Variable Selection by Bayesian Adaptive Lasso and Iterative Adaptive Lasso, with Application for Genome-wide Multiple Loci Mapping

Wei Sun∗ Joseph G. Ibrahim†
Fei Zou‡

∗University of North Carolina at Chapel Hill, wsun@bios.unc.edu
†University of North Carolina at Chapel Hill, ibrahim@bios.unc.edu
‡University of North Carolina at Chapel Hill, fzou@bios.unc.edu

This working paper is hosted by The Berkeley Electronic Press (bepress) and may not be commercially reproduced without the permission of the copyright holder.

http://biostats.bepress.com/uncbiostat/art10

Copyright ©2009 by the authors.
Variable Selection by Bayesian Adaptive Lasso and Iterative Adaptive Lasso, with Application for Genome-wide Multiple Loci Mapping

Wei Sun, Joseph G. Ibrahim, and Fei Zou

Abstract

Despite many recent methodological developments, variable selection in high-dimensional settings where the number of covariates (p) is larger than the sample size (n) remains a difficult problem, especially when the covariates with zero coefficients are correlated with some covariates with nonzero coefficients. One such example is genome-wide multiple loci mapping with dense genetic markers, where the number of covariates (i.e., the number of genetic markers) are often larger than the sample size and nearby markers often share similar genotype profiles due to linkage or linkage disequilibrium. The adaptive Lasso (Zou, H. 2006) is a state-of-the-art method for simultaneous variable selection and estimation in the setting of linear regression. However, it requires consistent initial estimates of the regression coefficients, which are generally not available in the aforementioned high-dimensional settings. In this paper, we propose two variable selection methods: the Bayesian adaptive Lasso and the iterative adaptive Lasso. These two methods extend the adaptive Lasso in the sense that they do not require any informative initial estimates of the regression coefficients. We systematically evaluate the variable selection performance of the proposed methods as well as several existing methods within the framework of genome-wide multiple loci mapping. We show that the proposed methods have improved variable selection performance compared to most existing methods and the iterative adaptive Lasso also has superior computational efficiency.
Variable Selection by Bayesian Adaptive Lasso and Iterative Adaptive Lasso, with Application for Genome-wide Multiple Loci Mapping

Wei Sun\textsuperscript{1,2,*}, Joseph G. Ibrahim\textsuperscript{1}, and Fei Zou\textsuperscript{1}

\textsuperscript{1}Department of Biostatistics, University of North Carolina at Chapel Hill, Chapel Hill, NC, U.S.A.
\textsuperscript{2}Department of Genetics, University of North Carolina at Chapel Hill, Chapel Hill, NC, U.S.A.

*email: wsun@bios.unc.edu

SUMMARY: Despite many recent methodological developments, variable selection in high dimensional settings where the number of covariates ($p$) is larger than the sample size ($n$) remains a difficult problem, especially when the covariates with zero coefficients are correlated with some covariates with nonzero coefficients. One such example is genome-wide multiple loci mapping with dense genetic markers, where the number of covariates (i.e., the number of genetic markers) are often larger than the sample size and nearby markers often share similar genotype profiles due to linkage or linkage disequilibrium. The adaptive Lasso (Zou, H. 2006) is a state-of-the-art method for simultaneous variable selection and estimation in the setting of linear regression. However, it requires consistent initial estimates of the regression coefficients, which are generally not available in the aforementioned high-dimensional settings. In this paper, we propose two variable selection methods: the Bayesian adaptive Lasso and the iterative adaptive Lasso. These two methods extend the adaptive Lasso in the sense that they do not require any informative initial estimates of the regression coefficients. We systematically evaluate the variable selection performance of the proposed methods as well as several existing methods within the framework of genome-wide multiple loci mapping. We show that the proposed methods have improved variable selection performance compared to most existing methods and the iterative adaptive Lasso also has superior computational efficiency.

KEY WORDS: Adaptive Lasso, Bayesian Adaptive Lasso, Iterative Adaptive Lasso, multiple loci mapping
1. Introduction

Variable selection by penalized regression or penalized likelihood have attracted increasing research interest recently (Fan and Li, 2001; Zou, 2006; Zou and Li, 2008). A state-of-the-art method is the adaptive Lasso (Zou, 2006), which extends Lasso (Tibshirani, 1996) by allowing different penalization parameters for different regression coefficients. Zou (2006) proved that the adaptive Lasso satisfies the oracle property (Fan and Li, 2001), i.e., the covariates with nonzero coefficients will be selected with probability tend to 1, and the estimators of nonzero coefficients have the same asymptotic distribution as the correct model.

However, the adaptive Lasso requires consistent initial estimates of the regression coefficients, which are generally not available in high dimensional settings where the number of covariates ($p$) is larger than the sample size ($n$). Huang et al. (2008) showed that under a partial orthogonality condition, in which the covariates with zero coefficients are weakly correlated with the covariates with nonzero coefficients, the adaptive Lasso still enjoys the oracle property in high-dimensional settings, with initial estimates obtained from the marginal regression. However, in many real-world problems, the covariates with zero coefficients are often strongly correlated with some covariates with nonzero coefficients. In this paper, we focus on one such problem: genome-wide multiple loci mapping with dense genetic markers.

It is well known that complex traits, including many common diseases, are controlled by multiple loci (Hoh and Ott, 2003). However, multiple loci mapping remains one of the most difficult problems in genetic studies, mainly due to the high dimensionality of genetic variants (given limited sample size) and incomplete genotype data. Recently, with the applications of high density SNP (Single Nucleotide Polymorphism) arrays and next generation sequencing techniques, genome-wide genotype data can be measured in a complete (or close to be complete) and accurate manner. Therefore the difficulty of multiple loci mapping is mainly the difficulty of variable selection in high dimensional settings. Throughout this paper, we
assume the traits are quantitative, although our methods can be modified for qualitative traits by the surrogate variable approach similar to Huang et al. (2007).

Suppose a quantitative trait and the genotype profiles of $p$ markers (e.g., SNPs) are measured in $n$ individuals. With the assumption of no epistatic interactions, we employ an additive linear model:

$$y_i = b_0 + \sum_{j=1}^{p} x_{ij}b_j + e_i,$$

where $y_i$ ($i = 1, ..., n$) is the trait value of individual $i$, $b_0$ is the intercept, $e_i \sim N(0, \sigma^2_e)$ is the residual error, and $x_{ij}$ is the genotype of the $j$-th marker of individual $i$. Let $y = (y_1, ..., y_n)^T$, $X = (x_{ij})_{n \times p}$, $b_0 = (b_0, ..., b_0)^T$, $b = (b_1, ..., b_p)^T$, and $e = (e_1, ..., e_n)^T$, equation (1) can be written as $y = b_0 + Xb + e$. We use this matrix form in some of our later derivations to simplify the notation. The specific coding of $x_{ij}$ depends on the study design and the inheritance model. For example, suppose the $j$-th marker has three possible genotypes: AA, AB, and BB. If additive inheritance is assumed, $x_{ij}$ can be coded as 0, 1, 2 for genotype AA, AB, and BB, respectively. If dominant inheritance of allele B is assumed, $x_{ij}$ can be coded as 0, 1, 1 for genotype AA, AB, and BB, respectively. The major objective of multiple loci mapping is to identify the correct subset model, i.e., to identify those $j$s, such that $b_j \neq 0$, and estimate the $b_j$'s. Despite their importance, epistatic interactions are omitted in this model for the following two reasons. First, even multiple loci mapping with this simple additive model is not well studied, especially in the high-dimensional settings. Secondly, a strictly additive model simplifies the methodology development and lays out the framework for extension to epistatic interaction mapping, which will be elaborated in the Discussion Section.

Marginal regression and step-wise regression are commonly used for multiple loci mapping. Because different markers are considered separately in these methods, a key question is how to choose a genome-wide threshold to account for the multiple tests across all the markers.
For marginal regression, a permutation-based threshold can be estimated for genome-wide maximum linkage/association statistic (Churchill and Doerge, 1994). Doerge and Churchill (1996) proposed two permutation-based thresholds for forward selection: the conditional empirical threshold (CET) and the residual empirical threshold (RET). Broman and Speed (2002) proposed a model selection criterion named BIC$_\delta$, which is further rewritten to a penalized LOD score criterion and implemented within a forward-backward model selection framework (Manichaikul et al., 2008), where the threshold of the penalized LOD score is estimated by permutations.

Several simultaneous multiple loci mapping methods have been developed, among which two commonly used approaches are Bayesian shrinkage estimates and Bayesian model selection. The Bayesian shrinkage method by Xu (2003) is a hierarchical model based on the additive linear model as equation (1), where each regression coefficient $b_j$ ($1 \leq j \leq p$) follows a normal distribution $N(0, \sigma_j^2)$. The coefficients are shrunk because their prior mean values are set to 0. The degree of shrinkage is related to the prior specification of the covariate-specific variance $\sigma_j^2$. Xu (2003) employed an improper prior $p(\sigma_j^2) \propto 1/\sigma_j^2$. Later Yi and Xu (2008) generated it to an inverse-Gamma prior,

$$p(\sigma_j^2|\delta, \tau) = \text{inv-Gamma}(\delta, \tau) = \frac{\tau^{\delta}}{\Gamma(\delta)}(\sigma_j^2)^{-1-\delta} \exp(-\tau/\sigma_j^2).$$

Note that $p(\sigma_j^2) \propto 1/\sigma_j^2$ is the limiting case of the inverse-Gamma prior as the hyperparameters $\delta \to 0$ and $\tau \to 0$. The corresponding unconditional prior of $b_j$ (i.e., $\int p(b_j|\sigma_j^2)p(\sigma_j^2)d\sigma_j^2$) is a Student’s t distribution. Therefore we refer to this hierarchical model as the Bayesian t.

Another choice for the prior of $\sigma_j^2$ is an exponential distribution (Yuan and Lin, 2005; Park and Casella, 2008; Yi and Xu, 2008), given by

$$p(\sigma_j^2|\kappa^2/2) = \text{Exp}(\kappa^2/2) = \frac{\kappa^2}{2} \exp\left(-\frac{\kappa^2}{2}\sigma_j^2\right),$$

where $\kappa$ is a hyper-parameter. In this case, the corresponding unconditional prior of $b_j$ is a double exponential distribution: $p(b_j) = \frac{\kappa}{2} e^{-\kappa|b_j|}$ that is closely related to the Bayesian
interpretation of the Lasso (Tibshirani, 1996). Therefore this hierarchical model has been named the \textit{Bayesian Lasso} (Park and Casella, 2008)\footnote{In fact, Park and Casella used an alternative setup and assign the priors of $b_j$ as $p(b_j | \sigma_j^2) \sim N(0, \sigma_j^2 \sigma_j^2)$. This setup has the advantage that the joint posterior of $(\mathbf{b}, \sigma_j^2)$ has at most one mode. However it dose not lead to any apparent improvements (regarding $b_j$) over the prior of $p(b_j | \sigma_j^2) \sim N(0, \sigma_j^2)$ in both the real dataset of Yi and Xu (2008) and in our simulation dataset.}.

Several Bayesian model selection methods have been applied for multiple loci mapping, for example, the reversible jump Markov Chain Monte Carlo (MCMC) method (Richardson and Green, 1997), the stochastic search variable selection (SSVS) method (Yi et al., 2003), and the composite model space approach (CMSA) (Yi, 2004). The CMSA embraces both the reversible jump MCMC method and the SSVS method. The composite space refers to the parameter space for all the markers, either the marker is within the current model or not. All the parameters in the composite space are recorded to increase the computational efficiency.

In this paper, we propose two novel variable selection methods: the Bayesian adaptive Lasso (BAL) and the iterative adaptive Lasso (IAL). The BAL is a Bayesian shrinkage method and the IAL is an Expectation/Conditional Maximization (ECM) algorithm (Meng and Rubin, 1993) motivated by the BAL. These two methods extend the adaptive Lasso in the sense that they do not require any informative initial estimates of the regression coefficients. We compare the variable selection performance of these two methods together with several existing methods by extensive simulations. Both the BAL and the IAL have superior variable performance than most existing methods in most simulation situations. The IAL also has superior computational efficiency.

The remainder of this paper is organized as follows. We first introduce the BAL and the IAL in Sections 2 and 3, respectively. We then evaluate them and several representative existing methods by extensive simulations under various situations in Section 4. Finally, we summarize and discuss the implications of our methodology in Section 5.
2. The Bayesian adaptive Lasso (BAL)

The BAL is a Bayesian hierarchical model. The priors are chosen as follows:

\[
p(b_0) \propto 1, \\
p(\sigma_e^2) \propto 1/\sigma_e^2, \\
p(b_j|\kappa_j) = \frac{1}{2\kappa_j} \exp\left(-\frac{|b_j|}{\kappa_j}\right), \\
p(\kappa_j|\delta, \tau) = \text{inv-Gamma}(\kappa_j; \delta, \tau) \propto \tau^{-1} \frac{1}{\Gamma(\delta)} \kappa_j^{-1-\delta} \exp\left(-\frac{\tau}{\kappa_j}\right),
\]

where \( \delta > 0 \) and \( \tau > 0 \) are two hyperparameters. Note different from the Bayesian t and the Bayesian Lasso, which use normal prior for \( b_j \), the BAL use double exponential prior for \( b_j \) so that each \( b_j \) has a Lasso-type of penalization parameter \( \kappa_j \), as in the adaptive Lasso.

One way to interpret the roles of \( \delta \) and \( \tau \) is to examine the unconditional prior of \( b_j \):

\[
p(b_j) \propto \int_0^{\infty} \kappa_j^{-2-\delta} \exp\left(-\frac{(|b_j| + \tau)}{\kappa_j}\right) d\kappa_j \propto (|b_j| + \tau)^{-1-\delta},
\]

which shows that \( \delta \) and \( \tau \) control the penalization of the coefficients \( b_j \). Larger \( \delta \) and smaller \( \tau \) lead to bigger penalization. In practice, it could be difficult to choose specific values for \( \delta \) and \( \tau \). We suggest a joint improper prior \( p(\delta, \tau) \propto \tau^{-1} \), and let the data estimate \( \delta \) and \( \tau \).

The posterior distribution of all the parameters is given by

\[
p(b, b_0, \sigma_e^2, \kappa_1, ..., \kappa_p|y, X) \\
\propto p(y|b, X, b_0, \sigma_e^2)p(\sigma_e^2) \prod_{j=1}^{p} p(b_j|\kappa_j)p(\kappa_j)p(\delta, \tau) \\
\propto \frac{1}{\sigma_e^{2+n}} \exp\left[-\frac{\text{RSS}}{2\sigma_e^2}\right] \frac{\tau^{\delta p-1}}{(\Gamma(\delta))^p} \prod_{j=1}^{p} \kappa_j^{-2-\delta} \exp\left(-\frac{|b_j| + \tau}{\kappa_j}\right),
\]

where \( \text{RSS} \) indicates residual sum of squares, i.e., \( \text{RSS} = \sum_{i=1}^{n} \left(y_i - b_0 - \sum_{j=1}^{p} x_{ij} b_j\right)^2 \). We sample from this posterior distribution using the following Gibbs sampler:

(1) Initialization. We initialize \((b_0, b_1, ..., b_p)\) with zero and initialize \((\sigma_e^2, \kappa_1, ..., \kappa_p)\) with a positive number.

(2) Update \( b_0 \): Let \( \Theta_{-\text{parameter}_1} \) denote all of the parameters except parameter\(_1\). The
conditional posterior distribution of $b_0$ is

$$p(b_0|\Theta) \sim N(\bar{b}_0, s_0^2), \quad (10)$$

where $\bar{b}_0 = (1/n) \sum_i^n (y_i - \sum_{j=1}^p x_{ij} b_j)$, and $s_0^2 = (1/n) \sigma_e^2$.

(3) Update $b_j$: The conditional posterior distribution of $b_j$ is

$$p(b_j|\Theta) \propto \exp \left[ -\frac{(b_j - \bar{b}_j)^2}{2\sigma_j^2} - \frac{|b_j|}{\kappa_j} \right], \quad (11)$$

where

$$\sigma_j^2 = \frac{\sigma_e^2}{\sum_{i=1}^n x_{ij}^2}, \quad \text{and} \quad \bar{b}_j = \left( \sum_{i=1}^n x_{ij}^2 \right)^{-1} \sum_{i=1}^n x_{ij} \left( y_i - b_0 - \sum_{k \neq j} x_{ik} b_k \right). \quad (12)$$

Although $p(b_j|\Theta)$ has no closed form, it is log-concave. Thus we sample $b_j$ by Adaptive Rejection Sampling (ARS) (Gilks, 1992). Note that whenever one of the $b_j$’s is updated, it is used immediately for updating the other $b_j$’s.

(4) Update $\sigma_e^2$:

$$p(\sigma_e^2|\Theta) \propto (\sigma_e^2)^{-1-n/2} \exp \left[ -\frac{\text{rss}}{2(2\sigma_e^2)} \right], \quad (13)$$

which is inv-Gamma($\sigma_e^2; n/2, \text{rss}/2$).

(5) Update $\kappa_j$:

$$p(\kappa_j|\Theta) \propto \kappa_j^{-2-\delta} \exp \left( -\frac{|b_j| + \tau}{\kappa_j} \right), \quad (14)$$

which is inv-Gamma($\kappa_j; 1 + \delta, |b_j| + \tau$).

(6) Update $\tau$:

$$p(\tau|\Theta) \propto \tau^{p-1} \exp \left( -\tau \sum_{j=1}^p \kappa_j^{-1} \right). \quad (15)$$

Therefore $\tau|\Theta \sim \Gamma(p\delta, \sum_{j=1}^p \kappa_j^{-1})$.

(7) Update $\delta$:

$$p(\delta|\Theta) \propto \left( \frac{\tau^p}{\prod_{j=1}^p \kappa_j} \right)^\delta \Gamma(\delta)^{-p}. \quad (16)$$

It is easy to show that $p(\delta|\Theta)$ is log-concave, so we sample $\delta$ using ARS.
3. The Iterative Adaptive Lasso (IAL)

One shortcoming of the Bayesian shrinkage methods is that few coefficients would be shrunk to zero, therefore leaving an open question: how to identify an appropriate threshold to select covariates for the final model? In addition, these Bayesian methods are implemented by MCMC, and are computationally intensive. Based on the BAL, we propose the iterative adaptive Lasso (IAL), which is an Expectation/Conditional Maximization (ECM) algorithm (Meng and Rubin, 1993). The IAL tends to penalize most coefficients to 0, and thus generates small models. In addition, it is computationally much more efficient than the Bayesian methods. Specifically, under the setup of the BAL (equations (4)-(7)), we treat \( \theta = (b_0, b_1, ..., b_p) \) as the parameter of interest and let \( \phi = (\sigma^2_e, \kappa_1, ..., \kappa_p) \) be the missing data. The observed data are \( y_i \) and \( x_{ij} \). The IAL is implemented as follows:

(1) Initialization: We initialize \( (b_0, b_1, ..., b_p) \) with zero and initialize \( (\sigma^2_e, \kappa_1, ..., \kappa_p) \) with a positive number.

(2) Conditional Maximization (CM) step:

(a) Update \( b_0 \),

\[
b_0 = (1/n) \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{p} x_{ij} b_j \right).
\]  

(b) For \( j = 1, ..., p \), update \( b_j \)

\[
\begin{cases}
  b_j = 0 & \text{if } -\sigma^2_j/\kappa_j \leq \bar{b}_j \leq \sigma^2_j/\kappa_j \\
  b_j = \bar{b}_j - \sigma^2_j/\kappa_j & \text{if } \bar{b}_j > \sigma^2_j/\kappa_j \\
  b_j = \bar{b}_j + \sigma^2_j/\kappa_j & \text{if } \bar{b}_j < -\sigma^2_j/\kappa_j
\end{cases}
\]

where

\[
\sigma^2_j = \frac{\sigma^2_e}{\sum_{i=1}^{n} x_{ij}^2}, \quad \text{and} \quad \bar{b}_j = \left( \sum_{i=1}^{n} x_{ij}^2 \right)^{-1} \sum_{i=1}^{n} x_{ij} \left( y_i - b_0 - \sum_{k \neq j} x_{ik} b_k \right).
\]  

(3) Expectation (E) step:

With the updated \( b_j \)'s, recalculate the residual sum of squares, \( \text{rss} \), and
(a) Update $\sigma^2_e$: $\sigma^2_e = \text{rss}/n$.

(b) Update $\kappa_j$: $\kappa_j = (|b_j| + \tau)/(1 + \delta)$.

We say the algorithm is converged if the coefficient estimates $\hat{b}_0, \hat{b}_1, ..., \hat{b}_p$ have little change across $L$ iterations. Then we output the average coefficients across the $L$ iterations.

The updates in the CM-step are simply to replace $b_0, b_1, ..., b_p$ with their MLE based on the corresponding conditional distributions (see Web Appendix A). Note that the CM-step update is actually quite similar to the shooting method for the Lasso calculation (Fu, 1998).

The E-step is less obvious. We briefly discuss the derivations here and give the details in Web Appendix A. Because we treat $\phi = (\sigma^2_e, \kappa_1, ..., \kappa_p)$ as missing data, we are only interested in the following complete data log-posterior of $\theta = (b_0, b_1, ..., b_p)$:

$$l(\theta|y, X, \phi) = C - \frac{\text{rss}}{2\sigma^2_e} - \sum_{j=1}^{p} \frac{|b_j|}{\kappa_j},$$

where $C$ is a constant with respect to $\theta$. Suppose in the $t$-th iteration, the parameter estimates are $\theta^{(t)} = (b_0^{(t)}, b_1^{(t)}, ..., b_p^{(t)})$. Then the conditional expectation of $l(\theta|y, X, \phi)$ with respect to the conditional density of $f(\phi|y, X, \theta^{(t)})$ is

$$Q(\theta|\theta^{(t)}) = \int l(\theta|y, X, \phi)f(\phi|y, X, \theta^{(t)})d\phi.$$  

After some derivations (see Web Appendix A), we can show that

$$Q(\theta|\theta^{(t)}) = C - (\text{rss}/2)/(\text{rss}^{(t)}/n) - \sum_{j=1}^{p} \frac{|b_j|}{\left(|b_j^{(t)}| + \tau\right)/(1 + \delta)},$$

where $\text{rss}^{(t)}$ is the residual sum of squares calculated based on $\theta^{(t)} = (b_0^{(t)}, b_1^{(t)}, ..., b_p^{(t)})$. Comparing equation (19) and (21), it is obvious that in order to obtain $Q(\theta|\theta^{(t)})$, we can simply update $\sigma^2_e = \text{rss}^{(t)}/n$ and $\kappa_j = (|b_j^{(t)}| + \tau)/(1 + \delta)$.

The asymptotic covariance matrix of the parameters in an ECM algorithm can be obtained by a Supplemented ECM algorithm (Meng and Rubin, 1991) from the complete-data covariance matrix. However, in the situation of multiple loci mapping with dense markers, the complete data covariance matrix is not full rank and due to the high correlations between the...
Bayesian adaptive Lasso and Iterative Adaptive Lasso

genotype profiles, the estimation of the covariance matrix could be highly unstable. Therefore we suggest using IAL as a variable selection method. After a small subset of markers are selected, the variance-covariance matrix can then be estimated.

4. Simulation Studies

In this section, we seek to evaluate the variable selection performance of marginal regression, forward regression, forward-backward regression (with penalized LOD as the model selection criterion), the composite model space approach (CMSA), the iterative adaptive Lasso (IAL), and the three Bayesian shrinkage methods: the Bayesian t, the Bayesian Lasso, and the Bayesian adaptive Lasso (BAL).

4.1 Simulation setup

Although simulated data can never perfectly represent real data, carefully controlled simulations often provide more specific evaluations of different methods than a real data analysis. On the other hand, simulations can also be tailored to produce more realistic data, though with the price of loss of specificity. With this trade-off in mind, we conduct two sets of simulation studies. In simulation study I, we first simulate genotype data, and then simulate the trait based on fixed quantitative trait loci (QTL). In simulation study II, we use real genotype data, and then simulate the trait based on randomly selected QTL in each simulation. The first set of simulations are better controlled since the genotype data is simulated and the QTL are fixed. Summarizing results across simulations is to summarize results across different realizations of one trait. The second set of simulations is more realistic since we use real genotype data; and it is more general since the QTL are randomly selected in each simulation so that summarizing results across simulations is to summarize results across different traits with similar underlying genetic components. In both simulation studies, dense genetic marker maps are used.

http://biostats.bepress.com/uncbiostat/art10
In simulation study I, we first simulate a marker map of 1,200 equally spaced markers from 20 chromosomes of length 90 cM, with 60 markers per chromosome (using function sim.map in R/qtl (Broman et al., 2003)). Then we simulate genotype data of 360 F2 samples based on the simulated marker map (using function sim.cross in R/qtl). In simulation study II, we use real genotype data from 266 F2 progeny of two inbred mouse strains: C3H/HeJ (apoE null) and C57BL/6J (apoE null) (Wang et al., 2006). The genotype data are available on 1295 SNPs from 19 autosomes at an average density of one SNP per 1.5 cM. There are less than 1% of missing values in this genotype dataset. We impute the missing values using the most likely genotype given the observed multipoint marker data (the Viterbi algorithm implemented in the function fill.geno of R/qtl) and then treat the imputed genotype data as known in the following studies. After imputation, the markers with the same genotype profiles are combined and we end up with 1025 distinct genotype profiles.

In each simulation study, we simulate quantitative traits in six situations with 100 simulations per situation. In simulation study I, ten markers are chosen as fixed QTL in each situation. In simulation study II, ten markers are randomly chosen as QTL in each simulation. Given the ten QTL, the trait is simulated based on the additive linear model of equation (1), where genotype \((x_{ij})\) is coded by the number of minor alleles, and the variance of residual error \((\sigma^2_e)\) is referred to as the environmental variance in the following discussions. The QTL effect sizes across the six situations are listed below, which are the same for simulation studies I and II:

1. Ten QTL, one per chromosome, with effect sizes 0.5, 0.4, -0.4, 0.3, 0.3, -0.3, 0.2, 0.2, -0.2, and -0.2; \(\sigma^2_e = 1\).
3. Ten QTL, two per chromosome, with effect sizes for each QTL pair from the same chromosome as (0.5, 0.3), (-0.4, -0.4), (0.3, 0.3), (0.2, 0.2), and (-0.2, -0.2); \(\sigma^2_e = 1\).
5. Ten QTL, two per chromosome, with effect sizes for each QTL pair from the same chromosome as $(0.5, -0.3), (0.4, -0.4), (0.3, -0.3), (0.2, -0.2),$ and $(0.2, -0.2); \sigma_e^2 = 1.$ Situations 2, 4, and 6 are the same as simulations 1, 3, and 5, respectively, except that \( \sigma_e^2 = 0.5. \) In summary, using the language of Broman and Speed (2002), situations 1-2 simulate un-linked QTL, situations 3-4 simulate QTL linked in coupling, and situations 5-6 simulate QTL linked in repulsion. The locations and effect sizes of the QTL in simulation study I are illustrated in Figure 1.

Because different QTL may have correlated genotype profiles (especially for linked QTL), the overall genetic effect is not simply the summation of their individual effects. We calculate the proportion of trait variance explained by the ten QTL in our simulated data and summarize them as below (Table 1).

4.2 The implementations of different methods

We have implemented marginal regression, forward regression, the Bayesian t, Bayesian Lasso, BAL, and IAL in an R package mlm (multiple loci mapping), with computationally intensive parts implemented by C code\(^2\). The R package mlm can be downloaded from http://www.bios.unc.edu/\~{}wsun/software/.

Because there are no missing values in genotype data, the F-statistic is proportional to \( r^2 \). For both marginal regression and forward regression, we simply use \( r^2 \) as the test-statistic because it can be calculated efficiently. We use 10,000 permutations to calculate the \( r^2 \)

\(^2\)Yi and Xu (2008) have implemented the Bayesian t and Bayesian Lasso using WinBUGS, which is easy to program but not efficient enough for our large scale simulations here. Therefore we have implemented both methods using C code. Our Gibbs samplers are mainly based on Yi and Xu (2008), but with small modifications for the Bayesian t to further improve its computational efficiency. We present the details of the Gibbs samplers for both methods in Web Appendix B and C.
cutoff corresponding to the permutation p-value of 0.05. For marginal regression, we only keep the most significantly linked marker in each chromosome to eliminate redundant loci, a strategy that has been used elsewhere (Wang et al., 2006). Some criteria can be used to dissect multiple QTL in one chromosome from the results of marginal regression. However, the implementation of such criteria requires ad-hoc considerations and is beyond the scope of this paper. Both CET and RET can be used as thresholds in forward regression. CET suffers from the “curse of dimensionality” because the number of permutation constraints increases exponentially as the model size increases; while RET is less powerful if the structure model assumption is wrong (Doerge and Churchill, 1996). In our simulations, the underlying structure model is known, and thus we use RET to decide the cutoff for forward regression.

We employ the function \texttt{stepqtl} in R/qtl (Broman et al., 2003) for the forward-backward regression with penalized LOD score as the model selection criterion. The function \texttt{stepqtl} also allows us to add two loci into the model each time. We use this option for simulations where we have two QTL from one chromosome.

There are two options for the priors of the Bayesian Lasso: \( p(b_j | \sigma_j^2) \sim N(0, \sigma_j^2) \) (Yi and Xu, 2008), and \( p(b_j | \sigma_j^2) \sim N(0, \sigma_e^2 \sigma_j^2) \) (Park and Casella, 2008), where \( \sigma_e^2 \) is the variance of the residual errors. The results we shall discuss are based on the former, while the latter yields similar results (data not shown). The Bayesian Lasso uses two hyperparameters \( r \) and \( s \) to specify the prior of \( \kappa^2/2 \) as Gamma\( (s, r) \). We set \( r = 0.01 \) and \( s = 0.01 \). Values smaller than 0.1 yield similar results (in terms of the number of true/false discoveries) while values larger than 0.1 yield slightly worse results.

We set the convergence criterion for the IAL as the consecutive square difference of any regression coefficient is smaller than \( 10^{-10} \) for at least 10 iterations, which essentially means that the coefficients would not change anymore. The parameter \( \tau \) in the IAL is set to \( 10^{-3} \). Values from \( 10^{-4} \) to \( 10^{-2} \) do not significantly change the results. The parameter \( \delta \) is set
to 0 because small $\delta$ values have little effect on the updates and large $\delta$ values lead to over-shrinkage.

R/qtlbim (Yandell et al., 2007) is used for the computation of the CMSA. We chose not to carry out interval mapping because the genetic markers are already dense enough and we have simulated the QTL on marker locations. The CMSA method requires an additional input, namely the expected number of QTL. We supply this parameter with the true number of simulated QTL. For each marker, we record its posterior probability belonging to the true model from the output of the CMSA.

All the Bayesian methods used here rely on MCMC methods to sample from the posterior distributions, and therefore a practical issue is how to decide on the number of burn-in iterations and the sizes of the final MCMC samples. To compare different methods in a fair manner, we apply the same number of burn-in iterations and final samples: 10,000 burn-in iterations followed by 10,000 iterations to obtain 1000 samples, one from every 10 iterations. To monitor convergence, we calculate the Gelman and Rubin scale reduction parameter ($\hat{R}$) (for 10 parallel chains) for each coefficient across all the simulation situations. The Bayesian Lasso, Bayesian t, and BAL all have acceptable $\hat{R}$ values with the vast majority of the $\hat{R}$'s smaller than 1.05.

4.3 Results evaluation

Because the vast majority of the coefficients/posterior probabilities are not zero in the Bayesian Lasso, Bayesian t, BAL, and CMSA, an additional cutoff is needed to select the final model. In a real data study of a single trait, researchers may examine the genome-wide results and use ad-hoc criteria to choose loci for follow up studies. However, to analyze the results from a large number of simulations, a more objective approach is to examine the results across different cutoffs. Although a coefficient cutoff is not crucial for IAL because most coefficients are already penalized to 0, IAL still benefits from filtering out covariates.
with small coefficients (in absolute value). Therefore we evaluate the performance of different methods by comparing the number of true discoveries and false discoveries across different cutoffs of coefficient size or posterior probability. Given a cutoff, we can obtain a final model. We count the number of true discoveries in the final model as follows. For each of the true QTL, we check whether any maker in the final model satisfies the following three criteria: (1) it is located on the same chromosome as the QTL, (2) it has the same effect direction (sign of the coefficient) as the QTL, (3) the $r^2$ between this marker and the QTL is bigger than 0.8. If there is no such maker, there is no true discovery for this QTL. Otherwise, if there is at least one such maker, the one with the highest $r^2$ with the QTL is recorded as a true discovery and it is excluded from searching the true discoveries of other QTL. After the true discoveries of all the QTL are identified, the remaining markers included in the model are defined as false discoveries. These false discoveries are further divided into two classes: false discoveries linked to at least one QTL (linked false discoveries) and false discoveries un-linked to any QTL (unlinked false discoveries). A false discovery is linked to a QTL if it satisfies the above three criteria.

We summarize the results of each method by an ROC-like curve that plots the median number of true discoveries versus the median number of false discoveries across different cutoff values. The methods with ROC-like curves closer to the upper-left corner of the plot have better variable selection performance because they have less false discoveries and more true discoveries. It is possible that a few cutoff values correspond to the same median of false discoveries but different medians of true discoveries. In this case, the largest median of true discoveries is plotted to simplify the figure. In other words, these ROC-like curves illustrate the best possible performance of different methods. For marginal regression, forward regression, and forward-backward regression, we simply use the cutoff corresponding to permutation p-value 0.05.
4.4 Simulation study I, simulated genotype data and fixed QTL

The results of simulation study I, depending on whether we count all the false discoveries or only those unlinked false discoveries, are presented in Figures 2 and 3, respectively. Forward regression greatly outperforms marginal regression in most situations. Therefore we omit the results for marginal regression for readability of the figures.

First, we examine all methods by counting the total number of false discoveries (Figure 2). For unlinked QTL, step-wise regression (forward or forward-backward regression) select more true discoveries than any other method at a median false discovery of one QTL. When QTL are linked in coupling, the IAL performs better than any other method. When QTL are linked in repulsion, the IAL and the CMSA are the best choices for relatively larger environmental variance, and forward-backward regression select more true discoveries than any other method (when the number of false discoveries is zero) for relatively smaller environmental variance.

Next, only the unlinked false discoveries are counted (Figure 3). For unlinked QTL, the step-wise regression, CMSA, and BAL all have similar variable selection performance and are better than the other methods. When QTL are linked in coupling, the IAL, BAL, CMSA, and Bayesian t perform better than the other methods. When QTL are linked in repulsion, the CMSA has the best variable selection performance for relatively larger environmental variance, and both the BAL and the CMSA are superior than the other methods for relatively smaller environmental variance.
4.5 Simulation study II, real genotype data and randomly selected QTL

The results of simulation study II, depending on whether we count all the false discoveries or only those unlinked false discoveries, are presented in Web Figure 1 and Web Figure 2, respectively. Similar to the previous section, we suppress the results of marginal regression since forward regression greatly outperforms it in most situations.

We first count the total number of false discoveries. The step-wise regression methods and the IAL have better variable selection performance than any other method in all six situations. The CMSA can achieve similar performance for QTL linked in coupling with relatively larger environmental variance and for QTL linked in repulsion with relatively smaller environmental variance. Comparison of the IAL and the step-wise regression methods is more tricky since it depends on how many false discoveries are acceptable. Nevertheless, their performance is similar if there is one false discovery for unlinked QTL or QTL linked in coupling, and zero false discovery for QTL linked in repulsion.

Next, we count the number of false discoveries by the number of unlinked false discoveries. For unlinked QTL, the step-wise methods, BAL, and CMSA have the best variable selection performance, and when the environmental variance is relatively small, the IAL also achieves similar performance. When QTL are linked in coupling, the BAL, CMSA, and Bayesian t all have the best variable selection performance. When QTL are linked in repulsion, the CMSA has the best performance and the IAL have similar performance if the environmental variance is relatively small.

4.6 Degree of penalization

One important factor affecting the performance of a shrinkage method is how much the parameters are shrunk. To further explore shrinkage characteristics of the three Bayesian shrinkage methods and the IAL, we examine the sizes of the output coefficients. Figure 4 shows the distributions of the largest coefficients (in absolute value) from the 100 simulations
of simulation study I, situation 2 (unlinked QTL and $\sigma_e^2 = 0.5$). It is obvious from this figure that if we rank these four methods by their coefficient sizes, Bayesian Lasso $< \text{Bayesian t} < \text{BAL} < \text{IAL}$. The same ranking is observed for the other simulation situations from either simulation study. Recall that in our simulations, the coefficients are within the range of $[0.2, 0.5]$ in absolute value with the largest coefficient being 0.5. Therefore an unbiased estimate of the largest coefficient should be around 0.5. However, as shown in Figure 4, the vast majority of the largest coefficients of the Bayesian Lasso are smaller than 0.2. This is consistent with the findings that “Lasso has had to choose between including too many variables or over shrinking the coefficients” (Radchenko and James, 2008). In fact, it has been shown that the Lasso does not have the oracle property if there is strong linear dependence among the covariates (Meinshausen and Bühlmann, 2006; Zou, 2006; Zhao and Yu, 2006), which is the case for multiple loci mapping using dense genetic markers. This explains why the Bayesian Lasso has limited performance in our study with dense genetic markers, but adequate performance when the genetic markers are relatively sparse (Yi and Xu, 2008): the linear dependence among the covariates is not strong for sparsely located markers. Small coefficients are also observed for the Bayesian t, although they are relatively bigger than the Bayesian Lasso. Such small coefficients, in addition to producing a biased model, also make it difficult to determine a coefficient cutoff in order to obtain the final model.

[Figure 4 about here.]

4.7 Computational efficiency

We conclude this section by carrying out a comparison of the computational efficiencies of different methods. The computational time certainly depends on the number of permutations/iterations used. Here we use 10,000 permutations for marginal regression, forward regression, and forward-backward regression. For all the Bayesian methods, we run 20,000
iterations, including the 10,000 burn-ins. Note that while all the other methods are implemented using customized C code, both the marginal regression and forward regression are implemented by R. However, the statistics in both methods are the $r^2$ between the trait and all the markers, which can be calculated efficiently in R. Thus we do not expect customized C code to be much faster, unless some computational tricks are used. As shown in Table 2, IAL is the most efficient method and the second most efficient method is CMSA. All of the computation was done using a Dual Xenon 2.0 Ghz Quadcore server.

Table 2 about here.

5. Discussion

In this paper, we have proposed two variable selection methods, namely the Bayesian Adaptive Lasso (BAL) and the Iterative Adaptive Lasso (IAL). These two methods extend the adaptive Lasso in the sense that they do not require any informative initial estimates of the regression coefficients. The BAL is implemented by MCMC. Through extensive simulations, we observe the BAL has apparently better variable selection performance than the Bayesian Lasso, slightly better performance than the Bayesian t, and comparable performance as the CMSA. The IAL, which is an ECM algorithm, aims at finding the mode of the posterior distribution. The IAL is computationally very efficient and it penalizes most coefficients to zero, therefore alleviating the cutoff selection problem in follow-up studies. The IAL, similar to step-wise methods, tends not to select linked false discoveries. When we compare the number of true discoveries against the total number of false discoveries, the IAL has clear advantages over any Bayesian method. These characteristics make the IAL more suitable for larger scale calculations such as gene expression quantitative trait (eQTL) studies where tens of thousands of genes expression traits are studied (Kendzioriski et al., 2006).

There is room to improve our methods. One important question left open in our methods as
well as in the Bayesian Lasso, Bayesian t, and CMSA is how to select a formal threshold given
the coefficient/posterior probability estimate of each covariate. In fact, threshold selection is
also a problem for marginal regression and step-wise regression: as the sample size increases,
one may want to use smaller permutation p-value to filter out more false positives while
maintaining reasonable power. We are actively exploring methods for threshold selection. With
efficient IAL, cross-validation is an option. Combining different methods is another option.
For example, the IAL results can be used to first exclude those chromosomes with only zero
coefficients, and then the results of the BAL in the remaining chromosomes can be used as
initial values for another round of the IAL.

In the current implementation, we handle missing genotype data by imputing it first (using
Viterbi algorithm implemented in R/qtl) and then take the imputed values as known. A more
sophisticated approach for the BAL is to take the genotype data as unknown and sample them
within MCMC; and for the IAL, we can summarize its results across multiple imputations
(Sen and Churchill, 2001). However, these sophisticated approaches are computationally
more intensive and are mainly designed for relatively sparse marker maps. The current
high-density SNP arrays often have high confidence genotype call rate larger than 98%
(Rabbee and Speed, 2006). Imputation methods are also an active research topic. Haplotype
information from related or unrelated individuals can be used to obtain accurate genotype
imputation (Marchini et al., 2007). Therefore simply imputing the genotype data and then
taking it as known may be sufficient for many studies using high-density SNP arrays, although
careful examination of missing data patterns is always important.

Our methodological development in this paper focuses on additive linear models without
epistatic interaction. This is not because we do not appreciate the importance of epistatic
interactions, but because we feel variable selection in high dimensional settings is still a
difficult problem, even for the simple additive model. In addition, our method can be
extended for epistatic interaction mapping with minor modifications. One straightforward extension is to code all the pair-wise interactions as new covariates and include them in the model, similar to the approaches used by Zhang and Xu (2005) and Yi et al. (2007). This approach may not be computationally feasible for the studies with a large number of markers. Thus prioritizing interactions by the significance of main effects or biological knowledge (e.g., genetic interaction or protein-protein interaction) may be needed. How to penalize the interaction term also warrants further study. Grouping the interaction terms and the corresponding main effects together and applying group penalties (Yuan and Lin, 2006) may be a better approach. Similar group penalty arguments can be applied if we study the additive and dominant effect of one marker simultaneously along with the interactions with other covariates such as sex (Wu and Lange, 2008).

In this paper, we only report the results in mouse F2 populations with a few thousands of markers. Recently, large-scale genome-wide association studies (GWAS) with millions of markers have been proved to be a valuable approach to dissect the genetic basis of complex traits (McCarthy et al., 2008). One of our future studies is to evaluate our methods in GWAS.

In addition to the two new methods, we emphasize a few more points. First, step-wise methods such as forward regression or forward-backward regression are conceptually simple, and have good variable selection performance as shown in our simulations, especially for unlinked QTL. In previous studies, the performance of these step-wise methods has not been extensively evaluated or compared with methods that simultaneously select multiple loci.

Secondly, it is important to evaluate the performance of multiple loci mapping methods by a large number of simulations using realistic genotype data. One method may have better performance than another in one or two simulations by chance. Therefore a large number of simulations are better suited to compare the performance of two methods. In fact, even with realistic genotype data and a large number of simulations, the performance of different
methods may still depend on the QTL locations and effect sizes. For example, in simulation study I, we observe that forward-backward regression has obviously better performance than forward regression for QTL linked in repulsion with relatively smaller environmental variance. However, this advantage disappears in simulation study II when we average results across different randomly selected QTL with the same effect sizes. Therefore in practice it may be helpful to run a few different methods and compare their results.

ACKNOWLEDGEMENTS

The authors want to thanks Dr. Jun Liu for valuable suggestions and thanks Dr. Aldons J. Lusis for providing the genotype data. WS’s research is supported in part by grant from National Institute Environmental Health Sciences (5 P30 ES10126-07). FZ’s research is supported by NIH grant GM074175.

SUPPLEMENTARY MATERIALS

Web Appendices and Figures are available under the Paper Information link at the Biometrics website http://www.biometrics.tibs.org.

REFERENCES


Figure 1. The locations and effect sizes of the QTL in simulation study I. (a) Situations 1 or 2. (b) Situations 3 or 4. The genetic distances/$r^2$ between two QTL from chromosome 6, 7, 10, 15, and 19 are 20cM/0.46, 29cM/0.28, 40cM/0.26, 42cM/0.20, and 37cM/0.22, respectively, where $r^2$ denotes the correlation square. (c) Situations 5 or 6. The genetic distances/$r^2$ between two QTL from chromosome 7, 8, 14, 18, and 19 are 29cM/0.33, 31cM/0.34, 40cM/0.22, 27cM/0.32, and 34cM/0.36, respectively.
Figure 2. Comparison of the number of true discoveries vs. the total number of false discoveries in simulation study I: simulated genotype data and fixed QTL.
Figure 3. Comparison of the number of true discoveries vs. the number of unlinked false discoveries in simulation study I: simulated genotype data and fixed QTL.
Figure 4. Distributions of the largest coefficients (in absolute value) from each of the 100 simulations in simulation study I, situation 2 (unlinked QTL and $\sigma^2_e = 0.5$).
Table 1
The proportion of trait variance explained by the genetic effects. Mean and standard deviation (sd) are calculated across the 100 simulations in each situation.

<table>
<thead>
<tr>
<th>simulation situation</th>
<th>mean</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>study I</td>
<td></td>
<td>0.35</td>
<td>0.53</td>
<td>0.39</td>
<td>0.56</td>
<td>0.19</td>
<td>0.32</td>
</tr>
<tr>
<td>sd</td>
<td></td>
<td>0.03</td>
<td>0.04</td>
<td>0.03</td>
<td>0.04</td>
<td>0.01</td>
<td>0.02</td>
</tr>
<tr>
<td>simulation mean</td>
<td></td>
<td>0.33</td>
<td>0.50</td>
<td>0.44</td>
<td>0.62</td>
<td>0.17</td>
<td>0.30</td>
</tr>
<tr>
<td>study II</td>
<td></td>
<td>0.03</td>
<td>0.04</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.07</td>
</tr>
</tbody>
</table>
Table 2

Computation time (in seconds) for genome scan of one trait. Recall that in simulation study I, we have 1200 markers in 360 mice, while in simulation study II, we have 1025 markers in 266 mice.

<table>
<thead>
<tr>
<th>Simu. Study</th>
<th>Marginal</th>
<th>Forward</th>
<th>Forward-Backward</th>
<th>CMSA</th>
<th>Bayesian Lasso</th>
<th>BAL</th>
<th>IAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>90</td>
<td>434</td>
<td>452</td>
<td>55</td>
<td>225</td>
<td>249</td>
<td>362</td>
</tr>
<tr>
<td>II</td>
<td>82</td>
<td>320</td>
<td>347</td>
<td>42</td>
<td>164</td>
<td>172</td>
<td>255</td>
</tr>
</tbody>
</table>