Published in final edited form as:

Stat Probab Lett. 2011 July 1; 81(7): 792–796. doi:10.1016/j.spl.2010.11.001.

Finding Quantitative Trait Loci Genes with Collaborative Targeted Maximum Likelihood Learning

Hui Wang^a, Sherri Rose^a, and Mark J. van der Laan^{a,1}

^a University of California, Berkeley, School of Public Health, Division of Biostatistics, 101 Haviland Hall, #7358, Berkeley, CA 94720, USA

Abstract

Quantitative trait loci mapping is focused on identifying the positions and effect of genes underlying an observed trait. We present a collaborative targeted maximum likelihood estimator in a semi-parametric model using a newly proposed 2-part super learning algorithm to find quantitative trait loci genes in listeria data. Results are compared to the parametric composite interval mapping approach.

Keywords

collaborative targeted maximum likelihood estimation; quantitative trait loci; super learner; machine learning

1. Introduction

The goal of quantitative trait loci (QTL) mapping is to identify genes underlying an observed continuous trait in the genome using genetic markers. In experimental crosses (e.g., in mice), it involves randomly crossing subjects between two inbred lines. Of primary interest are the positions and effect sizes of QTL genes. In this article, we address the QTL mapping problem through the use of a semi-parametric regression model and the collaborative targeted maximum likelihood estimator (C-TMLE) (van der Laan and Gruber, 2010). The only assumption of the semi-parametric regression model is that the phenotypic trait changes linearly with the QTL gene.

2. Semi-parametric regression model

Suppose the observed data are i.i.d. realizations of $O_i = (Y_i, M_i) \sim P_0$, $i = 1, \dots, n$, where Y represents the phenotypic trait value, M represents the marker genotypic values, and i indexes the i -th 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 as in Haley and Knott (1992). The semi-parametric regression model for the effect of A at

¹Corresponding author: laan@berkeley.edu, Tel: (510) 642-3241, Fax: (510) 643-5163.

2010 MSC: 62, 92

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

value $A = a$ relative to $A = 0$, adjusted for a user supplied set of other markers M^- , is given by:

$$E_0(Y|A=a, M^-) - E_0(Y|A=0, M^-) = \beta_0 a.$$

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, which also corresponds with a marginal average effect obtained by averaging this conditional effect over the distribution of M^- .

3. TMLE

With targeted maximum likelihood estimation of β_0 , one first obtains an initial machine learning (e.g., super learner) fit of $E_0(Y | M)$ based on the squared error loss function, which also yields a fit of $E_0(Y | A = 0, M^-)$. Given an estimator of $E_0(Y | A = 0, M^-)$, one can regress its residual with Y onto the parametric model βA to obtain the initial estimator of β_0 , and thereby of $E_0(Y | A, M^-)$ in the semi-parametric regression model (van der Laan and Rubin, 2006). After obtaining this initial estimator of $E_0(Y | A, M^-)$ of the semi-parametric form as enforced by the semi-parametric regression model, we carry out a single targeted update step. This involves adding an estimate of the clever covariate $A - E_0(A | M^-)$, and fitting the coefficient ε in front of this clever covariate with univariate regression, using the initial estimator of $E_0(Y | A, M^-) = \beta_0 A + E_0(Y | A = 0, M^-)$ as off-set.

The resulting update of the initial estimator of $E_0(Y | A, M^-)$ is now called the targeted maximum likelihood estimator (TMLE) of this true regression function in the

semiparametric regression model. In particular, the TMLE of β_0 is now simply $\beta_n^0 + \varepsilon_n$. Note that the estimation of the clever covariate requires an estimator of $E_0(A | M^-)$. The latter can be carried out with a machine learning algorithm regressing A on M^- . The TMLE of β_0 is consistent if either the initial estimator is consistent or if $E_0(A | M^-)$ is consistently estimated.

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 set M^- . 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. However, 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. This could be a subject-matter driven decision, or one could decide that the flanking markers cannot exceed a correlation of δ with A , and simulations suggest that $\delta = 0.7$ is a good choice (Tuglus and van der Laan, 2008). Another interesting option is to define a set of δ -values, and present the TMLE and corresponding statistical inference for each choice of δ and thereby each δ -specific effect parameter.

4. C-TMLE

One approach is to let the data decide what other markers to include in the estimator of $E_0(A | M^-)$ using a C-TMLE.

4.1. The C-TMLE algorithm

Let Q_n^0 be the initial estimate of Q_0 as 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 $W_1, \dots, W_K$ 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^- with $W = (W_1, \dots, W_K)$. 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 $\Omega = \{W_1, \dots, W_K\}$ be the full collection of main terms. A linear regression model fit g^K of $g_0(W) = E_0(A | W)$ using all main terms in Ω is viewed as the most non-parametric estimate of g_0 . For a given subset of main terms $\mathcal{S} \subset \Omega$, let $\mathcal{S}^c$ be its complement within Ω . For a given subset $\mathcal{S}^*$, we will define g^k as the least squares fit of the linear regression model for $E_0(A | W)$ that includes as main terms all the terms in $\mathcal{S}^*$. In the C-TMLE algorithm we use a forward selection algorithm that augments a given set $\mathcal{S}^*$ into a next set $\mathcal{S}^{k+1}$ obtained by adding the best main term among all main terms in the complement $\mathcal{S}^{k,c}$ of $\mathcal{S}^*$. In other words, the algorithm iteratively updates a current estimate g^k into a new estimate g^{k+1} , but the criterion for g is not measuring 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 $Q_0 = E_0(Y | A, W) = \beta_0 A + E_0(Y | A = 0, W)$. 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 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 off-set, and P_n denotes the empirical probability distribution of $O_1, \dots, O_n$. Here we used the notation $Pf \equiv \int f(o) dP(o)$. That is, an initial estimate Q , an estimate g , and the data $O_1, \dots, O_n$, is mapped into a new TML-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 $(Q^k, \mathcal{S}^k)$ and corresponding TMLEs Q^{k*} , $k = 0, \dots, K$, where Q^k represents an initial estimate, $\mathcal{S}^k$ 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 $\mathcal{S}^k$, as used to define g^k in this k -specific TMLE, increases in k , one unit at the time: $\mathcal{S}^0$ is empty, $|\mathcal{S}^{k+1}| = |\mathcal{S}^k| + 1$, $\mathcal{S}^K = \Omega$. 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:

Initiate algorithm: Set initial TMLE—Let $k = 0$. $Q^k = Q_n^0$ is the initial estimate of Q_0 , and $\mathcal{S}^k$ is the empty set so that g^k is the empirical mean of A . Thus, Q^{k*} is the TMLE estimate 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 $g_0(W) = E_0(A | W)$:

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

If

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

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

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

[In words: If the next best main term added to the fit of $E_0(A | W)$ yields a TMLE of $E_0(Y | A, W)$ 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 $E_0(Y | A, W)$, 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 $k = 1$ till K at which point $S^* = \Omega$. This yields a sequence (Q^k, g^k) and corresponding TMLE Q^{k*} , $k = 0, \dots, K$.

This sequence of candidate TMLEs Q^{k*} of Q_0 has the following property: the estimates g^k are increasingly non-parametric in k and $P_n L(Q^{k*})$ is decreasing in k , $k = 0, \dots, K$. 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 $k = 0, \dots, K$. 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 $E_0(A | M^-)$ such as one based on two flanking markers defined by a choice of window width, across a number of possible window widths. The above C-TMLE algorithm can data adaptively select the window width. 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 .

4.2. 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 criterion for selection of the main terms, then one will easily select main terms that have 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 $E_0(A | M^-)$ 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. $E_0(A | M^s)$, with M^s being a reduction of M^- that is rich enough so that $E_0(Y | A = 0, M^-)$ 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 $E_0(Y | A = 0, M^-)$ and its initial estimator $E_n(Y | A = 0, M^-)$ (see 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 $E_0(Y | A = 0, M^-)$, and the above presented C-TMLE algorithm indeed targets such main terms.

5. Listeria data set

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 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*. There were 131 codominant markers genotyped. When a mouse survived beyond 246 hours it was considered a survivor, and about 30% of the mice survived. This creates a spike in the phenotypic trait distribution, violating the normality assumption in traditional approaches of QTL mapping. We analyzed this data set with C-TMLE and the popular parametric composite interval mapping (CIM) approach (Jansen, 1993; Zeng, 1994).

The outcome Y was defined as the logarithm of the phenotypic trait. In an F2 intercross design, each gene is biallelic and associated with two orthogonal effects that we can estimate. One is the additive effect, which is the substitution effect of one allele replaced by the other. The second is the dominant effect that measures the interaction between the two alleles.

The first step of the C-TMLE is to obtain an initial estimator of $E_0(Y | M)$, which can then be mapped into an initial estimator of $E_0(Y | A, M^-)$, satisfying the semi-parametric regression model. Y can be decomposed into a binary trait of survival or non-survival and a continuous trait of survival time among non-survivors (Broman, 2003). We denote this binary trait of survival with $Z = I(Y = \log 264)$. Then, the expected value of Y given the marker data M can be represented as:

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

In the above formula, $P_0(Z = 1 | M)$ and $P_0(Z = 0 | M)$ 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 learning algorithm (van der Laan et al., 2007) for binary outcomes. $E_0(Y | Z = 0, M)$ is the conditional expectation of Y on M given that the mouse has died, which can be obtained by applying super learning on non-survivors. We refer to this new machine learning algorithm as the 2-part super learner.

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 $E_0(Y | M)$: DSA (Sinisi and van der Laan, 2004), random forests, super learner, and the 2-part super learner. The collection of algorithms in the super learner and 2-part super learner included DSA and random forests and the machine learning algorithms were only provided the additive and dominant markers that had a significant univariate effect based on p -value threshold 0.10. The honest cross-validated risks were as follows: DSA=0.2212, random forests=0.1581, super learner=0.1589, and 2-part super learner=0.1463. 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 honest cross-validated risk and was therefore selected as the estimator of $E_0(Y|M)$. The conditional mean of A given M^- was fit as described in the previous section.

The C-TMLE displays much less noise than CIM. Three additive genes on chromosome 1, 5, and 13 are clearly identified. Two additive effects on chromosome 15 and 18 are borderline significant. In addition, the C-TMLE also detected dominant effects on chromosome 12, 13 and 15. The QTL gene on chromosome 15 is identified to carry 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 chromosome 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 the major genes, however, with less significance and many more suspicious positives. See Figure 1 and Table 1. The analysis, performed in R, took less than 1 hour on a computer with Intel core duo processor running at 1.87 GHz. The required R functions are included in the web-only appendix.

6. Discussion

The C-TMLE incorporates state-of-the-art machine learning for assessing the effects of genes on a phenotype, 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. The algorithm also provides CLT-based statistical inference.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

References

- Boyartchuk VL, Broman KW, Mosher RE, D'Orazio SEF, Starnbach MN, Dietrich WF. Multigenic control of listeria monocytogenes susceptibility in mice. *Nat Genet.* 2001; 27:259–260. [PubMed: 11242105]
- Broman K. Mapping quantitative trait loci in the case of a spike in the phenotype distribution. *Genetics.* 2003; 163:1169–1175. [PubMed: 12663553]
- Haley C, Knott S. A simple regression method for mapping quantitative trait loci in line crosses using flanking markers. *Heredity.* 1992; 69:315–324. [PubMed: 16718932]
- Jansen RC. Interval mapping of multiple quantitative trait loci. *Genetics.* 1993; 135:205–211. [PubMed: 8224820]
- Jin C, Fine JP, Yandell BS. A unified semi-parametric framework for quantitative trait loci analysis, with application to spike phenotypes. *J Am Stat Assoc.* 2007; 102:56–67.
- van der Laan M, Gruber S. Collaborative double robust penalized targeted maximum likelihood estimation. *The International Journal of Biostatistics.* 2010; 6 Article 17.
- van der Laan M, Polley E, Hubbard A. Super learner. *Statistical Applications in Genetics and Molecular Biology.* 2007; 6 Article 25.
- van der Laan M, Rubin D. Targeted maximum likelihood learning. *The International Journal of Biostatistics.* 2006; 2 Article 1.
- Sinisi S, van der Laan M. Deletion/Substitution/Addition algorithm in learning with applications in genomics. *Statistical Applications in Genetics and Molecular Biology.* 2004; 3 Article 18.

- Tuglus, C.; van der Laan, M. Technical Report 233. Division of Biostatistics, University of California, Berkeley; 2008. Targeted Methods for Biomarker Discovery, the Search for a Standard. <http://www.bepress.com/ucbbiostat/paper233/>
- Zeng Z. Precision mapping of quantitative trait loci. *Genetics*. 1994; 136:1457–1468. [PubMed: 8013918]

Figure 1. P -value Profiles

The genome-wide p -value profile for additive and dominant effects in Listeria dataset where p -values are FDR adjusted and on a negative log10 scale. (a) profile for CIM; (b) profile for C-TMLE. Solid lines represent additive effects, dashed lines represent dominant effects, and dash/dot lines indicate 0.05 p -value threshold.

Effect estimates and Positions

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

Table 1

QTL ID	Type	CIM			C-TMLE		
		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	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	0.1960
9	dom	6	33.8	-0.1235	-	-	-
10	dom	12	41.8	-0.2352	12	40.1	-0.1372
11	add	13	22.7	-0.3409	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	22.1	0.0678
14	dom	15	12.0	0.2042	15	22.1	0.1438
15	add	18	-	-	18	14.1	-0.0692