FLEXIBLE HIERARCHICAL BAYESIAN MODELING EXTENSIONS TO IMPROVE WHOLE GENOME PREDICTION AND GENOME WIDE ASSOCIATION ANALYSES

By

Chunyu Chen

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Animal Science — Doctor of Philosophy

2017

ABSTRACT

FLEXIBLE HIERARCHICAL BAYESIAN MODELING EXTENSIONS TO IMPROVE WHOLE GENOME PREDICTION AND GENOME WIDE ASSOCIATION ANALYSES

By

Chunyu Chen

Whole genome prediction (WGP) has been widely implemented in animal and plant breeding for genomic selection of economically important traits, having already accelerated genetic progress for economically important traits in some species especially dairy cattle. Genome wide association (GWA) analysis is used for screening genomic regions that may include important candidate genes segregating for the trait of interest and is being increasingly integrated with WGP analysis. Both WGP and GWA typically represent $m \gg n$ problems as defined by a large number of single nucleotide polymorphism (SNP) markers (m) and comparably much smaller number of individuals (n). Two broad types of parametric models are typically considered for these analyses: traditional best linear unbiased prediction approaches based on SNP marker effects being normally distributed and Bayesian WGP models that allow more flexible specifications for SNP marker effects based on either heavy-tailed or variable selection specifications. Bayesian WGP models can achieve higher prediction accuracies than traditional approaches in many applications if properly tuned; however, their implementation can be computationally challenging. My dissertation was aimed to address some of these emerging issues in Bayesian WGP models as well as providing software tools for real data applications.

In Chapter 2, I developed an expectation maximization (EM) algorithm as a fast alternative to traditional Markov Chain Monte Carlo (MCMC) for Bayesian WGP models. I proposed EM implementations for two models, heavy-tailed BayesA and stochastic search and variable selection (SSVS) adapting the EM algorithm for maximum a posterior (MAP) inference of SNP

effects and adapting REML like strategies to estimate key hyperparameters. Using a comprehensive simulation study and real data analysis, I found that these empirical Bayes approaches can be quite sensitive to starting values for SNP effects. However, using a deterministic annealing variant of EM, I obtained hyperparameter estimates and prediction accuracies comparable to their MCMC counterparts. In Chapter 3, I further assessed the possibility using two Bayesian WGP models BayesA and SSVS for GWA studies. I also included a popular GWA analysis (EMMAX) based on the utilization of the linear mixed model. In addition to basing inferences on traditional single SNP tests and fixed genomic window tests, I assessed the merit of tests involving adaptively determined windows based on clustering genome into blocks based on linkage disequilibrium. I found that SSVS and BayesA under MCMC and adaptive window tests led to best receiver operating curve (ROC) properties. In Chapter 4, I extended SSVS to single step SSVS to incorporate phenotypes of non-genotyped individuals and compared its performance with corresponding models ignoring these genotypes for both WGP and GWA. I found single step SSVS to be a promising for WGP and GWA, particularly for genetic architectures characterized by a few genes with large effects. In Chapter 5, I combined much of the developments in Chapter 2 to Chapter 4 and beyond in a unified framework as an open source R package `BATools` to implement several different Bayesian models for WGP and GWA.

To my wife Gefan Li

ACKNOWLEDGMENTS

It is a great experience spending five years to explore and finally know just a tip of animal breeding and genetics. This dissertation research could not have been done without the help from my advisor, committee members, lab mates and the support from families and friends. I would like to sincerely thank everyone who helped and supported me through this incredible journey. Firstly, I would like to thank my Ph.D. advisor, Professor Robert J. Tempelman, for the guidance and patience during the past five years. Rob is someone you will love once you talk to him. He is always energetic, passionate and thoughtful. He is a good mentor, professor, and friend. Without his advice and suggestion, I would not have completed this work.

I also want to thank my committee members Dr. Juan Steibel, Dr. Yuehua Cui, Dr. Nora Bello and Dr. Qing Lu for their insightful suggestions and inspiring thoughts. I appreciate your help and suggestions. Special thanks to Dr. Juan Steibel, who inspired me a lot on writing reproducible code through his class and talks and made the swine data available for my research, and Dr. Yuehua Cui, for his introduction to statistical genetics.

Also, thanks to my former and current colleagues in the quantitative genetics lab including Jose-Luis Gualdron Duarte, Yeni Bernal Rubio, Wenzhao Yang, Heng Wang, Youngfang Lu, Lei Zhou, Pablo Reeb, Kaitlyn Daza, Yasir Nawaz, Deborah Velez, Ryan Corbett and Scott Funkhouser for the meaningful discussions and friendship. Special thanks to Jose-Luis Gualdron Duarte and Yeni Bernal Rubio for their assistance in preparing the swine data.

Finally, I am indebted to my wife and parents and I'd like to thank them for their love and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter1 Introduction

Whole genome prediction (WGP) using dense single nucleotide polymorphism (SNP) marker panels has been increasingly implemented in animal and plant breeding for genetic improvement of economically important traits. Currently, SNP marker panels with ~50,000 SNP markers are widely used for most livestock species, and high-density panels with ~770,000 SNP markers are also available (Meuwissen *et al.* 2016). In the 1000 bull genome project (http://www.1000bullgenomes.com/), 28.3 million variants have been obtained on 238 cattle using whole-genome sequence technology (Daetwyler *et al.* 2014) with the most recent updated number of sequenced cattle now at 1147 (http://www.canadacow.ca/). Therefore, WGP is a "big data" research and application area characterized albeit by a relatively small number of observations (*n*) compared to the number of predictors or SNP markers (*m*).

The seminal idea of using WGP for genomic selection (GS) of livestock was developed 16 years ago by Meuwissen *et al.* (2001) who exposited the use of best linear unbiased prediction (BLUP) and various Bayesian extensions to include information on SNP marker panels, even before such panels were developed for livestock! Genetic gain using genomic selection has doubled in dairy cattle traits compared to the period before the adoption of SNP marker information for genetic evaluations in 2009 (García-Ruiz *et al.* 2016) such that genomic selection is now considered to be a mature technology (Misztal 2016b). Nevertheless, improved methodologies and software tools for WGP still require further development.

Two broad categories of models for WGP are genomic BLUP (GBLUP) models and Bayesian models (Meuwissen *et al.* 2001; de Los Campos *et al.* 2013; Gianola 2013), both considered to be critical components of hierarchical linear or multilevel modeling. While GBLUP or Bayesian ridge regression (BRR) (Hoerl and Kennard 1970) is equivalent to

specifying a Gaussian prior on the SNP marker effects, other Bayesian models are often specified with more flexible priors on the SNP marker effect to provide differential shrinkage effects that may be appropriate for various types of genetic architecture; i.e., whether or not a trait is characterized by few (oligogenic) or many (polygenic) loci. These priors include heavy-tailed specifications such as a scaled Student-*t* (Meuwissen et al., 2001), variable selection specifications such as SSVS or stochastic search and variable selection (George and McCulloch 1993) or hybrids thereof such as a mixture of point mass at zero and scaled-*t* (BayesB) as originally proposed by Meuwissen *et al.* (2001). With greater flexibility in priors, hierarchical Bayesian models have been shown to provide higher WGP prediction accuracies than GBLUP/BRR in many applications (de Los Campos *et al.* 2013).

Unfortunately, implementation of hierarchical Bayesian models with flexible priors typically requires intensive computing demands as the posterior inference is typically based on simulation-based Markov chain Monte Carlo (MCMC) techniques. Conversely, GBLUP is relatively fast, particularly with small n (i.e. n x n matrices are easily inverted) in part because of a recent equivalent model realization that parameterizes genetic effects in terms of additive genetic effects rather than SNP marker effects (Stranden and Garrick 2009). Over the past few years, to reduce the computing time, several Expectation–Maximization (EM) algorithms have been developed to partly address computational limitations in hierarchical Bayesian models with flexible priors (Meuwissen *et al.* 2009; Shepherd *et al.* 2010; Karkkainen and Sillanpaa 2012; Sun *et al.* 2012). These EM implementations sometimes achieve comparable or slightly lower WGP accuracies to their MCMC counterparts. Typically, the hyperparameters in these EM algorithms are often arbitrarily specified (Karkkainen and Sillanpaa 2012; Sun *et al.* 2012) or sometimes determined by heritability-based rules (de Los Campos *et al.* 2013) that have been shown to be suboptimal compared to formally allowing for their uncertainty (Lehermeier *et al.* 2013; Yang *et al.* 2015b).

Furthermore, there does not seem to be an appreciation of the potential influence of starting values for the SNP marker effects in those EM based approaches with most implementations choosing zero as starting values. Hence, hyperparameter tuning/estimation and starting values for EM approaches require further investigation as to their effect on these analytical approximations.

Genome-wide association (GWA) analysis is also another important tool to help pinpoint the regions contain causal variants or quantitative trait loci (QTL) for complex traits. With the availability of high density SNP marker panels, the very first GWA result was reported in 2005 (Klein *et al.* 2005) followed by the first large scale GWA study by Wellcome Trust Case Control Consortium (2007). The early days of GWA studies were based on serial simple linear regression models on SNP markers, one at a time, without accounting for population structure and relatedness. Ignoring these features has been demonstrated to result in spurious GWA inferences (Martin and Eskin 2016). To account for population structure, linear mixed model (LMM) specifying all other marker effects as random except for the one of inferential interest as fixed have been proposed (Kang *et al.* 2008). Since then various computationally efficient enhancements to this LMM approach have been developed (Kang *et al.* 2010; Lippert *et al.* 2011; Zhou and Stephens 2012; Gualdron Duarte *et al.* 2014). The variance components of these LMM are typically estimated using restricted maximum likelihood (REML).

Hierarchical Bayesian models in WGP fit all SNP markers as random effects simultaneously (Meuwissen *et al.* 2001) and automatically accounts for the population structure just as LMM with random effects. However, random effects estimation using Gaussian prior specifications (equivalent to GBLUP/BRR) tends to overly shrink all marker effects to zero, particularly for higher marker densities (Hayes 2013) and has been deemed to be too conservative for GWA (Gualdron Duarte *et al.* 2014). Therefore, priors with less shrinkage to larger marker effects,

such as BayesA and SSVS, should be considered for such applications as they tend to provide far less shrinkage to larger marker effects than a Gaussian in WGP applications. GWA studies rely on SNPs to be correlated or in linkage disequilibrium (LD) with QTLs. In fact, many SNPs are likely to be in LD with a single QTL (Goddard *et al.* 2016), and single marker tests suffer from multicollinearity problems or low statistical power or both (Fernando *et al.* 2017). For this reason, GWA studies based on joint tests on all marker within a genomic window/region should be considered. Currently, however, window lengths tend to be arbitrarily specified (Schmid and Yang 2008; Moser *et al.* 2015), and in livestock species and crops, LD may extend for a long distance (Goddard *et al.* 2016). Hence, those window selection procedures may separate SNP markers that should conceptually be grouped in the same window because of high LD between them. Therefore, additional efforts are required to partition SNPs into windows with less arbitrary boundaries.

In large GS programs for animal and plant breeding, an increasingly important problem is that many if not most individuals to be genetically evaluated do not have genotype information. Traditionally, genomic evaluation uses deregressed breeding values from pedigree based BLUP to remove the contribution of relatives that are not related to the study; and then fit WGP models for genotyped animals in a 'two-step' model (VanRaden 2008; Hayes *et al.* 2009). A single-step GBLUP (ssGBLUP) approach that combines phenotypes on genotyped and non-genotyped animals with pedigree information in one regression model (Aguilar *et al.* 2010) has become popular for many livestock GS programs (Legarra *et al.* 2014). Because of extra phenotypic information that ssGBLUP has combined, many studies have found this procedure to have higher prediction accuracy than Bayesian models without such information (Lourenco *et al.* 2013; Legarra *et al.* 2014; Vallejo *et al.* 2016). The ssGBLUP models have also been implemented for

GWA. However, these GWA assessments were not based on formal measures of statistical significance (Wang *et al.* 2012; Zhang *et al.* 2016); therefore, such measures, e.g. *P*-value, needs to be developed. Recently, Fernando *et al.* (2014) proposed a framework to implement the single-step approach in hierarchical Bayesian models that combine information on both genotyped and non-genotyped individuals. Although studies have shown such models have higher WGP accuracies than ssGBLUP in beef cattle for traits controlled by large SNP marker effects (Lee *et al.* 2017), the GWA performance of Bayesian models that combine non-genotyped animals have not been comprehensively evaluated, as well as WGP accuracies in other species.

Currently, several open source software packages are available for WGP or GWA or both, and most of them are designed for specific models (Endelman 2011; Zhou and Stephens 2012). The popular `BGLR` R package (Perez and de los Campos 2014) includes a collection of models designed for WGP, but it does not focus on GWA features nor does it yet support Bayesian approaches that incorporate information on non-genotyped animals. It is crucial to have an open source R package to include both LMM and Bayesian models for both WGP and GWA that performs window based GWA and combines genotype, phenotype and pedigree information of both genotype and non-genotyped individuals.

With this in mind, there are four overall objectives in this dissertation to improve the computational efficiency and accuracy in both WGP and GWA. The first objective is to help improve the computational efficiency for Bayesian WGP models using EM algorithm and assess their ability to estimate hyperparameters and the influence of starting values on WGP accuracies (Chapter 2). The second objective is to develop a window based approach for traditional LMM and two Bayesian models: BayesA and SSVS; to examine the potential benefits of using

Bayesian models relative to classical LMM for GWA under a wide range of simulated architectures; to assess whether the choice of different fixed genomic window sizes versus window sizes inferred based on LD clustering, could impact GWA performance and to evaluate the relative merit of EM approaches to MCMC approaches for BayesA and SSVS (Chapter 3). The third objective is to extend hierarchical Bayesian model and traditional LMM to incorporate information on non-genotyped animals for both single SNP and window based GWA to provide formal statistical significance assessment and to compare these approaches for both WGP and GWA (Chapter 4). A capstone objective is to provide all the models/algorithm in previous Chapters as an efficient and accessible R package (Chapter 5). Finally, I summarize all the findings throughout the dissertation as well as provide ideas for future research (Chapter 6).

Chapter2 An Integrated Approach to Empirical Bayesian Whole Genome Prediction Modeling

2.1 Abstract

Computational efficiency is an increasing concern for whole genome prediction (WGP) based on denser genetic marker panels such that algorithms other than Markov Chain Monte Carlo (MCMC) warrant greater consideration, particularly for hierarchical models that flexibly confer either heavy-tailed (e.g., BayesA) or stochastic search and variable selection (SSVS) instead of Gaussian specifications on marker effect distributions. The expectation maximization (EM) algorithm is one attractive alternative; however, recently proposed hierarchical model implementations of EM have not addressed formal estimation of underlying hyperparameters even though their specifications are known to impact WGP accuracy. Furthermore, EM can be sensitive to starting values. I develop and explore the properties of an empirical Bayes strategy by conditioning EM implementations of BayesA or SSVS WGP models on marginal modal estimation of variance components and other key hyperparameters. These empirical Bayes implementations are compared against their MCMC counterparts for estimation of hyperparameters and WGP accuracy, both within the context of a simulation study and application to a loblolly pine dataset. In all cases, starting values were deemed to be important for EM-based estimates. Starting values based on MCMC posterior means were preferable whereas those based on setting all marker effects equal to zero generally led to inferior performance. Nevertheless, a recently proposed regularization procedure was useful in alleviating the impact of starting values in the EM implementation of the SSVS model, as was modifying the expectation step in the BayesA model to be based on relative variances rather than on relative precisions.

2.2 Introduction

Recent developments in genotyping technology have made dense single nucleotide polymorphism (SNP) genotype marker panels available for many livestock species. Typically, some of these SNP chips have up to hundreds of thousands or more markers. As these numbers continue to increase with emerging sequencing technologies (Daetwyler *et al.* 2014), it is important to develop statistically and computationally efficient methods that best use these genotypes to predict genomic estimated breeding values (GEBV) on economically important traits in whole genome prediction (WGP) models. These models are based on the premise that for a sufficient number of SNPs, at least some of them should be in close linkage disequilibrium (LD) with quantitative trait loci (QTL) for economically important traits (Meuwissen *et al.* 2001).

WGP typically represents a $m \gg n$ problem, whereby the number (m) of markers used for prediction is much larger than the number (n) of animals having phenotypes. Two broad parametric categories for WGP models are genomic or ridge regression best linear unbiased prediction (RRBLUP) and various hierarchical model extensions of RRBLUP, often satirically referred to as "Bayesian alphabet" models (Gianola 2013). RRBLUP is based on classical linear mixed model analyses whereby a common variance component is specified for each random SNP effect. Variance components can be readily estimated using Restricted Maximum Likelihood (REML) followed by BLUP of SNP effects conditional on these REML estimates. This two-stage approach has been characterized as empirical Bayes (EB) (Casella 1985) and so I might refer to corresponding genomic predictions as empirical RRBLUP (e-RRBLUP) . On the other hand, fully Bayesian inference in hierarchical WGP models is typically conducted using Markov chain Monte Carlo (MCMC) techniques. In a seminal paper, Meuwissen *et al.* (2001)

proposed BayesA which hierarchically extends RRBLUP by specifying every SNP effect as scaled Student *t* distributed. Since then, other hierarchical variations on heavy-tailed or variable selection specifications on the SNP effects have been proposed. Such extensions often lead to higher WGP accuracies compared to RRBLUP analyses, particularly when *n* is large relative to the number of QTL and *n*/*m* is not too small (Wimmer *et al.* 2013).

MCMC analyses are computationally expensive, especially for WGP models with large *m*. Expectation–Maximization (EM) algorithms have been developed to partly address computational limitations in hierarchical Bayes WGP (e.g., Meuwissen *et al.* 2009; Hayashi and Iwata 2010; Shepherd *et al.* 2010; Sun *et al.* 2012). These EM implementations have often been shown to lead to WGP accuracies comparable to their MCMC based counterparts but at a fraction of the computational cost.

A typically neglected issue in hierarchical WGP modeling is proper tuning/inference of hyperparameters as subsequent estimates on SNP effects have been observed to be sensitive to such specifications (Gianola *et al.* 2009). These hyperparameters are often arbitrarily specified or sometimes based on heritability-based rules (de Los Campos *et al.* 2013) that have been shown to be suboptimal (Lehermeier *et al.* 2013). Although formal Bayesian inference on hyperparameters using MCMC is possible (Yi and Xu 2008; de Los Campos *et al.* 2013; Yang *et al.* 2015b), there has been far less discussion on how to estimate hyperparameters in conjunction with EM-based implementations of hierarchical WGP models; in fact, it has been deemed to be nearly impossible (Karkkainen and Sillanpaa 2012). Furthermore, it has been implied that starting values are not important in these implementations with zero being a particularly popular starting value choice for SNP effects (e.g., Meuwissen *et al.* 2009; Shepherd *et al.* 2010; Karkkainen and Sillanpaa 2012). However, Gianola (2013) has warned that joint posterior

modal estimates may iteratively get trapped at local modes such that the specification of different starting values for SNP effects, never mind hyperparameter specifications, could have different implications for WGP accuracy. To partially address that concern in a variable selection model known as stochastic search and variable selection (SSVS) first introduced by George and McCulloch (1993), I investigate strategies proposed by Rockova and George (2014) to alleviate the influence of starting values for determining high posterior probability modes in an EM-based implementation of SSVS.

Our objectives were then to demonstrate an empirical Bayes (EB) strategy to estimate key hyperparameters as well as to investigate the effects of different starting values on SNP effect estimates and genomic merit in two widely used hierarchical WGP models, BayesA and SSVS. In Section 2, I outline this EB strategy for both BayesA and SSVS and describe a simulation study used to assess the effect of starting values on accuracy of genomic merit prediction for both models, using MCMC as a positive control method. I also demonstrate application of these same procedures to a loblolly pine dataset (Resende *et al.* 2012) illustrating the use of the regularization procedures proposed by Rockova and George (2014) for SSVS. Results are described in Section 3 with a concluding discussion provided in Section 4.

2.3 Materials and methods

2.3.1 The first stage linear WGP model

The first stage of a WGP linear mixed model is typically specified as follows:

$$y_i = \mathbf{x}_i'\boldsymbol{\beta} + \mathbf{z}_i'\mathbf{g} + e_i,\ i = 1, 2, ..., n \qquad [2.1]$$

where y_i denotes the phenotype on individual *i* connected to fixed effects $\boldsymbol{\beta}$ via a known incidence row vector $\mathbf{x}_i'$ and connected to SNP effects $\mathbf{g} = \{g_j\}_{j=1}^m$ via known genotypes

$\mathbf{z}_i^{'} = \begin{bmatrix} z_{i1} & z_{i2} & z_{i3} & \dots & z_{im} \end{bmatrix}$ on *m* SNP markers. Furthermore, I specify $e_i \overset{iid}{\sim} N\left(0, \sigma_e^2\right)$. Now although the SNP covariates are typically provided with values of z_{ij}^* = 0, 1, or 2 (i.e., number of copies of a reference allele at each SNP), they are often recoded in a number of different ways using, for example, just centering, i.e., $z_{ij} = z_{ij}^* - 2\hat{p}_j$ or centering and scaling, i.e., $z_{ij} = \dfrac{z_{ij}^* - 2\hat{p}_j}{\sqrt{2\hat{p}_j\left(1-\hat{p}_j\right)}}$ (de Los Campos *et al.* 2013), where $\hat{p}_j = \dfrac{1}{2n}\sum_{i=1}^{n} z_{ij}^*$ denotes the estimated frequency of the reference allele for SNP *j*=1,2,…,*m.* Recoding genotypes in this manner has been demonstrated to improve algorithmic stability (Stranden and Christensen 2011).

2.3.2 BayesA EM

In BayesA, the following hierarchical priors are typically specified:

$$g_j \sim N\left(0, \sigma_g^2 \tau_j\right) \qquad [2.2]$$

having density $p\left(g_j \mid \sigma_g^2, \tau_j\right)$ and

$$\tau_j \sim \chi^{-2}(\nu_g, \nu_g) \qquad [2.3]$$

having density $p\left(\tau_j \mid \nu_g\right)$ and used to denote a scaled inverted chi-square distribution with degrees of freedom and scale parameters both being $\nu_g\left(\nu_g > 0\right)$ such that τ_j =1 then defines a typical value falling between the prior mean $\left(\dfrac{\nu_g}{\nu_g - 2}; \nu_g > 2\right)$ and prior mode $\left(\dfrac{\nu_g}{\nu_g + 2}\right)$. This parameterization slightly differs from, but is marginally equivalent to, that provided in the seminal paper by Meuwissen *et al.* (2001) who directly specify prior distributions on

$\sigma^2_{g_j} \sim \sigma^2_g \tau_j$. That is, marginally, $g_j \,|\, \nu_g, \sigma^2_g \sim \int_{\tau_j} N\left(0, \sigma^2_g \tau_j\right) \chi^{-2}(\nu_g, \nu_g) d\tau_j = t_{\nu_g}(0, \sigma^2_g)$, i.e. a scaled Student *t* distribution with degrees of freedom ν_g and scale parameter σ^2_g.

Given these specifications, the joint posterior density for BayesA can be derived as follows:

$$\begin{aligned} & p\left(\boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\tau}, \sigma^2_e, \sigma^2_g, \nu_g \,|\, \mathbf{y}\right) \\ & \propto \left(\prod_{i=1}^{n} p\left(y_i | \boldsymbol{\beta}, \mathbf{g}, \sigma^2_e\right)\right)\left(\prod_{j=1}^{m} p\left(g_j | \sigma^2_g, \tau_j\right) p\left(\tau_j \,|\, \nu_g\right)\right) p(\boldsymbol{\beta})\, p\left(\sigma^2_g\right) p\left(\sigma^2_e\right) p\left(\nu_g\right), \end{aligned} \qquad [2.4]$$

where $\mathbf{y} = \{y_i\}_{i=1}^n$ and $\boldsymbol{\tau} = \left\{\tau_j\right\}_{j=1}^m$. Also Equation [2.4] is based on the product of arbitrarily and independently specified priors $p(\boldsymbol{\beta}), p\left(\sigma^2_g\right), p\left(\sigma^2_e\right),$ and $p\left(\nu_g\right)$ on $\boldsymbol{\beta}, \sigma^2_g, \sigma^2_e$, and ν_g, respectively. I will assume $p(\boldsymbol{\beta}) \propto 1$ in this paper as $p(\boldsymbol{\beta})$ is typically diffuse, although extensions to more informative specifications should be obvious.

The EM algorithm is based on computing the expectation of a log likelihood and/or a log joint posterior density with respect to augmented variables (E-step) followed by maximization of this expectation with respect to the remaining unknown parameters (M-step). Let's momentarily assume that each of $p\left(\sigma^2_g\right), p\left(\sigma^2_e\right),$ and $p\left(\nu_g\right)$ are point masses on known values for σ^2_g, σ^2_e, and ν_g, respectively, such that Equation [2.4] can be re-expressed as $p\left(\boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\tau} \,|\, \sigma^2_e, \sigma^2_g, \nu_g, \mathbf{y}\right)$ to reflect this conditioning. Taking its logarithm, the corresponding "augmented" conditional log density (L_A) for $\boldsymbol{\beta}$, $\mathbf{g}$ and the augmented variables $\boldsymbol{\tau}$, recognizing that $\log p(\boldsymbol{\beta}) = \text{constant}$, is as follows:

$$\begin{aligned} L_A &= \log\left(p\left(\boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\tau} \,|\, \sigma^2_e, \sigma^2_g, \nu_g, \mathbf{y}\right)\right) \\ &= \sum_{i=1}^{n} \log p\left(y_i | \boldsymbol{\beta}, \mathbf{g}, \sigma^2_e\right) + \sum_{j=1}^{m}\left(\log p\left(g_j | \sigma^2_g, \tau_j\right) + \log p\left(\tau_j \,|\, \nu_g\right)\right) + \text{constant}. \end{aligned} \qquad [2.5]$$

Following Sun *et al.* (2012), the E-step for $\boldsymbol{\tau}$ involves evaluating the expectation of Equation [2.5] with respect to $\boldsymbol{\tau}$. Taking terms in Equation [2.5] that only involve $\boldsymbol{\tau}$, by drawing on the corresponding components that derive from the logarithms of Equations [2.2] and [2.3], require evaluations of $\underset{\tau_j|.}{\mathrm{E}}\left(\frac{1}{\tau_j}\right)$ and $\underset{\tau_j|.}{\mathrm{E}}\left(\log\left(\tau_j\right)\right)$. Here $\tau_j|.$ denotes the expectation is conditional on all other terms and **y**, i.e.,

$$\begin{aligned}&\underset{\tau_j|.}{\mathrm{E}}\left(\sum_{j=1}^{m}\left(\log\left(p\left(g_j|\sigma_g^2,\tau_j\right)+\log p\left(\tau_j|\nu_g\right)\right)\right)\right)\\&=\sum_{j=1}^{m}\left(-\frac{1}{2}\frac{g_j^2}{\sigma_g^2}\underset{\tau_j|.}{\mathrm{E}}\left(\frac{1}{\tau_j}\right)-\left(\frac{\nu_g}{2}+1\right)\underset{\tau_j|.}{\mathrm{E}}\left(\log\left(\tau_j\right)\right)-\frac{\nu_g}{2}\underset{\tau_j|.}{\mathrm{E}}\left(\frac{1}{\tau_j}\right)\right)+\text{constant}.\end{aligned} \qquad [2.6]$$

Now, as previously provided by Sun *et al.* (2012) ,

$$\left(\hat{\tau}_j\right)^{-1}=\underset{\tau_j|.}{\mathrm{E}}\left(\frac{1}{\tau_j}\right)=\frac{\nu_g+1}{\frac{\hat{g}_j^2}{\sigma_g^2}+\nu_g} \qquad [2.7]$$

where $\hat{g}_j$ is g_j evaluated at the M-step (see later) whereas

$$\underset{\tau_j|.}{\mathrm{E}}\left(\log\left(\tau_j\right)\right)=\log\left(0.5\left(\frac{\hat{g}_j^2}{\sigma_g^2}+\nu_g\right)\right)-\Psi\left(\frac{\nu_g+1}{2}\right) \qquad [2.8]$$

where $\Psi(.)$ denotes the digamma function, i.e., $\Psi(x)=\frac{\partial\Gamma(x)}{\partial x}dx$ for $\Gamma(.)$ being the gamma function. Note that evaluating Equation [2.8] is only required if estimation of ν_g is desired.

Subsequently, the joint posterior modes, $\hat{\boldsymbol{\beta}}$ and $\hat{\mathbf{g}}$, respectively of $\boldsymbol{\beta}$ and $\mathbf{g}$ are evaluated in the maximization or M-step. This involves solving Henderson's mixed model equations (Sun *et al.* 2012), i.e.,

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z} \\ \mathbf{Z}'\mathbf{X} & \mathbf{Z}'\mathbf{Z}+\frac{\sigma_e^2}{\sigma_g^2}\hat{\mathbf{D}}^{-1} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\mathbf{g}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{Z}'\mathbf{y} \end{bmatrix} \quad [2.9]$$

where $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_n \end{bmatrix}'$, $\mathbf{Z} = \begin{bmatrix} \mathbf{z}_1 & \mathbf{z}_2 & \cdots & \mathbf{z}_n \end{bmatrix}'$ and $\hat{\mathbf{D}}^{-1} = diag\left\{\left(\hat{\tau}_j\right)^{-1}\right\}$.

Upon convergence of the E-steps and M-steps, one attains the joint posterior mode of **β** and **g** conditional on σ_g^2, σ_e^2, and ν_g .

As a deviation from this particular EM implementation, Karkkainen and Sillanpaa (2012) proposed that the E-step in a BayesA model be based on evaluating $\underset{\sigma_{g_j}^2|\cdot}{\mathrm{E}}\left(\sigma_{g_j}^2\right) = \underset{\tau_j|\cdot}{\mathrm{E}}\left(\sigma_g^2\tau_j\right)$ or, equivalently, of $\hat{\tau}_j^* = \underset{\tau_j|\cdot}{\mathrm{E}}\left(\tau_j\right) = \dfrac{\dfrac{\hat{g}_j^2}{\sigma_g^2}+\nu_g}{\nu_g-1}$ (i.e. relative variances) instead of $\underset{\tau_j|\cdot}{\mathrm{E}}\left(\dfrac{1}{\tau_j}\right)$ (i.e., relative precisions) as in Equation [2.7] even though the latter is implicitly required in a formal E-step for the EM algorithm as per Equation [2.6]. Note that $\hat{\tau}_j$ and $\hat{\tau}_j^*$ are subtly different, having the same numerators but different denominators. Typically, expectations of augmented variables in EM implementations are taken with respect to their functional forms in augmented log likelihoods or log joint posterior densities (i.e., using $\hat{\tau}_j$) whereas Karkkainen and Sillanpaa (2012) substitute $\hat{\tau}_j^*$ for $\hat{\tau}_j$ for the E-step in **D** of Equation [2.9]. This warranted further investigation on our part.

2.3.3 SSVS EM

I adapt the developments in the previous section by modifying the prior on τ_j to facilitate variable selection using the SSVS specification first introduced by George and McCulloch (1993) and recently revisited by Rockova and George (2014) i.e.

$$g_j \sim N\left(0, \sigma_g^2\tau_j + \frac{(1-\tau_j)\sigma_g^2}{c}\right) \quad [2.10]$$

with density $p\left(g_j|\sigma_g^2, \tau_j\right)$ and whereby $c >> 1$ such that Equation [9a] represents a mixture distribution on a "slab" component for effectively non-zero g_j ,characterized by variance σ_g^2, and a "spike" component with variance $\frac{\sigma_g^2}{c}$, the latter thereby absorbing negligible or near-zero g_j. Furthermore,

$$\tau_j \sim Bernoulli(\pi); \tau_j = 0,1 \quad [2.11]$$

is the Bernoulli distribution with density $p\left(\tau_j \mid \pi\right)$. The joint posterior density for SSVS can then be written as follows:

$$\begin{aligned} & p\left(\boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\tau}, \sigma_e^2, \sigma_g^2, \pi \mid \mathbf{y}\right) \\ & \propto \left(\prod_{i=1}^{n} p\left(y_i|\boldsymbol{\beta}, \mathbf{g}, \sigma_e^2\right)\right)\left(\prod_{j=1}^{m} p\left(g_j|\sigma_g^2, \tau_j\right) p\left(\tau_j \mid \pi\right)\right) p(\boldsymbol{\beta})\, p\left(\sigma_g^2\right) p\left(\sigma_e^2\right) p(\pi). \end{aligned} \quad [2.12]$$

Note that the components $\prod_{i=1}^{n} p\left(y_i|\boldsymbol{\beta}, \mathbf{g}, \sigma_e^2\right)$, $p(\boldsymbol{\beta})$, $p\left(\sigma_g^2\right)$, and $p\left(\sigma_e^2\right)$ are defined similarly as before with BayesA except with $p\left(g_j|\sigma_g^2, \tau_j\right)$ and $p\left(\tau_j \mid \pi\right)$ defined as in Equations [2.10] and [2.11], respectively, and $p(\pi)$, as the prior for π, substituted for $p\left(\nu_g\right)$. Let's momentarily assume that the key hyperparameters σ_g^2, σ_e^2, and π are specified to be known, similar to what I did earlier with BayesA. Then again, with $p(\boldsymbol{\beta}) \propto 1$, Equation [2.12] can be further condensed to reflect this conditioning:

$$p\left(\boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\tau}, | \sigma_e^2, \sigma_g^2, \pi, \mathbf{y}\right) \propto \prod_{i=1}^{n} p\left(y_i|\boldsymbol{\beta}, \mathbf{g}, \sigma_e^2\right) \prod_{j=1}^{m}\left(p\left(g_j|\sigma_g^2, \tau_j\right) p\left(\tau_j \mid \pi\right)\right) \quad [2.13]$$

Taking its logarithm, I denote the corresponding log augmented SSVS joint posterior density as L_S used for inferring β and g with the augmented variables again being $\boldsymbol{\tau}$, i.e.,

$$\begin{aligned} L_S &= \log\left(p\left(\boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\tau} \mid \sigma_e^2, \sigma_g^2, \pi, \mathbf{y}\right)\right) \\ &= \sum_{i=1}^{n} \log p\left(y_i | \boldsymbol{\beta}, \mathrm{g}, \sigma_e^2\right) + \sum_{j=1}^{m}\left(\log p\left(g_j | \sigma_g^2, \tau_j\right) + \log p\left(\tau_j \mid \pi\right)\right) \end{aligned} \qquad [2.14]$$

Taking only terms that involve $\boldsymbol{\tau}$ in Equation [2.14], based on the components contributed by Equations [2.10] and [2.11], the E-step requires an evaluation of $\underset{\tau_j|\cdot}{\mathrm{E}}\left(\tau_j\right)$, i.e.,

$$\begin{aligned} &\underset{\tau_j|\cdot}{\mathrm{E}}\left(\sum_{j=1}^{m} \log\left(p\left(g_j | \sigma_g^2, \tau_j\right) + \log p\left(\tau_j \mid \pi\right)\right)\right) \\ &= \sum_{j=1}^{m}\left(\underset{\tau_j|\cdot}{\mathrm{E}}\left(-\frac{1}{2} \frac{{g_j}^2}{\sigma_g^2\left(\tau_j + \frac{\left(1-\tau_j\right)}{c}\right)}\right) + \underset{\tau_j|\cdot}{\mathrm{E}}\left(\tau_j\right)\log(\pi) + \underset{\tau_j|\cdot}{\mathrm{E}}\left(1-\tau_j\right)\log(1-\pi)\right) + \text{constant} \\ &= \sum_{j=1}^{m} -\frac{1}{2}\sigma_g^{-2}{g_j}^2\left(\underset{\tau_j|\cdot}{\mathrm{E}}\left(\tau_j\right) + \left(1 - \underset{\tau_j|\cdot}{\mathrm{E}}\left(\tau_j\right)\right)c\right) + \underset{\tau_j|\cdot}{\mathrm{E}}\left(\tau_j\right)\log(\pi) + \left(1 - \underset{\tau_j|\cdot}{\mathrm{E}}\left(\tau_j\right)\right)\log(1-\pi) + \text{constant}. \end{aligned} \qquad [2.15]$$

as also provided by Rockova and George (2014). They further demonstrate that

$$\underset{\tau_j|\cdot}{\mathrm{E}}\left(\tau_j\right) = \hat{\tau}_j^{**} = \frac{\phi_{\hat{g}_j}\left(0, \sigma_g^2\right)\pi}{\phi_{\hat{g}_j}\left(0, \sigma_g^2\right)\pi + \phi_{\hat{g}_j}\left(0, \frac{\sigma_g^2}{c}\right)(1-\pi)}. \qquad [2.16]$$

Here, $\phi_x\left(\mu, \sigma^2\right)$ denotes, for example, the ordinate of a Gaussian pdf with mean μ and variance σ^2 evaluated at *x*, noting that $\hat{g}_j$ in Equation [2.16] denotes the M-step estimate of g_j for the current iterate. The M-step for providing the joint posterior mode of β and g is based on the use of the same set of mixed model equations provided in Equation [2.9], except that $\hat{\mathbf{D}}^{-1} = diag\left(\hat{\tau}_j^{**} + c\left(1 - \hat{\tau}_j^{**}\right)\right)$.

For all data analyses in this paper, our default value for *c* was 1000.

2.3.4 Hyperparameter estimation

2.3.4.1 Variance component estimation

The EM strategy for inferring upon $\boldsymbol{\beta}$ and **g** as outlined for BayesA and SSVS above is computationally similar to classical mixed model inference, holding constant hyperparameters such as variance components. In fact, BayesA defaults to RRBLUP of **g** as $\nu_g \to \infty$ whereas SSVS defaults to RRBLUP of **g** with $\pi = 1$ or c = 1, noting that genomic predictions or GBLUP of individual genetic merit, **u** = **Zg**, is simply **Z**(RRBLUP(**g**)). It should be quickly noted that the term GBLUP is typically reserved for the equivalent mixed effects model whereby one directly solves for **u** rather than for **g** in order to facilitate computational tractability since generally $n << m$ (Stranden and Garrick 2009); nevertheless, I take the liberty of referring to GBLUP(**u**) as being a linear function of RRBLUP(**g**).

I represent the vector of hyperparameters as $\boldsymbol{\theta} = \left(\sigma_e^2, \sigma_g^2, \nu_g\right)$ for BayesA and $\boldsymbol{\theta} = \left(\sigma_e^2, \sigma_g^2, \pi\right)$ for SSVS. I partition $\boldsymbol{\theta}$ into the variance components $\boldsymbol{\sigma} = \left(\sigma_e^2, \sigma_g^2\right)$ and remaining hyperparameters as $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$ such that, for example, $\boldsymbol{\theta}_{-\boldsymbol{\sigma}} = \nu_g$ in BayesA whereas $\boldsymbol{\theta}_{-\boldsymbol{\sigma}} = \pi$ in SSVS. Prior to the advent of MCMC, a convincing justification for EB in quantitative genetics was provided by Gianola *et al.* (1986) who recommended conditioning RRBLUP of **g** on REML estimates of $\boldsymbol{\sigma}$ recognizing that REML($\boldsymbol{\sigma}$) is equivalent to maximizing the marginal density of $p\left(\boldsymbol{\sigma} \mid \mathbf{y}\right)$ based on a flat prior for $\boldsymbol{\sigma}$, i.e., $p\left(\boldsymbol{\sigma}\right) \propto 1$ (Harville 1974). The resulting e-GBLUP estimates (i.e., **Z(**e-RRBLUP(**g**)) conditional on REML($\boldsymbol{\sigma}$)) are typically not practically different

from posterior means on **Zg** allowing for uncertainty on $\boldsymbol{\sigma}$, provided that $p(\boldsymbol{\sigma}|\mathbf{y})$ is reasonably symmetric.

Now prior uncertainty on the augmented variables $\boldsymbol{\tau}$ for both BayesA and SSVS is driven entirely by $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$ given that its prior distribution is written simply as $p(\boldsymbol{\tau}|\boldsymbol{\theta}_{-\boldsymbol{\sigma}})$, i.e., $\prod_{j=1}^{m} p(\tau_j|\nu_g)$ for BayesA and $\prod_{j=1}^{m} p(\tau_j|\pi)$ for SSVS. In the previous section, I maximized $p(\boldsymbol{\beta},\mathbf{g}|\boldsymbol{\theta},\mathbf{y})$ with respect to $\boldsymbol{\beta}$ and **g** by first evaluating $\underset{\tau_j|.}{\mathrm{E}}\left(\log p(\boldsymbol{\beta},\mathbf{g},\boldsymbol{\tau}|\boldsymbol{\theta},\mathbf{y})\right)$ in an E-step followed by maximizing this subsequent expectation with respect to $\boldsymbol{\beta}$ and **g** in a M-step for both of these models. With $\boldsymbol{\sigma}$ unknown, I start with:

$$p(\boldsymbol{\beta},\mathbf{g},\boldsymbol{\tau},\boldsymbol{\sigma}|\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y}) \propto p(\boldsymbol{\beta},\mathbf{g},\boldsymbol{\tau}|\boldsymbol{\sigma},\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y})\, p(\boldsymbol{\sigma}) \qquad [2.17]$$

where $p(\boldsymbol{\sigma}) = p(\sigma_e^2)\, p(\sigma_g^2)$ such that the marginal posterior density of augmented variables and variance components can be expressed as:

$$p(\boldsymbol{\tau},\boldsymbol{\sigma}|\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y}) \propto \int_{\boldsymbol{\beta}}\int_{\mathbf{g}} p(\boldsymbol{\beta},\mathbf{g},\boldsymbol{\tau}|\boldsymbol{\sigma},\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y})\, p(\boldsymbol{\sigma}) d\mathbf{g} d\boldsymbol{\beta} \qquad [2.18]$$

A "REML-like" strategy for inferring $\boldsymbol{\sigma}$ in BayesA or SSVS within an EM framework would then be to respectively evaluate the conditional expectation of L_A (Equation [2.5]) or L_S (Equation [2.14]) with respect to $\boldsymbol{\tau}$ as noted earlier, use its antilog to substitute for $p(\boldsymbol{\beta},\mathbf{g},\boldsymbol{\tau}|\boldsymbol{\sigma},\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y})$ in Equation [2.18] and maximize the resulting expression with respect to $\boldsymbol{\sigma}$ in order to evaluate the joint posterior mode of $p(\boldsymbol{\sigma}|\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y})$.

Recall again that with $p(\boldsymbol{\sigma}) \propto 1$ and the special conditions $\nu_g \to \infty$ for BayesA and $\pi = 1$ or $c = 1$ for SSVS, this strategy defaults to classical REML. The classical log REML function (Searle *et al.* 1992) can be written as follows:

$$l(\boldsymbol{\sigma} \mid \mathbf{y}) = -0.5\log|\mathbf{V}| - 0.5\log|\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}| - 0.5\mathbf{y}'\mathbf{P}\mathbf{y} \qquad [2.19]$$

with $\mathbf{V} = \mathbf{Z}\mathbf{D}\mathbf{Z}'\sigma_g^2 + \mathbf{I}\sigma_e^2$ and $\mathbf{P} = \mathbf{V}^{-1} - \mathbf{V}^{-1}\mathbf{X}\left(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right)^{-}\mathbf{X}'\mathbf{V}^{-1}$. In typical classical REML specifications involving uncorrelated random effects, **D** = **I**. I modify this expression for our BayesA and SSVS adaptations accordingly as:

$$\begin{aligned} l(\boldsymbol{\sigma}, \boldsymbol{\tau} \mid \boldsymbol{\theta}_{-\boldsymbol{\sigma}}, \mathbf{y}) &= \log\left(p(\boldsymbol{\sigma}, \boldsymbol{\tau} \mid \boldsymbol{\theta}_{-\boldsymbol{\sigma}}, \mathbf{y})\right) \\ &= \text{constant} - 0.5\log|\mathbf{V}| - 0.5\log|\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}| - 0.5\mathbf{y}'\mathbf{P}\mathbf{y} + \log p(\boldsymbol{\tau} \mid \boldsymbol{\theta}_{-\boldsymbol{\sigma}}) + \log p(\boldsymbol{\sigma}). \end{aligned} \qquad [2.20]$$

Recall for either hierarchical model, **D** is a function of $\boldsymbol{\tau}$ for which conditional expectations are used to derive $\hat{\mathbf{D}}^{-1} = diag\left\{\left(\hat{\tau}_j\right)^{-1}\right\}$ in BayesA or $\hat{\mathbf{D}}^{-1} = diag\left(\hat{\tau}_j^{**} + c\left(1 - \hat{\tau}_j^{**}\right)\right)$ in SSVS as noted earlier. Evaluating Equation [2.20] at $\hat{\mathbf{D}}^{-1}$ constitutes the only E-step for either model whereas the M-step is based on maximizing this resulting expression with respect to $\boldsymbol{\sigma}$. I denote the corresponding estimates as marginal modal a posteriori (MMAP) estimates in order to distinguish them from classical REML estimates.

Average Information REML (AIREML) is a particularly attractive hybrid Fisher's scoring/ Newton Raphson algorithm used to obtain REML estimates under classical Gaussian specifications for **g** based on the log likelihood of Equation [2.19] (Gilmour *et al.* 1995; Johnson and Thompson 1995). I adapt this algorithm for our proposed MMAP approach in Equation [2.20] by simply replacing $\boldsymbol{\tau}$ by $\hat{\boldsymbol{\tau}}$ from a previous E-step followed by maximizing Equation [2.20] with respect to $\boldsymbol{\sigma}$ in a M-step evaluated at $\hat{\boldsymbol{\tau}}$. To account for prior information in

$\log p(\boldsymbol{\sigma})$, I augment the AIREML first and second derivatives as provided by Johnson and Thompson (1995) with $\frac{\partial}{\partial \boldsymbol{\sigma}} \log p(\boldsymbol{\sigma})$ and $\frac{\partial^2}{\partial \boldsymbol{\sigma} \partial \boldsymbol{\sigma}'} \log p(\boldsymbol{\sigma})$, respectively. For all subsequent analyses in this paper, I use the non-informative prior for components of $\boldsymbol{\sigma}$ as advocated by Gelman (2006), i.e., $p\left(\sigma_e^2\right) \propto \left(\sigma_e^2\right)^{-\frac{1}{2}}$ and $p\left(\sigma_g^2\right) \propto \left(\sigma_g^2\right)^{-\frac{1}{2}}$.

2.3.4.2 Estimation of remaining hyperparameters

Suppose that one wishes to estimate $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$ (i.e., ν_g in BayesA or π in SSVS) as well so that a prior $p\left(\boldsymbol{\theta}_{-\boldsymbol{\sigma}}\right)$ is specified. Then Equation [2.20] could be further extended to additionally infer $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$ using the following expression:

$$l\left(\boldsymbol{\sigma}, \boldsymbol{\tau}, \boldsymbol{\theta}_{-\boldsymbol{\sigma}} \mid \mathbf{y}\right) = -0.5 \log \left|\mathbf{V}\right| - 0.5 \log \left|\mathbf{X}' \mathbf{V}^{-1} \mathbf{X}\right| - 0.5 \mathbf{y}' \mathbf{P} \mathbf{y} + \log p\left(\boldsymbol{\tau} \mid \boldsymbol{\theta}_{-\boldsymbol{\sigma}}\right) + \log p(\boldsymbol{\sigma}) + \log p\left(\boldsymbol{\theta}_{-\boldsymbol{\sigma}}\right). \quad [2.21]$$

Note that the "separability" (Rockova and George 2014) of Equation [2.21] into contributions involving $\boldsymbol{\sigma}$ versus $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$, i.e., $\frac{\partial}{\partial \boldsymbol{\sigma} \partial \boldsymbol{\theta}_{-\boldsymbol{\sigma}}'} l\left(\boldsymbol{\sigma}, \boldsymbol{\tau}, \boldsymbol{\theta}_{-\boldsymbol{\sigma}} \mid, \mathbf{y}\right) = \mathbf{0}$, allows independent M-steps for each of these two components of $\boldsymbol{\theta}$. In fact, I suggest a hybrid algorithmic strategy whereby AIREML-based optimization is used for estimating $\boldsymbol{\sigma}$ whereas only first derivative information is used for maximizing $p\left(\boldsymbol{\theta} \mid \mathbf{y}\right)$ with respect to $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$. That is, I propose maximizing Equation [2.21] with respect to $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$ by simply setting $\frac{\partial}{\partial \boldsymbol{\theta}_{-\boldsymbol{\sigma}}} \left(\log p\left(\boldsymbol{\tau} \mid \boldsymbol{\theta}_{-\boldsymbol{\sigma}}\right) + \log p\left(\boldsymbol{\theta}_{-\boldsymbol{\sigma}}\right)\right) d\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$, evaluated at the E-step, equal to 0. For all subsequent analyses in this paper, I considered

$p(\nu_g) \propto \frac{1}{(1+\nu_g)^2}$ and $p(\pi)$ to be a Beta(1,10) density, similar to what I've advocated in previous work (Yang and Tempelman 2012; Yang *et al.* 2015b).

Upon convergence, marginal modal estimates of $\boldsymbol{\sigma}$ and/or $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$ could be "plugged in" to provide EB estimates of $\boldsymbol{\beta}$ and **g** for BayesA as in Section 2.2 or for SSVS as in Section 2.3. I refer to these estimates as e-BayesA and e-SSVS respectively.

2.4 Data

2.4.1 Simulation Study

In order to address the feasibility of our proposed EB approaches defined by MMAP estimates of hyperparameters and subsequent EM-based estimates of SNP effects under BayesA and SSVS models, I developed a simulation study to compare those estimates to MCMC based posterior means as the gold standard. I simulated 20 replicated datasets using the R (R Core Team 2017) package `hypred` (Technow 2013). The simulated genome was composed of 5 chromosomes, each of length 1 Morgan and consisting of 10,000 equally spaced loci. Individuals were randomly mated to generate 400 animals within each of 2000 consecutive generations. The mutation rate was specified to be 2.5×10^{-5} per locus per generation. After Generation 2000, random matings were used to expand the population size to 2000 individuals in each of Generations 2001 and 2002. In Generation 2001, I deleted SNP genotypes with a minor allele frequency (MAF) less than 0.05 and deleted randomly one of any two adjacent SNP loci in complete LD with each other. I randomly chose 5000 of the remaining SNP to be our markers (i.e. **Z**) from which I further randomly selected $n_{qtl} = 30$ to be quantitative trait loci (QTL). I simulated allelic substitution effects, $\mathbf{g}_{qtl}$, for these QTL from a reflected gamma distribution with shape parameter 0.4 and scale parameter 1.66 (Hayes and Goddard 2001; Meuwissen *et al.*

2001); the corresponding genotypes $\mathbf{Z}_{qtl}$ for these animals were considered to be the a n_{qtl} column subset of the SNP genotype matrix $\mathbf{Z}$ such that the cumulative genetic merit or true breeding values ($\mathbf{u}_{TRUE}$) was $\mathbf{Z}_{qtl}\mathbf{g}_{qtl}$. Phenotypes for animals in Generation 2001 and 2002 were generated based on a heritability of 0.5, i.e., such that $\sigma_e^2 = \sigma_u^2$ = var($\mathbf{u}_{TRUE}$). Average pairwise LD among the 5000 SNP markers across all 20 replicates averaged 0.34.

In order to assess the effect of different marker densities on prediction accuracy, I selected every single, 2nd, 4th and 8th SNP markers resulting in 4 different marker densities, i.e., m= 5000, 2500, 1250, and 625 SNP markers across the 20 replicates.

2.4.2 Loblolly Pine Data

Resende *et al.* (2012) provided a data set involving 4854 SNP genotypes on 926 loblolly pine individuals. Upon excluding SNP with MAF<0.05 and those showing departure from Hardy Weinberg equilibrium (P<10^{-4}), 2684 SNPs remained. I further standardized elements of $\mathbf{Z}$ using $(z_{ij}^{*} - 2\hat{p}_j)\Big/\sqrt{2\hat{p}_j(1-\hat{p}_j)}$ as described previously. Although original phenotypes were not publicly available, Resende *et al.* (2012) provided deregressed EBV (DEBV) for 17 traits; DEBV are often used as proxies for phenotypes when raw data are not readily available. Deregression refers to the process by which the effect of unequal shrinkage on individual EBV based on a pedigreed analysis (i.e., without using genotypic information) is reversed to remove heterogeneity of variation due to unequal information in order to recreate "phenotypes" (see Garrick *et al.* 2009). I selected one disease resistance trait, absence or presence of rust, previously estimated to have a heritability of 0.21. After discarding data on individuals missing DEBV, 807 individuals remained.

To compare our proposed hierarchical EB estimation strategy with MCMC and with RRBLUP, I randomly split the data into 10 portions with each portion serving once as a test

dataset (81 individuals/dataset) and the remaining 9 portions serving as the training dataset (726 individuals/dataset) in a 10-fold cross-validation study.

2.5 Data Analysis

All parameters excluding ν_g in the BayesA model were estimated using MCMC based on 100,000 cycles of burn-in followed by an additional 100,000 cycles, saving every 10th sample for a total of 10,000 saved MCMC cycles. For our proposed EB strategy, MMAP estimation of $\boldsymbol{\theta}$ in RRBLUP, BayesA, and SSVS was based on a convergence criterion of $\frac{[\hat{\boldsymbol{\theta}}^{(k)} - \hat{\boldsymbol{\theta}}^{(k-1)}]'[\hat{\boldsymbol{\theta}}^{(k)} - \hat{\boldsymbol{\theta}}^{(k-1)}]}{[\hat{\boldsymbol{\theta}}^{(k)}]'[\hat{\boldsymbol{\theta}}^{(k)}]} < 10^{-4}$; after AIREML convergence of variance components, convergence on EM-based solutions to **g** were based on the same criteria. Because of the difficulties and slow convergence that I encountered in estimating ν_g based on our proposed MMAP strategy, ν_g was held constant to 5 for the BayesA model using both MMAP and MCMC. In our simulation study, the correlation between GEBV and $\mathbf{u}_{TRUE}$ in Generation 2002 was defined as the accuracy of (genomic) prediction whereby GEBV =$\mathbf{Z}\hat{\mathbf{g}}$ for $\hat{\mathbf{g}}$ being the posterior mean of **g** using MCMC or $\hat{\mathbf{g}}$ being EB estimates based on BayesA and SSVS and generated from analysis of Generation 2001 data. The RRBLUP model was also considered. Similarly, for the loblolly pine data analysis, I defined the accuracy of (cross-validation) prediction as the correlation between the test data DEBV and its predictions based on estimates derived from the training data.

Accuracies of prediction for both analyses of the simulated and the loblolly pine data were based on comparing the effects of three different sets of starting values of **g** in our EB methods: Set 1) e-RRBLUP of **g** conditional on REML($\boldsymbol{\sigma}$), Set 2) MCMC posterior means of all parameters based on same model as the corresponding EB approach and Set 3) **g** = 0 and MCMC posterior means for all other parameters based on same model as the corresponding EB

approach. To help derive properly scaled starting values ($\sigma^2_{g(0)}$) based on REML estimates ($\hat{\sigma}^2_{g(REML)}$) of σ^2_g for Set 1, I used $\sigma^2_{g(0)} = \frac{\nu_g - 2}{\nu_g}\hat{\sigma}^2_{g(REML)}$ for MMAP under e-BayesA and $\sigma^2_{g(0)} = \frac{\hat{\sigma}^2_{g(REML)}}{\hat{\pi}}$ for MMAP under e-SSVS with $\hat{\pi}$ being the posterior mean of π from the corresponding MCMC analyses. For e-BayesA, I additionally considered investigating the effect of basing the E-step on relative precisions (P), i.e., using $\hat{\tau}_j$, as per Equation [2.7] or on relative variances (V), i.e., using $\hat{\tau}^*_j$, as advocated by Karkkainen and Sillanpaa (2012) for each of the three sets of starting values described above. So, for example, Set 2(P) refers to using MCMC posterior means as starting values with the E-step based on $\hat{\tau}_j$ whereas Set 2(V) refers to using MCMC posterior means as starting values with the E-step based on $\hat{\tau}^*_j$ for e-BayesA. Since hyperparameters like σ^2_g or π are not truly well defined in simulation studies that attempt to mimic the LD between markers characteristic of real data (Yang *et al.* 2015b), our assessment of the relative performance of the different starting value/E-step strategies were based on their agreement with the corresponding MCMC posterior means given that MCMC inferences do not require asymptotic assumptions. A more ideal assessment might be based on cross-validation prediction accuracy, but I deemed that to be prohibitively expensive for a replicated simulation study.

2.6 Expectation maximization variable selection (EMVS)

By nature of the SSVS prior, estimates of SNP effects based on a SSVS model will be multimodal, such that EM estimates of **g** may precariously converge to local maxima. This was recognized by Rockova and George (2014) who recommended two strategies in tandem to

alleviate this problem: 1) a regularization procedure that they label as Expectation Maximization Variable Selection (EMVS) and 2) a deterministic annealing variant of EM (DAEM). EMVS involves gradually changing the values of *c* from relatively high values (highly peaked spikes on 0 for near-null elements of **g**) to relatively low values while executing EM. Rockova and George (2014) demonstrated that starting SSVS with a highly peaked spike (high *c*) and gradually increasing the variance of the spike (i.e., decreasing *c*) helps absorb negligible estimates of **g** in their examples. For the SSVS model characterized in Equation [9a], Rockova and George (2014) actually specified σ_g^2 as known, whereas I adaptively estimate it with our proposed MMAP approach as indicated earlier. Although Rockova and George (2014) provided no specific guidelines on how to choose the decreasing gradient set of values of *c*, I arbitrarily chose decreasing values of *c* within the set $S_c = \{100000, 60000, 10000, 5000, 2000, 1000\}$, ending with 1000 based on a trial and error assessment on the lowest value of *c* that maximized cross-validation accuracy on the loblolly pine dataset. Note that e-SSVS estimates of $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, and **g** based on one value of *c* are used as the starting values for e-SSVS estimation of $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, and **g** for the next value of *c* in S_c.

On the other hand, DAEM involves modifying the E-step in Equation [2.16] as follows:

$$\hat{\tau}_j^{**} = \frac{\left(\phi_{\hat{g}_j}\left(0, \sigma_g^2\right)\pi\right)^t}{\left(\phi_{\hat{g}_j}\left(0, \sigma_g^2\right)\pi\right)^t + \left(\phi_{\hat{g}_j}\left(0, \frac{\sigma_g^2}{c}\right)(1-\pi)\right)^t} \qquad [2.22]$$

Here $1/t$ ($0 < t < 1$) corresponds to a temperature specification on the degree of separation between multiple modes of the estimates; note that Equation [2.22] defaults to [2.16] when $t = 1$. Rockova and George (2014), following developments provided by Ueda and Nakano (1998), claimed that gradually increasing temperature settings starting at $1/t = 0$ increase the chances of finding the true global maximum. Note that at $1/t=0$, the corresponding estimates from this

procedure are effectively equivalent to e-RRBLUP(**g**) and REML($\boldsymbol{\sigma}$) (i.e. starting values defined in Set 2). Hence, at least as it pertains to **g**, using $1/t$=0 then corresponds to a situation whereby any local modes in its joint posterior density are completely smoothed away. As $1/t$ gradually decreases, multiple modes begin to appear. Hence the influence of starting values is weakened by keeping $1/t$ high during the early stages and gradually decreasing it to $1/t$ = 1 which directly corresponds to the joint posterior density of interest.

I jointly adapted EMVS and DAEM together, labeled as DAEMVS by Rockova and George (2014), by conducting our e-SSVS strategy over decreasing values of *c* in S_c within each of the decreasing temperature values (i.e. increasing *t*) in $S_t = \{0, 0.25, 0.5, 0.75, 1\}$ on each of the 10 loblolly pine training datasets. Within each subsequent specification of *t* in S_t I also conducted our e-SSVS strategy over increasing spike variance (i.e. decreasing *c*) in the set S_c such that e-SSVS estimates from each pair of *t* and *c* values cycling within *t* served as the starting values for the next pair of values of *t* and *c*.

2.7 Results

2.7.1 Simulation Study

Average MMAP estimates, based on the three different sets of starting values and the two different E-step strategies (relative precisions $\hat{\tau}_j$ vs. relative variances $\hat{\tau}_j^*$) as well as average MCMC estimates of $\boldsymbol{\sigma}$ across the 20 replicates under the BayesA model are provided in Table 2.1.

Table 2.1 Average MCMC and MMAP estimates of hyperparameters as a function of marker density, starting values and expectation-step (E-step) strategies under a BayesA model.

Average hyperparameter estimates across 20 simulated replicates								
Variance Component	Marker Density	E-step based on relative precisions(P)				E-step based on relative variances (V)		
		MCMC[1]	Set 1[2]	Set 2[3]	Set 3[4]	Set 1[2]	Set 2[3]	Set 3[4]
σ_g^2	625	2.26E-3	3.77E-3	3.77E-3	3.77E-3	2.22E-3	2.22E-3	2.22E-3
	1250	1.43E-3	2.55E-3	2.55E-3	2.55E-3	1.50E-3	1.50E-3	1.50E-3
	2500	6.98E-4	1.34E-3	1.32E-3	1.32E-3	7.75E-4	7.65E-4	7.68E-4
	5000	3.03E-4	6.35E-4	3.62E-4	6.84E-4	3.38E-4	3.29E-4	3.50E-4
σ_e^2	625	2.68	2.62	2.63	2.63	2.61	2.61	2.61
	1250	2.30	2.23	2.23	2.23	2.21	2.21	2.21
	2500	2.05	2.00	2.00	2.00	1.98	1.98	1.98
	5000	1.89	1.88	1.98	1.84	1.89	1.88	1.88

[1]MCMC: average posterior means, Other columns pertain to MMAP estimates based on relative precisions versus relative variances E-steps and three different sets of starting values: [2]Set 1) e-BLUP (**g**) and REML($\boldsymbol{\sigma}$),[3]Set 2) MCMC (**g** and $\boldsymbol{\sigma}$), [4]Set 3) **g** = 0 and MCMC ($\boldsymbol{\sigma}$)

By basing the E-step on relative precisions ($\left(\hat{\tau}_j\right)^{-1} = \mathrm{E}\left(\tau_j^{-1}\right)$), the three different sets of starting values lead to virtually identical estimates of $\boldsymbol{\sigma}$ at m = 625, 1250 and 2500 although the corresponding estimates of σ_g^2 in all three cases were consistently higher compared to MCMC estimates such that the opposite was observed for σ_e^2. However, at m = 5000, there was considerable disagreement between the three sets. When the E-step was based on relative variances ($\hat{\tau}_j^* = \mathrm{E}\left(\tau_j\right)$), MMAP led to average estimates that were much closer to average MCMC estimates for all specifications of m and starting value sets. The correlation between all 5 sets of estimates in Table 2.1 exceeded 0.995 for m = 625, 1250, and 2500, whereas these correlations dropped to 0.95-0.99 at m = 5000 with E-step based on relative precisions (results not reported). A closer look at the behavior of the three different sets of starting values and/or two different E-step strategies versus the corresponding MCMC posterior means for m = 5000 is

provided in Figure 2.1. It appears that starting with MCMC-based estimates (Set 2) for MMAP estimation generally led to closer agreement than the other two sets between MMAP estimates with the corresponding MCMC posterior means on σ_g^2 when the E-step was based on relative precisions. However, when the E-step was based on relative variances, the correspondence between MMAP and MCMC estimates were very close for all three sets of starting values.

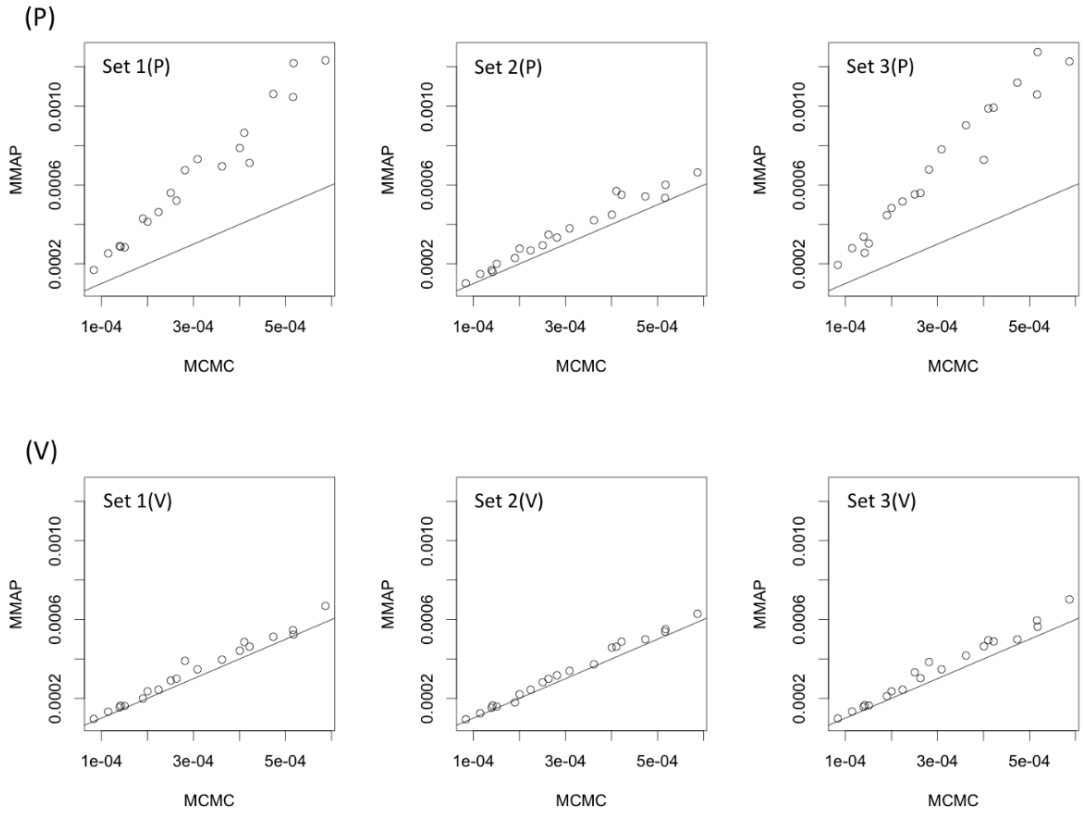

Figure 2.1 MMAP versus MCMC estimates of σ_g^2 in simulation study (20 replicated datasets each with 5,000 markers) under BayesA model with MMAP estimates based on different starting values for **g** and **σ**: Set 1) rrBLUP (**g**) and REML(**σ**), Set 2) MCMC (**g** and **σ**), or Set 3) **g** = 0 and MCMC (**σ**), Top panel of plots (P) pertain to use of E-step based on relative precisions whereas bottom panel of plots (V) pertain to use of E-step based on relative variances. Reference line of slope 1 and intercept 0 superimposed.

For the SSVS model, average MMAP estimates, based on the different starting values as well as average MCMC estimates of $\boldsymbol{\theta} = \left(\sigma_e^2, \sigma_g^2, \pi\right)$ across the 20 replicates are provided in Table 2.2.

Table 2.2 Average MCMC and MMAP estimates of hyperparameters as a function of marker density and starting values under a SSVS model.

Hyper-Parameter	Marker Density	Averages hyperparameter estimates across 20 simulated replicates			
		MCMC[1]	Set 1[2]	Set 2[3]	Set 3[4]
σ_g^2	625	1.04	0.67	1.08	3.76
	1250	0.70	0.68	0.71	2.68
	2500	0.32	0.47	0.34	1.59
	5000	0.20	0.48	0.21	0.96
σ_e^2	625	2.74	2.56	2.57	2.70
	1250	2.39	2.11	2.24	2.34
	2500	2.16	2.03	2.08	2.11
	5000	1.96	2.04	1.94	1.96
π	625	0.10	0.26	0.11	1.07E-3
	1250	0.05	0.16	0.05	4.57E-4
	2500	2.35E-2	0.04	1.85E-2	1.65E-4
	5000	5.74E-3	1.18E-3	3.91E-3	3.10E-5

[1]MCMC: average posterior means, other columns pertain to MMAP inference based on different sets of starting value sets or E-step strategies [2]Set 1) e-BLUP (**g)** and REML ($\boldsymbol{\sigma}$), [3]Set 2) MCMC (**g** and $\boldsymbol{\sigma}$**)**, and [4]Set 3) **g** = 0 and MCMC ($\boldsymbol{\sigma}$).

It seemed intuitively apparent that the best MMAP performance (i.e., highest correlation with MCMC estimates) was observed when starting values for $\boldsymbol{\theta}$ were based on MCMC estimates (Set 2); in fact, the correlation between Set 2 and MCMC estimates of σ_g^2 exceeded 0.99 for all marker densities. On the other hand, the worst performance involved the Set 3 starting values (i.e. **g** = **0**) where the corresponding correlation never exceeded 0.72 and was sometimes less than 0, although it did increase with marker density. A closer assessment of the relative agreement between MCMC estimates with MMAP estimates of σ_g^2 and π based on the three different sets of starting values and m = 5,000 markers is provided in Figure 2.2. MMAP estimates of σ_g^2 starting from MCMC estimates (Set 2) most closely aligned with MCMC

estimates of posterior means of σ_g^2. Likewise MMAP estimates of π starting at Set 2 agreed best with the corresponding MCMC estimates although MMAP estimates were typically slightly lower. However, MMAP estimates of σ_g^2 based on the other starting value sets (Sets 1 and 3) appeared to be badly biased upwards relative to the MCMC estimates. To seemingly compensate for that bias, estimates of π were badly biased downwards such that with Set 3) starting at **g** = 0, estimates of π rarely deviated from 0.

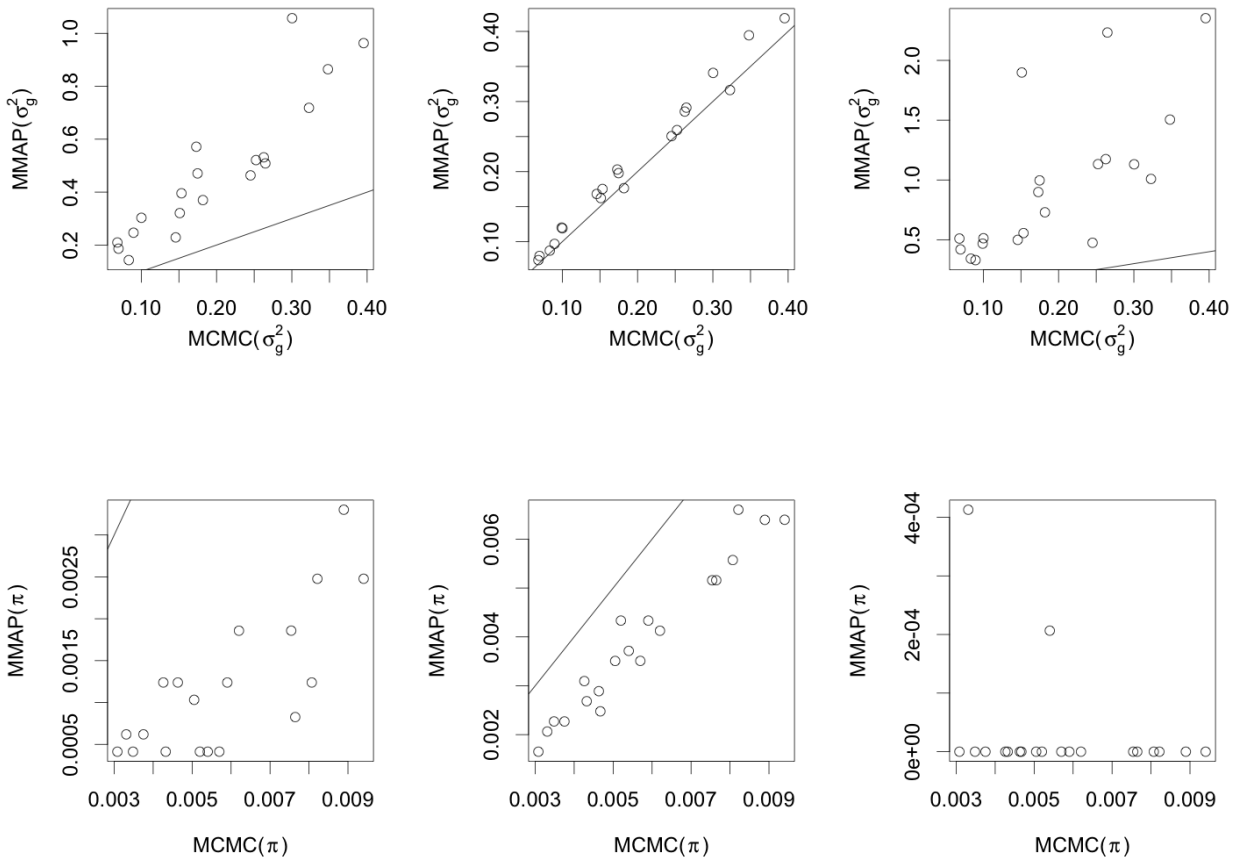

Figure 2.2 MMAP versus MCMC posterior means of σ_g^2 (top row) and of π (bottom row) in simulation study (20 replicated datasets each with 5,000 markers) under SSVS model with MMAP estimates based on different starting values for **g** and **θ**: Set 1) rrBLUP (**g**) and

REML($\boldsymbol{\sigma}$), Set 2) MCMC ($\mathbf{g}$ and $\boldsymbol{\theta}$), and Set 3) $\mathbf{g} = \mathbf{0}$ and MCMC($\boldsymbol{\theta}$) . Reference line of slope 1 and intercept 0 superimposed.

I compared GEBV in Generation 2002 based on estimates of $\mathbf{g}$ derived from the analysis of data on Generation 2001 using e-GBLUP, MCMC, and the various sets of starting values and/or E-step strategies as it pertains to our EB versions of e-BayesA and e-SSVS in Figure 2.3.

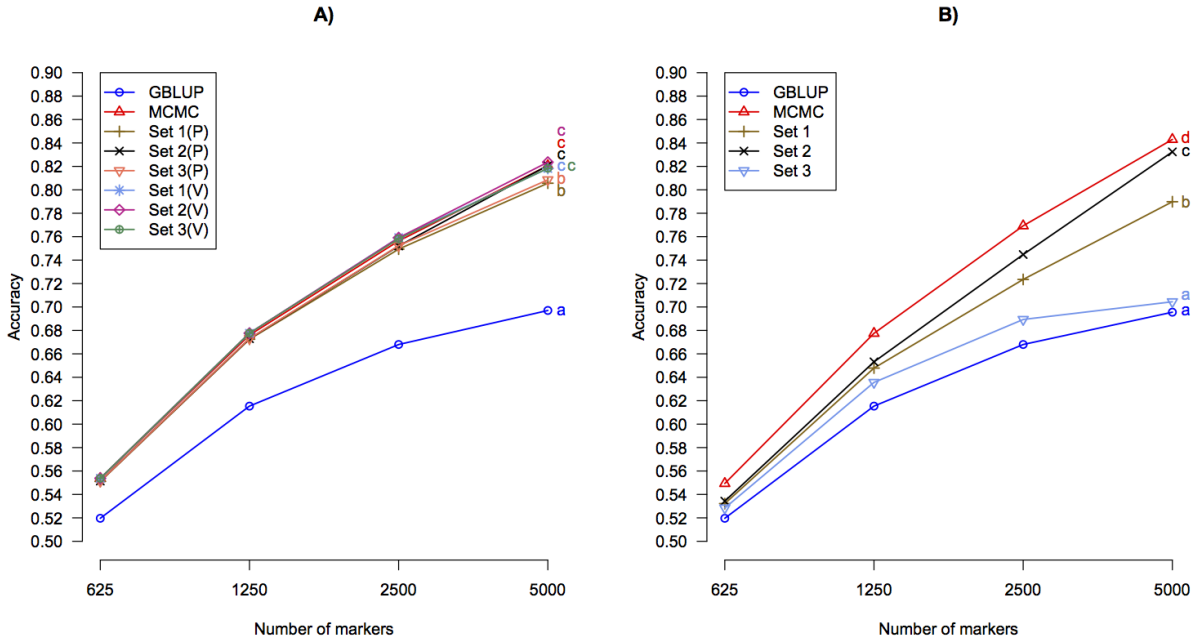

Figure 2.3 Mean accuracies of breeding value prediction for EB inference as a function of different marker densities (625, 1250, 2500, or 5000 markers) for BayesA (Panel A) and SSVS (Panel B) in simulation study (20 replicated datasets). e-GBLUP based on REML($\boldsymbol{\sigma}$), is same for both Panels A) and B) whereas MCMC refers to using fully Bayesian inference under MCMC for the corresponding model. Other lines pertain to EB inference based on different sets of starting value sets or E-step strategies Set 1) e-rrBLUP ($\mathbf{g}$) and REML ($\boldsymbol{\sigma}$), Set 2) MCMC ($\mathbf{g}$ and $\boldsymbol{\sigma}$), Set 3) $\mathbf{g} = 0$ and MCMC ($\boldsymbol{\sigma}$) with letter suffixes indicating whether the corresponding E-step was based on relative precisions (P) or relative variances (V). Letter codes used to separate estimates ($P<0.05$) having different accuracies based on the 5,000 marker analyses

e-GBLUP was always inferior to all other strategies with the gap in average accuracy generally increasing with increasing marker density. As further anticipated from our previous comparisons of MMAP estimates of σ_g^2 in Table 1 and Figure 2.2, BayesA MCMC posterior

means or e-BayesA estimates of GEBV based on starting values derived from these same MCMC estimates (Set 2) were generally superior to e-BayesA estimates based on other starting values (Sets 1 and 3) when the E-step was based on relative precisions (Figure 2.3A). However, no significant differences in accuracies were apparent between MCMC and e-BayesA estimates based on any of the three different sets of starting values when the E-step was based on relative variances. For SSVS (Figure 2.3B), MCMC posterior means and e-SSVS based on Set 2) starting values led to GEBV that were more accurate than GEBV starting at e-RRBLUP (Set 1) that were, in turn, more accurate than those starting with $\mathbf{g} = \mathbf{0}$ (Set 3).

2.7.2 Application to Loblolly Pine Data

The average cross-validation accuracies for the different models, sets of starting values, and E-step strategies (e-BayesA) over the 10 different replicates for the loblolly pine data analysis is summarized in Figure 2.4; additionally, results based on the DAEMVS strategies for e-SSVS are also provided. As consistent with BayesA in the simulation study, MCMC posterior means or e-BayesA estimates starting at MCMC estimates (Set 2) were generally superior to e-BayesA estimates based on either Set 1 or Set 3 when the E-step was based on relative precisions and to e-GBLUP. Similar conclusions could be drawn when the E-step was based on relative variances although there was significant improvement in the e-BayesA accuracies based on Set 1 or 3 starting values. For SSVS, MCMC dominated e-SSVS based on starting value Set 2 although Set 2 in turn outperformed Sets 1 and 3 or e-GBLUP as well.

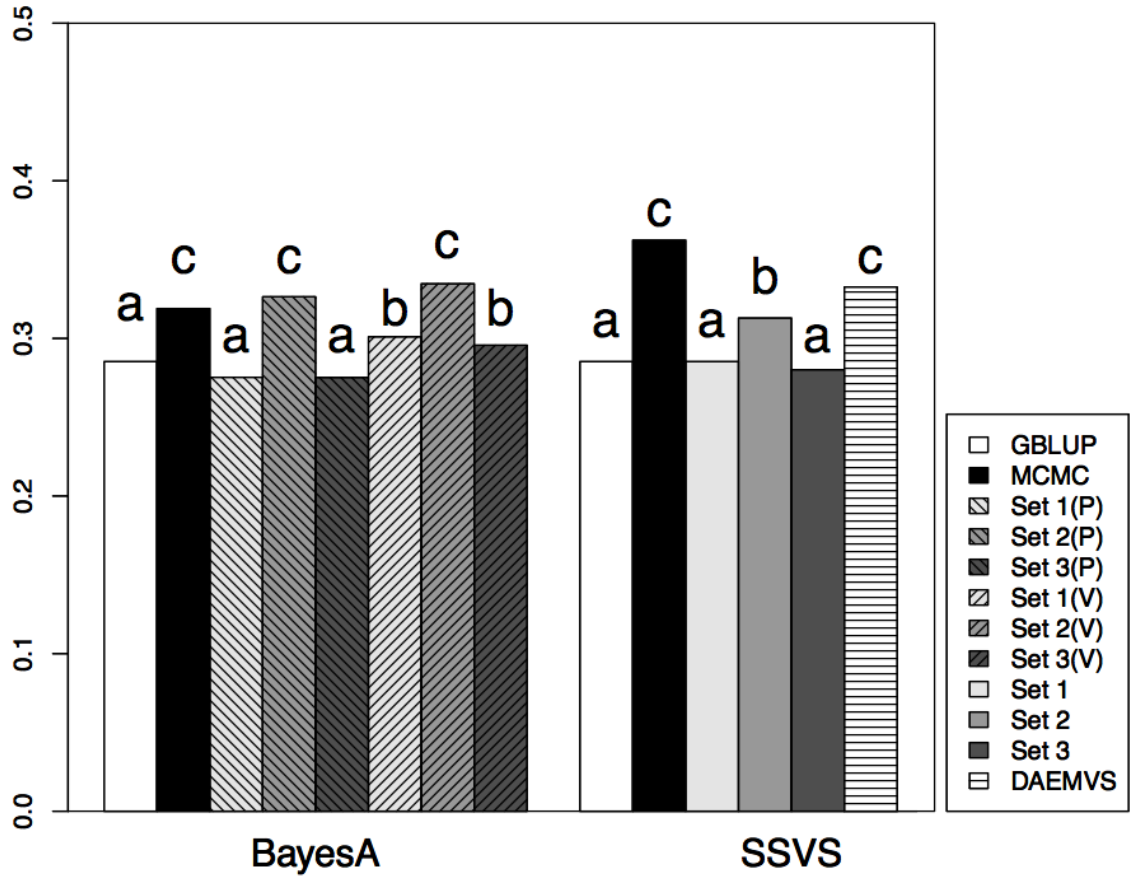

Figure 2.4 Average cross-validation accuracies for analysis of loblolly pine data based on empirical Bayes (EB) inference under BayesA (left cluster) or SSVS (right cluster) models. e-GBLUP, based on REML($\boldsymbol{\sigma}$), is same for both clusters whereas MCMC refers to using fully Bayesian inference for the corresponding model. Other bars pertain to EB inference based on different sets of starting value sets or E-step strategies: Set 1) e-rrBLUP ($\mathbf{g}$) and REML ($\boldsymbol{\sigma}$), Set 2) MCMC ($\mathbf{g}$ and $\boldsymbol{\sigma}$), or Set 3) $\mathbf{g} = 0$ and MCMC with letter suffixes indicating whether the corresponding E-step was based on relative precisions (P) or relative variances (V). Letter codes used to separate estimates (P<0.05) having different cross-validation prediction accuracies within each cluster.

For SSVS, I also evaluated whether DAEMVS could mitigate the impact of starting values for e-SSVS. Figure 2.5 provides a regularization plot (Rockova and George 2014) for one randomly chosen training dataset. In this plot, elements of $\hat{\mathbf{g}}$ are plotted as a function of pairs of

sequentially chosen values of c in S_t within t in S_c. Recall that at $t = 0$, $\hat{\mathbf{g}}$ are e-RRBLUP of $\mathbf{g}$ for

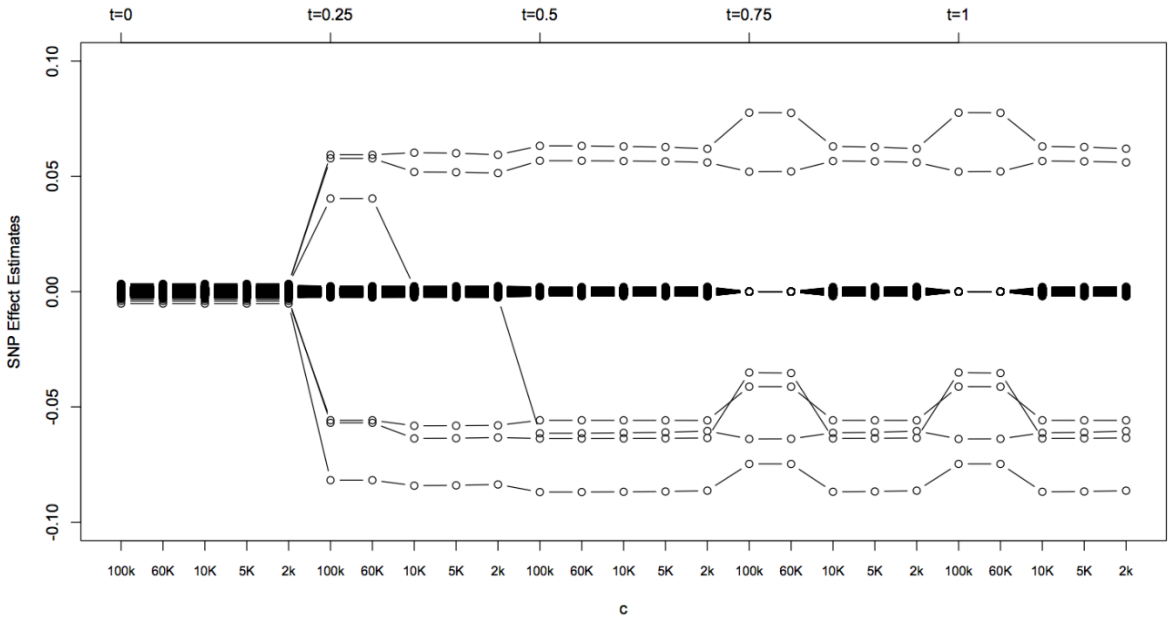

Figure 2.5 DAEMVS regularization plot for e-SSVS analysis of one training dataset analysis based on loblolly pine data. The x-axes pertain to precision on spike component variance (c at bottom) and inverse temperatures (t at top) whereas the y-axis denote SNP effect estimates $\hat{\mathbf{g}}$.

all values of c with Figure 2.5 indicating very little spread in elements of $\hat{\mathbf{g}}$ at $t = 0$. Beyond $t = 0.75$, it appeared that DAEM (i.e. impact of t) had little influence on spread of elements in $\hat{\mathbf{g}}$ whereas EMVS (i.e. impact of c) was far more influential. I did not consider values lower than $c = 1000$ since I noted that they compromised cross validation prediction accuracies (results not reported). Referring back to Figure 2.4, I determined that the DAEMVS based e-SSVS estimates of GEBV had a predictive accuracy comparable to MCMC estimates with greater ($P<0.05$) predictive accuracy than e-SSVS based on all other starting value sets.

2.8 Discussion

I have demonstrated that it is possible to develop computationally efficient empirical Bayes approaches to hierarchical Bayesian WGP models that additionally allow one to infer key

hyperparameters, whether those WGP are based on heavy-tailed (BayesA) or variable selection (SSVS) specifications on marker effects **g**. Our approach is based on a marginal modal inference procedure for estimating hyperparameters that closely emulates REML. Nevertheless, I have also demonstrated that the reliability of this EB strategy can critically depend on starting values. Starting at MCMC posterior means generally lead to accuracies in EB estimates that closely mirrored MCMC posterior means whereas a currently popular strategy based on starting all SNP effects at 0 appeared to be badly suboptimal.

At any rate, there appeared to be some promising possibilities for partially mitigating the effects of starting values. Our simulation study demonstrated no evidence of a difference between the various sets of starting values when the E-step was based on relative variances, as advocated by Karkkainen and Sillanpaa (2012) instead of relative precisions for e-BayesA implementations. However, the E-step based on relative variances was not quite as effective at mitigating the effects of starting values in our application to the loblolly pine data although it did improve cross-validation prediction accuracy relative to the conventional E-step for starting value sets not starting at MCMC posterior means for all parameters. I have no theoretical conjecture for the difference in performance between the two different E-step strategies. The relative variance strategy was based on a generalized EM (GEM) framework in which Karkkainen and Sillanpaa (2012) proposed that one merely iteratively computes conditional means (E-steps) or modes (M-steps) to substitute for random draws from the corresponding full conditional densities using MCMC. However, their GEM strategy does not suggest what full conditional parameterizations (e.g. relative variances versus relative precisions) that one should work with whereas EM is typically involves basing the E-step on the functional forms of the corresponding augmented variables in the joint posterior density. For e-SSVS, starting values

based on MCMC posterior means were also important for maximizing accuracy of converged GEBV; however, the application of the DAEMVS approach appeared to be rather effective for minimizing the influence of starting values in the loblolly pine data application such that it was even superior to e-SSVS starting at MCMC posterior means.

At any rate, it is reasonable to conclude from our results that implementations of e-BayesA or e-SSVS can lead to more accurate GEBV than the more common e-GBLUP strategy, mirroring what has been often concluded based on fully Bayesian (i.e. MCMC) implementations (Hayes *et al.* 2009; de Los Campos *et al.* 2013). Given that MCMC estimates lead to better starting values than, say, $\mathbf{g} = \mathbf{0}$, a practical recommendation might be that all initial analyses be first based on MCMC followed by computationally efficient regular EB updates at periodic intervals for WGP programs involving regular updates of phenotypes and genotypes (Wiggans *et al.* 2011). However, even this strategy warrants more rigorous study since it has been demonstrated that the relative importance of certain genomic regions contributing to GEBV in chickens may change over several generations (Fragomeni *et al.* 2014), likely because of the gradual breakdown of LD between SNP markers and QTL.

As indicated previously, there has not been much work addressing inferences on hyperparameters in EM-based implementations of the Bayesian alphabet models. The strategy proposed by Karkkainen and Sillanpaa (2012) was based on maximizing the joint posterior density of all parameters, including hyperparameters; however, their success in estimating these hyperparameters was limited, particularly for the BayesA model. It has been demonstrated previously that maximizing the joint posterior density of all parameters in a linear mixed model can be wrought with difficulties whereby "severe dependencies" can exist between the components of $\mathbf{g}$ and of $\boldsymbol{\theta}$ that hamper efficient estimation of $\boldsymbol{\theta}$ (Harville 1977). Conversely

marginal posterior estimates (i.e. MMAP) of $\boldsymbol{\theta}$ followed by joint posterior modal inference of $\mathbf{g}$ (and $\boldsymbol{\beta}$) conditional on MMAP($\boldsymbol{\theta}$) is typical of a more stable EB-based approach to inference with hierarchical models, similar to using REML followed by BLUP (Robinson 1991). Other researchers have taken yet a completely different approach by treating elements of $\boldsymbol{\theta}$ as if they were augmented variables whose uncertainty is accounted for by integrating them out of the joint posterior density whereas SNP-specific variances (i.e., $\boldsymbol{\tau}$) are considered as parameters to be estimated (Xu 2007; Cai *et al.* 2011; Huang *et al.* 2015). Given that each element of $\boldsymbol{\tau}$ defines the relative variance of a single element of $\mathbf{g}$, I are not sure that this is particularly advisable; nevertheless, more rigorous comparisons of their approach with our proposed strategy may be warranted.

I have also alleged, as others previously have (Karkkainen and Sillanpaa 2012; Sun *et al.* 2012), that computing time is substantially less for EM versus MCMC based implementations of BayesA; similar arguments naturally hold for SSVS. If hyperparameters ($\boldsymbol{\theta}$) are not to be estimated, this should be rather intuitive since EM is based on updating the full conditional means of $\boldsymbol{\beta}$, $\mathbf{g}$, and $\boldsymbol{\tau}$ whereas MCMC is based on drawing random samples from their corresponding full conditional Gaussian densities that require, in addition to specifications of these same conditional means, the specification of conditional variances followed by draws from a random Gaussian generator. Hence the computing time per MCMC cycle should be slightly greater per cycle than per EM iterate if $\boldsymbol{\theta}$ is not considered; furthermore, the number of EM iterates to reach convergence to a joint posterior mode for $\boldsymbol{\beta}$ and $\mathbf{g}$ is presumably less than the number of cycles to facilitate sufficiently reliable MCMC inference. This should be especially true of analyses updates based on the regular collection of more phenotypes and genotypes since EM would be programmed to start at the solutions from the most previous update. Even so, the

MCMC versus EM computing comparison is further complicated by the need to estimate $\boldsymbol{\theta}$, and the dimensionality of **g** amongst other things. We've advocated the use of a MMAP technique for $\boldsymbol{\theta}$ that closely emulates REML, advocating the use of the AIREML algorithm. To mitigate the dimensionality of **g** in determining some of the intermediate calculations in AIREML, most notably the inverse of the coefficient matrix in Equation [2.9], it is possible to reparameterize the model in terms of the breeding values **u**=**Zg**, to mirror current GBLUP implementations (Stranden and Garrick 2009).

At any rate, our results imply that researchers thinking about using current EM-based implementations of Bayesian alphabet models should be cognizant of the potential effect of starting values and potential remedies, as convergence problems will only intensify with increasing marker densities.

Chapter3 Genome Wide Association Analyses Based on Broadly Different Specifications for Prior Distributions, Genomic Windows, and Estimation Methods

3.1 Abstract

A currently popular strategy (EMMAX) for genome wide association (GWA) analysis infers association for the specific marker of interest by treating its effect as fixed while treating all other marker effects as classical Gaussian random effects. It may be more statistically coherent to specify all markers as sharing the same prior distribution, whether that distribution is Gaussian, heavy-tailed (BayesA), or has variable selection specifications based on a mixture of, say, two Gaussian distributions (SSVS). Furthermore, all such GWA inference should be formally based on posterior probabilities or test statistics as I present here, rather than merely being based on point estimates. I compared these three broad categories of priors within a simulation study to investigate the effects of different degrees of skewness for quantitative trait loci (QTL) effects and numbers of QTL using 43,266 SNP marker genotypes from 922 Duroc-Pietrain F2 cross pigs. Genomic regions were based either on single SNP associations, on non-overlapping windows of various fixed sizes (0.5 to 3 Mb) or on adaptively determined windows that cluster the genome into blocks based on linkage disequilibrium (LD). I found that SSVS and BayesA lead to the best receiver operating curve properties in almost all cases. I also evaluated approximate marginal a posteriori (MAP) approaches to BayesA and SSVS as potential computationally feasible alternatives; however, MAP inferences were not promising, particularly due to their sensitivity to starting values. I determined that it is advantageous to use variable selection specifications based on adaptively constructed genomic window lengths for GWA studies.

3.2 Introduction

Recent developments in genotyping technology have made single nucleotide polymorphism (SNP) genotype marker panels, based on thousands, and now millions, of markers, available for many livestock species (Wiggans *et al.* 2013; Kemper *et al.* 2015). Genome wide association (**GWA**) analyses have been increasingly used to help pinpoint regions containing potential causal variants or quantitative trait loci (**QTL**) for economically important phenotypes based on fitting SNP markers as covariates. An increasingly popular inferential approach for GWA is based on fitting phenotypes as a joint linear function of all markers using mixed-model procedures such as those invoked in the popular EMMAX procedure (Kang *et al.* 2010) and other similar procedures (Lippert *et al.* 2011; Zhou and Stephens 2012). Jointly accounting for all SNP effects when inferring upon a specific SNP marker of interest generally improves precision and power while also accounting for potential population structure (Kang *et al.* 2008).

Now GWA inferences in EMMAX and related procedures are based on treating the effect of the SNP marker of interest as fixed with all other marker effects as normally distributed random effects, noting that this process is repeated in turn for every single marker. These "fixed effects" hypothesis tests are based on generalized least squares (**GLS**) inference, with *P*-values being subsequently adjusted for the total number of markers or tests. Goddard *et al.* (2016) have recently pointed out the paradox with treating markers as fixed for inference but then otherwise as random to account for population structure for inference on association with other markers. Random effects modeling with all SNP effects treated as random, including the one of inferential interest, is synonymous with shrinkage based inference. Shrinkage or posterior inference has been demonstrated to facilitate reliable inference without any formal requirements for multiple comparison adjustments (Stephens and Balding 2009; Gelman *et al.* 2012). However, with SNP

markers treated as identically and independently distributed variables from a Gaussian distribution, the resulting shrinkage from random effects modeling can be too "hard", particularly with greater marker densities (Hayes 2013). Subsequently, this random effects test has been deemed to be far too conservative in various applications, as further demonstrated by Gualdron Duarte *et al.* (2014).

Prior specifications that are sparser than Gaussian may be more important for GWA since they more likely better characterize the true genetic architecture of most traits relative to Gaussian priors (de Los Campos *et al.* 2013). Sparser specifications have already been popularized in whole genome prediction (**WGP**), such as the Student *t* distribution used in BayesA (Meuwissen *et al.* 2001) and stochastic search and variable selection or SSVS (George and McCulloch 1993; Verbyla *et al.* 2009). Both specifications generally lead to far less shrinkage of large effects yet greater shrinkage of small effects compared to a Gaussian prior. In particular, the use of variable selection procedures facilitate the determination of posterior probabilities of association (**PPA**), whose control may be far more effective in maximizing both sensitivity and specificity of GWA (Fernando *et al.* 2017) compared to frequentist based inferences which require adjustments for multiple testing such as with EMMAX. Another common inferential strategy in GWA is to simply report the percent of variance explained by a marker or marker region (Fan *et al.* 2011; Tizioto *et al.* 2015; Wolc *et al.* 2016). However, point estimates of marker effects or percentage of variation explained, by themselves, do not provide formal evidence of association.

Most sparse prior WGP models have been implemented using Markov chain Monte Carlo (**MCMC**), which can be computationally expensive. Approximate analytical approaches based on the expectation–maximization (**EM**) algorithm to provide approximate maximum a posteriori

(**MAP**) estimates of SNP effects have been developed to address computational limitations in these sparse prior WGP models (Meuwissen *et al.* 2009; Hayashi and Iwata 2010; Sun *et al.* 2012; Chen and Tempelman 2015). Strategies for estimating/tuning hyperparameters for MAP inference have been proposed, including those proposed by Karkkainen and Sillanpaa (2012), Knürr *et al.* (2013) and Chen and Tempelman (2015), the latter adapting the average information restricted maximum likelihood (**AIREML**) algorithm for estimating hyperparameters in BayesA and SSVS specifications. These MAP implementations should also be assessed for their efficacy in GWA studies.

A pragmatic first objective in GWA is to pinpoint narrow genomic regions containing QTL rather than to specifically identify the QTL themselves, even though the latter is the ultimate goal. That is, a large number of SNP markers in a region surrounding a typically untyped QTL might be in high linkage disequilibrium (**LD**) with the QTL and with each other, thereby thwarting precise inference on the causal QTL. Different GWA methods may differ in the number of SNP markers inferred to have an association within a genomic region with, for example, EMMAX tending to draw associations with more SNP markers in LD with a QTL compared to use of SSVS (Guan and Stephens 2011; Goddard *et al.* 2016).

Increasingly, more GWA studies are based on inferences involving joint tests on all of the SNP markers within a narrow genomic region, recognizing that single SNP marker associations may be fraught by low statistical power or problems with multicollinearity or both(Fernando *et al.* 2017). Some GWA studies have been based on using several arbitrary window sizes based on either non-overlapping (Wolc *et al.* 2012; Moser *et al.* 2015; Wolc *et al.* 2016) or sliding windows (Schmid and Yang 2008). Because of the arbitrariness of fixed window sizes, whether defined by number of SNP markers or by physical length in base pairs, it is possible to split a

large LD block into 2 or more separate windows, thereby making such a division seemingly suboptimal for GWA. Substantially different window lengths have been used in different studies. For example, a 5 SNP window was used for GWA based on 51,385 SNP markers in pigs (Fan *et al.* 2011), whereas a 250 kilo base (**Kb**) window was used for 287,854 SNPs from the Welcome Trust Case Control Consortium (WTCCC) human data (Wellcome Trust Case Control Consortium 2007; Moser *et al.* 2015), and a 1 mega base (**Mb**) window was used for a 24,425 SNP marker panel in chickens (Wolc *et al.* 2012). Dehman *et al.* (2015) recently proposed an approach to adaptively cluster windows of SNP markers of varying sizes based on LD relationships. That is, they performed spatially constrained hierarchical clustering of SNPs by minimizing a distance measure derived from Ward's criterion based on LD r^2 between SNP markers. They surmised that this procedure would estimate a suitable specification of genomic windows within each chromosome using a modified version of the gap statistic. This method has been implemented in the R package `BALD` (Dehman and Neuvial 2015).

I had three primary objectives in this study. One was to examine the potential benefits of using sparser priors (i.e., BayesA and SSVS) relative to classical (i.e., based on normality) random effects specifications and strategies for GWA under a wide range of simulated architectures. A second objective was to assess whether the choice of different fixed genomic window sizes (specifically 0.5, 1, 2, and 3 Mb), versus adaptively inferred window sizes based on LD clustering, could impact GWA performance. A final objective was to assess the relative merit of approximate MAP approaches to theoretically exact yet computationally intensive MCMC approaches based on sparse prior specifications. Our assessments are based upon SNP marker genotypes and actual and simulated phenotypes on F2 pigs deriving from a Duroc-Pietrain cross.

3.3 Methods and materials

3.3.1 The hierarchical linear model

All analyses in this paper are based on a hierarchical linear model which be characterized by the classical mixed model specification:

$$\mathbf{y} = \mathbf{X\beta} + \mathbf{Zg} + \mathbf{e} \quad [3.1]$$

Here $\mathbf{y}$ is a *n* x 1 vector of phenotypes, **X** is a known *n* x *p* incidence matrix connecting **y** to the *p* x 1 vector of unknown fixed effects $\boldsymbol{\beta}$, **Z** is a known *n* x *m* matrix of genotypes connecting **y** to the *m* x 1 vector of unknown random SNP marker effects **g**, and **e** is the random error vector. I also assume throughout that $\mathbf{e} \sim N(0, \mathbf{I}\sigma_e^2)$ whereas $\mathbf{g} \mid \sigma_g^2, \mathbf{D} \sim N(0, \mathbf{D}\sigma_g^2)$ for **D** being a diagonal matrix of augmented data or variables (Chen and Tempelman 2015; Tempelman 2015). The prior specification on these diagonal elements is used to distinguish each of the competing models as described later.

For pedagogical reasons, I assume one record per individual although extensions to repeated records per individual are possible. An equivalent genomic animal (i.e., subject) effects model (VanRaden 2008) to Equation [3.1] can then be written as:

$$\mathbf{y} = \mathbf{X\beta} + \mathbf{a} + \mathbf{e} \quad [3.2]$$

with $\mathbf{a} = \mathbf{Zg}$ and all other terms defined previously as in [3.1] such that, conditionally on **D**,

$$\text{var}(\mathbf{a}) = \text{var}(\mathbf{Zg}) = \mathbf{Z}\,\text{var}(\mathbf{g})\mathbf{Z}' = \mathbf{ZDZ}'\sigma_g^2 \quad [3.3]$$

If $m \gg n$, it is generally computationally more tractable to work with the linear mixed model in Equation [3.2], along with the random effects specification in Equation [3.3], then back solve for the estimate of **g** that would be identical to those using a linear mixed model directly based on Equation [3.1] (Stranden and Garrick 2009).

3.3.2 Models

In the simplest model, which I denote as ridge regression (RR), there is no such data augmentation (i.e. **D** = **I**), such that the elements of **g** are marginally distributed as independent normal (de Los Campos *et al.* 2013). Sparser distributional specifications on **g** can be constructed as mixtures of normal densities (Andrews and Mallows 1974) by simply specifying prior distributions on functions of the diagonal elements of $\mathbf{D}$. Suppose that $\mathbf{D} = diag\left\{\tau_j\right\}_{j=1}^{m}$ with $\tau_j \sim \chi^{-2}(\nu_\tau, \nu_\tau)$; then it can be demonstrated that, marginally, elements of **g** are identically and independently distributed as a scaled Student *t* with scale parameter σ_g^2 and degrees of freedom ν_τ (Chen and Tempelman 2015). This model is typically referred to as BayesA (Meuwissen *et al.* 2001). Alternatively, if $\mathbf{D} = diag\left\{\tau_j + \left(1-\tau_j\right)/c\right\}_{j=1}^{m}$ where $\tau_j \sim Bernoulli\left(\pi_\tau\right); \tau_j = 0,1$ and *c>>1*, then the resulting model is Bayes SSVS in the spirit of George and McCulloch (1993). As a side-note, I use *c* = 1000 for all SSVS analyses in this paper.

As a final stage in each of the competing hierarchical models (RR, BayesA, and SSVS), I specify convenient conjugate priors wherever possible. For example, scaled inverted chi-square priors for variance components; i.e.

$$p\left(\sigma_e^2 \mid \nu_e, s_e^2\right) \propto \left(\sigma_e^2\right)^{-\left(\frac{\nu_e}{2}+1\right)} e^{-\frac{\nu_e s_e^2}{2\sigma_e^2}} \qquad [3.4]$$

and

$$p\left(\sigma_g^2 \mid \nu_g, s_g^2\right) \propto \left(\sigma_g^2\right)^{-\left(\frac{\nu_g}{2}+1\right)} e^{-\frac{\nu_g s_g^2}{2\sigma_g^2}} \qquad [3.5]$$

whereas I specify a Beta prior on π_τ in SSVS; i.e.,

$$p\left(\pi_\tau \mid \alpha_0, \beta_0\right) \propto \pi_\tau^{\alpha_0}\left(1-\pi_\tau\right)^{\beta_0}. \qquad [3.6]$$

As I explain later, I arbitrarily specify ν_τ as known (ν_τ=2.5), although conceptually it could also be estimated (Yang *et al.* 2015b). I assume throughout that $p(\boldsymbol{\beta}) \propto 1$ as $p(\boldsymbol{\beta})$ is typically

diffuse, although extensions to more informative specifications should be obvious. Furthermore, for all analyses in this paper, I specify Gelman's non-informative prior (Gelman 2006) for σ_e^2 in Equation [3.4] based on $v_e = -1$ and $s_e^2 = 0$ and for σ_g^2 in Equation [3.5] based on $v_g = -1$ and $s_g^2 = 0$. Furthermore, as per Yang and Tempelman (2012), I specify $\alpha_0 = 1$ and $\beta_0 = 9$.

3.3.3 Joint posterior density

Given the specifications above, the joint posterior density can be written as:

$$p\left(\boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\tau}, \sigma_e^2, \sigma_g^2, \theta_\tau \mid \mathbf{y}\right) \propto \left(\prod_{i=1}^{n} p\left(y_i \mid \boldsymbol{\beta}, \mathbf{g}, \sigma_e^2\right)\right)\left(\prod_{j=1}^{m} p\left(g_j \mid \sigma_g^2, \tau_j\right) p\left(\tau_j \mid \theta_\tau\right)\right) p(\boldsymbol{\beta})\, p\left(\sigma_g^2 \mid v_g, s_g^2\right) p\left(\sigma_e^2 \mid v_e, s_e^2\right) p\left(\theta_\tau\right) \quad [3.7]$$

Note that $p\left(\tau_j \mid \theta_\tau\right)$ specifies the $\chi^{-2}(\nu_\tau, \nu_\tau)$ density under BayesA (i.e., $\theta_\tau \equiv \nu_\tau$) whereas $p\left(\tau_j \mid \theta_\tau\right)$ specifies the $Bernoulli\left(\pi_\tau\right)$ density under SSVS (i.e., $\theta_\tau \equiv \pi_\tau$). Furthermore, $p\left(g_j \mid \sigma_g^2, \tau_j\right)$ is Gaussian with null means under all three competing models but with variance $\sigma_g^2 \tau_j$ under BayesA and variance $\sigma_g^2\left(\tau_j + \left(1 - \tau_j\right)/c\right)$ under SSVS. For RR, $\tau_j = 1 \forall j$ such that $p\left(g_j \mid \sigma_g^2, \tau_j = 1\right)$ is Gaussian with common variance $\sigma_g^2 \forall j$.

3.3.4 Algorithms

3.3.4.1 Markov Chain Monte Carlo

The MCMC sampling strategies that I use here for BayesA are similar to those provided in Yang and Tempelman (2012) and Yang *et al.* (2015b). However, since our parameterization is slightly different, I present the full conditional densities of interest for implementing BayesA in Appendix A. For similar reasons, I also provide the full conditional densities for SSVS in Appendix A as even our model differs from the model also labeled as SSVS in the genomic

prediction work of Verbyla *et al.* (2009) whereas it is virtually identical to the model presented in seminal SSVS paper by George and McCulloch (1993).

3.3.4.2 Maximum a posterior estimation

Complete details on our MAP procedure for both BayesA and SSVS are found in Chen and Tempelman (2015). Given that our application involved *m* >> *n*, I conducted MAP based inference on an equivalent subject-centric model using Equation [3.2] rather than based on a SNP effects model as in Equation [3.1]. Details on backsolving from a subject-centric model to provide estimates of SNP effects are provided in Appendix A. For pedagogical reasons, however, I work directly from the SNP-effect Model [3.1] in our subsequent developments. Conditional on **D**, the posterior variance-covariance matrix of **g**, or equivalently its prediction error variance-covariance (PEV) matrix from a frequentist viewpoint, can be written as: $\text{var}\left(\mathbf{g}\,|\,\mathbf{y},\mathbf{D},\sigma_g^2,\sigma_e^2,\theta_\tau\right)=\mathbf{C}^{gg|\mathbf{D}}$. This expression can be derived from the inverse of the mixed model coefficient matrix as:

$$\begin{bmatrix}\mathbf{C}^{\boldsymbol{\beta}\boldsymbol{\beta}|\mathbf{D}} & \mathbf{C}^{\boldsymbol{\beta}\mathbf{g}|\mathbf{D}}\\ \mathbf{C}^{\mathbf{g}\boldsymbol{\beta}|\mathbf{D}} & \mathbf{C}^{\mathbf{gg}|\mathbf{D}}\end{bmatrix}=\begin{bmatrix}\mathbf{X}'\mathbf{X}\sigma_e^{-2} & \mathbf{X}'\mathbf{Z}\sigma_e^{-2}\\ \mathbf{Z}'\mathbf{X}\sigma_e^{-2} & \mathbf{Z}'\mathbf{Z}\sigma_e^{-2}+\mathbf{D}^{-1}\sigma_g^{-2}\end{bmatrix}^{-1} \qquad [3.8]$$

That is, $\mathbf{C}^{gg|\mathbf{D}}$ is the random by random portion of the inverse coefficient matrix in Henderson's mixed model equations, conditional on **D**, σ_e^2, σ_g^2 and θ_τ ($\theta_\tau \equiv \nu_\tau$ for BayesA or $\theta_\tau \equiv \pi_\tau$ for SSVS). As noted earlier, values for hyperparameters such as σ_e^2, σ_g^2 and θ_τ required for Equation [3.8] can be determined using the REML or marginal maximum likelihood (MML) estimation strategies as described by Chen and Tempelman (2015) noting that I choose to fix ν_τ in BayesA as indicated earlier.

It can be readily demonstrated (Sorensen and Gianola 2002), that asymptotically $\text{MAP}(\mathbf{g}) \approx \text{E}(\mathbf{g}|\mathbf{y})$ whereby MAP(**g**) can be iteratively determined using EM based on Newton-Raphson for maximization (M-) steps interwoven with expectation (E-) steps on elements of **D** (Chen and Tempelman, 2015). Under RR, **D** = **I** such that $\mathbf{C}^{\mathbf{gg}} = \mathbf{C}^{gg|\mathbf{D}} \approx \text{var}(\mathbf{g}|\mathbf{y})$ represents the posterior variance-covariance matrix of **g** conditional on σ_e^2, σ_g^2 and θ_τ. In fact, MAP(**g**) is synonymous with BLUP(**g**) under RR. Furthermore, $\mathbf{C}^{\mathbf{gg}}$ is synonymous with the **g**-component of the observed information matrix of the joint conditional posterior density of **β** and **g**. This posterior density is formally defined in Equation [3.9].

$$p(\boldsymbol{\beta},\mathbf{g},|\sigma_e^2,\sigma_g^2,\theta_\tau|\mathbf{y}) \propto \left(\prod_{i=1}^{n} p(y_i|\boldsymbol{\beta},\mathbf{g},\sigma_e^2)\right)\left(\prod_{j=1}^{m}\int_{R_{\tau_j}} p(g_j|\sigma_g^2,\tau_j)\,p(\tau_j|\theta_\tau)d\tau_j\right) \qquad [3.9]$$

With **D** = **I**, there is no uncertainty on τ_j such the integration in Equation [3.9] is not necessary with $\mathbf{C}^{\mathbf{gg}}$ being directly obtainable for RR using Equation [3.8]. However, for BayesA and SSVS, uncertainty in **D** needs to be integrated out as per Equation [3.9]. An indirect strategy for asymptotically providing $\mathbf{C}^{\mathbf{gg}}$ for BayesA and SSVS is based on the strategy proposed by Louis (1982) with details provided in Appendix A. I subsequently use elements of $\mathbf{C}^{\mathbf{gg}}$ to asymptotically determine key components of $\text{var}(\mathbf{g}|\mathbf{y})$ for both single SNP and window based GWA testing using MAP under all three models, noting again that MAP and BLUP are synonymous under RR.

3.3.5 Conducting Genome Wide Association Analyses

3.3.5.1 Single SNP marker associations

I subsequently describe how I conducted GWA inference for single SNP associations based on the algorithms (MCMC vs. MAP) and models (RR, BayesA, and SSVS). With respect to

inference on association on SNP *j*, EMMAX is conceptually based on subsetting out Equation [3.1] as follows:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{z}_j g_j + \mathbf{Z}_{-j}\mathbf{g}_{-j} + \mathbf{e} \qquad [3.10]$$

That is, $\mathbf{Z}$ is partitioned into column *j*, $\mathbf{z}_j$, being the genotypes for SNP *j* and all other remaining columns in $\mathbf{Z}_{-j}$. In EMMAX, g_j is actually treated as fixed whereas $\mathbf{g}_{-j}$ is treated as classically random; i.e., characterized by a Gaussian prior distribution. Writing $\mathbf{W}_j = \begin{bmatrix} \mathbf{X} & \mathbf{z}_j \end{bmatrix}$ and $\mathbf{V}_{-j} = \mathbf{Z}_{-j}\mathbf{Z}'_{-j}\sigma_g^2 + \mathbf{I}\sigma_e^2$, the generalized least squares (GLS) estimator $\hat{g}_j$ of g_j, using all other markers to account for population structure, is the last element of the product $\left(\mathbf{W}_j'\mathbf{V}_{-j}^{-1}\mathbf{W}_j\right)^{-1}\mathbf{W}_j'\mathbf{V}_{-j}^{-1}\mathbf{y}$. Furthermore, the corresponding standard error $se\left(\hat{g}_j\right)$ is determined by the square root of the last diagonal element of $\left(\mathbf{W}_j'\mathbf{V}_{-j}^{-1}\mathbf{W}_j\right)^{-1}$. The test-statistic or "fixed effects" *z*-score for the EMMAX test can then be simply written as:

$$z_f = \frac{\hat{g}_j}{se\left(\hat{g}_j\right)} \qquad [3.11]$$

which is assumed to be $N(0,1)$ under H_o: $g_j = 0$. The "expedited" approach (Kang *et al.* 2010) in EMMAX, that I consider in this paper, is based on approximating $\mathbf{V}_{-j}$ with $\mathbf{V} = \mathbf{Z}\mathbf{Z}'\sigma_g^2 + \mathbf{I}\sigma_e^2$ for inference of association on all SNP $j = 1,2,\ldots,m$; furthermore, σ_g^2 and σ_e^2 estimated only once using REML in an initial analysis that treats all SNP marker effects as random. A GWA test for a particular SNP marker *j* using EMMAX then essentially involves treating its effect jointly as both fixed and random by replacing $\mathbf{Z}_{-j}\mathbf{g}_{-j}$ with $\mathbf{Z}\mathbf{g}$ on the right side of Equation [3.10], implying that this double counting of g_j as both fixed and random should be trivial with large *m*.

A classical shrinkage or random effects test for H_o: $g_j = 0$ is based on treating all SNP effects, including a marker *j* of particular interest, as having a Gaussian prior such that the point

estimate of the SNP substitution effect is based on fitting Equation [3.1] or, equivalently, backsolving from fitting Equation [3.3] as demonstrated by Stranden and Garrick (2009) and also in Appendix A. A corresponding test statistic (z_r) can be based on dividing $\tilde{g}_j$, the BLUP of g_j, by the square root of its prediction error variance (*PEV*) where $PEV\left(\tilde{g}_j\right) = \text{var}\left(\tilde{g}_j - g_j\right)$ from a frequentist perspective. From a Bayesian perspective, the corresponding test statistic can be interpreted as a posterior *z*-score (Gelman *et al.* 2012) since $\tilde{g}_j$ is analogous to a posterior mean (i.e., $\tilde{g}_j = \text{E}\left(g_j \mid \mathbf{y}, \hat{\sigma}_e^2, \hat{\sigma}_g^2\right) \approx \text{E}\left(g_j \mid \mathbf{y}\right)$) whereas the *PEV* is analogous to a posterior variance with $PEV\left(\tilde{g}_j\right) = \text{var}\left(g_j \mid \mathbf{y}, \hat{\sigma}_e^2, \hat{\sigma}_g^2\right) \approx \text{var}\left(g_j \mid \mathbf{y}\right)$. I refer to this inference strategy as **RR-BLUP**. It is important to indicate, nevertheless, that these RR-BLUP inferences are empirical Bayesian (Robinson 1991) since these posterior means and variances are typically conditioned upon REML estimates of σ_e^2 and σ_u^2 . The posterior *z*-score (Gelman *et al.* 2012) can then equivalently derived from both frequentist and Bayesian perspectives as indicated in Equation [3.12].

$$z_r = \frac{\tilde{g}_j}{\sqrt{PEV\left(\tilde{g}_j\right)}} = \frac{\text{E}\left(g_j \mid \mathbf{y}\right)}{\sqrt{\text{var}\left(g_j \mid \mathbf{y}\right)}} \qquad [3.12]$$

Now Gualdron Duarte *et al.* (2014), with a proof provided later by Bernal Rubio *et al.* (2016), determined that the "fixed effects" or EMMAX *z*-score, z_f in Equation [3.11], could be equivalently derived by treating all markers as classically random, but by dividing the corresponding BLUP $\tilde{g}_j$ for marker *j* by the square root of its frequentist definition of variance $\text{var}\left(\tilde{g}_j\right)$ as characterized by classical mixed model theory (Searle *et al.* 1992) in Equation [3.13].

$$\text{var}\left(\tilde{g}_j\right) = \text{var}\left(g_j\right) - PEV\left(\tilde{g}_j\right) = \sigma_g^2 - PEV\left(\tilde{g}_j\right) \quad [3.13]$$

In other words, one can rewrite the fixed effects test provided in both its frequentist (numerator = $\tilde{g}_j$) and Bayesian (numerator = $\text{E}\left(g_j \mid \mathbf{y}\right)$) representations as in Equation [3.14].

$$z_f = \frac{\tilde{g}_j}{\sqrt{\sigma_g^2 - PEV\left(\tilde{g}_j\right)}} = \frac{\text{E}\left(g_j \mid \mathbf{y}\right)}{\sqrt{\sigma_g^2 - \text{var}\left(g_j \mid \mathbf{y}\right)}}. \quad [3.14]$$

Note that the computation using Equation [3.14] is far more tractable than that implied with Equation [3.11]. That is, Equation [3.14] only requires computing BLUP of **g** and its corresponding

PEV in one single step determination for all *m* tests whereas Equation [3.11] imply *m* different mixed model analyses, each one in turn explicitly treating a different SNP marker effect as fixed.

I perceive no computationally tractable "fixed effects" test analogous to EMMAX that I could adapt for MAP based on sparser priors (e.g., BayesA and SSVS). For BayesA, for example, this would entail treating the marker of interest *j* as fixed with all other markers treated as scaled Student *t*- distributed. However, a posterior or random effects *z*-score test can be constructed using the MAP estimate of g_j as the numerator and its asymptotic posterior standard error as the denominator, noting that MAP and the posterior mean of **g** should approach each other asymptotically. Details on deriving those asymptotic standard errors (i.e., based on deriving $\mathbf{C}^{\mathbf{gg}}$) for use in Equation [3.11] for these sparse prior specifications are provided in Appendix A such that I refer to these two corresponding GWA inference strategies as MAP-BayesA and MAP-SSVS.

For SSVS based single SNP inferences using MCMC, I based inferences on the PPA for SNP marker *j* (i.e. PPA_j) as in Equation [3.15].

$$PPA_j = \frac{\sum_{l=1}^{N} \tau_{j(l)}}{N} \qquad [3.15]$$

Here N denotes the number of MCMC cycles saved for posterior inference and $\tau_{j(l)}$ is a binary draw from the full conditional distribution of τ_j at MCMC cycle l. I denote this GWA method as MCMC-SSVS.

Since there is no variable selection inherent with BayesA under MCMC, I based single SNP inferences on a Bayesian analog to a *P*-value using

$$\hat{p}_j = 2\min\left(\frac{\sum_{l=1}^{N} I\left(g_{j(l)} > 0\right)}{N}, 1 - \frac{\sum_{l=1}^{N} I\left(g_{j(l)} > 0\right)}{N} \right) \qquad [3.16]$$

(Bello *et al.* 2010) where the indicator variable $I(.)$ = 1 if the condition within the argument is true and 0 otherwise. I denote this particular GWA method as MCMC-BayesA,

3.3.5.2 Windows based associations

Window-based extensions to all of the above tests were also developed, some based on work previously presented above. Suppose that window k, k = 1,2, 3, … , K contains n_k markers such that $\mathbf{Z}$ can be partitioned accordingly into $\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_1 & \mathbf{Z}_2 & \cdots & \mathbf{Z}_K \end{bmatrix}$ with $\mathbf{Z}_k$ having n_k columns, implying then that window k contains n_k SNP markers. Similarly, the vector $\mathbf{g}$ is partitioned accordingly; i.e. $\mathbf{g} = \begin{bmatrix} \mathbf{g}_1' & \mathbf{g}_2' & \cdots & \mathbf{g}_K' \end{bmatrix}'$ such that $\mathbf{g}_k$ is of dimension n_k x 1. Recall that I denoted $\mathbf{C}^{gg} = PEV\left(\tilde{\mathbf{g}}\right)$. For our proposed windows-based test, the key components of $\mathbf{C}^{gg}$ can be partitioned into K different blocks along the block diagonal; i.e., $\mathbf{C}_1^{gg}$, $\mathbf{C}_2^{gg}$, …, $\mathbf{C}_K^{gg}$ where

$\mathbf{C}_k^{gg}$ is of dimension n_k x n_k . The extension to a joint "fixed effects" or EMMAX like test on n_k markers in window k involves the following extension of Equation [3.14].

$$\chi_f^2 = \tilde{\mathbf{g}}_k (\mathbf{I}_{n_k} \sigma_g^2 - \mathbf{C}_k^{gg})^{-1} \tilde{\mathbf{g}}_k \qquad [3.17]$$

That is, it can be readily demonstrated, extending results from Bernal Rubio *et al.* (2016), that χ_f^2 is chi-square distributed with n_k degrees of freedom under H_o: $\mathbf{g}_k = 0$. The corresponding extension to a joint classical "random effects" or RRBLUP test on window k is provided in Equation [3.18]

$$\chi_r^2 = \tilde{\mathbf{g}}_k (\mathbf{C}_k^{gg})^{-1} \tilde{\mathbf{g}}_k \qquad [3.18]$$

which would also be considered to be chi-square distributed with n_k degrees of freedom under H_o: $\mathbf{g}_k = 0$. Similarly, one could use Equation [3.18] to construct the same tests for MAP-BayesA and MAP-SSVS but basing the $\mathbf{C}^{\mathbf{gg}}$ on the corresponding asymptotic posterior variance-covariance matrices as derived in Appendix A.

For windows based inference using MCMC-SSVS, I simply compute the PPA for window k (i.e. PPA_k) in Equation [3.19], following that presented in Fernando *et al.* (2017).

$$PPA_k = \frac{\sum_{l=1}^{N} \left(I(\sum_{j=1}^{n_k} \tau_{kj(l)}) > 0 \right)}{N} \qquad [3.19]$$

Here, $\tau_{kj(l)}$ defines a binary draw from the full conditional distribution of τ_j for SNP marker j located within window k drawn during MCMC cycle l. Note then that $I(\sum_{j=1}^{n_k} \tau_{kj(l)}) > 0$ is equal to 1 when any of the draws of $\tau_{kj(l)}$ within window k are equal to 1.

For windows based GWA inference under MCMC-BayesA, I propose inferring upon the posterior probability of the proportion (q_w) of the genetic variance explained by the markers in a

genomic window relative to the total genetic variance as proposed by Fernando and Garrick (2013) and determined in the following manner. First note that the genotypic value that is attributed to a genomic window *k* is defined as in Equation [2.20].

$$\mathbf{a}_k = \mathbf{Z}_k \mathbf{g}_k \qquad [3.20]$$

Then the variance explained by the window is defined as

$$\sigma^2_{a_k} = \frac{\mathbf{a}_k^{'}\mathbf{a}_k}{n_k} - (\frac{\mathbf{1}_{n_k}^{'}\mathbf{a}_k}{n_k})^2 \qquad [3.21]$$

Similarly, the total genetic variance is computed as

$$\sigma^2_a = \frac{\mathbf{a}^{'}\mathbf{a}}{m} - (\frac{\mathbf{1}_m^{'}\mathbf{a}}{m})^2 \qquad [3.22]$$

Hence, the proportion of genetic variance that is explained by marker in window *k* is defined as

$$q_k = \frac{\sigma^2_{a_k}}{\sigma^2_a} \qquad [3.23]$$

Suppose that I deem genomic windows that explain more than 1% of the total genetic variance as being of potential interest. Hence, a variable selection modification of MCMC-BayesA can be simply be based on the proportion of MCMC samples for which the genetic variance (q_k) for window *k* exceeds 0.01 (Fernando and Garrick 2013). One advantage of this approach is that it can be applied to any MCMC analyses based on a model where variable selection is not explicitly specified.

3.4 Data

3.4.1 Simulation Study

In order to compare the various models (RR, BayesA, and SSVS) and algorithms (MAP vs. MCMC), I simulated data based on the Michigan State University Pig Resource Population (MSUPRP) raised at the Michigan State University Swine Teaching and Research Farm, East Lansing, MI (Edwards *et al.* 2008) . I specifically started with the SNP markers chosen for analysis by Gualdron Duarte *et al.* (2014) which included 928 Duroc-Pietrain F2 crosses. Roughly 1/3 of these pigs were directly genotyped using the Illumina Porcine SNP60 beadchip (60K) whereas the remaining F2 animals with genotyped using a lower density 9K set but whose genotypes were subsequently imputed to the 60K set (Gualdron Duarte *et al.* 2013). Edits excluded animals with more than 10% of their SNP markers missing, excluding SNP markers with more than 10% of animals missing genotypes for those markers, and excluding SNPs with minor allele frequency (MAF) below 0.01 (Gualdron Duarte *et al.* 2014). Some adjacent markers were in complete LD with each other. To circumvent multicollinearity issues, particularly its role in generating multimodality in the MCMC generated posterior densities for some SNP markers (Calus *et al.* 2015), I randomly deleted one SNP within an adjacent pair in complete LD with each other before further analyses. After invoking this edit, 43,266 SNPs remained. The original data source can be downloaded from https://msu.edu/~steibelj/JP_files/GBLUP.html.

To simulate different but representative genetic architectures, I generated QTL effects from three different Gamma densities with demonstrably different values of shape (γ) ranging from an effectively oligogenic density ($\gamma = 0.18$) which effectively specifies relatively much fewer QTL with large effects to an effectively polygenic Gaussian density ($\gamma = 3.00$) where most QTL have

intermediate effects with symmetrically small and large effects on either side. A third intermediate value ($\gamma = 1.48$) was also chosen. A good illustration of the gamma density of QTL effects based on these three different specifications for γ is provided in Figure 3.1. Note that this range in γ values for QTL effects has been reported for various traits in livestock based on previous empirical work (Hayes and Goddard 2001).

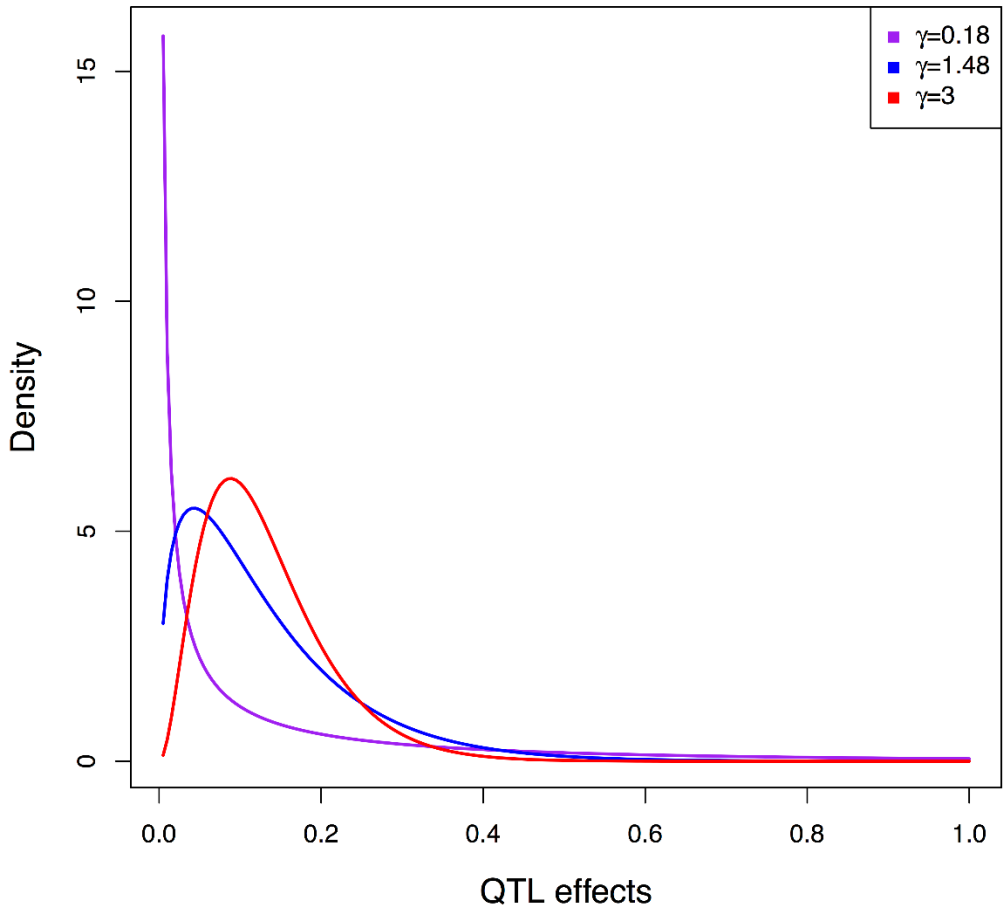

Figure 3.1 Distribution of quantitative trait loci effects under a Gamma distribution for different specifications of shape (magenta curve $\gamma = 0.18$, blue curve $\gamma = 1.48$ and red curve $\gamma = 3.00$)

In addition to the distribution of QTL effects, I conjectured that the number of QTLs (n_{qtl}) may also influence GWA performance such that I considered $n_{qtl} = 30$, 90, or 300. Hence, I

simulated 10 replicated populations under each of the 3 x 3 = 9 different scenarios pertaining to the 3 different values for each of γ and of n_{qtl}. Each of the 90 simulated datasets were based on utilization of the 43,266 SNP marker genotypes on the n = 922 MSUPRP F2 pigs as previously described. Within each dataset, allelic substitution effects, $\mathbf{g}_{qtl}$, were simulated for each of the n_{qtl} randomly chosen SNP markers from the corresponding gamma distribution having shape γ, with a randomly chosen half of those effects multiplied by -1 as per Meuwissen *et al.* (2001). The corresponding genotypes $\mathbf{Z}_{qtl}$ for QTL on these animals were then a n x n_{qtl} subset of the SNP genotype matrix $\mathbf{Z}$ such that the cumulative genetic merit or true breeding values was determined as $\mathbf{u}_{TRUE} = \mathbf{Z}_{qtl}\mathbf{g}_{qtl}$. Phenotypes for animals were generated based on a heritability of 0.45 as estimated for 13th-week tenth rib backfat from this same dataset. Only the remaining (i.e., non-QTL) marker genotypes $\mathbf{Z}_{-qtl}$ were used for all simulation study analyses.

In the simulation study, all parameters excluding v_τ in BayesA were estimated using both MCMC and MAP. For MCMC, I ran 200,000 iterations, discarding the first 100,000 iterations as burn-in and basing inference on saving every 10 of the remaining 100,000 cycles for a total of 10,000 samples from the posterior density. Using MAP, estimation of variance components ($\boldsymbol{\theta}$) for BayesA and SSVS was based on a convergence criterion of $\frac{[\hat{\boldsymbol{\theta}}^{(k)} - \hat{\boldsymbol{\theta}}^{(k-1)}]'[\hat{\boldsymbol{\theta}}^{(k)} - \hat{\boldsymbol{\theta}}^{(k-1)}]}{[\hat{\boldsymbol{\theta}}^{(k)}]'[\hat{\boldsymbol{\theta}}^{(k)}]} < 10^{-6}$. Based on our previous experience (Chen and Tempelman, 2015), I recognized that the specification of starting values in MAP-SSVS and MAP-BayesA was important for genomic prediction accuracy and, hence, likely important for GWA inferences as well. Strategies for specifying starting values for σ_g^2, σ_e^2, $\mathbf{g}$ and $\boldsymbol{\tau}$ may pragmatically involve using REML and RRBLUP inferences as in Chen and Tempelman (2015) since RRBLUP is not computationally intensive. For MAP-BayesA, starting values were based on REML estimates $\hat{\sigma}^2_{g(REML)}$ and

$\hat{\sigma}^2_{e(REML)}$ using $\sigma^2_{g(0)} = \frac{\nu_g - 2}{\nu_g}\hat{\sigma}^2_{g(REML)}$ for σ^2_g, $\mathbf{g}_0 = BLUP(\mathbf{g}) = \{g_{0j}\}_{j=1}^{m}$ for $\mathbf{g}$ and

$\tau_{(0)j} = \frac{\frac{g^2_{0j}}{\sigma^2_{g(0)}} + \nu_g}{\nu_g - 1}$ for τ_j, $j = 1, 2, \ldots, m$, based on the posterior expectation derived from its full conditional density. For MAP-SSVS, the corresponding starting values were $\sigma^2_{g(0)} = \frac{\hat{\sigma}^2_{g(REML)}}{\pi_0}$ for σ^2_g with the starting value $\pi_{\tau(0)}$ for π_τ based, in turn, on starting values for τ_j (i.e., SNP-specific PPA) which were determined in the following manner. First of all, EMMAX-based *P*-values for each SNP were converted to local false discovery rate (lFDR) estimates using the R package `ashr` (Stephens 2017). It has been demonstrated that these lFDR estimates, in turn, can be used to approximate PPA using PPA ≈ 1- lFDR (Stephens 2017). These approximate PPA values were then chosen as the starting values for τ_j in MAP-SSVS. In turn, these starting values for τ_j were used to derive the starting value for π_0 in MAP-SSVS using the posterior expectation from its full conditional density, i.e., $\pi_0 = \frac{\alpha_0 + \sum_{j=1}^{m}\tau_j}{\alpha_0 + \beta_0 + m}$. Upon convergence of variance components using the AIREML procedure outlined in Chen and Tempelman (2015), convergence of MAP-based solutions to **g** were based on the same criteria.

Single SNP marker inferences were based on the procedures outlined previously; i.e. for MAP by comparing z_r in Equation [3.12] for the random effect tests for RRBLUP, MAP-BayesA, and MAP-SSVS and z_f for the EMMAX test in Equation [3.11] to a standard normal distribution. Furthermore, the estimates of PPA and Bayesian *P*-values provided in Equations [3.15] and [3.16] were respectively used for GWA under MCMC-SSVS and MCMC-BayesA. Since the

remaining genotypes $\mathbf{Z}_{-qtl}$ did not include the simulated QTL, a SNP marker was declared a true positive if a QTL was located between that marker and its closest SNP neighbor on either side.

Window based inference was based on the procedures outlined previously; i.e. for MAP by computing χ_r^2 in Equation [18] for the random effect tests using RRBLUP, MAP-BayesA, and MAP-SSVS and χ_f^2 for fixed effects test in Equation [3.17] under EMMAX. These test statistics were compared to a chi-square distribution with degrees of freedom n_k. Furthermore, GWA was based on the PPA that $q_k > 0.01$ as provided in Equation [3.23] for MCMC-BayesA and on the PPA for MCMC-SSVS as provided in Equation [3.19].

For windows-based inference, four alternative fixed window sizes were chosen: 0.5, 1, 2, or 3 Mb. The genome map used was the *Sus Scrofa* build 10.2 (http://www.ensembl.org/Sus_scrofa/Info/Index). Also, as per Moser *et al.* (2015), two different within-chromosome starting positions (starting at location 0 or 0.25 Mb for window size 0.5; starting at 0 or 0.5 Mb location for window sizes 1 Mb; starting at 0 or 1 Mb location for window sizes 2Mb; and starting at 0 or 1.5 Mb location for window sizes 3Mb) for each chromosome were chosen to partly counteract the chance effect of different LD patterns being associated with non-overlapping windows. Finally, adaptive window sizes based on clustering SNP by LD r^2 were also determined using the `BALD` R package (Dehman and Neuvial 2015) using the procedure described by Dehman *et al.* (2015).

The relative performance of all methods and models were based on receiver operating characteristic (ROC) curves. In a ROC curve, the true positive rate (TPR) is plotted against the false positive rate (FPR) for each competing method (Metz 1978). I were more specifically interested in the partial area under the curve up until a FPR= of 5% (pAUC05) so as to not include somewhat irrelevant ROC regions with low levels of specificity (Ma *et al.* 2013). A

perfect classifier would have a pAUC05 of $0.05 \times 1 = 0.05$ whereas a random classifier would have a pAUC05 of $0.05^2/2 = 0.00125$. I subsequently rescaled all pAUC05 measures by 0.00125^{-1} such that a random classifier is rescaled to a relative pAUC05 = 1. I used the R package `ROCR` (Sing *et al.* 2005) to obtain replicate-specific ROC curves and pAUC05 for each of the 10 replicated datasets for each method and window specification within each n_{qtl} and γ combination. For each window specification, specific comparisons between methods were based on using the logarithm of pAUC05 as the response variable in a mixed model ANOVA with methods, n_{qtl} and γ and all of their interactions included as fixed effects and population replicate (nested within n_{qtl} and γ) as a random effect blocking factor. For windows-based inferences based on fixed window sizes, replicate-specific pAUC05 values were averaged over the two different starting positions as previously noted. Mean log(pAUC05) estimates were backtranformed (i.e. anti-logged) to the original scale for reporting. Overall marginal means were separated using Tukey's test whereas comparisons between methods were sliced out using ANOVA *t*-tests for each value of n_{qtl} or γ if the corresponding interaction between these factors with methods were significant ($P<0.05$). I are also conjecture that window size might actually influence of the power of detecting QTL using the same method; therefore, I conducted separate tests comparing pAUC05 for each of the different window sizes, including adaptively chosen windows based on `BALD`, separately within each method.

3.4.2 MSUPRP data

I also compared all models and algorithms on 13-week tenth rib backfat (mm) within the MSUPRP data as per Gualdron Duarte *et al.* (2014). Sex, contemporary group, and age of slaughter were treated as fixed effects (i.e., β). I compared each of the six competing methods, computing either PPA or *P*-values in the same manner as in the simulation study. For MCMC-

BayesA and MCMC-SSVS, I ran a total of 1 million MCMC iterations based on 500,000 burn in iterations and 500,000 iterations post burn-in saving every 10 iterations such that posterior inference was based on 50,000 random draws from the posterior distribution. Since I did not know the true positions of the causal QTL for this trait, GWA inferences were compared between the various methods, based on PPA for MCMC-BayesA and MCMC-SSVS, P-values for RRBLUP, MAP-BayesA, and MAP-SSVS, and Bonferroni adjusted P-values for EMMAX. Note that no adjustment for multiple testing were invoked for *P*-values determined using the shrinkage based procedures (RRBLUP, MAP-BayesA, and MAP-SSVS) as per Gelman *et al.* (2012) whereas a Bonferroni adjustment based on the number of markers, or number of genomic windows for windows-based analyses, was invoked for EMMAX.

3.5 Results

3.5.1 Simulation Study

Overall mean comparisons between methods for pAUC05 based on single SNP inferences are provided in Table 3.1, noting that two-way interactions were not detected ($P > 0.05$) between methods with γ or with n_{qtl}. There was no evidence of a sizeable difference between any of the methods given that pAUC05 ranged from 2.52 to 2.77 times that for a random classifier, although MCMC-BayesA did rank lowest.

Table 3.1 Overall mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different methods on single SNP associations

	Methods					
	MCMC-SSVS	MCMC-BayesA	EMMAX	MAP-SSVS	MAP-BayesA	RRBLUP
Mean pAUC05	2.61[a, b]	2.52[b]	2.77[a]	2.69[a, b]	2.76[a]	2.73[a]

Values not sharing the same letter have different (P <0.05) relative pAUC05. Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (30, 100, or 300) of quantitative trait loci (QTL), and shape parameter (0.18,1.48, or 3.00) for Gamma distribution of QTL effects.

For fixed 1Mb window sizes (Table 3.2), the two-way interactions between method and γ and between method and n_{qtl} were both significant ($P < 0.0001$). Therefore, methods were compared separately for each different value of γ and of n_{qtl}. Nevertheless, MCMC-SSVS and MCMC-BayesA had the largest pAUC05 ($P < 0.05$) for each different value of γ and of n_{qtl} as well as overall. EMMAX generally followed MCMC-SSVS and MCMC-BayesA with MAP-SSVS, MAP-BayesA and RRBLUP being the worst performing methods. Most notably, these latter three methods generally did worse than a random classifier (i.e. pAUC05 < 1) except for MAP-SSVS at $n_{qtl} = 30$.

Table 3.2 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different methods for associations based on genomic windows of length 1Mb. Comparisons are made within different specifications of shape parameter (γ) for Gamma distribution of quantitative trait loci (QTL) and number of QTL (n_{qtl})

	Methods					
Factors	MCMC-SSVS	MCMC-BayesA	EMMAX	MAP-SSVS	MAP-BayesA	RRBLUP
Shape γ						
0.18	2.82[a]	2.75[a]	1.78[b]	0.74[c, *]	0.63[c, *]	0.48[d, *]
1.48	4.22[a]	4.16[a]	2.54[b]	0.69[c, *]	0.38[d, *]	0.28[e, *]
3	4.63[a]	5.01[a]	2.47[b]	0.67[c, *]	0.40[d, *]	0.24[e, *]
n_{qtl}						
30	6.81[a]	7.28[a]	3.63[b]	1.69[c]	0.67[d, *]	0.31[e, *]
90	3.89[a]	3.78[a]	2.14[b]	0.61[c, *]	0.47[c, d,]	0.37[d, *]
300	2.08[a]	2.08[a]	1.42[b]	0.33[c, *]	0.31[c, *]	0.29[c, *]

Overall	3.81[a]	3.86[a]	2.23[b]	0.70[c, *]	0.46[d, *]	0.32[e, *]

Values not sharing the same letter within a row have different (P <0.05) relative pAUC05 within the row. * indicates the corresponding method is not better than a random classifier. Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (30, 100, or 300) of quantitative trait loci (QTL), and shape parameter (0.18,1.48, or 3.00) for Gamma distribution of QTL effects.

Table 3.3 highlights the comparisons between the various methods using the adaptive window sizes inferred by BALD. Here, the two-way interaction between method and n_{qtl} was important ($P < 0.05$) whereas the two-way interaction between method and γ was not; hence, I just compared different methods within each different value of n_{qtl}. As with the 1Mb window inferences, MCMC-SSVS and MCMC-BayesA had the highest pAUC05, followed by EMMAX within each different value of n_{qtl} such that these same rankings were found overall as well. Again, I found that MAP-SSVS, MAP-BayesA, and RRBLUP had lower pAUC05 compared to a random classifier except for MAP-SSVS when $n_{qtl} = 30$.

Table 3.3 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different methods for associations based on genomic windows adaptively chosen by the BALD software package. Comparisons are made within different specifications of number of quantitative trait loci (n_{qtl})

	Methods					
n_{qtl}	MCMC-SSVS	MCMC-BayesA	EMMAX	MAP-SSVS	MAP-BayesA	RRBLUP
30	8.83[a]	9.03[a]	3.57[b]	1.76[c]	0.87[d, *]	0.29[e, *]
90	5.34[a]	4.98[a]	2.13[b]	0.8[c, *]	0.50[d, *]	0.38[d, *]
300	3.89[a]	3.17[a]	1.30[b]	0.66[c, *]	0.62[c, *]	0.58[c, *]
Overall	5.68[a]	5.22[a]	2.15[b]	0.97[c, *]	0.65[d, *]	0.40[e, *]

Values not sharing the same letter within a row have different (P <0.05) relative pAUC05 within the row. * indicates the corresponding method is not better than a random classifier. Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (30, 100,

or 300) of quantitative trait loci (QTL), and shape parameter (0.18,1.48, or 3.00) for Gamma distribution of QTL effects.

I was also interested in pAUC05 comparisons between different window length specifications. Recognizing that the interaction between method and window length was important in our joint analysis involving all simulated datasets, I choose to focus on window length comparisons separately within each of MCMC-SSVS, MCMC-BayesA, and EMMAX (Table 3.4), given that all other methods performed worse than random classifier with windows based inference. For EMMAX, single SNP inferences has significantly larger pAUC05

Table 3.4 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) between different window sizes within each of EMMAX, MCMC-BayesA, and MCMC-SSVS.

EMMAX		MCMC-BayesA		MCMC-SSVS	
Window	pAUC05	Window	pAUC05	Window	pAUC05
Single	2.76[a]	Single	2.52[c]	Single	2.61[c]
0.5Mb	2.43[a, b]	0.5Mb	3.77[b]	0.5Mb	3.65[b]
1Mb	2.23[b, c]	1Mb	3.86[b]	1Mb	3.81[b]
2Mb	1.95[c]	2Mb	3.93[b]	2Mb	3.94[b]
3Mb	1.85[c]	3Mb	3.93[b]	3Mb	4.04[b]
Adaptive	2.15[b, c]	Adaptive	5.22[a]	Adaptive	5.67[a]

Values not sharing the same letter within a column have different ($P < 0.05$) relative pAUC05 within the column. Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 x 5 factorials on number (30, 100, or 300) of quantitative trait loci (QTL), shape parameter (0.18,1.48, or 3.00) for Gamma distribution of QTL effects, and genomic region size (single SNP, 0.5Mb, 1Mb, 2Mb, 3Mb or adaptively determined) for genome wide association.

compared to inferences based on the longer genomic windows (2 and 3 Mb) with inference based on adaptively determined windows using `BALD` and shorter genomic windows (0.5Mb and 1Mb) being intermediate in their performance. Conversely, for both MCMC-BayesA and MCMC-SSVS, single SNP inference had the lowest pAUC05 whereas adaptively determined window selection based on `BALD` yielded the highest pAUC05 with fixed window inferences being

intermediate in their performance. In fact, the best overall performance was based on using the two MCMC based methods with adaptively determined windows with a pAUC05 being over 5 times greater than that of a random classifier.

3.5.2 MSUPRP Data

Manhattan plots based on single SNP associations for 13-week tenth rib backfat (mm) in MSUPRP are provided in Figure 3.2. The statistically most significant marker identified by EMMAX was SNP label ALGA0104402 (P = 2.36e-10) at location 136.0844Mb in Chromosome 6, marking the same location identified as being most significantly associated with this trait by Gualdron Duarte *et al.* (2014). Another 11 nearby statistically significant markers ranged in location from 132.60Mb to 138.24Mb with 1 marker (SNP label MARC0035827) at 122.36Mb on Chromosome 6 being also statistically significant using EMMAX. For MCMC-SSVS, the marker (SNP label ALGA0122657) located at 136.0786Mb on Chromosome 6 had the highest PPA of 0.487 and was adjacent to the most significant marker ALGA0104402 as identified by EMMAX. MCMC-SSVS also inferred its second largest PPA=0.227 with SNP marker ALGA0104402. Hence, the top 2 SNP markers identified by MCMC-SSVS and EMMAX were the same, albeit their order of importance was reversed. Although the most significant single SNP associations were also determined within this same region for each of the four other methods, their levels of significance were clearly not important except perhaps for MAP-SSVS which started to approach statistical significance with SNP label MARC0035827 (P=0.08).

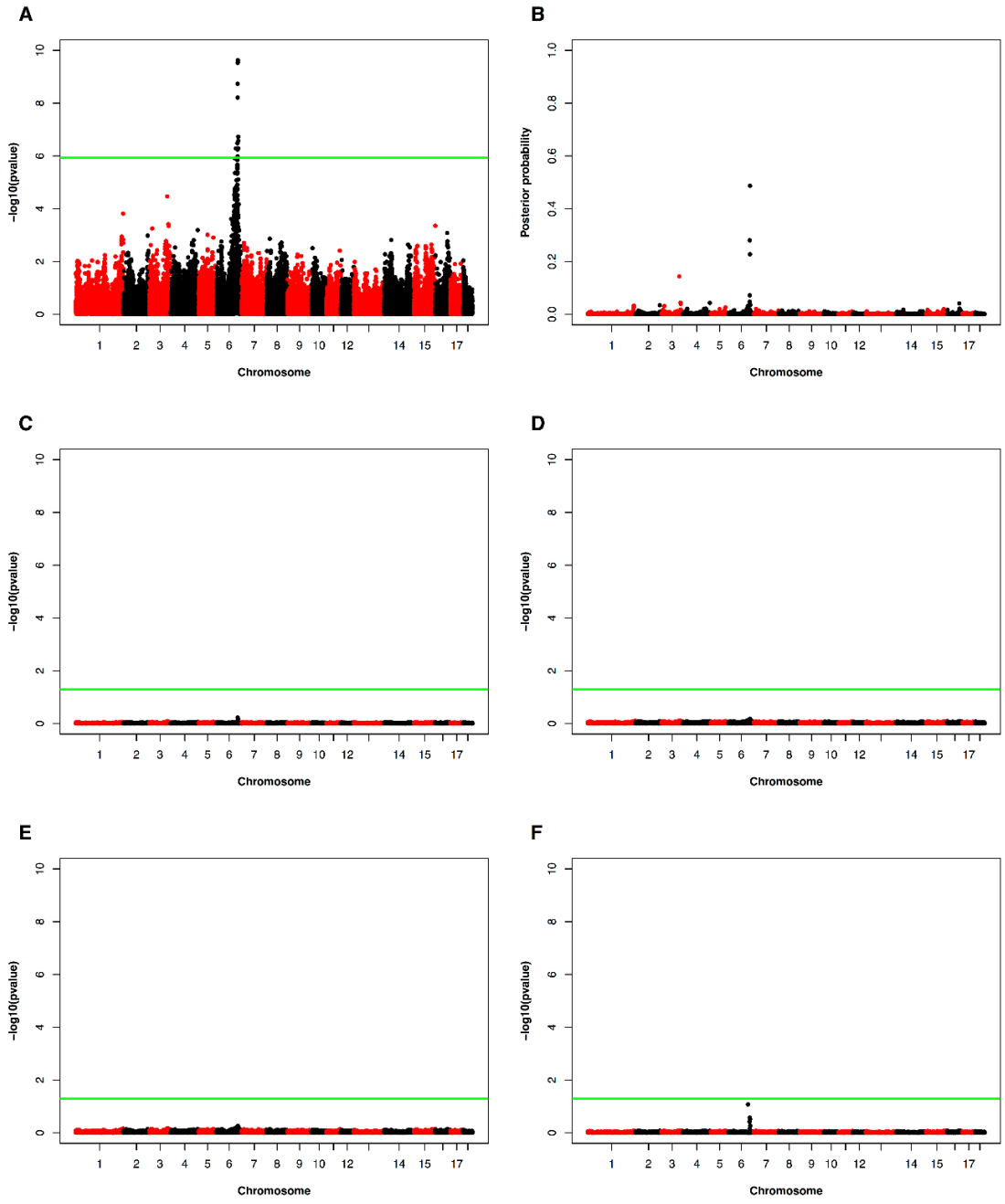

Figure 3.2 Manhattan plots for single SNP analysis on 13th week 10th rib backfat in Duroc Pietrain F2 cross (*n* = 922 pigs) based on different methods (Panel A: EMMAX, Panel B: MCMC-SSVS, Panel C: MCMC-BayesA, Panel D: RRBLUP, Panel E: MAP-BayesA and Panel F: MAP-SSVS)

For windows-based inference, I focused on the adaptively chosen window strategy based on LD using `BALD` (Figure 3.3). For EMMAX, the most significant window (P = 9.36e-08) ranged from 134.17Mb to 134.75Mb on Chromosome 6. Although this region did not contain any markers that were statistically significant based on single SNP based inferences, it was very close to a marker (SNP label ASGA0029653) at 134.14Mb that was deemed to be statistically significant in Figure 3.2. Four other windows on Chromosome 6 were also significant, covering regions 129.70-131.35Mb, 132.87-134.14Mb, 135.19-136.84Mb and 136.97-137.32Mb. These windows included some statistically significant or nearly significant markers based on single SNP inferences in Figure 3.2. Using MCMC-SSVS, the most significant window (Window 909) covered 135.19-136.84Mb with a PPA = 0.722; this window also contained the most significant markers based on single SNP inferences using EMMAX and MCMC-SSVS in Figure 3.2. Window 905 had the second highest PPA = 0.477 and ranged in location from 132.87-134.14Mb with all other windows having smaller PPA (< 0.2). A LD heatmap of the genomic region containing both windows are provided in Figure 3.4, indicating that some SNP markers in Window 905 are in relatively high LD with markers in Window 909. These two windows also had the highest PPA under MCMC-BayesA being 0.459 and 0.553 respectively. For RRBLUP, MAP-BayesA and MAP-SSVS, no window was deemed to be statistically significant (P>0.05).

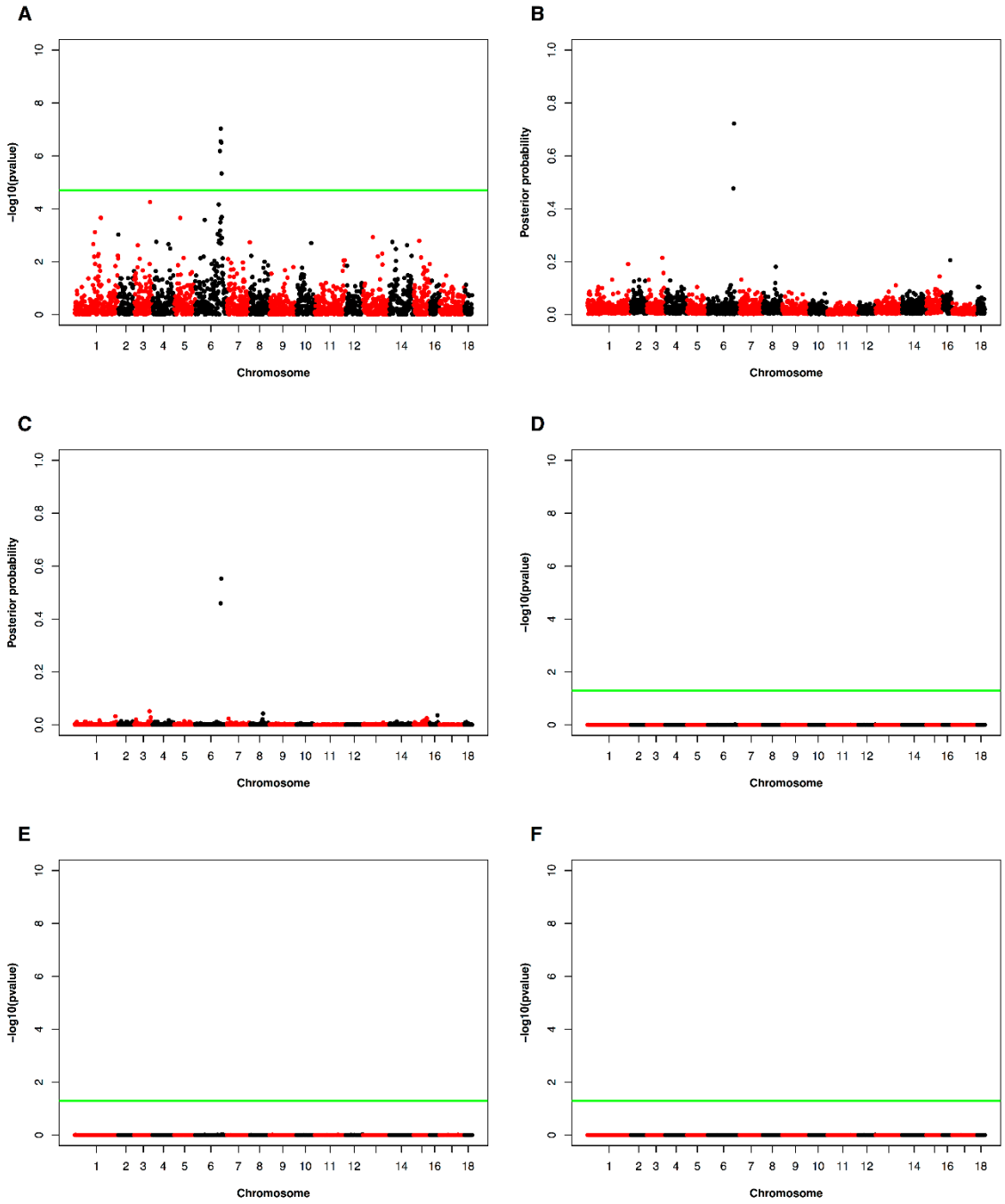

Figure 3.3 Manhattan plots for genomic window based associations on 13th week 10th rib backfat in Duroc Pietrain F2 cross (n = 922 pigs) based on different methods (Panel A: EMMAX, Panel B: MCMC-SSVS, Panel C: MCMC-BayesA, Panel D: RRBLUP, Panel E: MAP-BayesA and Panel F: MAP-SSVS) under adaptive window inference.

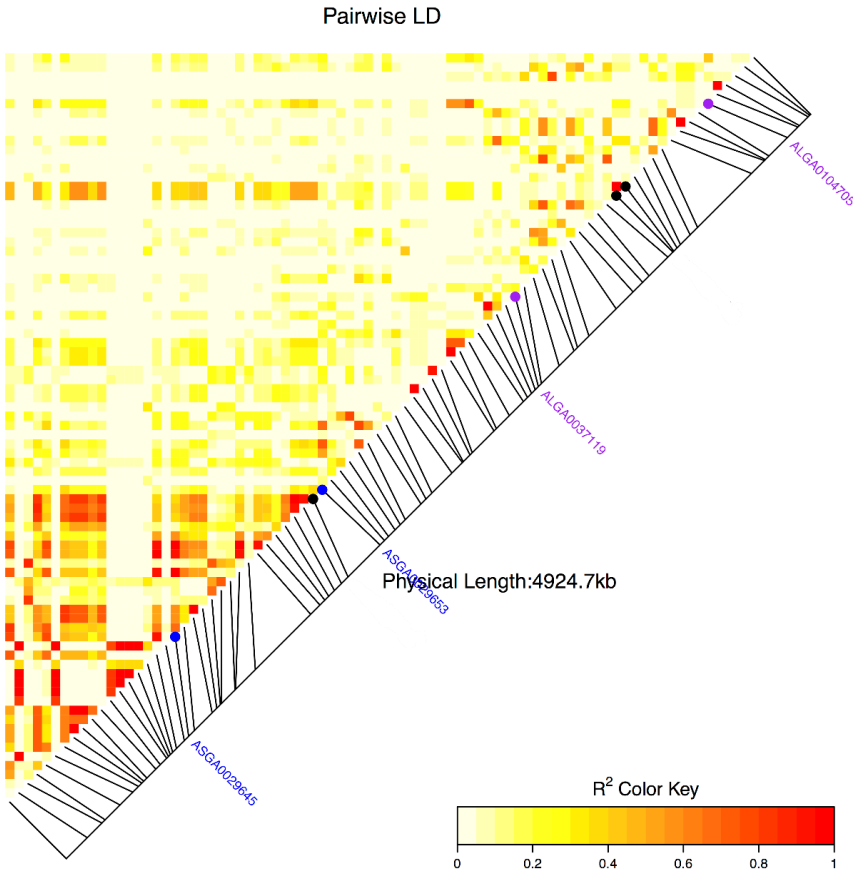

Figure 3.4 Linkage disequilibrium (r^2 metric) heatmap for genomic region containing Windows 905 - 909 on Chromosome 6 as adaptively determined by BALD software. Blue dots are starting and ending points for window 905 whereas purple dots are starting and ending points for window 909. Black dots are the 3 markers at 133.9292Mb, 136.0786Mb and 136.0844Mb that are top 3 SNPs by MCMC-SSVS. The blue oval is used to highlight a pocket of higher r^2 measures SNP markers in window 905 and 909.

3.6 Discussion

The objectives of our study were multifaceted in that I wished to very broadly address the impact of a) prior specifications on marker effects, b) single marker associations versus associations based on different specifications for genomic windows and c) of computationally tractable but analytical approximations for GWA inference based on sparse priors. Although our simulation study was based on genotypes derived from a specific population (MSUPRP), a wide

variety of potential genetic architectures were constructed on top of that framework based on different degrees of skewness of a Gamma distribution via alternative specifications of the shape parameter (γ) for QTL effects as well as different numbers (n_{qtl}) of QTL.

Most GWA studies have been conducted using single SNP inferences (Goddard and Hayes 2009; Visscher *et al.* 2012; Goddard *et al.* 2016). In this specific context, I determined that the difference in pAUC05 between all methods were relatively small and unimportant even though MCMC-BayesA had significantly lower pAUC05 and hence worse GWA performance. However, for all windows based analyses, MCMC-BayesA and MCMC-SSVS had significantly greater pAUC05 than all other methods across all combinations of γ and n_{qtl}, regardless of window size and whether these window sizes were fixed or adaptively inferred based on LD using the `BALD` software package. Conceptually, MCMC-BayesA might have even outperformed MCMC-SSVS for windows-based GWA as our comparisons may have been influenced by the arbitrariness of using 1% as a threshold for percentage of total genetic variance explained by a window when determining the PPA under MCMC-BayesA. That is, proper specification of such a threshold is likely to be density dependent. Admittedly, a BayesB like implementation (Meuwissen *et al.* 2001) could have captured the best features (i.e. variable selection and heavy-tailed priors) of both BayesA and SSVS. EMMAX typically ranked third whereas MAP implementations of BayesA and SSVS as well as RRBLUP did much more poorly for windows based association. The latter was not too surprising since previously Gualdron Duarte *et al.* (2014) also determined that RRBLUP was extremely conservative for GWA in this same dataset. Furthermore, this liability of RRBLUP has been noted by others including Hayes (2013). I noted that the median and mean lengths for windows adaptively chosen by `BALD` software were 0.59Mb and 0.91Mb (Panel A in Figure A1 in Appendix A), respectively, such

that it was reasonable to expect adaptively chosen windows to lead to an GWA performance closest to inferences based on either based on the 0.5Mb or 1Mb fixed window sizes as I did observe for the two MCMC based procedures.

What was initially surprising to us was that the pAUC05 for the analytical "shrinkage"-based procedures, namely RRBLUP, MAP-SSVS and MAP-BayesA, under windows based inference was often worse than that of a random classifier (i.e. pAUC05<1). This, at first, seemed counterintuitive to us. Hence, I briefly investigated a scenario where the number of SNP markers per window was fixed to be 10 rather than basing window sizes on a fixed physical distance. Basing genomic windows on a fixed number of SNPs has been a strategy also considered elsewhere (Zhang *et al.* 2016). In our particular case, the average length of a 10 SNP window was 0.51 Mb such that one might anticipate that inference based on 10 SNP marker windows might be comparable to using inference based on fixed 0.5 Mb length windows. Nevertheless, I determined that 10 SNP windows based inference lead to a ROC performance that was at least as good as a random classifier for each of RRBLUP, MAP-SSVS and MAP-BayesA (Figure A2 in Appendix A), conversely to what I observed previously to windows based on any fixed physical distance. This contrast in pAUC05 performance between fixed physical distance and fixed number of markers could be explained as follows. For the vast majority of windows based on either scenario (fixed number of markers or fixed physical distance), the P-values for the chi-square tests of these shrinkage based procedures were very large (i.e., $P > 0.85$). With inference based on a fixed number of SNP markers per window and random assignment of QTL to these markers, it was reasonable to expect that the pAUC05 of any of these procedures should be at least as large as a random classifier. However, with inference based on fixed physical distance in Mb or even adaptively determined based on LD relationships,

the number of SNP markers and hence the degrees of freedom for each window-specific chi-square test was highly variable, ranging from 1 to 35 with 0.5Mb windows, for example. Hence regions with few markers are more likely to have smaller P-values than regions with many markers by nature of a greater penalty incurred with a larger degrees of freedom chi-square test statistic. Furthermore, lower P-value regions with fewer markers are also less likely to contain a QTL because of random assignment of QTL to markers throughout the genome such that regions with the smallest P-values would more likely include a greater than expected number of false positive results relative to a random classifier.

One possible strategy to mitigate this problem is through use of a likelihood ratio test for the variance component characterizing the variance attributable to markers within a window can be considered for EMMAX or the MAP based approaches as then the degrees of freedom for that test does not depend on the number of markers in that window (Wu *et al.* 2010; Wang *et al.* 2013). Gualdron Duarte *et al.* (2014) present details for such a likelihood ratio test; nevertheless, this approach requires one to refit the entire model each time that a particular window is being tested and hence can be computationally challenging.

I specifically determined that adaptive window specifications based on `BALD` worked best for both MCMC-BayesA and MCMC-SSVS with significantly higher mean pAUC05 than inferences based on fixed window lengths or single SNP markers. In fact, there was no evidence of differences in pAUC05 between GWA associations based on windows of constant sizes ranging from 0.5 to 3Mb when using either MCMC-BayesA or MCMC-SSVS. Hence adaptive window clustering based on LD measures seems to be an important factor to consider when partitioning genomic windows, at least for Bayesian sparse prior specifications.

I have previously established that starting values are important for MAP-SSVS and MAP-BayesA (Chen and Tempelman 2015); in fact, I then demonstrated that starting marker effects at null values was very suboptimal, even though that is a common strategy for genomic prediction methods based on the use of the EM algorithm (Meuwissen *et al.* 2009; Karkkainen and Sillanpaa 2012). As I adapted in this study, a practical strategy is to base starting values on RRBLUP and genomic REML as I conducted in this study although I worried as to how suboptimal that might be, recognizing MAP estimates are asymptotic i.e., $\text{MAP}(\mathbf{g}) \rightarrow \text{E}(\mathbf{g}|\mathbf{y})$ only as $n \rightarrow \infty$ and such that $n >> m$. To further assess whether starting values based on RRBLUP and genomic REML estimates might lead to suboptimal GWA inferences, I also based starting values for MAP-SSVS and MAP-BayesA on posterior mean estimates derived from their MCMC counterparts, focusing only, however, on single SNP and adaptive window inference. I recognize that this would not be a practical MAP strategy as once MCMC based inferences are obtained, then asymptotic MAP based inferences would not have any extra value. As anticipated from our previous genomic prediction work (Chen and Tempelman 2015), using MCMC based starting values for MAP-SSVS lead to a larger pAUC05 compared to the use of RRBLUP or genomic REML starting values except for no evidence of a difference at n_{qtl} = 300 (Table A5 in Appendix A). However, for adaptively determined windows, even MAP-SSVS inferences based on MCMC based starting values were no better than a random classifier except for when n_{qtl} =30. Similar results for comparing different sets of starting values (MCMC-BayesA vs BLUP) for MAP-BayesA are provided in Table A6 in Appendix A. These supplementary results further illustrate how precarious is the use of MAP based procedures for Bayesian regression GWA analyses; again, I would believe the sensitivity of MAP to starting values would only be greater with the use of high density marker panels.

As our GWA inference for MCMC-SSVS was based on PPA (i.e. Prob(τ_j = 1|**y**)), it might seem reasonable to specify GWA inference for MAP-SSVS in a similar manner; i.e,, using the E-step values of τ_j at convergence as estimates of PPA. However, I noted that these E-step values uniformly drifted either towards 0 or 1 such that there were never any intermediate estimates of PPA. A comparison of PPA based on τ_j for Prob(τ_j =1|**y**) for MCMC-SSVS versus the E-step values of τ_j at convergence on the MSUPRP data is provided is given in Panel A of Figure A3 in Appendix A. Also, recall that the MAP-procedure is sensitive to starting values and that starting values for MAP-SSVS were based on RRBLUP as this might be a pragmatic and reasonable strategy in most cases. If I had based starting values on, say, their MCMC-SSVS posterior means, one would notice a different assortment of converged E-step values of τ_j compared to what I observed with RRBLUP starting values as I demonstrate with the MSUPRP data in Panel B of Figure A3 (Appendix A).

Recall that for MAP-SSVS, I based starting values for the SNP specific PPA on estimated local false discovery rates (lFDR) using the R package `ashr` since there is presumably a close relationship between them; i.e., PPA ≈ 1- lFDR (Stephens 2017). This procedure converts *P*-values to lFDR such that I based lFDR determinations from the *P*-values computed under EMMAX. This begged the question as to whether PPA could be simply based on lFDR processing of EMMAX P-values. However, upon comparing 1-lFDR estimated from the EMMAX *P*-values to PPA estimated using MCMC-SSVS of the MSUPRP data, it appeared that there was not generally very good agreement between the two sets of PPA estimates except for the some near-zero PPA and the largest PPA estimated using both procedures (Figure A4 in Appendix A).

I also wondered if the strategy for computing window-based PPA could be simplified further from that presented in Fernando *et al.* (2017) and used in this paper (i.e., Equation [3.19]) to that suggested by Moser *et al.* (2015) who simply summed SNP specific PPA (i.e., based on Equation [3.15]) within a window to determine the window-based PPA. One should anticipate that the approach of Moser *et al.* (2015) should lead to higher estimated PPA. I compared the two PPA determination approaches for pAUC05 in the simulation study and noted that there was significant interaction between PPA determination approach with γ and n_{qtl} but no significant interaction involving window size; hence I compared the two strategies within each value of γ and n_{qtl} averaged across window length (Table A7 in Appendix A). The only significant difference in pAUC05 occurred with γ=3 and n_{qtl} =300 for which the approach of Fernando *et al.* (2017) led to a higher pAUC05. Nevertheless, since point estimates of pAUC05 were always larger using the approach from Fernando *et al.* (2017) I would recommend their approach from Equation [3.19] for the determination of windows based PPA. Excellent analytical discussion on control of false positives in GWA using PPA is further provided in Fernando *et al.* (2017).

I did not estimate v_τ using either the procedures outlined in Yang *et al.* (2015b) for MCMC-BayesA or provided in Chen and Tempelman (2015) for MAP-BayesA primarly because of the extremely poor MCMC mixing for sampling this hyperparameter and its poor convergence in MAP-BayesA. A typical specification for v_τ in BayesA is 4 or 5 (Colombani *et al.* 2013; Perez and de los Campos 2014). The specification of $v_\tau = 2.5$ that I chose for this paper was based in part on results from Yang *et al.* (2015b) and Nadaf *et al.* (2012) who determined that lower specifications of v_g (i.e., heavier tails) could lead to higher genomic prediction accuracies when using Bayes A. To assess this further, I compared MCMC-BayesA using $v_\tau = 2.5$ versus $v_\tau = 5$

for pAUC05 based on the BALD derived adaptive window inference. In general, the use of $v_\tau = 2.5$ yielded a higher mean pAUC05 than $v_\tau = 5$ except for a non-significant difference at n_{qtl}=300 (Table A4). For large scale empirical analyses whereby hyperparameter inference seems daunting, researchers should consider conducting analyses based on a finite number of hyperparameter specifications, choosing those specifications that lead to the best cross-validation prediction accuracy. Similar arguments could be made for choosing the key hyperparameters in other Bayesian regression models. It is worth noting that even I ran our MCMC algorithm for 1 million iterations, the mixing of the MCMC chain was still rather poor as it pertained to inference on other hyperparameters. For example, for MCMC-BayesA, the effective sample size (ESS) for σ_g^2 was estimated to be 66.33 whereas for SSVS, the ESS was 61.03 for σ_g^2 and 53.48 for π_τ.

It should be apparent that given that MCMC-SSVS is a natural variable selection model, it might be favored over MCMC-BayesA which is not a natural variable selection model. Our strategy for computing the proportion of genetic variance explained by each window and determining the posterior probability that that percentage exceeds an arbitrary threshold (1% in our analyses) is based on the strategy presented by Fernando and Garrick (2013). The flexibility of MCMC modeling allows posterior probabilities (i.e., PPA) of this nature to be computed. However, one should be wary of the impact of the threshold since it obviously should depend upon marker density. That is, if the threshold is set too high, then sensitivity is lost. Based on the results from both simulation study and real data analysis, I demonstrated that random effects modeling can also be powerful tool for GWA as long as the suitable priors, i.e., in our case sparser priors, are used. Other variable selection implementations popularized in WGP including

BayesB (Meuwissen *et al.* 2001) or BayesR (Erbe *et al.* 2012; Moser *et al.* 2015) could be considered as well.

Our MSUPRP application was interesting in that I discovered that SNP markers in two different blocks can be in high LD even when they're not adjacent to each other. However, I would quickly note that these strange LD patterns may be due to genome assembly errors in the pig genome (Groenen 2016) with particular issues having been identified in the Chromosome 6 region (Warr *et al.* 2015) which contained the strongest associations in our study. This may somewhat complicate strategies for single SNP specific or even window-based inference. I also recognize that there is a movement towards the use of multi-SNP haplotype modeling which may improve GWA performance (Cuyabano *et al.* 2014). Our adaptive window based strategy seems to improve the performance of GWA relative to single SNP or fixed window length inference although, conceivably, there may be other better ways to group SNPs. With marker densities well beyond 50K, the adaptive window strategy might not be viable since it requires the computation and storage of matrix of LD r^2 values between every SNP marker within a chromosome before clustering analyses can be used to partition the genome into windows. Fernando *et al.* (2017) also suggested that PPA based on Bayesian GWA analyses similar to our MCMC-SSVS be based on whether non-zero associations were found not only in that marker's resident window but also in either of the two flanking windows. Their strategy was based on fixed window sizes such that it may be worthwhile to consider their flanking strategy in the context of adaptively chosen window sizes. I conjecture that if LD structure is appropriately used to partition the genome, the use of such flanking windows might not be necessary; however, this should be a topic for future research. It is also important to note that the comparisons in this paper are context specific in terms of the genomic LD relationships germane to a F2 cross in

pigs. This cross naturally leads to a higher pairwise LD between adjacent SNP markers than what might be found in outbreeding populations, and most notably humans. Different LD patterns would naturally change the relative comparisons between single SNP versus windows based inferences as well as the relative number and sizes of adaptively chosen windows based on LD relationships. Hence future investigation of our approaches in other populations is strongly warranted.

In summary, I found Bayesian variable selection to be a promising strategy for GWA when combined with window based inference. Nevertheless, it seems prudent that window selection be carefully chosen using rules based on LD information rather than predetermined constant physical window lengths (in Mb) for genomic regions. Also, recently proposed analytical approaches for Bayesian regression models should be discouraged for GWA studies.

Chapter4 Hierarchical Whole-Genome Prediction and Genome-Wide Association Modeling When Some Genotypes Are Missing

4.1 Abstract

Single step Genomic Best Linear Unbiased Prediction (ssGBLUP) has become increasingly popular for whole genome prediction (WGP) modeling as it utilizes any available pedigree and phenotypes on both genotyped and non-genotyped individuals. The WGP accuracy of ssGBLUP has been previously demonstrated to be higher than or equivalent to conventional Bayesian regression models. However, most of these assessments have typically not included phenotypes on non-genotyped individuals in Bayesian regression analyses, making the interpretation of these comparisons difficult. Recently, ssGBLUP has been increasingly used for genome-wide association (GWA) studies although there has been no clear guidance on how to determine formal statistical evidence of association in these analyses. I address this problem as well as propose a GWA based on a single step adaptation of Bayesian stochastic search and variable selection (ssSSVS) model that also incorporates phenotypes on non-genotyped animals. Our study was based on a dataset including 3186 Holstein cows from 6 US research stations using the USDA-ARS Bovine 60671 SNP marker panel as genotypes. In a simulation study based on the use of these same genotypes, a different number of causal variants (n_c = 30, 300, or 3000) were randomly assigned to the markers, masking 50% of cows as non-genotyped, for a trait having a heritability of 0.25. I determined that ssSSVS had a greater (P<0.05) WGP accuracy than ssGBLUP with simpler genetic architectures (n_c = 30 or n_c = 300). Moreover, ssSSVS always had better (P<0.05) GWA performance than ssGBLUP as based on the partial area under a receiver operating characteristic curve up until a false positive rate of 5%. In a 25-fold within station cross-validation study using phenotypes from the dairy consortium, I determined that

ssSSVS had, albeit slightly, greater (P<0.05) WGP accuracies in milkfat compared to ssGBLUP for genotyped individuals, whereas no such differences were detected for body weight. I also found no significant differences between ssSSVS and ssGBLUP for WGP accuracies for non-genotyped individuals for both traits. Overall, I find ssSSVS to be a promising method for both WGP and GWA, particularly for genetic architectures characterized by a few genes with large effects.

4.2 Introduction

Whole genome prediction (WGP) or genomic selection using dense marker panels has been increasingly implemented in animal and plant breeding and in human genetics studies. Two broad categories of models have been particularly popular for WGP analyses, namely Genomic Best Linear Unbiased Prediction (GBLUP) analysis and Bayesian regression or "Bayesian alphabet" (Gianola 2013) analysis. GBLUP is based on traditional linear mixed model inference whereby a genomic relationship matrix based on single nucleotide marker (SNP) marker genotypes is used to specify the correlation between random animal effects (VanRaden 2008). On the other hand, Bayesian regression models are typically more flexible with distributional specifications for SNP effects based on heavy tailed prior distributions like a scaled Student *t* (i.e., BayesA) (Meuwissen *et al.* 2001) or variable selection specifications such as stochastic search and variable selection (SSVS) (George and McCulloch 1993; Chen and Tempelman 2015). These Bayesian regression approaches been demonstrated to achieve higher WGP accuracies in many different applications (de Los Campos *et al.* 2013).

A single-step GBLUP (ssGBLUP) approach has been used to describe a procedure that combines phenotypes on genotyped animals and on non-genotyped animals with pedigree information (Aguilar *et al.* 2010) has been successfully applied to several livestock species

(Chen *et al.* 2011b; Gray *et al.* 2012; Lourenco *et al.* 2015). Because of the additional utilization of the phenotypic and pedigree information on non-genotyped individuals, ssGBLUP has been found to have higher WGP accuracies than one popular Bayesian alphabet model, BayesC (Lourenco *et al.* 2013; Legarra *et al.* 2014; Vallejo *et al.* 2016). However, this comparison might be unfair as most of these Bayesian analyses have not been extended to use phenotypes on non-genotyped individuals; furthermore, these comparisons may be sensitive to arbitrary specifications of some key hyperparameters, most notably the proportion of SNP effects deemed to be non-zero. Recently, Fernando *et al.* (2014) proposed a single-step approach for Bayesian alphabet models that combine information on both genotyped and non-genotyped individuals, later following up on that work with computational strategies for implementation with large livestock datasets (Fernando *et al.* 2016).

Genome-wide association (GWA) analysis is a useful tool to identify genomic regions containing putative causal variants or quantitative trait loci (QTL). Currently, popular tools such as EMMAX simultaneously fit all markers using the linear mixed model to account for population structure (Kang, 2010). Although EMMAX and somewhat related analyses have been demonstrated to increase the power of QTL detection compared to single-marker based regression (Kang *et al.* 2008), these analyses have not typically utilized phenotypic information on non-genotyped individuals. Wang *et al.* (2012) and Zhang *et al.* (2016) have demonstrated how to adapt ssGBLUP for GWA; however, their GWA assessments were not based on formal measures of statistical significance but merely point estimates of SNP estimates or percentage of genetic variance explained by sliding windows of SNP markers. This latter development is particularly important if one is interested in a fair assessment of whether flexible Bayesian

specifications might have merit over ssGBLUP for both WGP and GWA applications when some phenotyped animals are not genotyped.

GWA studies are increasingly based on joint tests on SNP markers within pre-defined genomic windows rather than just tests on single SNP marker as single SNP marker tests may have low statistical power or be adversely affected by multicollinearity or both (Chen *et al.* 2017; Fernando *et al.* 2017). Most studies have based their window using arbitrarily selected window sizes or number of markers (Wolc *et al.* 2012; Moser *et al.* 2015) whereas Dehman *et al.* (2015) proposed to adaptively cluster SNP markers into windows of varying size based on LD structure. Chen *et al.* (2017) demonstrated that GWA inferences based on an adaptive window approach enhance the GWA performance of Bayesian models, such as SSVS and BayesA compared to GWA associations derived from window sizes of arbitrary length or single SNP inferences.

I had several objectives in this study. The first objective was to present a single-step SSVS (ssSSVS) Bayesian strategy for WGP in conjunction with GWA. A second objective was to demonstrate formal *P*-value inference for a single-step extension of EMMAX or GBLUP (ssGBLUP) that allows use of phenotypes on non-genotyped individuals for GWA based single SNP based test as well as adaptive window approach. The last objective was to compare the performance of ssSSVS and ssGBLUP for WGP and GWA in both simulation study and real data analysis.

4.3 Methods and materials

4.3.1 The hierarchical linear model

Following Fernando *et al.* (2014) the linear model including genotyped and non-genotyped individual can be presented as in Equation [4.1]:

$$\begin{bmatrix} \mathbf{y}_n \\ \mathbf{y}_g \end{bmatrix} = \begin{bmatrix} \mathbf{X}_n \\ \mathbf{X}_g \end{bmatrix} \boldsymbol{\beta} + \begin{bmatrix} \mathbf{Z}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{Z}_g \end{bmatrix} \begin{bmatrix} \mathbf{u}_n \\ \mathbf{u}_g \end{bmatrix} + \mathbf{e} \qquad [4.1]$$

Here $\mathbf{y}_n$ and $\mathbf{y}_g$ are n_n x 1 and n_g x 1 vectors of phenotypes of non-genotyped and genotyped individuals, respectively. Also $\boldsymbol{\beta}$ is a vector of fixed environmental effects, with $\mathbf{X}_n$ and $\mathbf{X}_g$ being the corresponding incidence matrices on the non-genotyped and genotyped individuals, respectively. Furthermore, $\mathbf{u}_n$ and $\mathbf{u}_g$ are, respectively, q_n x 1 and q_g x 1 vectors of breeding values of non-genotyped and genotyped individuals, with $\mathbf{Z}_n$ and $\mathbf{Z}_g$ being the corresponding respective incidence matrices.

Now, breeding values can, in turn, be written as linear functions of SNP effects per Fernando *et al.* (2014) in Equation [4.2].

$$\begin{bmatrix} \mathbf{u}_n \\ \mathbf{u}_g \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{T}}_n\boldsymbol{\alpha} + \boldsymbol{\varepsilon} \\ \mathbf{T}_g\boldsymbol{\alpha} \end{bmatrix} \qquad [4.2]$$

Here $\boldsymbol{\alpha}$ is a m x 1 vector of random SNP marker effects that $\boldsymbol{\alpha} \sim N\left(\mathbf{0}, \mathbf{D}\sigma_\alpha^2\right)$ with $\mathbf{D}$ being a weighting matrix as described later. Furthermore, $\mathbf{T}_g$ is a standardized genotype matrix for genotyped individuals such that

$$\mathbf{T}_g = \frac{\left(\mathbf{M}_g - \mathbf{1}\mathbf{k}'\right)}{\sqrt{\sum_{j=1}^{m} 2p_j\left(1-p_j\right)}} \qquad [4.3]$$

where $\mathbf{M}_g$ is the original n_g x m genotype matrix with elements coded as "0, 1, 2" or the number of copies of the reference allele of the SNP marker within the corresponding column of $\mathbf{M}_g$. Furthermore, element j of the m x 1 vector $\mathbf{k}$ is the mean value ($2p_j$) for the corresponding column of $\mathbf{M}_g$, such that p_j is the allele frequency of the reference allele of SNP marker j =1,2,…,m (VanRaden 2008). Conversely, $\hat{\mathbf{T}}_n$ in Equation [4.2] is an "imputed" genotype matrix for the non-genotyped individuals. As demonstrated by Fernando et al., (2014), $\hat{\mathbf{T}}_n$ can be obtained by solving

$\mathbf{A}^{nn}\hat{\mathbf{T}}_n = -\mathbf{A}^{ng}\mathbf{T}_g$, where $\mathbf{A}^{nn}$ and $\mathbf{A}^{ng}$ are the partitions of $\mathbf{A^{-1}}$ corresponding to non-genotyped by non-genotyped and non-genotyped by genotyped animals, respectively. Finally, the imputation residuals $\boldsymbol{\varepsilon} \sim N\left(\mathbf{0}, (\mathbf{A}^{nn})^{-1}\sigma_u^2\right)$ in Equation [4.2] are the contributions of pedigree information to breeding values for non-genotyped animals as demonstrated by Fernando et al. (2014).

Combining Equations [4.1] and [4.2], a SNP effects model can be written as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{W}\boldsymbol{\alpha} + \mathbf{U}\boldsymbol{\varepsilon} + \mathbf{e} \qquad [4.4]$$

where $\mathbf{y} = \begin{bmatrix} \mathbf{y}_n \\ \mathbf{y}_g \end{bmatrix}$, $\mathbf{X} = \begin{bmatrix} \mathbf{X}_n \\ \mathbf{X}_g \end{bmatrix}$, $\mathbf{W} = \begin{bmatrix} \mathbf{W}_n \\ \mathbf{W}_g \end{bmatrix} = \begin{bmatrix} \mathbf{Z}_n\hat{\mathbf{T}}_n \\ \mathbf{Z}_g\mathbf{T}_g \end{bmatrix}$, and $\mathbf{U} = \begin{bmatrix} \mathbf{Z}_n \\ \mathbf{0} \end{bmatrix}$. Then the corresponding mixed model equations used to compute the BLUE, $\hat{\boldsymbol{\beta}}$, of $\boldsymbol{\beta}$, BLUP $\hat{\boldsymbol{\alpha}}$ of $\boldsymbol{\alpha}$ and BLUP $\hat{\boldsymbol{\varepsilon}}$ of $\boldsymbol{\varepsilon}$, as also illustrated by Fernando et al. (2014), is given in Equation [4.5]

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{W} & \mathbf{X}_n'\mathbf{Z}_n \\ \mathbf{W}'\mathbf{X} & \mathbf{W}'\mathbf{W} + \mathbf{D}^{-1}\dfrac{\sigma_e^2}{\sigma_\alpha^2} & \mathbf{W}_n'\mathbf{Z}_n \\ \mathbf{Z}_n'\mathbf{X}_n & \mathbf{Z}_n'\mathbf{W}_n & \mathbf{Z}_n'\mathbf{Z}_n + \mathbf{A}^{nn}\dfrac{\sigma_e^2}{\sigma_u^2} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{\alpha}} \\ \hat{\boldsymbol{\varepsilon}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{W}'\mathbf{y} \\ \mathbf{Z}_n'\mathbf{y}_n \end{bmatrix} \qquad [4.5]$$

4.3.2 The ssGBLUP model

When **D**=**I**, then the elements of $\boldsymbol{\alpha}$ are normally, identically and independently distributed, such that the corresponding analysis is single-step (ss) adaptation of GBLUP which I denote as ss-GBLUP. When the total number of animals $q = q_n + q_g$ is considerably smaller than the total number of SNP markers *m*, I believe it is convenient to re-parameterize Equation [4.4] further to Equation [4.6].

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u}^{(\alpha)} + \mathbf{U}\boldsymbol{\varepsilon} + \mathbf{e} \qquad [4.6]$$

where $\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{Z}_g \end{bmatrix}$, and $\mathbf{u}^{(\alpha)} = \mathbf{T}\boldsymbol{\alpha} = \begin{bmatrix} \mathbf{T}_n \\ \mathbf{T}_g \end{bmatrix}\boldsymbol{\alpha} = \begin{bmatrix} \mathbf{u}_n^{(\alpha)} \\ \mathbf{u}_g^{(\alpha)} \end{bmatrix}$ is the contribution of genotypes to breeding values whether based on actual genotypes ($\mathbf{T_g}$) for breeding values on genotyped animals or based on imputed genotypes ($\hat{\mathbf{T}}_n$) for non-genotyped animals. With $\mathbf{u}^{(\alpha)} = \mathbf{T}\boldsymbol{\alpha}$, then $\text{var}(\mathbf{u}^{(\alpha)}) = \mathbf{T}\mathbf{T}'\boldsymbol{\alpha}$ such that the corresponding mixed model equations can be written as follows to solve for the BLUE of $\boldsymbol{\beta}$ and the BLUP of $\mathbf{u}^{(\alpha)}$ and $\boldsymbol{\varepsilon}$ in Equation [4.7]

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z} & \mathbf{X}'\mathbf{U} \\ \mathbf{X}'\mathbf{Z} & \mathbf{Z}'\mathbf{Z} + \left(\mathbf{T}\mathbf{T}'\right)^{-1}\frac{\sigma_e^2}{\sigma_\alpha^2} & \mathbf{Z}'\mathbf{U} \\ \mathbf{X}'\mathbf{U} & \mathbf{U}'\mathbf{Z} & \mathbf{U}'\mathbf{U} + \mathbf{A}^{nn}\frac{\sigma_e^2}{\sigma_u^2} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\mathbf{u}}^{(\alpha)} \\ \hat{\boldsymbol{\varepsilon}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{Z}'\mathbf{y} \\ \mathbf{U}'\mathbf{y} \end{bmatrix} \qquad [4.7]$$

Note that σ_α^2, σ_u^2, and σ_e^2 can be readily estimated using REML; in fact, I adopt the average information restricted maximum likelihood (AIREML) algorithm (Gilmour *et al.* 1995; Johnson and Thompson 1995) to estimate the variance components σ_α^2, σ_u^2 and σ_e^2. Subsequently, $\hat{\boldsymbol{\beta}}, \hat{\mathbf{u}}^{(\alpha)}$, and $\hat{\boldsymbol{\varepsilon}}$ can be obtained by solving MME in Equation [4.7] conditional on these REML estimates. I label this strategy that separately estimates a common marker variance component σ_α^2 from the polygenic variance component σ_u^2 as the heterogeneous variance (HETVAR) approach, recognizing that these two variance components could be different from each other in real data applications if the markers do not capture all of the genetic variability or highly selected animals or animals from certain herds or stations are preferentially genotyped such that $\sigma_\alpha^2 \neq \sigma_u^2$. Furthermore, estimating the two variance components separately might be a better solution if the pedigree based relationship matrix and scaled genomic relationship matrix are not completely compatible with each other (Chen *et al.* 2011a).

The HETVAR approach differs from the more traditional ssGBLUP approach described in Aguilar *et al.* (2010) where it is implicitly assumed that $\sigma_\alpha^2 = \sigma_u^2$ This specification, which I label as HOMVAR, simplifies the MME in Equation [4.7] to that in Equation [4.8]

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z} & \mathbf{X}'\mathbf{U} \\ \mathbf{X}'\mathbf{Z} & \mathbf{Z}'\mathbf{Z}+\left(\mathbf{T}\mathbf{T}'\right)^{-1}\frac{\sigma_e^2}{\sigma_u^2} & \mathbf{Z}'\mathbf{U} \\ \mathbf{X}'\mathbf{U} & \mathbf{U}'\mathbf{Z} & \mathbf{U}'\mathbf{U}+\mathbf{A}^{nn}\frac{\sigma_e^2}{\sigma_u^2} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\mathbf{u}}_\alpha \\ \hat{\boldsymbol{\varepsilon}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{Z}'\mathbf{y} \\ \mathbf{U}'\mathbf{y} \end{bmatrix} \quad [4.8]$$

Similarly, σ_u^2 and σ_e^2 can be estimated using REML. Essentially, the MME in Equation [4.8] is equivalent to the following MME based on $\mathbf{H}^{-1}$ (Aguilar *et al.* 2010):

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z} \\ \mathbf{Z}'\mathbf{X} & \mathbf{Z}'\mathbf{Z}+\mathbf{H}^{-1}\lambda \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{Z}'\mathbf{y} \end{bmatrix}; \lambda = \frac{\sigma_e^2}{\sigma_u^2} \quad [4.9]$$

where

$$\mathbf{H}^{-1} = \mathbf{A}^{-1} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \left(\mathbf{T}_g\mathbf{T}_g'\right)^{-1} - \mathbf{A}_{gg}^{-1} \end{bmatrix}$$

If I partition $\mathbf{H}^{-1}$ as $\mathbf{H}^{-1} = \begin{bmatrix} \mathbf{H}^{nn} & \mathbf{H}^{ng} \\ \mathbf{H}^{gn} & \mathbf{H}^{gg} \end{bmatrix}$ into components due to non-genotyped (*n*) and genotyped (*g*) animals, then the MME in Equation [4.9] can be further rewritten as

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z}_n & \mathbf{X}'\mathbf{Z}_g \\ \mathbf{X}'\mathbf{Z} & \mathbf{Z}_n'\mathbf{Z}_n+\mathbf{H}^{nn}\lambda & \mathbf{Z}_n'\mathbf{Z}_g+\mathbf{H}^{ng}\lambda \\ \mathbf{X}'\mathbf{Z}_g & \mathbf{Z}_n'\mathbf{Z}_g+\mathbf{H}^{gn}\lambda & \mathbf{Z}_g'\mathbf{Z}_g+\mathbf{H}^{gg}\lambda \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta} \\ \hat{\mathbf{u}}_n \\ \hat{\mathbf{u}}_g \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{Z}_n'\mathbf{y}_n \\ \mathbf{Z}_g'\mathbf{y}_g \end{bmatrix} \quad [4.10]$$

Equations [4.8], [4.9] and [4.10] are based on equivalent linear models (Henderson 1975) and hence lead to identical estimated breeding values $\hat{\mathbf{u}}_g$ for the genotyped individuals whereas REML estimates for σ_u^2 and σ_e^2 are also identical. For non-genotyped individuals, the mixed

model equations (MME) in Equation [4.9] estimate the breeding values using imputed genotypes $\hat{\mathbf{u}}_n^{(\alpha)}$ and pedigree information in $\hat{\boldsymbol{\varepsilon}}$ separately, i.e. $\hat{\mathbf{u}}_n = \hat{\mathbf{u}}_n^{(\alpha)} + \hat{\boldsymbol{\varepsilon}}$.

To test $H_0 : \sigma_\alpha^2 = \sigma_u^2$, one can conduct a likelihood ratio test with l_0 being the maximized restricted log likelihood under the reduced HOMVAR model and l_1 being the maximized restricted log likelihood under the full HETVAR model. Then under $H_0 : \sigma_\alpha^2 = \sigma_u^2$, $-2(l_0 - l_1)$ follows a 50:50 mixture of χ_1^2 and χ_0^2 (Stram and Lee 1994).

4.3.3 The ssSSVS model

I consider a variable selection specification by writing $\mathbf{D} = diag\{\frac{(1-\phi_j)}{c} + \phi_j\}; j = 1, 2, ..., m$, where $c \gg 1$ and $p(\phi_j \mid \pi_\phi) = \pi_\phi^{\phi_j}(1-\pi_\phi)^{1-\phi_j}$, $\phi_j = 0,1$. The corresponding model is SSVS with further details provided by Chen *et al.* (2017). Note that π_ϕ is the probability that marker has a large variance, and subsequently large effect, with respect to the trait. This specification for **D** is equivalent to assigning the following mixture prior for the SNP marker effects:

$$p(\alpha_i \mid \sigma_\alpha^2, \phi_i) = \frac{1}{\sqrt{2\pi\sigma_\alpha^2\left(\frac{(1-\phi_i)}{c} + \phi_i\right)}} \exp\left(-\frac{1}{2}\frac{\alpha_j^2}{\sigma_\alpha^2\left(\frac{(1-\phi_i)}{c} + \phi_i\right)}\right); i = 1, 2, ..., m \qquad [4.11]$$

Additionally, the prior for π_ϕ is Beta distributed, i.e.,

$$p(\pi_\phi \mid \alpha_0, \beta_0) \propto \pi_\phi^{\alpha_0}(1-\pi_\phi)^{\beta_0} \qquad [4.12]$$

For the variance components, I specify scaled inverse chi-square distribution priors; i.e.,

$$p(\sigma_e^2 \mid \nu_e, s_e^2) \propto (\sigma_e^2)^{-\left(\frac{\nu_e}{2}+1\right)} e^{-\frac{\nu_e s_e^2}{2\sigma_e^2}}, \qquad [4.13]$$

$$p\left(\sigma_\alpha^2 \mid \nu_\alpha, s_\alpha^2\right) \propto \left(\sigma_\alpha^2\right)^{-\left(\frac{\nu_\alpha}{2}+1\right)} e^{-\frac{\nu_\alpha s_\alpha^2}{2\sigma_\alpha^2}}, \quad [4.14]$$

and

$$p\left(\sigma_u^2 \mid \nu_u, s_u^2\right) \propto \left(\sigma_u^2\right)^{-\left(\frac{\nu_u}{2}+1\right)} e^{-\frac{\nu_u s_u^2}{2\sigma_u^2}}. \quad [4.15]$$

For all analyses in this paper, I use non-informative priors where the degree of freedom $\nu_e = \nu_\alpha = \nu_u = -1$ and scale $s_e^2 = s_\alpha^2 = s_u^2 = 0$ (Gelman 2006).

Given the prior specification above, the joint posterior density is given as follows:

$$p\left(\boldsymbol{\beta}, \alpha_1, \alpha_2, ..., \alpha_m, \boldsymbol{\varepsilon}, \sigma_u^2, \sigma_\alpha^2, \sigma_e^2 \mid \mathbf{y}\right) \propto \left(\prod_{i=1}^{n} p\left(y_i \mid \boldsymbol{\beta}, \alpha_1, \alpha_2, ..., \alpha_m, \boldsymbol{\varepsilon}, \sigma_e^2\right)\right)$$
$$\left(\prod_{j=1}^{m} p\left(\alpha_j \mid \sigma_\alpha^2, \phi_j\right) p(\phi_j \mid \pi_\phi)\right) p\left(\boldsymbol{\varepsilon} \mid \sigma_u^2\right) p\left(\sigma_\alpha^2 \mid s_\alpha^2, \nu_\alpha\right) p\left(\sigma_u^2 \mid s_u^2, \nu_u\right) p\left(\sigma_e^2 \mid \nu_e, s_e^2\right) p\left(\pi_\phi \mid \alpha_0, \beta_0\right)$$

[4.16]

Then unknown parameters for ssSSVS can be sampled from their joint posterior density using Markov Chain Monte Carlo (MCMC). Details on the MCMC sampling scheme for ssSSVS is provided in the Supplementary File S1 and for SSVS in Chen et al. (2017).

4.3.4 Conducting Genome-Wide Association Analyses

4.3.4.1 **Single SNP marker associations**

An efficient strategy for providing formal GWA inference under EMMAX is provided by Gualdron Duarte *et al.* (2014) and further described in Chen *et al.* (2017) with a formal proof provided in Bernal Rubio *et al.* (2016). This approach is equivalent to treating the SNP marker effect of interest as fixed while treating all other SNP effects as random in a generalized least squares (GLS) approach. The same strategy can be used to derive formal GWA under a single-step modification of EMMAX which I denote as ssEMMAX, the test statistic for the ssEMMAX test on SNP marker *j* can then be simply written as

$$z_j = \frac{\hat{\alpha}_j}{se\left(\hat{\alpha}_j\right)} \quad [4.17]$$

Note that SNP effect estimates $\hat{\boldsymbol{\alpha}} = \left\{\hat{\alpha}_j\right\}_{j=1}^{m}$ can simply be backsolved from the solutions to Equation [4.8] using $\hat{\boldsymbol{\alpha}} = \mathbf{T}_g \mathbf{G}_g^{-1} \hat{\mathbf{u}}_g^{(\alpha)}$ where $\mathbf{G}_g = \mathbf{T}_g \mathbf{T}_g^{'}$ (Stranden and Garrick 2009). $se\left(\hat{\alpha}_j\right)$ is the square root of diagonal of $\text{var}\left(\hat{\mathbf{g}}\right)$, where $\text{var}\left(\hat{\mathbf{g}}\right) = \mathbf{T}_g^{'} \mathbf{G}_g^{-1} \left(\mathbf{G}_g \sigma_\alpha^2 - \mathbf{C}_{gg}^{uu} \sigma_e^2\right) \mathbf{G}_g^{-1} \mathbf{T}_g$. Here $\mathbf{C}_{gg}^{uu}$ is essentially the block diagonal of the inverse of the coefficient matrix in Equations [4.7] or [4.8] corresponding to $\mathbf{u}_g^{(\alpha)}$. That is, one can obtain $\mathbf{C}_{gg}^{uu}$ by inverting the coefficient matrix in Equation [4.7] for HETVAR and Equation [4.8] for HOMVAR. In other words, $\mathbf{C}_{gg}^{uu} \sigma_e^2$ is the prediction error covariance matrix of $\hat{\mathbf{u}}_g^{(\alpha)}$.

For MCMC-based single SNP inferences, I based inferences on the posterior probability of association (PPA) for SNP marker *j* (i.e. PPA_j):

$$PPA_j = \frac{\sum_{l=1}^{N} \phi_{j(l)}}{N} \quad [4.18]$$

For SSVS, *N* denotes the number of MCMC cycles saved for posterior inference and $\phi_{j(l)}$ is a binary draw from the full conditional distribution of ϕ_j at MCMC cycle *l*.

4.3.4.2 **Windows based associations**

The window based approach follows what has been developed in Chen *et al.* (2017). Suppose that window $k = 1,2,3,\ldots,K$ contains n_k markers such that $\mathbf{T_g}$ can be partitioned accordingly into $\mathbf{T}_g = \left[\mathbf{T}_{g1} \quad \mathbf{T}_{g2} \quad \ldots \quad \mathbf{T}_{gK}\right]$ with $\mathbf{T_g}_k$ having n_k columns, containing the submatrix representing the n_k SNP markers for window *k*. Therefore, the submatrix of $\text{var}\left(\hat{\boldsymbol{\alpha}}\right)$ for window

k is $\text{var}(\hat{\boldsymbol{\alpha}}_k) = \mathbf{T}_{gk}^{'}\mathbf{G}_g^{-1}\left(\mathbf{G}_g\sigma_\alpha^2 - \mathbf{C}_{gg}^{uu}\sigma_e^2\right)\mathbf{G}_g^{-1}\mathbf{T}_{gk}$. Similarly, the vector $\boldsymbol{\alpha}$ is partitioned accordingly; i.e. $\boldsymbol{\alpha} = \left[\boldsymbol{\alpha}_1^{'} \quad \boldsymbol{\alpha}_2^{'} \quad \cdots \quad \boldsymbol{\alpha}_K^{'}\right]'$ such that $\boldsymbol{\alpha}_k$ is of dimension n_k x 1. The extension to a joint EMMAX-like test on n_k markers in window k involves the following determination:

$$\chi_k^2 = \hat{\boldsymbol{\alpha}}_k(\text{var}(\hat{\boldsymbol{\alpha}}_k))^{-1}\hat{\boldsymbol{\alpha}}_k \qquad [4.19]$$

where χ_k^2 is chi-square distributed with n_k degrees of freedom under H_o: $\boldsymbol{\alpha}_k = 0$.

For windows based inference using SSVS under MCMC, I just compute the PPA for each window k (i.e. PPA_k) in Equation [4.20] as also presented in Chen *et al.* (2017) and similar to Fernando *et al.* (2017)

$$PPA_k = \frac{\sum_{l=1}^{N} I((\sum_{j=1}^{n_k}\phi_{kj(l)}) > 0)}{N} \qquad [4.20]$$

Here, $\phi_{kj(l)}$ defines a binary draw from the full conditional distribution of τ_j for SNP marker j located within window k drawn during MCMC cycle l. Note then that $I(\sum_{j=1}^{n_k}\phi_{kj(l)}) > 0$ is equal to 1 when any of the draws of $\tau_{kj(l)}$ within window k are equal to 1. This simply entails determining whether any of the SNP markers within region k have an association.

4.4 Data and Applications Strategies

4.4.1 Genotypes

The SNP marker genotypes of 3186 Holstein cows were provided from 6 US research stations including Iowa State University (**ISU**), Michigan State University (**MSU**), the University of Florida (**UF**), the University of Wisconsin-Madison (**UW**), the USDA Dairy Forage Research Center (**USDFRC**) in Madison, Wisconsin, and the USDA Animal Genomics and Improvement Laboratory (**AGIL**) in Beltsville, MD. Genotypes were obtained using Illumina® BovineSNP50

Genotyping BeadChip and then imputed and edited as in Lu (2016), which excluded SNPs with minor allele frequency (MAF) less than 0.05 and SNP markers in complete LD with each other, leaving 57,347 SNP markers for analysis.

4.4.2 Simulation study

To compare the two broad categories of models of interest, I conducted a simulation study based on the actual genotypes of 3186 Holstein cows was collected from 6 research stations as described above. I generated QTL effects $\boldsymbol{\alpha}_{qtl}$ from a Gamma distribution with shape parameter equal to 0.42 based on average estimates reported by Hayes and Goddard (2001). I conjectured that the number of QTLs might also influence WGP and GWA performance such that I considered n_{qtl} = 30, 300, or 3000. Here, I simulated 10 replicates for each specification of n_{qtl}, resulting in 30 different simulated datasets in total. For each dataset, QTL effects, $\boldsymbol{\alpha}_{qtl}$, were randomly assigned to n_{qtl} SNP markers across the genome with a random half of the effects multiplied by -1 as per Meuwissen *et al.* (2001). The corresponding genotypes $\mathbf{M}_{qtl}$ for QTL on these cows were then a n x n_{qtl} subset of the SNP genotype matrix $\mathbf{M}$ such that the true breeding values $\mathbf{u}_{TRUE} = \mathbf{M}_{qtl}\boldsymbol{\alpha}_{qtl}$. Phenotypes for animals were generated based on a heritability of 0.25 as estimated for milk fat from this same dataset. Only the remaining marker genotypes $\mathbf{M}_{-qtl}$ were used for the analyses; i.e., QTL genotypes were always masked.

For analysis, 50% of cows were masked as non-genotyped such that their SNP marker genotypes were treated as missing. Analyses were conducted using GBLUP, SSVS, ssGBLUP, and ssSSVS, where GBLUP and SSVS, as previously noted, do not include any phenotypes on non-genotyped animals. For GBLUP and ssGBLUP, I also use AIREML to estimate the variance components using both HETVAR and HOMVAR specifications with fixed and random effect estimates obtained by solving the MME provided in Equations [4.7] and [4.8]. For SSVS

and ssSSVS, I ran MCMC for 200,000 iterations in total, discarding the first 100,000 iterations as burn-in and basing inference on saving every 10 of the remaining 100,000 cycles for a total of 10,000 samples from the posterior density. It is known that estimating π_ϕ can be challenging when n_{qtl} is large (van den Berg *et al.* 2013), i.e., n_{qtl} =3000 in our case. Therefore, for n_{qtl} =3000, I considered 3 different specifications for π_ϕ: 1) estimating π_ϕ, 2) setting $\pi_\phi = 0.01$, and setting $\pi_\phi = 0.001$.

I defined the prediction accuracy of breeding values for WGP as the correlation between the estimated breeding values ($\hat{\mathbf{u}}$) and the true breeding values ($\mathbf{u}_{TRUE}$) in our simulation study. For all individuals, I compared the WGP accuracies of breeding values using GBLUP, SSVS, ssGBLUP and ssSSVS for each specification of n_{qtl}.

As for GWA, single SNP marker inferences were implemented as described in the Methods and Materials. *P*-values for EMMAX and ssEMMAX were based on the *z* test in Equation [4.17] whereas SSVS and ssSSVS provided posterior probabilities (i.e., PPA) based on the MCMC samples in Equation [4.18]. Since the remaining genotypes $\mathbf{M}_{g,-qtl}$ did not include the simulated QTL, SNP markers were treated as true positives if the QTL were located between themselves and an adjacent SNP marker.

I also conducted windows based inference based on the EMMAX and ssEMMAX procedures using *P*-values based on the chi-square test in Equation [4.19] whereas PPA within a window using SSVS and ssSSVS were based on randomly drawn MCMC samples as in Equation [4.20]. The length of each window was determined by the `BALD` R package (Dehman and Neuvial 2015), which adaptively determines window sizes based on LD using the procedure described by Dehman *et al.* (2015).

The performance of all methods and models were compared using the receiver operating characteristic (ROC) curves which plots the true positive rate (TPR) against the false positive rate (FPR) for each method (Metz 1978). I specifically chose to compare the performance of different methods using a partial area under the curve up until an FPR= of 5% (pAUC05) as also per Chen et al. (2017). Given that a random classifier has a pAUC05 of $0.05^2/2= 0.00125$, I further rescaled all pAUC05 by a factor of 0.00125^{-1} such the relative pAUC05 for a random classifier is 1. An ANOVA blocking on simulated data replicate was used to compare the different methods (GBLUP, SSVS, ssGBLUP, and ssSSVS) for pAUC05 for each specification of $n_{qtl.}$.

4.4.3 Dairy consortium data

The phenotypes that I choose for demonstration are the corresponding milk fat yields and body weights for the 3186 genotyped Holstein cows described earlier. The data was edited and described in Tempelman *et al.* (2015) and Lu *et al.* (2015). The complete breakdown of the number of genotyped and phenotyped cows for each station is provided in Table 4.1. Four generation pedigrees on all cows were provided by the USDA-AGIL.

Table 4.1 Number of cows by research station in dairy consortium study

Station[1]	Number of cows
ISU	930
UW	780
AGIL	488
UF	377
USDFRC	347
MSU	264
Total	3186

[1]ISU = Iowa State University; USDFRC = USDA Dairy Forages Research Center; AGIL = USDA Animal Genomics Improvement Laboratory; UF = University of Florida; UW= University of Wisconsin- Madison; MSU = Michigan State University.

Since our goal was to evaluate the performance of single step models for WGP and GWA, I randomly masked the genotypes (i.e., as non-genotyped) on a proportion of the cows. Specifically, cows were randomly partitioned into five equally sized subsamples stratified by station. Of the 5 subsamples within each station, the genotypes on one single subsample were masked as non-genotyped whereas the genotypes on the remaining 4 subsamples were kept. This masking arrangement was repeated for 5 times such that each of the 5 subsamples within a station had masked genotypes for one of the partitions that I label as P1-P5. An illustration of the partition for one station is given in Figure 4.1. Since P1-P5 were stratified by stations, these partitions were within-station partitions such that within station GWA assessments of the various methods were based on P1-P5.

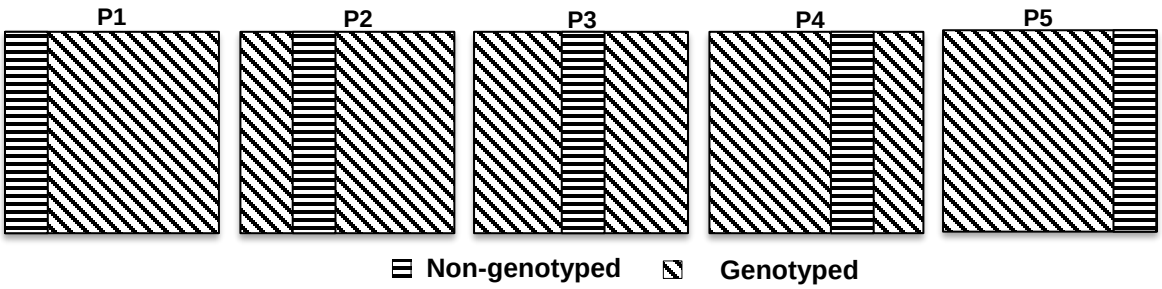

Figure 4.1 Illustration of within station partitions P1-P5 for one particular station. 20% of the cows are marked as non-genotyped with the remaining 80% cows treated as genotyped in each partition

The prediction accuracy of WGP was evaluated using 5-fold cross-validation (CV) for each single partition of P1-P5 across herds; i.e., a total of 25 folds. That is, each of the P1- P5 partitions within each station was further subpartitioned into 5 orthogonal subsets such that for each partition P1-P5, a 5-fold cross-validation of 4 training orthogonal subsets and 1 validation orthogonal subset included both genotyped and non-genotyped animals. An illustration for one

such partition, say P1, is provided in Figure 4.2.

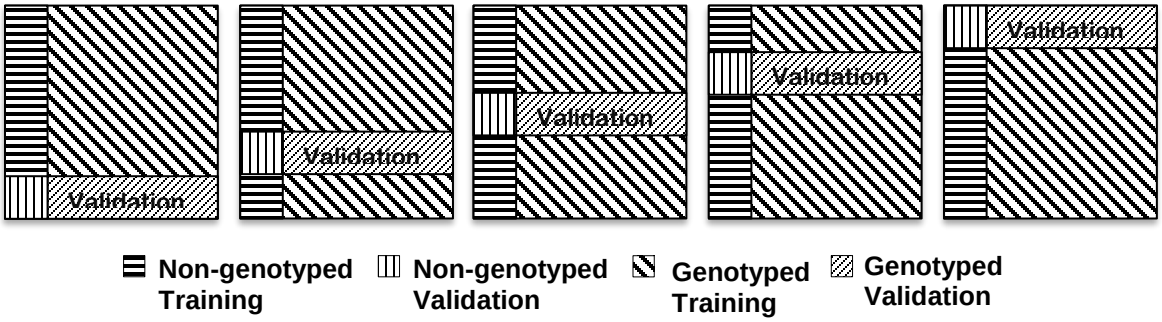

Figure 4.2 Example of training vs. validation partition for P1 (from Figure 4.1)

4.4.4 Benchmarking analysis for dairy consortium data

To provide benchmark gold standards for all assessments involving animals with missing genotypes, I conducted baseline WGP and GWA analyses using genotypes and phenotypes on all animals. For WGP, our benchmark assessments were based on conventional GBLUP and SSVS analyses using the complete data, i.e. using genotypes on all 3186 cows. Then the same total 25 sub partitions described above were used to assess CV prediction accuracy against this ideal situation. For GWA, our benchmark analyses were based on EMMAX and SSVS using the entire dataset and all the cows are treated as genotyped. GWA with non-genotyped cows in the cross-validation stud described below were then compared against the benchmark for locations having strong measures of association (i.e., low P-values or high PPA).

4.4.5 Cross-validation study for dairy data

I specified parity class as fixed effects, a fourth-order polynomial regression on days in milk (DIM), and the random effects of rations, test dates, and genetics (i.e. $\mathbf{u}^{(\alpha)}$ and $\boldsymbol{\varepsilon}$) in the WGP model. To save computing time and to stabilize REML convergence of variance components, variance components for ration effects and test date effects were estimated just once from the entire dataset using genotype information on all cows. The values for these variance components

were then fixed to those estimates for all subsequent cross-validation comparisons. I separately compared the WGP CV accuracies of GBLUP, SSVS, ssGBLUP and ssSSVS for genotyped cows, and the WGP CV accuracies of ssGBLUP and ssSSVS for non-genotyped cows.

For GWA, I examined the performance of different models both within and across station cross validation studies. I also fitted parity class, ration and test date as fixed effects, and up to a fourth-order polynomial on DIM as covariates in both situations. I based our 5-fold within station cross-validation study on partitions P1-P5 as described previously. For the across station study, I constructed a 6-fold cross-validation study by masking the genotypes from all cows within one station with genotype information available on cows from all other stations, one station at a time for a total of 6 folds. I compared the location of peaks (i.e. strongest GWA associations) of EMMAX, ssEMMAX, SSVS and ssSSVS with corresponding benchmarking results from EMMAX and SSVS which treated all genotypes as known. In addition, I also compared the peak strength of association in ssEMMAX versus EMMAX based on $-\log_{10}$(*P-value*), and ssSSVS versus SSVS based on PPA for the most significant single SNP or adaptive window based associations. EMMAX and ssEMMAX were implemented in the same manner as the simulation study. SSVS and ssSSVS were also implemented like in the simulation study except that I fixed $\pi_\phi = 0.0001$ for milkfat and $\pi_\phi = 0.02$ for body weight in all analyses because of poor mixing when estimating π_ϕ using the benchmark data. These values for π_ϕ were chosen from the set of [0.0001, 0.001, 0.005, 0.01, 0.02, 0.05, 0.1] having the highest average prediction accuracy from 5-fold cross-validation on the benchmark data in a manner similar to Lee et al. (2017). All reported WGP and GWA inferences were based on the HOMVAR specification for variance component for ssGBLUP and ssEMMAX, i.e. $\sigma_\alpha^2 = \sigma_u^2$, due to computational expedience and fast and stable convergence.

Additional standalone studies were conducted to assess whether the HOMVAR specification was a better fit than the HETVAR specification. I adopted the likelihood ratio tests using GWA partitions for milkfat. The analyses were conducted among 5 within station partitions (P1-P5) and among 6 across station splits of genotyped and non-genotyped animals. I reassessed GWA under HETVAR when $H_0 : \sigma_\alpha^2 = \sigma_u^2$ was rejected. In addition, for 6 across station splits, I also tested the hypothesis on the 6 "flipped" partitions, in which only 1 station was treated as genotyped while other 5 stations were treated as non-genotyped.

4.4.6 Software

In addition to `BALD` and `ROCR`, I have developed a tool to implement ssSSVS and ssEMMAX for both WGP and GWA (single SNP and window based) which is included in `BATools` R package (https://github.com/chenchunyu88/BATools).

4.5 Results

4.5.1 Simulation Study

Method specific boxplots of WGP accuracies of EBV across the 10 replicated data sets for each n_{qtl} are provided in Figure 4.3. The single-step approaches (both ssGBLUP and ssSSVS) had higher WGP accuracies than their conventional counterparts (GBLUP and SSVS) that ignored phenotypes on non-genotyped animals. In turn, ssSSVS had higher WGP accuracies than ssGBLUP except when n_{qtl} =3000, noting that the advantage for ssSSVS was largest for simpler genetic architectures; i.e., n_{qtl} = 30. For non-genotyped cows, ssSSVS led to a higher WGP accuracy compared to ssGBLUP when n_{qtl} = 30 and 300 with no evidence of a difference when n_{qtl} = 3000.

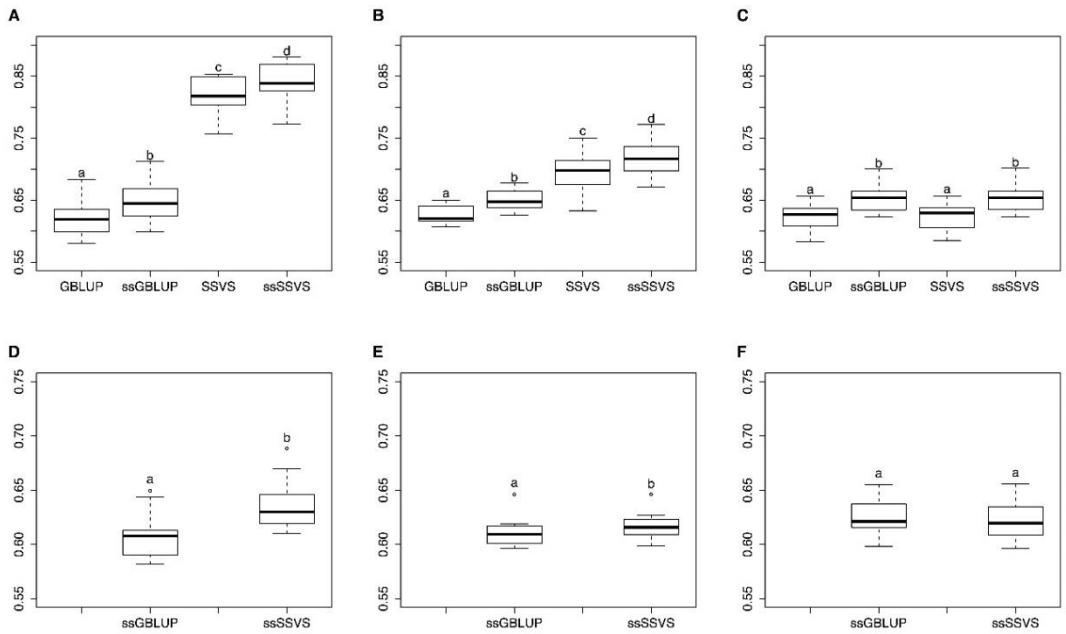

Figure 4.3 Boxplots of prediction accuracies of breeding values of genotyped and non-genotyped cows based on the simulation study of different n_{qtl} of 30, 300 and 3000. Panel A) n_{qtl}=30 for genotyped cows; Panel B) n_{qtl}=300 for genotyped cows; Panel C) n_{qtl}=3000 for genotyped cows; Panel D) n_{qtl}=30 for non-genotyped cows; Panel E) n_{qtl}=300 for non-genotyped cows; Panel F) n_{qtl}=3000 for non-genotyped cows; Methods not sharing the same letter code within each panel have different mean prediction accuracies (P<0.05).

Comparisons of relative pAUC05 on GWA performance between the various methods are provided in Figure 4.4. Using the adaptive window approach, SSVS and ssSSVS outperformed EMMAX and ssEMMAX. Furthermore, the single-step procedures did not typically lead to a higher pAUC05 relative to their conventional counterparts. When n_{qtl}=30, there was no evidence of an advantage for using a single-step approach whereas for n_{qtl}=300 and 3000; only ssEMMAX had a higher pAUC05 than EMMAX although these differences were very small. Using single SNP association testing, differences were only detected when n_{qtl}=30, where both EMMAX and ssEMMAX had higher pAUC05 than SSVS. As for the value for π_ϕ, I found that sampling π_ϕ or fixing $\pi_\phi = 0.01$ lead to equivalent pAUC05 for ssSSVS and SSVS; but fixing $\pi_\phi = 0.001$ led to ssSSVS having lower pAUC05 than SSVS (P<0.05) as shown in Figure 4.5.

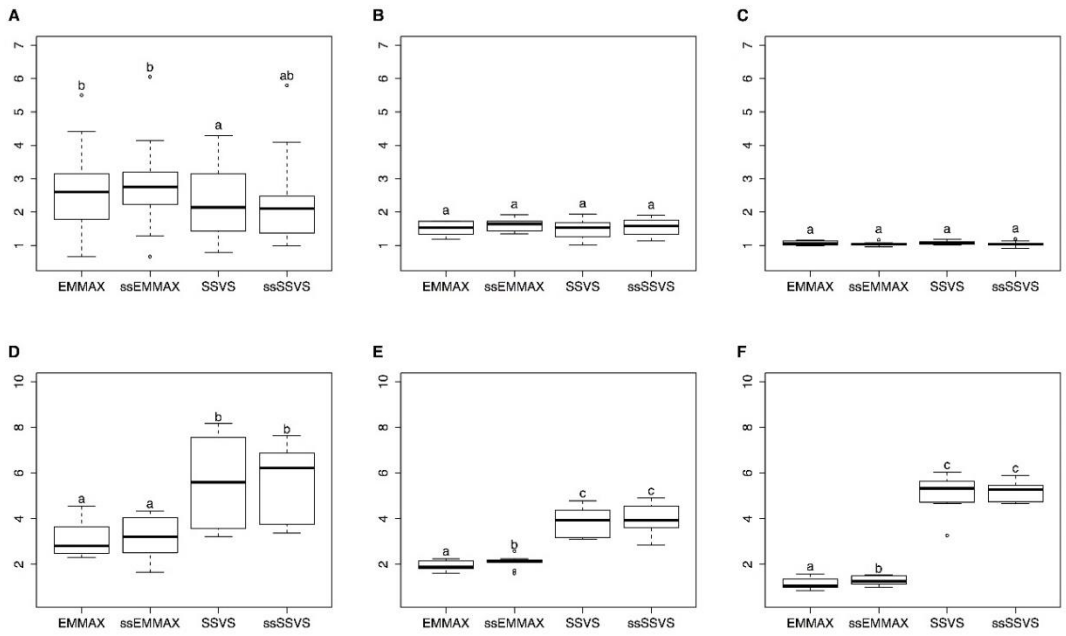

Figure 4.4 Boxplot of relative pAUC05 for each method on the simulation study of different n_{qtl} of 30, 300 and 3000. The first row is the relative pAUC05 using single SNP approach and the second row is the relative pAUC05 using adaptive window approach. Panel A) n_{qtl}=30 for single SNP; Panel B) n_{qtl}=300 for single SNP; Panel C) n_{qtl}=3000 for single SNP; Panel D) n_{qtl}=30 for adaptive window; Panel E) n_{qtl}=300 for adaptive window; Panel F) n_{qtl}=3000 for adaptive window. Methods not sharing the same letter code are significantly different from each other within each plot (P<0.05)

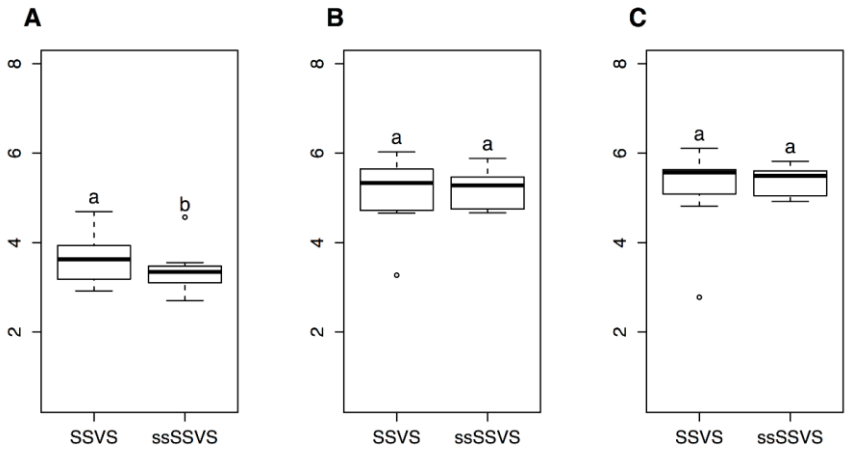

Figure 4.5 Boxplot of relative pAUC05 for SSVS and ssSSVS on the simulation study with n_{qtl}=3000 for adaptive window approach based on different specifications for π_ϕ: Panel A) $\pi_\phi = 0.001$, Panel B) $\pi_\phi = 0.01$ and Panel C) joint MCMC sampling of π_ϕ. Methods not sharing the same letter code are significantly different from each other within each plot (P<0.05).

4.5.2 Dairy Data

The mean cross-validation WGP accuracies based on treating all cows as genotyped is provided in Table 4.2 for benchmarking purposes. Here, SSVS outperformed GBLUP for milk fat whereas no difference in WGP accuracy between GBLUP and SSVS was determined for body weight. At any rate, differences were very small in either case (i.e. less than 2 percentage points).

Table 4.2 Cross-validation (25-fold) prediction accuracies for comparing GBLUP and SSVS (all animals genotyped) in benchmark analysis

Trait	GBLUP	SSVS
Milk fat	0.7126[a]	0.7156[b]
Body weight	0.7645[a]	0.7643[a]

Values not sharing the same letter within a row have different (P <0.05) prediction accuracy.

With genotypes on 20% of the cows being masked, SSVS demonstrated higher (*P*<0.05) WGP accuracies compared to GBLUP on genotyped cows for milkfat; similarly, ssSSVS outperformed ssGBLUP for milkfat (Table 4.3). However, for body weight, no difference in WGP accuracies were detected between ssSSVS and ssGBLUP whereas SSVS had higher WGP accuracy than GBLUP (*P*<0.05). At any rate, single step approaches outperformed their conventional counterparts for both traits within either model (GBLUP/SSVS) although, admittedly, differences were small. Nevertheless, no difference in GEBV accuracies were found between ssSSVS and ssGBLUP for either trait on non-genotyped cows (Table 4.4).

Table 4.3 Cross-validation (25-fold) prediction accuracies for GBLUP and SSVS and their respective single step extensions (ssGBLUP and ssSSVS) on genotyped cows

Trait	GBLUP	ssGBLUP	SSVS	ssSSVS
Milk fat	0.7037[a]	0.7102[b]	0.7081[b]	0.7123[c]
Body weight	0.7461[a]	0.7601[c]	0.7564[b]	0.7597[c]

Values not sharing the same letter within a row have different (P <0.05) prediction accuracies

Table 4.4 Cross-validation (25-fold) prediction accuracies for GBLUP and SSVS and their respective single step extensions (ssGBLUP and ssSSVS) on non-genotyped cows

Trait	ssGBLUP	ssSSVS
Milk fat	0.7101[a]	0.7085[a]
Body weight	0.7556[a]	0.7547[a]

Values not sharing the same letter within a row have different (P <0.05) prediction accuracies.

When genotypes on all cows were used for benchmarking GWA analyses on milkfat, the highest peak determined by EMMAX (Figure 4.6A) based on single SNP associations were located at 1801.116kb (SNP ARS-BFGL-NGS-4939) on chromosome 14 within the same region

as the DGAT1 gene located between 1795.425kb and 1804.838kb and known to be a major gene influencing milk fat yield (Grisart *et al.* 2002). The peak window identified by EMMAX using the adaptive window approach ranged from 1189.341kb to 1801.116kb on chromosome 14 whereas 5 other neighboring windows on chromosome 14 (1868.636kb to 2084.067kb; 2217.163kb - 2239.085kb, 2276.443-2674.264kb, 2790.501kb-2909.929kb, and 3029.996kb-3059.698kb) were also deemed to have significant associations with milk fat. For both sets of GWA cross-validation studies (i.e., within station and across station splits) using EMMAX or ssEMMAX, the single SNP associations based on all training data had the same peak as the benchmark analysis. Similarly, for the adaptive window associations, peaks shifted between the 4 windows listed above with the exception of two training datasets: partition P5 in the within station study where region 45150.817kb-46093.561kb on chromosome 16 had the strongest association; and the subset excluding UF data in the across station cross-validation study for which another region 44931.986kb-45039.750kb on chromosome 8 had the strongest association (Figures B5 and B10 in Appendix B).

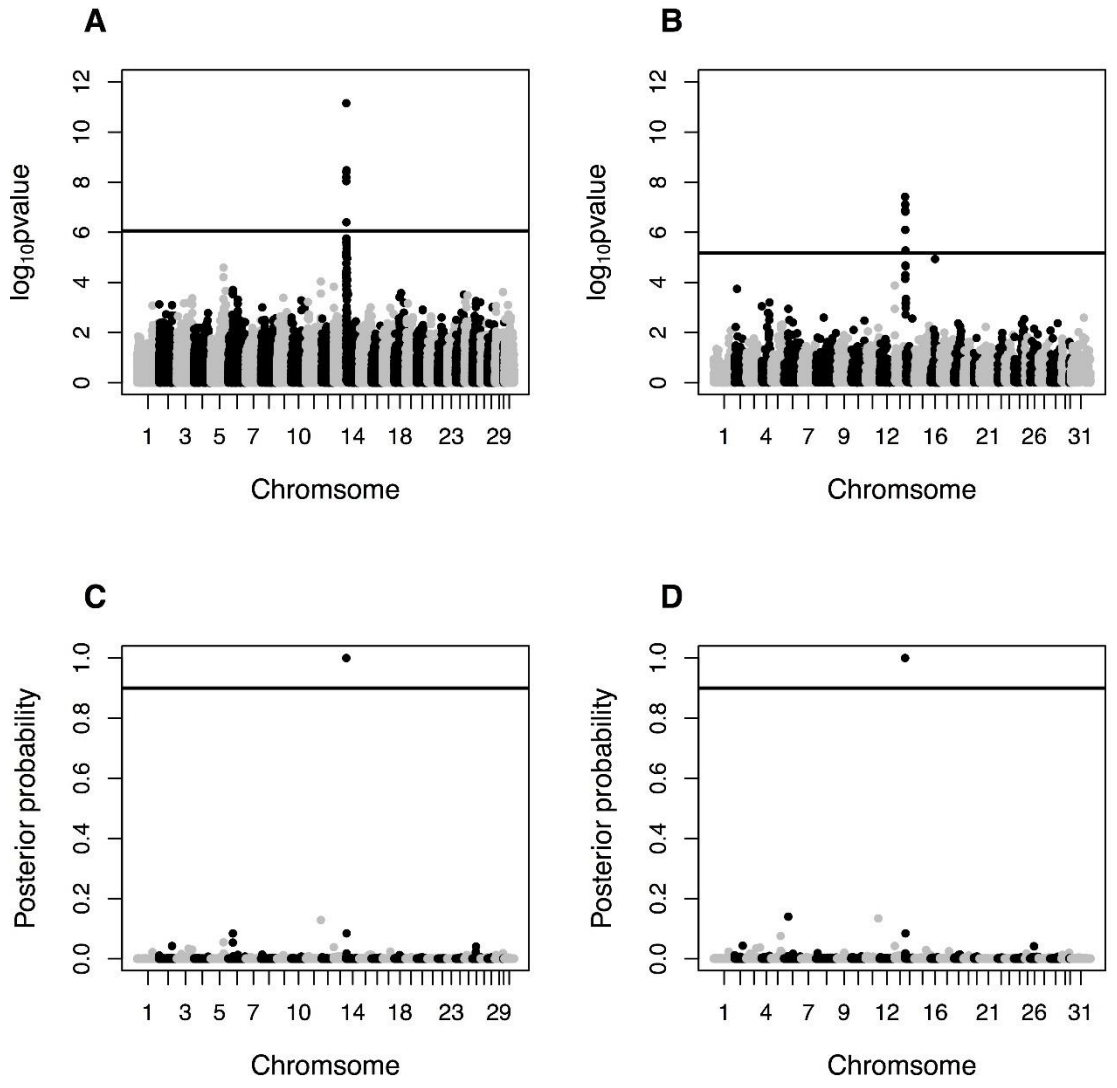

Figure 4.6 Manhattan plot for milkfat treating all cows as genotyped in benchmarking study. Panel A: single SNP inferences for EMMAX; Panel B: adaptive window inferences for EMMAX; Panel C: single SNP inferences for SSVS; Panel D: adaptive window inferences for SSVS.

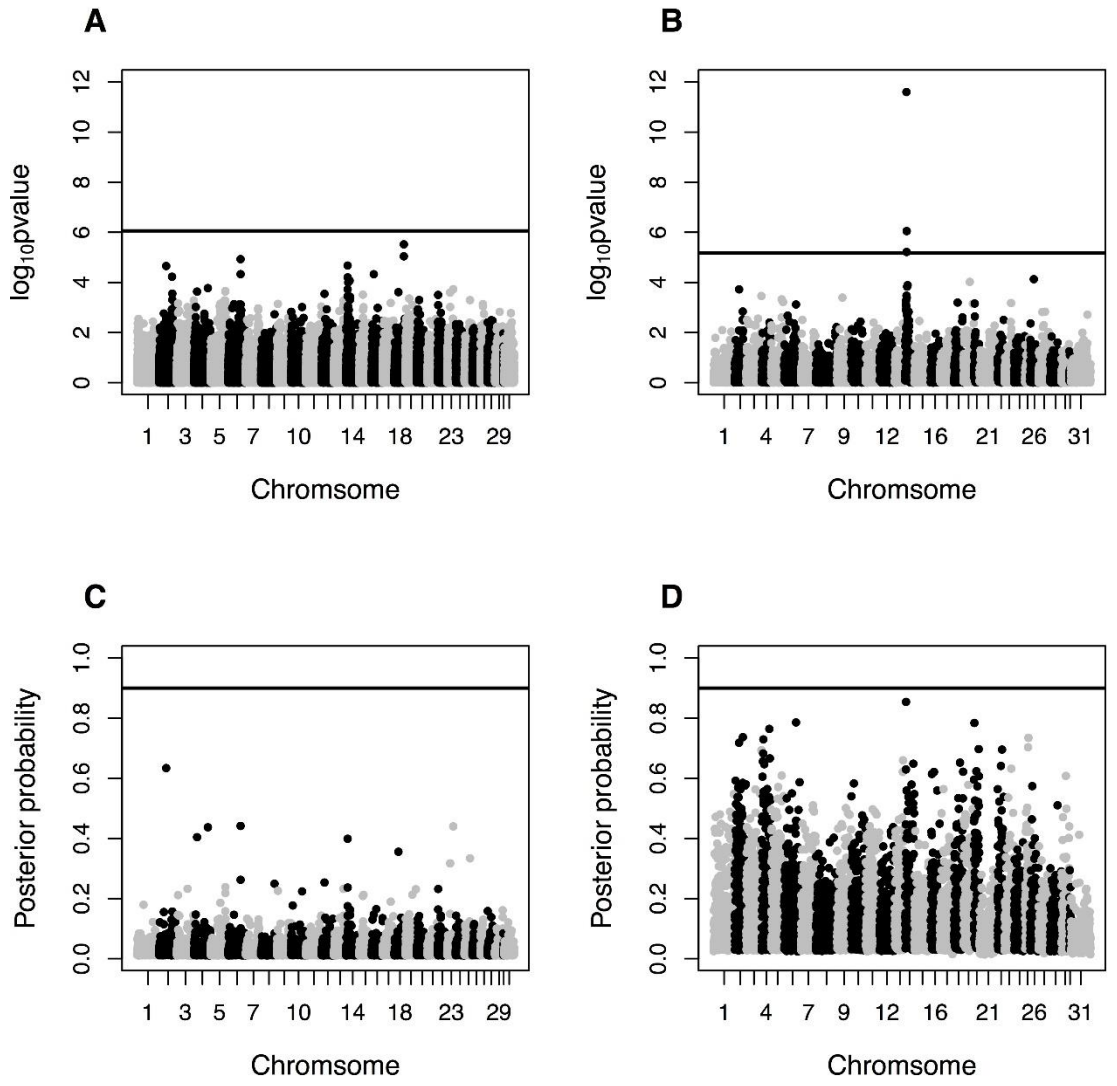

Figure 4.7 Manhattan plot for body weight treating all cows as genotyped in benchmarking study. Panel A: single SNP inferences for EMMAX; Panel B: adaptive window inferences approach for EMMAX; Panel C: single SNP inferences for SSVS; Panel D: adaptive window inferences for SSVS.

To further assess the benefit of adapting single-step extensions of GWA, I compared the measured strengths of association (i.e., $-\log_{10}$(*P-value*) or PPA) for the peak SNP or window for ssEMMAX or ssSSVS versus their conventional counterparts. For milk fat, SNP ARS-BFGL-NGS-4939 was the overwhelmingly most significant association in the benchmark analysis using

all available genotypes using both EMMAX or SSVS; hence our attention was focused on ARS-BFGL-NGS-4939 for milk fat. For body weight, I focused on SNP marker ARS-BFGL-NGS-109285 located at 57589.121kb on chromosome 18 since its inferred strength of associations dominated all other markers. I particularly focused our attention on the gains in $-\log_{10}$(*P*-value) or PPA, respectively, using their single step extensions on five separate analyses where the genotypes on partitions P1-P5 were masked for a within station assessment as well as on the six separate analyses where the genotypes on each station are masked in turn for the across herd analysis. For single SNP inferences, I noticed that the mean $-\log_{10}$(*P*-values) on ARS-BFGL-NGS-4939 were not different for the within station and across station analyses on milk fat when ssEMMAX was used instead of EMMAX (Table 4.5), similarly, there was no evidence of such a difference in mean $-\log_{10}$(*P*-values) for body weight (Table 4.6). The mean PPA for ARS-BFGL-NGS-4939 increased from 0.84 for SSVS to 0.91 to ssSSVS for within station analyses and from 0.69 to 0.88 for across station analyses, but the differences were not deemed significant (P>0.05); furthermore, the corresponding differences for body weight were rather trivial.

Table 4.5 Average ((n=5 fold for within herds and n=6 fold for across herds) measures of strength of association ($-\log_{10}$P-value using GBLUP or posterior probability using SSVS) for most significant SNP/genomic region using single-step compared to conventional specifications on milk fat

	$-\log_{10}$P-value					Posterior Probability				
	Single SNP		Adaptive window			Single SNP		Adaptive window	
Methods	Within	Across	Within	Across	Methods	Within	Across	Within	Across
EMMAX	8.73[a]	9.20[a]	5.67[a]	5.88[a]	SSVS	0.84[a]	0.69[a]	0.98[a]	0.75[a]
ssEMMAX	9.23[a]	8.94[a]	6.16[a]	5.61[a]	ssSSVS	0.91[a]	0.88[a]	0.91[a]	0.92[a]

Values not sharing the same letter within a column have different (P <0.05) height for peaks in the Manhattan plot. For single SNP approach, the reference SNP ARS-BFGL-NGS-4939 is located at 1801.116kb in chromosome 14 for both ssEMMAX and ssSSVS. For adaptive window approach, the reference window for ssEMMAX ranges from 1868.636kb to 2084.067kb (3868th

window) in chromosome 14 and the reference window for ssSSVS ranges from 1189.341kb to 1801.116kb (3867th window) in chromosome 14.

Table 4.6 Average (n=5 fold for within herds and n=6 fold for across herds) measures of strength of association (-$\log_{10}$P-value using GBLUP or posterior probability using SSVS) for most significant SNP/genomic region using single-step compared to conventional specifications on body weight

	-$\log_{10}$P-value				Posterior Probability				
	Single SNP		Adaptive window			Single SNP		Adaptive window	
Methods	Within	Across	Within	Across	Methods	Within	Across	Within	Across
EMMAX	4.70 [a]	4.79[a]	9.37[a]	9.93[a]	SSVS	0.32[a]	0.34[a]	0.64[a,1]	0.67[a,2]
ssEMMAX	4.88 [a]	4.82[a]	9.70[a]	10.04[a]	ssSSVS	0.32[a]	0.37[a]	0.68[a,1]	0.70[a,2]

Values not sharing the same letter within a column have different (P <0.05) height for peaks in the Manhattan plot.). The [1]P=0.08 and [2]P=0.06 comparing the difference between ssSSVS and SSVS. The adaptive window for ssEMMAX/EMMAX ranges from 8551.460kb to 8560.116kb in chromosome 14; and the adaptive window for ssSSVS/SSVS ranges from 88350.890kb to 88668.261kb in chromosome 6.

I conducted the same comparison based on adaptive window inferences. Based on the benchmark analyses, different but neighboring peak windows (1189.341kb to 1801.116kb) on chromosome 14 were determined by EMMAX and SSVS, respectively, as being most significant for milk fat. Using these as the respective reference regions for the assessment of single step extensions of these two models, it was again determined that ssEMMAX and EMMAX were not significantly different from each other whereas large albeit non-significant improvements in mean PPA were observed for single step extensions of SSVS from 0.75 to 0.92 based on across herd partition (Table 4.5). Similarly, for body weight, the windows were different between the two benchmark analyses for body weight being Window 3894 (ranging from 8551.460 to 8560.116 kb on chromosome 14) for EMMAX and Window 3886 (ranging from 7104.148-7342.696kb on chromosome 14) for SSVS. I noticed for single step extensions of SSVS for genomic window associations had slightly higher (but not statistically significant) PPA than

regular SSVS on body weight for both within and across station analysis with P=0.08 and P=0.06 (Table 4.6). However, for EMMAX, as with single SNP inferences, single step extensions had little merit for within and across herd splits of genotyped and non-genotyped animals.

I also explored the HETVAR ($\sigma_\alpha^2 \neq \sigma_u^2$) versus HOMVAR ($\sigma_\alpha^2 = \sigma_u^2$) specifications for the ssGBLUP/ssEMMAX model for both within and across station splits of genotyped and non-genotyped animals, focusing only on milk fat. As anticipated, with random within station splits of genotyped and non-genotyped animals, there was no statistical evidence to refute the HOMVAR specification. Now the HETVAR specification may converge very slowly (i.e., may not converge after 50 iterations) as it did so for Partition P2 in Table 4.7, even after using the HOMVAR estimates as joint starting values for σ_u^2 and σ_α^2 . Nevertheless, most partitions did converge within 10 iterations under the HETVAR analysis.

Table 4.7 Likelihood ratio test on $H_0 : \sigma_\alpha^2 = \sigma_u^2$ for within station study in milk fat

Partition	σ_α^2	σ_u^2	*p-value*
P1	0.0174	0.0139	0.35
P2			Did not converge
P3	0.0173	0.0096	0.20
P4	0.0172	0.0167	0.45
P5	0.0171	0.0251	0.20

I conducted the across station comparison of HOMVAR versus HETVAR specifications in two different ways. Firstly, the genotypes of each station were masked, one station at a time for 6 different analyses with likelihood ratio tests provided in Table 4.8. Most analyses either did not converge or led to analyses that failed to reject H_o: $\sigma_\alpha^2 = \sigma_u^2$. However, it was rather curious that the analyses based on masking the genotypes from ISU station indicated that its pedigee-

based assessment of polygenic variance σ_u^2 exceeded the genomic variance σ_α^2 for the other stations. When masking was "flipped", (i.e. all ISU genotypes were available with all other genoytpes masked), then $\sigma_\alpha^2 > \sigma_u^2$ (Table 4.9) confirming that the difference in genetic variability between ISU with the other stations was not only simply due to a difference in scaling between genomic and pedigree-based relationship matrices, but rather because of the heterogeneity of genetic variances across stations. In fact, when the HETVAR specification was used with the ISU genotypes being masked, stronger measures of association for the top SNP and genomic windows were determined using single step extensions of EMMAX and SSVS relative to using the same extensions under the HOMVAR specification (Figure 4.8).

Table 4.8 Likelihood ratio test on $H_0 : \sigma_\alpha^2 = \sigma_u^2$ for milk fat across station splits where respective analysis masked genotypes for research station as indicated below

Station with masked genotypes (# of cows per station)	σ_α^2	σ_u^2	*p-value*
ISU (930)	0.014	0.050	2.29e-08
MSU (264)			Did not converge
USDFRC (347)	0.016	0.027	0.13
UW (780)			Did not converge
FL (377)	0.017	0.006	0.06
AGIL (488)	0.016	0.018	0.40

Table 4.9 Likelihood ratio test on $H_0 : \sigma_\alpha^2 = \sigma_u^2$ for milk fat across station splits where respective analysis masked genotypes on all other research stations except for research station as indicated below

Genotype ***included*** station (# of cows per station)	σ_α^2	σ_u^2	*p-value*
ISU (930)	0.039	0.012	3.81e-06
MSU (264)	0.010	0.027	0.02
USDFRC (347)	0.023	0.022	0.48
UW (780)	0.008	0.032	4.25e-06
FL (377)	0.010	0.026	0.01
AGIL (488)	0.019	0.024	0.20

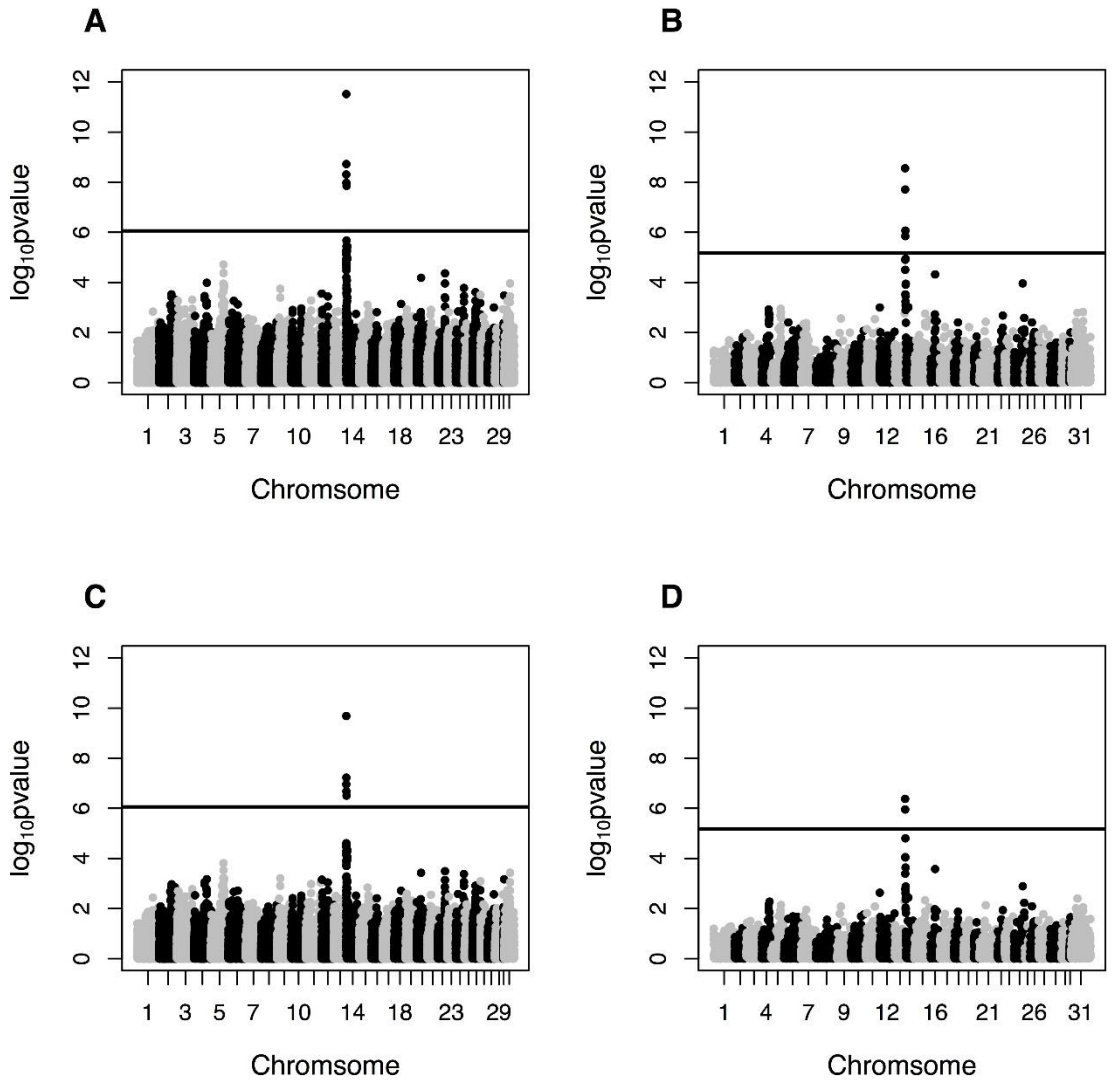

Figure 4.8 Manhattan plot for milkfat masking genotypes ISU cows using ssEMMAX. Panel A: single SNP inferences for HETVAR variance; Panel B: adaptive window inferences for HETVAR variance; Panel C: single SNP inferences for HOMVAR variance; Panel D: adaptive window inferences for HOMVAR variance.

4.6 Discussion

The goal of this study is to evaluate the potential merit of single step extensions for GBLUP and SSVS for WGP and GWA. Based on simulation, the prediction accuracies from using single-step procedures were generally greater within either class of model, i.e.,

ssGBLUP>GBLUP and ssSSVS>SSVS. For traits controlled by a relatively small number of QTL (n_{qtl} = 30 and 300), SSVS always had higher WGP accuracy than GBLUP and, correspondingly, ssSSVS always has higher WGP accuracy than ssGBLUP for genotyped animals as demonstrated by simulation. For more complex traits (n_{qtl} = 3000), there was no evidence that SSVS had a WGP accuracy different from GBLUP, nor ssSSVS from ssGBLUP. For non-genotyped cows, the usage of ssSSVS versus ssGBLUP appeared to be less important with perhaps a small advantage for ssSSVS under simpler genetic architectures (n_{qtl} = 30 and 300) based on the simulation study.

This WGP advantage for single step extensions was also demonstrated by a cross-validation application to milk fat and body weight data from a dairy cattle consortium, albeit the differences there were very small. It might not be too surprising that SSVS also outperformed GBLUP for milkfat which is known to be dominated by DGAT1 on Chromosome 14 (Grisart *et al.* 2002) with similar advantages of Bayesian methods having been found for milk fat in previous studies (Hayes *et al.* 2010). Body weight may be more complex (i.e., effectively more polygenic) than milk fat (Pryce *et al.* 2012) such that there may be less of a distinction between the two models. Nevertheless, SSVS appeared to have a higher cross validation WGP accuracy than GBLUP whereas there was no such evidence of a difference between ssSSVS and ssGBLUP for body weight. There appeared to be no such distinction in cross validation prediction accuracies for non-genotyped cows based on the analysis of either milk fat or body weight in the dairy consortium study; however, ssSSVS had higher prediction accuracy for the non-genotyped cows in our simulation study with n_{qtl} = 30 and 300. The difference between simulation study and real dairy data analysis can be explained by the difference in simulated genetic architecture and QTL distribution in the real dataset. Additionally, in the simulation, I masked 50% of the cows as non-

genotyped, however, in the real dairy dataset, about 20% cow were treated as non-genotyped; it is also reasonable to expect single-step having more advantage with higher non-genotyped/genotyped ratio because more information from phenotypes are incorporated compared to regular approach. Higher difference using single-step Bayesian approach could be observed in other datasets or applications, e.g., Lee *et al.* (2017). Finally, confirming results already previously summarized by Legarra *et al.* (2014), our results suggest that ssGBLUP can lead to higher WGP accuracies than conventional implementations of methods using only genotyped individuals, particularly for more complex traits. For example, ssGBLUP had a higher WGP accuracy than SSVS (using phenotypes on genotyped animals only) for body weight.

I also evaluated the merit of single step extensions for GWA. I determined that ssSSVS had higher pAUC05 than ssEMMAX based on an adaptive window approach to GWA pAUC05 for each different specification of n_{qtl}; however, I detected no such differences for single SNP associations. Thus, I recommend using ssSSVS for GWA based on adaptively selected windows. In fact, the use of genomic window associations lead to pAUC05 values that were often multiples of pAUC05 values derived from single SNP associations, thereby suggesting a proportionately greater number of more true positives using genomic window based associations up until a FPR = 0.05. Strangely enough, ssEMMAX had significantly better pAUC05 performance than EMMAX for more polygenic cases (n_{qtl} = 300 and 3000) using the adaptive window approach but again the differences were very small. For single SNP associations, ssEMMAX was not significantly different from EMMAX whereas ssSSVS was also not significantly different from SSVS for pAUC05. Our simulation suggested there may be little

advantage of using the single-step approach for GWA for single marker associations, at least for Bayesian methods.

Note that pAUC05 determinations are based on the rankings of the -$\log_{10}$(*P-values*) or PPA and not on any thresholds for declaring significance. I suspected that the single-step approach could lead to stronger measures of association than their conventional counterparts. Thus, I compared -$\log_{10}$(*P*-values) or PPA values between EMMAX and SSVS and their single step extensions for milkfat and body weight. For milkfat in a within station split cross-validation study, the peak SNP of ssSSVS was the same as the peak SNP for SSVS for 2 out of 5 partitions, in which SSVS and ssSSVS tied at PPA of 1.00 (Table 4.10). In partition P3 and P4 (Figure B3 and B4 in the Appendix B), ssSSVS had a lower peak for SNP ARS-BFGL-NGS-4939 and Window 3867 (1189.341kb to 1801.116kb on chromosome 14) than SSVS. This may be because some of the PPA were distributed to nearby SNP/region in high LD with ARS-BFGL-NGS-4939. For example, in partition P3, SNP ARS-BFGL-NGS-107379 had a PPA= 0.181 being just in the next Window 3868 for ssSSVS (conventional SSVS had PPA of 0). The LD heatmap in Figure 4.9 showed this SNP was in high LD with the most significant SNP ARS-BFGL-NGS-4939. A similar situation occurred for conventional SSVS in P5, where the PPA was distributed to another SNP ARS-BFGL-NGS-57820 with PPA of 0.708 (ssSSVS had PPA of 0 for this SNP) in the same window that was almost in very high LD with the originally most significant marker SNP ARS-BFGL-NGS-4939. This indicates window based approach can somewhat mitigate the multicollinearity issue well for SNPs in high LD within the same window, but for long range LD, the window based approach might still be affected by this problem. Therefore, the 'flanking' window approach (Fernando *et al.* 2017) can be applied for these type of issues.

Table 4.10 Full results of within station analyses for PPA in SSVS for most significant SNP/genomic region using single-step compared to conventional specifications on milkfat and body weight

	Milk fat					Body weight			
	Single SNP		Adaptive window			Single SNP		Adaptive window	
Partition	SSVS	ssSSVS	SSVS	ssSSVS	Partition	SSVS	ssSSVS	SSVS	ssSSVS
P1	1	1	1	1	P1	0.485	0.424	0.773	0.812
P2	1	1	1	1	P2	0.269	0.314	0.590	0.668
P3	1	0.820	1	0.820	P3	0.244	0.231	0.555	0.673
P4	1	0.753	1	0.753	P4	0.287	0.269	0.634	0.631
P5	0.213	1	0.921	1	P5	0.315	0.342	0.656	0.607

The single SNP for milkfat and body weight are ARS-BFGL-NGS-4939 and BTB-01412391 correspondingly. The adaptive window for milk fat ranges from 1189.341kb to 1801.116kb in chromosome 14; and the adaptive window for body weight ranges from 7104.14888350.890-7342.69688668.261kb in chromosome 14.

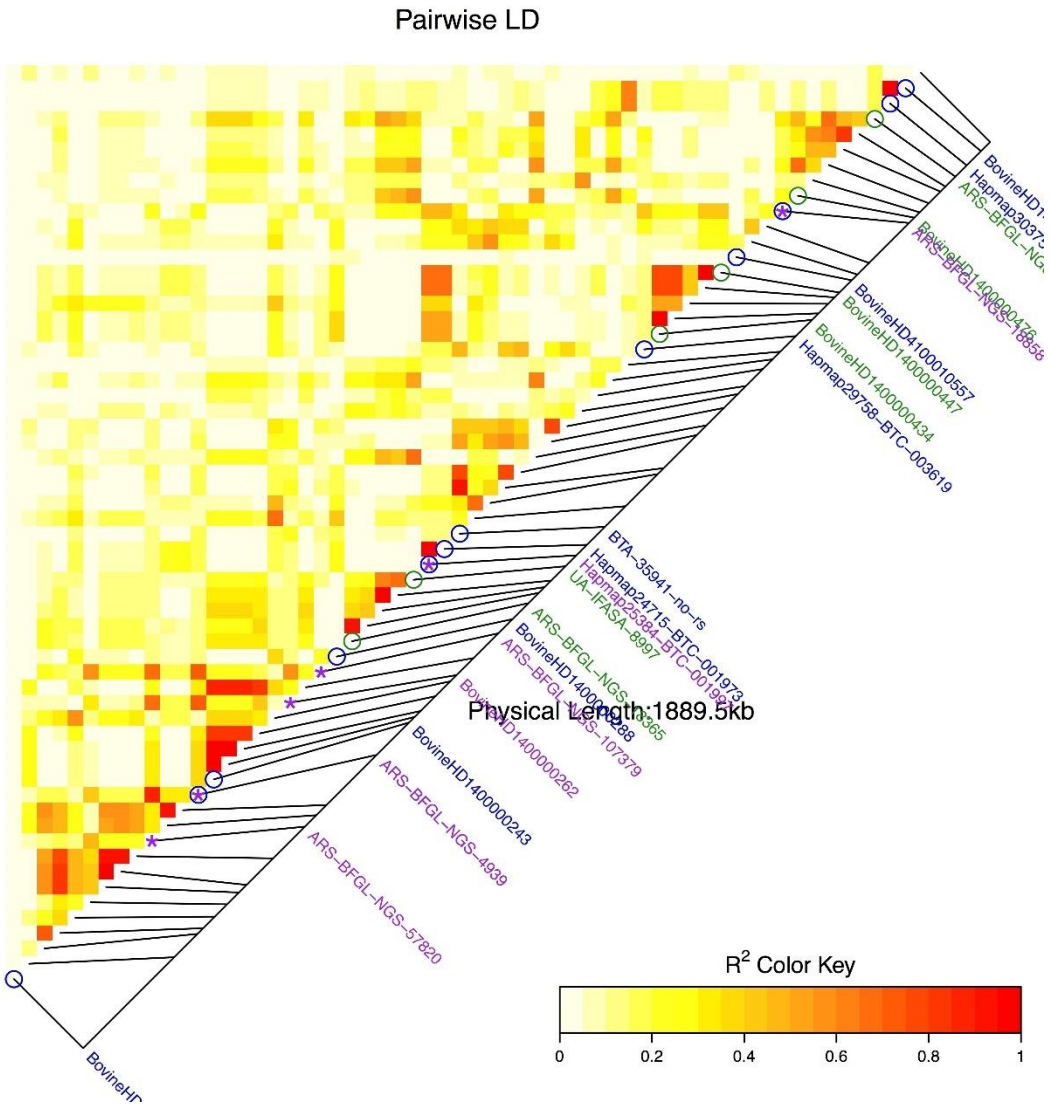

Figure 4.9 LD (r^2 metric) heatmap for chromosome 14 from 1189.341kb to 3059.698kb that contains all the SNP and windows selected by EMMAX based on the benchmark. Purple star mean are all the SNPs that deem to be significant by EMMAX; blue circle is the starting and ending SNPs for the windows deem to be significant by EMMAX; green circle is the starting and ending SNPs for other windows in the map.

For cross-validation assessments based on across station splits of the data, the assessment of single step extensions for both models appeared to be substantially more complicated. For two out of the 6 partitions, where the genotypes of stations ISU and USDFRC were each masked in turn, weaker measures of association using ssEMMAX for both top SNP with -$\log_{10}$(*P-values*) of

8.34 and 8.63 and top window with -$\log_{10}$(*P-values*) of 5.36 and 4.77 inferences were found compared to EMMAX with -$\log_{10}$(*P-values*) of 9.68 and 9.58 for top SNP, and -$\log_{10}$(*P-values*) of 6.99 and 5.58 for top window, which simply ignored the phenotypes on those respectively masked genotyped cows (Figure B6 and Figure B8). The reason might be the HOMVAR specification did not model the variance components correctly. For the top SNP (Table 4.11), ssSSVS was observed to have lower PPA of top SNP than SSVS for 1 out of 6 partitions (Figure B9), but this might have been caused by PPA being distributed between SNP ARS-BFGL-NGS-57820 with PPA of 0.207 and SNP ARS-BFGL-NGS-4939 with PPA of 0.290 in high LD in the same window. For the top window, ssSSVS always had a PPA higher than or equal to SSVS (3 out of 6 partitions tied). In the across station study, SSVS was more likely re-distribute PPA to other SNPs/regions because the estimated/observed r^2 of the genotyped cows in the cross-validation study might be different from benchmark population, whereas ssSSVS might have been more stable because it uses pedigree information to 'imputed' genotypes for non-genotyped cows. GWA inferences using ssSSVS may highly depend upon the kinship relationship between genotyped and non-genotyped cows.

For body weight, I noticed a slightly higher PPA for ssSSVS by just comparing the top window in the benchmark compared to SSVS for both within and across station analyses (Table 4.6). Furthermore, in Table 4.11, ssSSVS had a higher PPA than SSVS in 5 out 6 across station partitions, meaning that ssSSVS tends to do a better job in preserving the PPA of the top window from the benchmark. However, more reranking of top SNP markers or genomic windows occur for single SNP associations using ssEMMAX, and for both single SNP and window based associations using ssSSVS, relative to the benchmark analyses. The measures of strength of

association were often no different in EMMAX and ssEMMAX for both within and across station splits of genotyped and non-genotyped animals.

Table 4.11 Full results of across station analyses for PPA in SSVS for most significant SNP/genomic region using single-step compared to conventional specifications on milkfat and body weight

	Milk fat					Body weight			
	Single SNP		Adaptive window			Single SNP		Adaptive window	
Excluded herd	SSVS	ssSSVS	SSVS	ssSSVS	Excluded herd	SSVS	ssSSVS	SSVS	ssSSVS
ISU	0.426	1	0.442	1	ISU	0.275	0.372	0.641	0.672
MSU	0.335	1	0.633	1	MSU	0.631	0.648	0.698	0.753
USDFRC	1	1	1	1	USDFRC	0.209	0.184	0.702	0.683
UW	0.401	0.290	0.404	0.505	UW	0.128	0.149	0.372	0.397
FL	1	1	1	1	FL	0.465	0.533	0.762	0.816
AGIL	1	1	1	1	AGIL	0.312	0.361	0.869	0.884

The single SNP for milkfat and body weight are ARS-BFGL-NGS-4939 and BTB-01412391 correspondingly. The adaptive window for milk fat ranges from 1189.341kb to 1801.116kb in chromosome 14; and the adaptive window for body weight ranges from 7104.14888350.890-7342.69688668.261kb in chromosome 14.

The HOMVAR assumption, which considers marker based genomic variance and pedigree based genetic variance to be the same, is widely used in almost all current ssGBLUP applications, whether for genetic evaluations (Legarra *et al.* 2014) or for GWA (Wang *et al.* 2012; Zhang *et al.* 2016). Our analysis, particularly based on across station splits for cross-validation, suggests I should use this assumption carefully as the two variance components, σ_α^2 and σ_u^2, can be quite different (Table 4.8). In addition to excluding one station at a time, I also conducted likelihood ratio tests for the reverse situations, including one station at a time (or excluding 5 stations at a time), with some of the tests again suggesting that σ_α^2 can be different from σ_u^2 (Table 4.9), with the reversal in magnitude suggesting that the issue pertains to true

heterogeneity of genetic variances across herds rather than differences in scaling between pedigree versus genomic based relationship matrices. For example, for ISU, the test of treating it as non-genotyped or treating it as the only genotyped station both suggest that the genetic variation of ISU is different from other 5 stations. It then seems reasonable to estimate σ_α^2 and σ_u^2 separately in some cases, particularly when some herds contributing data are exclusively genotyped or non-genotyped animals. It seems reasonable and necessary to consider modeling herd-specific heterogeneity in both σ_α^2 and σ_u^2 jointly with herd-specific heterogeneity in σ_e^2 as recently explored by Ou *et al.* (2016) as WGP and GWA inferences could be quite sensitive to those specifications.

Currently the single-step Bayesian model are based on computing the 'imputed' genotypes for non-genotyped animal based on $\mathbf{A}^{nn}\hat{\mathbf{T}}_n = -\mathbf{A}^{ng}\mathbf{T}_g$. This procedure can be both memory and CPU demanding when number of non-genotyped animal is large because $\hat{\mathbf{T}}_n$ is not sparse. Fernando *et al.* (2016) provided an algorithm that avoids storage and multiplication of $\hat{\mathbf{T}}_n$ such that the number of non-genotyped animal is less of concern. Another important advantage of Bayesian single-step approach is the flexibility to use any prior to accommodate different genetic architectures and extension to all existing 'Bayesian Alphabet'. Recently, Lee *et al.* (2017) applied two single-step Bayesian regression (SSBR) models to Hanwoo beef cattle, in which they found SSBR lead to higher WGP accuracy than ssGBLUP for trait associated with small number of QTLs with large effect (similar to our milkfat trait with DGAT1 and n_{qtl}=30 or 300 in the simulation study) and no disadvantages of SSBR were found for all other traits (similar to our body weight trait and n_{qtl}=3000 in the simulation study). Therefore, based on this study and our

results, single-step Bayesian models are promising for WGP analysis with different types of traits.

Another important factor to consider in ssSSVS and SSVS is the specification of the hyperparameter π_ϕ. It is well known that hyperparameter specifications can significantly influence WGP accuracies (Lehermeier *et al.* 2013; Yang *et al.* 2015b). However, such hyperparameters can be difficult to estimate with large number of SNP marker using MCMC. Lee *et al.* (2017) demonstrated that specifications for hyperparameters like π_ϕ can be effectively determined using cross-validation, recognizing that poorly estimated or miss-specified π_ϕ may lead to inferior WGP than ssGBLUP for some traits. The previous study comparing ssGBLUP and Bayesian model in WGP, such as in Lourenco *et al.* (2013), might be somewhat flawed as the proportion of non-zero effect (similar to our π_ϕ) in the Bayesian was arbitrarily set to 0.04 without any prior assessment due to cross-validation or estimation. In the simulation study, the average estimated π_ϕ is 0.051 (with bad mixing) and it is observed that fixed π_ϕ of 0.01 resulted in higher pAUC05 for both SSVS and ssSSVS compared to fixed π_ϕ of 0.001. Moreover, fixed π_ϕ of 0.001 led to non-intuitive results where ssSSVS had lower pAUC05 than SSVS. Overall, miss specification of π_ϕ might lead to inferior GWA results (Figure 4.5). Furthermore, I noticed π_ϕ might be also an important factor in GWA because the smaller π_ϕ is, the fewer SNPs/regions will be selected and such that lower π_ϕ is more likely to force few loci to stand out. Whether or not WGP cross-validation based determinations for π_ϕ for GWA is optimal and how to effectively specify such hyperparameter for GWA for Bayesian variable selection models require further study.

Weighted ssGBLUP (WssGBLUP) has been proposed by Wang *et al.* (2012) and Zhang *et al.* (2016), using proportion of variance explained as indicator for GWA. I did not consider this approach for two reasons: 1) it does not facilitate a method for assessment for statistical significance; 2) the methods suffer from convergence difficulties such that WssGBLUP is typically stopped after a fixed number of iterations.; however, choosing the number of such iterations is quite arbitrary.

4.7 Summary and Conclusions

In conclusion, I determine that ssSSVS has higher WGP prediction accuracy than ssGBLUP for simpler genetic architectures, i.e., traits controlled by few major genes. The use of phenotypes on non-genotyped animals is important regardless of model (SSVS or GBLUP) or genetic architecture (simple or complex). The choice of model seems to be more important than use of phenotypes on non-genotyped animals for GWA based on results from our simulation study. Based on applications to data from a dairy consortium, single-step extensions for milk fat were deemed to be more useful than for body weight for WGP. Single-step extensions for SSVS for genomic windows adaptively determined using LD seem to be particularly useful for GWA.

Chapter5 BATools: A Hierarchical Modeling R Package for Genome Prediction and Genome-wide Association Analysis

5.1 Abstract

Whole genome prediction (WGP) and genome-wide association (GWA) analyses are being extensively used in animal breeding and other quantitative genetic applications. Both types of analyses are typically characterized by high dimensional inference based on thousands of SNP markers. Bayesian regression methods have been developed to address such problems by providing shrinkage based inference and variable selection. The `BATools` R-package (https://github.com/chenchunyu88/BATools) implements a collection of such Bayesian regression tools as well as genomic best linear unbiased prediction (GBLUP) for both WGP and GWA. Features of `BATools` include the incorporation of phenotypes of non-genotyped individuals using pedigree information, performing windows based GWA, and modeling correlation between adjacent SNP using a first order antedependence correlation assumption. Algorithm choices range between the use of Monte Carlo Markov Chain samplers or analytical approximations based on the use of the EM algorithm along with restricted maximum likelihood (REML) like estimators of variance components. The software is efficiently implemented utilizing C/C++ code for the most time-consuming computations. The focus of this article is to discuss the models in `BATools` and their usage in real-data analysis.

5.2 Introduction

Whole genome prediction (WGP) utilizing dense single nucleotide polymorphism (SNP) marker information has been increasingly adopted in animal and plant breeding as an important tool for genomic selection on economically important traits (de Los Campos *et al.* 2013). WGP has transformed traditional best linear unbiased prediction (BLUP) estimates of breeding values

(EBV) based on individual records and pedigree relationship into genomic EBV (GEBV) using the SNP marker panels (Meuwissen *et al.* 2016). Two broad categories of hierarchical linear parametric models are available for WGP. One is genomic BLUP (GBLUP) based on linear mixed model with a genomic relationship matrix created from SNP markers to specify the correlation between random individual effects with restricted maximum likelihood (REML) being used to estimate the underlying variance components (VanRaden 2008). The other is hierarchical Bayesian models which uses more flexible prior specifications on SNP marker effects, e.g. a scaled Student *t* (BayesA) (Meuwissen *et al.* 2001), mixture of scaled *t* and point mass at zero (BayesB) (Meuwissen *et al.* 2001) or stochastic search variable selection (SSVS) (George and McCulloch 1993; Chen and Tempelman 2015) amongst several others. The major algorithm used for inference in these Bayesian models is Markov Chain Monte Carlo (MCMC) based almost entirely on the use of the Gibbs sampler (Casella and George 1992).

Genome-wide association (GWA) analysis, on the other hand, is a useful tool to identify SNP markers or genomic regions that are associated with causal variants or quantitative trait loci (QTL). In a simple way, GWA is testing the null hypothesis that a marker or region has no effect with respect to a trait with tests running across all SNP markers/genomic regions. Similar to WGP, GWA has been based on the same two broad categories of models. A popular strategy, EMMAX, treats the SNP marker of interest as fixed with all other marker effect as random effects to account for population structure through traditional GBLUP-like or mixed effects models (Kang *et al.* 2010). Additional modifications and computational enhancements have been described elsewhere (Lippert *et al.* 2011; Zhou and Stephens 2012; Gualdron Duarte *et al.* 2014). But more recently, hierarchical Bayesian models have also been implemented for GWA based posterior probability of associations (PPA) (Moser *et al.* 2015; Fernando *et al.* 2017). Chen *et al.* (2017) extended both types of models for genomic region based GWA.

With more individuals genotyped over the last decade, yet with many if not most individuals not yet genotyped for various reasons, there has an increasing interest to combine the phenotypic information from both genotyped and non-genotyped individuals to improve accuracy of GEBV and GWA. The single step approach is one such model that utilizes genotype, phenotype, and pedigree information of both genotype and non-genotyped individuals in one single model; a previous strategy based on blending pedigree based EBV with GEBV in a "two-step" model (VanRaden 2008). The original single step WGP analyses was based on the GBLUP assumptions and hence known as ssGBLUP (Aguilar *et al.* 2010). Recently, Fernando *et al.* (2014) extended single step approach to allow for more flexible hierarchical Bayesian modeling assumptions. A recent study by Lee *et al.* (2017) and the work in Chapter 4 indicated that single step Bayesian models inherit the same favorable properties of regular Bayesian WGP models and even increased WGP accuracies for trait controlled by fewer number of QTLs. Chapter 4 also demonstrated that single step Bayesian models had better GWA performance than single step EMMAX extension for window based, as opposed to single SNP, inferences.

With either category (BLUP or Bayesian) of model, the marker effects are typically specified to be independently distributed. Gianola *et al.* (2003) conjectured that some of SNP marker effects might be spatially correlated within chromosomes. The Bayesian antedependence models proposed by Yang and Tempelman (2012) that extended BayesA and BayesB and modeled nonstationary spatial correlations between adjacent SNP markers, known respectively as anteBayesA and anteBayesB, leading to higher WGP accuracies with higher LD (r^2>0.24) marker panels. These results have been further corroborated by others in a multiple trait modeling context (Jiang *et al.* 2015). Yang and Tempelman (2012) and Tempelman (2015) also suggested potential benefit for sharper GWA signals using such models.

In summary, both WGP and GWA analyses are typically characterized by much larger *m* (number of SNP markers) relative to *n* (number of individual). This issue has been addressed with hierarchical linear models based on either traditional mixed model or Bayesian approaches. It seems important to develop a software package that provides a user-friendly interface for a variety of different models with full support for both WGP and GWA.

Currently, there are many different software packages for WGP or GWA or both, but most of them have a specific focus. BLUPF90 (Misztal *et al.* 2002) is a collection of software programs written in FORTRAN for GBLUP models including popular single-step GBLUP (ssGBLUP) that combines phenotypes on genotyped animals and on non-genotyped animals with pedigree information (Aguilar *et al.* 2010). Although BLUPF90 is efficient and suitable for analysis on large dataset (i.e. genomic evaluation with more than 1 million records), it does not allow for Bayesian analyses, and its weighted single-step approach suffers from convergence issues, leading thereby to heuristic solutions (Zhang *et al.* 2016). Furthermore, the GWA inferences in BLUPF90 programs do not provide formal measures of statistical significance (e.g. *P*-value) (Wang *et al.* 2012; Zhang *et al.* 2016). Gensel (Fernando and Garrick 2009) is a web-based analysis of genomic data platform that features a large selection of different Bayesian models for both WGP and GWA, but it is not available for public distribution. `BLR` (Perez *et al.* 2010) and `BGLR` (Perez and de los Campos 2014) are sister R-packages that implements various Bayesian and nonparametric models concentrated on WGP. While `BGLR` provides large selection of models and support different type of traits (continuous or categorical), they do not provide enough support for GWA. `rrBLUP` (Endelman 2011) is also an R-package that implements the GBLUP model for both WGP and GWA with a user-friendly interface compared to BLUPF90, but it does not support Bayesian analyses. `synbreed` (Wimmer *et al.* 2012) is a nice R-package

that provides rich data management and cleaning tools for WGP and GWA for animal and plant breeding, however, their model fitting is done through other software packages such as `BGLR`. `GEMMA` (Zhou and Stephens 2012) is a group of efficient tools written in C++ focus on GWA, although it support Bayesian WGP, the model is based on one particular prior for marker effects such that the option for the users is really limited for WGP. `JWAS` (Cheng *et al.* 2016) is an open-source software tool written in Julia (Bezanson *et al.* 2012) for Bayesian models applied to WGP and GWA. `JWAS` provides models such as BayesB and BayesC (Habier *et al.* 2011) as well as their single step extensions and it is the currently the only single step Bayesian software implementation publicly available. However, Julia is a new programming language and does not have the large user community. The package is also not fully documented and does not provide enough support for GWA in the documentation.

Although these software packages implement a few different types of hierarchical linear models, there is currently no known WGP/GWA open source R (R Core Team 2017) packages for single step approach that utilize genotype, phenotype and pedigree information of both genotype and non-genotyped individuals. I have developed R-package `BATools` to implement such an approach. Along with single-step, `BATools` also implement some other Bayesian model extensions/improvement that are not currently public available, including Bayesian antedependence models for spatial correlations between adjacent SNP markers (Yang and Tempelman 2012), a computationally tractable empirical Bayes approach for BayesA/SSVS based on Expectation–Maximization (EM) algorithm, and a window/region based approach for joint testing of SNP marker effects in GWA using a fast version of EMMAX (Gualdron Duarte *et al.* 2014; Bernal Rubio *et al.* 2016; Chen *et al.* 2017) and Bayesian extensions for GWA. The package includes a collection of models in a unified framework for genomic data analysis is

available on Github (https://github.com/chenchunyu88/BATools) and will shortly be available on CRAN. The objective of this paper is to demonstrate the models, algorithms and data implemented in the package, to present some example analyses demonstrating some key specifications and to provide a benchmark of computing time for the package.

5.3 Statistical Models and Algorithms

The `BATools` package currently supports the analysis of continuous traits. For both WGP and GWA, the base model can be presented as:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{T}\boldsymbol{\alpha} + \mathbf{e} \qquad [5.1]$$

with

$$\mathbf{T} = \frac{\left(\mathbf{M} - \mathbf{1k}'\right)}{\sqrt{\sum_{j=1}^{m} 2p_j\left(1-p_j\right)}} \qquad [5.2]$$

Here $\mathbf{y}$ is a n x 1 vector of phenotypes, $\mathbf{X}$ is a known n x p incidence matrix connecting $\mathbf{y}$ to the p x 1 vector of unknown fixed effects or/and covariates $\boldsymbol{\beta}$, $\mathbf{T}$ is a known n x m standardized matrix of genotypes connecting $\mathbf{y}$ to the m x 1 vector of unknown random SNP marker effects $\boldsymbol{\alpha}$, and $\mathbf{e}$ is the random error vector. $\mathbf{M}$ is the original n x m genotype matrix with elements coded as "0, 1, 2". Furthermore, element j of the m x 1 vector $\mathbf{k}$ is the mean value ($2p_j$) for the corresponding column of $\mathbf{M}$, such that p_j is the allele frequency of the reference allele of SNP marker j =1,2,…,m (VanRaden 2008). Recoding genotypes in this manner has been demonstrated to improve algorithmic stability (Stranden and Christensen 2011). This model can be also written as a subject-centric model (Henderson 1985):

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u} + \mathbf{e} \qquad [5.3]$$

Here $\mathbf{u} = \mathbf{T}\boldsymbol{\alpha}$ is the additive genetic effect of each subject. I also assume throughout that $\mathbf{e} \sim N(0, \mathbf{I}\sigma_e^2)$. If I assume $\boldsymbol{\alpha} \sim N(0, \mathbf{I}\sigma_\alpha^2)$, then $\mathbf{u} \sim N(0, \mathbf{G}\sigma_\alpha^2)$ with $\mathbf{G} = \mathbf{T}\mathbf{T}'$. Then the mixed model equation (MME) corresponding to the linear models in [5.1] and [5.3] are Equations [5.4] and [5.5] respectively.

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{T} \\ \mathbf{T}'\mathbf{X} & \mathbf{T}'\mathbf{T} + \mathbf{I}\sigma_e^2\sigma_\alpha^{-2} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{\alpha}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{T}'\mathbf{y} \end{bmatrix} \quad [5.4]$$

and

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{I} \\ \mathbf{I}'\mathbf{X} & \mathbf{I}'\mathbf{I} + \mathbf{G}^{-1}\sigma_e^2\sigma_\alpha^{-2} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{I}'\mathbf{y} \end{bmatrix} \quad [5.5]$$

The variance components (σ_α^2 and σ_e^2) in MME [5.4] and [5.5] can be estimated using Average Information REML (AIREML) (Gilmour *et al.* 1995; Johnson and Thompson 1995) with solutions for [5.4] and [5.5] often referred to as GBLUP (VanRaden 2008). In fact, model [5.1] and [5.3] are equivalent ($\hat{\boldsymbol{\alpha}} = \mathbf{T}'\mathbf{G}^{-1}\hat{\mathbf{u}}$) with equation [5.3] being preferred for computing efficiency when $m \gg n$ (Stranden and Garrick 2009).

In a Baysian context, I use priors instead and the residual variance has a scale-inverse χ^2 prior, i.e., $\chi^{-2}\left(\sigma_e^2 \mid v_e, v_e s_e^2\right)$ with degrees of freedom $v_e = -1$ and scale $s_e^2 = 0$ as default by `BATools` and can be changed by user. The fixed effects $\boldsymbol{\beta}$ are assigned with flat priors.

5.3.1 Priors for marker effects

All hierarchical linear models are based directly on equation [5.1] by assigning structural priors on SNP marker effects. Different types of Bayesian models differ from each other in the prior distributions for SNP marker effects $\boldsymbol{\alpha}$; therefore, different prior selections may influence WGP accuracy and GWA performance since they provide different shrinkage properties for

marker effects . Figure 5.1 provides a visualization of four types of base priors were implemented for the SNP marker effects in BATools: using the Gaussion prior often referred as Bayesian ridge regression (BRR) (Hoerl and Kennard 1970); a scaled-t distribution prior known as BayesA, which can be written as normal mixture of scaled inverse χ^2 or Gamma (Meuwissen *et al.* 2001); a mixture of point mass at zero and scaled-t prior known as BayesB (Meuwissen *et al.* 2001); and a mixture of two Gaussion densities known as SSVS (George and McCulloch 1993; Chen *et al.* 2017). In addition to those prior specifications, I also implemented the antedependence models, i.e. anteBayesA and anteBayesB, to model the spatially-induced correlations between adjacent SNP marker within the same chromosome (Yang and Tempelman 2012). Full details of the statistical expressions about these prior distributions are provided in Table 5.1.

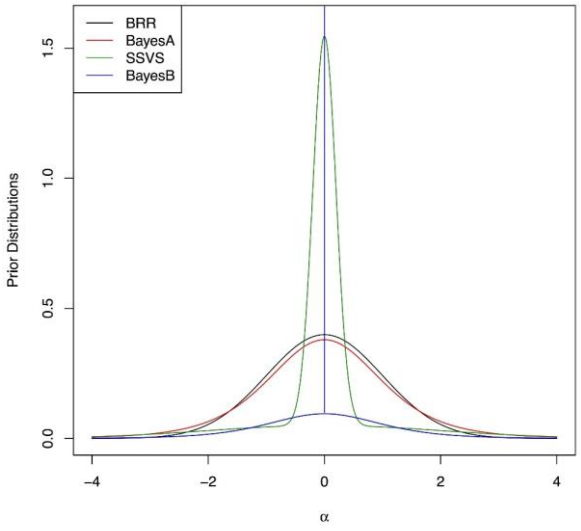

Figure 5.1 Visualization of prior distributions for SNP marker effects in BATools.

Table 5.1 List of models in BATools and their priors and hyperparameters

Model	Marker effect Priors	Hyperparameters treatment
BRR	$\alpha_j \sim N(0, \sigma_\alpha^2)$	$\sigma_\alpha^2 \sim \chi^{-2}(-1, 0)$
BayesA[1] anteBayesA[2] ssBayesA[3]	$\alpha_j \sim N(0, \sigma_{\alpha_j}^2)$ $\sigma_{\alpha_j}^2 \sim \chi^{-2}(\nu_\alpha, \nu_\alpha s_\alpha^2)$	$\nu_\alpha \propto (1+\nu_\alpha)^{-2}$ $s_\alpha^2 \sim Gamma(\alpha_s, \beta_s)$
BayesB[1] anteBayesB[2] ssBayesB[3]	$\alpha_j \sim N(0, \sigma_{\alpha_j}^2)$ $\sigma_{\alpha_j}^2 \begin{cases} = 0 & 1-\pi_\alpha \\ \sim \chi^{-2}(\nu_\alpha, \nu_\alpha s_\alpha^2) & \pi_\alpha \end{cases}$	$\nu_\alpha \propto (1+\nu_\alpha)^{-2}$ $s_\alpha^2 \sim Gamma(\alpha_s, \beta_s)$ $\pi_\alpha \sim Beta(\alpha_\pi, \beta_\pi)$
SSVS[4] ssSSVS[5] mapSSVS[6]	$\alpha_j = N(0, (\tau_j + (1-\tau_j)/c)\sigma_\alpha^2)$ $\tau_j \sim Bernoulli(\pi_\tau); c \geq 1$	$\sigma_\alpha^2 \sim \chi^{-2}(-1, 0)$ $\pi_\tau \sim Beta(\alpha_\pi, \beta_\pi)$
mapBayesA[6]	$\alpha_j \sim N(0, \sigma_\alpha^2 \tau_j)$ $\tau_j \sim \chi^{-2}(\nu_\alpha, \nu_\alpha)$	$\sigma_\alpha^2 \sim \chi^{-2}(-1, 0)$ ν_α Fixed
Antedependence[2]	$t_{j,j-1} \sim N(\mu_t, \sigma_t^2)$	$\mu_t \sim N(\mu_{t0}, \sigma_{t0}^2)$ $\sigma_t^2 \sim \chi^{-2}(\nu_t, \nu_t s_t^2)$

[1]Meuwissen *et al.* (2001); [2]Yang and Tempelman (2012); [3]Fernando *et al.* (2014) and Chapter 3; [4]George and McCulloch (1993); [5]Chapter 3; [6]Chen and Tempelman (2015) and Chen *et al.* (2017).

5.3.2 Single-step for BayesA/B and SSVS

A ssGBLUP approach was originally developed to combine phenotypes on genotyped and non-genotyped animals with pedigree information (Aguilar *et al.* 2010) and has been applied to many livestock species (Legarra *et al.* 2014). The single-step approach for Bayesian WGP was first proposed by Fernando *et al.* (2014) to include genotyped and non-genotyped individuals:

$$\begin{bmatrix} \mathbf{y}_n \\ \mathbf{y}_g \end{bmatrix} = \begin{bmatrix} \mathbf{X}_n \\ \mathbf{X}_g \end{bmatrix} \boldsymbol{\beta} + \begin{bmatrix} \hat{\mathbf{T}}_n \boldsymbol{\alpha} + \boldsymbol{\varepsilon} \\ \mathbf{T}_g \boldsymbol{\alpha} \end{bmatrix} + \mathbf{e} \quad [5.6]$$

Here the linear equation [5.3] partition the non-genotyped and genotyped individuals using subscripts *n* and *g*. Other terms stay the same as with equation [5.1] except that $\hat{\mathbf{T}}_n$ in Equation [5.6] is an "imputed" genotype matrix for the non-genotyped individuals that can be obtained by solving $\mathbf{A}^{nn}\hat{\mathbf{T}}_n = -\mathbf{A}^{ng}\mathbf{T}_g$, where $\mathbf{A}^{nn}$ and $\mathbf{A}^{ng}$ are the partitions of $\mathbf{A}^{-1}$ (inverse of the additive relationship matrix based on pedigree) corresponding to non-genotyped by non-genotyped and non-genotyped by genotyped animals, respectively (Fernando *et al.* 2014). The imputation residuals $\boldsymbol{\varepsilon} \sim N\left(\mathbf{0}, (\mathbf{A}^{nn})^{-1}\sigma_u^2\right)$ accounts for contributions of pedigree information to breeding values for non-genotyped animals (Fernando *et al.* 2014). In Chapter 4, I also demonstrated how to apply single-step SSVS (ssSSVS) to a simulated dataset and a USDA dairy consortium dataset. A similar implementation for updating $\boldsymbol{\varepsilon}$ was used for single-step BayesA/B (ssBayesA/B) while the rest parameters were updated the same with conventional BayesA/B. Lee *et al.* (2017) and my work in Chapter 4 recently determined that single step Bayesian models led to better WGP performance than ssGBLUP for trait controlled by few QTLs with large effects with no evidence of a disadvantage for other types of genetic architectures.

5.3.3 Antedependence implementation

The antedependence models use a vector of association variables to model serially correlated SNP markers in a nonstationary manner (Yang and Tempelman 2012). The model extends equation [5.1] such that

$$\alpha_j = \begin{cases} \delta_1 & \text{if } j = 1 \\ t_{j,j-1}\alpha_{j-1} + \delta_j & \text{if } 2 \le j \le m \end{cases} \quad [5.7]$$

Here $\delta_j \sim N(0, \sigma^2_{\delta_j}),\ j = 1, \ldots m$ and $t_{j,j-1} \sim N(\mu_t, \sigma^2_t)$ is the marker interval-specific antedependence parameter (Zimmerman and Nunez-Anton 2010) of α_j on α_{j-1}. Note that $t_{j,j-1}$ is set to zero at the end of each chromosome. Yang and Tempelman (2012) determined that anteBayesA and anteBayesB improved WGP accuracy compared to BayesA and BayesB for population with high LD levels (r^2 >0. 24) and would lead to even greater accuracies with higher density SNP marker panels. Antedependence models could also have potential benefits in GWA.

5.3.4 Algorithms

The majority of the models were implemented using MCMC via Gibbs sampler (Casella and George 1992) for updating marker effects. In the meantime, the hyperparameters such as v_α and s^2_α should be updated in each MCMC iteration to maximize accuracy of WGP (Yang *et al.* 2015b; Zhu *et al.* 2016). However, even if it is possible to estimate these hyperparameters using MCMC, the poor mixing for of hyperparameters may require a long MCMC chain for convergence to the joint posterior density in equilibrium with subsequently slow mixing for high density marker panels, making its implementation less practical for real data analysis. Therefore, `BATools` adopts a univariate Metropolis-Hastings (UNIMH) algorithm, which substantially improved mixing of MCMC chain, instead of Gibbs sampler for v_α and s^2_α to help mixing when both need to be updated in (ante)BayesA/B (Yang *et al.* 2015b). Nevertheless, with large number of SNP markers, even these improvements may still require a significant amount of computing time. A maximum a posterior (MAP) approach that analytical estimates the marker effects and hyperparameters was also implemented for BayesA and SSVS, known as MAP-BayesA and MAP-SSVS for WGP (Chen and Tempelman 2015). Both MAP-BayesA and MAP-SSVS require computing time comparable to GBLUP, however, they may lead to slightly lower

WGP accuracy than MCMC counterparts because of the possibility of converging to local maximum.

5.3.5 GWA implementation

Traditionally, GWA has been based on using single SNP tests. GWA studies are increasingly based on joint tests on SNP markers within pre-defined genomic windows rather than just tests on single SNP marker as single SNP marker tests may have low statistical power or adversely affected by multicollinearity or both (Chen *et al.* 2017; Fernando *et al.* 2017). A recent study by Chen *et al.* (2017) illustrated that adaptive window based on LD (Dehman *et al.* 2015) could have better GWA performance than using fixed window length or single SNP approach for Bayesian analyses. BATools provides window based GWA using Bayesian posterior probability or chi-square tests for 'Bayesian Alphabet' or MAP based approaches correspondingly (Chen *et al.* 2017; Fernando *et al.* 2017). Previously, ssGBLUP did not provide formal statistical evidence of association in GWA analyses, but merely point estimates of SNP estimates or percentage of genetic variance explained by sliding windows of SNP markers (Wang *et al.* 2012; Zhang *et al.* 2016). Formal tests were provided in Chapter 4 for both single-SNP and window based approaches to GWA inference. A summary for all models providing GWA are listed in Table 5.2.

Table 5.2 GWA output for different models for single SNP and window based approaches

Model	Single SNP	Window
BRR	Bayesian *p*-value[1]	Posterior probability[2]
BayesA, anteBayesA, ssBayesA	Bayesian *p*-value[1]	Posterior probability[2]
BayesB, anteBayesB, ssBayesB	Posterior probability[3]	Posterior probability[2]
SSVS, ssSSVS	Posterior probability[3]	Posterior probability[2]
GBLUP, ssGBLUP (EMMAX)	*p*-value[4,5]	*p*-value[3,5]
mapBayesA, mapSSVS	*p*-value[3]	*p*-value[3]

[1]Bello *et al.* (2010); [2]Fernando *et al.* (2017); [3]Chen *et al.* (2017); [4]Gualdron Duarte *et al.* (2014); [5]Chapter 4.

5.4 Data

The BATools package comes with a subset of MSUPRP data used in `gwaR` package (https://github.com/steibelj/gwaR) to demonstrate GWA using fast version of EMMAX in Gualdron Duarte *et al.* (2014), where they provided a strategy that fit Equation [5.3] once and derived equivalent tests to EMMAX (Bernal Rubio *et al.* 2016). I choose this dataset because it's in relatively small enough to allow a quick demonstration and contains all the phenotype, pedigree, genomic map and genotype information that is required for all the models included in BATools. The original dataset was described in (Gualdron Duarte *et al.* 2014). The subset of data contains 176 Duroc-Pietrain F2 crosses that are both phenotyped and genotyped with 20597 SNP markers. The subset of data come as `synbreed` data object, I pre-processed the data to create genomic window based on LD using `BALD` R package (Dehman and Neuvial 2015) and details of constructing such window is provided in Figure C.1 in Appendix C. The `Pig` data contains objects in Figure 5.2. `PigPheno` is a data.frame of phenotypes and its first column is trait driploss used for domenstration; `PigM` is marker genotypes coded as '0, 1, 2'; `PigMap` is the genomic map for each SNP with column `chr` (chromosome number), `pos` (position in Mb), and `idw` (window id based on BALD or user can create fixed size windows using `set.win` function); `PigAlleleFreq` is the allele frequency of genotype coded as '1' from F0 population; and `PigPed` is a data.frame of pedigree with the first column to be individual ID, second column to be sire ID and third column to be dam ID (unknown sire and dam must be `NA`).

```
rm(list=ls())
library(BATools)
data(Pig)
ls()
## [1] "PigAlleleFreq" "PigM"          "PigMap"        "PigPed"
## [5] "PigPheno"
```

Figure 5.2 Loading Pig data included in BATools

5.5 Interface and application examples

`BATools` is designed to fit the WGP prediction model provided in Equation [5.1] that includes fixed effects, random genetic effects, and residual effects. Before using `BATools`, data files such as genotype, phenotype and pedigree need to be prepared by the user. For animal and plant breeding, synbreed (Wimmer *et al.* 2012) can be used for recoding genotypes, imputation and etc.; plink (Purcell *et al.* 2007) can be also used for similar tasks; `BATools` leaves choices of the data cleaning and management tools to the user as long as the dataset follows the similar pattern as described in the 'Data' section. Using BATools often consists of three parts: l) loading data and setting up the genotype matrix; 2) setting up initial values for variance components/hyperparameters and options for running the model; 3) model fitting and comparisons. With data loaded as illustrated in Figure 5.2, extra steps for fitting the model are shown in Figure 5.3. To set up the genotype matrix, I can use either centered or standardized genotype matrix for the analysis to help improve algorithmic stability (Stranden and Christensen 2011).

```
#Standardize genotype matrix with method="s"
#for standardization using equation [2]
geno=std_geno(PigM,method="s",freq=PigAlleleFreq)
#Setup initial values for variance component/hyperparameters
#using heritability based rules with h2=0.5
init=set.init(~driploss,data=PigPheno,geno=geno,~id)
#set options
op=set.options(init=init)
#Fitting model
gblup<-baFit(driploss~sex+car_wt,data=PigPheno,geno=geno ,
             genoid = ~id,randomFormula = ~age_slg,options = op)
```

Figure 5.3 Basic model setting and fitting for a GBLUP model

To do that, a built-in function called `std_geno` is provided by BATools and it takes three arguments: `geno` for the original genotype matrix; `method` for standardize as equation [5.2] (`method='s'`) or center (`method='c', i.e.` $\mathbf{T} = \mathbf{M} - \mathbf{1k}'$) the genotype matrix with default to standardize; and the `freq` for the user supplied reference allele frequency, by default, if `freq` is not provided, BATools will compute `freq` based on `geno`. The next step is to set up the initial values for variance components/hyperparameters and options such as the number of total/maximum iterations, burn-in, screen printout messages and whether certain hyperparameters are to be estimated, etc. Figure 5.3 demonstrates how to fit a GBLUP model using the basic default settings, and Figure 5.4 show the full verbose code equivalent to Figure 5.3.

```
#Setup initial values for variance component/hyperparameters
#using heritability based rules for GBLUP using heritability (h2) of 0.5
init=set.init(~driploss,data=PigPheno,geno=geno,~id,h2=0.5,model="GBLUP")
#Default prior for GBLUP is χ-2(-1,0) for residual and marker variance
priors<-list(nu_e=-1,tau2_e=0,nu_s=-1,tau2_s=0)
#Whether to update variance components
update_para=list(vare=TRUE,scale=TRUE)
#Max iteration for running AIREML is set to 50 by default
run_para=list(maxiter=50)
#set options with model (GBLUP), method (REML), Priors, initial values,update scheme,
#maximum number of iterations, file saving location, and convergence criteria
op<-set.options(model="GBLUP",method="REML",priors=priors,init=init,
            update_para=update_para,run_para=run_para,save.at="GBLUP",convcrit=1E-4)
#Fitting model with fixed effect sex, covariates car_wt (carcass weight)
#and non-genetic random effect age_slg (age of slaughter)
gblup<-baFit(driploss~sex+car_wt,data=PigPheno,geno=geno ,genoid = ~id,
             randomFormula = ~age_slg,options = op)
```

Figure 5.4 Full model setting and fitting for GBLUP. Verbose counterpart to Figure 5.3

The `set.init` function to heritability based rules provided in de Los Campos *et al.* (2013) with a default heritability (`h2`) of 0.5 with a full documentation of the rules found in Appendix C. The `set.options` function is used to set up all the options including priors, procedural specifications (number of iterations, burn-in and skip or maximum iteration for REML/MAP), printout options, etc. In the Box 2 and Box 3 example, the `priors` are default $\chi^{-2}(-1,0)$ priors for both residual and marker variance; `update_para` indicates whether the user wants the variance components to be estimated/updated; `run_para` for GBLUP only have `maxiter` to indicate the maximum number of iterations for the AIREML algorithm. Full documentation can be found using `help(set.options)` and default values for each method is documented in the Appendix C.

To fit the model, I can call the `baFit` function with:

- `formula` to specify the response trait to the left of the tilde "~" and corresponding fixed effects. In Box 3 for example, `driploss` is the response and `sex` is the fixed effects as a `factor` and `car_wt` (carcass weight) is a `numeric` as covariates.

- `data` is a `data.frame` containing all the phenotypes, `data` should contain at least two columns: trait to be analyzed and a column that contains individual ID corresponds to the `rownames` of the genotype matrix `geno.`
- `geno` is a genotype matrix with `rownames` of individual ID.
- `genoid` is a `formula` to specify the column of `data` contains individual ID using "~" and corresponding column name. Therefore, the genotype and the data records do not have to be the same order and BATools will match the IDs. In the event that the ID in this column is not available in the genotype matrix, the IDs will be ignored in the analysis. However, if single-step approach is used, the IDs will be matched with the individual ID column in the required pedigree file
- `randomFormula` is a `formula` to specify the column of data to be treated as a random effects factor using "~" and corresponding column name. The random effects factor for this example is `age_slg` (age of slaughter)`.`
- `options` is an object of `options` created by `set.options` function

The return of the `baFit` function is class of object `ba`, which is basically a `list` containing important variables including estimates of fixed effects and covariates (`bhat`) as in Equation [5.1], estimate of random SNP marker effects (`ahat`), estimate of random non-genetic effects (`rhat`) and the predicted value of the phenotypes (`yhat`) as well as other variables such as hyperparameters/variance component estimates. When the GWA option is enabled with `GWA= "SNP" or "Win"`, it also returns the *p-value* or posterior probability for single SNP, or each window if specified using `idw` in the `map`. This window specification can be created by user using BALD or using the `set.win` function that creates fixed size windows based on either number of SNP markers per window or window size in MB (see Box S1). To quickly evaluate the result summary of the estimates, a `S3 print` function is implemented as in Figure 5.5. I

noticed that, the random `age_slg` really had a small variance compared to SNP marker variance, so I did not include it in the extra examples.

```
#Print out basic results
gblup
## Result of BATools:
##
## estimated fixed effects:
##  (Intercept)         sexM       car_wt
##  0.853443770 -0.158191267  0.005642562
##
## SD
## (Intercept)        sexM      car_wt
##   1.0811347   0.1785045   0.0133068
##
## estimated hyperparameters:
##        vare   varMarker var_age_slg
## 0.377903527 0.218162021 0.001028349
```

Figure 5.5 Summary of BATools results

Several examples and use cases also are provided for fitting models for cross-validation (Example 1), GWA using the faster version of EMMAX (Gualdron Duarte *et al.* 2014) and Bayesian variable selection (Example 2), fitting antedependence model for GWA (Example 3), fitting single step model for cross-validation (Example 4). The code for model fitting is provided with the text for GWA and WGP with complete cross-validation analysis. Because I carefully chose the example dataset, each example in the text was executed on a MacBook Pro (Retina, Mid 2012) with 2.3 GHz Intel Core i7 and 8GB of memory within 2-4 minutes depends on models (e.g. antedependence model will take about 4 minutes). Additional examples for each model/method can be also found at the `demo` folder of the `BATools` package or type in `help(baFit)`in R.

5.5.1 Example 1: Cross-validation using BRR, BayesA and SSVS

This example shows fitting a WGP model for cross-validation using three different methods. For demonstration purposes, I only run 5,000 MCMC iterations after 5,000 burn-in samples,

saving every 10th sample to compute the posterior means with niter=10000, burnIn=5000 and skip=10. Figure 5.7, shows the code for each of the three methods. For cross-validation, BATools provides `createCV` to automatically generate random *k*-fold cross-validation. Then I set up the initial values for all the three models using the heritability based rules described in the Appendix C. Running each model is similar to Figure 5.4 except some additional settings for initial values, updating parameter, running parameter and print out options. In `baFit` function, a contrast factor `train` was used to indicate column names for cross-validation. `baplot` can be used to create plot to visualize the results in Figure 5.6.

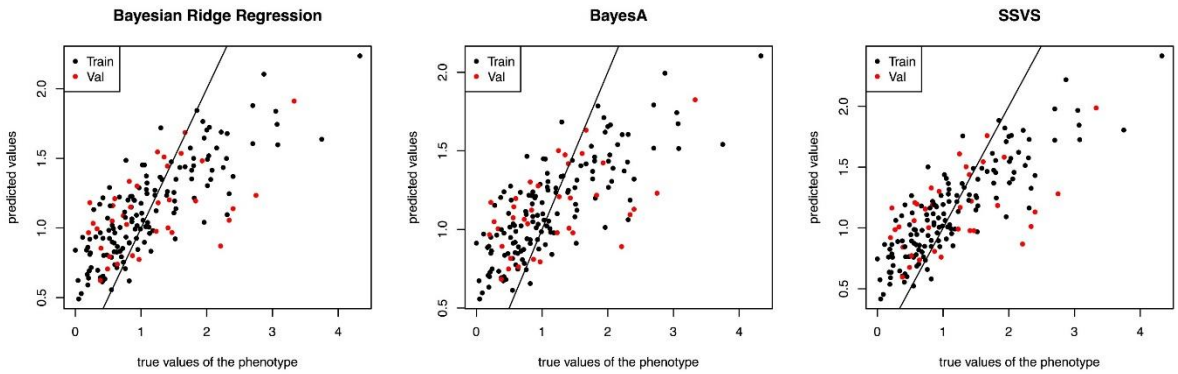

Figure 5.6 Visualization of cross-validation results for BRR, BayesA and SSVS via built-in BATool function `baplot`. Black dots are from training and red dots are from validation set.

I also completed a quick 5-fold cross-validation shown the code in Figure 5.7 with the cross-validation prediction accuracies shown in Table 5.3. In the small example, I didn't find any evidence that the three models differed from each other in cross-validation prediction. In real data applications, it is strongly advised to do more than 5-fold cross-validation (e.g. set *k*=100 in `createCV` function for 100-fold cross-validation) to improve power.

```
rm(list=ls());library(BATools);data("Pig")
geno=std_geno(PigM,method="s",freq=PigAlleleFreq)
#create cv-folds using createCV function
set.seed(1234)
PigPheno=createCV(~driploss,data = PigPheno,k=5)
# Set up parameters and run cv for Bayesian Ridge regression
init=set.init(~driploss,data=PigPheno,geno=geno,~id,h2=0.5,model="rrBLUP")
run_para=list(niter=10000,burnIn=5000,skip=10)
print_mcmc=list(piter=500) # Print status to screen every 500 iteration
update_para=list(scale=TRUE)
ListcvRR<-list()
for(i in 1:5){
  op<-set.options(model="rrBLUP",method="MCMC",init=init,
            update_para=update_para,run_para=run_para,print_mcmc=print_mcmc,seed=i)
  ListcvRR[[i]]<-baFit(driploss~sex,data=PigPheno,geno=geno ,genoid = ~id,options =
op, train=as.formula(paste0("~cv",i)))
}
# Set up parameters and run cv for BayesA
init=set.init(~driploss,data=PigPheno,geno=geno,~id,h2=0.5,model="BayesA")
update_para=list(df=TRUE,scale=TRUE)
ListcvBA<-list()
for(i in 1:5){
  op<-set.options(model="BayesA",method="MCMC",init=init,
              update_para=update_para,run_para=run_para,print_mcmc=print_mcmc,seed=i)
  ListcvBA[[i]]<-baFit(driploss~sex,data=PigPheno,geno=geno ,genoid = ~id,options =
op, train=as.formula(paste0("~cv",i)))
}
# Set up parameters and run cv for SSVS
init=set.init(~driploss,data=PigPheno,geno=geno,~id,pi_snp=0.001,h2=0.5,c=1000,model=
"SSVS")
update_para=list(df=FALSE,scale=TRUE,pi=TRUE)
ListcvSSVS<-list()
for(i in 1:5){
  op<-set.options(model="SSVS",method="MCMC",init=init,
            update_para=update_para,run_para=run_para,print_mcmc=print_mcmc,seed=i)
  ListcvSSVS[[i]]<-baFit(driploss~sex,data=PigPheno,geno=geno ,genoid = ~id,options =
op, train=as.formula(paste0("~cv",i)))
}
save(ListcvRR,ListcvBA,ListcvSSVS,file="ex1_5cv.RData")
# Plot the result for estimating 'driploss'
par(mfrow=c(1,3))
baplot(ListcvRR[[1]],main="Bayesian Ridge Regression")
baplot(ListcvBA[[1]],main="BayesA")
baplot(ListcvSSVS[[1]],main="SSVS")
# Compute cv accuracies
calc.acc<-function(x) cor(x$y[!x$train],x$yhat[!x$train])
acc<-cbind(sapply(ListcvRR,calc.acc),
      sapply(ListcvBA,calc.acc),sapply(ListcvSSVS,calc.acc))
colnames(acc)=c("BRR","BA","SSVS")
apply(acc,2,mean)
```

Figure 5.7 5-fold Cross-validation using BRR, BayesA and SSVS

Table 5.3 Cross-validation prediction accuracy for BRR, BayesA and SSVS

Model	Average cross-validation accuracy
BRR	0.415[a]
BayesA	0.415[a]
SSVS	0.408[a]

Values not sharing the same letter within a row have different (P <0.05) prediction accuracy

5.5.2 Example 2: GWA using EMMAX and SSVS

A computationally efficient algorithm for EMMAX has been recently proposed (Gualdron Duarte *et al.* 2014; Bernal Rubio *et al.* 2016; Chen *et al.* 2017) for GWA. I adapt this strategy as does the software gwaR (https://github.com/steibelj/gwaR) with `map` supplied and the `type` of GWA as either `SNP` (single SNP) or `Win` (window based); Note the window `idw` (a `numeric` indicate the window of each SNP) must be provided for the window based approach (Chen et al., 2017); I use adaptive window for the example dataset as determined by the R package BALD (Dehman and Neuvial 2015) with code in Figure C.1. Figure 5.8 shows the code used for GWA and created Manhattan plot in Figure 5.9. I found none of the SNPs/windows were statistically significant in the example dataset using a P-value threshold of 0.05 divided by the number of SNPs/windows to account for multiple comparison for EMMAX and a PPA of 0.9 for SSVS. This usually indicates that `driploss` is a more polygenic trait and explains the similar performance between BRR, BayesA and SSVS in WGP in Table 5.3.

```
rm(list=ls());library(BATools);data(Pig)
geno=std_geno(PigM,method="s",freq=PigAlleleFreq)
init=set.init(dripLoss~1,data=PigPheno,geno=geno,~id)
op=set.options(init=init)

#Fitting model with GBLUP using default values
#GWA enabled by supplying map and type of GWA
gblup<-baFit(dripLoss~sex,data=PigPheno,geno=geno ,
             genoid = ~id,options = op,map=PigMap,GWA="Win")

#Fitting model with SSVS
init=set.init(~dripLoss,data=PigPheno,geno=geno,~id,pi_snp=0.001,h2=0.5,c=1000,model=
"SSVS")
run_para=list(niter=10000,burnIn=5000,skip=10);print_mcmc=list(piter=500)
update_para=list(df=FALSE,scale=TRUE,pi=F)
op<-set.options(model="SSVS",method="MCMC",init=init,
                update_para=update_para,run_para=run_para,print_mcmc=print_mcmc)
SSVS<-baFit(dripLoss~sex,data=PigPheno,geno=geno ,
            genoid = ~id,options = op, map=PigMap,GWA="Win")

#Create Manhattan plot
par(mfrow=c(2,2))
man_plot_pvalue(gblup,ylim=c(0,6))
man_plot_pvalue(gblup,type="Win",ylim = c(0,6))
man_plot_prob(SSVS,ylim=c(0,1))
man_plot_prob(SSVS,type="Win",ylim=c(0,1))
```

Figure 5.8 GWA using EMMAX and SSVS

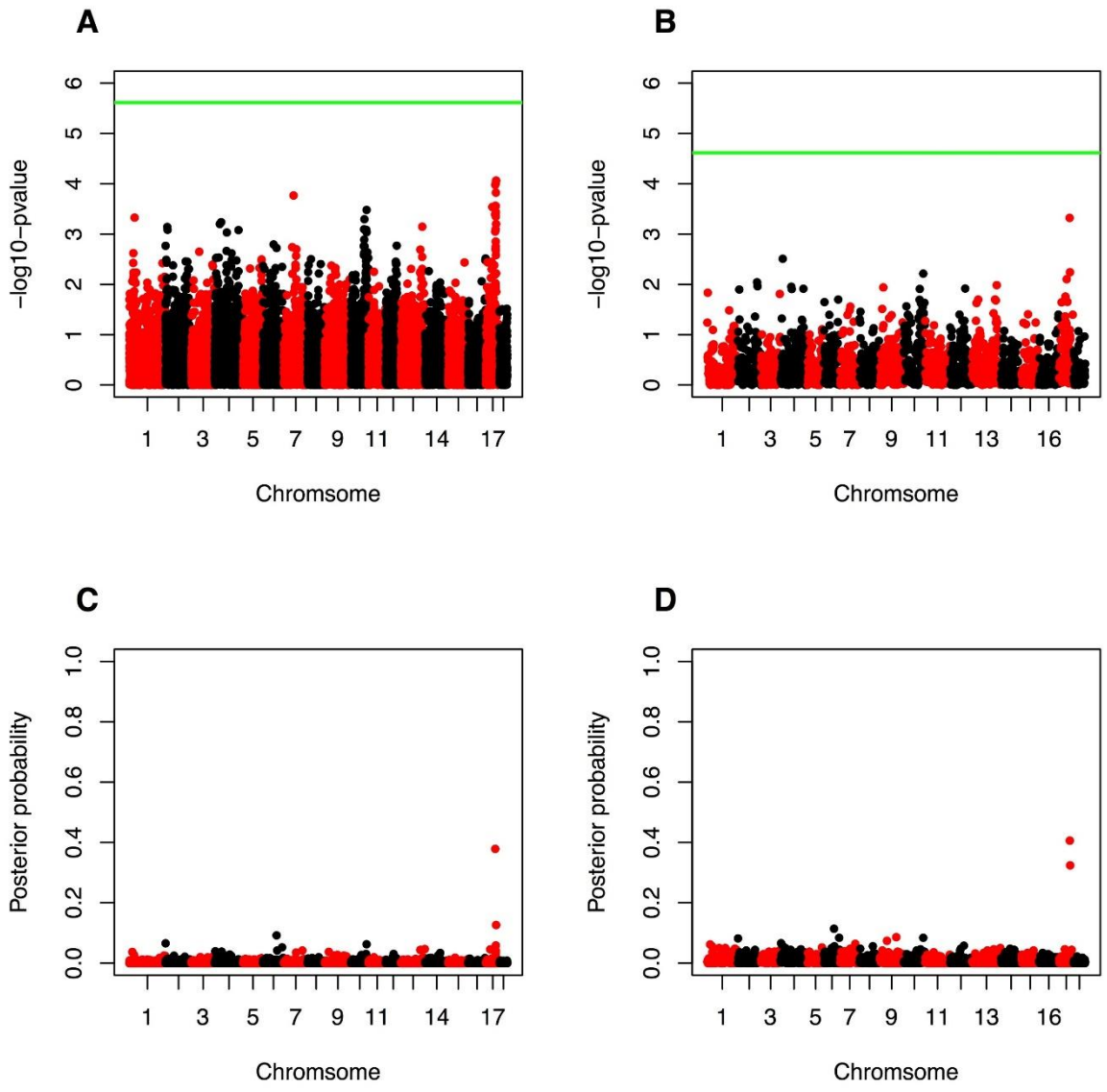

Figure 5.9 Manhattan plot from GWA using the example MSUPRP dataset. Panel A) EMMAX single SNP approach; Panel B) EMMAX adaptive window approach; Panel C) SSVS single SNP approach; Panel D) SSVS adaptive window approach.

5.5.3 Example 3: Fitting antedependence model for GWA

In this example, I illustrate fitting the antedependence models anteBayesA and anteBayesB using our package. As a matter of fact, the procedures are no different from fitting BayesA or BayesB model except that `map` must be provided as in Figure 5.10. Here I demonstrate using

anteBayesA and anteBayesB for GWA. The default initial value for association parameter **t** is **0** with $\mu_t = 0$ and $\sigma_t^2 = 0.5$ can be specified in `set.init` function and will be set to default if these values are `NULL`. The default prior of $\mu_t \sim N(0, 0.01)$ and $\sigma_t^2 \sim \chi^{-2}(-1, 0)$ are suggested by Yang and Tempelman (2012) and can be specified using the `prior` parameter in `set.options` function. For a Bayesian model that does not explicitly involve variable selection, such as BRR, BayesA and anteBayesA, BATools calculates the Bayesian *p*-value for single SNP approach and posterior probability of the window explained more than 1% of the total genetic variance. Figure 5.11 provides Manhattan plots for these two models based on executing the code in Figure 5.10. While I found anteBayesB has the peak in the same location as SSVS in Figure 5.9 for both single SNP and adaptive window based approach, anteBayesA did not have visible peaks. Yang and Tempelman (2012) suggested that the association parameter in the antedependence model might be used for GWA purposes (Panel E and Panel F in Figure 5.11), I found that for anteBayesB, the peak in association parameter corresponded to the peak using posterior probability for single SNP, while for anteBayesA, the *t*-distributed prior still provided too much shrinkage for relatively large marker effect compared to anteBayesB, such that the association parameter did provide any signal.

```
rm(list=ls());library(BATools);data(Pig)
geno=std_geno(PigM,method="s",freq=PigAlleleFreq)

#Set parameters for anteBayesA and fit the model
init=set.init(~driploss,data=PigPheno,geno=geno,~id,df=2.5,
               h2=0.5,mut=0,vart=0.5,model="anteBayesA")
run_para=list(niter=10000,burnIn=5000,skip=10);print_mcmc=list(piter=500)
update_para=list(df=F,scale=TRUE,mut=F)
priors=list(mu_m_t=0,sigma2_m_t=0.01,df_var_t=-1,scale_var_t=0)
op<-set.options(model="anteBayesA",method="MCMC",init=init,update_para=update_para,
          priors=priors,run_para=run_para,save.at="anteBayesA",print_mcmc=print_mcmc)

anteBA<-baFit(driploss~sex,data=PigPheno,geno=geno,
              genoid = ~id,options = op,map=PigMap,GWA="Win")

#Set parameters for anteBayesB and fit the model
init=set.init(~driploss,data=PigPheno,geno=geno,~id,
              df=2.5,pi_snp=0.001,h2=0.5,model="anteBayesB")
update_para=list(df=F,scale=TRUE,pi=F,mut=F)
op<-set.options(model="anteBayesB",method="MCMC",init=init,update_para=update_para,
                run_para=run_para,save.at="anteBayesB",print_mcmc=print_mcmc)
anteBB<-baFit(driploss~sex,data=PigPheno,geno=geno ,
              genoid = ~id,options = op,map=PigMap,GWA="Win")

#Create Manhattan plot
par(mfrow=c(3,2))
man_plot_prob(anteBA,ylim=c(0,6))
man_plot_prob(anteBA,type="Win",ylim = c(0,1))
man_plot_prob(anteBB,ylim=c(0,1))
man_plot_prob(anteBB,type="Win",ylim=c(0,1))
man_plot_assoc(anteBA,ylim=c(0,1))
man_plot_assoc(anteBB,ylim=c(0,1))
```

Figure 5.10 GWA using anteBayesA and anteBayesB

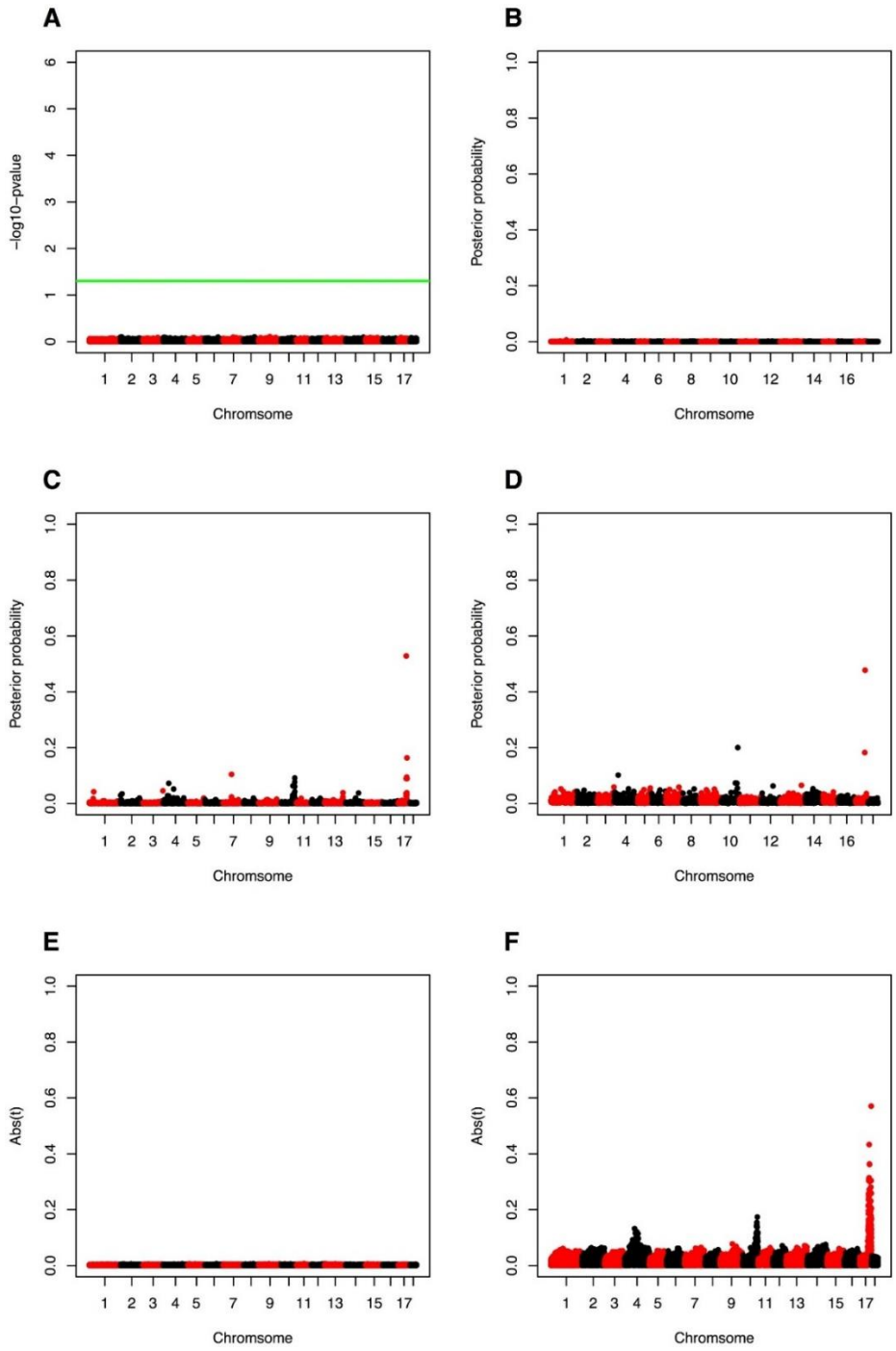

Figure 5.11 Manhattan plot from GWA using the example MSUPRP dataset. Panel A) anteBayesA single SNP approach; Panel B) anteBayesA adaptive window approach; Panel C) anteBayesB single SNP approach; Panel D) anteBayesB adaptive window approach; Panel E) absolute value of association parameter for anteBayesA; Panel F) absolute value of association parameter for anteBayesA.

5.5.4 Example 4: Fitting single-step model using ssGBLUP, ssBayesA, ssBayesB and ssSSVS

In single-step approach, only additional pedigree information (in a form of `data.frame` with three columns including individual ID, sire ID and dam ID) is required and the example dataset provided is `PigPed`. To demonstrate an example single-step extension, I artificially mask genotypes of some individuals as missing as `genoNew` (Figure 5.12). Then I fit the model with ssGBLUP, ssBayesA, ssBayesB and ssSSVS. The code for Table 5.4 after running 5-fold cross-validation is provided in Figure 5.12. Although ssBayesA, ssBayesB and ssSSVS has slightly higher cross-validation prediction accuracies than ssGBLUP, the difference was not significant. Since the examples are just an illustration on how to use the code on a relatively small sample size, Chapter 4 should be referred for the complete simulation study and real data analysis for different type of traits. For ssGBLUP, default will use homogenous genetic variance (`ssGBLUPvar = "homVAR"`) and to set it to heterogeneous genetic variance in Chapter 4, use the `set.options` function with `ssGBLUPvar= "hetVAR"` while be aware that `hetVAR` may not converge.

Table 5.4 Cross-validation prediction accuracy for ssGBLUP, ssBayesA, ssBayesB and ssSSVS

Model	Average cross-validation accuracy
ssGBLUP	0.406[a]
ssBayesA	0.420[a]
ssBayesB	0.421[a]
ssSSVS	0.430[a]

Values not sharing the same letter within a row have different ($P < 0.05$) prediction accuracy

```
rm(list=ls());library(BATools);data("Pig")
geno=std_geno(PigM,method="s",freq=PigAlleleFreq)
#Mask some genotype as missing to test single-step approach
set.seed(1001);n=dim(geno)[1];indexng<-sort(sample(1:n,n%/%5))
genoNew=geno[-indexng,]
#create cv-folds using createCV function
set.seed(1234)
PigPheno=createCV(~driploss,data = PigPheno,k=5)
# Set up parameters and run cv ssGBLUP
init=set.init(~driploss,data=PigPheno,geno=genoNew,~id,h2=0.5,model="ssGBLUP")
ListcvGBLUP<-list()
for(i in 1:5){
  op<-set.options(model="ssGBLUP",method="REML",init=init,seed=i)
  ListcvGBLUP[[i]]<-baFit(driploss~sex,data=PigPheno,geno=genoNew ,genoid = ~id,
                  ped= PigPed,options = op, train=as.formula(paste0("~cv",i)))
}
# Set up parameters and run cv for ssBayesA
init=set.init(~driploss,data=PigPheno,geno=genoNew,~id,h2=0.5,model="ssBayesA")
run_para=list(niter=10000,burnIn=5000,skip=10);print_mcmc=list(piter=500)
update_para=list(df=TRUE,scale=TRUE);ListcvBA<-list()
for(i in 1:5){
  op<-set.options(model="ssBayesA",method="MCMC",init=init,
              update_para=update_para,run_para=run_para,print_mcmc=print_mcmc,seed=i)
  ListcvBA[[i]]<-baFit(driploss~sex,data=PigPheno,geno=genoNew ,genoid = ~id,
                ped= PigPed,options = op, train=as.formula(paste0("~cv",i)))
}
# Set up parameters and run cv for ssBayesB
init=set.init(~driploss,data=PigPheno,geno=genoNew,~id,pi_snp=0.001,model="ssBayesB")
update_para=list(df=TRUE,scale=TRUE,pi=TRUE);ListcvBB<-list()
for(i in 1:5){
  op<-set.options(model="ssBayesB",method="MCMC",init=init,
               update_para=update_para,run_para=run_para,print_mcmc=print_mcmc,seed=
i)
  ListcvBB[[i]]<-baFit(driploss~sex,data=PigPheno,geno=genoNew ,genoid = ~id,
                ped= PigPed,options = op, train=as.formula(paste0("~cv",i)))
}
# Set up parameters and run cv for ssSSVS
init=set.init(~driploss,data=PigPheno,geno=genoNew,~id,pi_snp=0.001,
              h2=0.5,c=1000,model="ssSSVS")
update_para=list(df=FALSE,scale=TRUE,pi=T); ListcvSSVS<-list()
for(i in 1:5){
  op<-set.options(model="ssSSVS",method="MCMC",init=init,
              update_para=update_para,run_para=run_para,print_mcmc=print_mcmc,seed=i)
  ListcvSSVS[[i]]<-baFit(driploss~sex,data=PigPheno,geno=genoNew ,genoid = ~id,
                 ped= PigPed,options = op, train=as.formula(paste0("~cv",i)))
}
# Compute cv accuracies
calc.acc<-function(x){
  cor(x$y[!x$train],x$yhat[!x$train])
}
acc<-cbind(sapply(ListcvGBLUP,calc.acc),sapply(ListcvBA,calc.acc),sapply(ListcvBB,cal
c.acc),sapply(ListcvSSVS,calc.acc))
colnames(acc)=c("ssGBLUP","ssBA","ssBB","ssSSVS")
apply(acc,2,mean)
```

Figure 5.12 5-fold Cross-validation using ssGBLUP, ssBayesA, ssBayesB and ssSSVS

5.6 Performance and computing time

`BATools` uses C/C++ and FORTRAN subroutines to make sure it has the optimal performance on a single core. The most computing demanding portion of the code is sampling SNP marker effects using Gibbs sampler. I carried out the benchmark with five different sample sizes (*n= 1k, 2k, 3k, 4k* and *5k*) and five different marker densities (*m=1k, 5k, 20k, 60k* and *100k*) by fitting BayesB (Figure 5.13).

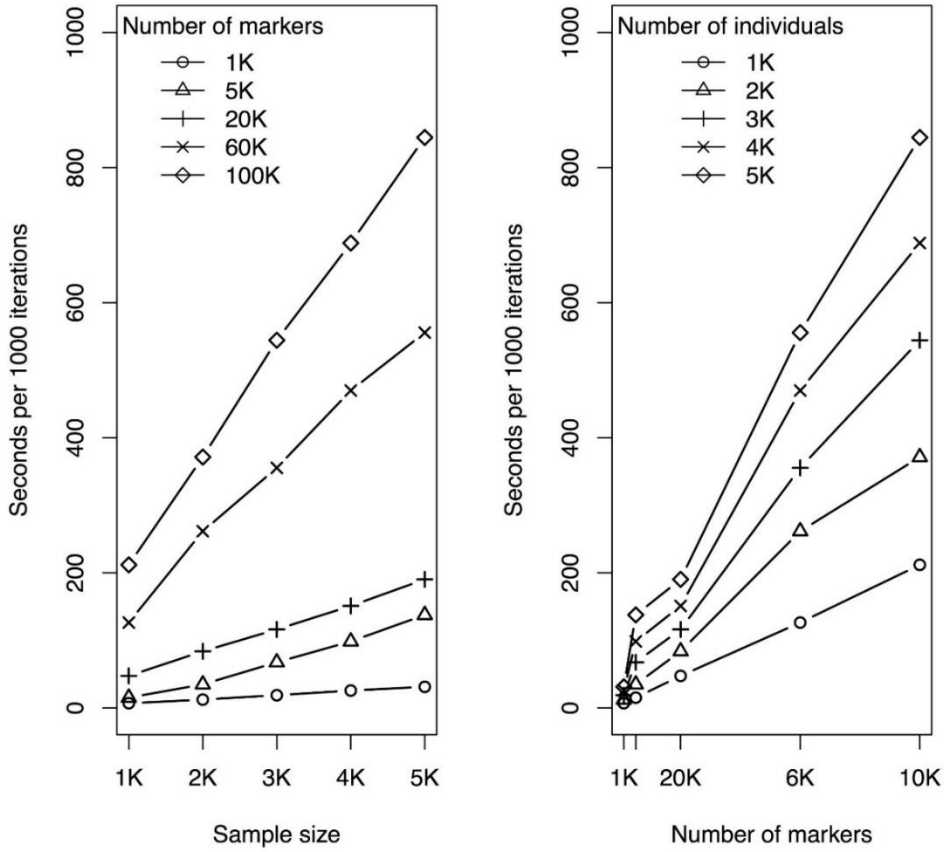

Figure 5.13 Computing time in seconds per 1000 iterations for BayesB for sampling all the marker effects by sample size and the number of marker. The benchmark was performed on a 2.4Ghz Intel Xeon E5-2680v4 CPU using a single core

The benchmark was computed on Michigan State University High Performance Computing Center (HPCC) on a single core of 2.4Ghz Intel Xeon E5-2680v4 CPU. The computing time was

both affected by the sample size and marker density and had almost linear relationship with both when the other variable was fixed. The most demanding scenario (*n=5k* and *m=100k*) took about 14 minutes to run 1000 iterations. I also found that either antedependence specification roughly doubled the computing time because of extra step to sample the association parameters with roughly similar length of SNP marker effects. In a typical analysis running *200k* iterations at *n=5k* and *m=60k*, analysis can be completed within a day on a single core.

5.7 Concluding remarks and future developments

BATools provides a common user interface for a suite of popular Bayesian models for WGP and GWA that allow for differences in shrinkage or variable selection options, models the association between adjacent SNP markers and combines phenotype of non-genotyped individual via pedigree information. BATools also provide easy tools for cross-validation and to visualize the results for WGP and GWA. Further extensions such as extending the antedependence models for the single-step approach, GxE using Bayesian models (Yang *et al.* 2015a) and modeling repeated records will be available through updates.

The most computationally intensive part of BATools consists of using the Gibbs sampler for SNP marker effects. Our approach is to use C/C++ and FORTRAN subroutines to reduce the computing time. Still, the computing time per thousand iterations will linearly increase with increasing marker density or sample size. The user should be aware of the fact the increasing marker density might lead to poor mixing in MCMC, therefore, extra iterations might be required to reach convergence. The Gibbs sampler particular for WGP does not appear to be parallelizable because sampling each marker effect depends on the current value of all other marker effects, whereas calculating the right-hand-side (*rhs*) of mixed model equation is parallelizable (Fernando *et al.* 2016). This can be achieved using shared memory multiprocessing via OpenMP

or GPU computing to accommodate increasing number of MCMC samples. Since R does not have good support of GPU computing and excessive environment setups are required for compiling the code, I decided not to have this feature in BATools R package. As for OpenMP, proper setups are required for executing the code in true parallel, therefore, a OpenMP version of BATools will only be available on Github (https://github.com/chenchunyu88/batools) after it is fully tested to allow experienced users to take advantage of parallelization.

With the increasing availability of the sequencing data, computing efficiency for WGP and GWA using Bayesian model will be a major challenge. In the 1000 bull genomes project, 28.3 million variants of 238 cattle were identified (Daetwyler *et al.* 2014). With this dataset, BATools will take ~4 hours per 1000 iterations, which is not very efficient. The much bigger problem is that it is not efficient to load the data into R because it will take ~160 GB of memory. To handle this type of dataset, a modification of BATools needs to be implemented: 1) use `bigmemory` (Kane *et al.* 2013) R package to load the data into R; 2) use `RcppEigen` (Bates and Eddelbuettel 2013) R package to pre-construct the additive genomic relationship matrix $\mathbf{G}$; 3) modify `BATools` to handle kinship matrix $\mathbf{G}$ instead of taking genotype matrix directly and output the $\mathbf{u}$ in equation [5.3**]**; 4) write another function to obtain *P*-value for single SNP using `RcppEigen` based on (Chen *et al.* 2017). These modifications are equivalent to use BATools for GWA using fast EMMAX. Then one might use only say 5% variants with smallest *P*-value for a WGP and the computing time per 1000 iteration will be just under ~20 minutes with 1.4 million variants. Even with this approach, Bayesian models might still need large number of iterations to converge for 1.4 million variants. Hybrid approaches using EM algorithm to set up starting values for MCMC could effectively skip burn-in and reduce the total computing time than the original MCMC approach (Wang *et al.* 2016). Overall, WGP and GWA with sequencing data is

challenging, but with further improvement, `BATools` can efficiently handle high density sequencing data.

Chapter6 Conclusions, Discussions and Future Work

This dissertation focused on extending existing statistical models and developing software tools for whole genome prediction (WGP) and genome wide association (GWA) analysis in animal and plant breeding. These tools have been used, respectively, to accelerate selection for economically important traits and identify important genomic regions based on high density SNP marker genotypes. The primary goal of this work was to make hierarchical modeling and software tools for WGP and GWA more accessible for academic research and industry applications. This included exploring computationally feasible, albeit approximate, alternative algorithms (Chapter 2), developing more powerful and more formal GWA strategies for hierarchical linear models (Chapter 3), extending flexible Bayesian models to allow for phenotypes on non-genotyped animals in GWA analyses (Chapter 4) and providing the associated software tools for these and other recent hierarchical linear model developments for GWA and WGP (Chapter 5).

During the time of preparing this dissertation, I designed the algorithms and methodologies based on the assumption that the WGP or GWA inferences were usually $m \gg n$ problems (m being the number of markers or covariates and n being the number of observations or animals). With the use of the EM algorithm in Chapters 2 and 3, I believed that I provided a computationally tractable alternative to MCMC for more flexible priors (n<=5,000, m>50,000); however, that assumption may not be true for some current or future applications. As a matter of fact, some genomic evaluation programs now have more individuals (> 1 million) than the number of SNP markers (~50,000) (Fernando *et al.* 2014; Masuda *et al.* 2016). In such cases, the EM based approaches in Chapter 2 can still work efficiently with SNP marker effect models. However, in many other applications, especially in plant breeding programs, $m \gg n$ might still

be true for some time yet since, based on my personal experience, some companies may have only a few thousand inbred lines (genetically similar individuals that are bred with each other for uniformity) in their breeding program and selection is more based on these lines rather than the hybrids (progeny of two inbred lines). In the applications for $m \ll n$, two major approaches can be considered. Firstly, Bayesian models deserve greater consideration as they do not require the large matrix inversion like REML but rather uses accelerated MCMC sampling via high performance parallel or GPU computing (Fernando *et al.* 2016). Secondly a modified GBLUP based algorithm for proven and young (APY) has been developed (Misztal 2016a), based on specifying the breeding values (BVs) of noncore animals to be an approximate function of only the BVs of core animals. This results in a computing cost for the inverse of genomic relationship matrix to be only cubic to the number of core animals which is significant since the computing time is only relevant to the number of these selected animals. One interesting development in the ssGBLUP approaches that incorporate information on non-genotyped animals is that researchers have attempted to differentially weight marker effects when building the genomic relationship matrix to improve prediction accuracies analogous to our EM based approaches (Zhang *et al.* 2016).Currently, such models also suffer from some of the same convergence issues as with EM, i.e., accuracies are higher in first few iterations than in later iterations when the algorithm is close to convergence (Zhang *et al.* 2016). Comparison in WGP accuracies between EM-based approaches and weighted ssGBLUP approaches deserve further investigation. To deal with the $m \ll n$ problem, I can use the SNP marker effects model directly. Although deterministic annealing for MAP-SSVS is a useful tool to help with convergence and avoid local maxima, it is computationally expensive, therefore, not suitable for large data applications.

In our GWA research in both Chapter 3 and Chapter 4, I found that Bayesian variable selection model is particularly effective when associations are based on genomic windows adaptively based on LD structure. I do realize a universally good performance may not be guaranteed in populations given that the LD structure can be different across different studies. Therefore, further studies are necessary. I also noticed WGP and GWA involve different goals even though they are increasingly based on the same/similar models. EM based approaches should be discouraged for GWA unless that the sensitivity to starting values and various convergence issues can be fully resolved. I demonstrated that the use of the Expectation Maximization variable selection (EMVS) strategy of Rockova and George (2014) can alleviate the starting value issue in Chapter 2, but it may be computationally intractable for large scale applications.

In Chapter 4, I found that ssSSVS which incorporates information on non-genotyped animals can lead to higher estimated posterior probability on peak associations compared to SSVS, particularly for a trait, milk fat, which is known to be heavily controlled by a major gene, DGAT1. In Chapter 3 I determined the highest single SNP and window posterior probably of association (PPA) of 0.478 and 0.772 correspondingly with 922 samples for backfat in swine whereas in Chapter 4, the highest single SNP and window PPA, they are both 100% with milkfat samples on 3186 dairy cows. I know that different species and different traits are not remotely comparable, but I think it's important to look into the sample size requirements and guidance for hierarchical model GWA.

Another important topic that I cannot avoid discussing is hyperparameter specification. Even though estimating hyperparameters via MCMC sampling have been highly recommended in previous WGP research (Yang *et al.* 2015b) and I did find hyperparameter specification is

important for GWA inferences in Chapter 3 and Chapter 4, estimating them with MCMC may not be always viable or take far too long to generate reliable inferences considering the poor mixing of some hyperparameters, especially for more polygenic traits. Conceivably, some hyperparameter specifications such as the degree of freedom v for *t*-distribution or the probability of association, π, in variable selection methods could be determined by cross-validation as long as other hyperparameters can be well estimated using MCMC when v or π is fixed. Lee *et al.* (2017) recently determined hyperparameter values based on such specifications that lead to the highest cross-validation WGP accuracies and hence might be a better solution when hyperparameters cannot be well estimated through MCMC. Knürr *et al.* (2013) also proposed to use many different values of hyperparameters and finally averaged the prediction results to obtain more robust inference. Hyperparameter specification is admittedly complicated, but WGP accuracies are undoubtedly dependent upon their proper specification (Wimmer *et al.* 2013). Therefore, comprehensive guidelines for hyperparameter tuning is worth further study for GWA and WGP.

The ssGBLUP approaches that incorporate information on non-genotyped animals have become mainstream for genomic prediction problems (Misztal 2016b). Recent work by Lee *et al.* (2017) and my Chapter 4 suggested Bayesian approaches that also incorporate such information, i.e. ssSSVS, led to higher accuracies than ssGBLUP where the traits are controlled by major genes; even for polygenic traits, ssSSVS had equivalent prediction accuracies with ssGBLUP because in extreme polygenic cases, ssSSVS with $\pi_\phi = 1$ is equivalent to ssGBLUP. However, these examples were the only two real data applications using Bayesian sparse priors and focused on a relatively small dataset (n<4000). Further research in large populations or national genomic evaluations seems necessary. In ssSSVS, it's natural to sample the variance component through

heterogeneous variance specification (i.e. estimate genetic variation due to marker effects and not accounted for by markers separately) without extra computational cost if it is clear that the genetic variability could be conceivably different between genotyped and non-genotyped animals. For ssGBLUP, however, most applications are based on a homogeneous genetic variance specification (Legarra *et al.* 2014). In Chapter 4, I found that heterogeneous genetic variance specifications could be particularly important. However, since a heterogeneous genetic variance specification seems to periodically suffer from convergence issues using AIREML, a single-step Bayesian approach with a Gaussian prior using MCMC might be a solution. Again, further research on this topic also needed. It is also conceivable that different herds have not only heterogeneous genetic variation but also heterogeneous residual variation so that WGP and GWA extensions that have already considered heterogeneous residual variance modeling (Ou *et al.* 2016) for different herds should be combined with the developments that I have provided in this thesis.

In Chapter 5, I demonstrated an R package `BATools` for implementing the models discussed in this dissertation. I extensively tested the package, including using it for Chapter 2-4 and cross-referenced with other software packages, in the meantime, I will continue to test it in more data sets through our research and industry applications. I also realize there is always room for improving the computational efficiency. Finally, I designed `BATools` package to be extendable, and many other models on our list, such as ss-anteBayesA/B, GxE extension on the Bayesian model and handling multiple record data, will be added to the package.

At the time this research was proposed, whole-genome sequence (WGS) data for livestock was not widely available, even today, it is still only available for few researchers. Brøndum *et al.* (2015) reported small (2-5%) increase in GEBV prediction reliability. Bayesian models are

expected to improve the WGP accuracies using WGS data compared to GBLUP because the genomic relationship in GBLUP can be well estimated using high density SNP data (777k) while Bayesian models model the marker effects directly may get additional benefit from higher density in WGS (Meuwissen *et al.* 2016). Since long-range LD may be extensive in some populations for WGS data, a window based inferences might be still more appropriate than single SNP inferences in GWA. However, the adaptive window approach used in Chapter 3 and 4 becomes increasingly inefficient since it requires computation of the LD matrix for the entire chromosome, hence, more efficient method for clustering variants into windows may need to be developed. I believe that Bayesian WGP is still valid for WGS data. Since BATools includes both GWA and WGP, I can slightly modify BATools to handle large WGS data set without the excessive usage of memory through dimension reduction: select top variants from WGS using LMM based GWA tools; then use top variants for our WGP models. Or I can reduce the number of random draws from posterior distribution by developing efficient variable selection methods that stop sampling some zero effects in the MCMC chain (Moser *et al.* 2015). Furthermore, parallelizable versions of Bayesian WGP and GWA based on orthogonal data augmentation should also be explored to deal with WGS data (Cheng *et al.* 2017).

APPENDICES

Appendix A: Chapter 3

Implementation Details on Maximum A Posteriori and Monte Carlo Markov Chain Inferences in BayesA and SSVS models

Expectation (E-) steps and Maximization (M-) steps

Following Chen and Tempelman (2015), the E-step or the expectation of the portion of the log joint posterior density that is a function of τ_j is given for MAP-BayesA in Equation [A1]:

$$\begin{aligned}&\underset{\tau_j|.}{\mathrm{E}}\left(\sum_{j=1}^{m}\left(\log\left(p\left(g_j|\sigma_g^2,\tau_j\right)+\log p\left(\tau_j|\nu_\tau\right)\right)\right)\right)\\&=\sum_{j=1}^{m}\left(-\frac{1}{2}\frac{g_j^2}{\sigma_g^2}\underset{\tau_j|.}{\mathrm{E}}\left(\frac{1}{\tau_j}\right)-\left(\frac{\nu_\tau}{2}+1\right)\underset{\tau_j|.}{\mathrm{E}}\left(\log\left(\tau_j\right)\right)-\frac{\nu_\tau}{2}\underset{\tau_j|.}{\mathrm{E}}\left(\frac{1}{\tau_j}\right)\right)+\text{constant}.\end{aligned} \quad [A1]$$

and for MAP-SSVS in Equation [A2]

$$\begin{aligned}&\underset{\tau_j|.}{\mathrm{E}}\left(\sum_{j=1}^{m}\log\left(p\left(g_j|\sigma_g^2,\tau_j\right)+\log p\left(\tau_j|\pi_\tau\right)\right)\right)\\&=\sum_{j=1}^{m}-\frac{1}{2}\sigma_g^{-2}{g_j}^2\left(\underset{\tau_j|.}{\mathrm{E}}\left(\tau_j\right)+\left(1-\underset{\tau_j|.}{\mathrm{E}}\left(\tau_j\right)\right)c\right)\\&+\underset{\tau_j|.}{\mathrm{E}}\left(\tau_j\right)\log\left(\pi_\tau\right)+\left(1-\underset{\tau_j|.}{\mathrm{E}}\left(\tau_j\right)\right)\log\left(1-\pi_\tau\right)+\text{constant}.\end{aligned} \quad [A2]$$

with $\underset{\tau_j|.}{\mathrm{E}}\left(\frac{1}{\tau_j}\right)=\hat{\tau}_j^{-1}=\frac{\nu_\tau+1}{\frac{\hat{g}_j^2}{\sigma_g^2}+\nu_\tau}$ for MAP-BayesA

and $\underset{\tau_j|.}{\mathrm{E}}\left(\hat{\tau}_j\right)=\frac{\phi_{\hat{g}_j}\left(0,\sigma_g^2\right)\pi_\tau}{\phi_{\hat{g}_j}\left(0,\sigma_g^2\right)\pi_\tau+\phi_{\hat{g}_j}\left(0,\frac{\sigma_g^2}{c}\right)\left(1-\pi_\tau\right)}$ for MAP-SSVS where $\phi_x\left(\mu,\sigma^2\right)$ denotes the ordinate of a Gaussian probability density function with mean μ and variance σ^2 evaluated at *x*.

A conditional maximization or M-step for $\boldsymbol{\beta}$ and $\mathbf{g}$ can be determined by solving the following MME using a SNP-centric model in Equation [A3]

$$\begin{bmatrix}\mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z}\\ \mathbf{Z}'\mathbf{X} & \mathbf{Z}'\mathbf{Z}+\mathbf{D}^{-1}\sigma_e^2\sigma_g^{-2}\end{bmatrix}\begin{bmatrix}\hat{\boldsymbol{\beta}}\\ \hat{\mathbf{g}}\end{bmatrix}=\begin{bmatrix}\mathbf{X}'\mathbf{y}\\ \mathbf{Z}'\mathbf{y}\end{bmatrix} \quad [A3]$$

or an animal-centric model in Equation [A4]

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}' \\ \mathbf{X} & \mathbf{I}+\mathbf{G}^{-1}\sigma_e^2\sigma_g^{-2} \end{bmatrix}\begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\mathbf{a}} \end{bmatrix}=\begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{Z}'\mathbf{y} \end{bmatrix} \quad [A4]$$

with $\mathbf{G}=\mathbf{ZDZ}'$ (Sun *et al.* 2012) with $\mathbf{D}^{-1}=diag\left\{\left(\hat{\tau}_j\right)^{-1}\right\}$ in BayesA or $\mathbf{D}^{-1}=diag\left(\hat{\tau}_j+c\left(1-\hat{\tau}_j\right)\right)$ in SSVS.

Hyperparameter estimation under MAP

Solutions based on the mixed model equations [A3] and [A4] are conditioned on the variance components and/or hyperparameters being known. In the classical mixed model literature, those variance components can be estimated using REML. The vector of hyperparameters are $\boldsymbol{\theta}=\left(\sigma_e^2,\sigma_g^2,\nu_\tau\right)$ for BayesA, and $\boldsymbol{\theta}=\left(\sigma_e^2,\sigma_g^2,\pi_\tau\right)$ for SSVS. I partition $\boldsymbol{\theta}$ into the variance components $\boldsymbol{\sigma}=\left(\sigma_e^2,\sigma_g^2\right)$ and remaining hyperparameters as $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}$ such that, for example, $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}=\nu_\tau$ in BayesA whereas $\boldsymbol{\theta}_{-\boldsymbol{\sigma}}=\pi_\tau$ in SSVS.

The classical log REML function (Searle *et al.* 1992) can be written as follows:

$$l\left(\boldsymbol{\sigma}\,|\,\mathbf{y}\right)=-0.5\log\left|\mathbf{V}\right|-0.5\log\left|\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right|-0.5\mathbf{y}'\mathbf{P}\mathbf{y} \quad [A5]$$

with $\mathbf{V}=\mathbf{ZDZ}'\sigma_g^2+\mathbf{I}\sigma_e^2$ and $\mathbf{P}=\mathbf{V}^{-1}-\mathbf{V}^{-1}\mathbf{X}\left(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right)^{-}\mathbf{X}'\mathbf{V}^{-1}$. In typical classical REML specifications involving uncorrelated random effects, **D** = **I**. I modify this expression for our BayesA and SSVS adaptations accordingly as:

$$\begin{aligned} &l\left(\boldsymbol{\sigma},\boldsymbol{\tau}\,|\,\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y}\right)=\log\left(p\left(\boldsymbol{\sigma},\boldsymbol{\tau}\,|\,\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y}\right)\right) \\ &=\text{constant}-0.5\log\left|\mathbf{V}\right|-0.5\log\left|\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right|-0.5\mathbf{y}'\mathbf{P}\mathbf{y}+\log p\left(\boldsymbol{\tau}\,|\,\boldsymbol{\theta}_{-\boldsymbol{\sigma}}\right)+\log p\left(\boldsymbol{\sigma}\right). \end{aligned} \quad [A6]$$

Recall for either hierarchical model, **D** is a function of $\boldsymbol{\tau}$ for which conditional expectations are used to derive $\mathbf{D}^{-1}=diag\left\{\left(\hat{\tau}_j\right)^{-1}\right\}$ in BayesA or $\mathbf{D}^{-1}=diag\left(\hat{\tau}_j+c\left(1-\hat{\tau}_j\right)\right)$ in SSVS as noted earlier. Upon evaluating Equation [A6] at $\mathbf{D}^{-1}$, this expression is maximized with respect to $\boldsymbol{\sigma}$. I again denote the corresponding estimates as marginal maximum likelihood (MML)

estimates in order to distinguish them from classical REML estimates (Chen and Tempelman 2015).

Average Information REML (AIREML) is a particularly attractive hybrid Fisher's scoring/Newton Raphson algorithm used to obtain REML estimates under classical Gaussian specifications for **g** based on the log likelihood of Equation [A5] (Gilmour *et al.* 1995; Johnson and Thompson 1995). I adapt this algorithm for our proposed MML approach in Equation [A6] by simply replacing $\boldsymbol{\tau}$ by $\hat{\boldsymbol{\tau}}$ from a previous E-step followed by maximizing Equation [A6] with respect to $\boldsymbol{\sigma}$ in a M-step evaluated at $\hat{\boldsymbol{\tau}}$ .To account for prior information in $\log p(\boldsymbol{\sigma})$, I augment the AIREML first and second derivatives as provided by Johnson and Thompson (1995) with $\frac{\partial}{\partial \boldsymbol{\sigma}} \log p(\boldsymbol{\sigma})$ and $\frac{\partial^2}{\partial \boldsymbol{\sigma} \partial \boldsymbol{\sigma}'} \log p(\boldsymbol{\sigma})$, respectively.

MML algorithm for variance component estimation

Recall that the classical log REML function (Searle *et al.* 1992) can be written as follows:

$$l(\boldsymbol{\sigma} \mid \mathbf{y}) = -0.5 \log |\mathbf{V}| - 0.5 \log \left| \mathbf{X}' \mathbf{V}^{-1} \mathbf{X} \right| - 0.5 \mathbf{y}' \mathbf{P} \mathbf{y} \qquad [A7]$$

with $\mathbf{V} = \mathbf{Z}\mathbf{D}\mathbf{Z}' \sigma_g^2 + \mathbf{I} \sigma_e^2$ and $\mathbf{P} = \mathbf{V}^{-1} - \mathbf{V}^{-1}\mathbf{X}\left(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{V}^{-1}$.

The Fisher scoring algorithm for iterate [*k*] in AIREML for MAP-BayesA and MAP-SSVS can be specified as follows:

$$\left(\boldsymbol{\sigma}^{[k]} - \boldsymbol{\sigma}^{[k-1]}\right) = \left(\mathrm{E}_y \left(- \frac{\partial^2 \log p(\boldsymbol{\sigma}, \boldsymbol{\tau} \mid \boldsymbol{\theta}_{-\boldsymbol{\sigma}}, \mathbf{y})}{\partial \boldsymbol{\sigma} \partial \boldsymbol{\sigma}'} \right) \Bigg|_{\boldsymbol{\sigma} = \boldsymbol{\sigma}^{[k-1]}} \right)^{-1} \frac{\partial \log p(\boldsymbol{\sigma}, \boldsymbol{\tau} \mid \boldsymbol{\theta}_{-\boldsymbol{\sigma}}, \mathbf{y})}{\partial \boldsymbol{\sigma}} \Bigg|_{\boldsymbol{\sigma} = \boldsymbol{\sigma}^{[k-1]}} \qquad [A8]$$

where the vector of first derivatives could be determined using Johnson and Thompson (1995) as:

$$\frac{\partial \log p(\boldsymbol{\sigma},\boldsymbol{\tau}\,|\,\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y})}{\partial \sigma_e^2} = -\frac{1}{2}tr(\mathbf{P}) + \frac{1}{2}\mathbf{y}'\mathbf{P}\mathbf{y} + \frac{\partial \log p(\boldsymbol{\sigma})}{\partial \sigma_e^2}$$
$$= -\frac{1}{2}\left(\frac{n - rank(\mathbf{X})}{\sigma_e^2} - \left(\frac{1}{\sigma_e^2}\right)\left(m - \frac{trace\left(\mathbf{D}^{-1}\mathbf{C}^{gg|\mathbf{D}}\right)}{\sigma_g^2}\right) - \frac{\hat{\mathbf{e}}'\hat{\mathbf{e}}}{\left(\sigma_e^2\right)^2}\right) + \frac{\nu_e s_e^2}{2\left(\sigma_e^2\right)^2} - \frac{\left(\nu_e + 2\right)}{2\sigma_e^2} \quad [A9]$$

and

$$\frac{\partial \log p(\boldsymbol{\sigma},\boldsymbol{\tau}\,|\,\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y})}{\partial \sigma_g^2} = -\frac{1}{2}tr\left(\mathbf{P}\mathbf{Z}\mathbf{D}\mathbf{Z}'\right) + \frac{1}{2}\mathbf{y}'\mathbf{P}\mathbf{Z}\mathbf{D}\mathbf{Z}'\mathbf{P}\mathbf{y} + \frac{\partial \log p(\boldsymbol{\sigma})}{\partial \sigma_g^2}$$
$$= -\frac{1}{2}\left(\frac{m}{\sigma_g^2} - \frac{trace\left(\mathbf{D}^{-1}\mathbf{C}^{gg|\mathbf{D}}\right)}{\left(\sigma_g^2\right)^2} - \frac{\hat{\mathbf{g}}'\mathbf{D}^{-1}\hat{\mathbf{g}}}{\left(\sigma_g^2\right)^2}\right) + \frac{\nu_g s_g^2}{2\left(\sigma_g^2\right)^2} - \frac{\left(\nu_g + 2\right)}{2\sigma_g^2} \quad [A10]$$

with $\mathbf{C}^{gg|\mathbf{D}}$ defined by Equation [A11]:

$$\begin{bmatrix} \mathbf{C}^{\beta\beta|\mathbf{D}} & \mathbf{C}^{\beta g|\mathbf{D}} \\ \mathbf{C}^{g\beta|\mathbf{D}} & \mathbf{C}^{gg|\mathbf{D}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{X}\sigma_e^{-2} & \mathbf{X}'\mathbf{Z}\sigma_e^{-2} \\ \mathbf{Z}'\mathbf{X}\sigma_e^{-2} & \mathbf{Z}'\mathbf{Z}\sigma_e^{-2} + \mathbf{D}^{-1}\sigma_g^{-2} \end{bmatrix}^{-1} \quad [A11]$$

and $\hat{\mathbf{e}}=\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}}-\mathbf{Z}\hat{\mathbf{g}}$.

The second derivative can be also obtained as described in Johnson and Thompson (1995). Inverting the coefficient matrix as in Equation [A11] is required to obtain $\mathbf{C}^{gg}$, however, this computation is nearly impossible with greater than tens of thousands of markers.

A reasonable strategy to use if *m* >> *n* is the animal effects model [A4], then back solve for SNP effect estimates using $\hat{\mathbf{g}} = \mathbf{D}\mathbf{Z}'\mathbf{G}^{-1}\hat{\mathbf{a}}$. When I adopt the animal effects model, the corresponding first derivatives are given by:

$$\frac{\partial \log p(\boldsymbol{\sigma},\boldsymbol{\tau}\,|\,\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y})}{\partial \sigma_e^2} = -\frac{1}{2}tr(\mathbf{P}) + \frac{1}{2}\mathbf{y}'\mathbf{P}\mathbf{y} + \frac{\partial \log p(\boldsymbol{\sigma})}{\partial \sigma_e^2}$$
$$= -\frac{1}{2}\left(\frac{n - rank(\mathbf{X})}{\sigma_e^2} - \left(\frac{1}{\sigma_e^2}\right)\left(n - \frac{trace\left(\mathbf{G}^{-1}\mathbf{C}^{aa|\mathbf{D}}\right)}{\sigma_g^2}\right) - \frac{\hat{\mathbf{e}}'\hat{\mathbf{e}}}{\left(\sigma_e^2\right)^2}\right) + \frac{\nu_e s_e^2}{2\left(\sigma_e^2\right)^2} - \frac{\left(\nu_e + 2\right)}{2\sigma_e^2} \quad [A12]$$

and

$$\frac{\partial \log p\left(\boldsymbol{\sigma},\boldsymbol{\tau}\,|\,\boldsymbol{\theta}_{-\boldsymbol{\sigma}},\mathbf{y}\right)}{\partial \sigma_g^2} = -\frac{1}{2}tr\left(\mathbf{PZDZ}'\right)+\frac{1}{2}\mathbf{y}'\mathbf{PZDZ}'\mathbf{Py}+\frac{\partial \log p\left(\boldsymbol{\sigma}\right)}{\partial \sigma_g^2}$$
$$= -\frac{1}{2}\left(\frac{n}{\sigma_g^2}-\frac{trace\left(\mathbf{G}^{-1}\mathbf{C}^{aa|\mathbf{D}}\right)}{\left(\sigma_g^2\right)^2}-\frac{\hat{\mathbf{a}}'\mathbf{G}^{-1}\hat{\mathbf{a}}}{\left(\sigma_g^2\right)^2}\right)+\frac{\nu_g s_g^2}{2\left(\sigma_g^2\right)^2}-\frac{\left(\nu_g+2\right)}{2\sigma_g^2} \qquad \text{[A13]}$$

with $\mathbf{C}^{aa|\mathbf{D}}$ defined as in Equation [A14]

$$\begin{bmatrix}\mathbf{C}^{\beta\beta|\mathbf{D}} & \mathbf{C}^{\beta a|\mathbf{D}}\\ \mathbf{C}^{a\beta|\mathbf{D}} & \mathbf{C}^{aa|\mathbf{D}}\end{bmatrix}=\begin{bmatrix}\mathbf{X}'\mathbf{X}\sigma_e^{-2} & \mathbf{X}'\mathbf{I}\sigma_e^{-2}\\ \mathbf{I}'\mathbf{X}\sigma_e^{-2} & \mathbf{I}'\mathbf{I}\sigma_e^{-2}+(\mathbf{ZDZ}')^{-1}\sigma_g^{-2}\end{bmatrix}^{-1} \qquad \text{[A14]}$$

Review of steps for MAP using animal-centric effects model and backtransforming to SNP-effects

I thereby highlight our MAP inference strategy as follows.

1. Set initial values for $\hat{\mathbf{g}}_{(0)}$, $\lambda_{(0)}=\sigma_{e(0)}^2/\sigma_{g(0)}^2$ and $t=1$.

2. Compute $\mathbf{G}_{(t)}=\mathbf{ZD}_{(t)}\mathbf{Z}'$ [A15]

where

$$\mathbf{D}_{(t)}=diag\left\{\left(\hat{\tau}_j\right)_{(t)}\right\}=diag\left\{\frac{\frac{\hat{g}_{j(t-1)}^2}{\sigma_{g(t-1)}^2}+\nu_g}{\nu_g-1}\right\} \qquad \text{[A16]}$$

for MAP-BayesA and

$$\mathbf{D}_{(t)}=diag\left(\hat{\tau}_{j(t)}+c\left(1-\hat{\tau}_{j(t)}\right)\right) \qquad \text{[A17]}$$

for MAP-SSVS with

$$\hat{\tau}_j=\frac{\phi_{\hat{g}_{j(t-1)}}\left(0,\sigma_g^2\right)\pi_{(t-1)}}{\phi_{\hat{g}_j(t-1)}\left(0,\sigma_{g_{(t-1)}}^2\right)\pi_{(t-1)}+\phi_{\hat{g}_{j(t-1)}}\left(0,\frac{\sigma_{g_{(t-1)}}^2}{c}\right)\left(1-\pi_{(t-1)}\right)} \qquad \text{[A18]}$$

3. Obtain $\mathbf{C}^{aa|\mathbf{D}}$ using

$$\begin{bmatrix} \mathbf{C}^{\beta\beta|\mathbf{D}} & \mathbf{C}^{\beta a|\mathbf{D}} \\ \mathbf{C}^{a\beta|\mathbf{D}} & \mathbf{C}^{aa|\mathbf{D}} \end{bmatrix}_{(t)} = \begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}' \\ \mathbf{X} & \mathbf{I}+\mathbf{G}_{(t)}^{-1}\lambda_{(t-1)} \end{bmatrix}^{-1} \sigma_e^2 \text{ and } \begin{bmatrix} \hat{\boldsymbol{\beta}}_t \\ \hat{\mathbf{a}}_t \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{I} \\ \mathbf{I}'\mathbf{X} & \mathbf{I}'\mathbf{I}+\mathbf{G}_{(t)}^{-1}\lambda_{(t-1)} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{Z}'\mathbf{y} \end{bmatrix}$$

[A19]

4. Compute $\hat{\mathbf{g}}_t = \mathbf{D}_{(t-1)}\mathbf{Z}'\mathbf{G}_{(t)}^{-1}\hat{\mathbf{a}}_{(t)}$ [A20]

5. Estimate variance components $\sigma^2_{e(t)}$ and $\sigma^2_{g(t)}$ from animal effects model using AIREML.

$\lambda_{(t)} = \sigma^2_{e(t)} / \sigma^2_{g(t)}$. Increment iterate number t to $t+1$.

6. Repeat Steps 2-5 until convergence.

Asymptotic standard errors of prediction under MAP

Asymptotic standard errors of prediction can be based on the observed information matrix for MAP-BayesA and MAP-SSVS:

$$\left[-\frac{\partial^2 \log\left(p\left(\boldsymbol{\eta} \mid \sigma_g^2, \sigma_e^2, \mathbf{y}\right)\right)}{\partial \boldsymbol{\eta} \partial \boldsymbol{\eta}'} \right]^{-1} \qquad [A21]$$

where $\log\left(p\left(\boldsymbol{\eta} \mid \sigma_g^2, \sigma_e^2, \mathbf{y}\right)\right)$ denotes the log posterior density of $\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\beta} \\ \mathbf{g} \end{bmatrix}$ conditional on the variance components but with the uncertainty on **D** integrated out. Using Louis (1982), I can derive Expression [A21] for MAP-BayesA in Equation [A22].

$$\begin{aligned} & -\frac{\partial^2 \log\left(p\left(\boldsymbol{\eta} \mid \sigma_g^2, \sigma_e^2, \mathbf{y}\right)\right)}{\partial \boldsymbol{\eta}^2} \\ & = -\int_{\mathbf{D}^{-1}} \frac{\partial^2 \log\left(p\left(\boldsymbol{\eta} \mid \sigma_g^2, \sigma_e^2, \mathbf{y}, \mathbf{D}^{-1}\right)\right)}{\partial \boldsymbol{\eta}^2} p\left(\mathbf{D}^{-1} \mid \boldsymbol{\eta}, \sigma_g^2, \sigma_e^2, \mathbf{Y}\right) d\mathbf{D}^{-1} - \underset{\mathbf{D}^{-1}|\boldsymbol{\eta},\mathbf{Y},\sigma_g^2,\sigma_e^2}{\text{var}} \left\{ \frac{\partial \log\left(p\left(\boldsymbol{\eta} \mid \sigma_g^2, \sigma_e^2, \mathbf{y}, \mathbf{D}^{-1}\right)\right)}{\partial \boldsymbol{\eta}} \right\} \\ & = \begin{bmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{Z}'\mathbf{R}^{-1}\mathbf{Z} + diag\left(\frac{1}{\bar{\sigma}^2_{g_j}} \right) \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & diag\left(\frac{2g_i^2}{\nu_\tau + 1} \left(\frac{1}{\bar{\sigma}^2_{g_j}} \right)^2 \right) \end{bmatrix} \qquad [A22] \\ & = \begin{bmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{Z}'\mathbf{R}^{-1}\mathbf{Z} + \boldsymbol{\Delta}^{-1} \end{bmatrix} \end{aligned}$$

where $\bar{\sigma}^2_{g_j} = \frac{g_i^{\,2} + \nu_\tau \sigma^2_g}{\nu_\tau + 1}$ and $\boldsymbol{\Delta}^{-1} = diag\left(\frac{1}{\bar{\sigma}^2_{g_i}} \left(1 - \frac{2g_i^2}{(\nu_\tau + 1)\bar{\sigma}^2_{g_i}} \right) \right)$.

Then I can obtain the asymptotic prediction error (co)variance (PEV) matrix ($\mathbf{C}^{gg}$) of the SNP effect estimates from the random by random portion of the inverse of Equation [A22]. For MAP-SSVS, the observed information matrix can be similarly obtained from Equation [A22] except that:

$$\boldsymbol{\Delta}^{-1} = diag\left(\frac{1}{\sigma^2_g} \left(diag\left(\hat{\tau}_j + c\left(1 - \hat{\tau}_j\right) \right) - \frac{g_j^2 \hat{\tau}_j \left(1 - \hat{\tau}_j\right)\left(1 - c\right)^2}{\sigma^2_g} \right) \right) \qquad [A23]$$

As inverting Expression [A22] is difficult for large *m*, I base inference on an animal effects model using

$$\begin{bmatrix} \mathbf{C}^{\beta\beta} & \mathbf{C}^{\beta a} \\ \mathbf{C}^{a\beta} & \mathbf{C}^{aa} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{I} \\ \mathbf{I}'\mathbf{X} & \mathbf{I}'\mathbf{I} + \mathbf{G}^{*-1}\lambda \end{bmatrix}^{-1} \sigma^2_e \qquad [A24]$$

where $\mathbf{G}^{*-1} = (\mathbf{Z}\boldsymbol{\Delta}\mathbf{Z}')^{-1}$ and $\lambda = \sigma^2_e / \sigma^2_g$. Note that the prediction error (co)variance matrix $\mathbf{C}^{gg} = PEV(\hat{\mathbf{g}})$ for the SNP effects can be derived from the prediction error (co)variance matrix $\mathbf{C}^{gg} = PEV(\hat{\mathbf{g}})$ for the animal effects. By definition,

$$\mathbf{C}^{gg} = PEV(\hat{\mathbf{g}}) = \text{var}(\mathbf{g} - \hat{\mathbf{g}}) = \text{var}(\mathbf{g}) - \text{var}(\hat{\mathbf{g}}) \qquad [A25]$$

and

$$\mathbf{C}^{aa} = PEV(\hat{\mathbf{a}}) = \text{var}(\mathbf{a} - \hat{\mathbf{a}}) = \text{var}(\mathbf{a}) - \text{var}(\hat{\mathbf{a}}) \qquad [A26]$$

Noting that $\hat{\mathbf{g}} = \mathbf{D}\mathbf{Z}'\mathbf{G}^{*-1}\hat{\mathbf{a}}$, I then have

$$\text{var}(\hat{\mathbf{g}}) = \text{var}(\mathbf{D}\mathbf{Z}'\mathbf{G}^{*-1}\hat{\mathbf{a}}) = \mathbf{D}\mathbf{Z}'\mathbf{G}^{*-1}\,\text{var}(\hat{\mathbf{a}})\mathbf{G}^{*-1}\mathbf{Z} \qquad [A27]$$

where

$$\text{var}(\hat{\mathbf{a}}) = \text{var}(\mathbf{a}) - \mathbf{C}^{aa} = \mathbf{G}^* \sigma^2_g - \mathbf{C}^{aa} \qquad [A28]$$

Using [A26], [A27] and [A28] in [A25], $\mathbf{C}^{gg}$ can be derived from $\mathbf{C}^{aa}$.

$$\begin{aligned}\mathbf{C}^{gg} &= \text{var}(\mathbf{g}) - \text{var}(\hat{\mathbf{g}}) = \mathbf{D}\sigma_g^2 - \mathbf{D}\mathbf{Z}'\mathbf{G}^{*-1}\,\text{var}(\hat{\mathbf{a}})\mathbf{G}^{*-1}\mathbf{Z}\mathbf{D} \\ &= \mathbf{D}\sigma_g^2 - \mathbf{D}\mathbf{Z}'\mathbf{G}^{*-1}(\mathbf{G}^*\sigma_g^2 - \mathbf{C}^{aa})\mathbf{G}^{*-1}\mathbf{Z}\mathbf{D}\end{aligned} \qquad [A29]$$

An important feature of Equation [A29] is that just diagonals of $\mathbf{C}^{gg}$ (for single SNP associations) or block diagonals of $\mathbf{C}^{gg}$ (for windows based inference) can be readily computed without computing all of [A29]. For example, suppose I write $\mathbf{M} = \mathbf{D}\mathbf{Z}'\mathbf{G}^{*-1}$. Hence for $\mathbf{m}'_j$ being row *j* of **M**, the corresponding j^{th} diagonal element, $c^{gg}_{j,j}$, of $\mathbf{C}^{gg}$, used to derive either a EMMAX, RRBLUP, MAP-BayesA or MAP-SSVS test for SNP *j*, can be determined as a function of a simple quadratic form; i.e.

$$c^{gg}_{j,j} = d_{j,j}\sigma_g^2 - \mathbf{m}'_j(\mathbf{G}^*\sigma_g^2 - \mathbf{C}^{aa})\mathbf{m}_j \qquad [A30]$$

Similarly, if one conducts windows based inference where $\mathbf{M}'_k$ denotes the subset of rows of **M** pertaining to the n_k SNP markers in window *k*, then the corresponding block diagonal $\mathbf{C}^{gg}_k$ of $\mathbf{C}^{gg}$ for window *k* can be written simply:

$$\mathbf{C}^{gg}_k = \mathbf{D}_k\sigma_g^2 - \mathbf{M}'_k(\mathbf{G}^*\sigma_g^2 - \mathbf{C}^{aa})\mathbf{M}_k \qquad [A31]$$

Here $\mathbf{D}_k$ is the diagonal sub-block of $\mathbf{D}$ pertaining to the n_k SNPs in window *k*.

Full Conditional Densities (FCD) for Markov Chain Monte Carlo Inference (MCMC) in BayesA and SSVS models

Recall the joint posterior density from Equation [3.7] as also provided again below

$$p\left(\boldsymbol{\beta},\mathbf{g},\boldsymbol{\tau},\sigma_e^2,\sigma_g^2,\theta_\tau \,|\, \mathbf{y}\right)$$
$$\propto \left(\prod_{i=1}^{n} p\left(y_i|\boldsymbol{\beta},\mathbf{g},\sigma_e^2\right)\right)\left(\prod_{j=1}^{m} p\left(g_j|\sigma_g^2,\tau_j\right)p\left(\tau_j\,|\,\theta_\tau\right)\right)p(\boldsymbol{\beta})\,p\left(\sigma_g^2\,|\,v_g,s_g^2\right)p\left(\sigma_e^2\,|\,v_e,s_e^2\right)p\left(\theta_\tau\right)$$

For the fixed effects, suppose the design matrix is $n\times p$, write:

$$\mathbf{X}_{nxp}=\begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1p} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2p} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & x_{n3} & \cdots & x_{np} \end{bmatrix}=\begin{bmatrix} \mathbf{x}_1' \\ \mathbf{x}_2' \\ \mathbf{x}_3' \\ \vdots \\ \mathbf{x}_n' \end{bmatrix}=\begin{bmatrix} \mathbf{x}_{.1} & \mathbf{x}_{.2} & \mathbf{x}_{.3} & & \mathbf{x}_{.p} \end{bmatrix} \qquad \text{[A32]}$$

Here $\mathbf{x}_{.j}$ is the vector of covariates or dummy variables for element *j* of the fixed effects.

For the marker effect, suppose design matrix is $n\times m$, write

$$\mathbf{Z}=\begin{bmatrix} z_{11} & z_{12} & z_{13} & \cdots & z_{1m} \\ z_{21} & z_{22} & z_{23} & \cdots & z_{2m} \\ z_{31} & z_{32} & z_{33} & \cdots & z_{3m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ z_{n1} & z_{n2} & z_{n3} & \cdots & z_{nm} \end{bmatrix}=\begin{bmatrix} \mathbf{z}_1' \\ \mathbf{z}_2' \\ \mathbf{z}_3' \\ \vdots \\ \mathbf{z}_n' \end{bmatrix}=\begin{bmatrix} \mathbf{z}_{.1} & \mathbf{z}_{.2} & \mathbf{z}_{.3} & \cdots & \mathbf{z}_{.m} \end{bmatrix} \qquad \text{[A33]}$$

where $\mathbf{z}_{.j}$ is the vector of genotype values for SNP marker *j*

Then the fully conditional distribution of any unknown parameters are outlined below, first for MCMC-BayesA and then for MCMC-SSVS.

Full conditional densities (FCD) under MCMC-BayesA

FCD for Fixed Effects

$$p(\beta_j \,|\, ELSE) \sim N\left(\tilde{\beta}_j,\tilde{v}_{\beta j}\right) \qquad \text{[A34]}$$

with

$$\tilde{\beta}_j = \frac{\mathbf{x}_{.j}^{'}\left(\left(\mathbf{y\text{-}X\beta} - \mathbf{Zg}\right) + \mathbf{x}_{.j}\beta_j\right)}{\mathbf{x}_{.j}^{'}\mathbf{x}_{.j}}$$

$$= \frac{\mathbf{x}_{.j}^{'}\left(\mathbf{e} + \mathbf{x}_{.j}\beta_j\right)}{\mathbf{x}_{.j}^{'}\mathbf{x}_{.j}} = \frac{\left(\mathbf{x}_{.j}^{'}\mathbf{e} + \mathbf{x}_{.j}^{'}\mathbf{x}_{.j}\beta_j\right)}{\mathbf{x}_{.j}^{'}\mathbf{x}_{.j}}$$

$$= \frac{\left(\sum_{i=1}^{n} x_{ij}e_i + \left(\sum_{i=1}^{n} x_{ij}^2\right)\beta_j\right)}{\left(\sum_{i=1}^{n} x_{ij}^2\right)}$$

and $\tilde{v}_{\beta j} = \sigma_e^2\left(\mathbf{x}_{.j}^{'}\mathbf{x}_{.j}\right)^{-1} = \sigma_e^2\left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1}$.

FCD for Marker effects

$$p\left(g_j \mid \mathbf{ELSE}\right) \propto \left(\prod_{i=1}^{n} p\left(y_i|\boldsymbol{\beta}, \mathbf{g}\right)\right) p\left(\mathbf{g}_j \mid \sigma_g^2, \tau_j\right) \sim N\left(\tilde{g}_j, \tilde{v}_{gj}\right);\ j = 1, 2, \ldots, m \qquad \text{[A35]}$$

where

$$\tilde{g}_j = \frac{\mathbf{z}_{.j}^{'}\left(\left(\mathbf{y\text{-}X\beta} - \mathbf{Zg}\right) + \mathbf{z}_{.j}g_j\right)}{\mathbf{z}_{.j}^{'}\mathbf{z}_{.j} + \sigma_e^2\left(\sigma_g^2\tau_j\right)^{-1}}$$

$$= \frac{\mathbf{z}_{.j}^{'}\left(\mathbf{e} + \mathbf{z}_{.j}^{*}g_j\right)}{\mathbf{z}_{.j}^{'}\mathbf{z}_{.j} + \sigma_e^2\left(\sigma_g^2\tau_j\right)^{-1}} = \frac{\mathbf{z}_{.j}^{'}\mathbf{e} + \mathbf{z}_{.j}^{'}\mathbf{z}_{.j}g_j}{\mathbf{z}_{.j}^{'}\mathbf{z}_{.j} + \sigma_e^2\left(\sigma_g^2\tau_j\right)^{-1}}$$

$$= \frac{\sum_{i=1}^{n} z_{ij}e_i + \left(\sum_{i=1}^{n} z_{ij}^2\right)g_j}{\left(\sum_{i=1}^{n} z_{ij}^2 + \sigma_e^2\left(\sigma_g^2\tau_j\right)^{-1}\right)}$$

and $\tilde{v}_{gj} = \left(\frac{\sum_{i=1}^{n}\left(z_{ij}\right)^2}{\sigma_e^2} + \left(\sigma_g^2\tau_j\right)^{-1}\right)^{-1}$

FCD for Marker-Specific Augmented Variables.

$$p\left(\tau_j \mid \mathbf{y},\mathbf{ELSE}\right) \propto p\left(g_j \mid \tau_j, \sigma_g^2\right) p\left(\tau_j \mid \nu_\tau\right)$$

$$\propto \left(2\pi\sigma_g^2\tau_j\right)^{-1/2} \exp\left(-\frac{1}{2\sigma_g^2\tau_j} g_j^{\,2}\right) \frac{\left(\frac{\nu_\tau}{2}\right)^{\frac{\nu_\tau}{2}}}{\Gamma\left(\frac{\nu_\tau}{2}\right)} \tau_j^{-\left(\frac{\nu_\tau}{2}+1\right)} e^{-\frac{\nu_\tau}{2\tau_j}} \qquad \text{[A36]}$$

$$\propto \left(\tau_j\right)^{-\left(\frac{\nu_\tau+1}{2}+1\right)} \exp\left(-\frac{1}{2}\frac{\left(\frac{g_j^2}{\sigma_g^2}+\nu_\tau\right)}{\tau_j}\right)$$

i.e., a scaled inverted chi-square density with degrees of freedom $\nu_\tau + 1$ and scale parameter $\frac{g_j^2}{\sigma_g^2} + \nu_\tau$.

FCD for Genetic Variance Component

$$p\left(\sigma_g^2 \mid \mathbf{y},\mathbf{ELSE}\right) \propto \prod_{j=1}^{m} p\left(g_j \mid \tau_j, \sigma_g^2\right) p\left(\sigma_g^2 \mid \nu_g, s_g^2\right)$$

$$\propto \left(\sigma_g^2\right)^{-\frac{m}{2}} \exp\left(-\frac{1}{2\sigma_g^2}\sum_{j=1}^{G}\frac{g_j^2}{\tau_j}\right) \frac{\left(\frac{\nu_g}{2}\right)^{\frac{\nu_g}{2}}}{\Gamma\left(\frac{\nu_\tau}{2}\right)} \sigma_g^{2\,-\left(\frac{\nu_g}{2}+1\right)} e^{-\frac{\nu_g s_g^2}{2\sigma_g^2}} \qquad \text{[A37]}$$

$$\propto \left(\sigma_g^2\right)^{-\frac{m+\nu_g}{2}+1} \exp\left(-\frac{1}{2\sigma_g^2}\left(\sum_{j=1}^{G}\frac{g_j^2}{\tau_j} + \nu_g s_g^2\right)\right)$$

i.e., a scaled inverted chi-square density with degrees of freedom $\nu_g + m$ and scale parameter $\sum_{j=1}^{m}\frac{g_j^2}{\tau_j} + \nu_g s_g^2$.

I adopt the Metropolis-Hastings sampling strategy provided by Yang *et al.* (2015b) to sample σ_g^2 with uncertainty on the τ_j's integrated out.

$$p\left(\sigma_g^2 \mid \mathbf{y}, \mathbf{ELSE}\right) \propto \left(\prod_{j=1}^{m} p\left(g_j \mid \nu_\tau, \sigma_g^2\right)\right) p\left(\sigma_g^2 \mid \alpha_g, \beta_g\right)$$
$$\propto \left(\prod_{j=1}^{m} \frac{\Gamma\left(\frac{\nu_\tau+1}{2}\right)}{\Gamma\left(\frac{\nu_\tau}{2}\right)} \left(\frac{1}{\nu_\tau \sigma_g^2}\right)^{1/2} \left(1+\frac{g_j^{\,2}}{\nu_\tau \sigma_g^2}\right)^{-\frac{\nu_\tau+1}{2}}\right) \left(\sigma_g^2\right)^{\alpha_g - 1} e^{-\beta_g \sigma_g^2} \qquad \text{[A38]}$$

Residual variance

$$p\left(\sigma_e^2 \mid ELSE\right) \propto \left(2\pi\sigma_e^2\right)^{-n/2} \exp\left(-\frac{1}{2\sigma_e^2}\sum_{j=1}^{n} e_j^2\right) \sigma_e^{2\,-\left(\frac{\nu_e}{2}+1\right)} e^{-\frac{\nu_e s_e^2}{2\sigma_e^2}}$$
$$\propto \sigma_e^{2\,-\left(\frac{\nu_e+n}{2}+1\right)} \exp\left(-\frac{1}{2\sigma_e^2}\left(\sum_{j=1}^{n} e_j^2 + \nu_e s_e^2\right)\right) \qquad \text{[A39]}$$

i.e., $\chi^{-2}\left(\nu_e + n, \sum_{j=1}^{n} e_j^2 + \nu_e s_e^2\right)$ with $e_j = y_j - \mathbf{x}_j^{'}\boldsymbol{\beta} - \mathbf{z}_j^{'}\mathbf{g}$

Full Conditional Densities for MCMC-SSVS

FCD for Fixed effects: same as that for MCMC-BayesA

FCD for Marker Effects

$$p\left(g_j \mid \mathbf{ELSE}\right) \propto \left(\prod_{i=1}^{n} p\left(y_i \mid \boldsymbol{\beta}, \mathbf{g}\right)\right) p\left(\mathbf{g}_j \mid \sigma_g^2, \tau_j\right) \sim N\left(\tilde{g}_j, \tilde{v}_{gj}\right);\ j = 1, 2, \ldots, m \qquad \text{[A40]}$$

where

$$\tilde{g}_j = \frac{\mathbf{z}_{.j}^{'}\left(\left(\mathbf{y}\text{-}\mathbf{X}\boldsymbol{\beta}-\mathbf{Z}\mathbf{g}\right)+\mathbf{z}_{.j}g_j\right)}{\mathbf{z}_{.j}^{'}\mathbf{z}_{.j}+\sigma_e^2\left(\sigma_g^2\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)\right)^{-1}}$$

$$= \frac{\mathbf{z}_{.j}^{'}\left(\mathbf{e}+\mathbf{z}_{.j}^{*}g_j\right)}{\mathbf{z}_{.j}^{'}\mathbf{z}_{.j}+\sigma_e^2\left(\sigma_g^2\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)\right)^{-1}} = \frac{\mathbf{z}_{.j}^{'}\mathbf{e}+\mathbf{z}_{.j}^{'}\mathbf{z}_{.j}g_j}{\mathbf{z}_{.j}^{'}\mathbf{z}_{.j}+\sigma_e^2\left(\sigma_g^2\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)\right)^{-1}}$$

$$= \frac{\sum_{i=1}^{n} z_{ij}e_i + \left(\sum_{i=1}^{n} z_{ij}^2\right)g_j}{\left(\sum_{i=1}^{n} z_{ij}^2+\sigma_e^2\left(\sigma_g^2\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)\right)^{-1}\right)}$$

and $\tilde{v}_{gj} = \left(\frac{\sum_{i=1}^{n}\left(z_{ij}\right)^2}{\sigma_e^2}+\left(\sigma_g^2\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)\right)^{-1}\right)^{-1}$

FCD for Marker-Specific Augmented (i.e. Indicator) variables τ_j

$$p\left(\tau_j \mid ELSE\right) \propto p\left(g_j \mid \sigma_g^2, \tau_j\right) p\left(\tau_j \mid \pi_\tau\right)$$

$$\propto \frac{1}{\sqrt{2\pi\sigma_g^2\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)}} \exp\left(-\frac{1}{2}\frac{g_j^2}{\sigma_g^2\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)}\right) \pi_\tau^{\tau_j}\left(1-\pi_\tau\right)^{1-\tau_j} \qquad \text{[A41]}$$

Such that it can be readily determined that:

$$\text{Prob}\left(\tau_j = 1 \mid \text{ELSE except } g_j\right) = \frac{h_1\pi_\tau}{h_0\left(1-\pi_\tau\right)+h_1\pi_\tau} \qquad \text{[A42]}$$

where $h_1 = p\left(g_j \mid \sigma_g^2, \tau_j = 1\right) \propto \frac{1}{\sqrt{\sigma_g^2}} \exp\left(-\frac{1}{2}\frac{g_j^2}{\sigma_g^2}\right)$ and

$$h_0 = p\left(g_j \mid \sigma_g^2, \tau_j = 0\right) \propto \frac{1}{\sqrt{\frac{\sigma_g^2}{c}}} \exp\left(-\frac{1}{2}\frac{g_j^2}{\frac{\sigma_g^2}{c}}\right)$$

such that $\text{Prob}\left(\tau_j = 1 \mid \text{ELSE except } g_j\right) = \frac{h_1\pi_\tau}{h_0\left(1-\pi_\tau\right)+h_1\pi_\tau} = \frac{\pi_\tau}{\frac{h_0}{h_1}\left(1-\pi_\tau\right)+\pi_\tau}$; i.e., τ_j can be drawn from a Bernoulli distribution.

FCD for genetic variance component

$$p\left(\sigma_g^2 \mid \mathbf{y,ELSE}\right) \propto \prod_{j=1}^{m} p\left(g_j \mid \tau_j, \sigma_g^2\right) p\left(\sigma_g^2 \mid \nu_g, s_g^2\right)$$

$$\propto \left(\sigma_g^2\right)^{-\frac{m}{2}} \exp\left(-\frac{1}{2\sigma_g^2}\sum_{j=1}^{G}\frac{g_j^2}{\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)}\right)\sigma_g^{2\,-\left(\frac{\nu_g}{2}+1\right)} e^{-\frac{\nu_g s_g^2}{2\sigma_g^2}}$$

$$\propto \left(\sigma_g^2\right)^{-\frac{m+\nu_g}{2}+1} \exp\left(-\frac{1}{2\sigma_g^2}\left(\sum_{j=1}^{G}\frac{g_j^2}{\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)}+\nu_g s_g^2\right)\right) \qquad \text{[A43]}$$

i.e., a scaled inverted chi-square density with degrees of freedom $\nu_g + m$ and scale parameter

$$\sum_{j=1}^{m}\frac{g_j^2}{\left(\frac{\left(1-\tau_j\right)}{c}+\tau_j\right)} + \nu_g s_g^2 .$$

FCD for Residual variance: same as that for MCMC-BayesA.

Supplementary Tables and Figures

Table A.1 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different methods for inferring associations based on non-overlapping genomic windows of length 0.5Mb. Comparisons are made within different specifications of shape parameter (γ) for Gamma distribution of quantitative trait loci (QTL) and number of QTL (n_{qtl})

	Methods					
Factors	SSVS	BayesA	EMMAX	MAP-SSVS	MAP-BayesA	RRBLUP
Shape						
0.18	2.71[a]	2.63[a]	1.80[b]	0.79[c,]	0.56[d,]	0.54[d, *]
1.48	4.15[a]	4.19[a]	2.73[b]	0.82[c,]	0.54[d,]	0.38[e, *]
3	4.31[a]	4.90[a]	2.92[b]	0.68[c,]	0.42[d,]	0.27[e, *]
n_{qtl}						
30	6.54[a]	6.96[a]	3.94[b]	1.52[c]	0.59[d,]	0.34[e, *]
90	3.75[a]	3.73[a]	2.42[b]	0.67[c,]	0.52[c,]	0.41[d, *]
300	1.98[a]	2.08[a]	1.50[b]	0.43[c,]	0.41[c,]	0.40[c, *]
Overall	3.65[a]	3.78[a]	2.43[b]	0.76[c,]	0.50[d,]	0.38[e, *]

Values not sharing the same letter within a row have different (P <0.05) relative pAUC05 within the row. * indicates the corresponding method is worse than a random classifier (pAUC05 = 1). Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (30, 100, or 300) of markers chosen to be quantitative trait loci (QTL) from the MSUPRP genotypes, and shape parameter (0.18,1.48, or 3.00) for Gamma distribution of QTL effects

Table A.2 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different methods for inferring associations based on non-overlapping genomic windows of length 2Mb. Comparisons are made within different specifications of shape parameter (γ) for Gamma distribution of quantitative trait loci (QTL) and number of QTL (n_{qtl})

	Methods					
Factors	SSVS	BayesA	EMMAX	MAP-SSVS	MAP-BayesA	RRBLUP
Shape γ						
0.18	2.87[a]	2.83[a]	1.67[b]	0.62[c, *]	0.65[c, *]	0.43[d, *]
1.48	4.33[a]	4.23[a]	2.15[b]	0.49[c, *]	0.31[d, *]	0.24[e, *]
3	4.91[a]	5.06[a]	2.07[b]	0.49[c, *]	0.33[d, *]	0.21[e, *]
n_{qtl}						
30	7.15[a]	7.17[a]	3.12[b]	1.14[c]	0.67[d, *]	0.33[e, *]
90	3.72[a]	3.76[a]	1.76[b]	0.44[c, *]	0.37[c, d,]	0.29[d, *]
300	2.30[a]	2.24[a]	1.36[b]	0.30[c, *]	0.27[c, *]	0.22[c, *]
Overall	3.94[a]	3.92[a]	1.95[b]	0.53[c, *]	0.41[c, *]	0.28[d, *]

Values not sharing the same letter within a row have different ($P < 0.05$) relative pAUC05 within the row. *indicates the corresponding method is worse than a random classifier (pAUC05 = 1). Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (n_{qtl} = 30, 100, or 300) of quantitative trait loci (QTL), and shape parameter (γ=0.18, 1.48, or 3.00) for Gamma distribution of QTL effects.

Table A.3 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different methods for inferring associations based on non-overlapping genomic windows of length 3Mb. Comparisons are made within different specifications of shape parameter (γ) for Gamma distribution of quantitative trait loci (QTL) and number of QTL (n_{qtl})

	Methods					
Factors	SSVS	BayesA	EMMAX	MAP-SSVS	MAP-BayesA	RRBLUP
Shape						
0.18	2.91[a]	2.77[a]	1.64[b]	0.73[c, *]	0.66[c, *]	0.37[d, *]
1.48	4.43[a]	4.30[a]	1.98[b]	0.55[c, *]	0.49[c, *]	0.24[d, *]
3	5.10[a]	5.10[a]	1.95[b]	0.50[c, *]	0.43[d, *]	0.17[e, *]
n_{qtl}						
30	7.52[a]	7.17[a]	2.93[b]	1.16[c]	0.96[c, *]	0.34[d, *]
90	3.64[a]	3.72[a]	1.71[b]	0.61[c, *]	0.48[c, *]	0.24[d, *]
300	2.41[a]	2.28[a]	1.26[b]	0.34[c, *]	0.30[c, *]	0.16[d, *]
Overall	4.044[a]	3.93[a]	1.85[b]	0.62[c, *]	0.52[c, *]	0.23[d, *]

Values not sharing the same letter within a row have different ($P < 0.05$) relative pAUC05 within the row. *indicates the corresponding method is worse than a random classifier (pAUC05 = 1). Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (n_{qtl} = 30, 100, or 300) of quantitative trait loci (QTL), and shape parameter (γ=0.18, 1.48, or 3.00) for Gamma distribution of QTL effects.

Table A.4 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different specifications of degrees of freedom hyperparameter (ν_g = 2.5 versus ν_g = 5.0) using MCMC-BayesA. Comparisons are made within different specifications of shape parameter (γ) for Gamma distribution of quantitative trait loci (QTL) and number of QTL (n_{qtl})

Factors	ν_g =2.5	ν_g =5
Shape γ		
0.18	3.60[a]	2.38[b]
1.48	5.87[a]	4.87[b]
3	6.74[a]	4.88[b]
n_{qtl}		
30	9.03[a]	4.79[b]
90	4.98[a]	3.77[b]
300	3.17[a]	3.14[a]
Overall	5.22[a]	3.84[b]

Values not sharing the same letter within a row have different (P <0.05) relative pAUC05 within the row. Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (n_{qtl} = 30, 100, or 300) of quantitative trait loci (QTL), and shape parameter (γ = 0.18, 1.48, or 3.00) for Gamma distribution of QTL effects.

Table A.5 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different sets of starting values for SNP effects (MCMC-SSVS vs RRBLUP) for MAP-SSVS. Comparisons are made within different specifications of number of quantitative trait loci (n_{qtl})

Window specification	Factor	Starting values for MAP-SSVS	
		MCMC	RRBLUP
	n_{qtl}		
	30	4.47[a]	3.79[b]
Single SNP	90	2.65[a]	2.49[b]
	300	1.72[a]	1.64[a]
	Overall	2.73[a]	2.49[b]
1Mb window	Overall	0.94[a, *]	0.70[b, *]
	n_{qtl}		
	30	2.36[a]	1.76[b]
Adaptive	90	0.98[a, *]	0.80[b, *]
	300	0.68[a, *]	0.66[a, *]
	Overall	1.16[a]	0.97[b, *]

Values not sharing the same letter within a row have different (P <0.05) relative pAUC05 within the row. *indicates the corresponding method is not better than a random classifier (pAUC05=1). Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (n_{qtl} = 30, 100, or 300) of quantitative trait loci (QTL), and shape parameter (γ = 0.18, 1.48, or 3.00) for Gamma distribution of QTL effects.

Table A.6 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different sets of starting values for SNP effects (MCMC-BayesA vs RRBLUP) for MAP-BayesA. Comparisons are made within different specifications of number of quantitative trait loci (n_{qtl})

		Starting values for SNP	
		MCMC-	RRBLUP
Single SNP	Overall	2.68[a]	2.76[a]
1Mb window	Overall	0.75[a, *]	0.46[b, *]
	n_{qtl}		
Adaptive	30	2.33[a]	0.87[b, *]
	90	1.14[a]	0.50[b, *]
	300	0.73[a, *]	0.62[a, *]
	Overall	1.25[a]	0.65[b, *]

Values not sharing the same letter within a row have different (P <0.05) relative pAUC05 within the row. * indicates the corresponding method is not better than a random classifier (pAUC05=1). Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (n_{qtl} = 30, 100, or 300) of quantitative trait loci (QTL), and shape parameter (γ = 0.18, 1.48, or 3.00) for Gamma distribution of QTL effects.

.

Table A.7 Least squares mean relative (random classifier = 1) partial areas under a receiving operating characteristic curve up until a false positive rate of 5% (pAUC05) for different methods for inferring associations averaging across all window size determinations (single SNP, 0.5 Mb, 1.0Mb, 2.0Mb, 3.0Mb and adaptive windows) based on two different methods for inferring posterior probabilities of association: 1) That proposed by Fernando et al., 2014 and 2) that proposed by Moser et al., 2015. Comparisons are made within different specifications of shape parameter (γ) for Gamma distribution of quantitative trait loci (QTL) and number of QTL (n_{qtl})

Factor	PPA determination strategy	
	Fernando et al., (2014)	Moser et al., (2015)
Shape γ		
0.18	3.03[a]	3.03[a]
1.48	4.60[a]	4.43[a]
3	5.08[a]	4.56[b]
n_{qtl}		
30	7.24[a]	7.23[a]
90	4.04[a]	3.99[a]
300	2.42[a]	2.11[b]
Overall	4.14[a]	3.94[b]

Values not sharing the same letter within a row have different ($P < 0.05$) relative pAUC05 within the row. Mean estimates based on 10 replicates per each of 9 populations of 3 x 3 factorial on number (n_{qtl} = 30, 100, or 300) of quantitative trait loci (QTL), and shape parameter (γ = 0.18, 1.48, or 3.00) for Gamma distribution of QTL effects.

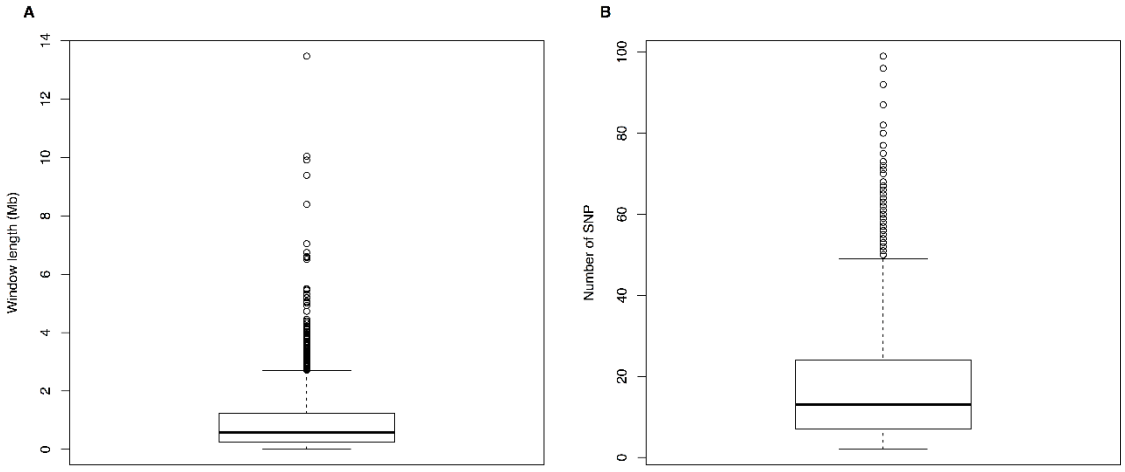

Figure A.1 Boxplot of window lengths for windows adaptively chosen based on the BALD software in terms of mega bases (Panel A) and number of SNP markers (Panel B)

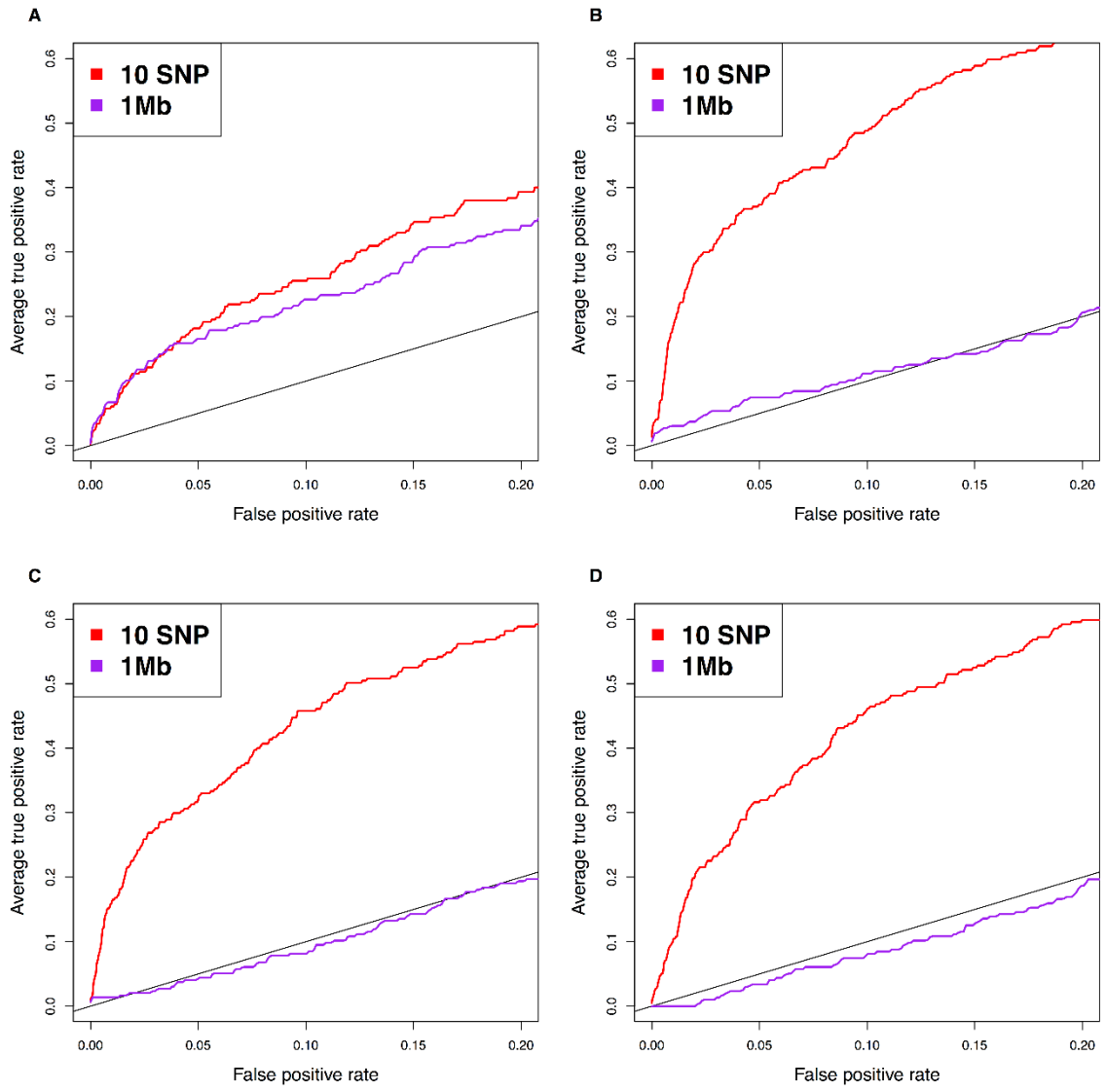

Figure A.2 Average ROC curve (10 replicates) for 1Mb versus 10 SNP windows using EMMAX (Panel A), MAP-SSVS (Panel B), MAP-BayesA (Panel C) and RRBLUP (Panel D) for 30 quantitative trait loci generated from a Gamma distribution with shape parameter 1.48

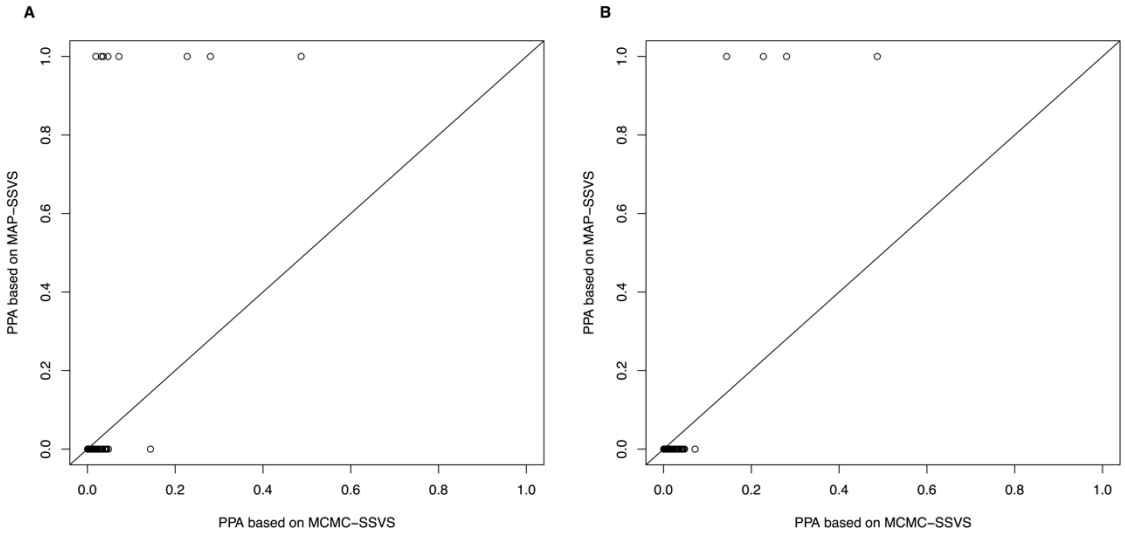

Figure A.3 Scatterplots of posterior probabilities of association (PPA) for MCMC-SSVS (x-axis) versus MAP-SSVS (y-axis) for analysis on 13th rib backfat on 922 pigs from the MSUPRP population based on with different starting values for MAP-SSVS: A) RRBLUP and B) MCMC-SSVS.

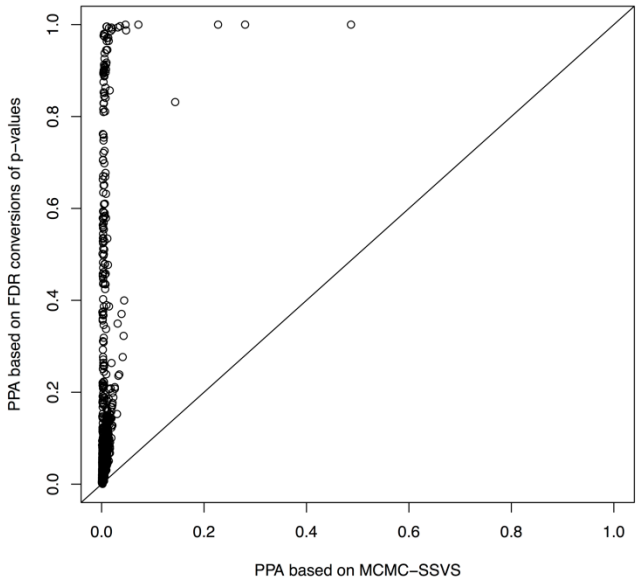

Figure A.4 Scatterplot of posterior probabilities of association (PPA) based on local false discovery rates (lFDR) conversions of *p*-values from EMMAX procedure (y-axis: PPA=1-lFDR) and MCMC-SSVS (x-axis) on 13th rib backfat on 922 pigs from the MSUPRP population.

Appendix B: Chapter 4

Implementation Details for Monte Carlo Markov Chain (MCMC) inferences in ssSSVS

Recall the joint posterior density from Equation [13] as also provided again below

$$p\left(\boldsymbol{\beta},\alpha_1,\alpha_2,...,\alpha_m,\boldsymbol{\varepsilon},\sigma_u^2,\sigma_\alpha^2,\sigma_e^2\,|\,\mathbf{y}\right)\propto\left(\prod_{i=1}^{n}p\left(y_i\,|\,\boldsymbol{\beta},\alpha_1,\alpha_2,...,\alpha_m,\boldsymbol{\varepsilon},\sigma_e^2\right)\right)$$

$$\left(\prod_{j=1}^{m}p\left(\alpha_j\,|\,\sigma_\alpha^2,\phi_j\right)p(\phi_j\,|\,\pi)\right)p\left(\boldsymbol{\varepsilon}\,|\,\sigma_u^2\right)p\left(\sigma_\alpha^2\,|\,s_\alpha^2,\nu_\alpha\right)p\left(\sigma_u^2\,|\,s_u^2,\nu_u\right)p\left(\sigma_e^2\,|\,\nu_e,s_e^2\right)p\left(\pi\,|\,\alpha_0,\beta_0\right)$$

$$\propto\left(\sigma_e^2\right)^{-n/2}\exp\left(-\frac{1}{2\sigma_e^2}\sum_{i=1}^{n}\left(y_i-\mathbf{x}_i'\boldsymbol{\beta}-\mathbf{w}_i'\boldsymbol{\alpha}-\mathbf{u}_i'\boldsymbol{\varepsilon}\right)^2\right)\left(\sigma_u^2\right)^{-q_1/2}\exp\left(-\frac{1}{2\sigma_u^2}\boldsymbol{\varepsilon}'\mathbf{A}^{nn}\boldsymbol{\varepsilon}\right)\left(\sigma_\alpha^2\right)^{-m/2}\exp\left(-\frac{1}{2\sigma_\alpha^2}\sum_{j=1}^{m}\alpha_j^2\right)$$

$$\sigma_\alpha^{2\left(-\left(\frac{\nu_\alpha}{2}+1\right)\right)}e^{-\frac{\nu_\alpha s_\alpha^2}{2\sigma_\alpha^2}}\sigma_u^{2\left(-\left(\frac{\nu_u}{2}+1\right)\right)}e^{-\frac{\nu_u s_u^2}{2\sigma_u^2}}\sigma_e^{2\left(-\left(\frac{\nu_e}{2}+1\right)\right)}e^{-\frac{\nu_e s_e^2}{2\sigma_e^2}}\pi^{\alpha_0}\left(1-\pi\right)^{\beta_0}$$

for the MME for equation [4.6].

For the fixed effects $\boldsymbol{\beta}$, suppose the design matrix is $n\times p$, write

$$\mathbf{X}_{nxp}=\begin{bmatrix}x_{11}&x_{12}&x_{13}&\cdots&x_{1p}\\x_{21}&x_{22}&x_{23}&\cdots&x_{2p}\\x_{31}&x_{32}&x_{33}&\cdots&x_{3p}\\\vdots&\vdots&\vdots&\ddots&\vdots\\x_{n1}&x_{n2}&x_{n3}&\cdots&x_{np}\end{bmatrix}=\begin{bmatrix}\mathbf{x}_1'\\\mathbf{x}_2'\\\mathbf{x}_3'\\\vdots\\\mathbf{x}_n'\end{bmatrix}=\begin{bmatrix}\mathbf{x}_{.1}&\mathbf{x}_{.2}&\mathbf{x}_{.3}&&\mathbf{x}_{.p}\end{bmatrix}$$

Here $\mathbf{x}_{.j}$ is the covariate/dummy variable values for variable j of the fixed effects.

Similarly, for the marker effects $\boldsymbol{\alpha}$, the design matrix is $n\times m$, write

$$\mathbf{W}=\begin{bmatrix}w_{11}&w_{12}&w_{13}&\cdots&w_{1m}\\w_{21}&w_{22}&w_{23}&\cdots&w_{2m}\\w_{31}&w_{32}&w_{33}&\cdots&w_{3m}\\\vdots&\vdots&\vdots&\ddots&\vdots\\w_{n1}&w_{n2}&w_{n3}&\cdots&w_{nm}\end{bmatrix}=\begin{bmatrix}\mathbf{w}_1'\\\mathbf{w}_2'\\\mathbf{w}_3'\\\vdots\\\mathbf{w}_n'\end{bmatrix}=\begin{bmatrix}\mathbf{w}_{.1}&\mathbf{w}_{.2}&\mathbf{w}_{.3}&\cdots&\mathbf{w}_{.m}\end{bmatrix}$$

where $\mathbf{w}_{.j}$ is the covariate/dummy variable values for SNP genotype j of the random marker effects.

Finally, for the imputation residual **ε**, the design matrix is $n_n \times q_n$, write

$$\mathbf{Z}_n = \begin{bmatrix} z_{11} & z_{12} & z_{13} & \cdots & z_{1q_1} \\ z_{21} & z_{22} & z_{23} & \cdots & z_{2q_1} \\ z_{31} & z_{32} & z_{33} & \cdots & z_{3q_1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ z_{n_n 1} & z_{n_n 2} & z_{n_n 3} & \cdots & z_{n_n q_n} \end{bmatrix} = \begin{bmatrix} \mathbf{z}'_{(n)1} \\ \mathbf{z}'_{(n)2} \\ \mathbf{z}'_{(n)3} \\ \vdots \\ \mathbf{z}'_{(n)n_n} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_{(n).1} & \mathbf{z}_{(n).2} & \mathbf{z}_{(n).3} & \cdots & \mathbf{z}_{(n).q_n} \end{bmatrix}$$

where $\mathbf{z}_{(1).j}$ is the covariate/dummy variable values for ungenotyped animal *j* of the imputation residuals.

Full conditional densities (FCD) under MCMC-ssSSVS

FCD for Fixed Effects

$$p(\beta_j \mid ELSE) \sim N\left(\tilde{\beta}_j, \tilde{v}_{\beta j}\right)$$

with

$$\tilde{\beta}_j = \frac{\mathbf{x}'_{.j}\left(\left(\mathbf{y}\text{-}\mathbf{X\beta} - \mathbf{W\alpha} - \mathbf{U\varepsilon}\right) + \mathbf{x}_{.j}\beta_j\right)}{\mathbf{x}'_{.j}\mathbf{x}_{.j}}$$

$$= \frac{\mathbf{x}'_{.j}\left(\mathbf{e} + \mathbf{x}_{.j}\beta_j\right)}{\mathbf{x}'_{.j}\mathbf{x}_{.j}} = \frac{\left(\mathbf{x}'_{.j}\mathbf{e} + \mathbf{x}'_{.j}\mathbf{x}_{.j}\beta_j\right)}{\mathbf{x}'_{.j}\mathbf{x}_{.j}}$$

$$= \frac{\left(\sum_{i=1}^{n} x_{ij}e_i + \left(\sum_{i=1}^{n} x_{ij}^2\right)\beta_j\right)}{\left(\sum_{i=1}^{n} x_{ij}^2\right)}$$

and $\tilde{v}_{\beta j} = \sigma_e^2\left(\mathbf{x}'_{.j}\mathbf{x}_{.j}\right)^{-1} = \sigma_e^2\left(\sum_{i=1}^{n} x_{ij}^2\right)^{-1}$.

FCD for Marker-Specific Augmented (i.e. Indicator) variables ϕ_j

The the full conditional density is given by

$$p\left(\phi_j \mid ELSE\right) \propto p\left(\alpha_j \mid \sigma_\alpha^2, \phi_j\right) p\left(\phi_j \mid \pi\right)$$

$$\propto \frac{1}{\sqrt{2\pi\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c}+\phi_j\right)}} \exp\left(-\frac{1}{2}\frac{\alpha_j^2}{\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c}+\phi_j\right)}\right) \pi^{\phi_j}\left(1-\pi\right)^{1-\phi_j}$$

Such that it can be readily determined that:

$$\text{Prob}\left(\phi_j = 1 \mid \text{ELSE} \;\; \text{except} \;\; g_j\right) = \frac{h_1\pi}{h_0\left(1-\pi\right)+h_1\pi}$$

where

$$h_1 = p\left(\alpha_j \mid \sigma_\alpha^2, \phi_j = 1\right) \propto \frac{1}{\sqrt{\sigma_\alpha^2}} \exp\left(-\frac{1}{2}\frac{\alpha_j^2}{\sigma_\alpha^2}\right) \text{ and}$$

$$h_0 = p\left(\alpha_j \mid \sigma_\alpha^2, \phi_j = 0\right) \propto \frac{1}{\sqrt{\frac{\sigma_\alpha^2}{c}}} \exp\left(-\frac{1}{2}\frac{\alpha_j^2}{\frac{\sigma_\alpha^2}{c}}\right)$$

such that $\text{Prob}\left(\phi_j = 1 \mid \text{ELSE} \;\; \text{except} \;\; g_j\right) = \frac{h_1\pi}{h_0\left(1-\pi\right)+h_1\pi} = \frac{\pi}{\frac{h_0}{h_1}\left(1-\pi\right)+\pi}$ .In fact, I can use the `dnorm` function in R to compute h_0 and h_1 directly. Then ϕ_j can be drawn from a Bernoulli distribution.

FCD for Marker Effects

Then the posterior distribution for marker effects are given as

$$p\left(\alpha_j \mid \mathbf{ELSE}\right) \propto \left(\prod_{i=1}^{n} p\left(y_i|\boldsymbol{\beta},\boldsymbol{\alpha}\right)\right) p\left(\alpha_j \mid \sigma_\alpha^2, \phi_j\right) \sim N\left(\tilde{\alpha}_j, \tilde{v}_{\alpha j}\right);\ j = 1, 2, \ldots, m$$

where

$$\tilde{\alpha}_j = \frac{\mathbf{w}_{.j}^{'}\left(\left(\mathbf{y}\text{-}\mathbf{X}\boldsymbol{\beta} - \mathbf{W}\boldsymbol{\alpha}\text{-}\mathbf{U}\boldsymbol{\varepsilon}\right) + \mathbf{w}_{.j}\alpha_j\right)}{\mathbf{w}_{.j}^{'}\mathbf{w}_{.j} + \sigma_e^2\left(\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c} + \phi_j\right)\right)^{-1}}$$

$$= \frac{\mathbf{w}_{.j}^{'}\left(\mathbf{e} + \mathbf{w}_{.j}^{*}\alpha_j\right)}{\mathbf{w}_{.j}^{'}\mathbf{w}_{.j} + \sigma_e^2\left(\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c} + \phi_j\right)\right)^{-1}} = \frac{\mathbf{w}_{.j}^{'}\mathbf{e} + \mathbf{w}_{.j}^{'}\mathbf{w}_{.j}\alpha_j}{\mathbf{w}_{.j}^{'}\mathbf{w}_{.j} + \sigma_e^2\left(\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c} + \phi_j\right)\right)^{-1}}$$

$$= \frac{\sum_{i=1}^{n} w_{ij}e_i + \left(\sum_{i=1}^{n} w_{ij}^2\right)\alpha_j}{\left(\sum_{i=1}^{n} w_{ij}^2 + \sigma_e^2\left(\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c} + \phi_j\right)\right)^{-1}\right)}$$

and $\tilde{v}_{\alpha j} = \left(\frac{\sum_{i=1}^{n}\left(w_{ij}\right)^2}{\sigma_e^2} + \left(\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c} + \phi_j\right)\right)^{-1}\right)^{-1}$

*FCD for Imputation Residuals: **ε***

Note that the **ε** terms only cross-reference to the non-genotyped animals. That's because

$$\mathbf{U}\boldsymbol{\varepsilon} = \begin{bmatrix} \mathbf{Z}_n \\ \mathbf{0} \end{bmatrix}\boldsymbol{\varepsilon} = \begin{bmatrix} \mathbf{Z}_n\boldsymbol{\varepsilon} \\ \mathbf{0} \end{bmatrix}$$

Hence, the vector of residuals could be broken down into two components:

$$\begin{bmatrix} \mathbf{e}_n \\ \mathbf{e}_g \end{bmatrix} = \begin{bmatrix} \mathbf{y}_n\text{-}\mathbf{X}_n\boldsymbol{\beta} - \mathbf{W}_n\boldsymbol{\alpha} - \mathbf{Z}_n\boldsymbol{\varepsilon} \\ \mathbf{y}_g\text{-}\mathbf{X}_g\boldsymbol{\beta} - \mathbf{W}_g\boldsymbol{\alpha} \end{bmatrix}$$

Also, let's write the rows of $\mathbf{A}^{nn}$ as follows:

$$\mathbf{A}^{nn} = \begin{bmatrix} \mathbf{a}_1^{nn\prime} \\ \mathbf{a}_2^{nn\prime} \\ \mathbf{a}_3^{nn\prime} \\ \\ \mathbf{a}_{q_1}^{nn\prime} \end{bmatrix}$$

Then $p(\varepsilon_k \mid ELSE) \sim N\left(\tilde{\varepsilon}_k, \tilde{v}_{\varepsilon k}\right); k = 1, 2, \ldots, q_1$ where

$$\tilde{\varepsilon}_k = \frac{\mathbf{z}'_{(n).k}\left(\left(\mathbf{y}_n - \mathbf{X}_n\boldsymbol{\beta} - \mathbf{W}_n\boldsymbol{\alpha} - \mathbf{Z}_n\boldsymbol{\varepsilon}\right) + \mathbf{z}_{(n).k}\varepsilon_k\right) - \lambda_u \mathbf{a}_k^{nn\prime}\boldsymbol{\varepsilon} + \lambda_u a_{kk}^{nn}\varepsilon_k}{\mathbf{z}'_{(n).k}\mathbf{z}_{(n).k} + a_{kk}^{nn}\lambda_u}$$

$$= \frac{\mathbf{z}'_{(n).k}\left(\mathbf{e}_n + \mathbf{z}_{(n).k}\varepsilon_k\right) - \lambda_u \mathbf{a}_k^{nn\prime}\boldsymbol{\varepsilon} + \lambda_u a_{kk}^{nn}\varepsilon_k}{\mathbf{z}'_{(n).k}\mathbf{z}_{(n).k} + a_{kk}^{nn}\lambda_u} = \frac{\mathbf{z}'_{(n).k}\mathbf{e}_n + \mathbf{z}'_{(n).k}\mathbf{z}_{(n).k}\varepsilon_k - \lambda_u \mathbf{a}_k^{nn\prime}\boldsymbol{\varepsilon} + \lambda_u a_{kk}^{nn}\varepsilon_k}{\mathbf{z}'_{(n).k}\mathbf{z}_{(n).k} + a_{kk}^{nn}\lambda_u}$$

$$= \frac{\sum_{i=1}^{n_1} z_{(n)ij}e_i + \left(\sum_{i=1}^{n_1} z_{(n)ij}^2\right)\varepsilon_k - \lambda_u \mathbf{a}_k^{nn\prime}\boldsymbol{\varepsilon} + \lambda_u a_{kk}^{nn}\varepsilon_k}{\left(\sum_{i=1}^{n_1} z_{(n)ij}^2 + a_{kk}^{nn}\lambda_u\right)}$$

with $\lambda_u = \dfrac{\sigma_e^2}{\sigma_u^2}$ and

$$\tilde{v}_{\varepsilon k} = \left(\frac{\sum_{i=1}^{n_1} z_{(n)ij}^2}{\sigma_e^2} + a_{kk}^{nn}\sigma_u^{-2}\right)^{-1}$$

Note that a_{kk}^{nn} is element *k,k* of $\mathbf{A}^{nn}$.

FCD for Residual Variance σ_e^2

$$p\left(\sigma_e^2 \mid ELSE\right) \propto \left(2\pi\sigma_e^2\right)^{-n/2} \exp\left(-\frac{1}{2\sigma_e^2}\sum_{j=1}^{n} e_j^2\right)\sigma_e^{2\,-\left(\frac{\nu_e}{2}+1\right)} e^{-\frac{\nu_e s_e^2}{2\sigma_e^2}}$$

$$\propto \sigma_e^{2\,-\left(\frac{\nu_e+n}{2}+1\right)} \exp\left(-\frac{1}{2\sigma_e^2}\left(\sum_{j=1}^{n} e_j^2 + \nu_e s_e^2\right)\right)$$

where

$$e_j = y_j - \mathbf{z}_j'\boldsymbol{\beta} - \mathbf{w}_j'\boldsymbol{\alpha} - \mathbf{u}_j'\boldsymbol{\varepsilon}$$

i.e., it is $\chi^{-2}\left(\nu_e + n, \sum_{j=1}^{n} e_j^2 + \nu_e s_e^2\right)$

FCD for marker variance

$$p\left(\sigma_\alpha^2 \mid ELSE\right) \propto \left(\prod_{j=1}^{m} p\left(g_j \mid \sigma_\alpha^2, \phi_j\right)\right) p\left(\sigma_\alpha^2 \mid \nu_\alpha, \nu_\alpha s_\alpha^2\right)$$

$$\propto \prod_{j=1}^{m}\left(\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c}+\phi_j\right)\right)^{-1/2} \exp\left(-\frac{1}{2}\sum_{j=1}^{m}\frac{\alpha_j^2}{\sigma_\alpha^2\left(\frac{\left(1-\phi_j\right)}{c}+\phi_j\right)}\right)\sigma_\alpha^{2\,-\left(\frac{\nu_\alpha}{2}+1\right)} e^{-\frac{\nu_\alpha s_\alpha^2}{2\sigma_\alpha^2}}$$

$$\propto \left(\sigma_\alpha^2\right)^{-m/2} \exp\left(-\frac{1}{2\sigma_\alpha^2}\sum_{j=1}^{m}\frac{\alpha_j^2}{\left(\frac{\left(1-\phi_j\right)}{c}+\phi_j\right)}\right)\sigma_\alpha^{2\,-\left(\frac{\nu_\alpha}{2}+1\right)} e^{-\frac{\nu_\alpha s_\alpha^2}{2\sigma_\alpha^2}}$$

$$\propto \sigma_\alpha^{2\,-\left(\frac{\nu_\alpha+m}{2}+1\right)} \exp\left(-\frac{1}{2\sigma_\alpha^2}\left(\sum_{j=1}^{m}\frac{\alpha_j^2}{\left(\frac{\left(1-\phi_j\right)}{c}+\phi_j\right)}+\nu_\alpha s_\alpha^2\right)\right)$$

The FCD is $\chi^{-2}\left(\nu_\alpha + m, \nu_\alpha s_\alpha^2 + \sum_{j=1}^{m}\frac{\alpha_j^2}{\left(\frac{\left(1-\phi_j\right)}{c}+\phi_j\right)}\right)$

FCD for Polygenic Variance σ_u^2

$$p\left(\sigma_u^2 \mid ELSE\right) \propto \left(\sigma_u^2\right)^{-q_1/2} \exp\left(-\frac{1}{2\sigma_u^2}\boldsymbol{\varepsilon}'\mathbf{A}^{nn}\boldsymbol{\varepsilon}\right) p(\sigma_u^2 \mid \nu_u, s_u^2)$$

$$= \left(2\pi\sigma_u^2\right)^{-q_1/2} \exp\left(-\frac{1}{2\sigma_u^2}\boldsymbol{\varepsilon}'\mathbf{A}^{nn}\boldsymbol{\varepsilon}\right) \frac{\left(\frac{\nu_u s_u^2}{2}\right)^{\frac{\nu_u}{2}}}{\Gamma\left(\frac{\nu_u}{2}\right)} \sigma_u^{2^{-\left(\frac{\nu_u}{2}+1\right)}} e^{-\frac{\nu_u s_u^2}{2\sigma_u^2}}$$

$$\propto \left(\sigma_u^2\right)^{-\left(\frac{q_1+\nu_u}{2}+1\right)} \exp\left(-\frac{1}{2\sigma_u^2}\left(\boldsymbol{\varepsilon}'\mathbf{A}^{nn}\boldsymbol{\varepsilon} + \nu_u s_u^2\right)\right)$$

Hence, the FCD of σ_u^2 is a scaled inverse chi-square with degree of freedom of $q_1 + \nu_u$ and scale of $\boldsymbol{\varepsilon}'\mathbf{A}^{nn}\boldsymbol{\varepsilon} + \nu_u s_u^2$, i.e. $\chi^{-2}\left(q_1 + \nu_u, \boldsymbol{\varepsilon}'\mathbf{A}^{nn}\boldsymbol{\varepsilon} + \nu_u s_u^2\right)$

FCD for π

The posterior of π is given by,

$$p\left(\pi \mid ELSE\right) \propto \left(\prod_{j=1}^{m} p(\phi_j \mid \pi)\right) \pi^{\alpha_0} \left(1-\pi\right)^{\beta_0} \propto \pi^{m_1+\alpha_0-1} \left(1-\pi\right)^{m-m_1+\beta_0-1}$$

where $m_1 = \sum_{j=1}^{m} \phi_j$ denotes the number of "large variance" genetic effects as determined in the current cycle.

Figures

Figures B1-B5: Supplementary Manhattan plot figures for within-station splits genotyped and masked genotyped animals for within-station partitions P1 (Figure B1), P2 (Figure B2), P3 (Figure B3), P4 (Figure B4), and P5 (Figure B5) for milk fat. Panel A: Plot of –log10(P-value) versus genomic region for single SNP associations using EMMAX without using phenotypes on non-genotyped animals; Panel B: Plot of –log10(P-value) versus genomic region for genomic window associations using EMMAX without using phenotypes on non-genotyped animals; Panel C: Plot of posterior probabilities versus genomic region for genomic window associations using SSVS without using phenotypes on non-genotyped animals; Panel D: Plot of posterior probabilities versus genomic region for genomic window associations using SSVS without using phenotypes on non-genotyped animals; Panel E: Plot of –log10(P-value) versus genomic region for single SNP associations using ssEMMAX using phenotypes on non-genotyped animals; Panel F: Plot of –log10(P-value) versus genomic region for genomic window associations using ssEMMAX using phenotypes on non-genotyped animals; Panel G: Plot of posterior probabilities versus genomic region for genomic window associations using ssSSVS using phenotypes on non-genotyped animals; Panel H: Plot of posterior probabilities versus genomic region for genomic window associations using ssSSVS using phenotypes on non-genotyped animals.

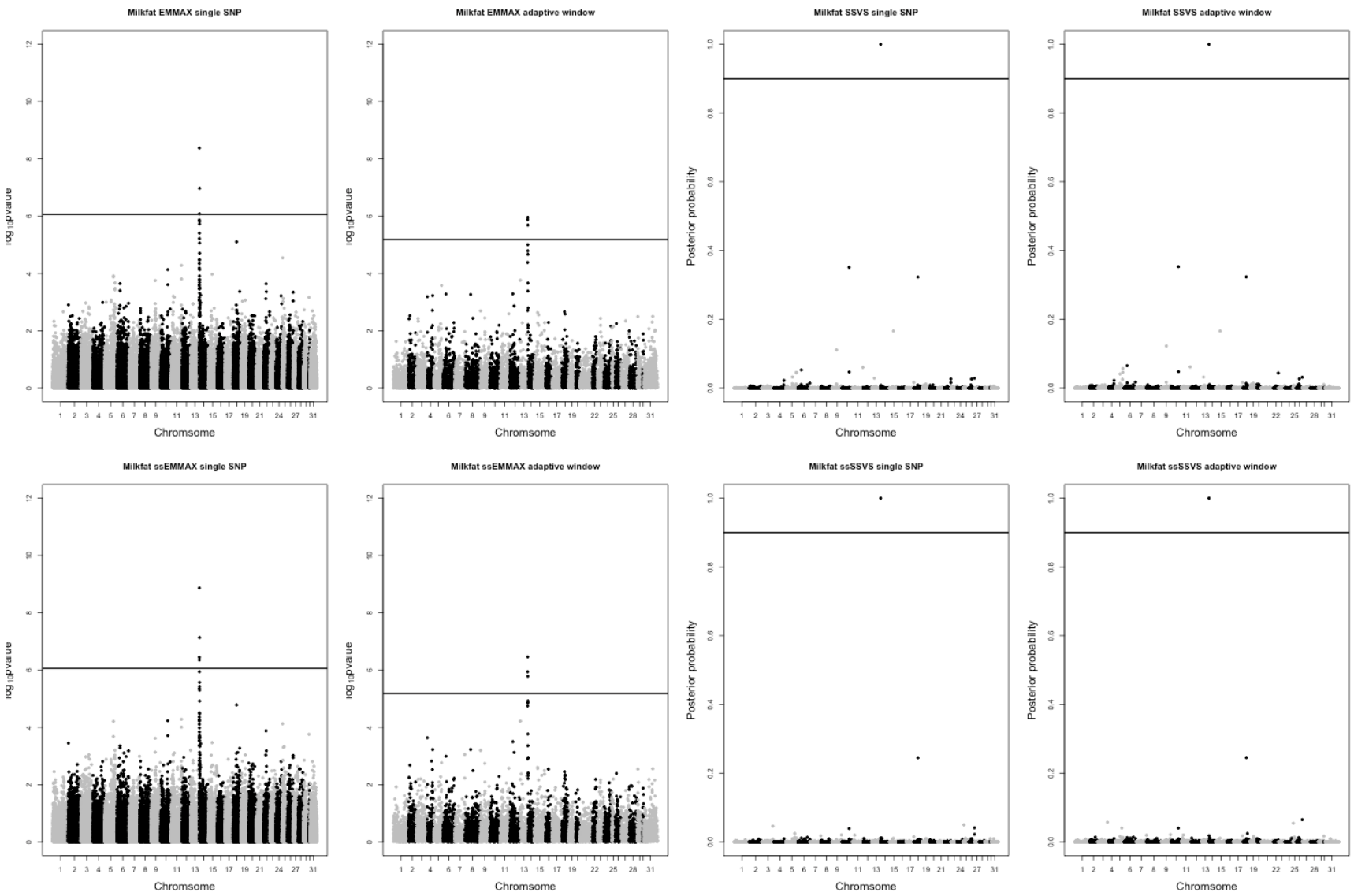

Figure B.1 Partition P1: Manhattan plot for milkfat in within station splits of genotyped and non-genotyped animals.

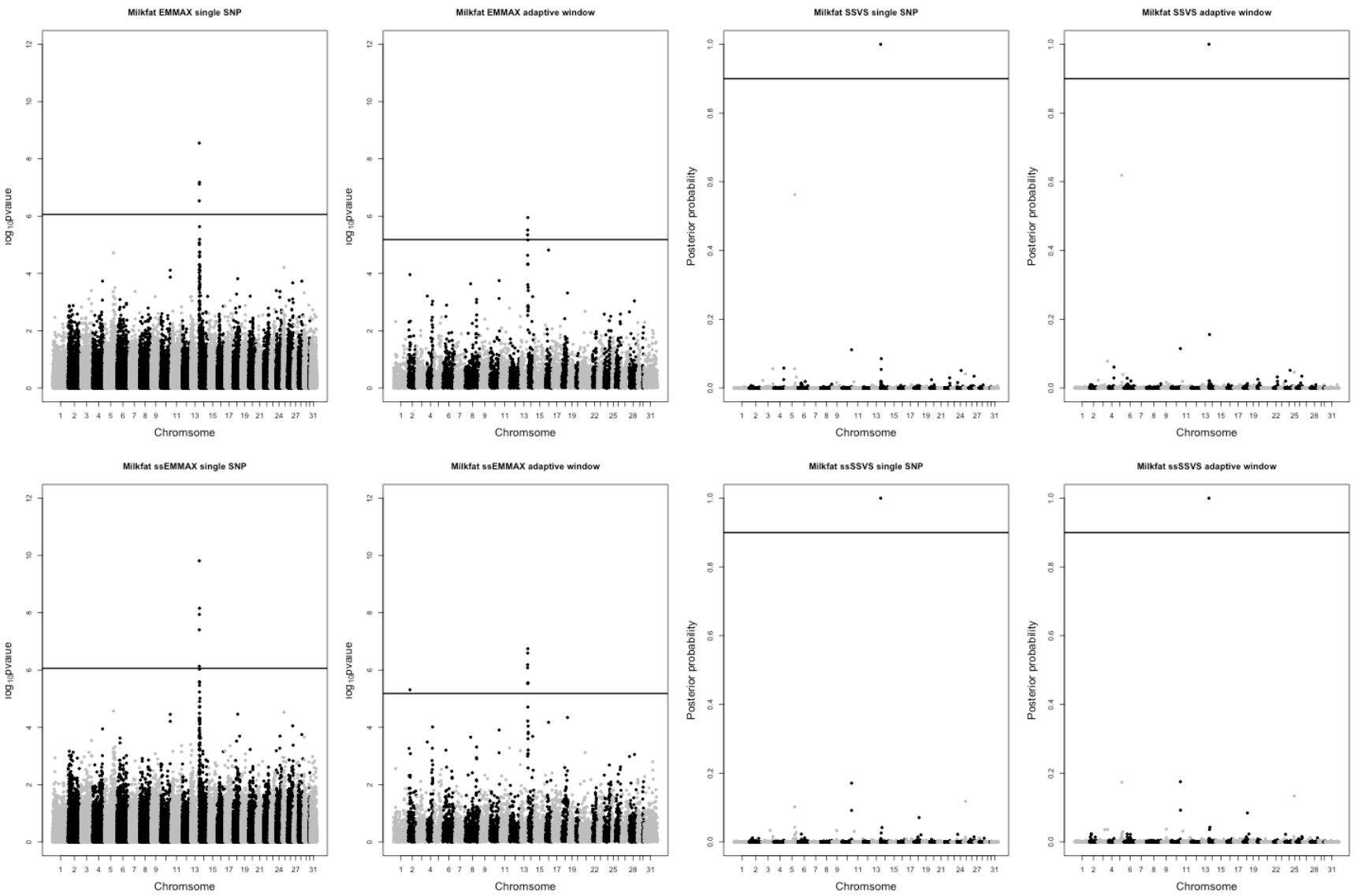

Figure B.2 Partition P2: Manhattan plot for milkfat in within station splits of genotyped and non-genotyped animals.

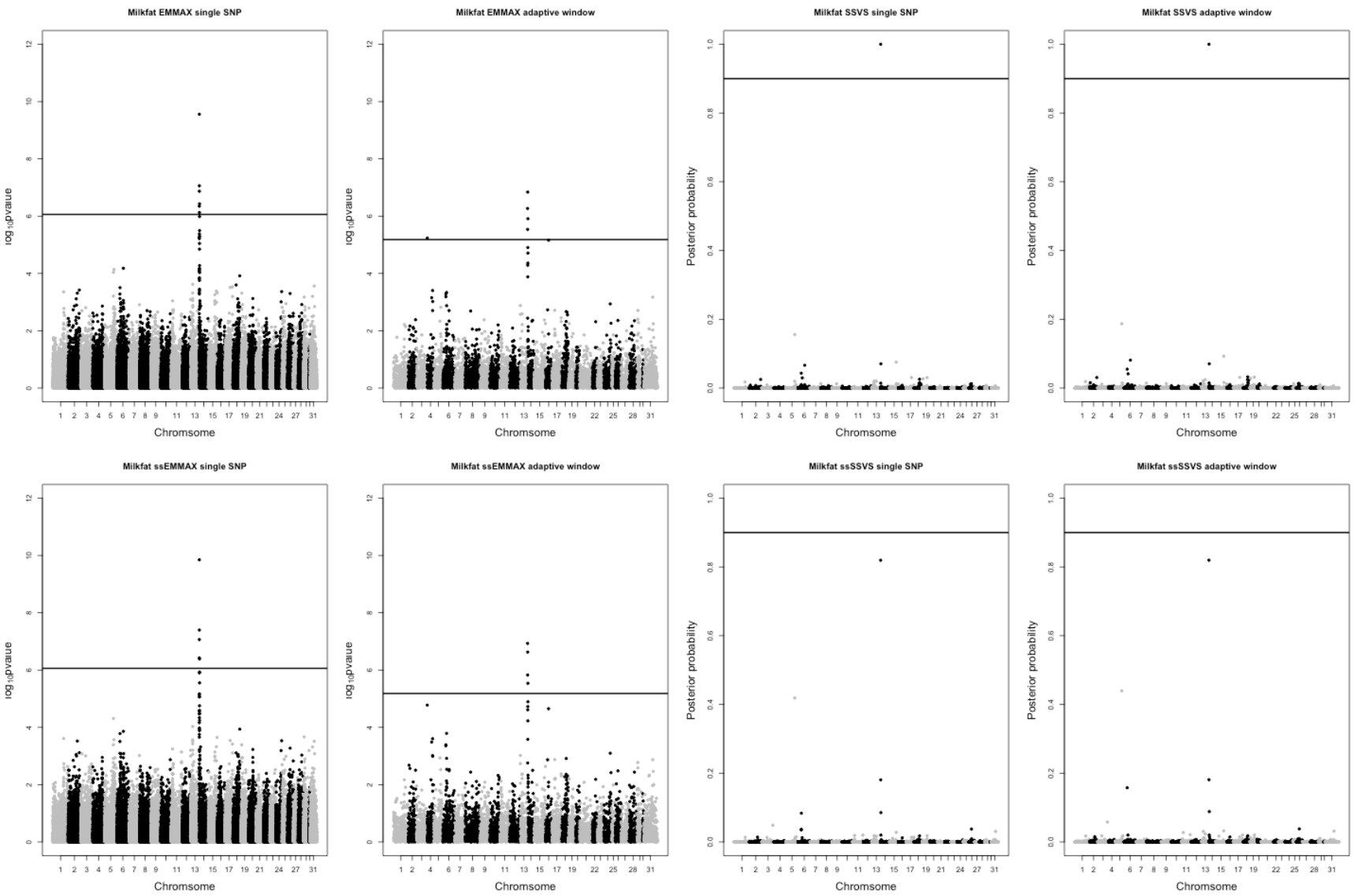

Figure B.3 Partition P3: Manhattan plot for milkfat in within station splits of genotyped and non-genotyped animals.

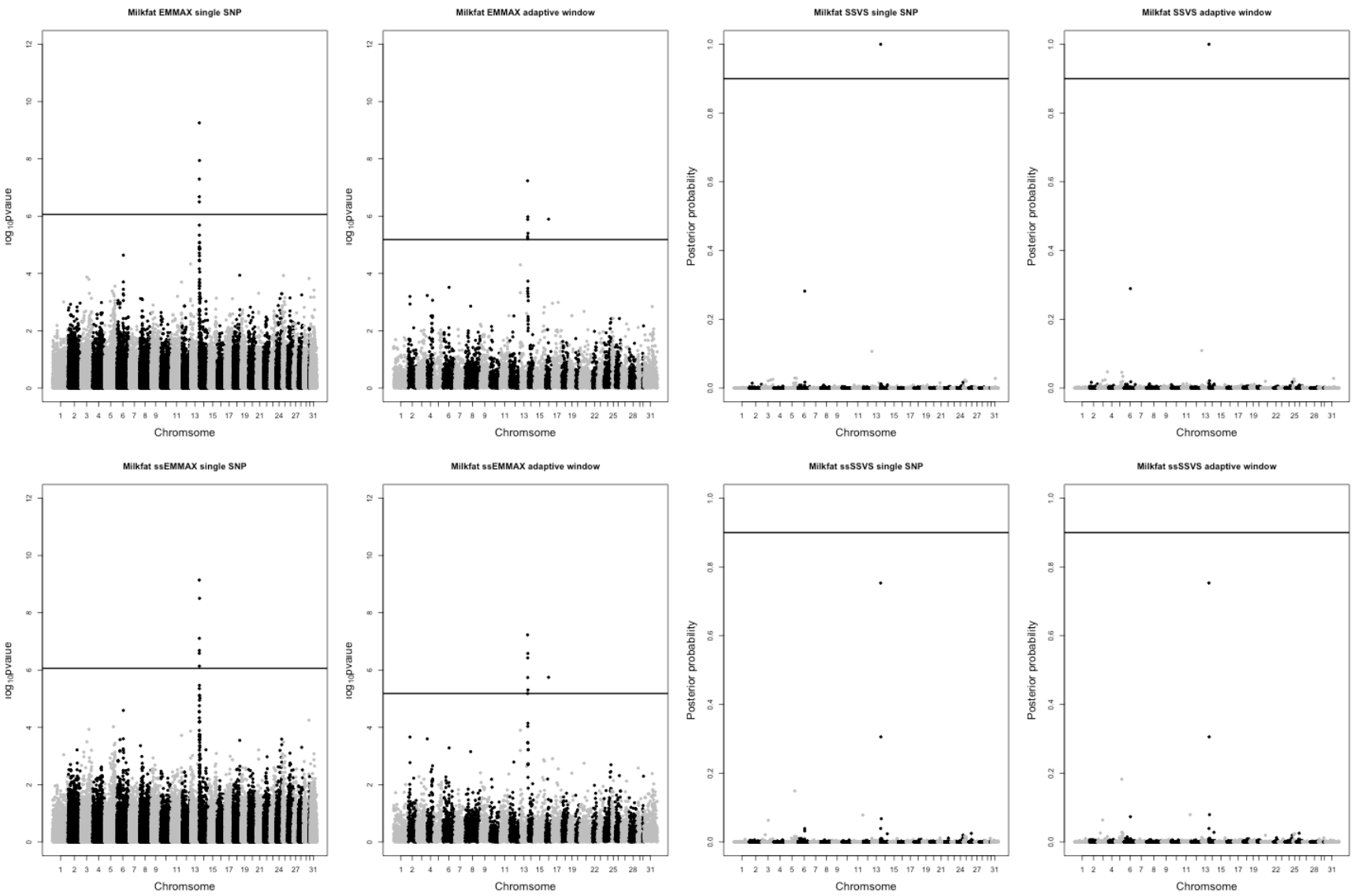

Figure B.4 Partition P4: Manhattan plot for milkfat in within station splits of genotyped and non-genotyped animals.

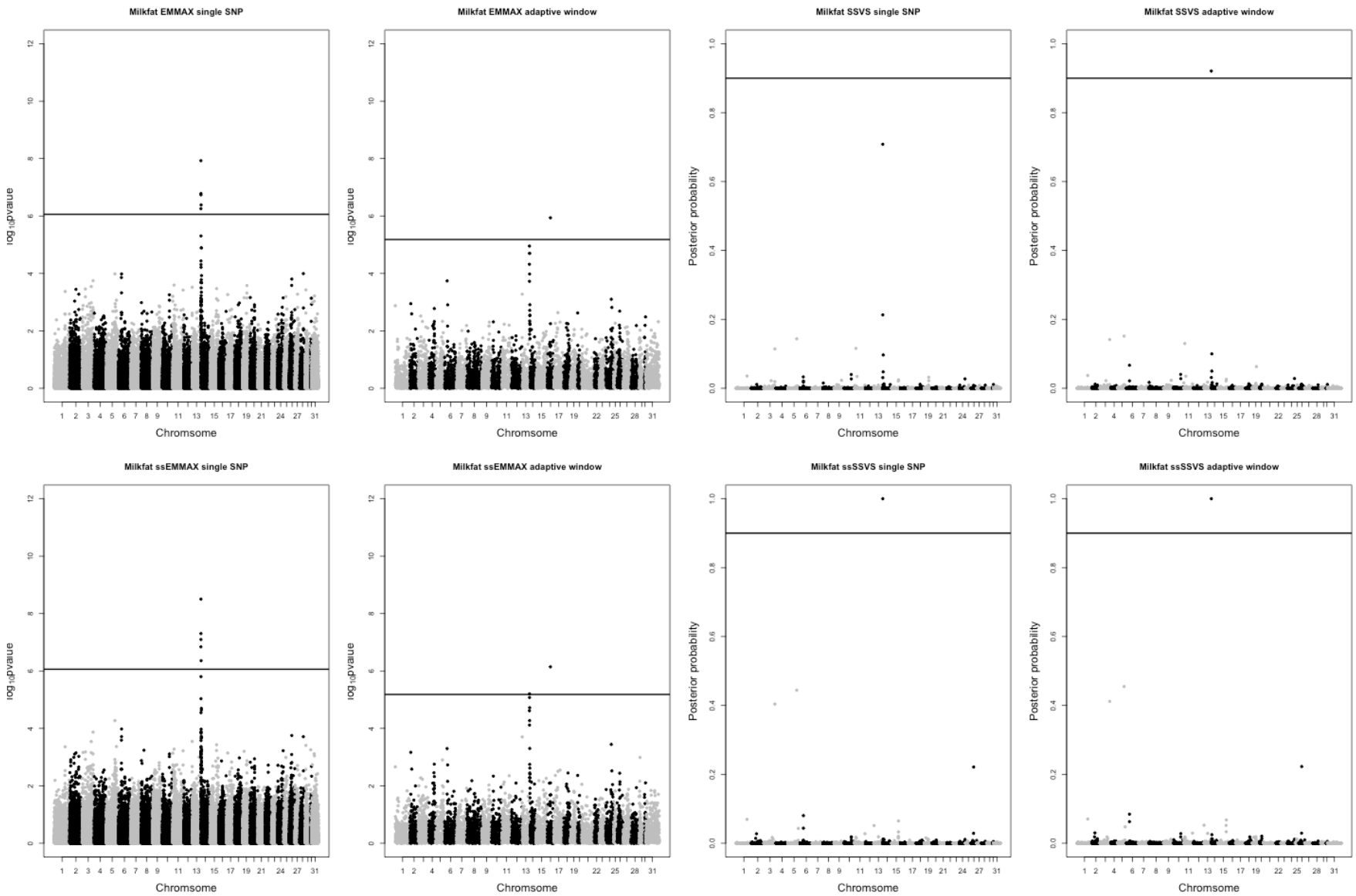

Figure B.5 Partition P5: Manhattan plot for milkfat in within station splits of genotyped and non-genotyped animals.

Figures B6-B11: Supplementary Manhattan plot figures for acroos-station splits for genotyped and masked genotyped animals with genotype masking on cows from ISU (Figure B6), MSU (Figure B7), USDFRC (Figure B8), UW (Figure B9), and FL (Figure B10) and AGIL (Figure B11) for milk fat. Panel A: Plot of –log10(P-value) versus genomic region for single SNP associations using EMMAX without using phenotypes on non-genotyped animals; Panel B: Plot of –log10(P-value) versus genomic region for genomic window associations using EMMAX without using phenotypes on non-genotyped animals; Panel C: Plot of posterior probabilities versus genomic region for genomic window associations using SSVS without using phenotypes on non-genotyped animals; Panel D: Plot of posterior probabilities versus genomic region for genomic window associations using SSVS without using phenotypes on non-genotyped animals; Panel E: Plot of –log10(P-value) versus genomic region for single SNP associations using ssEMMAX using phenotypes on non-genotyped animals; Panel F: Plot of –log10(P-value) versus genomic region for genomic window associations using ssEMMAX using phenotypes on non-genotyped animals; Panel G: Plot of posterior probabilities versus genomic region for genomic window associations using ssSSVS using phenotypes on non-genotyped animals; Panel H: Plot of posterior probabilities versus genomic region for genomic window associations using ssSSVS using phenotypes on non-genotyped animals.

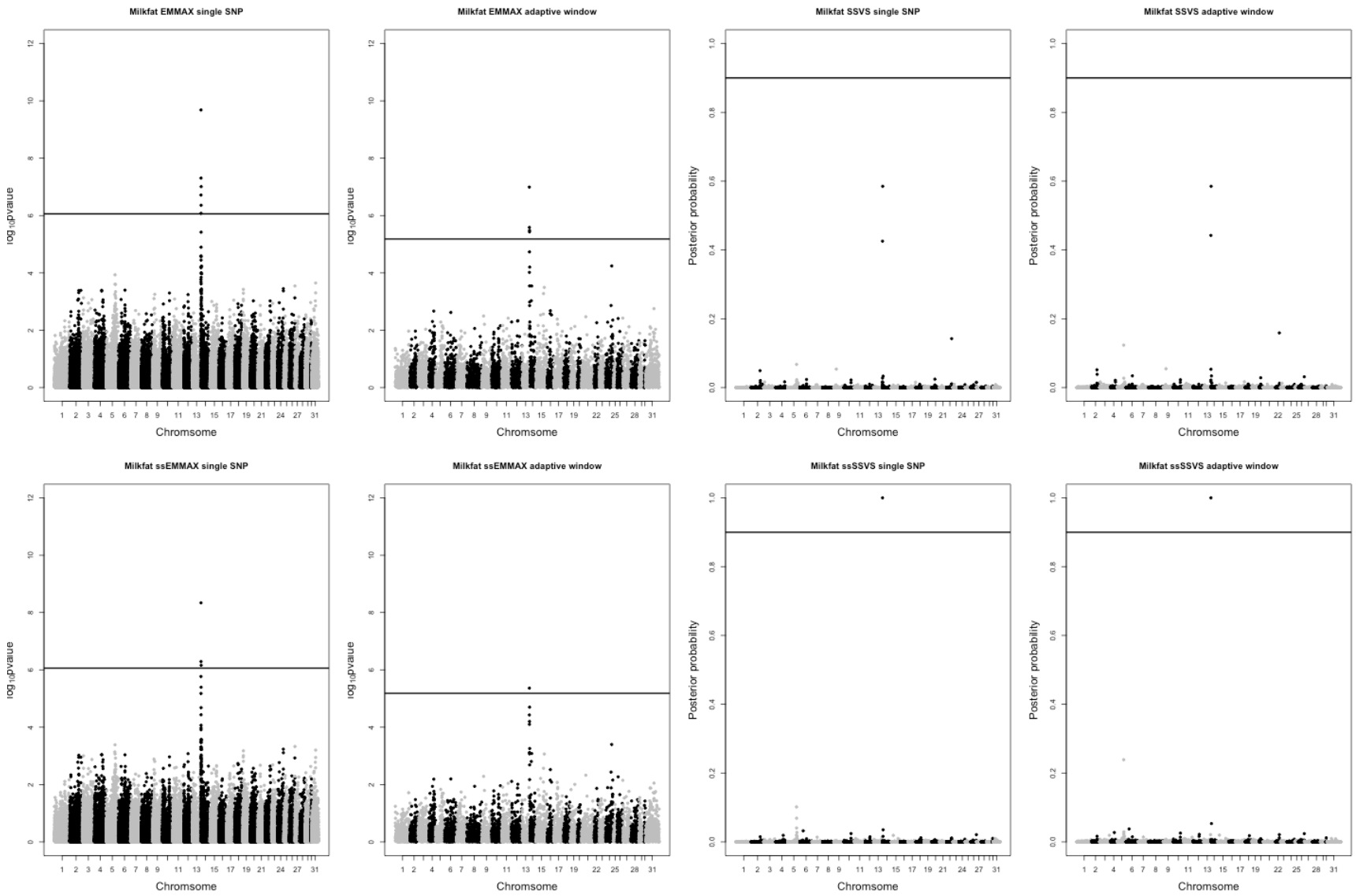

Figure B.6 Without the genotype of ISU: Manhattan plot for milkfat in across station study.

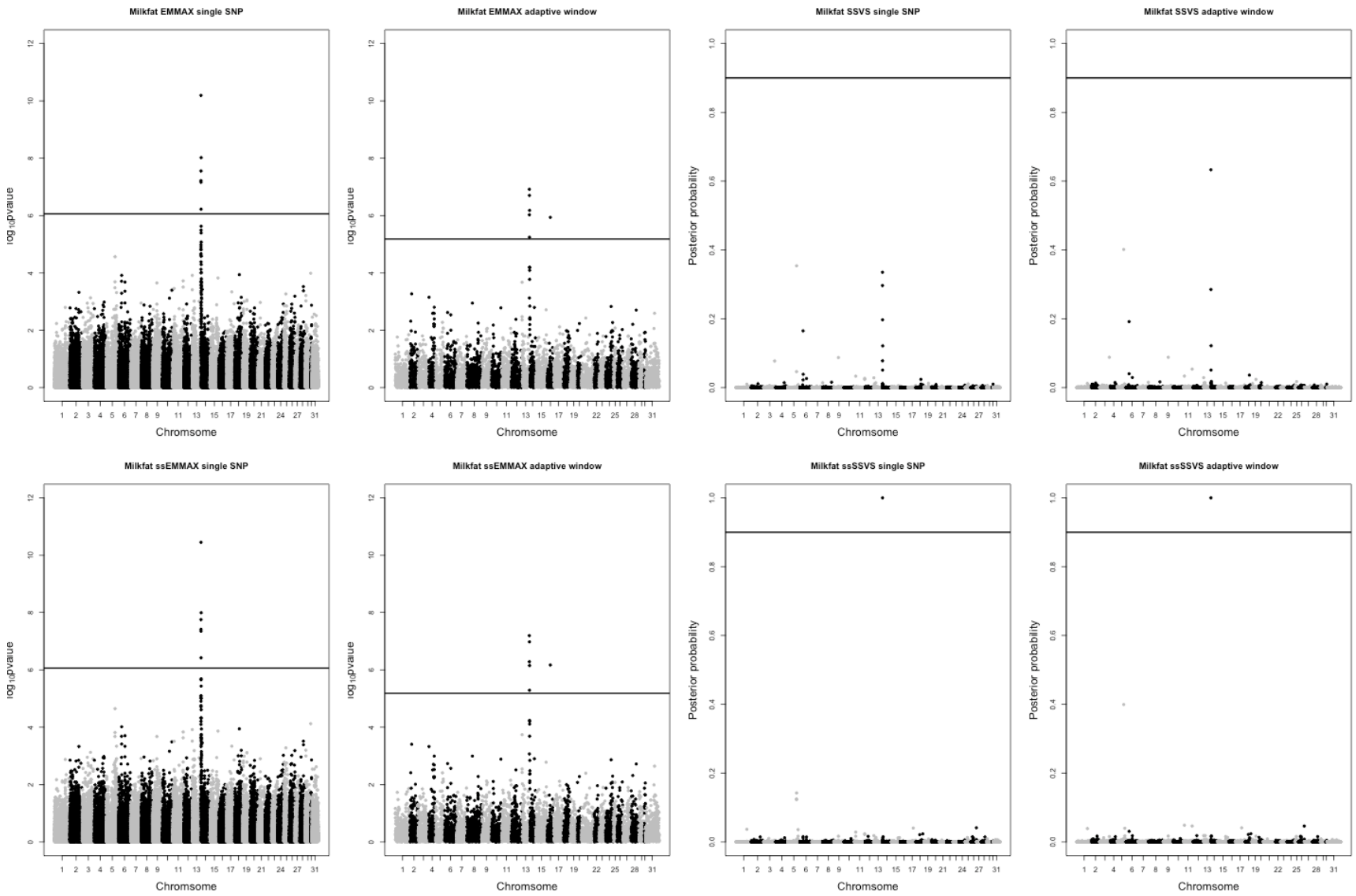

Figure B.7 Without the genotype of MSU: Manhattan plot for milkfat in across station study.

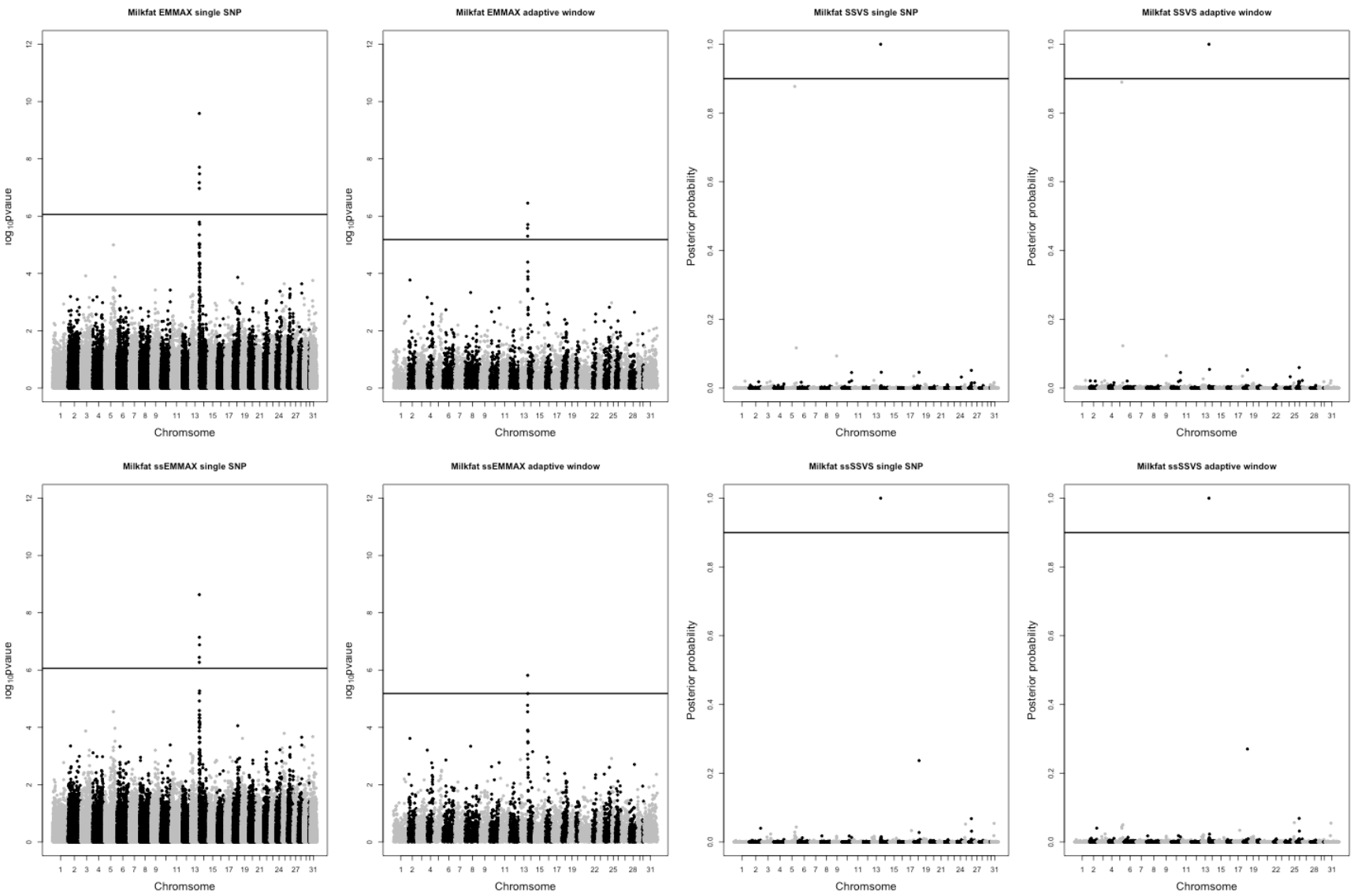

Figure B.8 Without the genotype of USDFRC: Manhattan plot for milkfat in across station study.

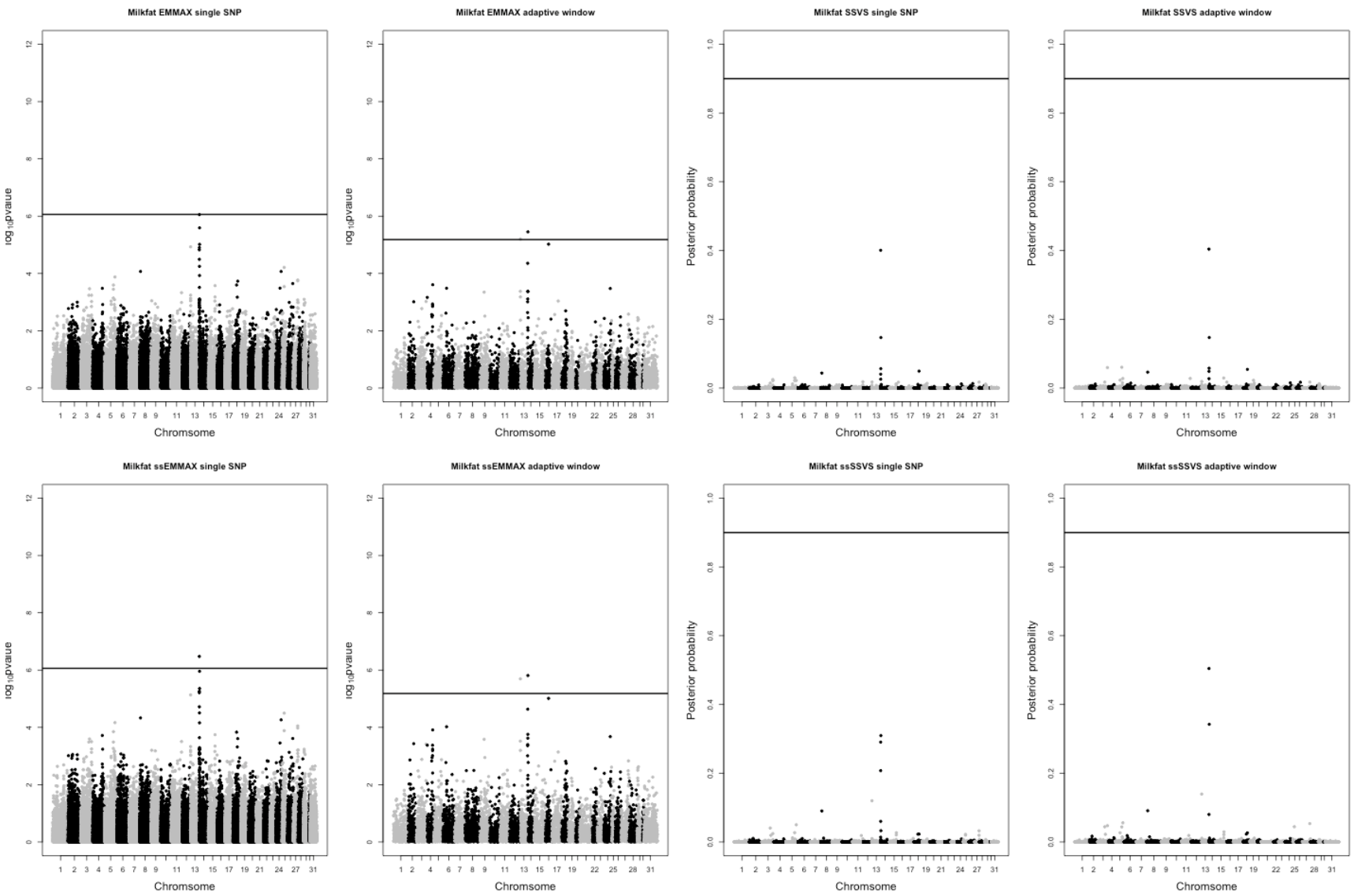

Figure B.9 Without the genotype of UW: Manhattan plot for milkfat in across station study.

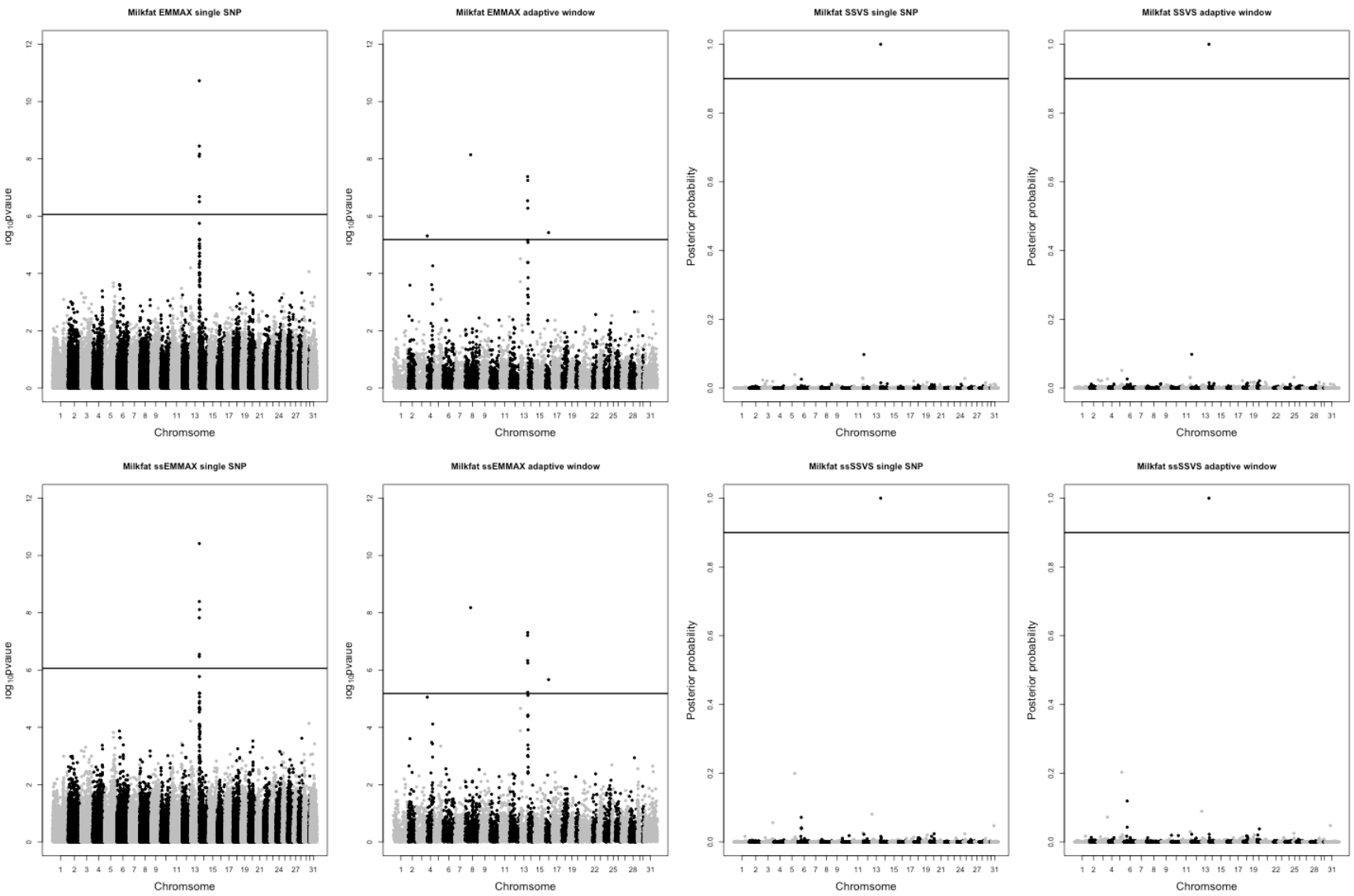

Figure B.10 Without the genotype of FL: Manhattan plot for milkfat in across station study.

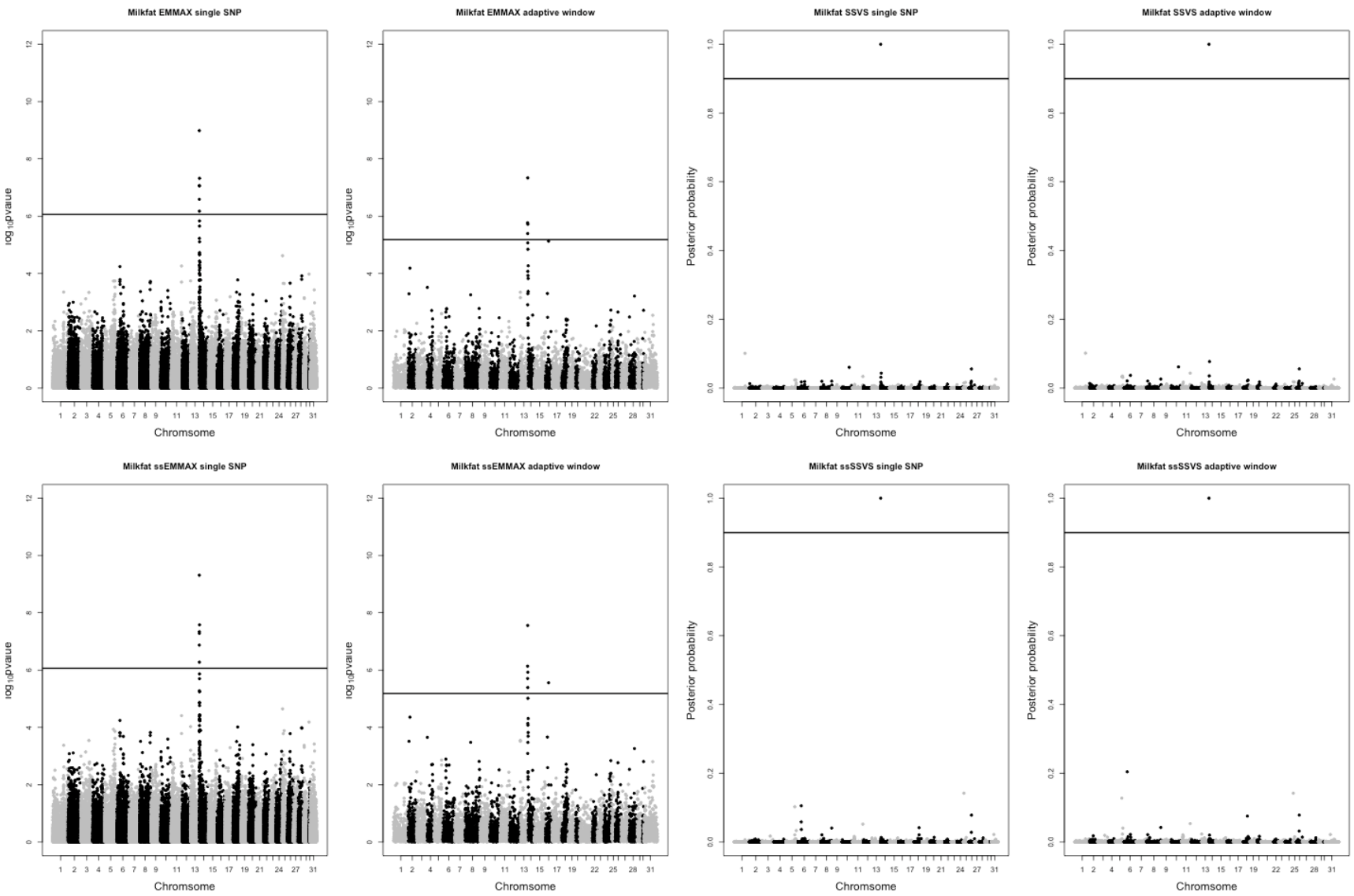

Figure B.11 Without the genotype of AGIL: Manhattan plot for milkfat in across station study.

Figures B12-B16: Supplementary Manhattan plot figures for within-station splits genotyped and masked genotyped animals for within-station partitions P1 (Figure B12), P2 (Figure B13), P3 (Figure B14), P4 (Figure B15), and P5 (Figure B16) for body weight. Panel A: Plot of –log10(P-value) versus genomic region for single SNP associations using EMMAX without using phenotypes on non-genotyped animals; Panel B: Plot of –log10(P-value) versus genomic region for genomic window associations using EMMAX without using phenotypes on non-genotyped animals; Panel C: Plot of posterior probabilities versus genomic region for genomic window associations using SSVS without using phenotypes on non-genotyped animals; Panel D: Plot of posterior probabilities versus genomic region for genomic window associations using SSVS without using phenotypes on non-genotyped animals; Panel E: Plot of –log10(P-value) versus genomic region for single SNP associations using ssEMMAX using phenotypes on non-genotyped animals; Panel F: Plot of –log10(P-value) versus genomic region for genomic window associations using ssEMMAX using phenotypes on non-genotyped animals; Panel G: Plot of posterior probabilities versus genomic region for genomic window associations using ssSSVS using phenotypes on non-genotyped animals; Panel H: Plot of posterior probabilities versus genomic region for genomic window associations using ssSSVS using phenotypes on non-genotyped animals.

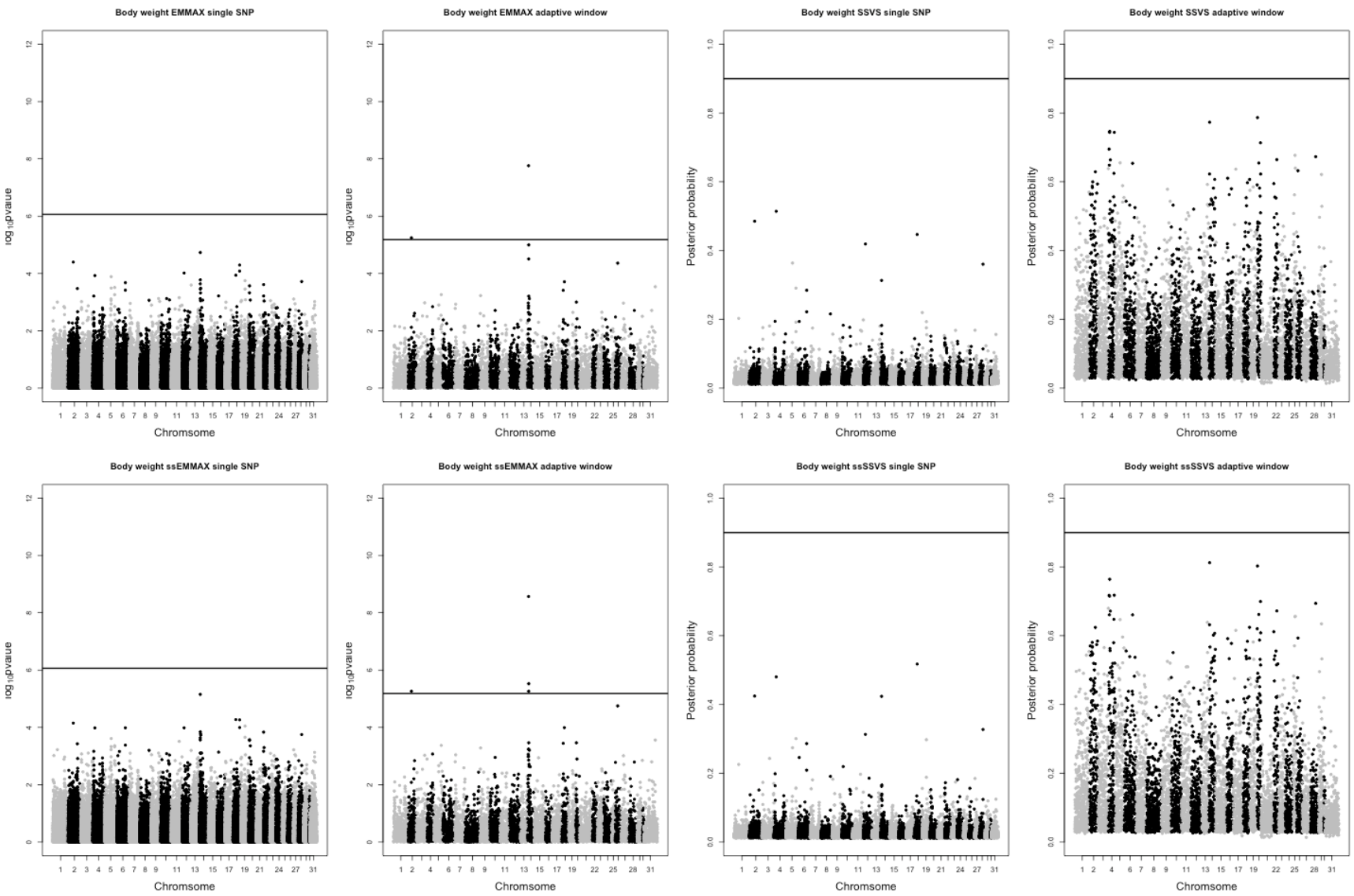

Figure B.12 Partition 1: Manhattan plot for body weight in within station study.

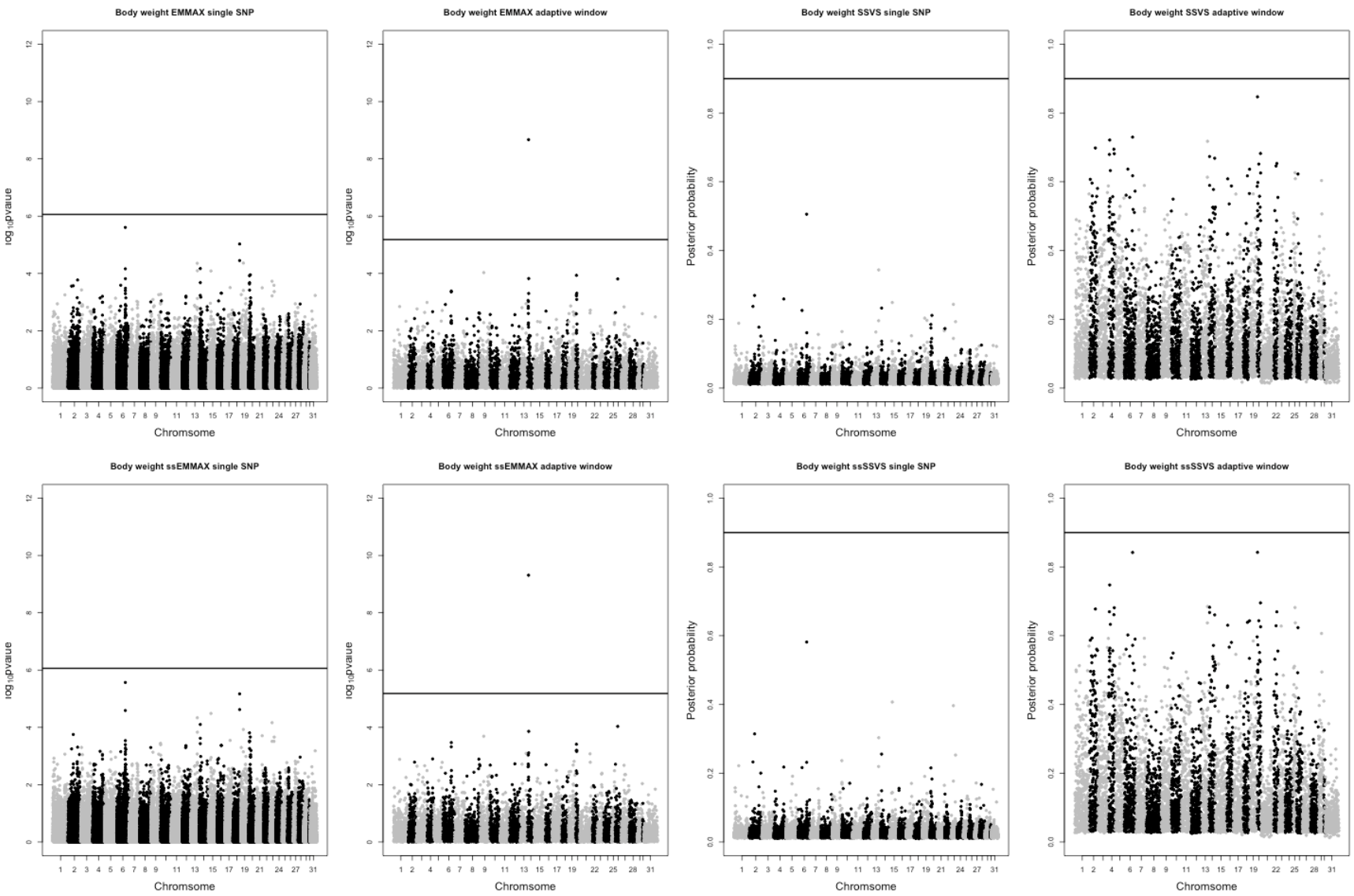

Figure B.13 Partition 2: Manhattan plot for body weight in within station study.

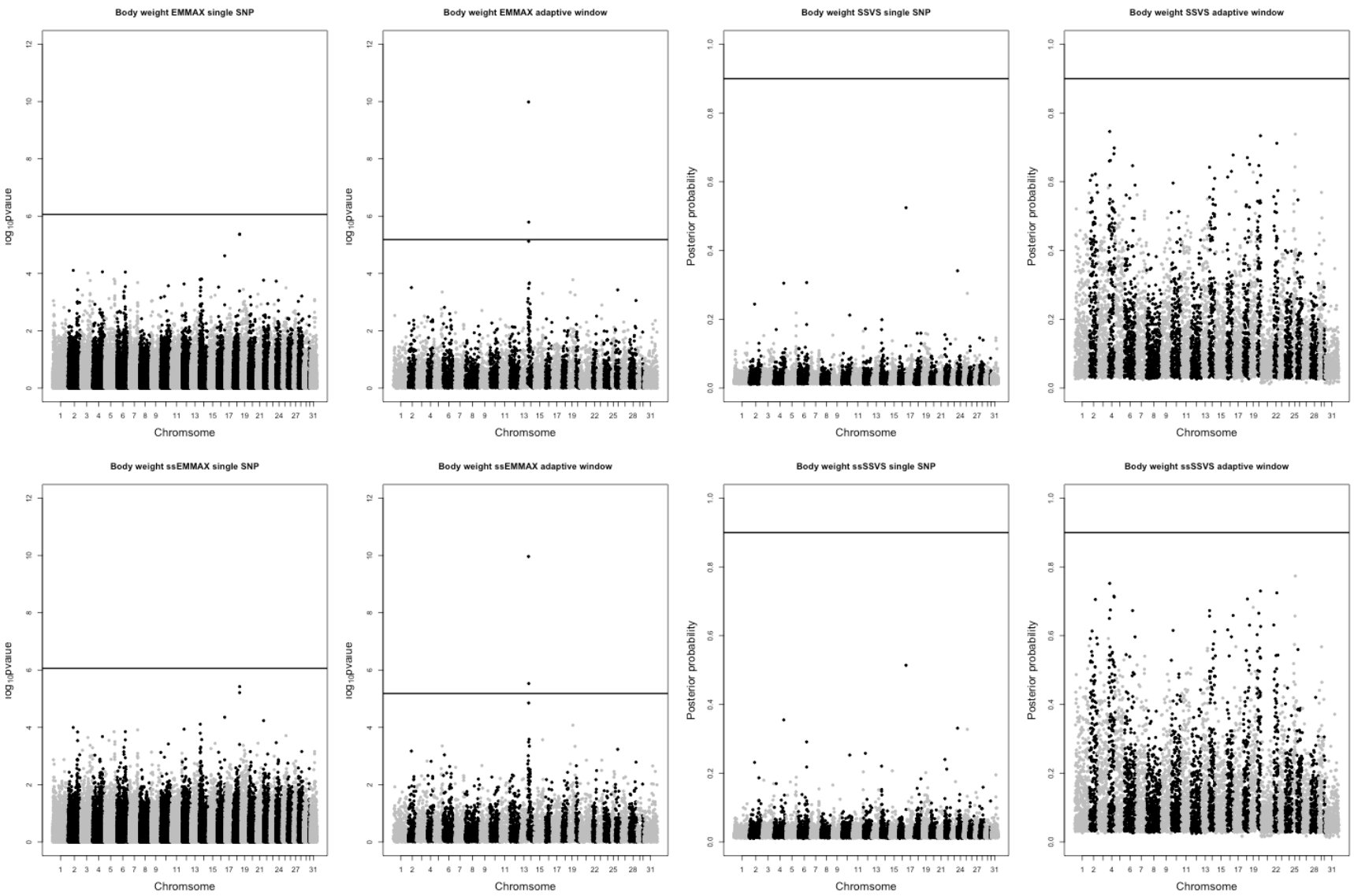

Figure B.14 Partition 3: Manhattan plot for body weight in within station study.

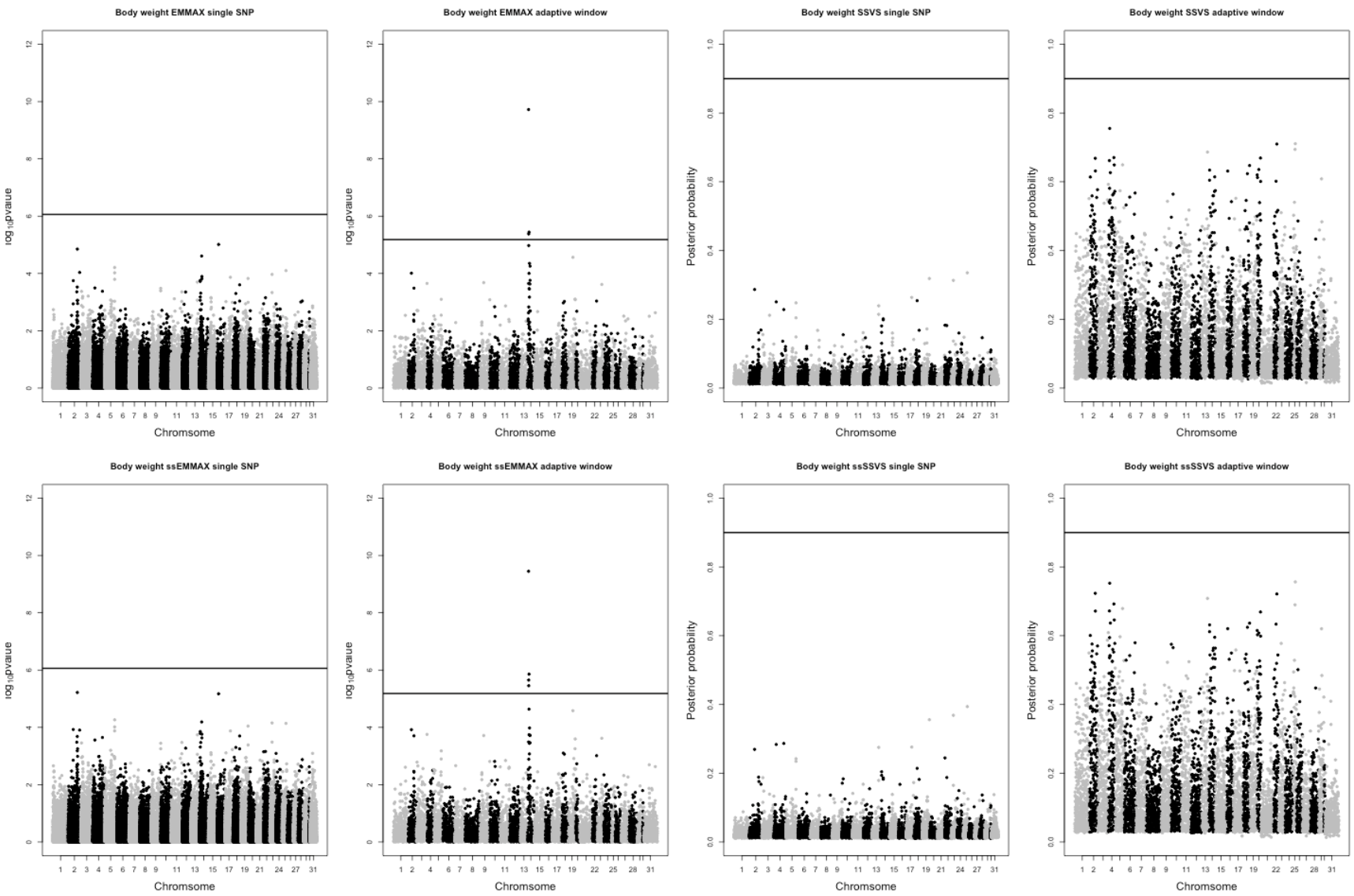

Figure B.15 Partition 4: Manhattan plot for body weight in within station study.

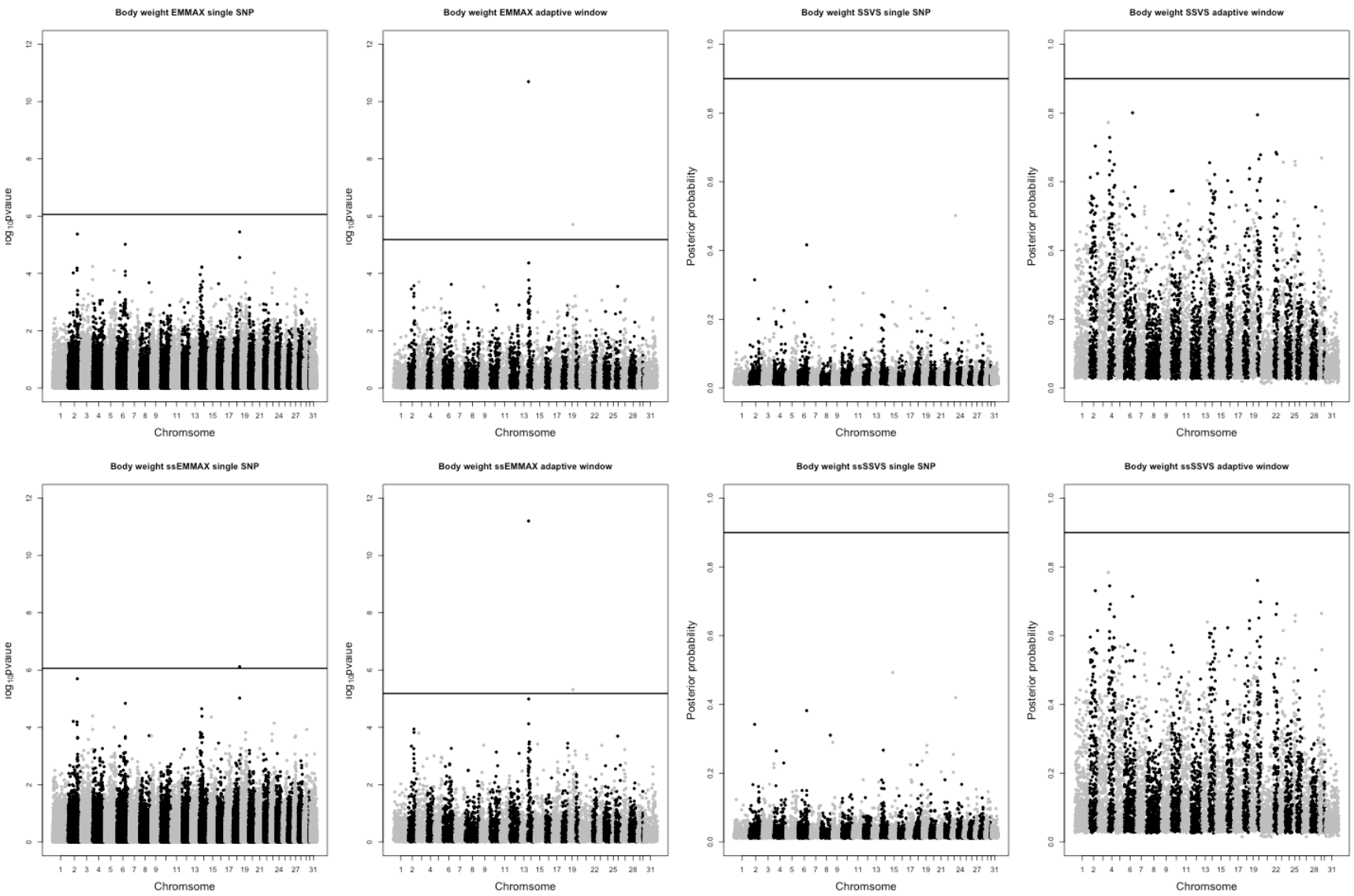

Figure B.16 Partition 5: Manhattan plot for body weight in within station study.

Figures B17-B22: Supplementary Manhattan plot figures for acroos-station splits for genotyped and masked genotyped animals with genotype masking on cows from ISU (Figure B17), MSU (Figure B18), USDFRC (Figure B19), UW (Figure B20), and FL (Figure B21) and AGIL (Figure B22) for milk fat. Panel A: Plot of –log10(P-value) versus genomic region for single SNP associations using EMMAX without using phenotypes on non-genotyped animals; Panel B: Plot of –log10(P-value) versus genomic region for genomic window associations using EMMAX without using phenotypes on non-genotyped animals; Panel C: Plot of posterior probabilities versus genomic region for genomic window associations using SSVS without using phenotypes on non-genotyped animals; Panel D: Plot of posterior probabilities versus genomic region for genomic window associations using SSVS without using phenotypes on non-genotyped animals; Panel E: Plot of –log10(P-value) versus genomic region for single SNP associations using ssEMMAX using phenotypes on non-genotyped animals; Panel F: Plot of –log10(P-value) versus genomic region for genomic window associations using ssEMMAX using phenotypes on non-genotyped animals; Panel G: Plot of posterior probabilities versus genomic region for genomic window associations using ssSSVS using phenotypes on non-genotyped animals; Panel H: Plot of posterior probabilities versus genomic region for genomic window associations using ssSSVS using phenotypes on non-genotyped animals.

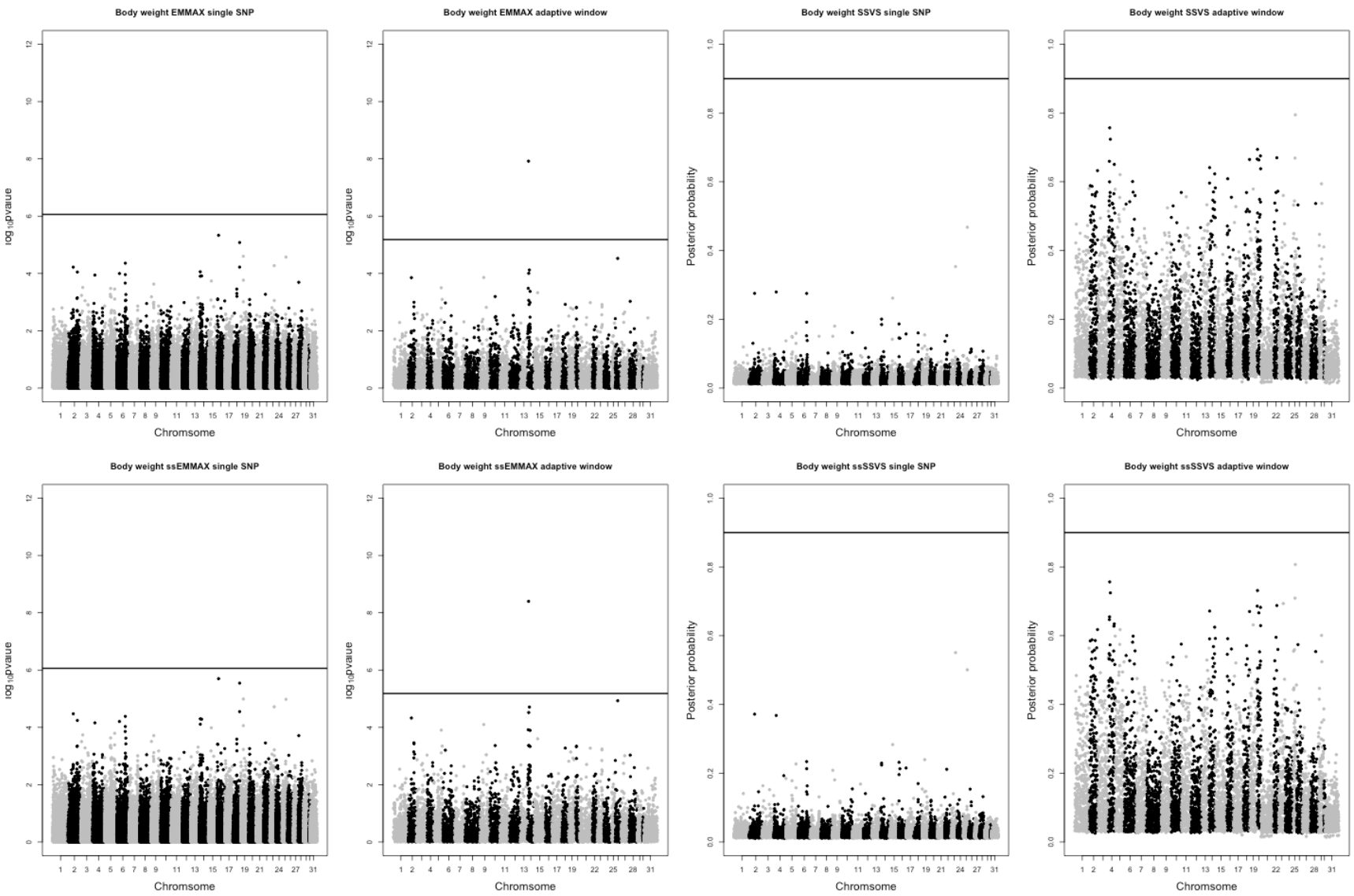

Figure B.17 Without the genotype of ISU: Manhattan plot for body weight in across station study.

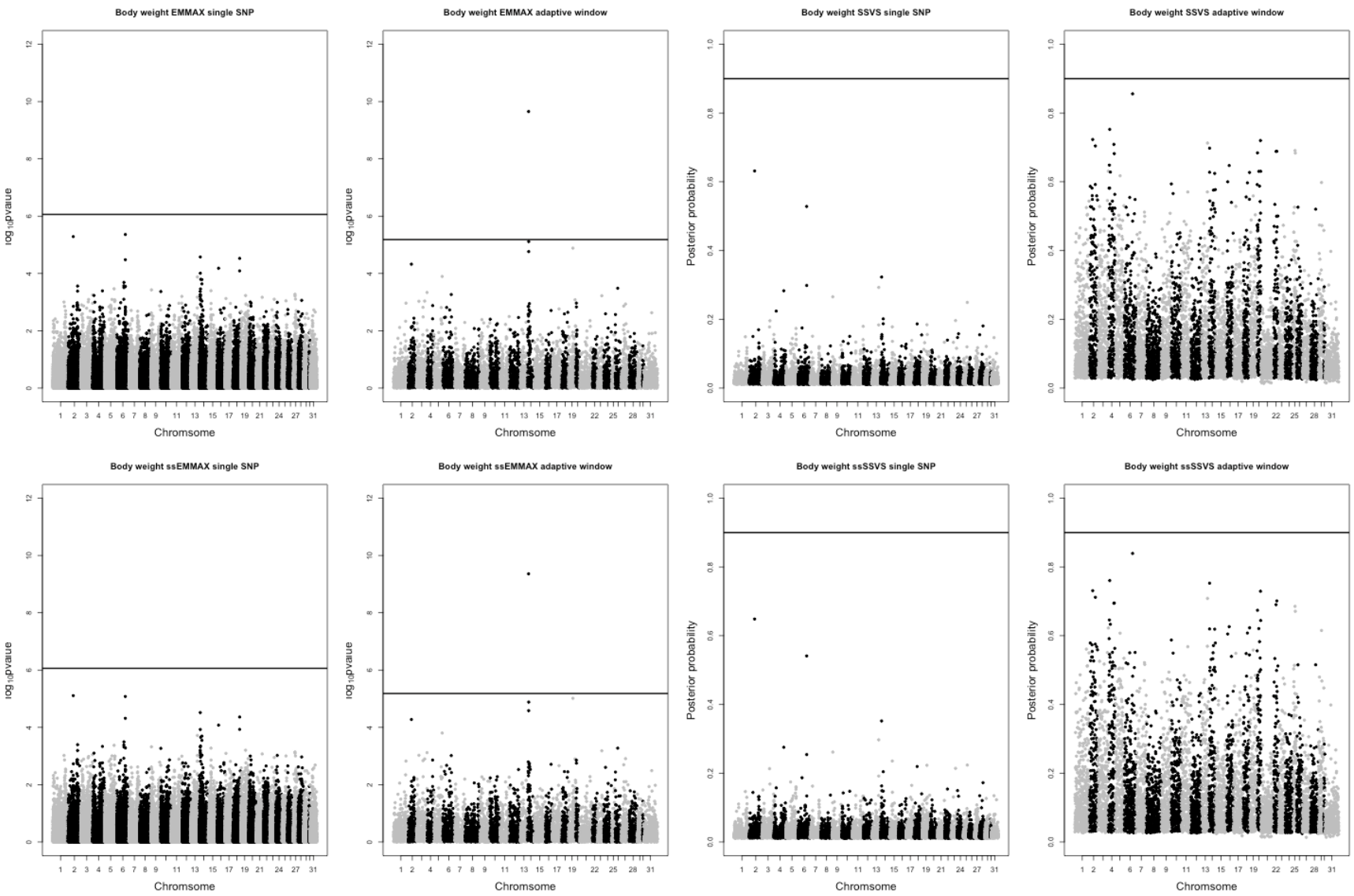

Figure B.18. Without the genotype of MSU: Manhattan plot for body weight in across station study.

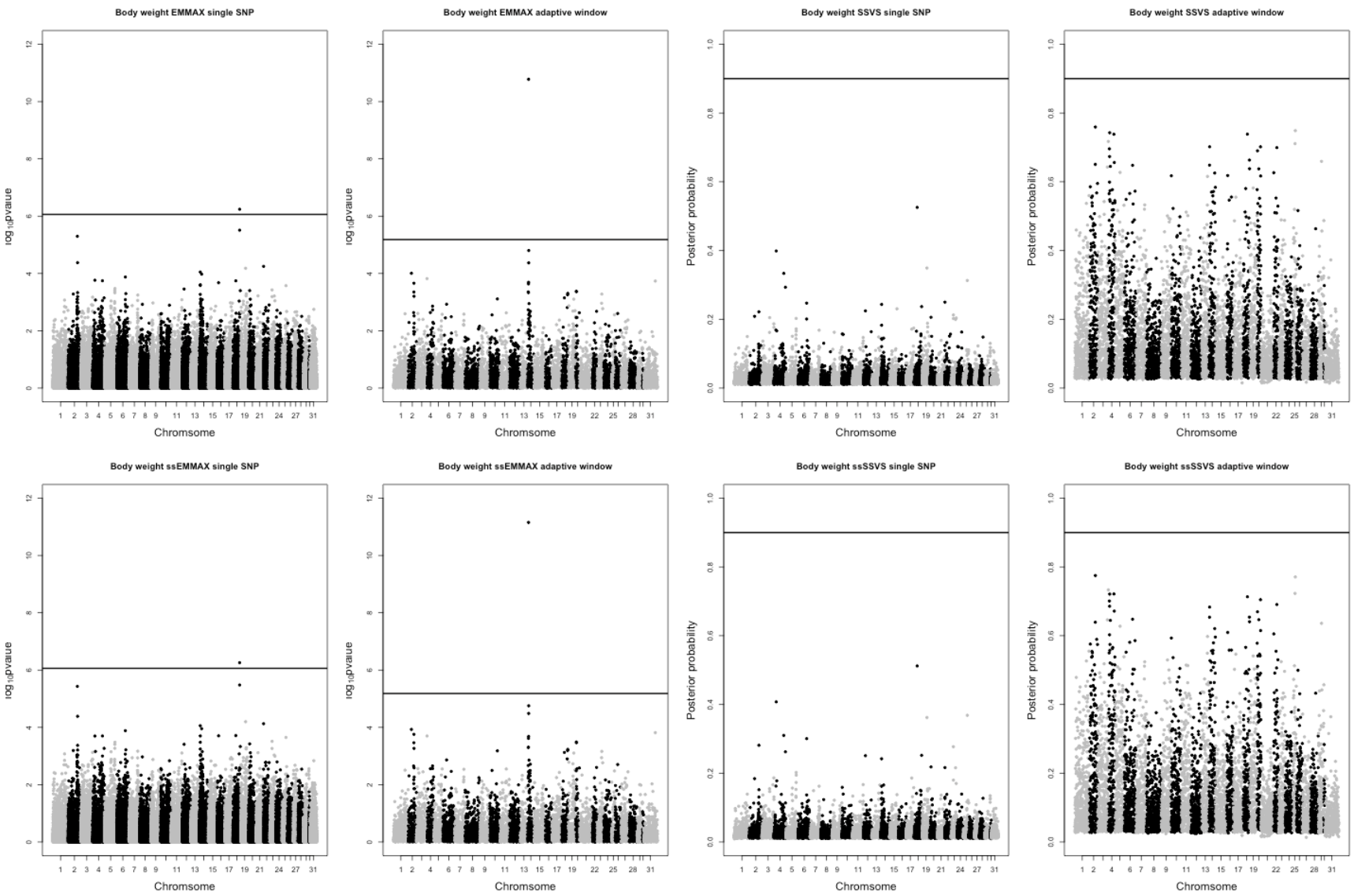

Figure B.19 Without the genotype of USDFRC: Manhattan plot for body weight in across station study.

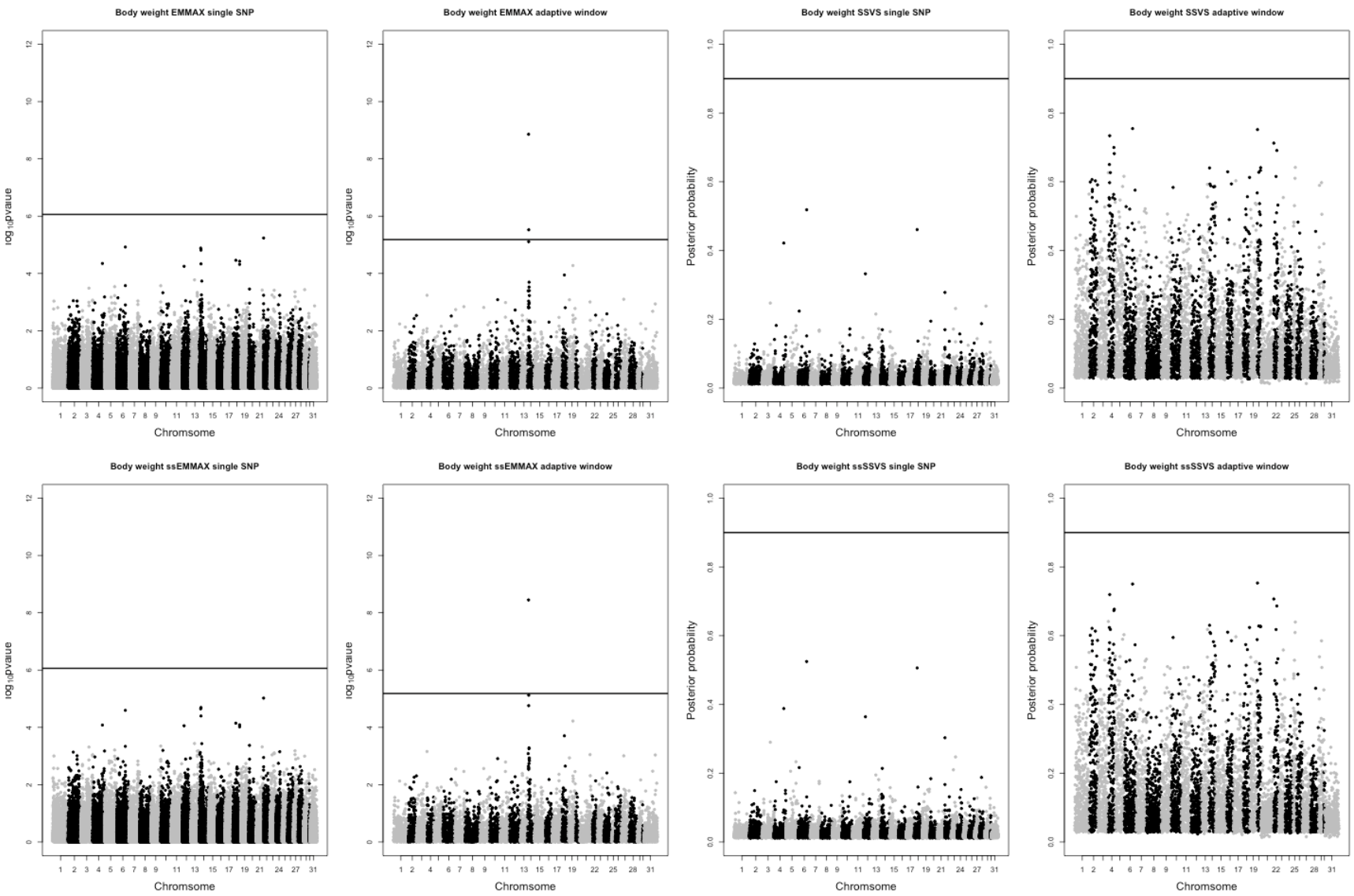

Figure B.20. Without the genotype of UW: Manhattan plot for body weight in across station study.

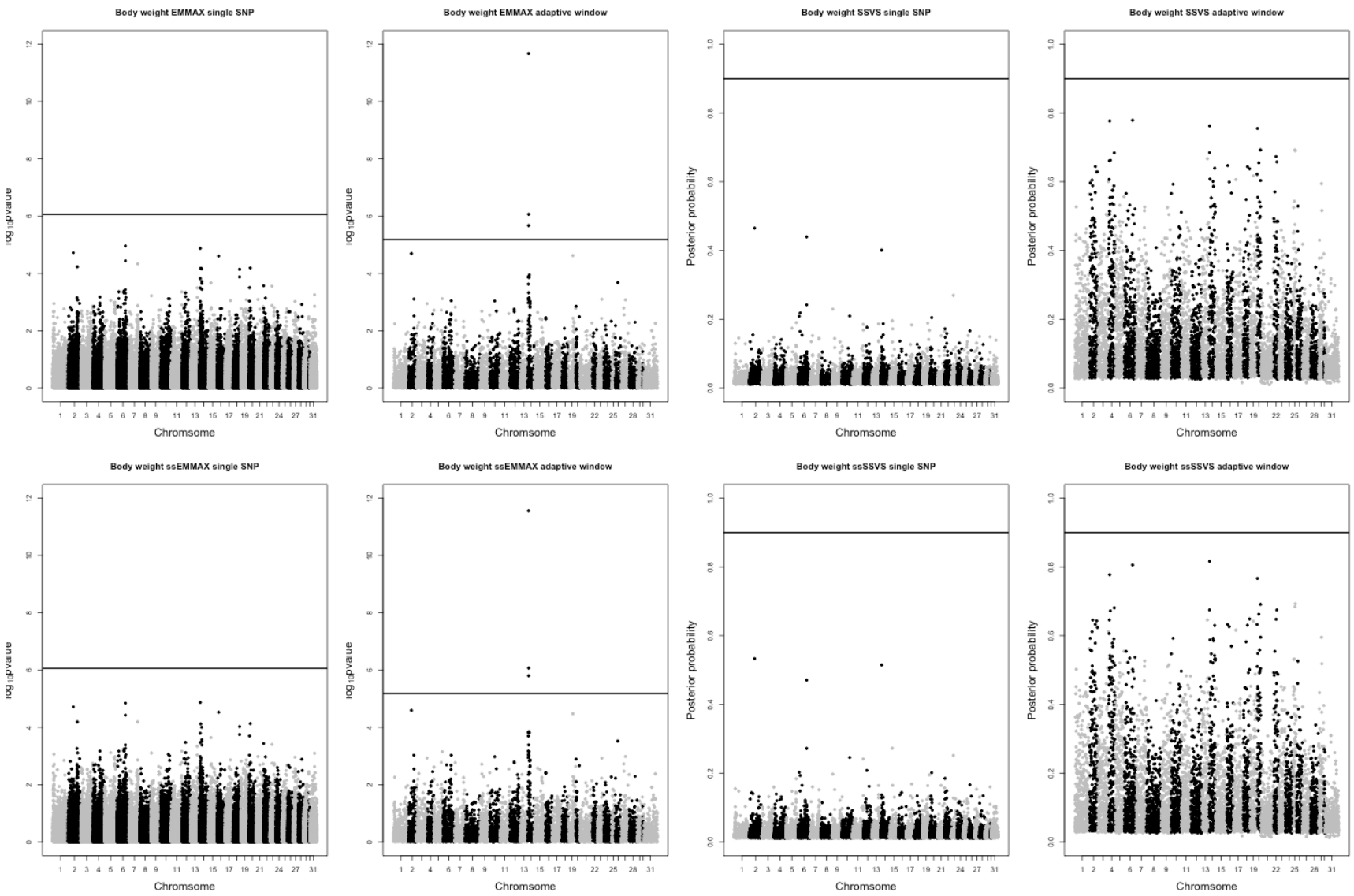

Figure B.21 Without the genotype of FL: Manhattan plot for body weight in across station study.

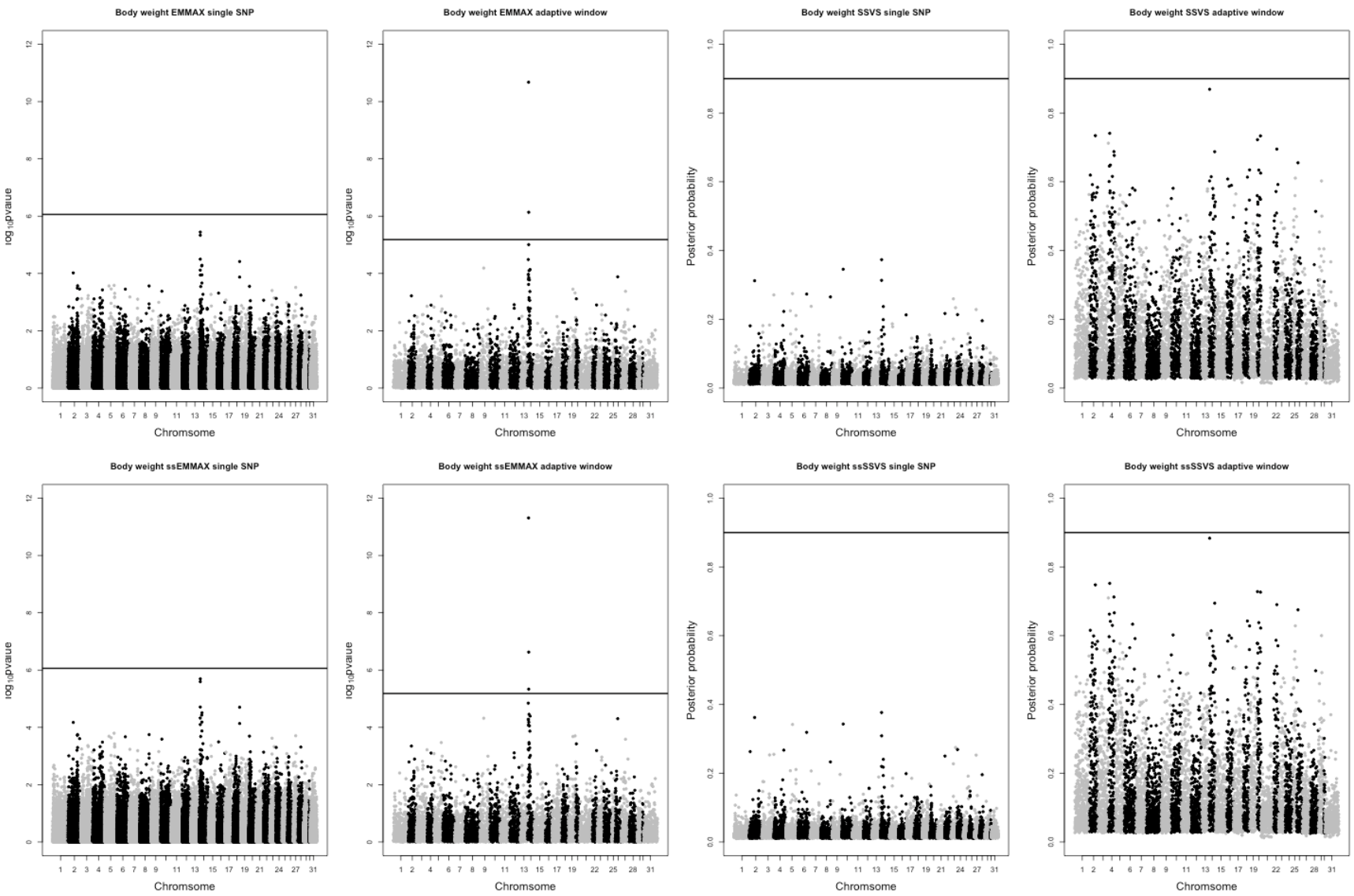

Figure B.22 Without the genotype of AGIL: Manhattan plot for body weight in across station study.

Appendix C: Chapter 5

Initial value for hyperparameters

In this section, I discuss the initial values for priors based on rules provided by de Los Campos *et al.* (2013). To start with, the default heritability `h2` is 0.5 and can be changed in `set.init` function. The initial starting value for residual variance σ_e^2 is $\text{var}(y)\times(1-h2)\times(v_e+2) = \text{var}(y)\times(1-h2)$ as v_e=-1 by default. This setting is true for all models. For all antedependence models, the default initial starting values for $\mu_t = 0$ and $\sigma_t^2 = 0.5$ with prior $\mu_t \sim N(0, 0.01)$ and $\sigma_t^2 \sim \chi^{-2}(-1, 0)$ as suggested by Yang and Tempelman (2012). Note all the initial values can be set manually if prior knowledge about the data is available, see `help(set.options)` for details.

BRR/GBLUP/ssGBLUP

The marker variance is initially set to be $\sigma_\alpha^2 = \text{var}(y)\times h2 / MS_M$ as $v_\alpha = -1$ for BRR/GBLUP where $MS_M = n^{-1}\sum_{i=1}^{n}\sum_{j=1}^{m} M_{ij}^2$ is sum of the sample variance of the column of the marker genotype type matrix. For ssGBLUP if it's `hetVAR` the initial value for the genetic variance not accounted by marker genotype σ_u^2 is set to $\text{var}(y)\times h2 / MS_M$ as well and this is also the same with all ssBayesA, ssBayesB and ssSSVS. For `ssGBLUP` with `homVAR`, $\sigma_\alpha^2 = \sigma_u^2 = \text{var}(y)\times h2 / MS_M$.

BayesA/ssBayesA/anteBayesA/e-BayesA

The starting degrees of freedom parameter v_α is set to 5 by default and the scale parameter then is set to $s_\alpha^2 = \text{var}(y) \times h2 \times (v_\alpha - 2) / v_\alpha / MSm$ and the same value is used for σ_α^2 in mapBayesA. The shape parameter $a_s = 0.5$ and rate parameter $\beta_s = 0$ of *Gamma* corresponds to $\sigma_\alpha^2 \sim \chi^{-2}(-1,0)$ suggested by Gelman (2006). For the UNIMH sample of both v_α and s_α^2, the tuning procedure is the same as suggested by Yang *et al.* (2015b).

BayesB/ssBayesB/anteBayesB

The degree of freedom v_α is set to 5 by default and the scale parameter then is set to $s_\alpha^2 = \text{var}(y) \times h2 \times (v_\alpha - 2) / v_\alpha / MS_M / \pi_\alpha$ where π_α is set to 0.05 initially by default with a prior of $\pi_\alpha \sim Beta(1,9)$ that has prior mean of 0.1. The shape parameter $a_s = 0.1$ and rate parameter $\beta_s = 0.1$ as in Yang and Tempelman (2012).

SSVS/ssSSVS/e-SSVS

The initial variance component is $\sigma_\alpha^2 = \text{var}(y) \times h2 \times c / MS_M / (c+(1-\pi_\tau) \times (1-c))$ where π_τ initialized as 0.05 like BayesB with prior of $\pi_\tau \sim Beta(1,9)$ and c=1000 by default.

```
# `BALD` package is available at http://www.math-evry.cnrs.fr/logiciels/bald.
#Linux or Mac are recommended since `BALD` is not available at CRAN.
#On windows,`RTools` needs to be pre-installed from https://cran.r-project.org/bin/windows/Rtools/ before installing BALD.
#Installing pre-required R packages for BALD
source("https://bioconductor.org/biocLite.R")
biocLite("chopsticks")
biocLite("snpStats")
biocLite("ROC")
install.packages(c("LDheatmap","quadrupen", "ROC", "grplasso","snpStats"))
# Download and install BALD using the following commands
system("wget http://www.math-evry.cnrs.fr/_media/logiciels/bald_0.2.1.tar.gz")
system("R CMD INSTALL bald_0.2.1.tar.gz")
# Then we load BALD package and load the Pig data
library(BALD)
library(BATools)
data(Pig)
map=PigMap

# Spilt the map for each chromosome because adaptive window is computed by chromosome
chrs=list()
for(i in 1:max(map$chr)){
  ii=which(map$chr==i)
  chrs[[i]]=geno[,ii]
}

#Then we can create adaptive window for each chromosome
#This will take 6-7 hours to run in serial
#It is suggested to run it in parallel for each chromosome in a computing cluster
adaptiveWindows=list()
for(i in 1:length(chrs)){
  Z=chr[[i]]+1
  p=dim(Z)[2]
  gapS <- gapStatistic(Z, min.nc=2, max.nc=p-1, B=50)
  gapS$best.k
  adaptiveWindows[[i]] <- cutree(gapS$tree, gapS$best.k)
}

#Finally, we compute the window id for each SNP and add it to the map
idw<-adaptiveWindows[[1]]
for(i in 2:length(adaptiveWindows)){
  tmp<-max(idw)
  idw<-c(idw,adaptiveWindows[[i]]+tmp)
}
map$idw=idw

#For fixed size window, use set.win function in BATools
#For example, to create 1Mb window, simply run
map<-set.win(map = map,len=1,unit = "Mb")
#To create 5-SNP window, simply run
map<-set.win(map = map,len = 5,unit="count")
```

Figure C.1 Example on creating adaptive window using BALD and fix size window for the Pig data

REFERENCES

REFERENCES

Aguilar, I., I. Misztal, D. L. Johnson, A. Legarra, S. Tsuruta *et al.*, 2010 Hot topic: a unified approach to utilize phenotypic, full pedigree, and genomic information for genetic evaluation of Holstein final score. J Dairy Sci 93**:** 743-752.

Andrews, D. F., and C. L. Mallows, 1974 Scale mixtures of normal distributions. J R Stat Soc Series B Methodol 36**:** 99-102.

Bates, D., and D. Eddelbuettel, 2013 Fast and Elegant Numerical Linear Algebra Using the RcppEigen Package. Journal of Statistical Software; Vol 1, Issue 5 (2013).

Bello, N. M., J. P. Steibel and R. J. Tempelman, 2010 Hierarchical Bayesian modeling of random and residual variance-covariance matrices in bivariate mixed effects models. Biom J 52**:** 297-313.

Bernal Rubio, Y. L., J. L. Gualdron Duarte, R. O. Bates, C. W. Ernst, D. Nonneman *et al.*, 2016 Meta-analysis of genome-wide association from genomic prediction models. Anim Genet 47**:** 36-48.

Bezanson, J., S. Karpinski, V. B. Shah and A. Edelman, 2012 Julia: A fast dynamic language for technical computing, pp. arXiv preprint arXiv:1209.5145.

Brøndum, R. F., G. Su, L. Janss, G. Sahana, B. Guldbrandtsen *et al.*, 2015 Quantitative trait loci markers derived from whole genome sequence data increases the reliability of genomic prediction. J Dairy Sci 98**:** 4107-4116.

Cai, X., A. Huang and S. Xu, 2011 Fast empirical Bayesian LASSO for multiple quantitative trait locus mapping. BMC bioinformatics 12**:** 211.

Calus, M. P., J. Vandenplas and J. Ten Napel, 2015 Ever-growing data sets pose (new) challenges to genomic prediction models. J Anim Breed Genet 132**:** 407-408.

Casella, G., 1985 An Introduction to Empirical Bayes Analysis. The American Statistician 39**:** 83-87.

Casella, G., and E. I. George, 1992 Explaining the Gibbs Sampler. The American Statistician 46**:** 167-174.

Chen, C., J. P. Steibel and R. J. Tempelman, 2017 Genome-Wide Association Analyses Based on Broadly Different Specifications for Prior Distributions, Genomic Windows, and Estimation Methods. Genetics 206**:** 1791.

Chen, C., and R. J. Tempelman, 2015 An integrated approach to empirical Bayesian whole genome prediction modeling. JABES 20**:** 491-511.

Chen, C. Y., I. Misztal, I. Aguilar, A. Legarra and W. M. Muir, 2011a Effect of different genomic relationship matrices on accuracy and scale. J Anim Sci 89**:** 2673-2679.

Chen, C. Y., I. Misztal, I. Aguilar, S. Tsuruta, T. H. E. Meuwissen *et al.*, 2011b Genome-wide marker-assisted selection combining all pedigree phenotypic information with genotypic data in one step: An example using broiler chickens. J Anim Sci 89**:** 23-28.

Cheng, H., R. Fernando and D. Garrick, 2017 Parallel Computing to Speed up Whole-Genome Bayesian Regression Analyses Using Orthogonal Data Augmentation. bioRxiv.

Cheng, H., D. Garrick and R. Fernando, 2016 JWAS: Julia implementation of whole-genome analyses software using univariate and multivariate Bayesian mixed effects model, pp.

Colombani, C., A. Legarra, S. Fritz, F. Guillaume, P. Croiseau *et al.*, 2013 Application of Bayesian least absolute shrinkage and selection operator (LASSO) and BayesCpi methods for genomic selection in French Holstein and Montbeliarde breeds. J Dairy Sci 96**:** 575-591.

Cuyabano, B. C., G. Su and M. S. Lund, 2014 Genomic prediction of genetic merit using LD-based haplotypes in the Nordic Holstein population. Bmc Genomics 15**:** 1171.

Daetwyler, H. D., A. Capitan, H. Pausch, P. Stothard, R. van Binsbergen *et al.*, 2014 Whole-genome sequencing of 234 bulls facilitates mapping of monogenic and complex traits in cattle. Nat Genet 46**:** 858-865.

de Los Campos, G., J. M. Hickey, R. Pong-Wong, H. D. Daetwyler and M. P. Calus, 2013 Whole-genome regression and prediction methods applied to plant and animal breeding. Genetics 193**:** 327-345.

Dehman, A., C. Ambroise and P. Neuvial, 2015 Performance of a blockwise approach in variable selection using linkage disequilibrium information. BMC bioinformatics 16**:** 148.

Dehman, A., and P. Neuvial, 2015 BALD: Blockwise Approach using Linkage Disequilibrium information. R package version 0.2.1.

Edwards, D. B., C. W. Ernst, N. E. Raney, M. E. Doumit, M. D. Hoge *et al.*, 2008 Quantitative trait locus mapping in an F2 Duroc x Pietrain resource population: II. Carcass and meat quality traits. J Anim Sci 86**:** 254-266.

Endelman, J. B., 2011 Ridge Regression and Other Kernels for Genomic Selection with R Package rrBLUP. Plant Genome-Us 4**:** 250-255.

Erbe, M., B. J. Hayes, L. K. Matukumalli, S. Goswami, P. J. Bowman *et al.*, 2012 Improving accuracy of genomic predictions within and between dairy cattle breeds with imputed high-density single nucleotide polymorphism panels. J Dairy Sci 95**:** 4114-4129.

Fan, B., S. K. Onteru, Z. Q. Du, D. J. Garrick, K. J. Stalder *et al.*, 2011 Genome-wide association study identifies Loci for body composition and structural soundness traits in pigs. PloS one 6**:** e14726.

Fernando, R., and D. Garrick, 2013 Bayesian Methods Applied to GWAS, pp. 237-274 in *Genome-Wide Association Studies and Genomic Prediction*, edited by C. Gondro, J. van der Werf and B. Hayes. Humana Press.

Fernando, R., A. Toosi, A. Wolc, D. Garrick and J. Dekkers, 2017 Application of Whole-Genome Prediction Methods for Genome-Wide Association Studies: A Bayesian Approach. Journal of Agricultural, Biological and Environmental Statistics 22**:** 172-193.

Fernando, R. L., H. Cheng, B. L. Golden and D. J. Garrick, 2016 Computational strategies for alternative single-step Bayesian regression models with large numbers of genotyped and non-genotyped animals. Genetics Selection Evolution 48**:** 96.

Fernando, R. L., J. C. M. Dekkers and D. J. Garrick, 2014 A class of Bayesian methods to combine large numbers of genotyped and non-genotyped animals for whole-genome analyses. Genetics Selection Evolution 46.

Fernando, R. L., and D. J. Garrick, 2009 in *GenSel - user manual*.

Fragomeni, B. d. O., I. Misztal, D. L. Lourenco, I. Aguilar, R. Okimoto *et al.*, 2014 Changes in variance explained by top SNP windows over generations for three traits in broiler chicken. Frontiers in Genetics 5**:** 332.

García-Ruiz, A., J. B. Cole, P. M. VanRaden, G. R. Wiggans, F. J. Ruiz-López *et al.*, 2016 Changes in genetic selection differentials and generation intervals in US Holstein dairy cattle as a result of genomic selection. Proceedings of the National Academy of Sciences of the United States of America 113**:** E3995-E4004.

Garrick, D. J., J. F. Taylor and R. L. Fernando, 2009 Deregressing estimated breeding values and weighting information for genomic regression analyses. Genetics Selection Evolution 41**:** 55.

Gelman, A., 2006 Prior distributions for variance parameters in hierarchical models (Comment on an Article by Browne and Draper). Bayesian Anal 1**:** 515-533.

Gelman, A., J. Hill and M. Yajima, 2012 Why I (Usually) don't have to worry about multiple comparisons. J Res Educ Effectiveness 5**:** 189-211.

George, E. I., and R. E. McCulloch, 1993 Variable selection via Gibbs sampling. J Amer Statist Assoc 88**:** 881 - 889.

Gianola, D., 2013 Priors in whole-genome regression: the bayesian alphabet returns. Genetics 194**:** 573-596.

Gianola, D., G. de los Campos, W. G. Hill, E. Manfredi and R. Fernando, 2009 Additive Genetic Variability and the Bayesian Alphabet. Genetics 183**:** 347-363.

Gianola, D., J. L. Foulley and R. Fernando, 1986 Prediction of breeding values when variances are not known. Genetics, Selection, Evolution 18**:** 485-498.

Gianola, D., M. Perez-Enciso and M. A. Toro, 2003 On marker-assisted prediction of genetic value: Beyond the ridge. Genetics 163**:** 347-365.

Gilmour, A. R., R. Thompson and B. R. Cullis, 1995 Average information REML: An efficient algorithm for variance parameter estimation in linear mixed models. Biometrics 51**:** 1440-1450.

Goddard, M. E., and B. J. Hayes, 2009 Mapping genes for complex traits in domestic animals and their use in breeding programmes. Nature reviews. Genetics 10**:** 381-391.

Goddard, M. E., K. E. Kemper, I. M. MacLeod, A. J. Chamberlain and B. J. Hayes, 2016 Genetics of complex traits: prediction of phenotype, identification of causal polymorphisms and genetic architecture. P Roy Soc B-Biol Sci 283.

Gray, K. A., J. P. Cassady, Y. J. Huang and C. Maltecca, 2012 Effectiveness of genomic prediction on milk flow traits in dairy cattle. Genetics Selection Evolution 44.

Grisart, B., W. Coppieters, F. Farnir, L. Karim, C. Ford *et al.*, 2002 Positional candidate cloning of a QTL in dairy cattle: Identification of a missense mutation in the bovine DGAT1 gene with major effect on milk yield and composition. Genome Res 12**:** 222-231.

Groenen, M. A., 2016 A decade of pig genome sequencing: a window on pig domestication and evolution. Genetics, selection, evolution : GSE 48**:** 23.

Gualdron Duarte, J. L., R. O. Bates, C. W. Ernst, N. E. Raney, R. J. Cantet *et al.*, 2013 Genotype imputation accuracy in a F2 pig population using high density and low density SNP panels. BMC genetics 14**:** 38.

Gualdron Duarte, J. L., R. J. Cantet, R. O. Bates, C. W. Ernst, N. E. Raney *et al.*, 2014 Rapid screening for phenotype-genotype associations by linear transformations of genomic evaluations. BMC bioinformatics 15**:** 246.

Guan, Y., and M. Stephens, 2011 Bayesian variable selection regression for genome-wide association studies and other large-scale problems. Ann Appl Stat 5**:** 1780-1815.

Habier, D., R. L. Fernando, K. Kizilkaya and D. J. Garrick, 2011 Extension of the bayesian alphabet for genomic selection. BMC bioinformatics 12**:** 186.

Harville, D. A., 1974 Bayesian inference for variance components using only error contrasts. Biometrika 61**:** 383-385.

Harville, D. A., 1977 Maximum Likelihood Approaches to Variance Component Estimation and to Related Problems. Journal of the American Statistical Association 72**:** 320-338.

Hayashi, T., and H. Iwata, 2010 EM algorithm for Bayesian estimation of genomic breeding values. BMC genetics 11**:** 3.

Hayes, B., 2013 Overview of statistical methods for genome-wide association Studies (GWAS), pp. 149-169 in *Genome-Wide Association Studies and Genomic Prediction*, edited by C. Gondro, J. van der Werf and B. Hayes. Humana Press.

Hayes, B., and M. E. Goddard, 2001 The distribution of the effects of genes affecting quantitative traits in livestock. Genetics Selection Evolution 33**:** 209-229.

Hayes, B. J., P. J. Bowman, A. J. Chamberlain and M. E. Goddard, 2009 Invited review: Genomic selection in dairy cattle: Progress and challenges. J Dairy Sci 92.

Hayes, B. J., J. Pryce, A. J. Chamberlain, P. J. Bowman and M. E. Goddard, 2010 Genetic architecture of complex traits and accuracy of genomic prediction: coat colour, milk-fat percentage, and type in Holstein cattle as contrasting model traits. Plos Genet 6**:** e1001139.

Henderson, C. R., 1975 Best Linear Unbiased Estimation and Prediction under a Selection Model. Biometrics 31**:** 423-447.

Henderson, C. R., 1985 Equivalent Linear Models to Reduce Computations. J Dairy Sci 68**:** 2267-2277.

Hoerl, A. E., and R. W. Kennard, 1970 Ridge Regression - Biased Estimation for Nonorthogonal Problems. Technometrics 12**:** 55-&.

Huang, A., S. Xu and X. Cai, 2015 Empirical Bayesian elastic net for multiple quantitative trait locus mapping. Heredity 114**:** 107-115.

Jiang, J., Q. Zhang, L. Ma, J. Li, Z. Wang *et al.*, 2015 Joint prediction of multiple quantitative traits using a Bayesian multivariate antedependence model. Heredity 115**:** 29-36.

Johnson, D. L., and R. Thompson, 1995 Restricted maximum likelihood estimation of variance components for univariate animal models using sparse matrix techniques and average information. J Dairy Sci 78**:** 449-456.

Kane, M., J. W. Emerson and S. Weston, 2013 Scalable Strategies for Computing with Massive Data. Journal of Statistical Software; Vol 1, Issue 14 (2013).

Kang, H. M., J. H. Sul, S. K. Service, N. A. Zaitlen, S. Y. Kong *et al.*, 2010 Variance component model to account for sample structure in genome-wide association studies. Nat Genet 42**:** 348-354.

Kang, H. M., N. A. Zaitlen, C. M. Wade, A. Kirby, D. Heckerman *et al.*, 2008 Efficient control of population structure in model organism association mapping. Genetics 178**:** 1709-1723.

Karkkainen, H. P., and M. J. Sillanpaa, 2012 Back to basics for Bayesian model building in genomic selection. Genetics 191**:** 969-987.

Kemper, K. E., C. M. Reich, P. J. Bowman, C. J. Vander Jagt, A. J. Chamberlain *et al.*, 2015 Improved precision of QTL mapping using a nonlinear Bayesian method in a multi-breed

population leads to greater accuracy of across-breed genomic predictions. Genetics, selection, evolution : GSE 47**:** 29.

Klein, R. J., C. Zeiss, E. Y. Chew, J.-Y. Tsai, R. S. Sackler *et al.*, 2005 Complement Factor H Polymorphism in Age-Related Macular Degeneration. Science (New York, N.Y.) 308**:** 385-389.

Knürr, T., E. Läärä and M. J. Sillanpää, 2013 Impact of prior specifications in ashrinkage-inducing Bayesian model for quantitative trait mapping and genomic prediction. Genetics, selection, evolution : GSE 45**:** 24-24.

Lee, J., H. Cheng, D. Garrick, B. Golden, J. Dekkers *et al.*, 2017 Comparison of alternative approaches to single-trait genomic prediction using genotyped and non-genotyped Hanwoo beef cattle. Genetics Selection Evolution 49**:** 2.

Legarra, A., O. F. Christensen, I. Aguilar and I. Misztal, 2014 Single Step, a general approach for genomic selection. Livest Sci 166**:** 54-65.

Lehermeier, C., V. Wimmer, T. Albrecht, H. J. Auinger, D. Gianola *et al.*, 2013 Sensitivity to prior specification in Bayesian genome-based prediction models. Statistical applications in genetics and molecular biology 12**:** 375-391.

Lippert, C., J. Listgarten, Y. Liu, C. M. Kadie, R. I. Davidson *et al.*, 2011 FaST linear mixed models for genome-wide association studies. Nat Methods 8**:** 833-835.

Louis, T. A., 1982 Finding the observed information matrix when using the EM algorithm. J R Stat Soc Series B Methodol 44**:** 226-233.

Lourenco, D. A. L., I. Misztal, H. Wang, I. Aguilar, S. Tsuruta *et al.*, 2013 Prediction accuracy for a simulated maternally affected trait of beef cattle using different genomic evaluation models. J Anim Sci 91**:** 4090-4098.

Lourenco, D. A. L., S. Tsuruta, B. O. Fragomeni, Y. Masuda, I. Aguilar *et al.*, 2015 Genetic evaluation using single-step genomic best linear unbiased predictor in American Angus. J Anim Sci 93**:** 2653-2662.

Lu, Y., 2016 Quantitative Genetic and Genomic Modeling of Feed Efficiency in Dairy Cattle, pp. in *Animal Scienece*. Michigan State University.

Lu, Y., M. J. Vandehaar, D. M. Spurlock, K. A. Weigel, L. E. Armentano *et al.*, 2015 An alternative approach to modeling genetic merit of feed efficiency in dairy cattle. J Dairy Sci 98**:** 6535-6551.

Ma, H., A. I. Bandos, H. E. Rockette and D. Gur, 2013 On use of partial area under the ROC curve for evaluation of diagnostic performance. Statistics in Medicine 32**:** 3449-3458.

Martin, L. S., and E. Eskin, 2016 Review: Population Structure in Genetic Studies: Confounding Factors and Mixed Models. bioRxiv.

Masuda, Y., I. Misztal, S. Tsuruta, A. Legarra, I. Aguilar *et al.*, 2016 Implementation of genomic recursions in single-step genomic best linear unbiased predictor for US Holsteins with a large number of genotyped animals. J Dairy Sci 99**:** 1968-1974.

Metz, C. E., 1978 Basic Principles of Roc Analysis. Semin Nucl Med 8**:** 283-298.

Meuwissen, T., B. Hayes and M. Goddard, 2016 Genomic selection: A paradigm shift in animal breeding. Animal Frontiers 6**:** 6-14.

Meuwissen, T. H., T. R. Solberg, R. Shepherd and J. A. Woolliams, 2009 A fast algorithm for BayesB type of prediction of genome-wide estimates of genetic value. Genetics, selection, evolution : GSE 41**:** 2.

Meuwissen, T. H. E., B. J. Hayes and M. E. Goddard, 2001 Prediction of total genetic value using genome-wide dense marker maps. Genetics 157**:** 1819-1829.

Misztal, I., 2016a Inexpensive Computation of the Inverse of the Genomic Relationship Matrix in Populations with Small Effective Population Size. Genetics 202**:** 401-409.

Misztal, I., 2016b Is genomic selection now a mature technology? J. Anim. Breed. Genet. 133**:** 81-82.

Misztal, I., S. Tsuruta, T. Strabel, B. Auvray, T. Druet *et al.*, 2002 BLUPF90 and related programs (BGF90) in *7th world congress on genetics applied to livestock production*, Montpellier.

Moser, G., S. H. Lee, B. J. Hayes, M. E. Goddard, N. R. Wray *et al.*, 2015 Simultaneous discovery, estimation and prediction analysis of complex traits using a Bayesian mixture model. Plos Genet 11**:** e1004969.

Nadaf, J., V. Riggio, T. P. Yu and R. Pong-Wong, 2012 Effect of the prior distribution of SNP effects on the estimation of total breeding value. BMC proceedings 6 Suppl 2**:** S6.

Ou, Z., R. J. Tempelman, J. P. Steibel, C. W. Ernst, R. O. Bates *et al.*, 2016 Genomic Prediction Accounting for Residual Heteroskedasticity. G3: Genes|Genomes|Genetics 6**:** 1.

Perez, P., and G. de los Campos, 2014 Genome-wide regression and prediction with the BGLR statistical package. Genetics 198**:** 483-495.

Perez, P., G. de Los Campos, J. Crossa and D. Gianola, 2010 Genomic-Enabled Prediction Based on Molecular Markers and Pedigree Using the Bayesian Linear Regression Package in R. 3**:** 106-116.

Pryce, J. E., J. Arias, P. J. Bowman, S. R. Davis, K. A. Macdonald *et al.*, 2012 Accuracy of genomic predictions of residual feed intake and 250-day body weight in growing heifers using 625,000 single nucleotide polymorphism markers. J Dairy Sci 95**:** 2108-2119.

Purcell, S., B. Neale, K. Todd-Brown, L. Thomas, M. A. Ferreira *et al.*, 2007 PLINK: a tool set for whole-genome association and population-based linkage analyses. Am J Hum Genet 81**:** 559-575.

R Core Team, 2017 R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria., pp., Vienna, Austria.

Resende, M. F. R., P. Munoz, M. D. V. Resende, D. J. Garrick, R. L. Fernando *et al.*, 2012 Accuracy of Genomic Selection Methods in a Standard Data Set of Loblolly Pine (Pinus taeda L.). Genetics 190**:** 1503-1510.

Robinson, G. K., 1991 That BLUP is a good thing: the estimation of random effects. Statist. Sci. 6**:** 15-32.

Rockova, V., and E. I. George, 2014 EMVS: The EM Approach to Bayesian Variable Selection. Journal of the American Statistical Association 109**:** 828-846.

Schmid, K., and Z. Yang, 2008 The trouble with sliding windows and the selective pressure in BRCA1. PloS one 3**:** e3746.

Searle, S. R., G. Casella and C. E. McCulloch, 1992 *Variance components*. Wiley, New York.

Shepherd, R. K., T. H. Meuwissen and J. A. Woolliams, 2010 Genomic selection and complex trait prediction using a fast EM algorithm applied to genome-wide markers. BMC bioinformatics 11**:** 529.

Sing, T., O. Sander, N. Beerenwinkel and T. Lengauer, 2005 ROCR: visualizing classifier performance in R. Bioinformatics 21**:** 3940-3941.

Sorensen, D., and D. Gianola, 2002 *Likelihood, Bayesian, and MCMC methods in quantitative genetics*. Springer-Verlag, New York.

Stephens, M., 2017 False discovery rates: a new deal. Biostatistics 18**:** 275-294.

Stephens, M., and D. J. Balding, 2009 Bayesian statistical methods for genetic association studies. Nat Rev Genet 10**:** 681-690.

Stram, D. O., and J. W. Lee, 1994 Variance Components Testing in the Longitudinal Mixed Effects Model. Biometrics 50**:** 1171-1177.

Stranden, I., and O. F. Christensen, 2011 Allele coding in genomic evaluation. Genetics, Selection, Evolution 43**:** 25.

Stranden, I., and D. J. Garrick, 2009 Technical note: Derivation of equivalent computing algorithms for genomic predictions and reliabilities of animal merit. J Dairy Sci 92**:** 2971-2975.

Sun, X., L. Qu, D. J. Garrick, J. C. Dekkers and R. L. Fernando, 2012 A fast EM algorithm for BayesA-like prediction of genomic breeding values. PloS one 7**:** e49157.

Technow, F., 2013 Simulation of genomic data in applied genetics. R package version 0.4., pp.

Tempelman, R. J., 2015 Statistical and computational challenges in whole genome prediction and genome-wide association analyses for plant and animal breeding. JABES 20**:** 442-466.

Tempelman, R. J., D. M. Spurlock, M. Coffey, R. F. Veerkamp, L. E. Armentano *et al.*, 2015 Heterogeneity in genetic and nongenetic variation and energy sink relationships for residual feed intake across research stations and countries. J Dairy Sci 98**:** 2013-2026.

Tizioto, P. C., J. F. Taylor, J. E. Decker, C. F. Gromboni, M. A. Mudadu *et al.*, 2015 Detection of quantitative trait loci for mineral content of Nelore longissimus dorsi muscle. Genetics, selection, evolution : GSE 47**:** 15.

Ueda, N., and R. Nakano, 1998 Deterministic annealing EM algorithm. Neural Networks 11**:** 271-282.

Vallejo, R. L., T. D. Leeds, B. O. Fragomeni, G. Gao, A. G. Hernandez *et al.*, 2016 Evaluation of Genome-Enabled Selection for Bacterial Cold Water Disease Resistance Using Progeny Performance Data in Rainbow Trout: Insights on Genotyping Methods and Genomic Prediction Models. Front Genet 7**:** 96.

van den Berg, I., S. Fritz and D. Boichard, 2013 QTL fine mapping with Bayes C(pi): a simulation study. Genetics Selection Evolution 45.

VanRaden, P. M., 2008 Efficient methods to compute genomic predictions. J Dairy Sci 91**:** 4414-4423.

Verbyla, K., B. Hayes, P. Bowman and M. Goddard, 2009 Accuracy of genomic selection using stochastic search variable selection in Australian Holstein Friesian dairy cattle. Genet Res 91**:** 307 - 311.

Visscher, Peter M., Matthew A. Brown, Mark I. McCarthy and J. Yang, 2012 Five Years of GWAS Discovery. The American Journal of Human Genetics 90**:** 7-24.

Wang, H., I. Misztal, I. Aguilar, A. Legarra and W. M. Muir, 2012 Genome-wide association mapping including phenotypes from relatives without genotypes. Genetics research 94**:** 73-83.

Wang, T., Y.-P. P. Chen, P. J. Bowman, M. E. Goddard and B. J. Hayes, 2016 A hybrid expectation maximisation and MCMC sampling algorithm to implement Bayesian mixture model based genomic prediction and QTL mapping. Bmc Genomics 17**:** 744.

Wang, X., N. J. Morris, X. Zhu and R. C. Elston, 2013 A variance component based multi-marker association test using family and unrelated data. BMC genetics 14**:** 17.

Warr, A., C. Robert, D. Hume, A. L. Archibald, N. Deeb *et al.*, 2015 Identification of Low-Confidence Regions in the Pig Reference Genome (Sscrofa 10.2). Frontiers in Genetics 6.

Wellcome Trust Case Control Consortium, 2007 Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. Nature 447**:** 661-678.

Wiggans, G. R., T. A. Cooper, C. P. Van Tassell, T. S. Sonstegard and E. B. Simpson, 2013 Technical note: Characteristics and use of the Illumina BovineLD and GeneSeek Genomic Profiler low-density bead chips for genomic evaluation. J Dairy Sci 96**:** 1258-1263.

Wiggans, G. R., P. M. VanRaden and T. A. Cooper, 2011 The genomic evaluation system in the United States: Past, present, future. J Dairy Sci 94**:** 3202-3211.

Wimmer, V., T. Albrecht, H. J. Auinger and C. C. Schon, 2012 synbreed: a framework for the analysis of genomic prediction data using R. Bioinformatics 28**:** 2086-2087.

Wimmer, V., C. Lehermeier, T. Albrecht, H.-J. Auinger, Y. Wang *et al.*, 2013 Genome-Wide Prediction of Traits with Different Genetic Architecture Through Efficient Variable Selection. Genetics 195**:** 573-587.

Wolc, A., J. Arango, P. Settar, J. E. Fulton, N. P. O'Sullivan *et al.*, 2016 Mixture models detect large effect QTL better than GBLUP and result in more accurate and persistent predictions. J Anim Sci Biotechnol 7**:** 7.

Wolc, A., J. Arango, P. Settar, J. E. Fulton, N. P. O'Sullivan *et al.*, 2012 Genome-wide association analysis and genetic architecture of egg weight and egg uniformity in layer chickens. Anim Genet 43 Suppl 1**:** 87-96.

Wu, M. C., P. Kraft, M. P. Epstein, D. M. Taylor, S. J. Chanock *et al.*, 2010 Powerful SNP-set analysis for case-control genome-wide association studies. Am J Hum Genet 86**:** 929-942.

Xu, S., 2007 An Empirical Bayes Method for Estimating Epistatic Effects of Quantitative Trait Loci. Biometrics 63**:** 513-521.

Yang, W., C. Chen, J. P. Steibel, C. W. Ernst, R. O. Bates *et al.*, 2015a A comparison of alternative random regression and reaction norm models for whole genome predictions. J Anim Sci 93**:** 2678-2692.

Yang, W., C. Chen and R. J. Tempelman, 2015b Improving the computational efficiency of fully Bayes inference and assessing the effect of misspecification of hyperparameters in whole-genome prediction models. Genetics, selection, evolution : GSE 47**:** 13.

Yang, W., and R. J. Tempelman, 2012 A Bayesian antedependence model for whole genome prediction. Genetics 190**:** 1491-1501.

Yi, N., and S. Xu, 2008 Bayesian LASSO for quantitative trait loci mapping. Genetics 179**:** 1045-1055.

Zhang, X., D. Lourenco, I. Aguilar, A. Legarra and I. Misztal, 2016 Weighting Strategies for Single-Step Genomic BLUP: An Iterative Approach for Accurate Calculation of GEBV and GWAS. Front Genet 7**:** 151.

Zhou, X., and M. Stephens, 2012 Genome-wide efficient mixed-model analysis for association studies. Nat Genet 44**:** 821-U136.

Zhu, B., M. Zhu, J. Jiang, H. Niu, Y. Wang *et al.*, 2016 The Impact of Variable Degrees of Freedom and Scale Parameters in Bayesian Methods for Genomic Prediction in Chinese Simmental Beef Cattle. PloS one 11**:** e0154118.

Zimmerman, D. L., and V. A. Nunez-Anton, 2010 Antedependence models for longitudinal data, pp. xvii, 270 p. in *Monographs on statistics and applied probability 112*. Chapman & Hall/CRC,, Boca Raton, FL.