University of California, Berkeley
U.C. Berkeley Division of Biostatistics Working Paper Series

Year 2011 *Paper* 285

Targeted Methods for Finding Quantitative Trait Loci

Hui Wang* Sherri Rose†
Mark J. van der Laan‡

*Stanford University, hwangui@stanford.edu
†University of California, Berkeley, sherrirosephd@gmail.com
‡University of California, Berkeley, laan@berkeley.edu

http://biostats.bepress.com/ucbbiostat/paper285

Targeted Methods for Finding Quantitative Trait Loci

Hui Wang, Sherri Rose, and Mark J. van der Laan

Abstract

Conventional genetic mapping methods typically assume parametric models with Gaussian errors, and obtain parameter estimates through maximum likelihood estimation. We propose a general semiparametric model to map quantitative trait loci (QTL) in experimental crosses. In contrast with widely-used interval mapping (IM) derived methods, our model requires fewer assumptions and also accommodates various machine learning algorithms. Estimation using both targeted maximum likelihood and collaborative targeted maximum likelihood methods is compared to a composite interval mapping (CIM) approach. We demonstrate with simulations and real data analyses that, on average, our semiparametric targeted learning approach produces less biased QTL effect estimates than those from parametric models.

Targeted Methods for Finding Quantitative Trait Loci

Hui Wang, Sherri Rose, and Mark van der Laan

Division of Biostatistics, School of Public Health, University of California, Berkeley, California 94720

1 Introduction

The goal of quantitative trait loci (QTL) mapping is to identify genes underlying an observed trait in the genome using genetic markers. In experimental organisms, the QTL mapping experiment usually involves crossing two inbred lines with substantial differences in a trait, and then scoring the trait in the segregating progeny. A series of markers along the genome is genotyped in the segregating progeny, and associations between the trait and the QTL can be evaluated using the marker information. Of primary interest are the positions and effect sizes of QTL genes.

Early literature (Sax 1923; Thoday 1960) focused on directly analyzing a single marker using analysis of variance (ANOVA). The biggest disadvantage of such marker-based analysis is its inability to assess QTL genes between markers. In 1989, Lander and Botstein proposed the interval mapping (IM) method (Lander and Botstein 1989). With IM, the genotypic value of a QTL follows a multinomial distribution, determined by the distance of the QTL to its flanking markers and the genotypes of the flanking markers. The trait value is modeled as a Gaussian mixture with the mixing proportions being the multinomial probabilities of the QTL genotype. The significance of the QTL effect is then assessed using likelihood ratio test. By testing positions at small increments along the genome, a whole-genome finely scaled test statistic profile can be constructed. IM has greatly increased the accuracy of estimating QTL parameters, and it has gained wide popularity in the genetic mapping community. Later, Haley and Knott developed a regression method to approximate IM (Haley and Knott 1992). This method imputes the unobserved genotypic value of a putative QTL with its expected value.

IM methods unrealistically assume there is only one gene underlying the observed trait in the entire genome, represented as testing each potential position separately (Lander and Botstein 1989) or computing the univariate association between the expected genotypic value and the phenotypic trait in Haley–Kott regression. In other words, IM only considers the current QTL; all other QTL genes are ignored. When this assumption is violated, the effects of other QTL genes are contained within the residual variance, affecting the assessment of QTL parameters.

To handle multiple QTL genes, Jansen (1993) and Zeng (1994) developed a composite interval mapping (CIM) approach. In CIM, background markers are added to a standard IM statistical model to reduce noise and increase the precision of QTL effect estimates. Thus, the CIM approach estimates QTL effects adjusted for confounding markers and can substantially improve the performance of IM when the background markers are properly chosen. Multiple interval mapping (MIM) was also developed to simultaneously estimate effects and positions of multiple QTL genes (Kao et al. 1999). MIM enjoys greater power but is computationally difficult. It also has a long-standing estimator selection problem: Which QTL genes are to be included? Bayesian approaches have also been studied and applied in QTL mapping (Satagopan et al. 1996; Heath 1997; Sillanpaa and Arjas 1998).

In recent years, with finely scaled single nucleotide polymorphism (SNP) markers replacing the traditional widely spaced microsatellite markers, identifying QTL genes between markers has become less concerning. Due to the high-dimensional nature of SNP data, the univariate marker-trait regression is widely used for its simplicity and computational feasibility despite its noisy results.

Machine learning algorithms, such as random forests (Breiman 2001), are also used to map QTL genes (Lee et al. 2008). While machine learning algorithms are powerful statistical tools, their

use in QTL mapping is limited. These algorithms are particularly good at identifying interactions between genes and predicting the conditional expectation of the outcome given the covariates (in our case, genetic markers). However, their variable importance measurements (VIMs) lack p-values and are otherwise not targeted toward the effects of interest.

Many of the QTL methods discussed above are fully parametric and typically assume a Gaussian distribution for the phenotypic trait, as well as require specification of a parametric regression model. The estimation of QTL effects often relies on the method of maximum likelihood estimation. Maximum likelihood estimation based on such parametric regression models is widely used and well studied, with software available in many platforms. However, quite often, these parametric models represent an over-simplified description of the underlying genetic mechanism and leads to biased estimates. In addition, if the parametric model is data-adaptively selected among a set of candidate parametric regression models, then the reported standard errors and the p-values are not interpretable.

In this technical report, we address the QTL mapping problem through the use of a semiparametric regression model, the targeted maximum likelihood estimator (TMLE) (van der Laan and Rubin 2006; van der Laan and Rose 2011) and the collaborative TMLE (C-TMLE) (van der Laan and Gruber 2010; van der Laan and Rose 2011). The only assumption of the semiparametric regression model is that the phenotypic trait changes linearly with the QTL gene. We define the TMLE and, in particular, the C-TMLE, which is an appealing estimator for high-dimensional genomic data structures. Our approach allows one to explore a much larger model space with fewer restrictions while still being computational feasible with its incorporation of machine learning algorithms. The (C-)TMLE approach targets the VIMs of interest (i.e., the QTL genes) and can provide improved QTL gene effect estimates and rankings by taking advantage of the prediction power of machine learning algorithms. Excerpts from this technical report have been published in the refereed literature (Wang et al. 2010, 2011) with several remaining unpublished sections submitted for publication in a final third paper.

2 Methods

Typical segregating designs include the backcross (B1) design and the intercross (F2) design. Backcross is produced by back crossing the first generation (F1) to one of its parental strains; F2 is produced by intercrossing the first generation (F1 x F1). In a backcross population, there are two possible genotypes *Aa* and *aa* at any locus; in an F2 population, there are three genotypes *AA*, *Aa*, and *aa*. For the ease of presentation, we will use backcross to demonstrate our method. All the derivations can be readily extended to F2 population and other types of experimental crosses.

2.1 Semiparametric Regression Model

Suppose the observed data are i.i.d. realizations of $O_i = (Y_i, M_i) \sim P_0$, $i = 1, \ldots, n$, where Y represents the phenotypic trait value, M represents the marker genotypic values, and i indexes the ith subject. Let A be the genotypic value of the putative QTL under consideration. When A lies on a marker, A is observed. When A lies between markers, it is unobserved. In this case, we impute A with its expected value from a multinomial distribution computed from the genotypes and the

relative locations of its flanking markers. This is the same strategy used in Haley–Knott regression (Haley and Knott 1992), and we will thus only be estimating the effect of an imputed A. The semiparametric regression model for the effect of A at value $A = a$ relative to $A = 0$, adjusted for a user-supplied set of other markers M^-, is given by

$$E_0(Y \mid A = a, M^-) - E_0(Y \mid A = 0, M^-) = \beta_0 a. \tag{1}$$

Other parametric forms, such as $a\sum_{j=1}^{J} \beta_j V_j$ incorporating effect modification by other markers V_j, can be incorporated as well. We view β_0 as our parameter of interest, which also corresponds with a marginal average effect obtained by averaging this conditional effect over the distribution of M^-.

In a backcross population, when the homozygote aa is coded 0 and the heterozygote Aa is coded 1, β_0 measures the effect of the Aa genotype on Y relative to aa. In an F2 population, with the coding $(AA, Aa, aa) = (1, 0, -1)$, β_0 can be interpreted as the difference in Y when A changes from heterozygote to homozygote. In the above model, the only assumption we make is the linearity of the QTL effect (i.e. $\beta_0 A$) on the phenotype. We do not impose any distributional assumption on the data and any functional form on all functions $f(M^-)$ of M^-. For β_0 to be estimable and to be well defined, we also need the assumption that A is not a perfect surrogate of M^-. In other words, if we choose to estimate $E_0(A \mid M^-)$, the R^2 (coefficient of determination) from the estimator has to be less than 1.

2.2 The TMLE

The TMLE of β_0 (van der Laan and Rubin 2006; Tuglus and van der Laan 2011) involves an initial machine learning (e.g., super learner) fit of $E_0(Y \mid M)$ based on the squared error loss function, which yields a fit of $E_0(Y \mid A = 0, M^-)$, mapping the latter into an initial estimator of β_0 and thereby of $E_0(Y \mid A, M^-)$ in the semiparametric regression model. After obtaining this initial estimator of $E_0(Y \mid A, M^-)$ of the semiparametric form as enforced by the semiparametric regression model, we carry out a single targeted update step by adding an estimate of the clever covariate $A - E_0(A \mid M^-)$, and fitting the coefficient ε in front of this clever covariate with univariate regression, using the initial estimator of $E_0(Y \mid A, M^-)$ as offset. Note that the TMLE of β_0 is now simply $\beta_n^0 + \varepsilon_n$. The TMLE algorithm defined below is performed for each A.

Obtain an initial estimator Q_n^0 for $E_0(Y \mid A, M^-)$. This initial estimator has to respect the semiparametric model in equation (1) and takes the form $Q_n^0 = \beta_n^0 A + f_n(M^-)$.

Obtain a reasonable estimate $g_n(W)$ of the marker confounding mechanism $E_0(A \mid W)$. We typically only need to focus on a subset W of M^- that is viewed as potential confounders of the effect of A on Y. Hence, we can rewrite the $g_0(M^-)$ as $g_0(W)$, and we denote its estimator with $g_n(W)$. In our application, W is the set of markers on the same chromosome as A.

Compute $r(A, W) = A - g_n(W)$. The $r(A, W)$ is the residual of $g_n(W)$, also referred to as the "clever covariate". It plays the key role of correcting the bias in the initial estimator.

Fit the "ε-regression." This regression is given by $Y' \sim \varepsilon r(A, W)$ where $Y' = Y - Q_n^0(A, M^-)$ and the regression coefficient estimate is denoted ε_n.

Update. The initial estimate of β_n^0 is updated with $\beta_n^1 = \beta_n^0 + \varepsilon_n$, and the initial fitted value Q_n^0 with $Q_n^1(A, M^-) = Q_n^0(A, M^-) + \varepsilon_n r(A, W)$.

Compute the variance estimate σ_n^2 for β_n^1. Using influence-curve-based methods, we calculate the variance estimate

$$\sigma_n^2 = \frac{\sum_i (Y_i - Q_n^1(A_i, M_i^-))^2 r_i(A_i, W_i)^2}{(\sum_i A_i r(A_i, W_i))^2}.$$

The TMLE estimator will be consistent if either the Q_n^0 or the $g_n(W)$ is consistent, and will be efficient when Q_n^0 is consistent. In other words, when Q_n^0 is correctly specified, the TMLE estimate of β essentially stays unchanged with minor modification from the second stage. When Q_n^0 is mis-specified, a correct specification of the $g_n(W)$ will achieve a full bias reduction for Q_n^0 and β^0.

The estimation of the clever covariate requires an estimator of $E_0(A \mid M^-)$ (or $E_0(A \mid W)$). The latter can be carried out with a machine learning algorithm regressing A on M^-. In particular, one could decide to fit this regression of the marker of interest on two flanking markers, thereby dramatically simplifying the estimation problem, while potentially capturing most of the confounding by the total marker set M^-. The choice of how great the distance between the flanking markers will be is a delicate issue. If one selects the flanking markers right next to the marker of interest, the data might not allow the separation of the effect of interest from the effect of the flanking markers. That is, one is aiming to adjust for confounders that are too predictive of the marker of interest. On the other hand, if one selects the flanking markers too far away from the marker of interest, the flanking markers will not adjust well for the markers that are in between the marker of interest and the flanking markers. Simulations in the previous chapter suggest that the TMLE shows no sign of deterioration for correlations smaller than 0.7 between the marker of interest and the confounders. This could be used to set the window width defined by the two flanking markers. Subject matter considerations, such as that the scientist would be satisfied with a claim that the targeted effect of the marker can be due to other markers in a window of a particular size, could also be used to set this window width of the flanking markers.

An alternative approach is to let the data decide what other markers to include in the model for $E_0(A \mid M^-)$. For that purpose, we can employ the C-TMLE (using a linear regression working model for fluctuation of initial estimator) for estimation of an additive effect $E_0(E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W))$ for the observed data structure $O = (W, A, Y)$ and nonparametric model for the probability distribution P_0 of O. This C-TMLE has also been implemented for this estimation problem, but, obviously, now in terms of TMLEs in this semiparametric regression model. Thus, this algorithm involves using forward selection of main terms to build a main term linear regression fit for $E_0(A \mid M^-)$, based on the sum of squared residuals (i.e., MSE) of the corresponding TMLE of $E_0(Y \mid A, M^-)$ that uses this main term regression fit of $E_0(A \mid M^-)$ in the clever covariate. Cross-validation is used to select the number of main terms (i.e., the number of forward selection steps that the algorithm carries out) that will actually be included in the fit of $E_0(A \mid M^-)$. The candidate main terms can include fits of $E_0(A \mid M^-)$ such as one based on two flanking markers defined by a choice of window width, across a number of possible window widths. In this manner the C-TMLE algorithm can data-adaptively decide how aggressive the targeting step should be in its effort to reduce bias due to residual confounding.

The C-TMLE implementation may also involve the selection of a penalty to be added to the MSE in order to make the procedure more robust in the context of having to adjust for highly cor-

related markers. Details are presented in the next section. C-TMLE allows one to data-adaptively determine the markers to include in the fit of $E_0(A \mid W)$. For example, one may wish to only adjust for the two closest markers that are farther than δ-apart from the marker A, and one can use C-TMLE to data-adaptively select this choice δ based on the log-likelihood of the TMLE of the semiparametric regression fit. In our simulations and data analysis we have implemented both TMLEs as well as C-TMLEs.

2.3 The C-TMLE

Let $Q_n^0 = m(A,V \mid \beta_n^0) + r(M^-)$ be the initial estimate of Q_0 contained in the same semiparametric regression model that we also used in the TMLE. The C-TMLE is concerned with iteratively updating this initial estimate of Q_0. Firstly, we compute a set of K univariate covariates $W_1,\ldots,W_K$ from M^-, which we will refer to as main terms, even though a term could be an interaction term or a super learning fit of the regression of A on a subset of the components of M^-. Let's refer to M^- by $W = (W_1,\ldots,W_K)$. In this subsection we will suppress in the notation for estimates of Q_0 and g_0 their dependence on the sample size n. Let $\Omega = \{W_1,\ldots,W_K\}$ be the full collection of main terms. A linear regression model fit g^K of $g_0(W) = E_0(A \mid W)$ using all main terms in Ω is viewed as the most nonparametric estimate of g_0. For a given subset of main terms $\mathscr{S} \subset \Omega$, let $\mathscr{S}^c$ be its complement within Ω. For a given subset $\mathscr{S}^k$, we will define g^k as the least squares fit of the linear regression model for $E_0(A \mid W)$ that includes as main terms all the terms in $\mathscr{S}^k$. In the C-TMLE algorithm we use a forward selection algorithm that augments a given set $\mathscr{S}^k$ into a next set $\mathscr{S}^{k+1}$ obtained by adding the best main term among all main terms in the complement $\mathscr{S}^{k,c}$ of $\mathscr{S}^k$. In other words, the algorithm iteratively updates a current estimate g^k into a new estimate g^{k+1}, but the criterion for g does not measure how well g fits g_0; it measures how well the TMLE using this g fits Q_0.

Let $L(Q)(O) = (Y - Q(A,W))^2$ be the squared error loss function for the true regression function $Q_0 = E_0(Y \mid A,W) = \beta_0 A + E_0(Y \mid A = 0,W)$. For a given initial estimate Q, let $Q_g(\varepsilon) = Q + \varepsilon(A - g(W))$ be the parametric working fluctuation model used in the TMLE of Q_0 defined in the previous section. For a given estimate g of g_0 and initial Q of Q_0, the corresponding TMLE (as defined in the previous section) of Q_0 is given by $Q_g(\varepsilon_n)$, where $\varepsilon_n = \arg\min_\varepsilon P_n L(Q_g(\varepsilon))$ is the univariate least squares estimator of ε using the initial estimate Q as offset, and P_n denotes the empirical probability distribution of $O_1,\ldots,O_n$. Here we used the notation $Pf \equiv \int f(o)dP(o)$. That is, an initial estimate Q, an estimate g, and the data $O_1,\ldots,O_n$ are mapped into a new targeted maximum likelihood estimate $Q^* = Q_g(\varepsilon_n)$. Let's refer to this mapping as $Q^* = \text{TMLE}(Q,g)$, suppressing its dependence on P_n.

The C-TMLE algorithm defined below generates a sequence $(Q^k, \mathscr{S}^k)$ and corresponding TMLEs Q^{k*}, $k = 0,\ldots,K$, where Q^k represents an initial estimate, $\mathscr{S}^k$ a subset of main terms that defines g^k, and Q^{k*} the corresponding TMLE that updates Q^k using g^k. These TMLEs Q^{k*} represent subsequent updates of the initial estimator Q_n^0, and the corresponding main term set $\mathscr{S}^k$, as used to define g^k in this k-specific TMLE, increases in k, one unit at a time: $\mathscr{S}^0$ is empty, $\mid \mathscr{S}^{k+1} \mid = \mid \mathscr{S}^k \mid +1$, $\mathscr{S}^K = \Omega$. The C-TMLE uses cross-validation to select k, and thereby to select the TMLE Q^{k*} that yields the best fit of Q_0 among the $K+1$ k-specific TMLEs that are increasingly aggressive in their bias-reduction effort. This C-TMLE algorithm is defined as follows:

Initiate algorithm: Set initial TMLE. Let $k = 0$. $Q^k = Q_n^0$ is the initial estimate of Q_0, and $\mathscr{S}^k$

is the empty set so that g^k is the empirical mean of A. Thus, Q^{k*} is the TMLE updating this initial estimate Q^k using as clever covariate $A-g^k$.

Determine next TMLE. Determine the next best main term to add to the linear regression working model for $g_0(W) = E_0(A \mid W)$:

$$\mathscr{S}^{k+1,cand} = \arg \min_{\mathscr{S}^k \cup W_j : W_j \in \mathscr{S}^{k,c}} P_n L(\text{TMLE}(Q^k, \mathscr{S}^k \cup W_j)).$$

If

$$P_n L(\text{TMLE}(Q^k, \mathscr{S}^{k+1,cand})) \leq P_n L(\text{TMLE}(Q^{k*})),$$

then $(\mathscr{S}^{k+1} = \mathscr{S}^{k+1,cand}, Q^{k+1} = Q^k)$, else $Q^{k+1} = Q^{k*}$, and

$$\mathscr{S}^{k+1} = \arg \min_{\mathscr{S}^k \cup W_j : W_j \in \mathscr{S}^{k,c}} P_n L(\text{TMLE}(Q^{k*}, \mathscr{S}^k \cup W_j)).$$

[In words: If the next best main term added to the fit of $E_0(A \mid W)$ yields a TMLE of $E_0(Y \mid A, W)$ that improves upon the previous TMLE Q^{k*}, then we accept this best main term, and we have our next TMLE Q^{k+1*}, g^{k+1} (which still uses the same initial estimate as Q^{k*} uses). Otherwise, reject this best main term, update the initial estimate in the candidate TMLEs to the previous TMLE Q^{k*} of $E_0(Y \mid A, W)$, and determine the best main term to add again. This best main term will now always result in an improved fit of the corresponding TMLE of Q_0, so that we now have our next TMLE Q^{k+1*}, g^{k+1} (which now uses a different initial estimate than Q^{k*} used).]

Iterate. Run this from $k = 1$ to K at which point $\mathscr{S}^K = \Omega$. This yields a sequence (Q^k, g^k) and corresponding TMLE Q^{k*}, $k = 0, \ldots, K$.

This sequence of candidate TMLEs Q^{k*} of Q_0 has the following property: the estimates g^k are increasingly nonparametric in k and $P_n L(Q^{k*})$ is decreasing in k, $k = 0, \ldots, K$. It remains to select k. For that purpose we use V-fold cross-validation. That is, for each of the V splits of the sample in a training and validation sample, we apply the above algorithm for generating a sequence of candidate estimates $(Q^{k*} : k)$ to a training sample, and we evaluate the empirical mean of the loss function at the resulting Q^{k*} over the validation sample, for each $k = 0, \ldots, K$. For each k we take the average over the V-splits of the k-specific performance measure over the validation sample, which is called the cross-validated risk of the k-specific TMLE. We select the k that has the best cross-validated risk, which we denote with k_n. Our final C-TMLE of Q_0 is now defined as Q^{k_n*}, and the corresponding updated regression coefficient is our TMLE β_n^* of β_0.

Remark. The candidate main terms can also include fits of $E_0(A \mid M^-)$ such as one based on two flanking markers defined by a choice of window width, across a number of possible window widths. In this manner, the above C-TMLE algorithm data-adaptively decides which window width yields effective bias reduction.

Penalty. C-TMLE implementation in the following data analysis involved a penalized mean squared error as a measure of fit instead of the mean squared error, where the penalty is defined

as a variance estimator of the corresponding TMLE of β_0. With C-TMLE we select among a sequence of candidates using a likelihood-based criterion. In our setting, for a putative QTL A with the TMLE fit Q_n^1, this log-likelihood can be reduced to the empirical MSE:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Q_n^1(A_i, M_i^-))^2.$$

To protect the algorithm from breaking down in borderline identifiable cases, the MSE is also penalized with the variance estimate of β_n. We denote this penalized MSE as "pMSE", indexed by $g_n(W)$:

$$\text{pMSE}(g_n(W)) = \text{MSE} + \sigma_n^2.$$

The pMSE criterion is used in C-TMLE to construct a sequence of increasingly nonparametric candidates of $g_n(W)$. For the confounding marker set W of dimension m_w, we start with an intercept model, and then in a manner similar to that of forward stepwise regression, we grow $g_n(W)$ by adding markers from W one at a time based on pMSE. Every time the $g_n(W)$ grows bigger, we define it as a "move". Let k, $k = 1 \cdots K$, be the number of moves. In our case, k is equivalent to the number of covariate terms in $g_n(W)$. The capital K is the maximum possible number of moves, and in our implementation $K = \min(m_w, 10)$. We can then index each candidate $g_n(W)$ with k. Notationally, we will use g_n^k to represent the $g_n(W)$ with k moves. The sequence of g_n^k should be increasingly nonparametric, which means $\text{pMSE}(g_n^k) > \text{pMSE}(g_n^{k+1})$. In the case that adding markers in the g_n^k does not result in a smaller pMSE, we augment the ε-regression with a new clever covariate computed from the g_n^{k-1}. This will ensure that the $\text{pMSE}(g_n^k)$ is always smaller than the $\text{pMSE}(g_n^{k-1})$. Hence, for a realized g_n^k, it will be associated with a fixed number h of clever covariates in the ε-regression. In this way, we can create a sequence of candidates g_n^k with increasing size and log-likelihood, and one can then choose the best g_n^k using cross validation. Below is a practical implementation of the C-TMLE procedure using this penalty:

1. For $k = 0$, set $h = 1$ and initialize the g_n^0 with the intercept model, which essentially does zero adjustment for $Q_n^{(0)}$.

2. For $k = 1, \cdots, K$, carry out the following two steps:

   (a) At $k = k$, exclude the markers in the g_n^{k-1} from W, and index the remaining markers with j, $j = 1, \cdots, m_w - k + 1$. For each remaining W_j, incorporate it into the g_n^{k-1}. This will result in $m_w - k + 1$ candidates for g_n^k indexed by g_n^{k-1} and W_j, and we denote them with $c(g_n^{k-1}, W_j)$. Compute $\text{pMSE}(c(g_n^{k-1}, W_j))$ for each j. If

   $$\min\left\{\text{pMSE}(c(g_n^{k-1}, W_j))\right\} < \text{pMSE}(g_n^{k-1}), \tag{2}$$

   take the $c(g_n^{k-1}, W_j)$ with the minimum pMSE as the k-th candidate g_n^k.

   (b) If the inequality 2 is not satisfied, we update the TMLE initial fit in the k-th move with the TMLE estimates from the $(k-1)$-th move, i.e.:

   $$Q_n^{0_k} = Q_n^{1_{k-1}} \quad \text{and} \quad \beta_n^{0_k} = \beta_n^{1_{k-1}}.$$

This is equivalent to having an additional clever covariate in the ε-regression, and h will be increased by 1. We then redo step (a) with the updated $Q_n^{0_k}$ and $\beta_n^{0_k}$. This step guarantees that $\text{pMSE}(g_n^k) < \text{pMSE}(g_n^{k-1})$.

3. When all the K candidates g_n^k are built, corresponding to each g_n^k, we will also have K exact algorithms for building the initial estimator $Q_n^{0_k}$, which is relevant to the number of clever covariates in g_n^k. Now one can carry out TMLE with each g_n^k and the corresponding $Q_n^{0_k}$, and select the best g_n^k based on the cross validated MSE. In addition, on the parsimonious side, the g_n^k of a smaller model should be given more preference. To serve this purpose, we use a BIC (Bayesian Information Criterion) like criterion to penalize the size of the g_n^k. We recognize that one clever covariate has one degree of freedom. When there are multiple moves within a single clever covariate, we evenly partition the one degree of freedom among its moves. Hence, for a particular g_n^k with k moves and h clever covariates, its size is defined as:

$$s_{g_n^k} = (h-1) + \frac{k_h}{l_h},$$

where k_h is the position of the k-th move within the h-th clever covariate, and l_h is the total number of moves within h. For example, for the $g_n^{k=5}$ in a candidate sequence list with $k = (1, \cdots, 7)$ and $h = (1,1,1,2,2,2,3)$,

$$s_{g_n^2} = (2-1) + \frac{2}{3} = 1\frac{2}{3}.$$

We can then define a criterion pMSE^* indexed by g_n^k based on the cross validated MSE:

$$\text{pMSE}^*(g_n^k) = n\log(\text{CV MSE} + \sigma_n^2) + s_{g_n^k}\log(n). \tag{3}$$

We then choose the g_n^k associated with the minimum pMSE^* as the working $g_n(W)$ in the TMLE.

A particular problem we want to highlight here is overfitting in the initial estimator Q_n^0. In the ε-regression, the dependent variable is the residual from the Q_n^0. Overfitting can destroy useful signals in these residuals, and hence does harm to TMLE. In particular, we have encountered problems with TMLE caused by overfitting in Random Forest, while we have not run into the same problem with machine learning algorithms using internal cross validation to select the fine tuning parameters. Nevertheless, since it does present problems, we want to discuss here how to avoid overfitting in Q_n^0 and possible remedies for a TMLE estimator with an overfitted Q_n^0.

In the first place, we want to prevent the usage of an overfitted Q_n^0 without compromising the fit. Overfitting is usually represented as more parameters than needed to achieve the same cross validation performance. Hence, when there are multiple Q_n^0 candidates, a natural solution is to choose the Q_n^0 with the minimum cross validated (CV) L_2 risk. However, we found out that the CV risk alone is often not enough to address the overfitting problem for TMLE, and we suggest to use the difference between the CV risk and the empirical (EM) risk as an additional penalty. This penalty term is explicitly aimed at penalizing overfitting because an overfitted model often has a much bigger CV risk than its EM risk. We then have the pMSE^{**} criterion:

$$\text{pMSE}^{**} = 2 \times \text{CV Risk} - \text{EM Risk}. \tag{4}$$

This criterion gives us better discretion in borderline cases when an over-complex model produces a similar CV risk as that of a small model.

Alternatively, if there is only a single candidate of Q_n^0 available and overfitting is a concern, we can use the CV Q_n^0 in place of the EM Q_n^0. Suppose the samples are divided into two disjoint sets: a training set and a validation set. The validation set contains the ith observation. The cross validated Q_n^0 of the i-th observation is then obtained through predicting the Y_i from the model learned from the training set. When an aggressive prediction algorithm is used for Q_n^0 and the empirical Q_n^0 is overfitting, the cross validated version of Q_n^0 will not be overfitting and will actually provide an effective remedy to the overfitting problem in the initial estimator without affecting the properties of TMLE.

In principle, when obtaining the initial estimator, a separate Q_n^0 should be computed. This may create a substantial computational burden when there are many markers and complex machine learning algorithms are used. To alleviate this burden, one can first obtain a background estimate $B_n(M)$ for the conditional expectation $E(Y \mid M)$ on all the markers M, and then, for each A, perform the projecting regression $Y \sim \beta A$ with the offset $B_n(Y \mid M^A = 0, M^-)$, where M^A is the marker set closely linked to A including A itself. This same idea is also implemented in the CIM. In practice, one can take the M^A as all the markers within a window size (for example, 10 cM) of A. Another straightforward choice is to take the M^A as all the markers on the same chromosome as A.

Very often, direct variance estimates σ_n^2 of QTL effects from a model obtained using machine learning algorithms are on the small side because the models are chosen data adaptively. This problem is alleviated in TMLE. The variance estimate of TMLE estimator is based on influence curves and, in the first order, only relevant to the limit of Q_n^0 when the sample size is large (van der Laan and Robins 2003). Hence TMLE variance estimates are not affected by how Q_n^0 is constructed, and TMLE p-values are generally more honest with less false positives.

Statistical Properties of the C-TMLE. To understand the appeal of the C-TMLE, we make the following remarks. Including a main term in the fit of the clever covariate that has no effect on the outcome will only harm the TMLE of β_0 both with respect to bias and mean squared error. If one uses the log-likelihood (i.e., MSE) of the regression of A on M^- as a criterion for selection of the main terms, then one will easily select main terms that have a weak effect on the outcome, while truly important main terms are not included. Therefore, it is crucial to use a main term selection criterion for $E_0(A \mid M^-)$ that actually measures the fit of the resulting TMLE of the outcome regression. In addition, one can formally prove that the TMLE achieves the full bias reduction with respect to β_0 if the clever covariate uses a true regression, $E_0(A \mid M^s)$, with M^s being a reduction of M^- that is rich enough so that $E_0(Y \mid A = 0, M^-)$ is captured. In fact, the result is stronger, since M^s only needs to capture the function of M^- that is obtained by taking the difference between the true $E_0(Y \mid A = 0, M^-)$ and its initial estimator $E_n(Y \mid A = 0, M^-)$ (van der Laan and Gruber 2010). Thus, theory indeed fully supports that we should be selecting main terms in the clever covariate that are predictive of residual bias of the initial estimator of $E_0(Y \mid A = 0, M^-)$, and the C-TMLE algorithm presented above indeed targets such main terms.

3 Simulations

We carried out three simulations to study the behavior of targeted methods. Simulation I compares TMLE with the CIM, Simulation II studies the effect of overfitted initial estimators on the performance of TMLE, and Simulation III is a demonstration of C-TMLE.

3.1 Simulation I

A single chromosome of 100 markers was simulated on 600 backcross subjects. Markers were evenly spaced at 2 centimorgan (cM). A single QTL main effect was generated at marker position 100 cM, denoted by $M_{(100)}$. Here, the number in the subscript of M indicates the position of the marker. There were also four epistatic effects on markers $M_{(60)}$, $M_{(90)}$, $M_{(120)}$, and $M_{(150)}$. Phenotypic values were generated from the data-generating distribution: $Y = 5 + 1.2M_{(100)} - 0.8M_{(60)}M_{(90)} - 0.8M_{(90)}M_{(120)} - 0.8M_{(120)}M_{(150)} - 0.8M_{(150)}M_{(60)} + U$, where U is the error term drawn from an exponential distribution scaled to have a variance of 10. We generated 500 simulated data sets of this type.

In this simulation, the density of markers is fairly high, the phenotypic outcome follows a nonnormal distribution, and there are strong counteracting epistatic effects in linked markers. A univariate regression effect estimate of the effect of, for example, $M_{(100)}$ will be biased due to the lack of adjustment for the effect of the highly correlated markers. Indeed, the CIM estimate for the effect of $M_{(100)}$ is negative, far away from the true value 1.2. On the other hand, taking the CIM prediction function as the initial estimator $\bar{Q}_n^0$, TMLE was then able to recover some of the signal and hence improved on the CIM estimates. In TMLE, the true regression of A on the other 99 markers, M^-, was estimated with a main terms linear regression including two flanking markers with a prespecified distance to A. We used two distances, 20 cM and 40 cM, and denote the estimators by $\text{TMLE}_{(20)}$ and $\text{TMLE}_{(40)}$. The CIM analysis was carried out using QTL Cartographer (Basten et al. 2001), with default settings. We analyzed markers without considering positions between them. For CIM, the mean effect estimate for $M_{(100)}$ is -0.2731 and is dominated by the epistatic effects from its nearby markers. $\text{TMLE}_{(40)}$ is able to correct some of the bias, and its effect estimate is 0.5365. $\text{TMLE}_{(20)}$ utilizes an estimator of $E_0(A \mid M^-)$ with more predictive power than $\text{TMLE}_{(40)}$ and produced an estimate closest to the truth. We list the averages of the effect estimates for $M_{(100)}$ across 500 simulations in Table 1 along with their standard errors for CIM, $\text{TMLE}_{(20)}$, and $\text{TMLE}_{(40)}$.

We also used a univariate regression (UR) fit for $\bar{Q}_n^0$ within TMLE, and these results can be

Table 1: Mean effect estimates of $M_{(100)}$ over 500 simulations

	$\bar{Q}_n^0$=CIM		$\bar{Q}_n^0$=UR	
	Estimate	SE	Estimate	SE
Initial Estimate	-0.2731	0.3273	-0.6248	0.2684
$\text{TMLE}_{(40)}$	0.5365	0.4538	0.2705	0.3135
$\text{TMLE}_{(20)}$	0.8478	0.4508	0.8093	0.4079

found in Table 1. The UR initial estimate was even more biased than that of CIM. TMLE$_{(20)}$, using UR as $\bar{Q}_n^0$, produced very similar estimates to TMLE$_{(20)}$ using CIM as initial estimator. On the other hand, TMLE$_{(40)}$ using the CIM as initial estimator produced a better estimator than TMLE$_{(40)}$ using the univariate regression as initial estimator. This demonstrates the robustness of TMLE with respect to misspecification of the initial estimator, which predicts that the more predictive the regression of A on M^-, the more robust TMLE will be to the choice of its initial estimator. A closer look at Table 1 also reveals that compared to TMLE$_{(40)}$, the additional bias reduction of TMLE$_{(20)}$, using univariate regression as initial estimator, comes with an increase in standard error.

3.2 Simulation II

Simulation II imitates the classic QTL mapping scenario with widely spaced markers and a more complex structure of QTL genes. Forty-one markers on a single large chromosome of 800 cM were generated for 1000 backcross subjects, evenly spaced at 20 cM. Six main QTL effects and four epistatic effects were placed on the chromosome with effect sizes ranging from 1% to 10% of the total phenotypic variance. Positions of these QTL genes either overlap with a marker or lie in between markers. Details of these simulated QTL genes can be found in Table 2. We set the population mean equal to 5.0, and the error is normally distributed with mean 0 and variance 10. We replicated this simulation for 100 times.

To obtain a background marker model $B_n(M)$, we evaluated three algorithms: Deletion Substitution Addition (DSA) (Sinisi and van der Laan 2004), random forests (RF), and super learner (SL) (van der Laan et al. 2007). DSA is a search algorithm using polynomial basis functions and minimizing residual sum of squares over subsets of covariates in a regression. When restricted to main term linear models, it often produces results similar to CIM. RF is a tree-based nonparametric machine learning algorithm. In this simulation, the least aggressive fit is a linear main-term model from DSA, and we consider RF to be the most aggressive fit as it is likely to capture interaction terms. SL takes both DSA and RF as its candidate learners and finds an optimal combination of these two. Here, on average, it combines these algorithms with weights 0.59 for DSA and 0.41 for RF. See Table 3 for the average CV risk, the empirical (EM) risk, and the pMSE** risk of the $B_n(M)$ for these three algorithms. If we had only considered the CV risk, we would have chosen SL as the best $B_n(M)$. The differences between the CV risks for all three algorithms were subtle, while the differences between the EM risks were more substantial. It is apparent that RF is overfitting because its EM risk is much smaller than its CV risk. SL takes RF as a candidate learner, and is thus also affected by this overfitting. Taking this into consideration, DSA produces the best Q_n^0. Hence, in the following analysis, we use a TMLE initialized with DSA as a reference line.

The entire chromosome was scanned with a 2-cM incremental step. In $g_n(W)$, W_1 and W_2 are the flanking markers 40 cM to A. In Table 4 and Figure 1, we report the TMLE estimates and their standard errors at genomic locations with a main QTL effect, for the RF initial estimator, TMLE(Q_n^0=EM RF), TMLE(Q_n^0=CV RF), and TMLE(Q_n^0=DSA). (We use TMLE(Q_n^0) to index TMLE with its initial estimator Q_n^0.) For RF, the initial effect estimates are far from the truth due to overfitting. When using the RF fit as Q_n^0, TMLE produced better estimates than the RF initial estimator, but it was not better than TMLE(Q_n^0=DSA). We then used a cross-validated RF fit as Q_n^0. TMLE was able to fully recover the effect size estimates to the level of TMLE(Q_n^0=DSA), while the

Table 2: **True positions and effect sizes of QTL genes in Simulation II**

Position (cM)	Effect Size	Proportion (%) of variance	Effect type
40	2.40	6.57	Main
310	3.00	10.27	Main
330	-3.00	10.27	Main
675	2.00	4.56	Main
710	-1.60	2.82	Main
780	1.20	1.64	Main
70 / 120	1.85	2.92	Epistatic
200 / 240	3.00	7.71	Epistatic
450 / 610	2.31	4.56	Epistatic
490 / 525	1.85	2.92	Epistatic

Note: For epistatic effects, positions of interacting QTLs are indicated with a slash. Proportions of explained variance for epistatic effects were computed assuming interacting QTLs are independent of each other.

Table 3: **The mean risk of $B_n(M)$ from Simulation II**

	SL	DSA	RF
EM risk	7.55(64.02%)	12.27(41.46%)	3.20(84.74%)
CV risk	12.97(38.12%)	13.30(36.58%)	13.61(35.06%)
pMSE**	18.40	14.32	24.03

Percentages in the brackets are the coefficients of determination (R^2).

Table 4: **The estimates of QTL main effects and their standard errors in Simulation II**

Truth		RF Initial			TMLE								
					Q_n^0=EM RF			Q_n^0=CV RF			Q_n^0=DSA		
Pos	Effect	Pos	Est	SE	Pos	Est	SE	Pos	Est	SE	Pos	Est	SE
40	2.4	46	2.0952	0.3367	46	2.0802	0.3295	44	2.2841	0.3220	46	2.6822	0.4348
310	3.0	296	0.6684	0.1887	298	0.9385	0.2169	298	1.4429	0.2927	306	1.6206	0.3937
330	-3.0	344	-0.6349	0.1628	338	-0.9266	0.2031	338	-1.4878	0.2999	334	-1.5446	0.3781
675	2.0	674	0.7209	0.2015	678	0.8926	0.2463	678	1.3198	0.3216	674	1.7316	0.5324
710	-1.6	714	-0.3633	0.1316	724	-0.6901	0.1596	724	-1.1989	0.2426	706	-0.9472	0.6162
780	1.2	784	0.6300	0.1525	774	0.7519	0.1616	774	1.1238	0.2253	774	1.2895	0.3507

Positions (Pos) are in centi-morgans.

position estimates stayed essentially unchanged compared to TMLE(Q_n^0=EM RF). TMLE(Q_n^0=CV RF) also generated a more conservative p-value profile than TMLE(Q_n^0=EM RF), as illustrated in Figure 1.

3.3 Simulation III

In Simulation III, we demonstrate the performance of C-TMLE in a simple example. We simulated 600 backcross subjects and 120 markers each spaced at 5 cM on a pseudo-chromosome. Two QTL genes were placed at marker position 110 cM and marker position 310 cM, denoted with $M_{(110)}$ and $M_{(310)}$. An epistatic effect was situated at $M_{(200)}$ and $M_{(240)}$. Error terms were drawn from $N(0, 10)$. The two main effects at $M_{(110)}$ and $M_{(310)}$ account for 3.3% and 2.3% of the total phenotypic varaiance, respectively, and the epistatic effect accounts for 1.7% of the total variance. We generated 100 such simulation sets. Since markers were dense, our analysis did not consider positions in-between markers. Therefore, all As are observed. The CIM output was used as Q_n^0. For $g_n(W)$ in C-TMLE, we arbitrarily excluded a 20 cM region around A from W to avoid adjusting for strongly correlated markers.

C-TMLE improved the original CIM estimates, with a better resolution and less significant p-values, as illustrated in Figure 2. We also analyzed this simulation with TMLE$_{(20)}$ (see Simulation I). Compared to TMLE$_{(20)}$, C-TMLE produced similar effect estimates yet smaller standard errors, resulting in smaller p-values. As noted before, by adjusting for markers highly correlated with A in $g_n(W)$, we remove more bias in the estimate. However, additional adjustment also means larger standard errors. In the extreme case, the model parameter β becomes unidentifiable. Thus, there is a trade-off between using a highly predictive $g_n(W)$ and a conservative one, and the C-TMLE algorithm is designed to determine an optimal trade-off based on the data. In this simulation, C-TMLE selects a more conservative $g_n(W)$ than TMLE$_{(20)}$. In general, C-TMLE is indeed less aggressive than TMLE$_{(20)}$. Choice of Q_n^0 also affects C-TMLE. As an illustration, we used the univariate regression (UR) fit as Q_n^0. C-TMLE with the UR initial produced satisfactory results, considering the poor performance of UR. However, with the CIM initial, C-TMLE is performs even better with improved resolution and better separation of two linked interacting QTL genes (Figure 2).

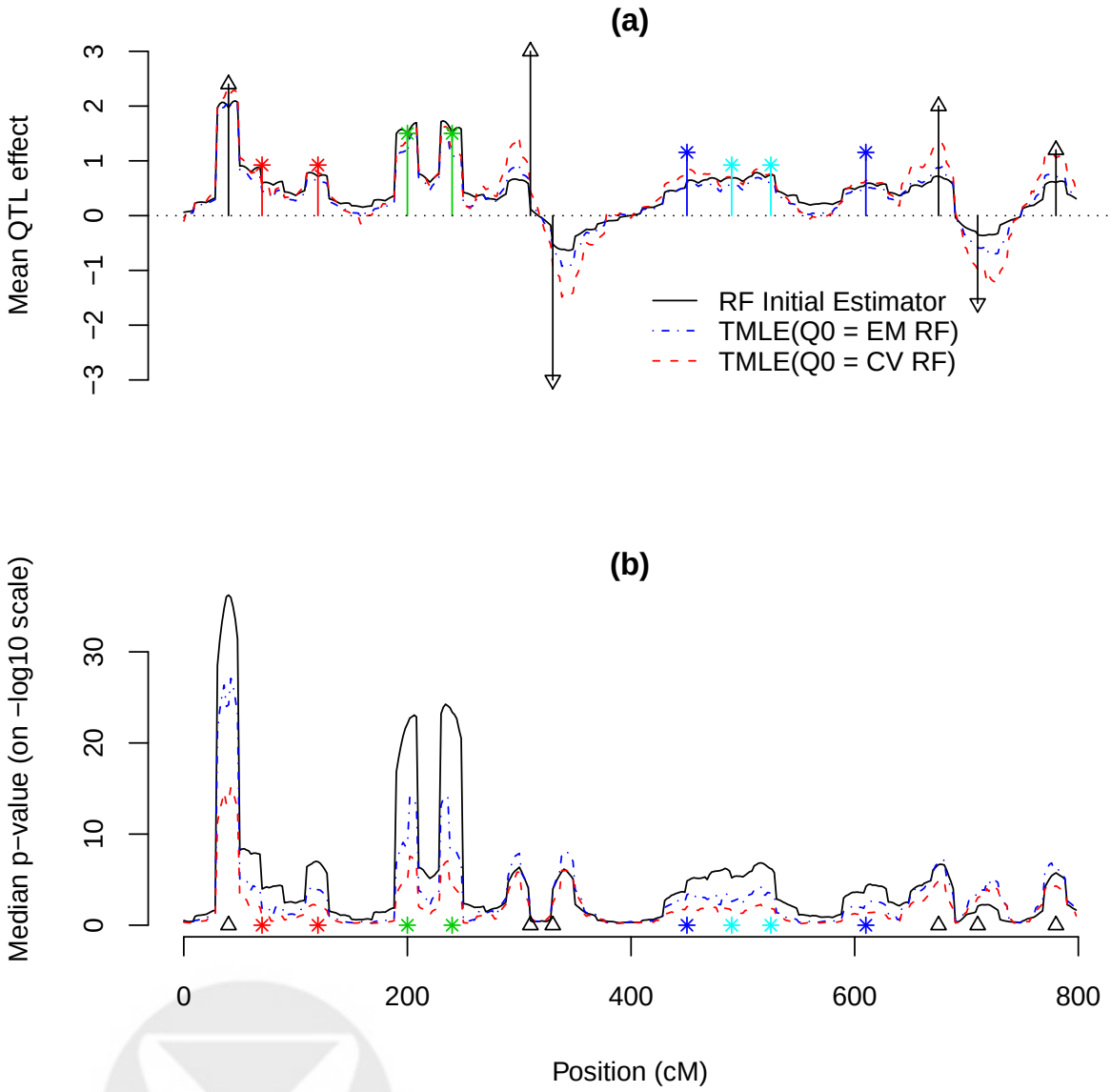

Figure 1: The mean effect estimate and the median p-value of β_n from simulation II. In (a), the mean effect estimate across all simulations are plotted at each tested position, for the initial RF estimator, TMLE($Q_n^{(0)}$=EM RF), and TMLE($Q_n^{(0)}$=CV RF). True effect sizes and positions are superimposed. Black triangles represent main QTL effects. Colored stars represent epistatic effects. Epistatic effects are halved for a clear display, and the interacting QTLs are grouped in the same color. In (b), the median p-values are plotted at each tested position in correspondence with (a).

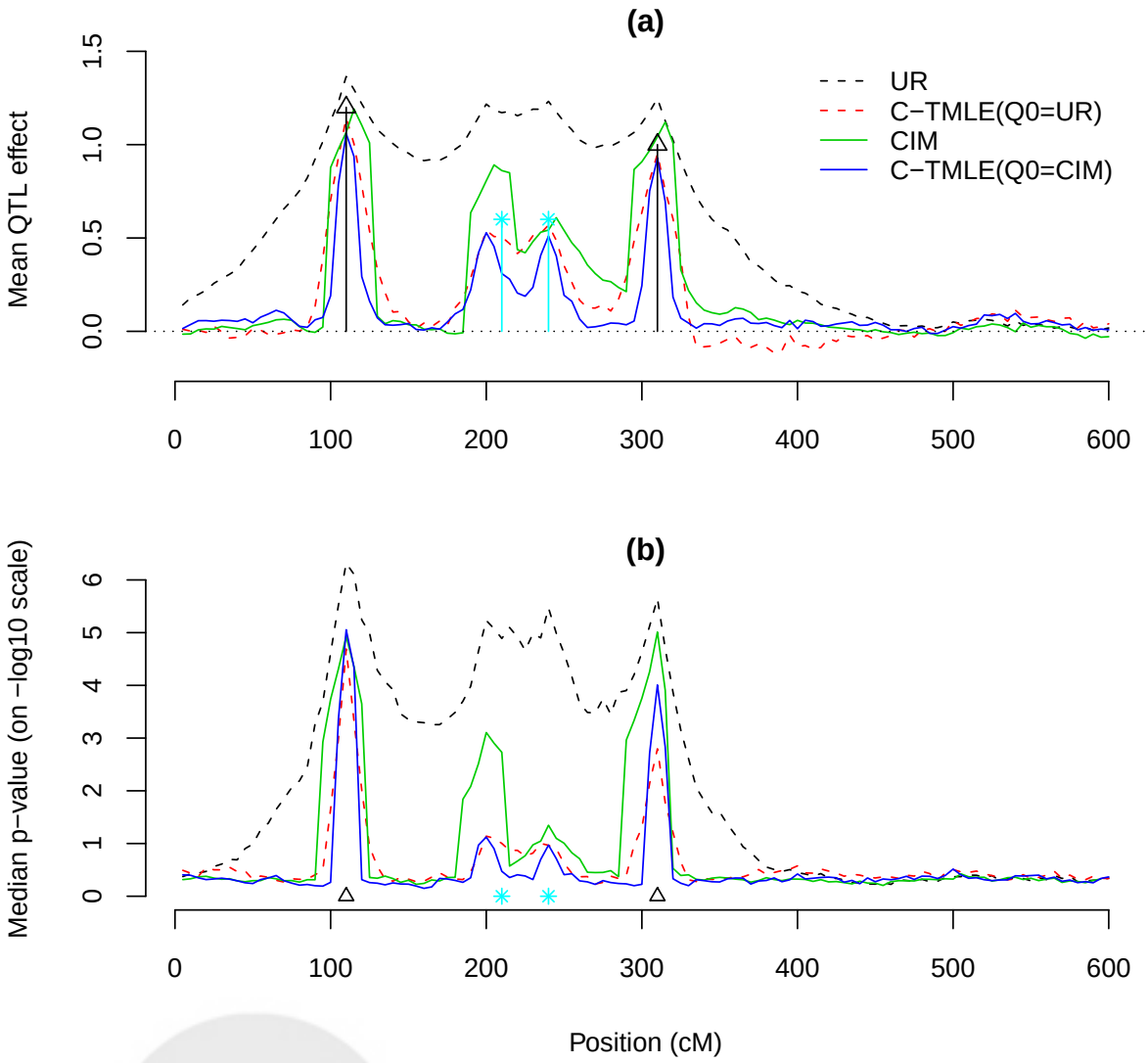

Figure 2: A demonstration of C-TMLE performance. In (a), the average QTL effect estimate across 100 simulations is plotted against its position, for UR, CIM, C-TMLE with UR initial, and C-TMLE with CIM initial. UR and C-TMLE with UR initial are grouped by broken lines. CIM and C-TMLE with CIM initial are grouped by solid lines. Black triangles indicate two main effects at position 110 and 310 cM. Blue stars indicate the epistatic effect at the location 200 and 240 cM, and for the ease of plotting, this epistatic effect is evenly divided between $M_{(200)}$ and $M_{(240)}$. Dotted line is the zero line.In (b),the median p-value of β_n across all simulations at each marker is plotted, in correspondence with (a).

Table 5: **The estimates of QTL effect sizes and positions from CIM and C-TMLE for the wound-healing trait**

	CIM			C-TMLE		
QTL ID	Chr	cM	Effect size	Chr	cM	Effect size
1	1	43.91	-0.1170	1	51.4	-0.1098
2	2	44.11	-0.0433	-	-	-
3	2	56.41	-0.0460	2	58.3	-0.0453
4	3	28.81	-0.0531	3	32.5	-0.0487
5	4	20.61	-0.0993	4	19.4	-0.0891
6	4	55.01	-0.1024	4	57.4	-0.0979
7	6	0.01	0.0444	6	3.4	0.0478
8	6	32.21	-0.0992	6	25.4	-0.1120
9	-	-	-	6	33.4	-0.1148
10	6	51.91	-0.0457	6	55.4	-0.0412
11	7	30.91	0.1089	7	39.4	0.0969
12	9	43.31	-0.1582	9	46.3	-0.1714
13	-	-	-	11	38.5	0.05972
14	12	2.01	0.0563	12	4.3	0.05752
15	13	45.91	-0.0785	13	44.1	-0.0822
16	13	58.61	-0.0783	-	-	-
17	-	-	-	17	8.1	0.05124
18	-	-	-	17	30.1	-0.0585

4 Data Analyses

We present three data application studies to demonstrate the utility of targeted methods in QTL data.

4.1 Wound-Healing Application

In this section, we analyze a data set published in Masinde et al. (2001). The original study was designed to identify QTL genes involved in the wound-healing process. A genomewide scan of 119 codominant markers was performed using 633 F2 (MRL/MP x SJL/J) mice. Each mouse was punctured with a 2-mm hole in its ear, and the phenotypic trait was the hole closure measurement at day 21. The marginal distribution of the phenotypic trait is bell-shaped.

We analyzed this data set with TMLE, C-TMLE, and CIM. Based on the evaluation of a discrete super learner (van der Laan et al. 2007) that included both DSA and random forests, the DSA machine learning algorithm was selected as initial estimator of $E_0(Y \mid M)$, and subsequently mapped into the desired initial estimator for $E_0(Y \mid A, M^-)$ satisfying the semiparametric regression model. To lessen the computational load, we first screened additive and dominant effects of all markers with univariate regression and supplied to this machine learning algorithm the markers

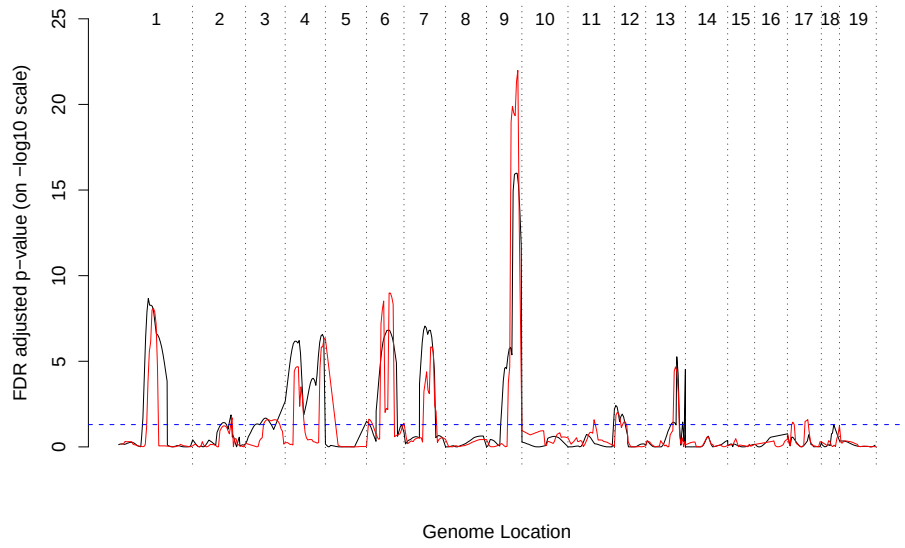

Figure 3: The genomewide FDR adjusted p-value profile for the additive effects in the wound-healing data set. The *black line* represents CIM, and the *red line* represents C-TMLE. Chromosome numbers are superimposed on top of the picture.

with a p-value less than 0.10. In the TMLE, the conditional mean of A, given M^- is fitted with a main terms linear regression model with main terms A_c, W_1^a, W_1^d, W_2^a, W_2^d, where A_c denotes the dominant effect of A when A is additive and the additive effect of A when A is dominant, W_1 and W_2 are the closest flanking markers 20 cM away from A, and the superscript a denotes the additive effect and d the dominant effect. In C-TMLE, when A is additive, W is defined as the additive effects of all markers on the same chromosome 20 cM away from A and the dominant effects of all markers on the same chromosome as A, and vice versa when A is dominant.

Four hundred putative QTL positions were tested at 2-cM increments for both the additive and dominant effects. The p-values were adjusted using FDR (Benjamini and Hochberg 1995). The TMLE and C-TMLE produced similar results, thus we only present C-TMLE results. Figure 3 displays the genomewide FDR-adjusted p-value profile for the additive effect at each tested position. Table 5 summarizes significant QTL genes at level 0.05. The CIM p-values were computed from the asymptotic χ^2 distribution. No significant dominant effect was detected in this data set. The C-TMLE essentially identified the same QTL genes as CIM, albeit with an improved resolution. Many of these genes were also reported in Masinde et al. (2001). However, on chromosome 6, the C-TMLE suggests two linked QTL genes instead of one, as indicated by CIM.

4.2 Listeria Application

Boyartchuk et al. (2001) published a data set on the survival time of 116 age-matched female mice following infection with *Listeria monocytogenes*, a Gram-positive bacteria causing a wide range

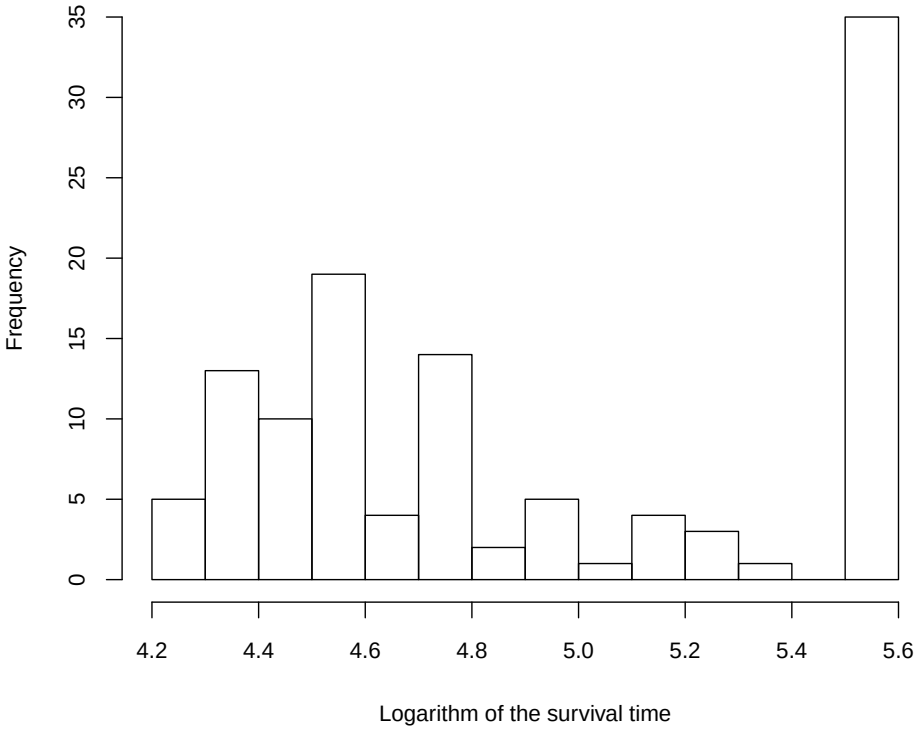

Figure 4: Histogram of the survival time for mice upon infection with *Listeria monocytogenes*, on logarithm scale.

of diseases. The mice were an F2 intercross population derived from susceptible BALB/cByJ and resistant C57BL/6ByJ strains, and the goal of the study was to map genetic factors of susceptibility to *L. monocytogenes*. The phenotypic trait is the recorded time to death for each mouse upon infection with *L. monocytogenes*. One hundred and thirty-one codominant markers were genotyped on the autosomal chromosomes. When a mouse survived beyond 240 h, it was considered recovered. About 30% of the mice recovered, and we refer to them as survivors and the remaining mice as nonsurvivors. This creates a spike in the phenotypic trait distribution, violating the normality assumption in traditional approaches of QTL mapping (Figure 4).

The outcome Y was defined as the logarithm of the phenotypic trait. Y can be decomposed into a binary trait of survival or nonsurvival and a continuous trait of survival time among nonsurvivors (Broman 2003). We denote this binary trait of survival by $Z = I(Y = \log 264)$. Then, the expected value of Y given the marker data M can be represented as

$$E_0(Y \mid M) = P_0(Z = 1 \mid M) \log 264 + P_0(Z = 0 \mid M) E_0(Y \mid Z = 0, M).$$

In the above formula, $P_0(Z = 1 \mid M)$ and $P_0(Z = 0 \mid M)$ are conditional probabilities of whether a mouse has survived ($Z = 1$) or died ($Z = 0$) given the marker data M. We fit this with a super

Table 6: **Mean risk of candidate initial regressions in discrete super learner from the Listeria data set**

	DSA	RF	SL	2-part SL
CV risk	0.2212	0.1581	0.1589	0.1463
EM risk	0.0938	0.0293	0.0293	0.0246
pMSE**	0.3586	0.2868	0.2885	0.2681

Table 7: **The estimates of effect sizes and positions of QTL genes from CIM, TMLE, and C-TMLE in the Listeria dataset. QTL genes with FDR adjusted p-values smaller than 0.05 are reported.**

QTL		CIM			TMLE			C-TMLE		
ID	Type	Chr	cM	Effect size	Chr	cM	Effect size	Chr	cM	Effect size
1	dom	1	15.0	-0.2351	-	-	-	-	-	-
2	dom	1	72.8	0.1606	-	-	-	-	-	-
3	add	1	78.8	-0.1349	1	76.1	-0.1114	1	78.1	-0.1074
4	dom	2	14.0	-0.2623	-	-	-	-	-	-
5	add	2	18.0	-0.1744	-	-	-	-	-	-
6	dom	5	0.0	-0.1468	-	-	-	-	-	-
7	dom	5	61.0	-0.1693	-	-	-	-	-	-
8	add	5	18.1	0.2764	5	26.1	0.1743	5	26.1	0.1960
9	dom	6	33.8	-0.1235	-	-	-	-	-	-
10	dom	12	41.8	-0.2352	12	40.1	-0.1372	12	40.1	-0.1372
11	add	13	22.7	-0.3409	13	16.4	-0.1599	13	14.4	-0.1668
12	dom	13	25.9	0.3525				13	26.4	0.1458
13	add	15	25.1	0.1540	15	26.1	0.0647	15	22.1	0.0678
14	dom	15	12.0	0.2042				15	22.1	0.1438
15	add	18	-	-	18	14.1	-0.0679	18	14.1	-0.0692

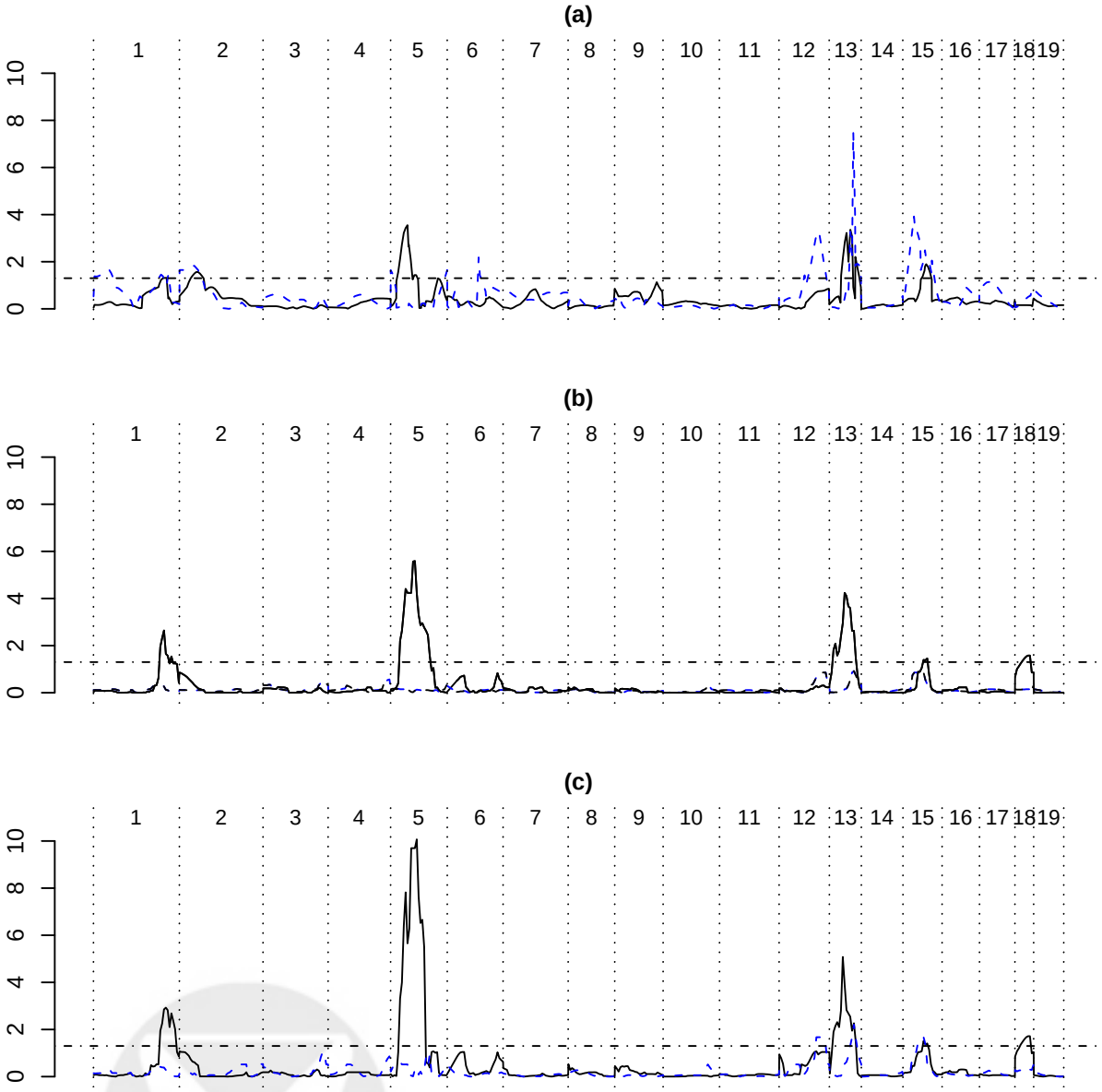

Figure 5: The genomewide p-value profile for the additive and dominant effects in the Listeria dataset. The p-values are FDR adjusted and on a negative log10 scale. (a) the p-value profile from the CIM. (b) p-value profile from the TMLE. (c) p-value profile from the C-TMLE. In all three panels, the *black solid line* represents additive effects, and the *blue dashed line* represents dominant effect. The *black dash-dot line* indicates the 0.05 p-value threshold. Chromosome numbers are superimposed on top of each panel.

learning algorithm for binary outcomes. $E_0(Y \mid Z = 0, M)$ is the conditional expectation of Y on M given that the mouse has died, which can be obtained by applying super learning on nonsurvivors. We refer to this machine learning algorithm as the 2-part super learner.

The collection of algorithms in the super learner included DSA and random forests. As before, the machine learning algorithms were only provided the additive and dominant markers that had a significant univariate effect based on a p-value threshold of 0.10. Since we wished to evaluate if this 2-part super learner provided a better fit than a regular super learner, we implemented a discrete super learner whose library consisted of a total of four algorithms for estimation of $E_0(Y \mid M)$: DSA, random forests, super learner, and a 2-part super learner. In Table 6, we report the EM risk, the CV risk, and the pMSE** of DSA, RF, SL, and the 2-part SL. In the super learning fits, more than 95% of the weight was put on random forests, thereby strongly favoring a fit that allows for complex interactions.

The 2-part super learner had the smallest risk for all three types of risk and was therefore selected as the estimator of $E_0(Y \mid M)$. In the TMLE, we fitted the conditional mean of A, given M^-, with a main term linear regression model including the main terms used $A_c, W_1^a, W_1^d, W_2^a, W_2^d$, where A_c denotes the dominant effect of A when A is additive and the additive effect of A when A is dominant, W_1 and W_2 are the closest flanking markers 20 cM away from A, and the superscript a denotes the additive effect and d the dominant effect.

When inspecting Fig. 5, TMLE and C-TMLE display much less noise than the parametric CIM. Three additive genes on chromosomes 1, 5, and 13 are clearly identified. Two additive effects on chromosomes 15 and 18 are borderline significant. In addition, C-TMLE also detected dominant effects on chromosomes 12, 13, and 15. The chromosome 15 QTL gene is identified as carrying both the additive and dominant effects. The literature suggests that the chromosome 1 QTL gene has an effect on how long a mouse can live given it will eventually die, the chromosome 5 gene has an effect on a mouse's chance of survival, and the genes on chromosomes 13 and 15 are involved in both (Boyartchuk et al. 2001; Broman 2003; Jin et al. 2007). We detected all of these genes and, in addition, an additive gene on chromosome 18 and a dominant gene on chromosome 12. CIM also identified those major genes, however, with less significance and many more suspicious positives. See Table 7.

4.3 Yeast Data Set

In this section, we analyze an expression QTL dataset. The original data came from Brem et al. (2002), consisting of 6216 expression traits and 3312 markers on 112 haploid segregates of budding yeast. Genotypes of markers are dichotomous, and many markers have identical genotypes. We dropped all the redundant markers, resulting in 972 markers. Missing markers were imputed based on the linkage disequilibrium (LD) information of the nearby regions. A fast version of TMLE was applied to this dataset. The initial estimator Q_n^0 is from univariate regression (UR), and $g_n(W)$ is a linear regression with covariates W_1 and W_2, where W_1 and W_2 are markers on both sides of A with LD $R^2 = 0.2$ to A. This strategy essentially consists of three simple linear regressions: $Y \sim A$, $A \sim W_1 + W_2$, and the ε-regression. In this special case, the TMLE estimate for β will be equivalent to the coefficient of A from the multiple regression $Y \sim A + W_1 + W_2$, which means that our semiparametric model is reduced to a simple parametric model. Since we are using a simple Q_n^0 that is unlikely to capture the truth, the consistency of β_n now relies largely on the fit of $g_n(W)$.

Table 8: **Significant QTL hotspots detected by the UR and the TMLE in the yeast dataset. Significant QTL hotspots are defined as the QTL linked to more than 26 gene expression traits for the UR, and 20 traits for the TMLE. Column "n" is the number of genes linked to the QTL hotspot**

Univariate Regression					TMLE				
chr	start	end	n	cis-linked genes	chr	start	end	n	cis-linked genes
1	40	60	32	OAF1, YAL049C, YAL046C	1	40	60	25	SAME
					1	180	200	21	YAR028W, UIP3, MST28
2	300	320	75	N/A					
2	360	380	142	ECM2, TRM7, NRG2, YBR064W, TIP1, TAT1	2	360	380	21	SAME
2	500	580	593	CNS1, TBS1, CSH1, DEM1, PEX32, TOS1, TYR1, NPL4, YBR137W	2	500	580	333	SAME
2	640	680	86	N/A					
3	60	100	279	LEU2, RNQ1, FRM2	3	60	100	201	SAME
3	100	120	65	N/A	3	100	120	50	N/A
3	200	220	67	YCR041W, MATALPHA2, MATALPHA1, TAF2, RSC6	3	200	220	50	SAME
4	920	940	111	N/A	4	920	940	73	N/A
					4	1140	1160	29	YDR339C
4	1520	1540	28	YDR544C, YRF1-1	4	1520	1540	27	SAME
5	100	120	49	URA3, GEA2	5	100	120	41	URA3
5	340	360	148	N/A	5	340	360	23	N/A
5	380	400	72	N/A	5	380	400	23	N/A
5	420	440	155	N/A	5	420	440	32	N/A
7	40	60	135	TAD1	7	40	60	29	SAME
8	80	120	151	GPA1, YAP3, YHL010C, SHU1	8	80	120	90	SAME
					9	20	40	44	YIL169C, YIL166C, YIL163C
10	20	40	36	YJL213W, YJL218W, YJL217W	10	20	40	22	YJL218W, YJL217W
10	80	100	30	SWI3, ATG27, CPS1					
12	500	520	36	YLR173W, DPH5, IDP2, YLR179C, RFX1	12	500	520	32	SAME
12	640	720	256	HAP1, NEJ1, YLR283W, YLR287C	12	640	660	121	HAP1
12	880	900	27	N/A					
12	940	960	27	YLR414C					
12	1040	1060	55	YLR455W	12	1040	1080	84	YLR455W, YLR462W, YLR464W, YLR463C
13	40	60	109	N/A	13	40	60	123	N/A
					13	540	560	39	N/A
14	440	460	413	YPT53, RHO2, YNL089C, TOP2	14	440	460	330	SAME
14	480	500	222	LAT1, MSK1	14	480	540	162	LAT1, MSK1, AQR1
14	540	560	39	SLM2, YNL046W, YNL040W					
15	140	200	517	HMI1, SPO21, RFC4, YOL092W, HAL9, YOL085C, ATG19, PHM7	15	140	180	405	SAME
15	280	300	28	YOL019W, YOL014W	15	280	300	22	SAME
15	560	580	62	YOR131C	15	560	580	56	SAME
					16	420	440	41	CWC27

Despite the simplicity of this approach, there were substantial improvements in TMLE compared to UR (Figure 6), and one can use this fast TMLE as a more reliable screening tool.

We surveyed all the gene-marker pairs with TMLE and UR. *P*-values were adjusted with FDR, pooling all the tests. The 0.05 FDR cutoff for UR was 0.00028, and for TMLE was 0.00011. Redundant linkages were handled as in Wang et al. (2006), where at most one QTL is assumed on each chromosome. In Table 9, we report the number of gene expression traits with multiple QTL at an FDR 0.05 level for both UR and TMLE. With UR, there are 2997 gene expression traits detected with more than 2 QTL, while with TMLE, this number is reduced to 1837. In Zou and Zeng (2009), 1242 traits are claimed to have multiple QTL with a sequential multiple interval mapping procedure. We also summarized our results in terms of the QTL hotspots — small genomic regions linked to multiple gene expression profiles. The yeast genome was divided into 611 20-Kb bins, and the number of non-redundant linkages to each bin were counted. For UR, there are 6430 linkages to 752 markers; for TMLE, there are 4304 linkages to 702 markers.

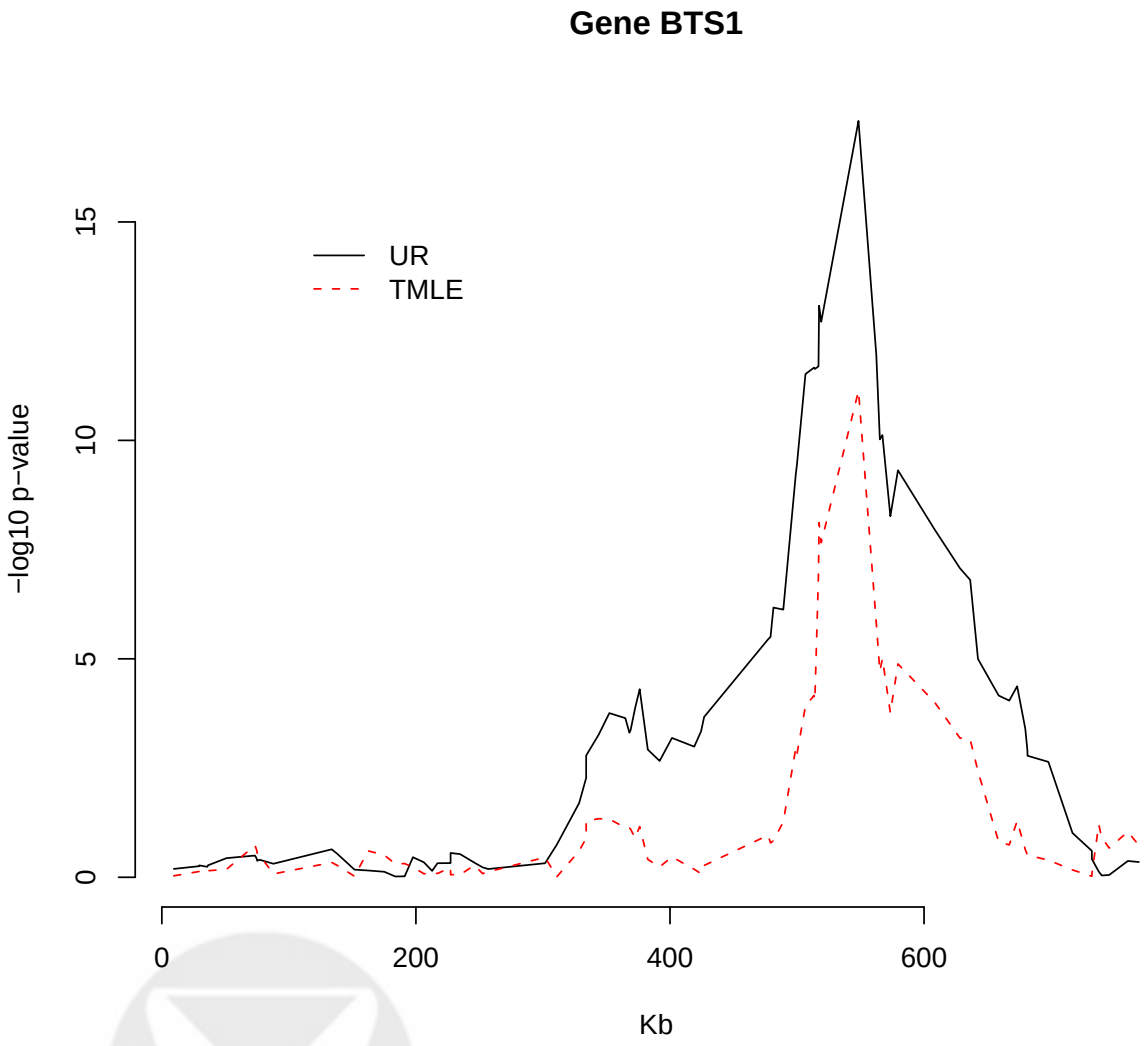

Figure 6: An individual example of how the TMLE achieves a better resolution than the UR for a QTL on chromosome 2 for gene expression trait BTS1. The y-axis is the negative log10 p-value, the x-axis represents the chromosome 2 in Kb. Black line represents the UR, and red line represents the TMLE.

Table 9: **The number of genes tabulated with the number of QTL linked to the gene for the yeast data set**

Number of QTL	0	1	2	3	4	5	6	7
UR	1127	2051	1726	895	297	65	13	1
TMLE	2042	2296	1249	460	108	16	4	0

We also assessed cis-linked genes at significant QTL hotspots and listed these genes in Table 8. A cis-gene is defined as an expression trait gene with a QTL linked to itself within a 10 Kb upstream and downstream window. Our results are essentially consistent with what has been reported in the literature (Brem et al. 2002; Sun et al. 2007).

5 Discussion

Current practice for assessing the effects of genes on a phenotype involves the utilization of parametric regression models. One of the advantages of parametric regression models is that they also provide a p-value, allowing one to rank the different estimated effects and assess their significance. However, both the effect estimates as well as the reported statistical significance are subject to bias due to model misspecification. On the other hand, machine learning algorithms such as random forest, are not sufficient when used alone since these algorithms are tailored for prediction, report generally poor effect estimates, and do not provide a measure of significance. C-TMLE allows us to incorporate the state of the art in machine learning, without significant computational burden (the targeting step is relatively trivial, although it needs to be carried out for each effect), while still providing an estimate tailored for the effect of interest and CLT-based statistical inference.

References

C.J. Basten, B.S. Weir, and Z.B. Zeng. *QTL Cartographer*, 2001. URL `http://statgen.ncsu.edu/qtlcart/`.

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J R Stat Soc Ser B*, 57:289–300, 1995.

V.L. Boyartchuk, K.W. Broman, R.E. Mosher, S.E.F. D'Orazio, M.N. Starnbach, and W.F. Dietrich. Multigenic control of listeria monocytogenes susceptibility in mice. *Nat Genet*, 2001.

L. Breiman. Random forests. *Mach Learn*, 45:5–32, 2001.

R.B. Brem, G. Yvert, R. Clinton, and L. Kruglyak. Genetic dissection of transcriptional regulation in budding yeast. *Science*, 296:752–755, 2002.

K.W. Broman. Mapping quantitative trait loci in the case of a spike in the phenotype distribution. *Genetics*, 2003.

C.S. Haley and S.A. Knott. A simple regression method for mapping quantitative trait loci in line crosses using flanking markers. *Heredity*, 1992.

S.C. Heath. Markov chain Monte Carlo segregation and linkage analysis of oligogenic models. *Am J Hum Genet*, 1997.

R.C. Jansen. Interval mapping of multiple quantitative trait loci. *Genetics*, 1993.

C. Jin, J.P. Fine, and B.S. Yandell. A unified semiparametric framework for quantitative trait loci analysis, with application to spike phenotypes. *J Am Stat Assoc*, 2007.

C.H. Kao, Z.B. Zeng, and R.D. Teasdale. Multiple interval mapping for quantitative trait loci. *Genetics*, 1999.

E.S. Lander and D. Botstein. Mapping Mendelian factors underlying quantitative traits using RFLP linkage maps. *Genetics*, 1989.

S.S.F. Lee, L. Sun, R. Kustra, and S.B. Bull. EM-random forest and new measures of variable importance for multi-locus quantitative trait linkage analysis. *Bioinformatics*, 2008.

G.L. Masinde, X. Li, W. Gu, H. Davidson, S. Mohan, and D.J. Baylink. Identification of wound healing/regeneration quantitative trait loci (QTL) at multiple time points that explain seventy percent of variance in (MRL/MpJ and SJL/J) mice F2 population. *Genome Res*, 2001.

J.M. Satagopan, B.S. Yandell, M.A. Newton, and T.C. Osborn. A Bayesian approach to detect quantitative trait loci using Markov chain Monte Carlo. *Genetics*, 1996.

K. Sax. The association of size difference with seed-coat pattern and pigmentation in *Phaseolus vulgaris*. *Genetics*, 1923.

M.J. Sillanpaa and E. Arjas. Bayesian mapping of multiple quantitative trait loci from incomplete inbred line cross data. *Genetics*, 1998.

S.E. Sinisi and M.J. van der Laan. Deletion/Substitution/Addition algorithm in learning with applications in genomics. *Stat Appl Genet Mol*, 3(1), 2004. Article 18.

W. Sun, T. Yu, and K.C. Li. Detection of eqtl modules mediated by activity levels of transcription factors. *Bioinformatics*, 23:2290–2297, 2007.

J.M. Thoday. Location of polygenes. *Nature*, 1960.

C. Tuglus and M.J. van der Laan. Targeted methods for biomarker discovery. In M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*. Springer, Berlin Heidelberg New York, 2011.

M.J. van der Laan and S. Gruber. Collaborative double robust penalized targeted maximum likelihood estimation. *Int J Biostat*, 6(1):Article 17, 2010.

M.J. van der Laan and J.M. Robins. *Unified Methods for Censored Longitudinal Data and Causality*. Springer, Berlin Heidelberg New York, 2003.

M.J. van der Laan and S. Rose. *Targeted Learning: Causal Inference for Observational and Experimental Data*. Springer, Berlin Heidelberg New York, 2011.

M.J. van der Laan and Daniel B. Rubin. Targeted maximum likelihood learning. *Int J Biostat*, 2 (1):Article 11, 2006.

M.J. van der Laan, E.C. Polley, and A.E. Hubbard. Super learner. *Stat Appl Genet Mol*, 6(1): Article 25, 2007.

H. Wang, S. Rose, and M.J. van der Laan. Finding quantitative trait loci genes with collaborative targeted maximum likelihood learning. *Stat Prob Lett*, published online 11 Nov (doi: 10.1016/j.spl.2010.11.001), 2010.

H. Wang, S. Rose, and M.J. van der Laan. Finding quantitative trait loci genes. In M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*. Springer, Berlin Heidelberg New York, 2011.

S. Wang, M Yehya, E.E. Schadt, H. Wang, T.A. Drake, and A.J. Lusis. Genetic and genomic analysis of a fat mass trait with complex inheritance reveals marked sex specificity. *PLoS Genet*, 2:e15, 2006.

Z.B. Zeng. Precision mapping of quantitative trait loci. *Genetics*, 1994.

W. Zou and Z.B. Zeng. Multiple interval mapping for gene expression qtl analysis. *Genetica*, 137: 125–134, 2009.