We introduce these data and the required format to be used in the workflow presented in Figure 2.

```
> library(mppR)
```

The data consist of three parts obtained from www.panzea.org. `USNAM_geno` is a random sample of the US-NAM population including the marker information of 506 genotypes and 102 markers. The entries include the parents and 500 recombinant inbred lines (RIL) coming from 5 crosses between the central line B73 and the peripheral parents CML103, CML322, CML52, Hp301, and M37W.

```
> data("USNAM_geno", package = "mppR")
> dim(USNAM_geno)

[1] 506 102

> rownames(USNAM_geno)[1:6]

[1] "B73"    "CML103" "CML322" "CML52"  "Hp301"  "M37W"

> table(substr(rownames(USNAM_geno)[-c(1:6)], 1, 4))

Z002 Z006 Z008 Z010 Z016
 100  100  100  100  100
```

Genotypic data

Raw genotypic data used in **mppR** must be bi-allelic markers. The genotypic data are expected to be formatted as a `character matrix`, with one letter for each allele. So, when using the

ACTG coding all possible genotypes are AA, CC, AC, etc. Missing values must always be coded `NA`. We impose a strict data format to make mppR functioning smoothly. Any deviation from this format will produce an informative error message.

To start the data processing, the genotypic data must be split into offspring and parent genotype scores. These matrices represent the `geno.off` and `geno.par` arguments in the `create.mppData()` function. For these two arguments, the genotypes define the rows with genotype identifiers as row names and the markers are in columns with the marker identifiers as column names. The order of the markers must be the same as the one in the `map` argument. The `geno.off` genotypes list must also be in the same order as the one of the `pheno` argument.

```
> geno.off <- USNAM_geno[7:506, ]
> geno.par <- USNAM_geno[1:6, ]
```

Map data

`USNAM_map` is a three columns genetic map with marker indicator, chromosomes and map positions given in centi-Morgan (cM). It has the required format for the argument `map` in function `create.mppData()`. The marker identifiers must be `character`. The chromosomes and genetic positions must be `numeric`. The list of markers must be column names of the `geno.off` and `geno.par` arguments.

```
> data("USNAM_map", package = "mppR")
> head(USNAM_map)
```

```
  mk.names chr pos.cM
1   L00411   1    0.0
2   L00569   1    3.7
3   L00068   1    9.7
4   L01003   1   13.4
5   L00196   1   15.6
6   L00609   1   17.9
```

```
> map <- USNAM_map
```

Phenotypic data

The file `USNAM_pheno` is a `numeric matrix` containing the phenotypic measurements of 500 offspring genotypes for the trait upper leaf angle (ULA). The row names represent the genotype identifiers. They must be identical to the row names of `geno.off`. `USNAM_pheno` can be used as `pheno` argument for the `create.mppData()` function. The `pheno` argument can contain several traits. A cross indicator `character vector` can be formed by subsetting the genotype names. It specifies to which cross each genotype belongs and can be used for the `cross.ind` argument in `create.mppData()`.

```
> data("USNAM_pheno", package = "mppR")
> head(USNAM_pheno)
```

```
          ULA
Z002E0001  75
Z002E0002  55
Z002E0005  60
Z002E0010  70
Z002E0011  75
Z002E0012  70
```

```
> pheno <-  USNAM_pheno
> cross.ind <- substr(rownames(pheno), 1, 4)
```

Cross parents information

The last raw data source is provided via the `par.per.cross` argument. It is a three column `character matrix` with one row per cross specifying: 1) the cross indicator that must be identical and appear in the same order with the one used in `cross.ind` ; 2-3) the parents 1 and 2 of the crosses. The parents' identifiers must be identical to the row names of `geno.par`.

```
> par.per.cross <- cbind(unique(cross.ind), rep("B73", 5),
+                        rownames(geno.par)[2:6])
```

The `par.per.cross` matrix can be used in the `design_connectivity()` function to obtain and visualize the connected parts. For example using the illustration of Figure (1), we have

```
> ppc_ex <- cbind(paste0("c", 1:7),
+                 c("PA", "PA", "PB", "PA", "PE", "PE", "PG"),
+                 c("PB", "PC", "PC", "PD", "PF", "PG", "PF"))
> design_connectivity(ppc_ex)

$`1`
[1] "PA" "PB" "PC" "PD"

$`2`
[1] "PE" "PF" "PG"
```

For the next part of the illustration, we assume that the `geno.off`, `geno.par`, `map`, `pheno`, `cross.ind`, and `par.per.cross` objects are loaded in the global environment.

3.3. Data processing

The initial step is the processing of the raw data to gather all required data in a single `mppData` object. The functions `create.mppData()`, `QC.mppData()`, `IBS.mppData()`, `IBD.mppData()`, and `parent_cluster.mppData()` must be called in the defined sequence to form a complete `mppData` object. Any deviation from this sequence will be signaled by an error message.

create a mppData object

The function `create.mppData()` creates a unified `mppData` object containing all raw data sources.

```
> mppData <- create.mppData(geno.off = geno.off, geno.par = geno.par,
+                           map = map, pheno = pheno, cross.ind = cross.ind,
+                           par.per.cross = par.per.cross)

mppData object created!

500 genotypes
5 crosses
6 parents
1 phenotype(s)
1 connected part
```

Marker quality control - QC

Before QTL analysis, the raw data should go through a quality control (QC). This procedure will ensure that marker format is correct and that markers are informative, meaning that their segregation rate is sufficient to provide a reliable basis to investigate and estimate the QTL effects. The function `QC.mppData()` performs a default QC.

The user should be aware that it is difficult to propose general settings for QC that will be suitable for all MPPs. Moreover, the type of model fitted should also guide the QC. For example, if the user wants to give more emphasis to the cross-specific model, the QC should ensure that there is enough within cross segregation. On the other hand, the bi-allelic model assumes that the QTLs segregate in the whole population. Therefore for this model, minimum segregation can be evaluated at the whole population level. The procedure implemented in `QC.mppData()` is the following:

1. Remove markers with more than two alleles.
2. Remove markers that are monomorphic or fully missing in the parents.
3. Remove markers with a missing rate higher than `mk.miss` across the entire MPP.
4. Remove genotypes with more missing markers than `gen.miss`.
5. Remove crosses with less than `n.lim` genotypes.
6. Keep only the most polymorphic marker when multiple markers map at the same position.
7. Filter markers based on minor allele frequency (MAF). Different options are possible.

   A) The first one filter marker based on MAF at the whole population level, and/or on MAF within crosses. The markers with a MAF below a threshold given by `MAF.pop.lim` at the whole population level will be discarded.

   The user can specify the critical values for MAF within cross using `MAF.cr.lim`. By default, the within cross MAF values are defined by the following function of the cross-size N_c: $MAF(N_c) = 0.5$ if $N_c \in [0, 10]$ and $MAF(N_c) = (4.5/N_c) + 0.05$ if $N_c > 10$. This means that up to 10 genotypes, the critical within cross MAF is set to 50%. Then it decreases when the number of genotype increases until 5% set as a lower bound.

If the within cross MAF is below the limit in at least one cross, then marker scores of the problematic cross are either put as missing (`MAF.cr.miss = TRUE`) or the whole marker is discarded (`MAF.cr.miss = FALSE`). By default, `MAF.cr.miss = TRUE`, which allows to include a larger number of markers and to cover a wider genetic diversity.

B) An alternative is to select only markers that segregate in at least one cross at the `MAF.cr.lim2` rate.

```
> mppData <- QC.mppData(mppData = mppData, n.lim = 15, MAF.pop.lim = 0.05,
+                       MAF.cr.miss = TRUE, mk.miss = 0.1,
+                       gen.miss = 0.25, verbose = TRUE)

Check genotyping error                          : 0 markers removed
Remove monomorphic/missing marker in parents  : 2 markers removed
Remove marker with missing rate < 0.1         : 2 markers removed
Remove genotype with missing rate < 0.25      : 2 genotypes removed
Remove crosses with less than 15 observations : 0 genotypes removed
Remove markers at the same position           : 0 markers removed
Remove markers with MAF < 0.05                : 0 markers removed

   End             : 98 marker(s) remain after the check
                     498 genotypes(s) remain after the check
```

IBS processing

To compute a bi-allelic model, the user needs to convert genotype data into IBS format using `IBS.mppData()`. This function transforms genotype scores into 0, 1, 2 coding where the score represents the number of minor allele copies. The user can also perform imputation of the missing values (`impute = TRUE`). Different options are available, some of them rely on the `codeGeno()` function from the package **synbreed** (Wimmer *et al.* 2012).

```
> mppData <- IBS.mppData(mppData = mppData, impute = TRUE,
+                        impute.type = 'random')

     Summary of imputation
    total number of missing values               : 1737
    number of random imputations                 : 1737

>
```

IBD processing

The other models (cross-specific, parental, and ancestral) use IBD probabilities. The function `IBD.mppData()` estimates IBD probabilities after converting the marker genotype data into within cross ABH format. For each cross, the maker scores of the two cross parents are used as reference. Homozygous offspring genotype scores similar to parent 1 get score `"A"`

("B" for parent 2). Heterozygous genotypes are scored "H". If at least one of the parents is missing or the parents are monomorphic, the offspring will receive NA. The regular ABH conversion assumes that the reference parent scores are fully homozygous or missing. However, if some parent marker scores are heterozygous, the ABH conversion can still be performed setting argument `het.miss.par = TRUE`. In that case, when a parent score is heterozygous or missing and the other parent is homozygous, the function will try to infer the score of the allele that was transmitted by the heterozygous or the missing parent looking at the segregation pattern. Then the computation of the IBD probabilities is done by calling the function `calc.genoprob()` of the R/**qtl** package (Broman *et al.* 2003). For that purpose a temporary csv file, which can be loaded by R/**qtl**, will be saved at the location specified in `dir`.

The type of population must be specified in argument `type`. Different population types are possible: F-type ("F"), back-cross ("BC"), backcross followed by selfing ("BCsFt"), double haploid ("DH"), and recombinant imbred lines ("RIL"). The number of F and BC generations can be specified using `F.gen` and `BC.gen`. The agument `type.mating` specifies if F and RIL populations were obtained by selfing or by sibling mating. DH and RIL populations are read as back-cross by R/qtl. For these two population types, heterozygous scores will be treated as missing values.

```
> mppData <- IBD.mppData(mppData = mppData, het.miss.par = TRUE, type = 'RIL',
+                        type.mating = 'selfing')

 --Read the following data:
        498  individuals
        98  markers
        1  phenotypes
 --Cross type: bc
```

Parent clustering

The final step of data processing is to integrate the parent clustering information to the `mppData` object using the function `parent_cluster.mppData()`. The clustering of the parental lines is necessary to calculate the ancestral model. If the parent clustering is skipped, the other models (cross-specific, parental, bi-allelic) can still be computed.

At a single marker position, two parents can be grouped into a similar ancestral classes if we assume that they receive there allele from a common ancetor. The parent clustering information (`par.clu`) describe parental relatedness and which parent belong to which ancestral group. For example, at marker i, we could have five parents (pA, pB, pC, pD, pE) and the following clustering information (1, 2, 1, 2, 3). This means that pA and pC received their allele from the same ancetor (A1). pB and pD also have a shared ancestor (A2) who is different from (A1). And pE was not included in any group and can be seen as an indepedent ancestral group (A3).

The parent clustering information is provided via `par.clu`. It is an `interger matrix` with markers in row and parents in columns. At a particular marker position, parents with the same value are assumed to inherit from the same ancestor.

The parent clustering can be performed using the R package **clusthaplo** that can be found there: https://cran.r-project.org/src/contrib/Archive/clusthaplo/. The clusthaplo option is not integrated in this version of mppR. However, a version of mppR with function calling clusthaplo can be found on github https://github.com/vincentgarin/mppR.

```
> data("par_clu", package = "mppR")
> mppData <- parent_cluster.mppData(mppData = mppData, par.clu  = par_clu)
>
```

A summary of the `mppData` objects can be obtained calling the generic function `summary()`.

```
> summary(mppData)

object of class 'mppData'

         Type of population:  Recombinant inbred line by selfing

         No. Genotypes:  498

Crosses  Z002   Z006   Z008  Z010  Z016
Parent 1 B73    B73    B73   B73   B73
Parent 2 CML103 CML322 CML52 Hp301 M37W
N        100    100    98    100   100

         Phenotype(s):  ULA
         Percent phenotyped:  100

         Total marker:  98
         No. markers: 56 42
```

3.4. mppData manipulation

Subsets from `mppData` objects can be obtained using the generic fuction `subset()`. The `mppData` objects can be subsetted by markers (`mk.list`) and/or by genotypes (`gen.list`).

```
> mppData_sub <- subset(x = mppData, mk.list = mppData$map[, 2] == 1,
+                       gen.list = sample(mppData$geno.id, 200))
```

3.5. QTL analysis

When all data elements are ready, the user can start the QTL analysis. In all functions involving the computation of a QTL model (`mpp_CIM()`, `mpp_CV()`, `mpp_perm()`, `mpp_proc()`, `mpp_SIM()`, `MQE_proc()`, `QTL_gen_effects()`, etc.), the arguments `Q.eff` describes the type of QTL effect. `Q.eff` takes the values `"cr"`, `"par"`, `"anc"`, `"biall"` for the cross-specific, parental, ancestral, and bi-allelic model, respectively. A complete QTL analysis can be performed using the generic function `mpp_proc()`. This is a wrapping function for individual

functions, indicated in parenthesis, performing each a part of the following QTL detection procedure:

1. SIM scan to select cofactors (`mpp_SIM()`). In that case, we fit an ancestral model (`Q.eff = "anc"`).
2. Cofactor selection with SIM -log10(p) value above the `threshold` value with a minimum distance (`win.cof`) between selected positions (`QTL_select()`). The cofactor selection procedure is done per chromosome. It first selects the most significant position and then removes the positions in the neighbourhood of the selected position from the candidate list of QTLs/cofactors. The process continues until no position is significant anymore. The threshold (`thre.cof` or `thre.QTL`) values can be determined by permutation using `mpp_perm()`.
3. CIM scan (`mpp_CIM()`) using the selected cofactors except within an exclusion window (`window`) around the selected cofactors where they are excluded from the model. It is possible to perform several consecutive run of CIM using `N.cim` .
4. Selection of QTL candidates with CIM -log10(p) value above `thre.QTL` and a minimum distance (`win.QTL`) between the selected positions (`QTL_select()`). The selection procedure is the same as for the cofactors.
5. If `backward = TRUE`, backward elimination on the list of selected QTL positions (`mpp_back_elim()`).
6. Estimation of the QTLs genetic effects (`QTL_gen_effects()`), the global and partial QTLs R^2 (`QTL_R2()`).
7. If `CI = TRUE`, computation of QTL confidence intervals from a CIM- profile (excluding cofactors on the scanned chromosome). The confidence interval is based on a -log10(p) value drop-off value (`drop`).
8. plot of the -log10(p) value CIM QTL profile (`plot.QTLprof()`) and if `plot.gen.eff = TRUE`, visualisation of the genome-wide significance of the QTL effect per cross or per parent.

```
> QTL_proc <- mpp_proc(pop.name = "USNAM", trait.name = "ULA", trait = "ULA",
+                      mppData = mppData, Q.eff = "anc",
+                      plot.gen.eff = TRUE, N.cim = 1, thre.cof = 3,
+                      win.cof = 20, window = 20, thre.QTL = 3,
+                      win.QTL = 20, CI = TRUE, drop = 1.5,
+                      verbose = FALSE, output.loc = tempdir())
```

The results of `mpp_proc()` are returned as a list of R objects. These results are also saved in different files at the location specified in argument `output.loc`. The created folder contains a report (QTL_REPORT.txt) with a summary of results such as the number of detected QTL, the global R^2, and for each QTL the estimated genetic effects per cross or parent.

3.6. Multi QTL Effect (MQE) model

A multi-QTL effects model 2.7 can be determined using `MQE_proc()`. The user has to specify the types of tested QTL effects in the argument `Q.eff`. The `window` argument specifies the

distance on both sides of an already detected QTL position where the search will be forbidden. A backward elimination on the final list of detected QTLs can be performed using (`backward = TRUE`). The results of the last MQE CIM run can be plotted using the function `MQE_plot()`. This `MQE_plot()` will colour the QTL positions corresponding to the type of QTL effect assumed at the position. This will be automatically done by `MQE_proc()` if `plot.MQE = TRUE`. The plot (plot_MQE.pdf) will be saved with the other results at `output.loc`.

```
> MQE <- MQE_proc(pop.name = "USNAM", trait.name = "ULA", mppData = mppData,
+                 Q.eff = c("par", "anc", "biall"), window = 20, verbose = FALSE,
+                 output.loc = tempdir())
```

3.7. QTL effects estimation

Once a list of QTL candidates has been determined, it is possible to estimate the QTL effect per cross or per parents (parental, ancestral, and bi-allelic model) using the function `QTL_gen_effects()`. For the cross-specific model (`Q.eff = "cr"`) (2.3.1), the effects are given in absolute value and represent the substitution effect of one allele copy from the parent increasing the trait.

For the parental and ancestral models (`Q.eff = "par"` (2.3.2) or `Q.eff = "anc"` (2.3.3)), the QTL effects are given per parents and must be interpreted as deviation with respect to most frequent allele within the connected part set as reference. For the parental and ancestral models, the parental alleles are listed from the most (top) to the least frequent (bottom). For the ancestral model, parents with the same score correspond to the same ancestral allele according to **clusthaplo** results.

The QTL effects can also be calculated using the sum to zero constraint (`sum_zero = TRUE`). In that case individual parental (ancestral) allele effect represent deviation with respect to the average allelic effects. In a NAM population, all crosses are connected via the central parent (B73), which is the most frequent allele, and was therefore set as reference.

```
> SIM <- mpp_SIM(mppData = mppData, Q.eff = "anc")
> cofactors <- QTL_select(Qprof = SIM)
> CIM <- mpp_CIM(mppData = mppData, Q.eff = "anc", cofactors = cofactors,
+                plot.gen.eff = TRUE)
> QTL <- QTL_select(Qprof = CIM)
> gen.eff <- QTL_gen_effects(mppData = mppData, QTL = QTL, Q.eff = "anc")
> summary(gen.eff, QTL = 1)

QTL effects
***********

Number of QTL(s): 1


QTL 1
-----
```

```
 mk.names chr pos.cM
   L00929   2      1


QTL effect per cross or parent:

          Effect   Std.Err    t-test      p-value Sign Con.part Par.all
B73     0.000000 0.0000000  0.000000 1.000000e+00             c1      AA
CML103 -1.636412 0.7505413 -2.180310 2.971851e-02    *       c1      CC
CML52  -1.636412 0.7505413 -2.180310 2.971851e-02    *       c1      CC
Hp301  -2.688274 0.7321971 -3.671517 2.681344e-04  ***       c1      CC
M37W   -2.688274 0.7321971 -3.671517 2.681344e-04  ***       c1      CC
CML322 -4.864590 1.0477228 -4.643012 4.435899e-06  ***       c1      CC
```

For the bi-allelic model (2.3.5), the estimated genetic effect represents the additive effect of the minor allele with respect to the most frequent one, the latter set as reference. When parental genotype information is given, the results are given for each parent by multiplying the allele additive effect by the number of parent allele copies.

Results visualisation

A QTL profile can be obtained by passing the `mpp_SIM()` or `mpp_CIM()` results to the `x` argument of the function `plot.QTLprof()` (Figure 3). QTL or cofactors positions can also be plotted on the graph (dashed lines) using the argument `QTL`.

```
> plot(x = CIM, QTL = QTL, type = "l")
```

`Mpp_SIM()` or `mpp_CIM()` results obtained with `plot.gen.eff = TRUE` can also be passed to the function `plot.QTLprof()` with argument `gen.eff = TRUE` to obtain a visualisation of the genetic effect distribution along the genome (Figure 4). Once again, the positions passed to the `QTL` argument will be drawn on the graph.

```
> plot(x = CIM, gen.eff = TRUE, mppData = mppData, QTL = QTL, Q.eff = "anc")
```

The interpretation of the genetic effect plot depends on the type of QTL effects. For a cross-specific model, red colour means that the allele coming from parent 1(A) increases the trait value, blue means that allele coming from parent 2(B) increases the trait.

For the parental and ancestral models, the effect must be interpreted as deviation with respect to the reference within a connected part. The reference allele is always defined as the most frequent allele. It can change at different positions. Therefore, it is not possible to establish a unique reference allele for the whole genome. The parental alleles significances are assessed per connected part. The red (blue) colour means that the parental allele decrease (increase) the trait value. Parents are ordered from the top to the bottom given the number of times their allele was set as reference in the whole genome. For example the upper parent was the one for which its allele was the highest number of times set as reference in the whole genome. These plots should be interpreted as a rough indication of signal distribution.

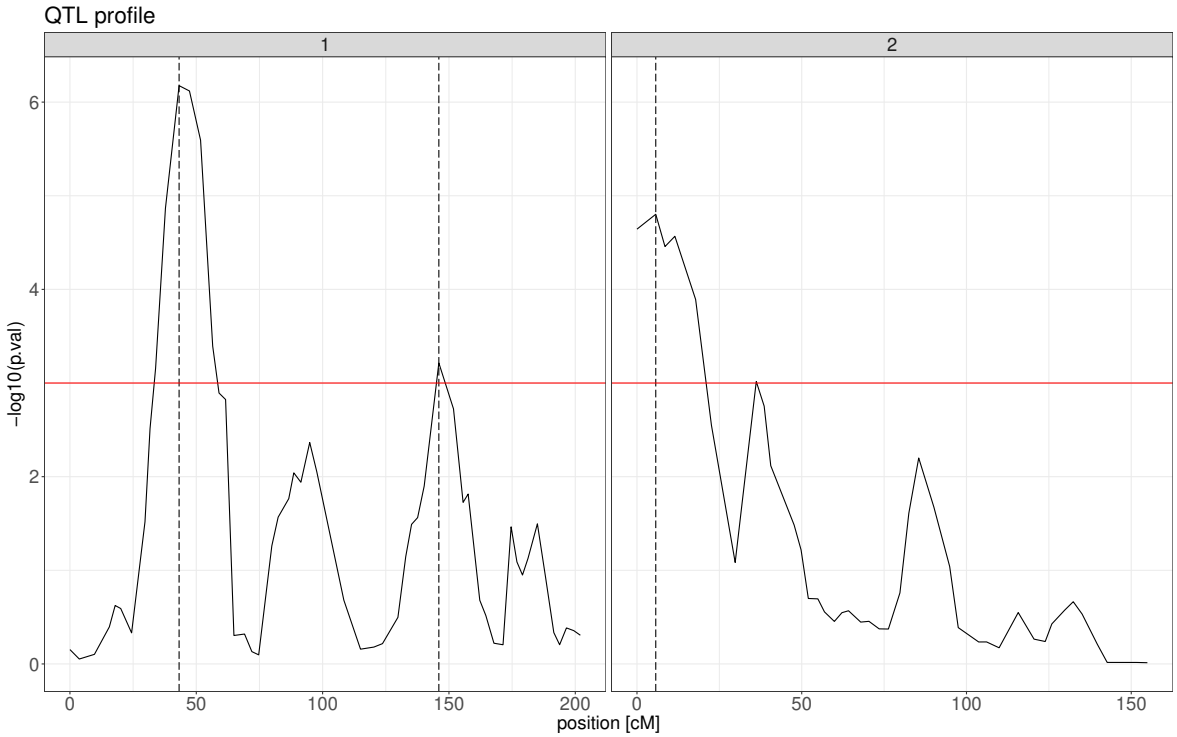

Figure 3: CIM QTL profile from an ancestral model using HRT. The cofactors positions are represented by vertical dashed lines.

3.8. Cross-validation

The cross-validation procedure (2.9) can be performed by the function `mpp_CV()`. The arguments `Rep` and `k` represent the number of repetitions of the k-fold procedure. The heritability, allowing to express R^2 in terms of percentage of the explained genotypic variance, can be specified in `her`. By default `her = 1`, therefore the results are expressed in terms of phenotypic variation. The results of the CV procedure will be saved in a folder at `output.loc`. A transparency plot of the CV QTL profiles (Figure 5) will also be saved at `output.loc`.

```
> set.seed(89341)
> CV <- mpp_CV(pop.name = "USNAM", trait.name = "ULA", mppData = mppData,
+              Q.eff = "cr", her = 0.4, Rep = 1, k = 3, verbose = FALSE,
+              output.loc = tempdir())
```

3.9. Parallelization

All functions involving genome scan(s) (`mpp_perm()`, `mpp_SIM()`, `mpp_CIM()`, `mpp_proc()`, `mpp_CV()` or `MQE_proc()`) can be executed in parallel. The number of cores can be specified using `n.cores`. Parallelization is done using functions from the **parallel** library.

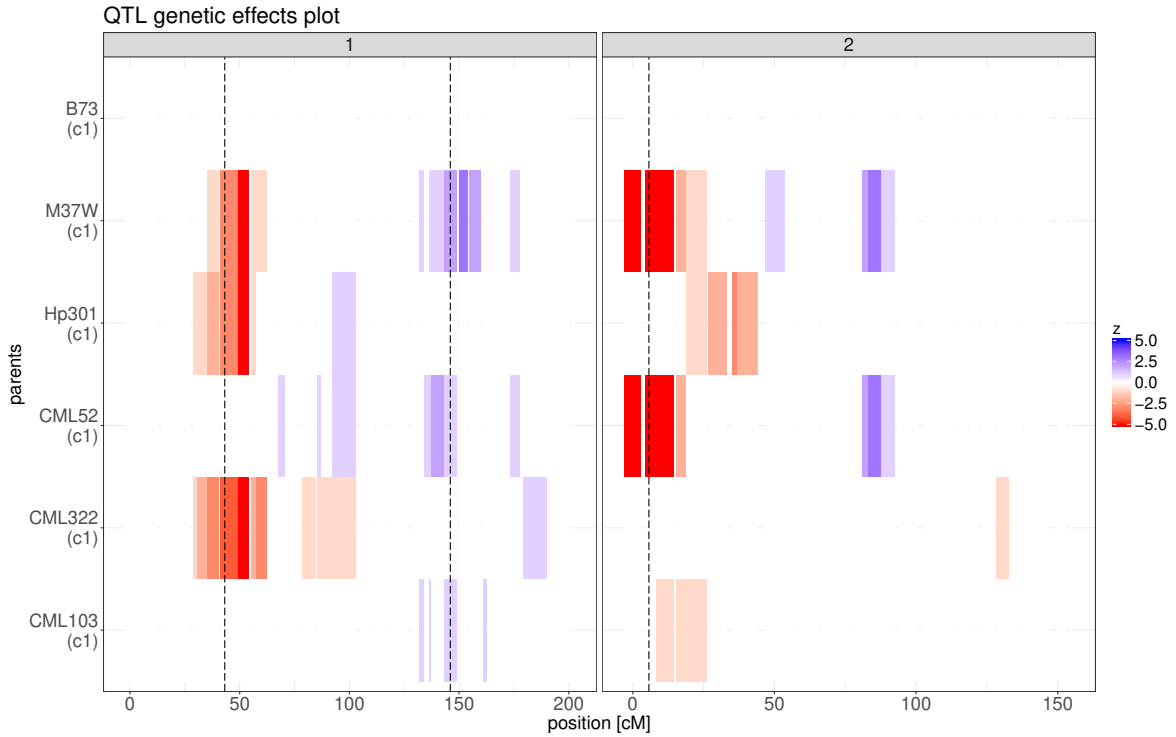

Figure 4: Visualisation genome-wide genetic effect significance from a CIM ancestral model using HRT. The vertical dashed lines represent the detected QTL positions.

4. Summary

mppR is a package for QTL analysis in multi-parent populations working in the R environment. It can analyse any type of MPP design composed of more than one cross between at least three parents like NAM populations, diallels or factorial designs, where individual crosses are of the Fx, BCx, BCsFt, RIL or DH type. It contains functions to perform all the steps of the QTL detection from the data processing to the results visualisation.

Acknowledgements

The development of **mppR** is part of a project financed by KWS Saat SE.

References

Aulchenko YS, Ripke S, Isaacs A, Van Duijn CM (2007). "**GenABEL**: An R Library for Genome-Wide Association Analysis." *Bioinformatics*, **23**(10), 1294–1296. URL http://www.genabel.org/.

Bardol N, Ventelon M, Mangin B, Jasson S, Loywick V, Couton F, Derue C, Blanchard P, Charcosset A, Moreau L (2013). "Combined Linkage and Linkage Disequilibrium QTL

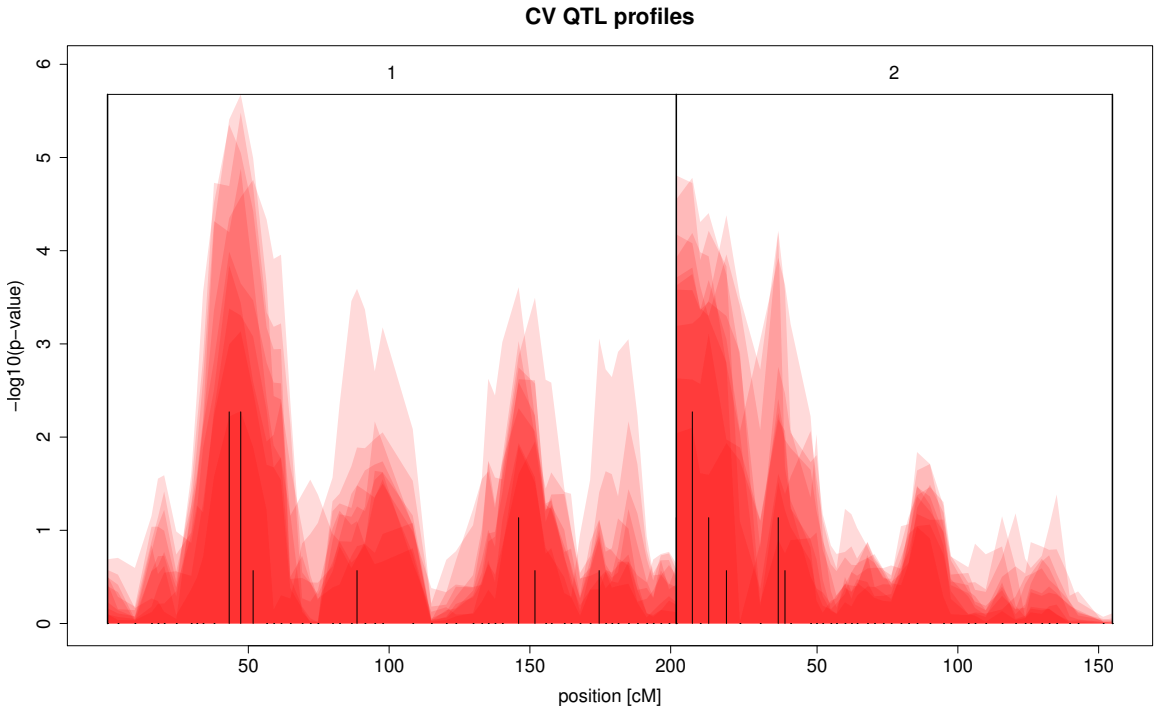

Figure 5: Transparency plot of CV results representing an overlay of QTL profiles. The black bars are proportional to the number of times a position was detected as QTL.

Mapping in Multiple Families of Maize (Zea mays L.) Line Crosses Highlights Complementarities Between Models Based On Parental Haplotype and Single Locus Polymorphism." *Theoretical and applied genetics*, **126**(11), 2717–2736.

Blanc G, Charcosset A, Mangin B, Gallais A, Moreau L (2006). "Connected Populations for Detecting Quantitative Trait Loci and Testing for Epistasis: An Application in Maize." *Theoretical and Applied Genetics*, **113**(2), 206–224.

Bradbury PJ, Zhang Z, Kroon DE, Casstevens TM, Ramdoss Y, Buckler ES (2007). "TASSEL: Software for Association Mapping of Complex Traits in Diverse Samples." *Bioinformatics*, **23**(19), 2633–2635. URL http://www.maizegenetics.net/tassel.

Broman KW, Wu H, Sen Ś, Churchill GA (2003). "R/pkgqtl: QTL Mapping in Experimental Crosses." *Bioinformatics*, **19**(7), 889–890. URL http://www.rqtl.org/.

Butler D, Cullis BR, Gilmour A, Gogel B (2009). "**ASReml-R** Reference Manual." *The State of Queensland, Department of Primary Industries and Fisheries, Brisbane*. URL https://www.vsni.co.uk/software/asreml/.

Cavanagh C, Morell M, Mackay I, Powell W (2008). "From Mutations to MAGIC: Resources for Gene Discovery, Validation and Delivery in Crop Plants." *Current opinion in plant biology*, **11**(2), 215–221.

Churchill GA, Doerge RW (1994). "Empirical Threshold Values for Quantitative Trait Mapping." *Genetics*, **138**(3), 963–971.

Covarrubias-Pazaran G (2016). "Genome Assisted Prediction of Quantitative Traits Using the R Package **Sommer**." *PLoS ONE*, **11**, 1–15. URL https://cran.r-project.org/web/packages/sommer/index.html.

Doerge RW (2002). "Mapping and Analysis of Quantitative Trait Loci in Experimental Populations." *Nature Reviews Genetics*, **3**(1), 43–52.

Garin V, Wimmer V, Mezmouk S, Malosetti M, van Eeuwijk F (2017). "How do the type of QTL effect and the form of the residual term influence QTL detection in multi-parent populations? A case study in the maize EU-NAM population." *Theoretical and Applied Genetics*, **130**(8), 1753–1764.

Giraud H, Lehermeier C, Bauer E, Falque M, Segura V, Bauland C, Camisan C, Campo L, Meyer N, Ranc N, *et al.* (2014). "Linkage Disequilibrium With Linkage Analysis of Multiline Crosses Reveals Different Multiallelic QTL for Hybrid Performance in the Flint and Dent Heterotic Groups of Maize." *Genetics*, **198**(4), 1717–1734.

Jansen RC, Jannink JL, Beavis WD (2003). "Mapping Quantitative Trait Loci in Plant Breeding Populations." *Crop Science*, **43**(3), 829–834.

Jourjon MF, Jasson S, Marcel J, Ngom B, Mangin B (2005). "MCQTL: Multi-Allelic QTL Mapping in Multi-Cross Design." *Bioinformatics*, **21**(1), 128–130. URL https://carlit.toulouse.inra.fr/MCQTL/.

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**(3), 1709–1723.

Leroux D, Rahmani A, Jasson S, Ventelon M, Louis F, Moreau L, Mangin B (2014). "**Clusthaplo**: A Plug-in for MCQTL to Enhance QTL Detection Using Ancestral Alleles in Multi-Cross Design." *Theoretical and Applied Genetics*, **127**(4), 921–933. URL https://cran.r-project.org/src/contrib/Archive/clusthaplo/.

Li R, Lyons MA, Wittenburg H, Paigen B, Churchill GA (2005). "Combining Data from Multiple Inbred Line Crosses Improves the Power and Resolution of Quantitative Trait Loci Mapping." *Genetics*, **169**(3), 1699–1709.

Lipka AE, Tian F, Wang Q, Peiffer J, Li M, Bradbury PJ, Gore MA, Buckler ES, Zhang Z (2012). "GAPIT: Genome Association and Prediction Integrated Tool." *Bioinformatics*, **28**(18), 2397–2399. URL http://www.maizegenetics.net/gapit.

McMullen MD, Kresovich S, Villeda HS, Bradbury P, Li H, Sun Q, Flint-Garcia S, Thornsberry J, Acharya C, Bottoms C, *et al.* (2009). "Genetic Properties of the Maize Nested Association Mapping Population." *Science*, **325**(5941), 737–740.

Myles S, Peiffer J, Brown PJ, Ersoz ES, Zhang Z, Costich DE, Buckler ES (2009). "Association Mapping: Critical Considerations Shift from Genotyping to Experimental Design." *The Plant Cell*, **21**(8), 2194–2202.

Rebaï A, Goffinet B (1993). "Power of Tests for QTL Detection Using Replicated Progenies Derived from a Diallel Cross." *Theoretical and Applied Genetics*, **86**(8), 1014–1022.

Rebaï A, Goffinet B (2000). "More About Quantitative Trait Locus Mapping with Diallel Designs." *Genetical research*, **75**(02), 243–247.

Rincent R, Moreau L, Monod H, Kuhn E, Melchinger AE, Malvar RA, Moreno-Gonzalez J, Nicolas S, Madur D, Combes V, *et al.* (2014). "Recovering Power in Association Mapping Panels with Variable Levels of Linkage Disequilibrium." *Genetics*, **197**(1), 375–387.

Utz HF, Melchinger AE, Schön CC (2000). "Bias and Sampling Error of the Estimated Proportion of Genotypic Variance Explained by Quantitative Trait Toci Determined From Experimental Data in Maize Using Cross Validation and Validation with Independent Samples." *Genetics*, **154**(4), 1839–1849.

van Eeuwijk FA, Bink MC, Chenu K, Chapman SC (2010). "Detection and Use of QTL for Complex Traits in Multiple Environments." *Current opinion in plant biology*, **13**(2), 193–205.

Varshney RK, Singh VK, Hickey JM, Xun X, Marshall DF, Wang J, Edwards D, Ribaut JM (2015). "Analytical and Decision Support tools for Genomics-Assisted Breeding." *Trends in plant science*.

Wald A (1943). "Tests of Statistical Hypotheses Concerning Several Parameters when the Number of Observations Is Large." *Transactions of the American Mathematical society*, **54**(3), 426–482.

Weeks DL, Williams DR (1964). "A Note on the Determination of Connectedness in an N-Way Cross Classification." *Technometrics*, **6**(3), 319–324.

Wimmer V, Albrecht T, Auinger HJ, Schön CC (2012). "**Synbreed**: A Framework for the Analysis of Genomic Prediction Data using R." *Bioinformatics*, **28**(15), 2086–2087. URL http://synbreed.r-forge.r-project.org/.

Würschum T, Liu W, Gowda M, Maurer H, Fischer S, Schechert A, Reif J (2012). "Comparison of Biometrical Models for Joint Linkage Association Mapping." *Heredity*, **108**(3), 332–340.

Xavier A, Xu S, Muir W, Rainey K (2015). "**NAM**: Association Studies in Multiple Populations." *Bioinformatics*, p. btv448. URL https://cran.r-project.org/web/packages/NAM/index.html.

Yu J, Holland JB, McMullen MD, Buckler ES (2008). "Genetic Design and Statistical Power of Nested Association Mapping in Maize." *Genetics*, **178**(1), 539–551.

Yu J, Pressoir G, Briggs WH, Bi IV, Yamasaki M, Doebley JF, McMullen MD, Gaut BS, Nielsen DM, Holland JB, *et al.* (2006). "A Unified Mixed-Model Method for Association Mapping that Accounts for Multiple Levels of Relatedness." *Nature genetics*, **38**(2), 203–208.

Zeng ZB (1993). "Theoretical Basis for Separation of Multiple Linked Gene Effects in Mapping Quantitative Trait Loci." *Proceedings of the National Academy of Sciences*, **90**(23), 10972–10976.

Zeng ZB (1994). "Precision Mapping of Quantitative Trait Loci." *Genetics*, **136**(4), 1457–1468.

Affiliation:

Vincent Garin
E-mail: vincent.garin6@gmail.com