BOSTON UNIVERSITY

GRADUATE SCHOOL OF ARTS AND SCIENCES

Dissertation

HIERARCHICAL BAYESIAN MODELS FOR GENOME-WIDE ASSOCIATION STUDIES

by

IAN JOHNSTON

Master of Arts in Mathematics, Boston University, 2013
Bachelor of Science in Mathematics, Drexel University, 2010

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

2015

Approved by

First Reader
Luis Carvalho, PhD
Assistant Professor

Second Reader
Josée Dupuis, PhD
Professor

Third Reader
Uri Eden, PhD
Associate Professor

Acknowledgments

For Dorothy.

HIERARCHICAL BAYESIAN MODELS FOR GENOME-WIDE ASSOCIATION STUDIES

(Order No.)

IAN JOHNSTON

Boston University, Graduate School of Arts and Sciences, 2015

Major Professor: Luis Carvalho, Assistant Professor

ABSTRACT

I consider a well-known problem in the field of statistical genetics called a genome-wide association study (GWAS) where the goal is to identify a set of genetic markers that are associated to a disease. A typical GWAS data set contains, for thousands of unrelated individuals, a set of hundreds of thousands of markers, a set of other covariates such as age, gender, smoking status and other risk factors, and a response variable that indicates the presence or absence of a particular disease. Due to biological phenomena such as the recombination of DNA and linkage disequilibrium, parents are more likely to pass parts of DNA that lie close to each other on a chromosome together to their offspring; this non-random association between adjacent markers leads to strong correlation between markers in GWAS data sets. As a statistician, I reduce the complex problem of GWAS to its essentials, i.e. variable selection on a large-p-small-n data set that exhibits multicollinearity, and develop solutions that complement and advance the current state-of-the-art methods. Before outlining and explaining my contributions to the field in detail, I present a literature review that summarizes the history of GWAS and the relevant tools and techniques that researchers have developed over the years for this problem.

Contents

List of Symbols

y	an $n \times 1$ vector of response variables
X	an $n \times p$ matrix of genotypes (SNPs)
π_j	the minor allele frequency of the j^{th} SNP
V	a matrix of additional optional covariates
R^2	the squared correlation coefficient between two variables
ρ	a tuning parameter used by the CLUSTAG program
β_0	the baseline intercept term in a regression model
β_j	the effect size of the j^{th} SNP
η	the effect sizes of the covariates recorded in V in a regression model
ϵ_i	the residual term for the i^{th} individual in a regression model
τ^2_{pop}	the population variance in a regression model
H_0	the null hypothesis in a statistical test
H_a	the alternative hypothesis in a statistical test
α_{sig}	the significance level in a statistical test
β	a $p \times 1$ vector of effect sizes of SNPs
$L(y; \beta, \tau^2_{\text{pop}})$	the likelihood function, or joint probability function, for y
$\exp(\cdot)$	the exponential function
$\hat{\beta}_{\text{OLS}}$	the ordinary least squares estimator of β
$P(\beta)$	a function that specifies a penalty on the values of β
$\tilde{\beta}$	the estimator of β in a penalized regression model
$\lvert\lvert\beta\rvert\rvert_k$	the L^k-norm of β
$\tilde{\beta}_{\text{Ridge}}$	the estimator of β in the Ridge penalized model

$\tilde{\beta}_{\text{LASSO}}$ .. the estimator of β in the LASSO penalized model

I_p .... the estimator of β in a penalized regression model

υ_0 .... a tuning parameter used in a penalized regression model

υ_1 .... a tuning parameter used in the Ridge penalized model

υ_2 .... a tuning parameter used in the LASSO penalized model

υ_g .... a tuning parameter used in the group LASSO penalized model

υ_f .... a tuning parameter used the fused LASSO penalized model

$\mathbb{P}(\beta|y)$ .. the posterior probability density function for β given y

$\propto$ ..... a symbol used to indicate that two values are proportional

$\mathbb{P}(y; \beta, \tau^2_{\text{pop}})$ the likelihood of the data y for given values of β and τ^2_{pop}

$\mathbb{P}(\beta)$ ... the prior probability density function for β

$\hat{\beta}_{\text{MAP}}$ ... the maximum a posterior estimator for β

$\log(\cdot)$ ... the natural logarthmic function

$\arg\max\limits_{\beta}$ .. the argument of the maximum of a function of β

$\arg\min\limits_{\beta}$ .. the argument of the maximum of a function of β

θ ..... a set of p latent binary variables that indicate which SNPs are significantly associated to a given trait

$\delta_0(\cdot)$ ... the dirac delta function

κ ..... a constant that denotes the separation between the variance components of a continuous spike-and-slab prior distribution

α_{sp} .... the prior probability of association for each SNP in a simple Bayesian variable selection model

σ^2_{sas} .... a random variable that denotes the variance of the spike component in a continuous spike-and-slab prior distribution

$\mathbf{r}$ ..... a $G \times 1$ vector of gene relevances

$\mathbf{w}_j^\top$ .... a $G \times 1$ vector of arbitrary gene-SNP weights

ϕ ..... a range parameter used in the definition of $\mathbf{w}_j^\top$

s_j .... the genomic position of the j^{th} SNP

g_l ..... the starting position of a particular gene

g_r .... the ending position of a particular gene

$w_{j,g}$ .... the spatial weight afforded to the j^{th} SNP by the g^{th} gene

∞ .... a symbol used to denote the concept of positive infinity

$\mathbf{1}_G$ .... a $G \times 1$ vector of ones

Z ..... an $n \times p$ matrix of *latent* genotypes

α_{gp} .... the prior probability a gene being “active” in a hierarchical Bayesian model on simple latent genotypes

ξ_0 .... a tuning parameter to indirectly control the prior probability of association for all SNPs

ξ_1 .... a tuning parameter to control the scale of the gene boost effect in the prior distribution on each θ_j

$\mathbf{a}$ ..... a vector of binary variables that indicate whether or not a given gene is “active” in a Bayesian model on simple latent genotypes

σ^2 .... a random variable that denotes the variance of the spike component in a continuous spike-and-slab prior distribution

ν ..... the shape parameter in the prior distribution of σ^2

λ ..... the scale parameter in the prior distribution of σ^2

Chapter 1

Introduction

1.1 Literature Review

1.1.1 Background

Although there is no variation in virtually all of the DNA across humans, we do observe some locations along the chromosomes that vary from individual to individual due to random biological events such as mutations of parts of the genetic code that children receive from their parents. Researchers have postulated that these points of variation, also called single nucleotide polymorphisms (SNPs) or more recently single nucleotide variants (SNVs), genetic markers, or simply variants, could ultimately lead to errors in the coding of genes or other regions of the genome that may result in benign effects such as a change in eye color, but that also may result in malignant effects such as an increased risk of developing a particular disease [11]. Not long after scientists successfully sequenced the human genome at the beginning of the 21^{st} century [54, 83], researchers began to create public databases such as the international HapMap project to store information about SNPs [30]. As the throughput of genotype technologies have increased while cost has decreased, geneticists have already developed arrays that can identify over a million of an individual's unique configuration of SNPs by exploiting the biochemical principle that nucleotides bind to their complementary partners [53].

Genome-wide association studies (GWAS) [19] aim to solve the important problem in statistical genetics of first identifying SNPs associated with a particular disease and then using them to search the surrounding regions of the chromosome to possibly draw a connection to other points of interest such as genes that could be related to that disease. By fitting an appropriate statistical model to GWAS data that is capable of detecting causal SNPs, researchers may then work with geneticists and biologists to gain a better understanding of how a disease develops and even gain better insight on how to construct a cure. Since correlation does not imply causation, researchers typically use a multi-stage approach in their analysis of GWAS data and seek to validate the signals observed in one data set by trying to replicate their findings in an second, independent population sample; even if the signal can be replicated in an independent data set, a functional study is required before they can draw any specific conclusions about the mechanisms of a disease [39].

A typical GWAS data set contains a vector of n response variables, y, where y_i denotes the measured trait of interest for the i^{th} individual, and a matrix of SNP data, X, where $X_i^\top$ denotes the numbers of minor alleles (either 0, 1, or 2) present at each of p SNPs for the i^{th} individual. An allele in this context refers to one of the two nucleotides that a child inherits from his or her parents (either A, T, C, or G) and the minor allele of the j^{th} SNP is defined as the allele that occurs less frequently in the overall population with minor allele frequency (MAF) π_j. It is also common to include an additional matrix of covariates, V, that encode features such as the age, gender, and smoking status of each individual. Through a naturally occurring biological event called recombination, chromosomes may exchange genetic information with each other, for instance through the breaking apart and rejoining of DNA strands, to form novel combinations of alleles [2]. The probability of a recombination event occurring between any two points on a chromosome increases as a function of the distance between them and so two alleles that are relatively close to each other are more likely to remain together after recombination has taken place. After many generations of recombinations, the observed non-random association of SNPs at the population level is called linkage disequillibrium (LD) [5] and it results in patterns of correlation in the columns of X.

LD has pros and cons: it is possible to indirectly identify additional SNPs or genes by showing that they are in LD with a SNP [52, 48]; however, the multicollinearity in X leads not only to a violation of the independence assumption when performing single SNP tests of association but also to inefficient estimation of model parameters in multiple regression models [85]. A common strategy to account for the bias induced by multicollinearity when analyzing SNPs jointly is to replace a group of highly correlated SNPs with only one of its members [42]; for instance, researchers have developed many algorithms that explot known LD patterns in a given region of the genome to replace a block of correlated markers with a subset of markers called tag SNPs that are representative of that region's variation [45, 78, 86, 14]. For example, the CLUSTAG [4] method in particular uses hierarchical clustering and set-cover algorithms to obtain a set of tag SNPs that can represent all the known SNPs in a chromosomal region, subject to the constraint that all SNPs must have a squared correlation $R^2 > \rho$ with at least one tag SNP, where ρ is specified by the user. The developers of this method point out that when clustering SNPs for such a task, an appropriate measure of distance is $1 - R^2$ since the required sample size for a tag SNP to detect an indirect association with a disease is inversely proportional to the R^2 between the tag SNP and the causal SNP.

Before analyzing the statistical relationship between y and X (and V), researchers must inspect X for other potential sources of bias such as population stratification or genotyping errors. The problem known as population stratification refers to the situation where an imbalance in the relative numbers of healthy individuals and diseased individuals sampled from different populations that have different disease prevalances can lead to false-positives in association studies [39]. An ideal model for SNP data called the Hardy-Weinberg model or Hardy-Weinberg Equilibrium (HWE) [36] assumes, in simple terms, that alleles for the j^{th} SNP are passed independently from one generation to the next with *constant* minor allele frequency π_j so that $X_{ij} \sim \text{Binomial}(2, \pi_j)$. Significant deviations from HWE are one way in which researchers can detect problematic SNPs in GWAS data sets perhaps due to the aforementioned biases [89]. Principal component analysis, a technique that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables [47], has also been shown to be an effective tool in revealing the population structure in a GWAS data set and subsequently allowing for a correction for population stratification [73].

The standard per-marker quality control for a GWAS data set consists of at least four steps: (i) identification of SNPs with an excessive missing genotype, (ii) identification of SNPs showing a significant deviation from HWE, (iii) identification of SNPs with significantly different missing genotype rates between cases and controls and (iv) the removal of all rare variants, i.e. markers with a very low minor allele frequency (e.g. $\pi_j < 1\%$) [3]. Since the power to detect association with rare variants is low [65], they are usually removed from GWAS data sets; however, it is noteworthy that researchers have recently begun to consider specialized models that try boosting the power to detect association for instance by prioritizing rare variants in a given region in a way that exploits weights based on their MAFs [90] or by collapsing and summarizing rare variants in a region [55].

Presently the most popular method of analyzing GWAS data sets, at least in a prescreening step, is to write a computer program or to use a well-known, freely available tool set like PLINK [74] to first apply data quality control filters like the ones described above and then to fit a simple linear (or logistic) regression model for a quantitative (or qualitative) response variable, one SNP at a time, of the form:

$$y_i = \beta_0 + \beta_j X_{ij} + V_i^\top \eta + \epsilon_i, \quad \text{for} \quad i = 1, \ldots, n$$

where β_0 is a baseline intercept term, β_j denotes the effect size of the j^{th} SNP, η denote the effect sizes of the additional covariates in V and the residual term are assumed to be independent and identically distributed random variables such that $\epsilon_i \overset{\text{iid}}{\sim} \text{Normal}(0, \tau^2_{\text{pop}})$. After fitting each of the p models, researchers scan the genome for regions that may be associated to the trait by computing and plotting $-\log_{10}(\text{p-value}_j)$ where p-value$_j$ is the p-value of the following hypothesis:

$$H_0 : \beta_j = 0 \ \text{ vs. } \ H_a : \beta_j \neq 0$$

When conducting p hypothesis tests using the same response variable, there is an inflated risk of committing a Type I error, i.e. rejecting the null hypothesis by mistake due to the multiple comparisons problem [23]. Supposing a significance level of α_{sig} for each of the p *independent* single SNP tests, the probability of committing at least one Type I error becomes $1 - (1 - \alpha_{\text{sig}})^p$. A simple solution to the multiple comparisons problem is to define α_{sig} in such a way that controls an experiment or genome-wide significance level for instance by using a technique such as the Bonferroni correction [23]. This technique is fast and easy to implement; however, the resulting genome-wide significance level (α_{sig}/p) may be too conservative because it fails to take into account the patterns of correlation in typical GWAS data sets [68]. An ideal but unfortunately impractical solution would be to collect many more samples to make it possible to analyze the markers jointly. Recent approaches adopt a more practical stategy and aim at gaining more statistical power by pooling information across studies through meta-analysis [24].

For case-control GWAS where y is a binary vector that encodes the presence or absence of a disease for each individual, another popular but computationally intense solution to the multiple comparisons problem is to apply a permutation test and assess significance based on a distribution of the test statistics computed after permuting the case-control labels a large number of times. The permutation test is considered a gold standard in GWAS because it takes into account the correlation among SNPs caused by LD [27]. While some reseachers have worked on reducing the burden of those permuting computations by developing approximation algorithms or considering alternative frameworks [67], other researchers have made progress by considering calculating a genome-wide significance level based on the effective number of independent tests [16, 22].

There are currently many proposed models for GWAS [7] including but not limited to penalized regression approaches [91], approaches based on Bayesian variable selection [33], and machine learning methods [79]. The models that analyze SNPs independently or in blocks are computationally fast to fit but suffer from drawbacks such as the inflation in the probability of a Type I error and an oversimplification of the problem. On the other hand, the models that analyze all of the SNPs jointly are more representative but consequently computationally expensive to fit. Moreover, rather than analyze the SNPs first and then inspect the genome for candidate genes that might be related to a given trait of interest, some researchers have pointed out that a gene-centric approach that somehow better accounts for genes directly in the modeling procedure may be more appropriate for GWAS [49]. Researchers have recently begun incorporating external knowledge about genes and even molecular pathways [87] into models to group SNPs based on criteria such as gene membership, but there is no universal agreement on how to define such criteria. Although progress has been made on GWAS since the dawn of the 21st century, it is still a relevant and challenging problem with goals such as modeling interactions between SNPs, genes, and environment effects that await beyond the obstacles already mentioned [38].

1.1.2 Related Work

As a statistician I view the goal of GWAS as variable selection in a large-p-small-n data set that exhibits multicollinearity, and I build a representative model for analyzing SNPs jointly in a way that exploits external biological knowledge to not only de-correlate markers but to also prioritize markers that are close in genomic distance to relevant genes or other regions. In this section I expand the background on GWAS from Section 1.1.1 to include relevant work related to my research. As before letting y_i denote the response variable for the i^{th} individual, $X_i^\top$ denote a set of p corresponding covariates, e.g. SNPs, and ϵ_i denote a corresponding residual term, I define a multiple regression model by assuming that the residual terms are independent and identically distributed random variables such that $\epsilon_i \overset{\text{iid}}{\sim} \text{Normal}(0, \tau^2_{\text{pop}})$ and by setting

$$y_i = X_i^\top \beta + \epsilon_i, \text{ for } i = 1, \ldots, n.$$

This is equivalent to assuming that the y_i's are independent random variables such that $y_i \overset{\text{ind}}{\sim} \text{Normal}(X_i^\top \beta, \tau^2_{\text{pop}})$, and thus I can write the likelihood of the observed data, $L(y; \cdot)$

in terms of β and τ^2_{pop} as follows:

$$L(y;\beta,\tau^2_{\text{pop}}) \propto \exp\left[-\frac{1}{2\tau^2_{\text{pop}}}\sum_{i=1}^{n}(y_i - X_i^\top\beta)^2\right]$$

A well-known result in statistics is the maximum likelihood, or ordinary least squares, solution for β in the multiple regression model [66]: $\hat{\beta}_{\text{OLS}} = (X^\top X)^{-1}X^\top y$. This estimator maximizes $L(y;\beta,\tau^2_{\text{pop}})$ and we can normally use it and its asymptotic distribution to conduct inference about β; however, in the case of GWAS we cannot because $p > n$ and so, for instance, it is impossible to compute $(X^\top X)^{-1}$ and subsequently $\hat{\beta}_{\text{OLS}}$. To overcome this so-called large-p-small-n problem, it is common to use a penalized regression model such as ridge regression [40], least absolute shrinkage and selection operator (LASSO) [81], or the elastic net [95] instead. In penalized regression models we regularize β by adding a penalty term, $P(\beta)$, so that it is possible to obtain a unique solution when minimizing a new objective function:

$$\tilde{\beta} = \arg\min_{\beta}\left[-\log(L(y;\beta,\tau^2_{\text{pop}})) + P(\beta)\right]$$

Letting $||\beta||_k = (\sum_{j=1}^{p}|\beta|^k)^{\frac{1}{k}}$ denote the L^k-norm of β, we have the ridge regression model when $P(\beta) = \upsilon_0||\beta||_2^2$. Adding this penalty translates into adding a "ridge" to the diagonal of $X^\top X$; letting I_p denote an identity matrix of size p, we have for the ridge regression model: $\tilde{\beta}_{\text{Ridge}} = (X^\top X + \upsilon_0 I_p)^{-1}X^\top y$. This simple regularization guarantees the existence of a unique $\tilde{\beta}$ that depends on the choice of the tuning parameter υ. Researchers typically use cross-validation to find an optimal value of υ_0 for a given data set [31]; larger values of υ_0 result in a more stringent regularization that shrinks the magnitude of $\tilde{\beta}_{\text{Ridge}}$.

Noting that genome-wide ridge regression may detect SNPs missed by univariate methods by incorporating multi-SNP dependencies in the model [50], researchers have successfully applied models based on ridge regression to GWAS data sets [77]. While ridge regression is certainly a useful tool for overcoming the large-p-small-n problem, a penalty on the L^2-norm does not allow us to automatically perform variable selection. The LASSO model penalizes the L^1-norm of β instead, i.e. $P(\beta) = \upsilon_0||\beta||_1$, and by doing so gains the ability to shrink elements of $\tilde{\beta}$ exactly to zero; however, since it produces biased estimates of the model parameters and tends to only select one parameter in a group of correlated parameters [95], it is not ideal for GWAS. The elastic net which strikes a compromise between ridge regression and LASSO by using a linear combination of the L^1 and the L^2 norms as a penalty term, i.e. $P(\beta) = \upsilon_1||\beta||_1 + \upsilon_2||\beta||_2^2$, has been found to be effective [17].

The choice of penalty term affects the computational complexity of the algorithm needed to obtain $\tilde{\beta}$ as well as the properties of $\tilde{\beta}$ [25]. Figure 1.1 compares these first three penalty functions and in particular shows the flexibility afforded by the elastic net penalty. It is possible to derive $\tilde{\beta}_{\text{Ridge}}$ analytically; however, as mentioned above, although the ridge penalty can shrink the magnitude of $\tilde{\beta}$, it cannot shrink its values exactly to zero. On the other hand, although the LASSO penalty can perform variable selection by shrinking some elements of $\tilde{\beta}$ exactly to zero, $\tilde{\beta}_{\text{LASSO}}$ can only be obtained by solving a convex optimization problem wherein it can select at most n variables before it saturates [95]. By combining these penalties, the elastic net overcomes these shortcomings by first performing a ridge-like shrinkage and then a LASSO-like selection of $\tilde{\beta}$.

Although these penalized regression models have been proposed and have been shown to outperform the popular single SNP tests for GWAS [6], fitting them in practice to a large data set (e.g. $p \geq 100{,}000$) is computationally intense and thus so is the process of selecting an optimal value for any tuning parameters. Side-stepping this problem, researchers who have developed a unified framework for penalized multiple regression analysis of GWAS data (PUMA) have had success in applying a suite of penalty terms (e.g. LASSO, NEG, MCP, LOG) to a pre-screened subset of markers and investigating the concordance of markers across each of the final models [41]. The penalty terms used in PUMA primarily differ in the rate at which their derivatives trail off to zero as illustrated in Figure 1.2. Although a pre-screening of markers from a marginal regression would ideally retain almost all of the relevant variables, researchers have found that penalized models such as LASSO could likely be improved by using a larger number of SNPs than those which pass an initial

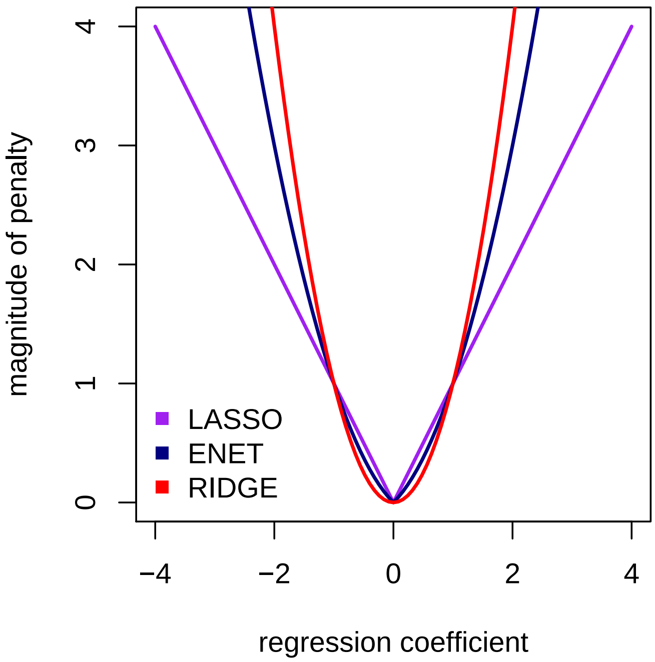

Figure 1.1: Comparison of the different penalty terms based on the L^1 and L^2 norms.

screening step (e.g. a genome-wide significance threshold) [51].

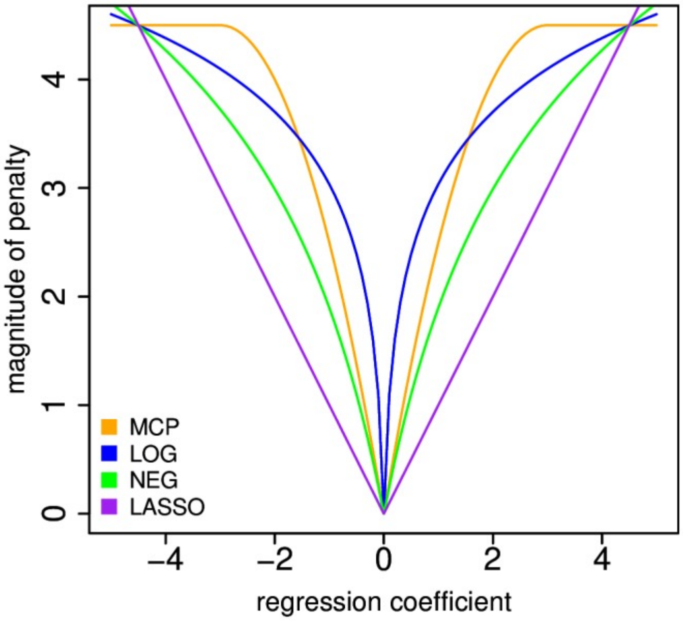

Figure 1.2: Comparison of the different penalty terms used in the PUMA software [41].

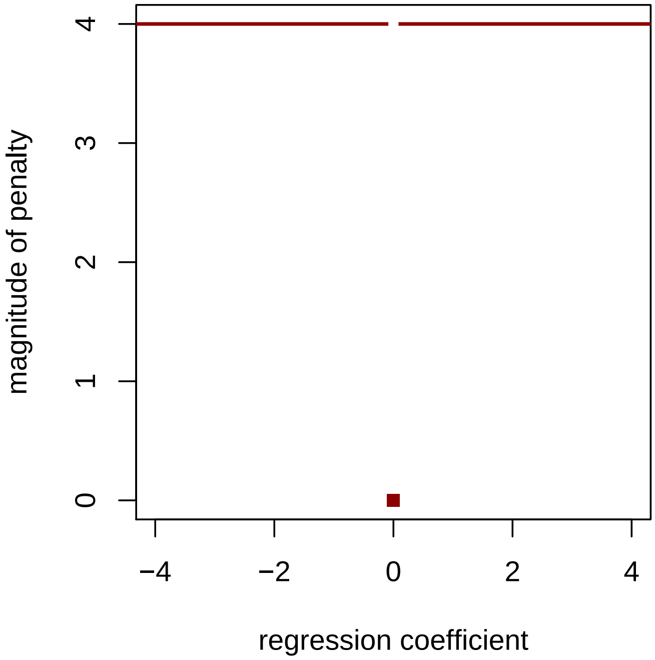

Figure 1.3: A possible ideal penalty for regression terms.

Under the simplifying assumption that an insignificant regression coefficient's true value is zero, the penalty functions considered so far take on low values for regression coefficients close to zero and high values for regression coefficients far from zero. Moreover, their tuning parameters control how quickly the magntiudes of the penalties increase as the magnitudes of the regression coefficients increase. Although only applicable for the truly insignificant regression terms, Figure 1.3 depicts one example of an ideal penalty function that takes on the value zero when the regression coefficient is zero and a "large" value everywhere else. By choosing a sufficiently large value for such a penalty function, e.g. ∞ in the most extreme case, $\tilde{\beta}$ would be forced to take on the value of zero for each penalized element. This infinite-leap penalty function is useless in practice; however, it illustrates an important point: a uniform penalty function for all regression coefficients is not always appropriate.

Researchers have explored group penalties that move beyond the uniform penalties such as ridge and LASSO for more flexibility in GWAS analyses. Techniques like group LASSO, fused LASSO [82], or sparse group LASSO [26] further attempt to account for the structure of genes and markers or LD by assigning SNPs to groups based on criteria such as gene membership and then placing additional penalties on the L_1 norm of the vector of coefficients for each group or on the L_1 norm of the difference in coefficients of consecutive SNPs. Given G groupings of SNPs indexed by g, we can use these models to penalize each group separately using g tuning parameters: $P(\beta) = \sum_{g=1}^{G} \upsilon_g \left[\sum_{k \in g} |\beta_k|\right]$; moreover we can consider fusing together adjacent SNPs through a penalty of the form: $P(\beta) = \upsilon_f \sum_{j=1}^{p-1} |\beta_{j+1} - \beta_j|$. Although these relatively more complicated penalties allow us to incorporate external biological knowledge into our models, it is difficult to define gene membership universally since genes have varying lengths and may overlap with each other; moreover, the penalty on the L_1 norm of the difference in consecutive SNPs neglects any information contained in the genomic distance between them. Some researchers have further pointed out that the ambiguity in the choice of the reference allele for scoring genotypes makes the fused LASSO not applicable [59].

It may be possible to develop additional, effective penalty terms within models, such as ones based on the L_1 and L_2 norms, to address the issues present in GWAS data in a penalized regression framework, but because genotypes are more correlated for markers that are close in genomic space due to LD, the most effective penalties would need to capture the relevance of a particular SNP as a function of its location on the genome. Moreover, since it is typically easier to study the biological function of larger substructures of the DNA, e.g. genes, we are particularly interested in SNPs that lie close to relevant features of a chromosome [49]; as a result, the most desirable penalties would likely be SNP-specific. Researchers have accomplished this by setting SNP-specific prior distributions on the model parameters in a hierarchical Bayesian model [56]. Since the fundamental principle of Bayesian statistics is to incorporate prior knowledge when fitting model parameters, it is natural to exploit knowledge from biology about the structure of the genome through prior distributions on the model parameters in a Bayesian framework.

In fact the optimal solutions to penalized regression models can be shown to be equivalent, from a Bayesian perspective, to maximum a posteriori (MAP) estimators under appropriate prior specifications. For instance, for LASSO, the L_1 penalty with tuning parameter υ_0 can be translated into an independent Laplace prior distribution for each β_j with location parameter equal to zero and scale parameter equal to υ_0^{-1}, i.e. $\mathbb{P}(\beta) \propto \exp(-\upsilon_0 \sum_{j=1}^{p} |\beta_j|)$. Noting that $L(y;\beta,\tau^2_{\text{pop}}) = \mathbb{P}(y;\beta,\tau^2_{\text{pop}})$, and assuming that τ^2_{pop} is a known constant for simplicity, we have that the posterior distribution of β given the observed data y becomes:

$$\begin{aligned}
\mathbb{P}(\beta|y) &\propto \mathbb{P}(y;\beta,\tau^2_{\text{pop}}) \times \mathbb{P}(\beta) \\
&\propto \exp\left[-\frac{1}{2\tau^2_{\text{pop}}}\sum_{i=1}^{n}(y_i - X_i^\top\beta)^2 - \upsilon_0\sum_{j=1}^{p}|\beta_j|\right]
\end{aligned}$$

Letting $\hat{\beta}_{\text{MAP}}$ denote the MAP estimator for this distribution, and recalling that for LASSO, $P(\beta) = \upsilon_0||\beta||_1$, we note that since $\log(\cdot)$ is a monotonic function,

$$\begin{aligned}
\hat{\beta}_{\text{MAP}} &= \arg\max_{\beta} \mathbb{P}(\beta|y) \\
&= \arg\max_{\beta} \log\left[\mathbb{P}(\beta|y)\right] \\
&= \arg\max_{\beta} \left[-\frac{1}{2\tau^2_{\text{pop}}}\sum_{i=1}^{n}(y_i - X_i^\top\beta)^2 - \upsilon_0\sum_{j=1}^{p}|\beta_j|\right] \\
&= \arg\min_{\beta} \left[\frac{1}{2\tau^2_{\text{pop}}}\sum_{i=1}^{n}(y_i - X_i^\top\beta)^2 + \upsilon_0||\beta||_1\right] \\
&= \arg\min_{\beta} \left[-\log(L(y;\beta,\tau^2_{\text{pop}})) + P(\beta)\right] \\
&= \tilde{\beta}_{\text{LASSO}}
\end{aligned}$$

In a similar fashion we can establish a link between other penalized regression models, e.g. ridge, and corresponding Bayesian models, e.g. an independent Normal prior distribution for each β_j with mean parameter equal to zero and variance parameter equal to $2/\upsilon_0$ so that $\mathbb{P}(\beta) \propto \exp\left(\upsilon_0 \sum_{j=1}^{p} \beta_j^2\right)$. Moreover, by using a hierarchical Bayesian model we can allow the prior distribution for each β_j to vary based on our prior belief of whether or not the j^{th} SNP is causal. In particular, for Bayesian variable selection we most commonly use the spike-and-slab prior distribution [29] for β_j which is a mixture between a point mass at zero (the spike) and a normal distribution centered at zero with a large variance (the slab). The latent binary variable θ_j indicates the association status of the j^{th} SNP and thus determines the component of the mixture used to define the prior distribution of β_j. Given a tuning parameter to control the population variance of a quantitative trait, τ^2_{pop}, a tuning parameter to control the variance of the slab component of the prior distribution, σ^2_{sas}, a prior probability of association for each SNP, α_{sp}, we can write this model as follows:

$$\begin{aligned} y_i \mid X_i^\top \beta &\overset{\text{ind}}{\sim} \text{Normal}(X_i^\top \beta, \tau^2_{\text{pop}}) \\ \beta_j \mid \theta_j &\overset{\text{ind}}{\sim} (1-\theta_j)\cdot\delta_0(\cdot) + \theta_j \cdot \text{Normal}(0, \sigma^2_{\text{sas}}) \\ \theta_j &\overset{\text{iid}}{\sim} \text{Bernoulli}(\alpha_{\text{sp}}) \end{aligned} \tag{1.1}$$

A relaxed version of this model replaces the mixture of the dirac delta function, $\delta_0(\cdot)$, and a normal distribution with a mixture of two normal distributions centered at zero with different variances. This continuous version of the spike-and-slab model trades the ability to perform exact variable selection through $\delta_0(\cdot)$ for computational convenience. By introducing another tuning parameter to denote the separation between the variances of the normal distributions, κ, we can write this updated model as follows:

$$\begin{aligned} y_i \mid X_i^\top \beta &\overset{\text{ind}}{\sim} \text{Normal}(X_i^\top \beta, \tau^2_{\text{pop}}) \\ \beta_j \mid \theta_j &\overset{\text{ind}}{\sim} \text{Normal}(0, \sigma^2_{\text{sas}}[\theta_j\kappa + 1 - \theta_j]) \\ \theta_j &\overset{\text{iid}}{\sim} \text{Bernoulli}(\alpha_{\text{sp}}) \end{aligned} \tag{1.2}$$

Recalling the connection between the ridge regression penalty and the normal prior distribution, Figure 1.4 shows an example of the different possible penalty functions for

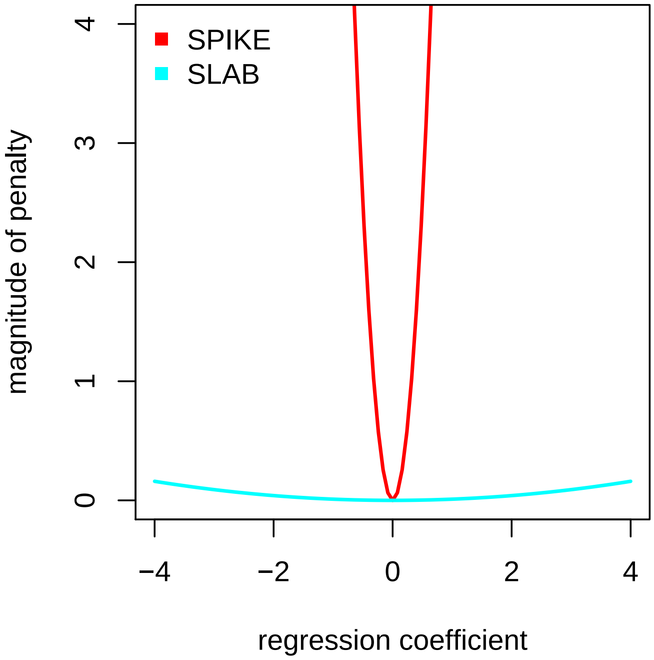

Figure 1.4: Possible penalties in the spike-and-slab model for Bayesian variable selection.

β_j in the continuous spike-and-slab model of 1.2. Through the perspective of penalized regression, when we compute the MAP estimator of β in the continuous spike-and-slab model, the latent variable, θ_j, offers the model the flexibility to select either the spike or the slab penalty to use for β_j. This is an important step towards an ideal penalty for variable selection because now we can encode a prior belief on how many SNPs we expect to be associated with a trait in α_{sp} and then dynamically place strong penalties on coefficients that we deem to be not associated ($\theta_j = 0$), and less stringent penalities on those that we deem to be associated ($\theta_j = 1$). Perhaps the real power of this method lies in the fact that the hierarchy can be extended even further to allow α_{sp} to vary based on each SNP.

In state-of-the-art Bayesian variable selection models for GWAS and other large scale problems, researchers have considered exactly this type of hierarchical Bayesian model (e.g. [33, 94]). For instance, some recent models exploit Bayesian methods in particular to allow for data-driven SNP-specific prior distributions [34] which depend on a random variable that describes the proportion of SNPs to be selected. These approaches have adopted a continuous spike-and-slab prior distribution [43] on the model parameters, set an inverse Gamma prior distribution on the variance of the spike component of the prior, and control the difference in the variance of the spike and slab components of the prior using a tuning parameter. My potential for a creative improvement in these models lies in the specification of the prior distribution on the θ_j's.

Rather than use a uniform α_{sp} for each θ_j, I develop methods to incorporate external knowledge about a SNP's location on its chromosome relative to other important features to define its unique prior probability of association. I have previously introduced a hierarchical gene model on latent genotypes for GWAS [46] that estimates simple latent genotypes, Z, from X and then models each y_i in terms of $Z_i^\top$ based on the following hierarchy:

$$\begin{aligned} y_i \mid Z_i^\top \beta &\overset{\text{ind}}{\sim} \text{Normal}(Z_i^\top \beta, \tau_{\text{pop}}^2) \\ \beta_j \mid \theta_j &\overset{\text{ind}}{\sim} (1-\theta_j)\cdot\delta_0(\cdot) + \theta_j \cdot \text{Normal}(0, \sigma_{\text{sas}}^2) \\ \theta_j &\overset{\text{ind}}{\sim} \text{Bernoulli}(\xi_0 + \xi_1 \mathbf{w}_j^\top \mathbf{a}) \\ \mathbf{a}_g &\overset{\text{iid}}{\sim} \text{Bernoulli}(\alpha_{\text{gp}}) \end{aligned} \tag{1.3}$$

where $\mathbf{w}_j^\top$ is a vector of G binary variables that, for each g of G genes, indicate whether or not the j^{th} SNP lies inside the g^{th} gene, $\mathbf{a}$ is a complementary vector of G binary variables that indicate whether or not each gene is "active", i.e. plays an active biological role in determining the outcome of the response variable, α_{gp} is now a tuning parameter that controls the prior probability of a gene being active, ξ_1 is a tuning parameter that controls the scale of the boost awarded to the j^{th} SNP's prior probability of association due to its location on the genome relative to active genes, and ξ_0 is a tuning parameter that controls the prior probability of association for all SNPs that do not lie inside any active genes. Although this model exploits knowledge about the structure of the genome in a way that makes it useful for selecting not only SNPs but also genes that may be linked to a quantitative trait of interest, it is computationally intense to fit using a Gibbs sampler.

Seeking to incorporate external information in a hierarchical Bayesian model in a similar way, other researchers analyzing a different kind of data, gene expression levels, have recently considered relating a linear combination of a set of predictor-level covariates that quantify the relationships between the genes to their prior probabilities of association through a probit link function [69]. This formulation leads to a second-stage probit regression on the probability that any gene is associated with a trait of interest using a set of predictor-level covariates that could be, for instance, indicator variables of molecular pathway membership. With respect to 1.3, this is akin to letting $\mathbf{w}_j^\top$ be a vector of abitrary covariates that encode various features, e.g. indicators of structural or functional properties, about the j^{th} SNP and letting $\mathbf{a}$ be their corresponding effect sizes. In an updated variable selection model I propose considering a special case of this formulation tailored for GWAS data where: (i) I use the logit link instead of the probit link, (ii) the predictor-level covariates are spatial weights that quantify a SNP's position on the genome relative to neighboring genes, and (iii) the coefficients of each of the predictor-level covariates are numerical scores that quantify the relevance of a particular gene to the trait of interest.

1.1.3 Outline of Contributions

In order to help move towards a unifying framework for GWAS that allows for the large-p-small-n problem and the SNP-specific issues to be addressed simultaneously in a principled manner, I propose a hierarchical Bayesian model that exploits spatial relationships on the genome to define SNP-specific prior distributions on regression parameters. More specifically, while drawing inspiration from the increased flexibility in the proposed priors for θ_j with an eye toward computational efficiency, in my proposed setting I model markers jointly, but I explore a variable selection approach that uses marker proximity to relevant genomic regions, such as genes, to help identify associated SNPs. My contributions are:

1. I exploit a simultaneous auto-regressive (SAR) model [75] in a data pre-processing step to replace short contiguous blocks of correlated markers with block-wise inde-

pendent latent genotypes for subsequent analyses.

2. I focus on binary traits which are common to GWAS, e.g., case control studies, but more difficult to model due to lack of conjugacy. To circumvent the need for a Metropolis-Hastings step when sampling from the posterior distribution on model parameters, I use a recently proposed data augmentation strategy for logistic regression based on latent Pólya-Gamma random variables [71].

3. I perform variable selection by adopting a spike-and-slab prior [29, 43] and propose a principled way to control the separation between the spike and slab components using a Bayesian false discovery rate similar to [88].

4. I use a novel weighting scheme to establish a relationship between SNPs and genomic regions and allow for SNP-specific prior distributions on the model parameters such that the prior probability of association for each SNP is a function of its location on the chromosome relative to neighboring regions. Moreover, I allow for the "relevance" of a genomic region to contribute to the effect it has on its neighboring SNPs and consider "relevance" values calculated based on previous GWAS results in the literature, e.g. see [61].

5. Before sampling from the posterior space using Gibbs sampling, I use an expectation-maximization [EM, [21]] algorithm in a filtering step to reduce the number of candidate markers in a manner akin to distilled sensing [37]. By investigating the update equations for the EM algorithm, I suggest meaningful values to tune the hyperprior parameters of my model and illustrate the induced relationship between SNPs and genomic regions.

6. I derive a more flexible centroid estimator [15] for SNP associations that is parameterized by a sensitivity-specificity trade-off. I discuss the relation between this parameter and the prior specification when obtaining estimates of model parameters.

I present my hierarchical Bayesian model for GWAS, the spatial boost model, in Chapter 2, and briefly follow-up with an extension to quantitative traits in Chapter 3. I present my SAR model for de-correlating SNPs in Chapter 4. In the final chapter of my thesis, Chapter 5, I combine and extend the models from the preceeding chapters and present an application to two binary traits.

Chapter 2

Spatial Boost Model

Motivated by the important problem of detecting association between genetic markers and binary traits in genome-wide association studies, in this chapter I present a novel Bayesian model that establishes a hierarchy between markers and genes by defining weights according to gene lengths and distances from genes to markers. The proposed hierarchical model uses these weights to define unique prior probabilities of association for markers based on their proximities to genes that are believed to be relevant to the trait of interest. I use an expectation-maximization algorithm in a filtering step to first reduce the dimensionality of the data and then sample from the posterior distribution of the model parameters to estimate posterior probabilities of association for the markers. I offer practical and meaningful guidelines for the selection of the model tuning parameters and propose a pipeline that exploits a singular value decomposition on the raw data to make my model run efficiently on large data sets. I demonstrate the performance of the model in simulation studies and conclude by discussing the results of a case study using a real-world dataset provided by the Wellcome Trust Case Control Consortium (WTCCC).

2.1 Model Definition

I perform Bayesian variable selection by analyzing binary traits and using the structure of the genome to dynamically define the prior probabilities of association for the SNPs. My data are the binary responses $y \in \{0,1\}^n$ for n individuals and genotypes $X_i \in \{0,1,2\}^p$ for p markers per individual, where x_{ij} codes the number of *minor* alleles in the i-th individual for the j-th marker. For the likelihood of the data, I consider the logistic regression:

$$y_i \mid X_i, \beta \overset{\text{ind}}{\sim} \mathsf{Bernoulli}\big(\text{logit}^{-1}(\beta_0 + X_i^\top \beta)\big), \quad \text{for } i = 1, \ldots, n. \tag{2.1}$$

I note that GWA studies are usually *retrospective*, i.e. cases and controls are selected irrespectively of their history or genotypes; however, as [62] point out, coefficient estimates for β are not affected by the sampling design under a logistic regression. Thus, from now on, to alleviate the notation I extend X_i to incorporate the intercept, $X_i = (x_{i0} = 1, x_{i1}, \ldots, x_{ip})$, and also set $\beta = (\beta_0, \beta_1, \ldots, \beta_p)$.

I use latent variables $\theta \in \{0, 1\}^p$ and a continuous spike-and-slab prior distribution for the model parameters with the positive constant $\kappa > 1$ denoting the separation between the variance of the spike and the slab components:

$$\beta_j \mid \theta_j, \sigma^2 \overset{\text{ind}}{\sim} \mathsf{Normal}\big(0, \sigma^2[\theta_j \kappa + (1 - \theta_j)]\big), \quad \text{for } j = 1, \ldots, p. \tag{2.2}$$

For the intercept, I set $\beta_0 \sim \text{Normal}(0, \sigma^2\kappa)$ or, equivalently, I define $\theta_0 = 1$ and include $j = 0$ in (2.2). In the standard spike-and-slab prior distribution the slab component is a normal distribution centered at zero with a large variance and the spike component is a point mass at zero. This results in exact variable selection through the use of the θ_j's, because $\theta_j = 0$ would imply that the j-th SNP coefficient is exactly equal to zero. I use the continuous version of the spike-and-slab distribution to allow for a relaxed form of this variable selection that lends itself easily to an EM algorithm (see Section 2.2.1).

For the variance σ^2 of the spike component in (2.2) I adopt an inverse Gamma (IG) prior distribution, $\sigma^2 \sim \mathsf{IG}(\nu, \lambda)$. I expect σ^2 to be reasonably small with high probability in order to enforce the desired regularization that distinguishes associated markers from non-associated markers. Thus, I recommend choosing ν and λ so that the prior expected value of σ^2 is small.

In the prior distribution for θ_j, I incorporate information from relevant genomic regions.

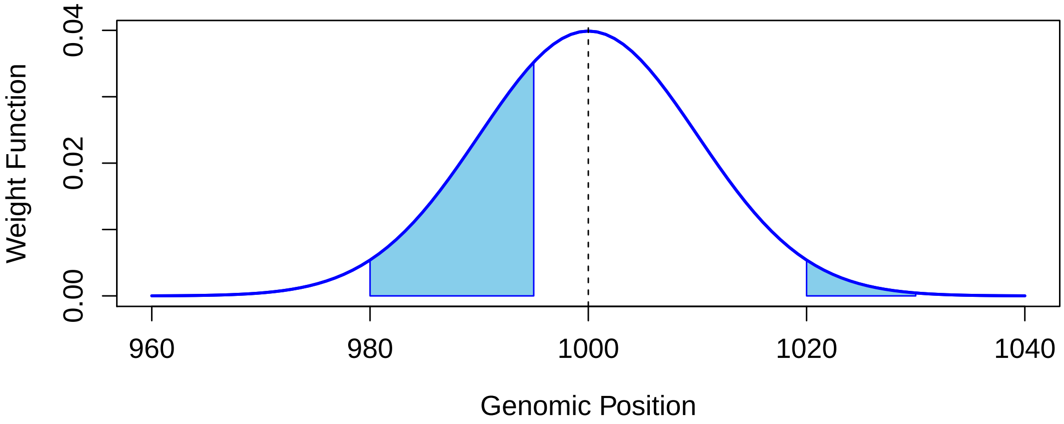

Figure 2.1: Gene weight example: for the j-th SNP at position $s_j = 1{,}000$ and two surrounding genes a and b spanning $(980, 995)$ and $(1020, 1030)$ I obtain, if setting $\phi = 10$, weights (areas shaded in blue) of $w_{j,a} = 0.29$ and $w_{j,b} = 0.02$, respectively.

The most common instance of such regions are *genes*, and so I focus on these regions in what follows. Thus, given a list of G genes with gene *relevances* (see Section 2.1.2 for some choices of definitions), $\mathbf{r} = [r_1, r_2, \ldots, r_G]$, and weights, $\mathbf{w}_j(\phi) = [w_{j,1}, w_{j,2}, \ldots, w_{j,G}]$, the prior on θ_j is

$$\theta_j \overset{\text{ind}}{\sim} \mathsf{Bernoulli}\big(\text{logit}^{-1}(\xi_0 + \xi_1 \mathbf{w}_j(\phi)^\top \mathbf{r})\big), \quad \text{for } j = 1, \ldots, p. \tag{2.3}$$

The weights $\mathbf{w}_j$ are defined using the structure of the SNPs and genes and aim to account for gene lengths and their proximity to markers as a function of a spatial parameter ϕ, as I see in more detail next.

2.1.1 Gene Weights

To control how much a gene can contribute to the prior probability of association for a SNP based on the gene length and the distance of the gene boundaries to that SNP I introduce a *range* parameter $\phi > 0$. Consider a gene g that spans genomic positions g_l to g_r, and the

j-th marker at genomic position s_j; the gene weight $w_{j,g}$ is then

$$w_{j,g} = \int_{g_l}^{g_r} \frac{1}{\sqrt{2\pi\phi^2}} \exp\left\{ -\frac{(x-s_j)^2}{2\phi^2} \right\} \mathrm{d}x.$$

Generating gene weights for a particular SNP is equivalent to centering a Gaussian curve at that SNP's position on the genome with standard deviation equal to ϕ and computing the area under that curve between the start and end points of each gene. Figure 2.1 shows an example. As $\phi \to 0$, the weight that each gene contributes to a particular SNP becomes an indicator function for whether or not it covers that SNP; as $\phi \to \infty$, the weights decay to zero. Intermediate values of ϕ allow then for a variety of weights in $[0,1]$ that encode *spatial* information about gene lengths and gene proximities to SNPs. In Section 2.3.1 I discuss a method to select ϕ.

According to (2.3), it might be possible for multiple, possibly overlapping, genes that are proximal to SNP j to boost θ_j. To avoid this effect, I take two precautions. First, I break genes into non-overlapping genomic blocks and define the relevance of a block as the mean gene relevance of all genes that cover the block. Second, I normalize the gene weight contributions to θ_j in (2.3), $\mathbf{w}_j(\phi)^\top \mathbf{r}$, such that $\max_j \mathbf{w}_j(\phi)^\top \mathbf{r} = 1$. This way, it is possible to compare estimates of ξ_1 across different gene weight and relevance schemes. It is also possible to break genes into their natural substructures, e.g. exons, introns, and to prioritize these substructures differently through the use of $\mathbf{r}$.

2.1.2 Gene Relevances

I allow for the further strengthening or diminishing of particular gene weights using gene relevances $\mathbf{r}$. If I set $\mathbf{r} = \mathbf{1}_G$ and allow for all genes to be uniformly relevant, then I have a "non-informative" case. Alternatively, if I have some reason to believe that certain genes are more relevant to a particular trait than others, for instance on the basis of previous research or prior knowledge from an expert, then I can encode these beliefs through $\mathbf{r}$. In particular, I recommend using either text-mining techniques, e.g. [1], to quantify the relevance of a gene to a particular disease based on citation counts in the literature, or relevance scores compiled from search hits and citation linking the trait of interest to genes, e.g. [61].

2.2 Model Fitting and Inference

The ultimate goal of my model is to perform inference on the posterior probability of association for SNPs. However, these probabilities are not available in closed form, and so I must resort to Markov chain Monte Carlo techniques such as Gibbs sampling to draw samples from the posterior distributions of the model parameters and use them to estimate $\mathbb{P}(\theta_j = 1 \,|\, y)$. Unfortunately, these techniques can be slow to iterate and converge, especially when the number of model parameters is large [20]. Thus, to make my model more computationally feasible, I propose first filtering out markers to reduce the size of the original dataset in a strategy similar to distilled sensing [37], and then applying a Gibbs sampler to only the remaining SNPs.

To this end, I design an EM algorithm based on the hierarchical model above that uses all SNP data simultaneously to quickly find an approximate mode of the posterior distribution on β and σ^2 while regarding θ as missing data. Then, for the filtering step, I iterate between (1) removing a fraction of the markers that have the lowest conditional probabilities of association and (2) refitting using the EM procedure until the predictions of the filtered model degrade. In my analyses I filtered 25% of the markers at each iteration to arrive at estimates β^* and stopped if $\max_i |y_i - \text{logit}^{-1}(X_i^\top \beta^*)| > 0.5$. Next, I discuss the EM algorithm and the Gibbs sampler, and offer guidelines for selecting the other parameters of the model in Section 2.3.

2.2.1 EM algorithm

I treat θ as a latent parameter and build an EM algorithm accordingly. If $\ell(y, \theta, \beta, \sigma^2) = \log \mathbb{P}(y, \theta, \beta, \sigma^2)$ then for the M-steps on β and σ^2 I maximize the expected log joint $Q(\beta, \sigma^2; \beta^{(t)}, (\sigma^2)^{(t)}) = \mathbb{E}_{\theta \,|\, y, X; \beta^{(t)}, (\sigma^2)^{(t)}}[\ell(y, \theta, \beta, \sigma^2)]$. The log joint distribution ℓ, up to a normalizing constant, is

$$\ell(y, \theta, \beta, \sigma^2) = \sum_{i=1}^{n} y_i X_i^\top \beta - \log(1 + \exp\{X_i^\top \beta\}) \\ - \frac{p+1}{2} \log \sigma^2 - \frac{1}{2\sigma^2} \sum_{j=0}^{p} \beta_j^2 \left(\frac{\theta_j}{\kappa} + 1 - \theta_j \right) - (\nu + 1) \log \sigma^2 - \frac{\lambda}{\sigma^2}, \quad (2.4)$$

and so, at the t-th iteration of the procedure, for the E-step I just need to compute and store $\langle \theta_j \rangle^{(t)} \doteq \mathbb{E}_{\theta \,|\, y; \beta^{(t)}, (\sigma^2)^{(t)}}[\theta_j]$. But since

$$\langle \theta_j \rangle = \mathbb{P}(\theta_j = 1 \,|\, y, \beta, \sigma^2) = \frac{\mathbb{P}(\theta_j = 1, \beta_j \,|\, \sigma^2)}{\mathbb{P}(\theta_j = 0, \beta_j \,|\, \sigma^2) + \mathbb{P}(\theta_j = 1, \beta_j \,|\, \sigma^2)},$$

then

$$\operatorname{logit}\langle \theta_j \rangle = \log \frac{\mathbb{P}(\theta_j = 1, \beta_j \,|\, \sigma^2)}{\mathbb{P}(\theta_j = 0, \beta_j \,|\, \sigma^2)} = -\frac{1}{2} \log \kappa - \frac{\beta_j^2}{2\sigma^2} \left(\frac{1}{\kappa} - 1 \right) + \xi_0 + \xi_1 \mathbf{w}_j^\top \mathbf{r} \quad (2.5)$$

for $j = 1, \ldots, p$ and $\langle \theta_0 \rangle \doteq 1$.

To update β and σ^2 I employ conditional maximization steps [63], similar to cyclic gradient descent. From (2.4) I see that the update for σ^2 follows immediately from the mode of an inverse gamma distribution conditional on $\beta^{(t)}$:

$$(\sigma^2)^{(t+1)} = \frac{\frac{1}{2} \sum_{j=0}^{p} \left(\beta_j^{(t)}\right)^2 \left(\frac{\langle \theta_j \rangle^{(t)}}{\kappa} + 1 - \langle \theta_j \rangle^{(t)} \right) + \lambda}{\frac{p+1}{2} + \nu + 1}. \quad (2.6)$$

The terms in (2.4) that depend on β come from the log likelihood of y and from the expected prior on β, $\beta \sim N(0, \Sigma^{(t)})$, where

$$\Sigma^{(t)} = \operatorname{Diag}\left(\frac{\sigma^2}{\langle \theta_j \rangle^{(t)} / \kappa + 1 - \langle \theta_j \rangle^{(t)}} \right).$$

Updating β is equivalent here to fitting a ridge regularized logistic regression, I exploit the usual iteratively reweighted least squares (IRLS) algorithm [60]. Setting $\mu^{(t)}$ as the vector of expected responses with $\mu_i^{(t)} = \text{logit}^{-1}(X_i^\top \beta^{(t)})$ and $W^{(t)} = \text{Diag}(\mu_i^{(t)}(1-\mu_i^{(t)}))$ as the variance weights, the update for β is then

$$\beta^{(t+1)} = \left(X^\top W^{(t)} X + \left(\Sigma^{(t)}\right)^{-1}\right)^{-1} \left(X^\top W^{(t)} X \beta^{(t)} + X^\top (y - \mu^{(t)})\right), \tag{2.7}$$

where I substitute $(\sigma^2)^{(t)}$ for σ^2 in the definition of $\Sigma^{(t)}$.

Rank truncation of design matrix

Computing and storing the inverse of the $(p+1)$-by-$(p+1)$ matrix $X^\top W^{(t)} X + \left(\Sigma^{(t)}\right)^{-1}$ in (2.7) is expensive since p is large. To alleviate this problem, I replace X with a rank truncated version based on its singular value decomposition $X = UDV^\top$. More specifically, I take the top l singular values and their respective left and right singular vectors, and so, if $D = \text{Diag}(d_i)$ and $\mathbf{u}_i$ and $\mathbf{v}_i$ are the i-th left and right singular vectors respectively,

$$X = UDV^\top = \sum_{i=1}^{n} d_i \mathbf{u}_i \mathbf{v}_i^\top \approx \sum_{i=1}^{l} d_i \mathbf{u}_i \mathbf{v}_i^\top = U_{(l)} D_{(l)} V_{(l)}^\top,$$

where $D_{(l)}$ is the l-th order diagonal matrix with the top l singular values and $U_{(l)}$ (n-by-l) and $V_{(l)}$ ($(p+1)$-by-l) contain the respective left and right singular vectors. I select l by controlling the mean squared error: l should be large enough such that $\|X - U_{(l)} D_{(l)} V_{(l)}^\top\|_F / (n(p+1)) < 0.01$.

Since $X^\top W^{(t)} X \approx V_{(l)} D_{(l)} U_{(l)}^\top W^{(t)} U_{(l)} D_{(l)} V_{(l)}^\top$, I profit from the rank truncation by defining the (upper) Cholesky factor C_w of $D_{(l)} U_{(l)}^\top W^{(t)} U_{(l)} D_{(l)}$ and $S = C_w V_{(l)}^\top$ so that

$$\begin{aligned} \left(X^\top W^{(t)} X + \left(\Sigma^{(t)}\right)^{-1}\right)^{-1} &\approx \left(S^\top S + \left(\Sigma^{(t)}\right)^{-1}\right)^{-1} \\ &= \Sigma^{(t)} - \Sigma^{(t)} S^\top \left(I_l + S \Sigma^{(t)} S^\top\right)^{-1} S \Sigma^{(t)} \end{aligned} \tag{2.8}$$

by the Kailath variant of the Woodbury identity [70]. Now I just need to store and compute the inverse of the l-th order square matrix $I_l + S\Sigma^{(t)} S^\top$ to obtain the updated $\beta^{(t+1)}$ in (2.7).

2.2.2 Gibbs sampler

After obtaining results from the EM filtering procedure, I proceed to analyze the filtered dataset by sampling from the joint posterior $\mathbb{P}(\theta, \beta, \sigma^2 \,|\, y)$ using Gibbs sampling. I iterate sampling from the conditional distributions

$$[\sigma^2 \,|\, \theta, \beta, y], \quad [\theta \,|\, \beta, \sigma^2, y], \quad \text{and} \quad [\beta \,|\, \theta, \sigma^2, y]$$

until assessed convergence.

I start by taking advantage of the conjugate prior for σ^2 and draw each new sample from

$$\sigma^2 \,|\, \theta, \beta, y \sim \mathsf{IG}\left(\nu + \frac{p+1}{2},\, \lambda + \frac{1}{2}\sum_{j=0}^{p} \beta_j^2 \Big(\frac{\theta_j}{\kappa} + 1 - \theta_j\Big)\right).$$

Sampling θ is also straightforward: since the θ_j are independent given β_j,

$$\theta_j \,|\, \beta, \sigma^2, y \overset{\text{ind}}{\sim} \mathsf{Bernoulli}(\langle \theta_j \rangle), \quad \text{for } j = 1, \ldots, p,$$

with $\langle \theta_j \rangle$ as in (2.5). Sampling β, however, is more challenging since there is no closed-form distribution based on a logistic regression, but I use a data augmentation scheme proposed by [71]. This method has been noted to perform well when the model has a complex prior structure and the data have a group structure and so I believe it is appropriate for the spatial boost model.

Thus, to sample β conditional on θ, σ^2, and y I first sample latent variables ω from a Pólya-Gamma (PG) distribution,

$$\omega_i \,|\, \beta \sim \mathsf{PG}(1, X_i^\top \beta), \quad i = 1, \ldots, n,$$

and then, setting $\Omega = \text{Diag}(\omega_i)$, $\Sigma = \text{Diag}(\sigma^2(\theta_j \kappa + 1 - \theta_j))$, and $V_\beta = X^\top \Omega X + \Sigma^{-1}$, sample

$$\beta \,|\, \omega, \theta, \sigma^2, y \sim \text{Normal}(V_\beta^{-1} X^\top (y - 0.5 \cdot \mathbf{1}_n),\, V_\beta^{-1}).$$

I note that the same rank truncation used in the EM algorithm from the previous section works here, and I gain more computational efficiency by using an identity similar to (2.8) when computing and storing V_β^{-1}.

2.2.3 Centroid estimation

To conduct inference on θ I follow statistical decision theory [9] and define an estimator based on a generalized Hamming loss function $H(\theta, \tilde{\theta}) = \sum_{j=1}^{p} h(\theta_j, \tilde{\theta}_j)$,

$$\hat{\theta}_C = \underset{\tilde{\theta} \in \{0,1\}^p}{\arg\min}\, \mathbb{E}_{\theta\,|\,y}\big[H(\theta, \tilde{\theta})\big] = \underset{\tilde{\theta} \in \{0,1\}^p}{\arg\min}\, \mathbb{E}_{\theta\,|\,y}\left[\sum_{j=1}^{p} h(\theta_j, \tilde{\theta}_j)\right]. \tag{2.9}$$

I assume that h has symmetric error penalties, $h(0,1) = h(1,0)$ and that $h(1,0) > \max\{h(0,0), h(1,1)\}$, that is, the loss for a false positive or negative is higher than for a true positive and true negative. In this case, I can define a *gain* function g by subtracting each entry in h from $h(1,0)$ and dividing by $h(1,0) - h(0,0)$:

$$g(\theta_j, \tilde{\theta}_j) = \begin{cases} 1, & \theta_j = \tilde{\theta}_j = 0, \\ 0, & \theta_j \neq \tilde{\theta}_j, \\ \gamma \doteq \dfrac{h(1,0) - h(1,1)}{h(1,0) - h(0,0)}, & \theta_j = \tilde{\theta}_j = 1. \end{cases}$$

Gain $\gamma > 0$ represents a sensitivity-specificity trade-off; if $h(0,0) = h(1,1)$, that is, if true positives and negatives have the same relevance, then $\gamma = 1$.

Let us define the marginal posteriors $\pi_j \doteq \mathbb{P}(\theta_j = 1\,|\,y)$. The above estimator is then equivalent to

$$\begin{aligned} \hat{\theta}_C &= \underset{\tilde{\theta} \in \{0,1\}^p}{\arg\max}\, \mathbb{E}_{\theta\,|\,y}\left[\sum_{j=1}^{p} g(\theta_j, \tilde{\theta}_j)\right] \\ &= \underset{\tilde{\theta} \in \{0,1\}^p}{\arg\max} \sum_{j=1}^{p} (1 - \tilde{\theta}_j)(1 - \pi_j) + \gamma \tilde{\theta}_j \theta_j = \underset{\tilde{\theta} \in \{0,1\}^p}{\arg\max} \sum_{j=1}^{p} \left(\pi_j - \frac{1}{1+\gamma}\right)\tilde{\theta}_j, \end{aligned}$$

which can be obtained position-wise,

$$(\hat{\theta}_C)_j = I\left(\pi_j - \frac{1}{1+\gamma} \geq 0\right). \tag{2.10}$$

The estimator in (2.9) is known as the *centroid estimator*; in contrast to maximum *a posteriori* (MAP) estimators that simply identify the highest peak in a posterior distribution, centroid estimators can be shown to be closer to the mean than to a mode of the posterior space, and so offer a better summary of the posterior distribution [15]. Related formulations of centroid estimation for binary spaces in (2.10) have been proposed in many bioinformatics applications in the context of maximum expected accuracy [35]. Moreover, if $\gamma = 1$ then $\hat{\theta}_C$ is simply a consensus estimator and coincides with the median probability model estimator of [8].

Finally, I note that the centroid estimator can be readily obtained from MCMC samples $\theta^{(1)}, \ldots, \theta^{(N)}$; I just need to estimate the marginal posterior probabilities $\hat{\pi}_j = \sum_{s=1}^{N} \theta_j^{(s)}/N$ and substitute in (2.10).

2.3 Guidelines for Selecting Prior Parameters

Since genome-wide association is a large-p-small-n problem, I rely on adequate priors to guide the inference and overcome ill-posedness. In this section I provide guidelines for selecting hyperpriors κ in the slab variance of β, and ϕ, ξ_0, and ξ_1 in the prior for θ.

2.3.1 Selecting ϕ

Biologically, some locations within a chromosome may be less prone to recombination events and consequently to relatively higher linkage disequilibrium. LD can be characterized as correlation in the genotypes, and since I analyze the entire genome, high correlation in markers within a chromosome often results in poor coefficient estimates for the logistic regression model in 2.1. To account for potentially varying spatial relationships across the genome, I exploit the typical correlation pattern in GWAS data sets to suggest a value for

ϕ that properly encodes the spatial relationship between markers and genes in a particular region as a function of genomic distance. To this end, I propose the following procedure to select ϕ:

1. Divide each chromosome into regions such that the distance between the SNPs in adjacent regions is at least the average length of a human gene, or 30,000 base pairs [80]. The resulting regions will be, on average, at least a gene's distance apart from each other and may possibly exhibit different patterns of correlation.

2. Merge together any adjacent regions that cover the same gene. Although the value of ϕ depends on each region, I want the meaning of the weights assigned from a particular gene to SNPs in the Spatial Boost model to be consistent across regions. As a practical example, by applying the first two steps of the pre-processing procedure on chromosome 1, I obtain 1,299 windows of varying sizes ranging from 1 to 300 markers.

3. Iterate over each region and select a value of ϕ that best fits the magnitude of the genotype correlation between any given pair of SNPs as a function of the distance between them. I propose using the normal curve given in the definition of the gene weights to first fit the magnitudes, and then using the mean squared error between the magnitudes in the sample correlation matrix of a region and the magnitudes in the fitted correlation matrix as a metric to decide the optimal value of ϕ. In particular, given two SNPs located at positions s_i and s_j, I relate the magnitude of the correlation between SNPs i and j to the area

   $$|\rho_{i,j}|(\phi) = 2\Phi\left(- \frac{|s_i - s_j|}{\phi} \right),$$

   where Φ is the standard normal cumulative function.

   Figure 2.2 shows an example of application to chromosome 1 based on data from the case study discussed in Section 2.4. I note that the mean squared error criterion

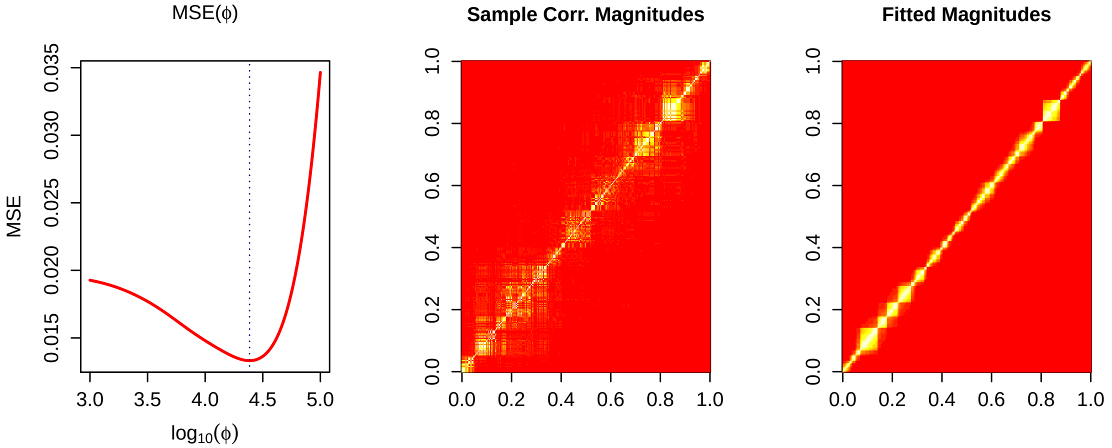

Figure 2.2: Example of selection of ϕ: when using the proposed values of $|\rho_{i,j}|$ to fit the sample correlation magnitudes, I obtain an optimal choice of $\phi = 13{,}530$ for a random window. The second two plots are heatmaps of the pair-wise correlation magnitudes between all SNPs in the window.

places more importance on fitting relatively larger magnitudes close to the diagonal of the image matrix, and so there is little harm in choosing a moderate value for ϕ that best fits the magnitudes of dense groups of correlated SNPs in close proximity.

2.3.2 Selecting ξ_0 and ξ_1

According to the centroid estimator in (2.10), the j-th SNP is identified as associated if $\pi_j \geq (1+\gamma)^{-1}$. Following a similar criterion, but with respect to the conditional posteriors, I have $\mathbb{P}(\theta_j = 1 \,|\, y, \beta, \sigma^2) = \langle \theta_j \rangle \geq (1+\gamma)^{-1}$, and so, using (2.5),

$$\text{logit}\langle \theta_j \rangle = -\frac{1}{2}\log\kappa + \xi_0 + \xi_1 \mathbf{w}_j^\top \mathbf{r} + \frac{\beta_j^2}{2\sigma^2}\left(1 - \frac{1}{\kappa}\right) \geq -\log\gamma.$$

After some rearrangements, I see that, in terms of β_j, this criterion is equivalent to $\beta_j^2 \geq \sigma^2 s_j^2$ with

$$s_j^2 \doteq \frac{2\kappa}{\kappa - 1}\left(\frac{1}{2}\log\kappa - \xi_0 - \xi_1 \mathbf{w}_j^\top \mathbf{r} - \log\gamma\right), \tag{2.11}$$

that is, I select the j-th marker if β_j is more than s_j "spike" standard deviations σ away from zero.

This interpretation based on the EM formulation leads to a meaningful criterion for defining ξ_0 and ξ_1: I just require that $\min_{j=1,\ldots,p} s_j^2 \geq s^2$, that is, that the smallest number of standard deviations is at least $s > 0$. Since $\max_{j=1,\ldots,p} \mathbf{w}_j^\top \mathbf{r} = 1$,

$$\min_{j=1,\ldots,p} s_j^2 = \frac{2\kappa}{\kappa - 1}\left(\frac{1}{2}\log\kappa - \xi_0 - \xi_1 - \log\gamma\right) \geq s^2,$$

and so,

$$\xi_1 \leq \frac{1}{2}\log\kappa - \xi_0 - \log\gamma - \frac{s^2}{2}\left(1 - \frac{1}{\kappa}\right). \tag{2.12}$$

For a more stringent criterion, I can take the minimum over κ in the right-hand side of (2.12) by setting $\kappa = s^2$. When setting ξ_1 it is also important to keep in mind that ξ_1 is the largest allowable gene boost, or better, increase in the log-odds of a marker being associated to the trait.

Since ξ_0 is related to the prior probability of a SNP being associated, I can take ξ_0 to be simply the logit of the fraction of markers that I expect to be associated *a priori*. However, for consistency, since I want $\xi_1 \geq 0$, I also require that the right hand side of (2.12) be non-negative, and so

$$\xi_0 + \log\gamma \leq \frac{1}{2}\log\kappa - \frac{s^2}{2}\left(1 - \frac{1}{\kappa}\right). \tag{2.13}$$

Equation (2.13) constraints ξ_0 and γ jointly, but I note that the two parameters have different uses: ξ_0 captures my prior belief on the probability of association and is thus part of the model specification, while γ defines the sensitivity-specificity trade-off that is used to identify associated markers, and is thus related to model inference.

As an example, if $\gamma = 1$ and I set $s = 4$, then the bound in (2.12) with $\kappa = s^2$ is $\log(s^2)/2 - s^2(1 - 1/s^2)/2 = -6.11$. If I expect 1 in 10,000 markers to be associated, I have $\xi_0 = \text{logit}(10^{-4}) = -9.21 < -6.11$ and the bound (2.13) is respected. The upper bound for ξ_1 in (2.12) is thus 3.10.

2.3.3 Selecting κ

I propose using a metric similar to the Bayesian false discovery rate [BFDR, [88]] to select κ. The BFDR of an estimator is computed by taking the expected value of the false discovery proportion under the marginal posterior distribution of θ:

$$\text{BFDR}(\hat{\theta}) = \mathbb{E}_{\theta \,|\, y}\left[\frac{\sum_{j=1}^{p} \hat{\theta}_j(1-\theta_j)}{\sum_{j=1}^{p} \hat{\theta}_j}\right] = \frac{\sum_{j=1}^{p} \hat{\theta}_j(1-\pi_j)}{\sum_{j=1}^{p} \hat{\theta}_j}.$$

Since, as in the previous section, I cannot obtain estimates of $\mathbb{P}(\theta_j = 1 \,|\, y)$ just by running my EM algorithm, I consider instead an alternative metric that uses the conditional posterior probabilities of association given the fitted parameters, $\langle\theta_j\rangle = \mathbb{P}(\theta_j = 1 \,|\, y, \hat{\beta}_{EM}, \hat{\sigma}^2_{EM})$. I call this new metric EMBFDR:

$$\text{EMBFDR}(\hat{\theta}) = \frac{\sum_{j=1}^{p} \hat{\theta}_j(1-\langle\theta_j\rangle)}{\sum_{j=1}^{p} \hat{\theta}_j}.$$

Moreover, by the definition of the centroid estimator in (2.10), I can parameterize the centroid EMBFDR using γ:

$$\text{EMBFDR}(\hat{\theta}_C(\gamma)) = \text{EMBFDR}(\gamma) = \frac{\sum_{j=1}^{p} I[\langle\theta_j\rangle \geq (1+\gamma)^{-1}](1-\langle\theta_j\rangle)}{\sum_{j=1}^{p} I[\langle\theta_j\rangle \geq (1+\gamma)^{-1}]}.$$

I can now analyze a particular data set using a range of values for κ and subsequently make plots of the EMBFDR metric as a function of the threshold $(1+\gamma)^{-1}$ or as a function of the proportion of SNPs retained after the EM filter step. Thus, by setting an upper bound for a desired value of the EMBFDR I can investigate these plots and determine an appropriate choice of κ and an appropriate range of values of γ. In Figure 2.3 I illustrate an application of this criterion. I note that the EMBFDR has broader application to Bayesian variable selection models and can be a useful metric to guide the selection of tuning parameters, in particular the spike-and-slab variance separation parameter κ.

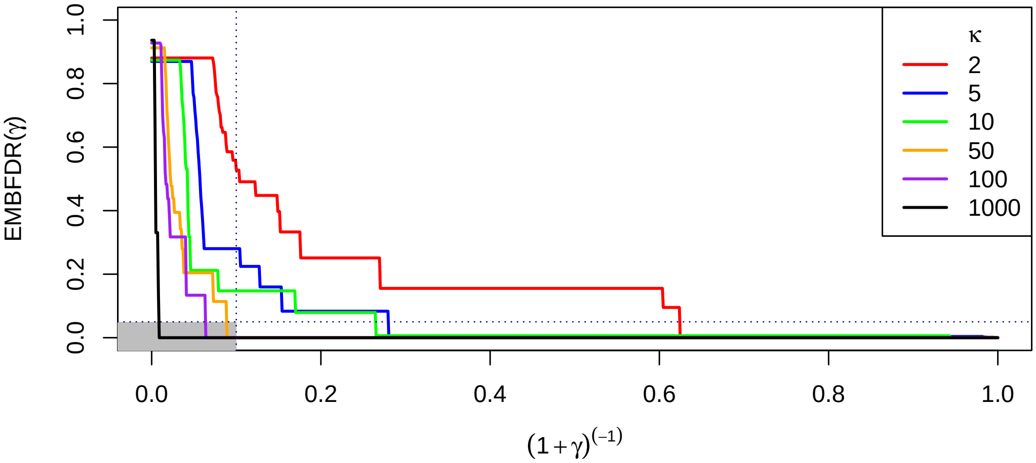

Figure 2.3: When analyzing a data set generated for a simulation study as described in Section 2.4, I inspect the behavior of the BFDR as a function of γ for various values of κ and see that a choice of $\kappa = 1,000$ would be appropriate to achieve a BFDR no greater than 0.05 when using a threshold of $(1+\gamma)^{-1} = 0.1$.

2.3.4 Visualizing the relationship between SNPs and genes

For a given configuration of κ, γ, and σ^2, I can plot the bounds $\pm\sigma s_j$ on β_j that determine how large $|\beta_j|$ needs to be in order for the j^{th} SNP to be included in the model, and then inspect the effect of parameters ϕ, ξ_0, and ξ_1 on these bounds. SNPs that are close to relevant genes have thresholds that are relatively lower in magnitude; they need a relatively smaller (in magnitude) coefficient to be selected for the final model. With everything else held fixed, as $\phi \to \infty$ the boost received from the relevant genes will decrease to zero and my model will coincide with a basic version of Bayesian variable selection where $\theta_j \overset{\text{iid}}{\sim} \mathsf{Bernoulli}(\text{logit}^{-1}(\xi_0))$. I demonstrate this visualization on a mock chromosome in Figure 2.4.

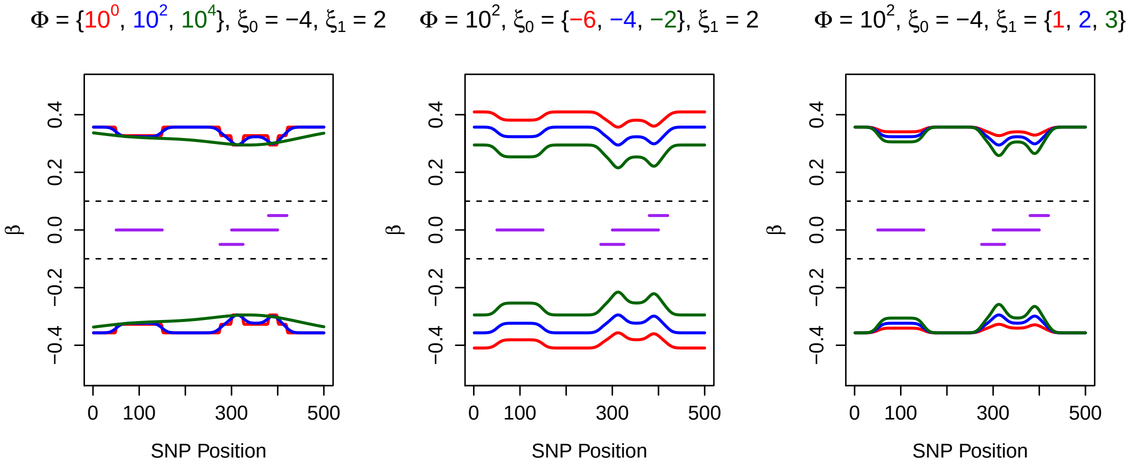

Figure 2.4: I illustrate the effect of varying ϕ, ξ_0 and ξ_1 on the thresholds on the posterior effect sizes, β_j, in a simple window containing a single gene in isolation, and a group of three overlapping genes. On the left, I vary ϕ and control the smoothness of the thresholds. In the middle, I vary ξ_0 and control the magnitude of the thresholds, or in other words the number of standard deviations (σ) away from zero at which they are placed. On the right, I vary ξ_1 and control the sharpness of the difference in the thresholds between differently weighted regions of the window. For this illustration, I set $\sigma^2 = 0.01$, $\kappa = 100$, and $\gamma = 1$. I mark the distance σ away from the origin with black dashed lines.

2.4 Empirical Studies

I conduct two simulation studies. First, I compare the performance of our method to other well-known methods including single SNP tests, LASSO, fused LASSO, group LASSO, PUMA, and BSLMM. Then I assess the robustness of our method to misspecifications of the range parameter, ϕ, and gene relevances. I describe each study in detail below, but I first explain how the data is simulated in each scenario.

2.4.1 Simulation Study Details

To provide a fair comparison across methods and to better assess the robustness of my method to misspecifications, I adopt an independent model to simulate data. I use the GWAsimulator program [58] because it can achieve a more representative LD structure from real data, it keeps the retrospective nature of my design, and it is widely used. GWAsimulator generates both genotypes and phenotypes based on the following inputs: disease prevalence, genotypic relative risk, number of cases and controls, haplotypes (phased genotypes), and the locations of causal markers. It is also possible to specify individuals' gender and optionally two-way interactions between SNPs; to avoid gender biases in my study, I sampled each individual as male, and I did not consider any interactions.

The phenotypes are already specified by the number of cases and controls. To give the minimum contrast between cases and controls and to simplify the simulated data sets, I always chose a balanced design and sampled an equal number of cases and controls. Genotypes are sampled separately for causal and non-causal markers. Causal marker genotypes are sampled retrospectively from a logistic regression where the effect sizes are calculated from the disease prevalence, genotypic relative risk and frequency of risk alleles (computed from the inputted haplotypes). Then, genotypes for non-causal markers are simulated based on the haplotype data with the aim of maintaining Hardy-Weinberg equilibrium, allele frequencies, and linkage disequilibrium patterns in the inputted haplotypes. Because GWAsimulator retains the observed LD patterns in the input phased data sets, I argue that it offers a realistic example of data for my study.

For both studies, I simulated two scenarios for n, the number of individuals, and p, the number of markers: $n = 120, p = 6{,}000$, the "small" scenario, and $n = 1{,}200, p = 60{,}000$, the "large" scenario. The input haplotypes to GWAsimulator came from phased data provided by the 1000 Genomes Project [18]. The program requires that there be only one causal SNP per chromosome; thus, if I wish to sample m causal markers, I divide the total number of markers, p, into m equally sized blocks, i.e. each block with p/m contiguous markers, one per chromosome, and randomly sample the causal marker within each block. In both studies I have $m = 15$. The causal markers were sampled uniformly within each block from all markers with MAF $> 5\%$.

After sampling the causal markers, I input them to GWAsimulator which, in turn, determines the effect sizes as a function of the disease prevalence and relative risks. For all simulations I kept the default disease prevalence of 5% because it represents the realistic and challenging nature of GWAS data. The parameters that describe how disease prevalence and relative risk affect effect size are specified to GWAsimulator using a control file. For each causal marker in my simulated datasets I randomly select one of the default configurations of effect size parameters listed in the control file that ships with the program so that the genotypic relative risk (GRR) of the genotype with one copy of the risk allele versus that with zero copies of the risk allele is either 1.0, 1.1, or 1.5, and the genotypic relative risk of the genotype with two copies of the risk allele versus that with zero copies of the allele is either 1.1, 1.5, 2.0, a multiplicative effect (GRR $\times$ GRR), or a dominance effect (GRR).

In each simulation scenario and dataset below I fit the model as follows: I start with $\xi_0 = \text{logit}(100/p)$, a moderate gene boost effect of $\xi_1 = -0.5\xi_0$, and run the EM filtering process until at most 100 markers remain. At the end of the filtering stage I run the Gibbs sampler with $\xi_0 = \text{logit}(m/100)$ and $\xi_1 = -0.5\xi_0$. This ratio of $\xi_1/\xi_0 = -0.5$ is kept across all EM filtering iterations, and is a simple way to ensure that the guideline from (2.12) is followed with $\kappa = s^2 = \gamma = 1$. Parameter κ is actually elicited at each EM filtering iteration using EMBFDR, and I fix $\phi = 10{,}000$ for simplicity and to assess robustness.

2.4.2 Comparison Simulation Study

In this study I generated 20 batches of simulated data, each containing 5 replicates, for a total of 100 simulated data sets for each configuration of n, p above. In each batch I simulate $m = 15$ blocks, where each block comprises p/m markers that are sampled contiguously from the whole set of annotated markers in each chromosome, that is, for each block I sample an initial block position (marker) from its respective chromosome and take consecutive p/m markers from that position. After simulating the data, I fit my model and compared its performance in terms of the area under the Receiver Operating Characteristic (ROC) curve, or AUC [10], to the usual single SNP tests, LASSO, fused LASSO, group LASSO, PUMA, and BSLMM methods. I used the **penalized** package in R to fit the LASSO and fused LASSO models; I used two-fold cross-validation to determine the optimal values for the penalty terms. For computational feasibility, before fitting the fused and group LASSO models when $p = 60{,}000$, I used the same pre-screening idea that is employed by the PUMA software, i.e. first run the usual single SNP tests and remove any SNP that has a p-value above 0.01. Similarly, I used the **gglasso** package in R to fit the group LASSO model where I defined the groups such that any two adjacent SNPs belonged to the same group if they were within 10,000 base pairs of each other; I used 5-fold cross validation to determine the optimal value for the penalty term. Finally, I used the authors' respective software packages to fit the PUMA and BSLMM models.

To calculate the AUC for any one of these methods, I took a final ranking of SNPs based on an appropriate criterion (see more about this below), determined the points on the receiver operating characteristic (ROC) curve using my knowledge of the true positives and the false positives from the simulated data's control files, and then calculated the area under this curve. For my model, I used either the ranking (in descending order) of $\mathbb{E}[\theta_j|\hat{\beta}_{\text{EM}}, \hat{\sigma}^2_{\text{EM}}, y]$ for a particular EM filtering step or $\hat{\mathbb{P}}(\theta_j = 1|y)$ using the samples obtained by the Gibbs sampler; for the single SNP tests I used the ranking (in ascending order) of the p-values for each marker's test; for LASSO, fused LASSO and group LASSO I used the ranking (in descending order) of the magnitude of the effect sizes of the SNPs in the final model; for the other penalized regression models given by the PUMA program, I used the provided software to compute p-values for each SNP's significance in the final model and used the ranking (in ascending order) of these p-values; for BSLMM I used the ranking (in descending order) of the final estimated posterior probabilities of inclusion for each SNP in the final model.

I summarize the results in Figure 2.5; my methods perform better than the other methods in the "small" simulation scenario, but comparably in the "large" simulation scenario. Not surprisingly, the "null" ($\xi_1 = 0$) and "informative" model ($\xi_1 > 0$) yield similar results in the small scenario since the markers were simulated uniformly and thus independently of gene relevances. Interestingly, for this scenario, EM filtering is fairly effective in that my models achieve better relative AUCs under low false positive rates, as the bottom left panel in Figure 2.5 shows. I computed the relative AUC, i.e. the area under the ROC curve up to a given false positive rate divided by that false positive rate, in the bottom panels up to a false positive rate of 20%.

When compared to the small scenario, the relatively worse results in the large scenario can be explained mostly by two factors: (i) an inappropriate choice for the range parameter ϕ: because ϕ is relatively large given a higher density of markers, more markers neighboring gene regions have artificially boosted effects which then inflate the false positive rate; and (ii) a more severe model misspecification: having more markers translates to higher LD since the markers tend to be closer. Because of the first factor the informative model does not give competitive results here; nonetheless, it still outperforms the PUMA suite of models and BSLMM at lower false positive rates. The null model, however, performs comparably to single SNP tests and the LASSO models, since none of these models can account well for high genotypical correlation. However, as the bottom right panel in Figure 2.5 shows and as observed in the small scenario, the EM filtering procedure improves the performance of my model at lower false positive rates, with more pronounced gains in the informative model.

2.4.3 Relevance Robustness Simulation Study

To investigate the effect of misspecifications of ϕ and r on the performance of my model, I again considered the possible configurations where $n = 120, p = 6{,}000$ ("small") and $n = 1{,}200, p = 60{,}000$ ("large") and randomly selected one of the 100 simulated data sets from the comparison study to be the ground truth in each scenario. I varied $\phi \in \{10^3, 10^4, 10^5\}$ and, for each value of ϕ, simulated 25 random relevance vectors r. The relevances r were simulated in the following way: each gene g has, independently, a probability ρ of being "highly relevant"; if gene g is sampled as "highly relevant" then $r_g \sim F_r$, otherwise $r_g = 1$. I set ρ and F_r using MalaCards relevance gene scores for Rheumatoid Arthritis (RA): ρ is defined as the proportion of genes in the reference dataset (UCSC genome browser gene set) that are listed as relevant for RA in the MalaCards database, and F_r is the empirical distribution of gene scores for these genes deemed relevant.

Hyper-prior parameters ξ_0, ξ_1, and κ were elicited as in Section 2.4.1. For each simulated replication I then fit my model and assess performance using the AUC, as in the previous study. I focus on the results for the large scenario since they are similar, but more pronounced than the small scenario. Figure 2.6 illustrates the distribution of scores for relevant genes for RA in MalaCards and how the performance of the model varies at each EM filtering iteration as a function of ϕ. Since the proportion of relevant genes ρ is small, $\rho \approx 0.001$, the results are greatly dependent on ϕ and vary little as the scores r change, in comparison. Both small and large values of ϕ can degrade model performance since, as pointed out in Section 2.1.1, markers inside relevant genes can either be overly favored as ϕ gets closer to zero, or, in the latter case when ϕ is large and extends gene influence, all genes become irrelevant, that is, I have a "null" model. In contrast, the relevance scores have more impact when ϕ is in an adequate range, as the bottom left panel of Figure 2.6 shows. Thus, the model is fairly robust to relevance misspecifications, but can achieve good performances for suitable values of range ϕ.

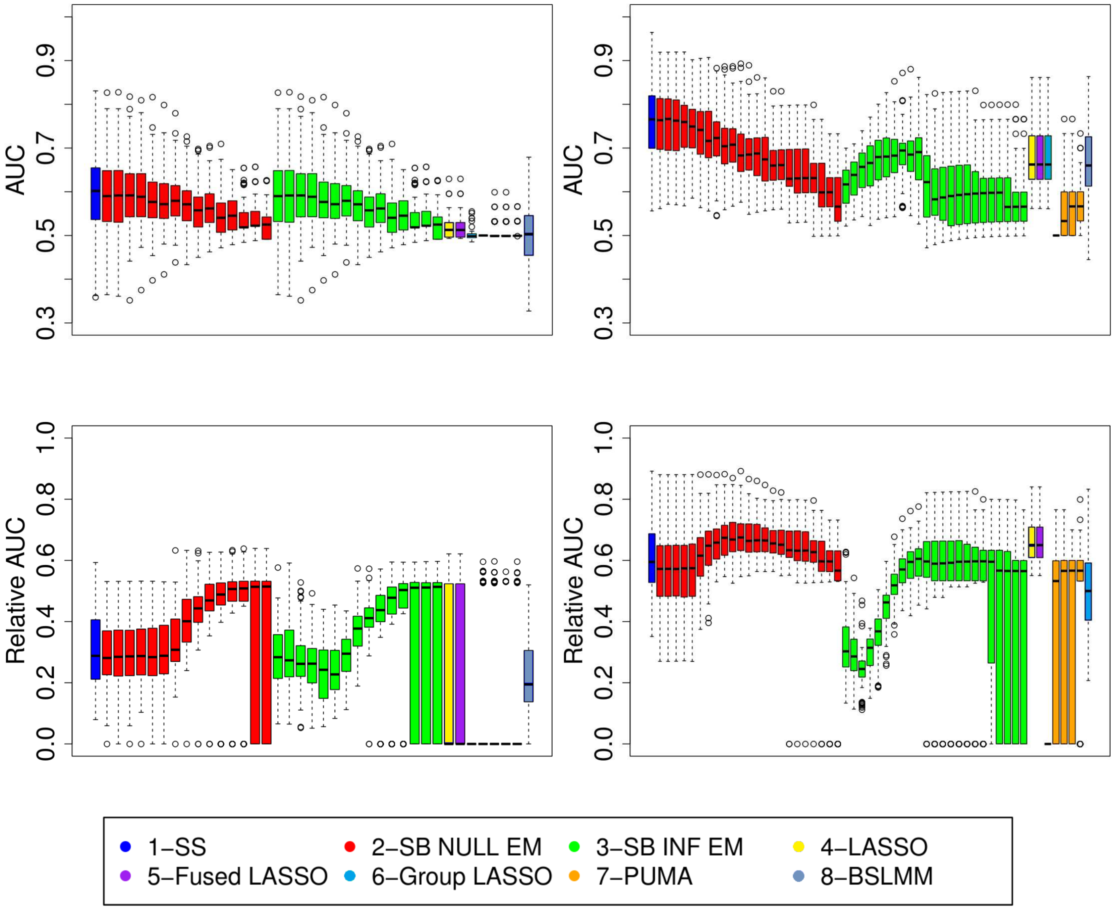

Figure 2.5: Results from the comparison simulation study. Left panels show AUC (top) and relative AUC at maximum 20% false positive rate (bottom) for "small" study, while right panels show respective AUC results for the "large" study. The boxplots are, left to right: single SNP (SS) tests (blue); spatial boost "null" model at each EM filtering iteration (red); spatial boost "informative" model at each EM filtering iteration (green); LASSO (yellow); fused LASSO (magenta); grouped LASSO (sky blue); PUMA with models NEG, LOG, MCP, and adaptive LASSO (orange); and BSLMM (sea blue).

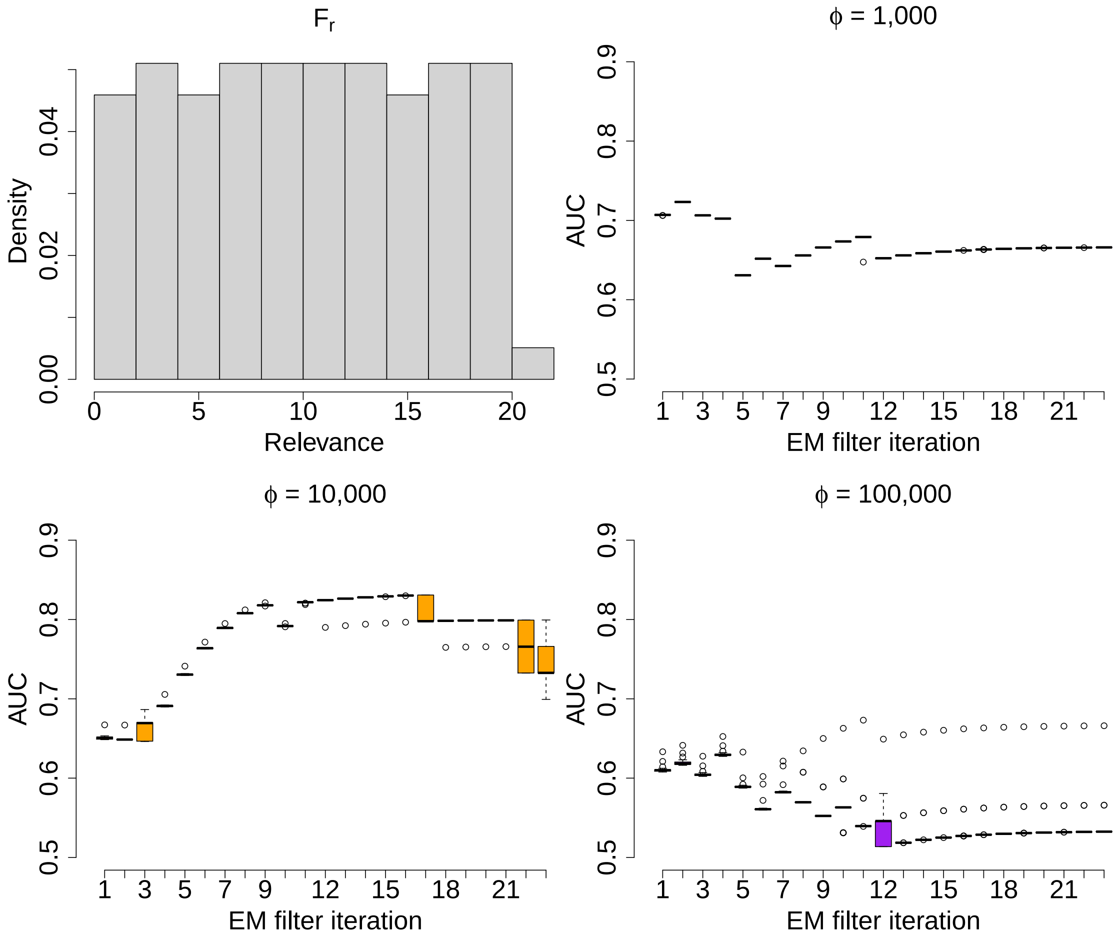

Figure 2.6: Results from the simulation study to assess robustness to gene relevances and range. Top left: distribution of gene relevance scores for RA in MalaCards. Remaining panels: AUC boxplots across simulated relevance vectors, at each EM filtering iteration, for different values of ϕ.

2.5 Case Study

Using data provided by the WTCCC, I analyzed the entire genome (342,502 SNPs total) from a case group of 1,999 individuals with rheumatoid arthritis (RA) and a control group of 1,504 individuals from the 1958 National Blood Bank dataset. For now I addressed the issues of rare variants and population stratification by only analyzing SNPs in Hardy-Weinberg Equilibrium [89] with minor allele frequency greater than 5%. There are 15 SNPs that achieve genome-wide significance when using a Bonferroni multiple testing procedure on the results from a single SNP analysis. Table 6.1 provides a summary of these results for comparison to those obtained when using the spatial boost model.

When fitting the spatial boost model, I broke each chromosome into blocks and selected an optimal value of ϕ for each block using my proposed method metric, $|\rho_{i,j}|(\phi)$. I used the EMBFDR to select a choice for κ from the set $\{10^2, 10^3, 10^4, 10^5, 10^6\}$ at each step of my model fitting pipeline so that the BFDR was no greater than 0.05 while retaining no larger than 5% of the total number of SNPs. With a generous minimum standard deviation $s = 1$ I have that trivially $\xi_0 < 0$ from (2.13), but I set $\xi_0 = -8$ to encode a prior belief that around 100 markers would be associated to the trait on average *a priori*. The bound on ξ_1 is then $\xi_1 \leq 8$, but I consider log odds-ratio boost effects of $\xi_1 \in \{1, 4, 8\}$. A value of $\xi_1 = 1$ is more representative of low power GWA studies; however, the larger boost effects offer more weight to my prior information. For comparison, I also fit a model without any gene boost by setting $\xi_1 = 0$ (the "null" model), and also fit two models for each possible value of ξ_1 trying both a non-informative gene relevance vector and a vector based on text-mining scores obtained from [61].

To speed up the EM algorithm, I rank-truncate X using $l = 3{,}259$ singular vectors; the mean squared error between X and this approximation is less than 1%. I apply the EM filtering 29 times and investigate a measure similar to posterior predictive loss [PPL, [28]] to decide when to start the Gibbs sampler. If, at the t-th EM iteration, $\hat{y}_i^{(t)} = \mathbb{E}[y_{i,\text{rep}} \,|\, \hat{\beta}_{\text{EM}}^{(t)}, y]$ is the i-th predicted response, the PPL measure under squared error loss is approximated

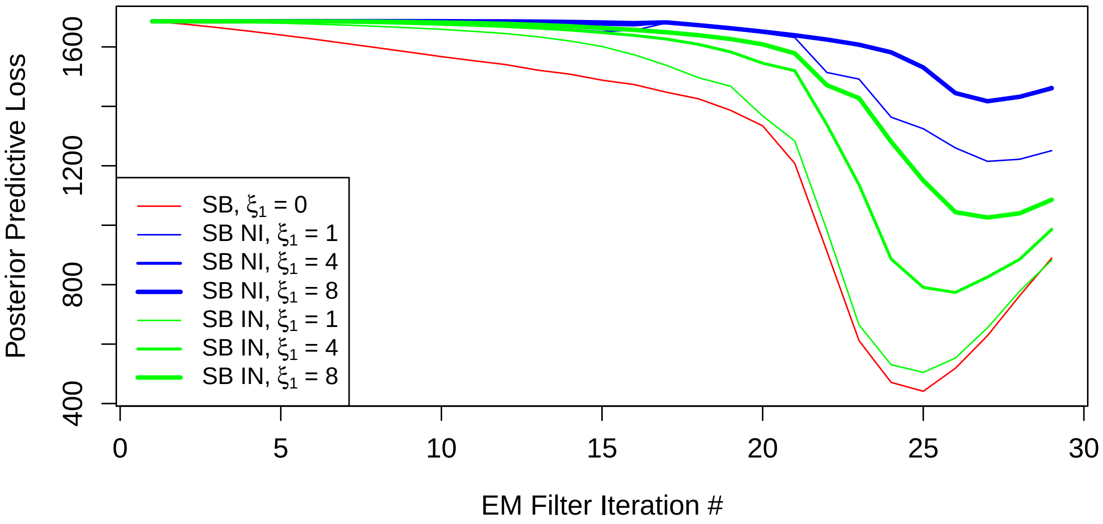

Figure 2.7: Although I run the EM filter until the number of retained markers < 100 (iteration #29), the PPL metric often tells me to keep between 200 to 250 markers (iterations #25–26).

by

$$\mathrm{PPL}(t) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \sum_{i=1}^{n} \mathrm{Var}[y_{i,\mathrm{rep}} \,|\, \hat{\beta}^{(t)}_{EM}, y] = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \hat{y}_i(1 - \hat{y}_i).$$

As Figure 2.7 shows, in all of my fitted models, the PPL decreases slowly and uniformly for the first twenty or so iterations, and then suddenly decreases more sharply for the next five iterations until it reaches a minimum and then begins increasing uniformly until the final iteration. For comparison to the 15 SNPs that achieve genome-wide significance in the single marker tests, Tables 6.2 through 6.15 list, for each spatial boost model, the top 15 SNPs at the optimal EM filtering step, i.e. the step with the smallest PPL, and the top 15 SNPs based on the posterior samples from my Gibbs sampler when using only the corresponding set of retained markers.

I observe the most overlap with the results of the single SNP tests in my null model where $\xi_1 = 0$ and in my models that use informative priors based on relevance scores from

MalaCards. Although there is concordance between these models in terms of the top 15 SNPs, it is noteworthy that I select only a fraction of these markers after running either the EM algorithm or the Gibbs sampler. Based on the results from my simulation study where I observe superior performances for the spatial boost model at low false positive rates, I believe that an advantage of my method is this ability to highlight a smaller set of candidate markers for future investigation.

Indeed, after running my complete analysis, I observe that the usual threshold of 0.5 on $\hat{\mathbb{P}}(\theta_j = 1|y)$ would result in only the null spatial boost model ($\xi_1 = 0$), the low gene boost non-informative model ($\xi_1 = 1$), and the informative models selecting SNPs for inclusion in their respective final models. The SNPs that occur the most frequently in these final models are the first top hits from the single SNP tests: rs4718582, rs10262109, rs6679677, and rs664893, with respective minor allele frequencies: 0.08, 0.06, 0.06, 0.14, and 0.12. The SNP with the highest minor allele frequency in this set is rs6679677; this marker has appeared in several top rankings in the GWAS literature (e.g. [13]) and is in high LD with another SNP in gene PTPN22 which has been linked to RA [64].

If I only consider the final models obtained after running the EM filter, we see another interesting SNP picked up across the null and informative models: rs1028850. In Figure 2.8, I show a closer look at the region around this marker and compare the trace of the Manhattan plot with the traces of each spatial boost model's $\mathbb{E}[\theta_j|\hat{\beta}_{EM}, \hat{\sigma}^2_{\text{EM}}, y]$ values at the first iteration of the EM filter. To the best of my knowledge this marker has not yet been identified as being associated to RA; moreover, it is located inside a non-protein coding RNA gene, LINC00598, and is close to another gene that has been linked to RA, FOXO1 [32].

As I increase the strength of the gene boost term with a non-informative relevance vector, the relatively strong prior likely leads to a mis-prioritization of all SNPs that happen to be located in regions rich in genes. In the supplementary tables I list the lengths of the genes that contain each SNP and I see that indeed the non-informative gene boost models tend to retain SNPs that are near large genes that can offer a generous boost. Perhaps due to prioritizing the SNPs incorrectly in these models, I do not actually select any markers at either the optimal EM filtering step or after running the Gibbs sampler. However, some of the highest ranking SNPs for these models, rs1982126 and rs6969220, are located in gene PTPRN2 which is interestingly a paralog of PTPN22.

2.6 Conclusions

I have presented a novel hierarchical Bayesian model for GWAS that exploits the structure of the genome to define SNP-specific prior distributions for the model parameters based on proximities to relevant genes. While it is possible that other "functional" regions are also very relevant—e.g. regulatory and highly conserved regions—and that mutations in SNPs influence regions of the genome much farther away—either upstream, downstream, or, through a complex interaction of molecular pathways, even on different chromosomes entirely—I believe that incorporating information about the genes in the immediate surroundings of a SNP is a reasonable place to start.

By incorporating prior information on relevant genomic regions, I focus on well annotated parts of the genome and was able to identify, in real data, markers that were previously identified in large studies and highlight at least one novel SNP that has not been found by other models. In addition, as shown in a simulation study, while logistic regression under large-p-small-n regimen is challenging, the spatial boost model often outperforms simpler models that either analyze SNPs independently or employ a uniform penalty term on the L_1 norm of their coefficients.

My main point is that I regard a fully joint analysis of all markers as essential to overcome genotype correlations and rare variants. This approach, however, entails many difficulties. From a statistical point of view, the problem is severely ill-posed so I rely on informative, meaningful priors to guide the inference. From a computational perspective, I also have the daunting task of fitting a large scale logistic regression, but I make it feasible by reducing the dimension of both data—intrinsically through rank truncation—

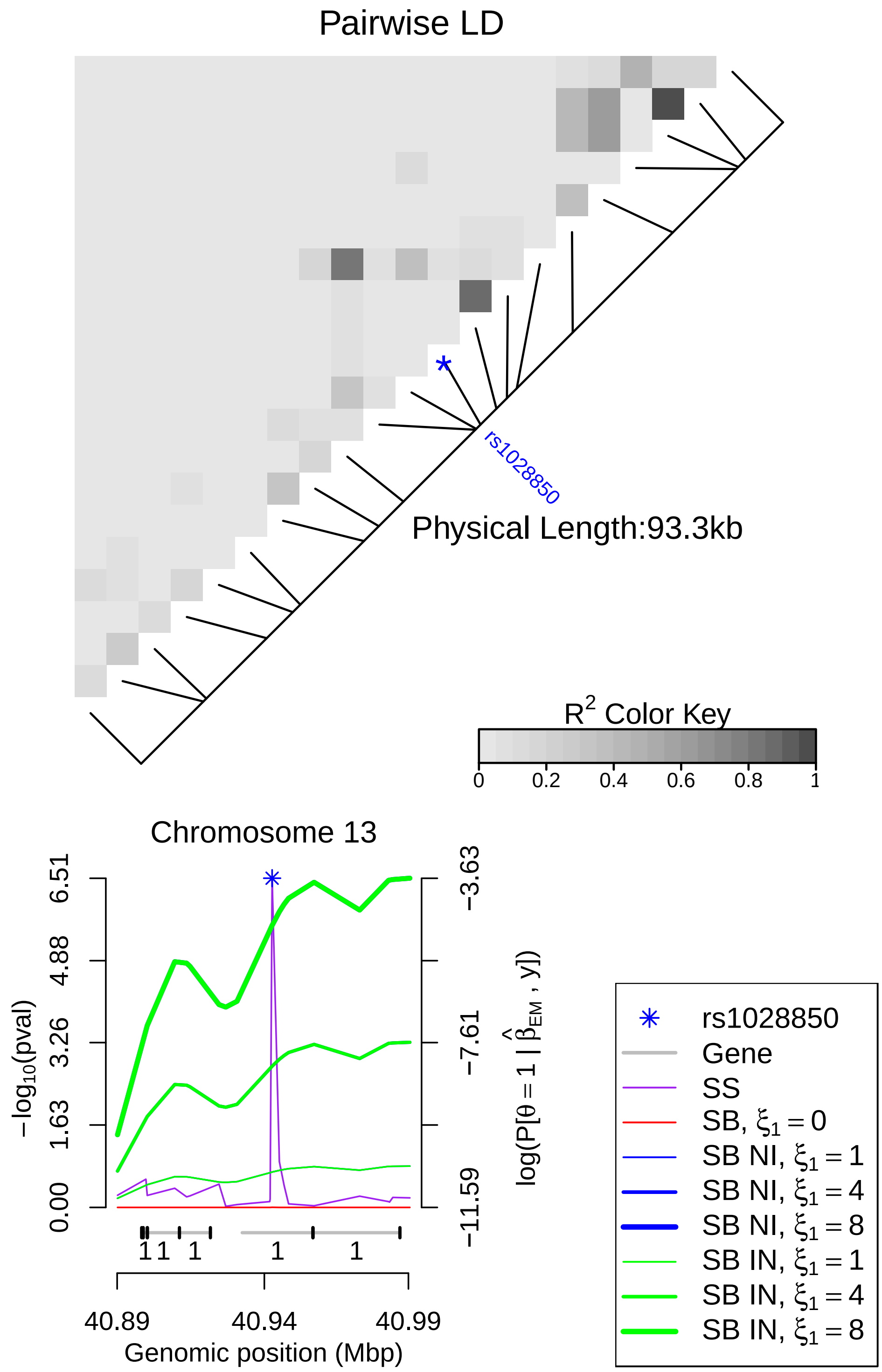

Figure 2.8: Although rs1028850 has a relative peak in the Manhattan plot (SS), it does not achieve genome-wide significance. The spatial boost (SB) model initially prioritizes markers that are closer to the center of regions rich in genes, but selects rs1028850 for inclusion in the final model by the end of the EM filter (not shown) under several configurations.

and parameters—through EM filtering. Moreover, from a practical point of view, I provide guidelines for selecting hyper-priors, reducing dimensionality, and implement the proposed approach using parallelized routines.

From the simulation studies in Section 2.4 I can further draw two conclusions. First, as reported by other methods such as PUMA, filtering is important; my EM filtering procedure seems to focus on effectively selecting true positives at low false positive rates. This feature of my method is encouraging, since practitioners are often interested in achieving higher sensitivity by focusing on lower false positive rates. Second, because I depend on good informative priors to guide the selection of associated markers, I rely on a judicious choice of hyper-prior parameters, in particular of the range parameter ϕ and how it boosts markers within neighboring genes that are deemed relevant. It is also important to elicit gene relevances from well curated databases, e.g. MalaCards, and to calibrate prior strength according to how significant these scores are.

I have shown that my model performs at least comparatively to other variable selection methods, but that it can suffer in the case of severe model misspecification. As a way to flag misspecification I suggest to check monotonicity in a measure of model fit such as PPL as I filter markers using EM. In addition, refining the EM filtering by using a lower threshold ($< .25$) at each iteration can help increase performance, especially at lower false positive rates.

When applying the spatial boost model to a real data set, I was able to confidently isolate at least one marker that has previously been linked to the trait as well as find another novel interesting marker that may be related to the trait. This shows that although I can better explore associations jointly while accounting for gene effects, the spatial boost model still might lack power to detect associations between diseases and SNPs due to the high correlation induced by linkage disequilibrium.

In Chapter 3, I develop a version of the spatial boost model for quantitative traits and explore the trade-off between performance and computational efficiency of this new model when using different rank truncations for the singular value decomposition approximation to the observed SNP data. In Chapters 4 and 5, I aim to increase the power of the spaital boost model even further by extending the model to include a data pre-processing step that attempts to formally correct for the collinearity between SNPs.

Chapter 3

Spatial Boost Model for Quantitative Trait GWAS

As I have pointed out in the preceding chapters, Bayesian variable selection provides a principled framework for incorporating prior information to regularize parameters in high-dimensional large-p-small-n regression models such as genome-wide association studies. Although these models can continually exploit the most recently available prior information in this way, researchers often disregard them in favor of simpler models because of their high computational cost. In this short chapter, I extend my spatial boost model described in Chapter 2 to quantitative traits. I then explore the trade-off of performance versus computational efficiency in comparison to single association tests through a simulation study based on real genotypes.

3.1 Model Definition

It is straightforward to extend the spatial boost model to a quantitative trait; I simply need to make a change to the likelihood function defined in Equation 2.1. The rest of the spatial boost model remains the same; however, this simple change significantly affects the update equations and posterior distributions used in the EM algorithm and Gibbs sampler. I now model the expected value of the i^{th} individual's quantitative trait, $\mathbb{E}[y_i]$, as a linear combination of the number of alleles present at a set of p SNPs encoded in $x_i^\top \in \{0, 1, 2\}^p$, and model the phenotypic variation that is not attributed to the genotypes as τ^2. Given a vector of effect sizes, β, and assuming that the observations are independent, I thus have:

$$y \mid X\beta, \tau^2 \sim \mathsf{MVN}(X\beta, \tau^2 I_n). \tag{3.1}$$

The rest of the model is as defined in Section 2.1, with only one change: I allow for τ^2 to have an inverse Gamma (IG) prior distrubution with hyper-parameters ν_1 and λ_1; consequently, I re-label the hyper-parameters of the variance of the spike component in the continuous spike-and-slab prior, σ^2, to be ν_2 and λ_2.

3.2 Model Fitting and Inference

As in Chapter 2, I want to use the centroid estimator [15] to conduct inference on θ and so I must compute $\mathbb{P}(\theta_j = 1|y)$. However, to speed up the analysis of large data sets, I first treat the θ_j as latent variables and derive an EM algorithm to obtain estimates $\beta_j^*, {\sigma^2}^*, {\tau^2}^*$ and approximate $\mathbb{P}(\theta_j = 1|y) \approx \mathbb{P}(\theta_j = 1|\beta_j^*, {\sigma^2}^*, {\tau^2}^*, y)$. I then filter SNPs by ranking $\mathbb{P}(\theta_j = 1|\beta_j^*, {\sigma^2}^*, {\tau^2}^*, y)$ in descending order and removing the bottom quartile. I repeat this process until I either reach a desired smaller number of SNPs or until the predictive accuracy of my model deteriorates beyond a certain point. Finally, I compute estimates of $\mathbb{P}(\theta_j = 1|y)$ for the remaining SNPs using a Gibbs sampler.

3.2.1 Expectation-Maximization Filter

My updated algorithm for a quantitative trait is similar to a recently proposed EM approach to Bayesian variable selection [76]. Omitting the superscripts (t) to denote the t-th iteration of the algorithm, in the E-step I compute $\mathbb{E}[\theta_j|\beta_j, \sigma^2, \tau^2, y] = \text{logit}^{-1}(S_j)$ where:

$$S_j = \xi_0 + \xi_1 \cdot w_j^\top(\phi) r + 0.5 \cdot \beta_j^2(\kappa - 1)/[\sigma^2\kappa] - 0.5 \cdot \log(\kappa). \tag{3.2}$$

In the M-step I optimize the other random variables in the model using the complete data log likelihood and the current values of $\sigma_{\theta_j}^{-2} = [\text{logit}^{-1}(S_j)/\kappa + 1 - \text{logit}^{-1}(S_j)]/\sigma^2$. Letting $\Sigma_\theta^{-1} = \text{Diag}(\sigma_{\theta_j}^{-2})$, I update β as follows:

$$\beta = (\Sigma_\theta^{-1} + \tau^{-2} X^\top X)^{-1} (\tau^{-2} X^\top y). \tag{3.3}$$

I update τ^2 and σ^2 using the modes of their respective posterior distributions:

$$\tau^2 = (\lambda_1 + 0.5 \cdot \sum_{i=1}^{n} (y_i - x_i^\top \beta)^2)/(\nu_1 + n/2 + 1), \tag{3.4}$$

$$\sigma^2 = \frac{\lambda_2 + 0.5 \cdot (\beta_0^2/\kappa + \sum_{j=1}^{p} \beta_j^2 [\text{logit}^{-1}(S_j)/\kappa + 1 - \text{logit}^{-1}(S_j)])}{\nu_2 + p/2 + 1}. \tag{3.5}$$

As before, I exploit a truncated singular value decomposition (SVD) to speed up the computation in (3.3) by replacing X with an approximation $\sum_{l=1}^{k} u_{(l)} d_{(l)} v_{(l)}^\top$. By applying the Kailath Variant matrix inverse identity, I can substitute the inversion of a p-by-p matrix with the inversion of an k-by-k matrix.

3.2.2 Gibbs Sampler

I derive the conditional posterior distributions of β, τ^2, and σ^2 as follows:

$$\beta \mid \theta, \sigma^2, y \sim \mathsf{MVN}[(\Sigma_\theta^{-1} + \tau^{-2} X^\top X)^{-1} (\tau^{-2} X^\top y), (\Sigma_\theta^{-1} + \tau^{-2} X^\top X)^{-1}], \tag{3.6}$$

$$\tau^2 \mid y, \beta \sim \mathsf{IG}(\nu_1 + n/2, \lambda_1 + 0.5 \cdot \sum_{i=1}^{n} [y_i - x_i^\top \beta]^2), \tag{3.7}$$

$$\begin{aligned}\sigma^2 \mid \beta, \theta \sim \mathsf{IG}(&\nu_2 + p/2, \\ &\lambda_2 + 0.5(\beta_0^2/\kappa + \sum_{j=1}^{p} \beta_j^2 [\text{logit}^{-1}(S_j)/\kappa + 1 - \text{logit}^{-1}(S_j)])).\end{aligned} \tag{3.8}$$

I then use Equation (3.2) to compute $P(\theta_j = 1 | \beta_j, \sigma^2, \tau^2, y)$ and derive the conditional posterior distribution of each θ_j:

$$\theta_j \mid \beta_j, \sigma^2 \sim \mathsf{Bernoulli}[\text{logit}^{-1}(S_j)]. \tag{3.9}$$

After initializing the values for β, τ^2, σ^2, and θ, I draw samples sequentially from (3.6), (3.7), (3.8), and (3.9) until I have reached a desired total number of samples for each random variable. In practice, I generate several chains of posterior samples and assess convergence using the Brooks & Gelman scale reduction factor [12] on the complete data log likelihood. I compute my final estimates of $\mathbb{P}(\theta_j = 1|y)$ for each SNP using N posterior samples as $\hat{\mathbb{P}}(\theta_j = 1|y) = \sum_{t=1}^{N} \theta_j^{(t)}/N$.

3.3 Calibration Simulation Study

Having extended the spatial boost model to quantitative traits and updated the model fitting algorithms accordingly, I now explore the computational efficiency of the model in comparison to the single SNP tests in a simulation study. To setup the study, I generate 100 matrices of size $n = 10^2$ and $p = 10^3$ by randomly selecting contiguous blocks of genotypes from an overall list of 29,711 SNPs on chromosome 2 in 3,503 individuals in a data set provided by the Wellcome Trust Case Consortium. I only consider common variants in my analyses, i.e., SNPs with minor allele frequency $> 5\%$ and variants that do not statistically significantly deviate from Hardy-Weinberg Equilibrium [89]. I choose ϕ using the guidelines in Section (2.3.1), and set $\mathbf{r} = \mathbf{1}_G$. After normalizing the gene weights given in (2.3) so that the maximum value in each data set is 1, the distribution of all gene weights is heavily left-skewed with 97.2% of the values occurring below 0.5.

In my first simulation study, I start by setting $\sigma^2 = 10^{-4}$ and $\tau^2 = 10^2$ and then sample values for θ, β and y for all 100 data sets under six different gene boost and effect size combinations. For each replicate s, I highlight the effect of the gene boost by considering both a boost-less model with $\xi_0 = \text{logit}(10/p_s)$ and $\xi_1 = 0$ as well as a model with $\xi_0 = \text{logit}(1/p_s)$ and $\xi_1 = -\text{logit}(1/p_s)$ where p_s is the number of SNPs in the s^{th}

data set. I enforce consistency in the number of true positives across data sets by sampling values for θ such that $\sum_{j=1}^{p_s} \theta_j = 10$. To vary the effect sizes of the SNPs I use a metric denoted by h^2 that is based on the heritability that is attributable to the genotypes in my dataset. More specifically, assuming that $X_{ij} \sim \text{Binomial}(2, \pi_j)$ independently where π_j is the minor allele frequency of the j^{th} SNP, I consider an approximation for h^2 as follows:

$$h^2 \approx \frac{\mathbb{E}_X[\kappa\sigma^2 \sum_{j:\theta_j=1} X_{ij}^2]}{\mathbb{E}_X[\kappa\sigma^2 \sum_{j:\theta_j=1} X_{ij}^2 + \sigma^2 \sum_{j:\theta_j=0} X_{ij}^2 + \tau^2]}. \tag{3.10}$$

To explore fitting my model to data sets where the heritability that is attributable to the covariates varies from a small proportion to a large proportion, I select κ for each data set in my simulations using Equation (3.10) to ensure a desired level of $h^2 \in \{0.1, 0.5, 0.9\}$. In my study this corresponds to choosing average values of $\kappa \in \{15{,}000, 140{,}000, 1{,}300{,}000\}$ respectively. It is noteworthy however that only $h^2 = 0.1$ provides a mildly realistic scenario for the heritability that is attributable to the genotypes in human traits. After simulating values for β and y I first apply my EM filtering algorithm to reduce the number of SNPs in each data set to a consistent 300 and then run my Gibbs sampler on the retained set of markers to obtain final estimates of $P(\theta_j = 1|y)$ using $N = 1{,}500$. In the EM filtering step I try using X as well as three different truncated SVD approximations to X where the MSE tolerance is either 1%, 10% or 25%. For comparison I run the usual association tests on my simulated data using the PLINK [74] software.

Since κ explicitly controls the difference in variability of $\beta_j \mid \theta_j, \sigma^2$ and thus greatly influences my variable selection, I investigate the sensitivity of my model to misspecifications of κ when all other model tuning parameters are ideally set. I use the first 300 consecutive SNPs in each data set and define $\sigma^2 = 10^{-4}$, $\tau^2 = 10^2$, $\xi_0 = \text{logit}(1/300)$ and $\xi_1 = -\text{logit}(1/300)$ and again sample θ such that I have 10 true positives. I consider true values of $\kappa \in \{10^3, 10^5\}$ and compute estimates of $\mathbb{P}(\theta_j = 1|y)$ for each SNP after running my Gibbs sampler for $N = 1{,}500$ iterations in 7 different models where I set $\kappa \in \{10^1, 10^2, \ldots, 10^7\}$.

Moreover, since ξ_1 determines the strength of the influence of neighboring genes on θ_j, I also investigate the sensitivity of my model to misspecifications of it. I use the same setup as above but instead set $\kappa = 10^3$, consider true values of (ξ_0, ξ_1) of either $(\text{logit}(10/300), 0)$ or $(\text{logit}(1/300), -\text{logit}(1/300))$, and fit 7 different models where I set $\xi_1 \in \{0, 1, \ldots, 6\}$. In each of my simulation studies, I set $\nu_1 = 1.1$, $\lambda_1 = 10$, $\nu_2 = 101$, and $\lambda_2 = 10^{-2}$ and assess the model performance by computing the AUC using my knowledge of the true and false positives.

3.3.1 Results

In my first simulation study I observe in Figure 3.1 that the spatial boost (SB) model outperforms the single SNP tests across all h^2 scenarios when there is a gene boost using either X or one of three SVD approximations to X with MSE tolerances of 1%, 10% and 25%. When there is not a gene boost my model suffers due to the potential sequential loss of true positive weak signals during the EM filtering step and thus achieves an average performance similar to the single SNP tests across all h^2 scenarios when using either X or an approximation with an MSE tolerance of 1%. Moreover, as expected, the performance deteriorates when using a coarser approximation for traits with moderate and high h^2 since the variation in the genotypes explains more of the variation in y. Interestingly, as also observed in my simulation studies in Chapter 2, I can achieve roughly the same level of performance by computing AUC using the final estimates of $\text{logit}^{-1}(S_j)$ after running the EM filter in place of the final estimates of $\mathbb{P}(\theta_j = 1|y)$ after running my Gibbs sampler. Based on the running times for each aspect of the SB model and the single SNP tests across several different configurations of n and p given in Table 6.16, I see that after computing the SVD of X, it is often *faster* to run a single pass of my EM filter on a coarse approximation to X (MSE tolerance of 25%) than to fit the single SNP tests. For the largest data size I considered ($n = 10^3$, $p = 10^4$), I see reductions in the time it takes to run the EM filter 5 times by 33.2%, 80.7% and 97.3% when using MSE tolerances of 1%, 10% and 25% respectively. In a few cases, it takes slightly longer to run the EM filter when using a fine approximation to X, e.g. MSE tolerance of 1%, possibly due to the extra memory needed to store three matrices instead of one.

In my second simulation study, I observe better performances from my model in Figure 3.2 when I choose $\kappa \leq 10^4$ even if the true value of κ is larger. This is likely due to the difficulty in detecting both weak and strong signals simultaneously when using a large value for κ. By selecting a relatively smaller value for κ I opt for sensitivity rather than specificity. When viewing the quartiles of the distribution of points on all 100 ROC curves for the two special cases when I select $\kappa \in \{10^1, 10^7\}$ in data sets where $\kappa = 10^5$, I do not see any benefit from being more specific in the early part of the curve by choosing $\kappa = 10^7$. In my third simulation study, I observe in Figure 3.3 that the SB model is robust to misspecifications of ξ_1 when there is no gene boost, but is sensitive to them otherwise.

3.4 Conclusions

I find that in a variety of gene boost and h^2 configurations, my extended pipeline for analyzing quantitative trait GWAS data sets using the SB model is also an efficient way of fitting a representative model to SNPs *jointly* that exploits proximities to relevant genes to uniquely define prior probabilities of association. Although it takes an impractical amount of time to run my Gibbs sampler, I achieve the same level of performance at a reasonable fraction of that computational cost by settling for the final estimates of $\text{logit}^{-1}(S_j)$ after running my EM filter in place of the final estimates of $\mathbb{P}(\theta_j = 1|y)$ after running the Gibbs sampler. Computing the SVD of X is the next largest computational cost when using my model; however, researchers may already perform such a computation when they apply principal components analysis to genotype data for instance to adjust for population stratification [72] before any subsequent analysis. To maintain a competitive edge when analyzing whole genomes in the future, I may further benefit from analyzing chromosomes in blocks defined based on genomic distance or linkage disequilibrium. In the next chapters, I explore this direction and introduce a model that accounts for the collinearity in X directly

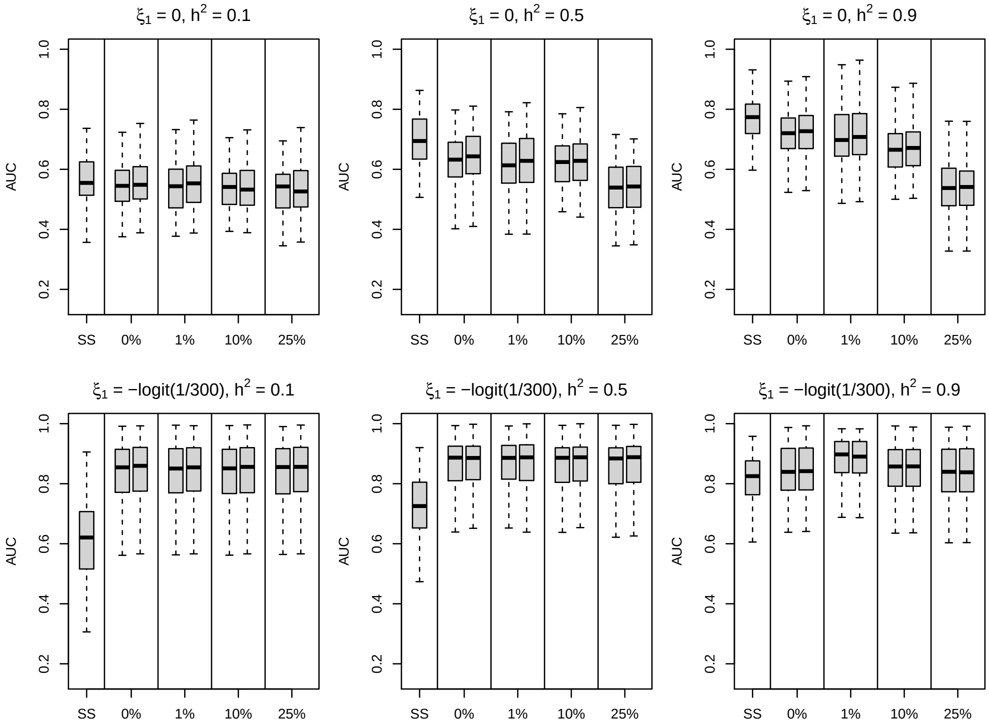

Figure 3.1: These boxplots depict the performance of the single SNP tests (SS) and the SB model across 6 different gene boost and h^2 combinations and 100 different genotype patterns. The %'s indicate the tolerance on MSE that I required when replacing X with an approximation. For each set of SB model results, I present a boxplot (left) for the AUC values based on the final estimates of $\text{logit}^{-1}(S_j)$ after running the EM filter and a boxplot (right) for the AUC values based on the final estimates of $\mathbb{P}(\theta_j = 1|y)$ after running the Gibbs sampler.

in the modeling procedure.

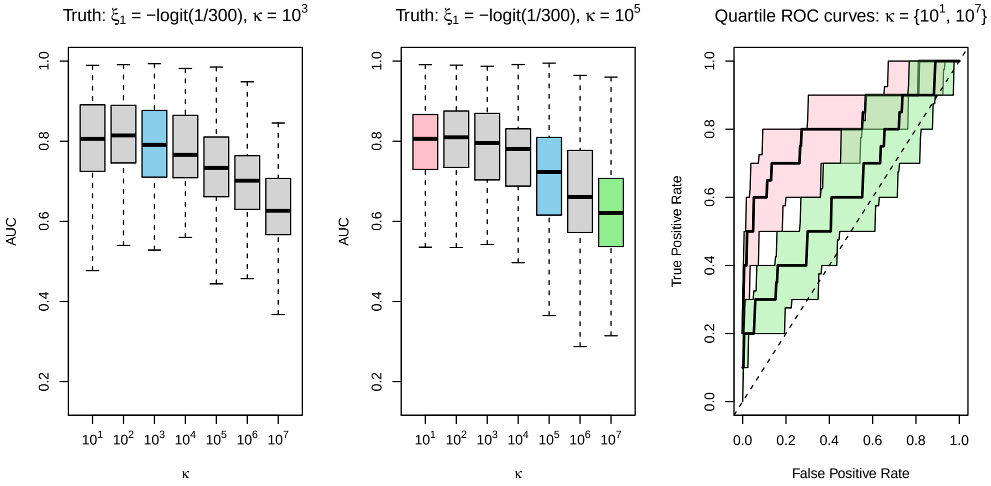

Figure 3.2: These boxplots depict the performance of the SB model in our second simulation study where I vary κ and fit my model to 100 data sets simulated from two different models where $\kappa = 10^3$ (left) and $\kappa = 10^5$ (middle). The blue boxplots show the results when all parameters are ideally set. In the right plot, I explore the distribution of ROC curves that generated the AUC values for the first and last boxplots in the middle plot.

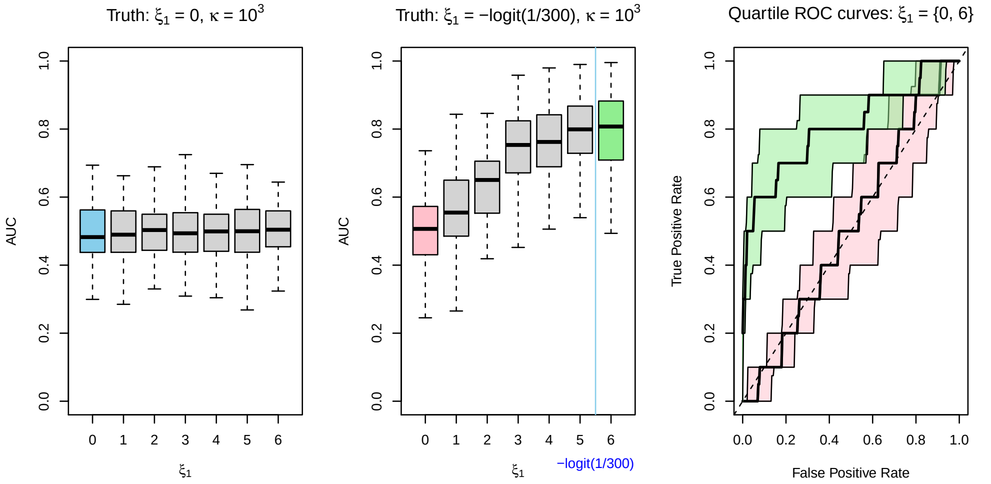

Figure 3.3: These boxplots depict the performance of the SB model in my third simulation study where I vary ξ_1 and fit my model to 100 data sets simulated from two different models where $\xi_1 = 0$ (left) and $\xi_1 = -\text{logit}(1/300)$ (middle). The blue boxplot shows the results when all parameters are ideally set. In the right plot, I explore the distribution of ROC curves that generated the AUC values for the first and last boxplots in the middle plot.

Chapter 4

Block-Wise Latent Genotypes

Recombination events that occur along chromosomes during reproduction can create more variation across individuals globally but can also impose less variation across genetic markers locally. This non-random association of adjacent markers introduces strong correlation in typical genome-wide association study data sets. In this chapter, I present a model for de-correlating blocks of the genome at a time by replacing the markers within each block with an independent continuous latent genotype that is estimated using the observed marker data and their spatial positions in a simultaneous auto-regressive model. I explore fitting a model that exploits the response variable and the observed genotypes simultaneously to estimate and select the significant latent genotypes and apply my method to the hypertension trait in the Genetic Analysis Workshop (GAW) 18 data set.

4.1 Methods

Researchers have recently been aiming to increase the power to detect significant markers in single association tests by combining the signals within groups such as gene sets; however, to maximize the benefit of these analyses we must also account for biases stemming from the typically strong patterns of correlation between neighboring markers due to linkage disequilibrium [84]. Simultaneous auto-regressive (SAR) models are especially useful for explaining the similarity between the observations collected from spatially close locations or subjects [75] Since the average correlation between markers is inversely proportional to the distance between them [5], my objective is to exploit SAR models in a data pre-processing step to replace short contiguous blocks of correlated markers with block-wise independent latent genotypes for subsequent analyses.

4.1.1 Block Definition

As an optional first step, I consider applying an algorithm such as one described in Section 1.1.1, CLUSTAG [4], to obtain a set of tag SNPs that can represent all the known SNPs in a chromosomal region, subject to the constraint that all SNPs must have a squared correlation $R^2 > \rho$ with at least one tag SNP, where ρ is specified by the user. The default choice of ρ in the program that ships with the software, 0.8, leads to a useful subset of representative SNPs that may still be strongly correlated with each other without having any pair be almost perfectly correlated with each other, i.e., $|R| \leq .9$. I then break a chromosome into blocks such that any two adjacent SNPs are in the same block if they lie within ζ units of genomic distance of each other.

Since larger values of ζ result in a larger number of SNPs in each block, the average decay rate of the relationship between the average magnitude of correlation between SNPs in a block and the genomic distance between them decreases as a function of ζ. For reference, the lower and upper quartiles of the distribution of pair-wise distances between *adjacent* SNPs on the longest chromosome analyzed in the WTCCC data set in Section 2.5 are 849 and 9,781; for the GAW18 data set analyzed in Section 4.5, they are 1,334 and 7,584. Although the range of appropriate choices for ζ may depend on the data set being analyzed, in practice I propose selecting a value of $\zeta \leq 10^4$ that strikes a balance between preserving a relatively strong inverse relationship between genomic distance and average magnitude of correlation and producing a computationally feasible number of blocks.

4.1.2 Simultaneous Autoregressive Model

Given a set of p SNPs for the i^{th} individual denoted by X_i, I define a corresponding set of independent random variables, “latent genotypes”, denoted by Z_i that have a multivariate normal (MVN) probability distribution parameterized by a mean vector μ and a diagonal covariance matrix Σ with entries τ^2, i.e.,

$$Z_i \overset{\text{ind}}{\sim} \mathsf{MVN}(\mu, \Sigma). \tag{4.1}$$

I now introduce spatially correlated latent genotypes denoted by U_i using the SAR modeling framework; given a matrix, B, of spatial weights, $B_{ij} \geq 0$ that encode the spatial proximity between a pair of SNPs such that $B_{jj} = 0$, I have

$$U_i = BU_i + Z_i. \tag{4.2}$$

Defining $C = (I - B)^{-1}$, the prior distribution on Z_i in Equation 4.1 induces the following distribution on U_i through the SAR model in Equation 4.2:

$$U_i \overset{\text{ind}}{\sim} \mathsf{MVN}(C\mu, C\Sigma C^\top) \tag{4.3}$$

The spatial proximity measures in B affect both the expected value and the covariance structure of the spatially correlated latent genotypes, U_i; moreover, in the trivial case where B is a matrix of zeros, I have that $C = I_p$ and so $U_i = Z_i$. I now propose the following model to establish a connection between X_i, U_i and Z_i:

$$X_{ij} \mid U_{ij} \overset{\text{ind}}{\sim} \mathsf{Binomial}(2, \text{logit}^{-1}\left[U_{ij}\right]) \tag{4.4}$$

Through this formulation, I treat an individual's observed SNP data in X_i as being a censored version of their spatially correlated latent genotypes in U_i. Since U_i is in turn defined as a function of itself, B, and Z_i, I can use 4.2 to re-write 4.4 and to obtain:

$$X_{ij} \mid Z_i \overset{\text{ind}}{\sim} \mathsf{Binomial}\big(2, \text{logit}^{-1}\big[C_j^\top Z_i\big]\big) \tag{4.5}$$

Paralleling the well-known inverse relationship between the average magnitude of correlation and the genomic distance between SNPs [5], I define the spatial weight between

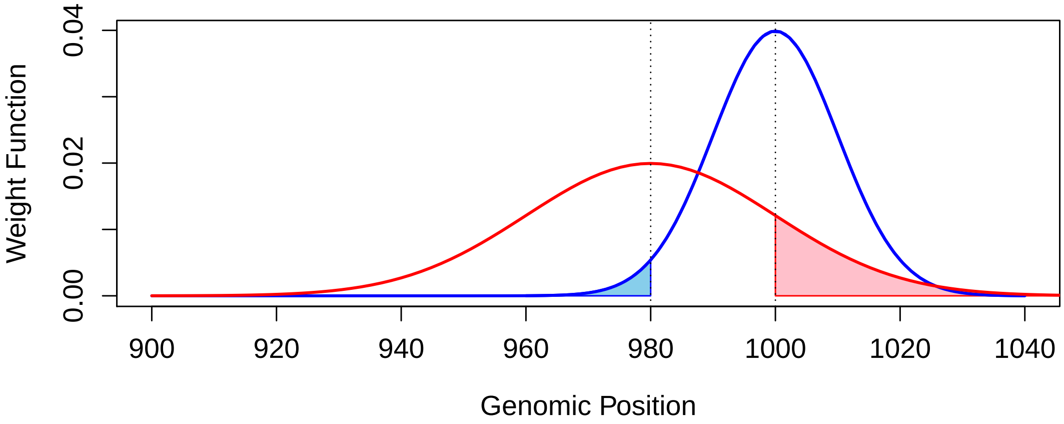

Figure 4.1: Spatial weight example: for the j^{th} SNP at position $s_j = 980$ with $\phi_j = 20$ and the k^{th} SNP at position $s_k = 1{,}000$ with $\phi_k = 10$, I obtain, $B_{jk} = 0.18$.

the j^{th} and k^{th} SNPs at genomic locations s_j and s_k to be

$$B_{jk} = \Phi\left(\frac{-|s_j - s_k|}{\phi_j}\right) + \Phi\left(\frac{-|s_j - s_k|}{\phi_k}\right),$$

where $\Phi(\cdot)$ is the cumulative distribution function of a standard normal random variable and ϕ_j and ϕ_k are tuning parameters that encode the strength of the influence of neighboring SNPs on the j^{th} and k^{th} spatially correlated latent genotypes. For a given SNP with genomic position, s_j, the radius of spatial influence from neighboring SNPs grows as a function of ϕ_j. By requiring that each $\phi_j \leq \zeta/3$, the spatial weight between the j^{th} SNP and SNPs from other blocks becomes so negligible that B and C exhibit block-wise diagonal structure. Although I recommend this as a useful upper bound for computational convenience, in Section 4.2, I provide more guidelines for choosing these tuning parameters. In contrast to the spatial boost model's gene weights defined in Section 2.1.2, I allow for potentially every SNP to have a different corresponding value of ϕ_j; this flexibility can better accommodate for recombination hotspots that may be scattered across the genome.

Due to the large size of typical GWAS data sets, it would be impractical to estimate a unique latent genotype for each SNP for each individual. To overcome this computational obstacle, I instead propose replacing the full vector of p latent genotypes, Z_i, with a subset of K block-wise latent genotypes, $\tilde{Z}_i$ that best summarize, through the SAR modeling framework, the observed DNA fingerprint of X_i. In particular, given a configuration of blocks such that b_j denotes the block to which the j^{th} SNP belongs, I define:

$$Z_{ij} = \tilde{Z}_{ib_j} + \delta_j.$$

For extra modeling flexibility, I allow Z_{ij} to deviate from $\tilde{Z}_{ib_j}$ through a residual term, δ_j; however, to ensure identifiability of this model, I add the constraint that $\sum_{k \in b_j} \delta_k = 0$. I model each block independently of the rest under the assumption that the ϕ_j's have been chosen in such a way that C is block-wise diagonal. Now, for an arbitrary block, b, letting $\mathbf{z}_{ib}$ denote the vector of Z_{ik}'s such that $k \in b$, letting $\delta_{-|b|}$ denote the vector of deviations for block b without its last element, and defining $\mathbf{v}_{ib} = \{\tilde{Z}_{ib}, \delta_{-|b|}\}$, I can write the i^{th} individual's $|b|$ latent genotypes within block b as a linear mapping, T, from $\mathbf{v}_{ib}$:

$$\mathbf{z}_{ib} = T\mathbf{v}_{ib}. \tag{4.6}$$

To enforce the relationship in 4.6, I have that for the collection of SNPs in block b, T is a square matrix of size $|b|$ defined according to the following pattern:

$$T = \begin{pmatrix} 1 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & 0 & \cdots & 1 \\ 1 & -1 & -1 & -1 & \cdots & -1 \end{pmatrix}.$$

For a given block of the genome, I describe how to choose the tuning parameters and hyper-parameters of my model such that the naturally occurring minor allele frequencies and linkage disequillibrium patterns are preserved in 4.5 in Section 4.2. Once I determine the values of the ϕ's, μ, and Σ for a given block, the corresponding inverse mapping, T^{-1}, applied to the appropriate block of latent genotypes, $\mathbf{z}_{ib}$, determines a prior distribution on that block's values of $\mathbf{v}_{ib}$; omitting the subscripts on μ and Σ that denote the sub-vector or sub-matrix corresponding to block b for simplicity, the prior distribution induced by 4.1 and4.6 is as follows:

$$\mathbf{v}_{ib} \mid \phi, \mu, \Sigma \overset{\text{ind}}{\sim} \mathsf{Normal}(T^{-1}\mu, T^{-1}\Sigma T^{-\top}) \tag{4.7}$$

Letting $\tilde{Z}_i^\top$ denote the i^{th} individual's collection of block-wise latent genotypes, I now propose the following Bayesian model for a set of n binary response variables y:

$$\begin{aligned} y_i \mid \tilde{Z}_i^\top\gamma &\overset{\text{ind}}{\sim} \mathsf{Bernoulli}\big(\text{logit}^{-1}\big[\tilde{Z}_i^\top\gamma\big]\big) \\ \gamma_b \mid \theta_b, \sigma^2 &\overset{\text{ind}}{\sim} \mathsf{Normal}(0, \sigma^2[\theta_b\kappa + 1 - \theta_b]) \\ \theta_b &\overset{\text{ind}}{\sim} \mathsf{Bernoulli}(\psi) \\ \sigma^2 &\sim \mathsf{IG}(\nu, \lambda) \end{aligned} \tag{4.8}$$

The model in 4.8 corresponds to simple Bayesian variable selection with a continuous spike-and-slab prior distribution for the effect sizes, γ, of the block-wise latent genotypes, $\tilde{Z}_i^\top$, where the latent variables, θ, indicate which *blocks* are significantly associated to the response variable. Similar to the spatial boost model, I use an inverse-gamma prior distribution for the variance term, σ^2, in the spike-and-slab prior with hyper-parameters ν and λ, and use the EMBFDR in practice to choose an appropriate value of κ. Each block independently has a prior probability of ψ of being associated to the trait of interest.

A fundamental difference between my approach and other methods is that instead of modeling y_i *given* a linear combination of the i^{th} individual's covariates, e.g., $X_i^\top\beta$ for some $\beta \in \mathbb{R}^p$, I use y_i and $X_i^\top$ simultaneously, along with the priors on the $\mathbf{v}_{ib}$'s, to estimate $\tilde{Z}_i^\top$ and then model y_i *given* $\tilde{Z}_i^\top\gamma$. Although I describe an algorithm in Section 4.3 for fitting the model in this way, I also explore a simpler idea in the comparison simulation study in Section 4.4.2 where I estimate $\tilde{Z}_i^\top$ only using $X_i^\top$ in a pre-processing step and then perform single block association tests in an analysis similar to the usual single marker tests. In Chapter 5, I merge the ideas here and in Chapter 2 to extend the prior on the θ_j's in this model to prioritize the blocks that lie close to relevant regions of a chromosome.

4.2 Selecting Prior Tuning Parameters

Equation 4.5 defines the relationship between the i^{th} individual's observed SNP data, X_i, and the corresponding unobserved latent genotypes, Z_i. It is important to choose the hyper-parameters for the prior distribution on Z in such a way that preserves certain naturally occurring relationships between SNPs. In particular I want to choose ϕ, μ, and Σ in a way that not only minimizes, for each j, the discrepancy between the expected value of the j^{th} SNP and the ideal value based on a Hardy-Weinberg model, i.e., two times that SNP's minor allele frequency, π_j, but also preserves the linkage disequillibrium patterns known to exist in the population under investigation. Since a possible measure of LD is the correlation coefficient, I can accomplish both objectives by selecting the hyper-parameters so that the first two moments of X correspond to the known biology. To get started, in a simplifying assumption I set the diagonal elements of Σ equal to a common $\tau^2 = 1$; I explore the effect of this choice in the simulation study in Section 4.4.2. Then in a manner similar to coordinate descent, I otherwise iteratively update the values of μ and ϕ for a given block so as to preserve its natural MAF and LD patterns.

It is noteworthy that for the following sections, I will once again assume that the ϕ_j's have been chosen so that C is block-wise diagonal. This way, I am able to simplify the overall algorithm by operating on one block at a time, sequentially updating only the elements of μ and ϕ that affect that particular block. For simplicity, I once again omit the subscripts on μ and Σ that would denote the sub-vector or sub-matrix corresponding to a particular block that I update.

4.2.1 Preserving Minor Allele Frequencies

Also omitting the subscripts to denote the i^{th} individual and the b^{th} block of a given chromosome, let $\mathbf{z}$ denote an arbitrary individual's vector of latent genotypes inside a block, b, where $|b| > 1$, and let D denote the corresponding b^{th} block of the block-wise diagonal C. Assuming that the j^{th} SNP is one of the SNPs inside the b^{th} block, I now apply the law of total expectation to X_{ij}:

$$\begin{aligned}\mathbb{E}_{X_{ij}}[X_{ij}] &= \mathbb{E}_{\mathbf{z}}[\mathbb{E}_{X_{ij}|\mathbf{z}}[X_{ij}]] \\ &= \mathbb{E}_{\mathbf{z}}[2\,\text{logit}^{-1}(D_j^\top \mathbf{z})] \\ &= 2\,\mathbb{E}_{\mathbf{z}}[\text{logit}^{-1}(D_j^\top \mathbf{z})]\end{aligned} \tag{4.9}$$

Letting $\pi(\cdot) = \text{logit}^{-1}(\cdot)$, $\tilde{\mu}_j = \mathbb{E}_{\mathbf{z}}[D_j^\top \mathbf{z}] = D_j^\top \mu$ and $\tilde{\Sigma}_{jk} = [D \Sigma D^\top]_{jk}$ for simplicity, to select the hyper-parameters in a way that preserves the minor allele frequencies of the SNPs, I now desire to minimize the difference between $\mathbb{E}_{\mathbf{z}}[\text{logit}^{-1}(D_j^\top \mathbf{z})]$ and π_j. Using the new notation, I will approximate $\mathbb{E}_{\mathbf{z}}[\pi(D_j^\top \mathbf{z})]$ using a Taylor expansion; first I write

$$\pi(D_j^\top \mathbf{z}) \approx \pi(\tilde{\mu}_j) + \pi^{(1)}(\tilde{\mu}_j)\,(D_j^\top \mathbf{z} - \tilde{\mu}_j) + \frac{1}{2}\,\pi^{(2)}(\tilde{\mu}_j)\,(D_j^\top \mathbf{z} - \tilde{\mu}_j)^2 + \dots \tag{4.10}$$

Then taking the expectation of both sides of 4.10 with respect to $\mathbf{z}$, I have

$$\mathbb{E}_{\mathbf{z}}[\pi(D_j^\top \mathbf{z})] \approx \pi(\tilde{\mu}_j) + \frac{1}{2}\,\pi^{(2)}(\tilde{\mu}_j)\,\tilde{\Sigma}_{jj} + \dots \tag{4.11}$$

Given values of ϕ, Σ, and an initial set of values for the $\tilde{\mu}_j$'s, I use a Newton's method algorithm to update the values of $\tilde{\mu}_j$ to minimize the difference between the first two terms in 4.11 and π_j for each $j \in b$; the objective function for the j^{th} SNP is given by

$$f(\tilde{\mu}_j) = \pi(\tilde{\mu}_j) + \frac{1}{2} \cdot \pi^{(2)}(\tilde{\mu}_j)\,\tilde{\Sigma}_{jj} - \pi_j.$$

The first derivative of the objective function with respect to the input $\tilde{\mu}_j$ is then

$$f^{(1)}(\tilde{\mu}_j) = \pi^{(1)}(\tilde{\mu}_j) + \frac{1}{2}\,\pi^{(3)}(\tilde{\mu}_j)\,\tilde{\Sigma}_{jj}.$$

Combining these equations, I iteratively update the value of $\tilde{\mu}_j$ until convergence using the standard update equation:

$$\tilde{\mu}_j^{(t+1)} = \tilde{\mu}_j^{(t)} - \frac{f(\tilde{\mu}_j^{(t)})}{f^{(1)}(\tilde{\mu}_j^{(t)})}.$$

Finally, after achieving convergence, I obtain new estimates for μ by computing $D^{-1}\tilde{\mu}$.

4.2.2 Preserving Linkage Disequillibrium Patterns

To preserve the LD pattern in block b, I focus on choosing the ϕ_j's for $j \in b$ so that the expected covariance between any two SNPs in that block matches a given pattern from either an external biological database with LD information or simply from the sample covariance matrix. By applying the law of total covariance to the j^{th} and k^{th} SNPs in block b, X_{ij} and X_{ik}, I first have the following:

$$\operatorname{cov}(X_{ij}, X_{ik}) = \mathbb{E}_{\mathbf{z}}[\operatorname{cov}_{X_{ij},X_{ik}|\mathbf{z}}(X_{ij}, X_{ik})] + \operatorname{cov}_{\mathbf{z}}(\mathbb{E}_{X_{ij}|\mathbf{z}}[X_{ij}], \mathbb{E}_{X_{ik}|\mathbf{z}}[X_{ik}]). \tag{4.12}$$

Since I assume that the X_{ij}'s are conditionally independent given $\mathbf{z}$, the first term on the right hand side of 4.12 is zero for $j \neq k$; and otherwise $\mathbb{E}_{\mathbf{z}}[\operatorname{cov}_{X_{ij},X_{ik}|\mathbf{z}}(X_{ij}, X_{ik})] = \mathbb{E}_{\mathbf{z}}[\operatorname{var}_{X_{ij}|\mathbf{z}}(X_{ij})]$. The first derivative of the inverse logit function, $\pi^{(1)}(\cdot) = \pi(\cdot)(1 - \pi(\cdot))$, and so I can write $\operatorname{var}_{X_{ij}|\mathbf{z}}(X_{ij}) = 2\pi^{(1)}(D_j^{\top}\mathbf{z})$. Another Taylor expansion for $\pi^{(1)}(D_j^{\top}\mathbf{z})$ gives the following:

$$\pi^{(1)}(D_j^{\top}\mathbf{z}) \approx \pi^{(1)}(\tilde{\mu}_j) + \pi^{(2)}(\tilde{\mu}_j)\,(D_j^{\top}\mathbf{z} - \tilde{\mu}_j) + \frac{1}{2}\,\pi^{(3)}(\tilde{\mu}_j)\,(D_j^{\top}\mathbf{z} - \tilde{\mu}_j)^2 + \ldots \tag{4.13}$$

Taking the expectation of both sides of 4.13 with respect to $\mathbf{z}$, I can write

$$\mathbb{E}_{\mathbf{z}}[\pi^{(1)}(D_j^\top \mathbf{z})] \approx \pi^{(1)}(\tilde{\mu}_j) + \frac{1}{2}\,\pi^{(3)}(\tilde{\mu}_j)\,\tilde{\Sigma}_{jj} + \ldots \tag{4.14}$$

I will use the first two terms on the right hand side of 4.14 in my approximation to the first term of 4.12. As for the second term in 4.12, I first simplify the original expression and note that $\mathrm{cov}_{\mathbf{z}}(\mathbb{E}_{X_{ij}|\mathbf{z}}[X_{ij}], \mathbb{E}_{X_{ik}|\mathbf{z}}[X_{ik}]) = 4\,\mathrm{cov}_{\mathbf{z}}(\pi(D_j^\top \mathbf{z}), \pi(D_k^\top \mathbf{z}))$. By the definition of covariance, $\mathrm{cov}_{\mathbf{z}}(\pi(D_j^\top \mathbf{z}), \pi(D_k^\top \mathbf{z})) = \mathbb{E}_{\mathbf{z}}[\pi(D_j^\top \mathbf{z})\,\pi(D_k^\top \mathbf{z})] - \mathbb{E}_{\mathbf{z}}[\pi(D_j^\top \mathbf{z})]\,\mathbb{E}_{\mathbf{z}}[\pi(D_k^\top \mathbf{z})]$. Using another Taylor expansion to approximate the first term in this expression, and using the previously derived approximations for each factor of the second term, and then simplifying the result, I obtain the following:

$$\begin{aligned}\mathrm{cov}_{\mathbf{z}}(\mathbb{E}_{X_{ij}|\mathbf{z}}[X_{ij}], \mathbb{E}_{X_{ik}|\mathbf{z}}[X_{ik}]) \approx&\\ \pi(\tilde{\mu}_j)\,\pi(\tilde{\mu}_k) + \frac{1}{2}\,\pi^{(2)}(\tilde{\mu}_j)\,\pi(\tilde{\mu}_k)\,\tilde{\Sigma}_{jj} &+ \frac{1}{2}\,\pi(\tilde{\mu}_j)\,\pi^{(2)}(\tilde{\mu}_k)\,\tilde{\Sigma}_{kk}\\ &+ \pi^{(1)}(\tilde{\mu}_j)\,\pi^{(1)}(\tilde{\mu}_k)\,\tilde{\Sigma}_{jk} + \ldots\end{aligned} \tag{4.15}$$

Combining equations 4.12 through 4.15, and using the indicator function I[·] that returns one if its contents are true and zero otherwise, I obtain a final simplified approximation for the covariance between the j^{th} and k^{th} SNPs:

$$\begin{aligned}\mathrm{cov}(X_{ij}, X_{ik}) \approx\ & 4\,\pi^{(1)}(\tilde{\mu}_j)\,\pi^{(1)}(\tilde{\mu}_k)\,\tilde{\Sigma}_{jk} - \pi^{(2)}(\tilde{\mu}_j)\,\pi^{(2)}(\tilde{\mu}_k)\,\tilde{\Sigma}_{jj}\,\tilde{\Sigma}_{kk}\\ &+ \mathrm{I}[j=k]\,\Big[2\,\pi^{(1)}(\tilde{\mu}_j) + \pi^{(3)}(\tilde{\mu}_j)\,\tilde{\Sigma}_{jj}\Big]\end{aligned} \tag{4.16}$$

To preseve the naturally occuring LD patterns in a given block, I now optimize the values of ϕ by iterating over a grid of possible values and selected the configuration that minimizes the mean squared error between the known LD pattern in that block and the

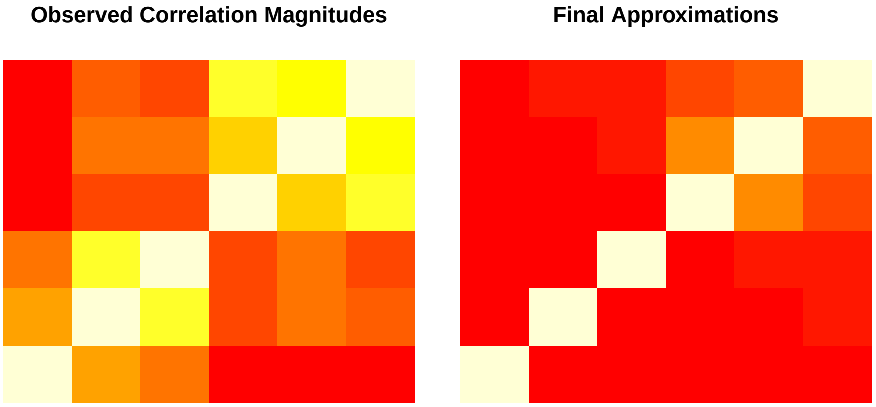

Figure 4.2: An example of my approximation to an observed correlation structure.

approximation that derives from 4.16. Exploiting the idea of coordinate descent, e.g. as in [92], I iteratively update the values of μ using the algorithm in Section 4.2.1 and the values of ϕ as described here, for each block, until convergence. I show an example of a correlation pattern in a block of six SNPs on chromosome 1 in the GAW18 data set and our approximation using the above methods in Figure 4.2. The images shown are heat maps of the absolute values of the observed correlation matrix, and the absolute values of my approximations based on 4.16. Rather than capture every off-diagonal contour that appears in the sample covariance matrix, my approximation tends to conservatively set most of the off-diagonal entries to be relatively small values.

4.3 Model Fitting and Inference

After selecting the hyper-parameters for the prior distributions on the latent genotypes, I derive a model fitting procedure for 4.8 similar to the one described in Chapter 2 for the spatial boost model. It is noteworthy that a fully Bayesian approach is impractical for the latent genotype model because of the introduction of so many additional latent parameters for each individual. Instead, taking inspiration from my previous empirical observation that the final estimates of posterior probabilities of association obtained from the Gibbs sampler on a binary trait did not significantly differ from the conditional probabilities of association obtained by the EM algorithm, I propose focusing on the same EM filtering pipeline for the latent genotype model as well. In this model however, I need to add a step to also fit all of the block-wise latent genotypes, $\tilde{Z}$, and all of the block-wise deviations of the SNPs, δ. Thus, in the new procedure, at each step of the overall filter, I alternate between: (1) fitting $\tilde{Z}$ and δ given the rest of the model parameters using a Newton's method algorithm, and (2) fitting γ, θ, σ^2 given the newly updated values of $\tilde{Z}$ and δ using an EM algorithm. I continue to update all variables until either the values stop changing or the algorithm passes a certain maximum number of iterations.

4.3.1 Estimating Block-Wise Latent Genotypes and Deviations

Letting $\tilde{\mu}_b$ and $\tilde{\Sigma}_b$ now respectively be the sub-vector or sub-matrix of $T^{-1}\mu$ or $T^{-1}\Sigma T^{-\top}$ that corresponds to the b^{th} block of latent genotypes, letting $\tilde{C}_b$ denote the b^{th} block of the block-wise diagonal C transformed such that $[\tilde{C}_b]_{ij} = [C_b]_{ij} - [C_b]_{i|b|}$ for $i = 1, \ldots, |b|$ and $j = 1, \ldots, |b| - 1$, letting $\tilde{\delta}$ denote the first $|b| - 1$ deviations within block b, and letting $\mathbf{d}_b$ denote the vector of row sums of $\tilde{C}_b$, I define the log joint distribution, ℓ, given all observed data and parameters, up to a constant:

$$\begin{aligned}
\ell(\cdot) \propto{} & \sum_{i=1}^{n}\{y_i \tilde{Z}_i^\top \gamma - \log[1 + \exp(\tilde{Z}_i^\top \gamma)]\} - \frac{1}{2}\sum_{i=1}^{n}\sum_{b=1}^{K}[\mathbf{v}_{ib} - \tilde{\mu}_b]^\top \tilde{\Sigma}_b^{-1}[\mathbf{v}_{ib} - \tilde{\mu}_b] \\
& + \sum_{i=1}^{n}\sum_{j=1}^{p}\{X_{ij}\log\pi(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta}) + (2 - X_{ij})\log[1 - \pi(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta})]\} \\
& - \frac{1}{2\sigma^2}\sum_{b=1}^{K}\gamma_b^2\left(\frac{\theta_b}{\kappa} + 1 - \theta_b\right) - \sum_{b=1}^{K}\theta_b \text{logit}(\psi) - \left(\nu + \frac{K}{2} + 1\right)\log\sigma^2 - \frac{\lambda}{\sigma^2}. \quad (4.17)
\end{aligned}$$

4.3.1.1 Estimating $\tilde{Z}_{ib}$

Since each individual has a different set of block-wise latent genotypes, I update each $\tilde{Z}_{ib}$ one at a time, conditional on all of the other model parameters. The terms in 4.17 that depend on $\tilde{Z}_{ib}$ are the component from the data likelihood, the component from the block-wise prior distribution on $\mathbf{v}_{ib}$, and the component from the SNPs within block b. Isolating these components, I define an objective function, $f(\tilde{Z}_{ib})$, that, when maximized, provides an optimal value for $\tilde{Z}_{ib}$.

$$\begin{aligned} f(\tilde{Z}_{ib}) = {} & y_i \tilde{Z}_{ib}\gamma_b - \log[1+\exp(\tilde{Z}_i^\top \gamma)] - \frac{1}{2}[\mathbf{v}_{ib} - \tilde{\mu}_b]^\top \tilde{\Sigma}_b^{-1}[\mathbf{v}_{ib} - \tilde{\mu}_b] \\ & + \sum_{j\in b}\{X_{ij}\log \pi(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta}) + (2 - X_{ij})\log[1-\pi(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta})]\} \end{aligned} \tag{4.18}$$

Taking the first derivative of this objective function, I have the following:

$$\begin{aligned} f^{(1)}(\tilde{Z}_{ib}) = {} & y_i\gamma_b - \pi(\tilde{Z}_i^\top \gamma)\gamma_b - (\tilde{Z}_{ib} - \tilde{\mu}_{b_1})[\tilde{\Sigma}_b^{-1}]_{11} \\ & - \frac{1}{2}\sum_{j=2}^{|b|}(\delta_{j-1} - \tilde{\mu}_{b_j})[\tilde{\Sigma}_b^{-1}]_{j1} + \sum_{j\in b}\{X_{ij}\mathbf{d}_{b_j} - 2\pi(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta})\mathbf{d}_{b_j}\} \end{aligned} \tag{4.19}$$

I maximize 4.18 by using a Newton's method algorithm to find a zero of 4.19. To set up the algorithm properly I take the second derivative of the objective function and obtain:

$$f^{(2)}(\tilde{Z}_{ib}) = -\pi^{(1)}(\tilde{Z}_i^\top \gamma)\gamma_b^2 - [\tilde{\Sigma}_b^{-1}]_{11} - 2\sum_{j\in b}\pi^{(1)}(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta})\mathbf{d}_{b_j}^2 \tag{4.20}$$

I now iteratively update each $\tilde{Z}_{ib}$ by repeatedly applying the standard update equation, $\tilde{Z}_{ib}^{(t+1)} = \tilde{Z}_{ib}^{(t)} - f^{(1)}(\tilde{Z}_{ib}^{(t)})/f^{(2)}(\tilde{Z}_{ib}^{(t)})$, until convergence.

4.3.1.2 Estimating δ

Unlike the block-wise latent genotypes, each individual shares the same deviation δ_j for their j^{th} SNP, so I borrow information from all samples to update δ conditional on all of the other model parameters. I use another Newton's method algorithm to update δ one block at a time; for an arbitrary block, b, the terms in 4.17 that depend on the first $|b| - 1$

deviations for that block, $\tilde{\delta}$, are the components from the block-wise prior distribution on $\mathbf{v}_{ib}$, and the components from the SNPs within block b for all individuals. Since I require that in a given block, b, $\sum_{k \in b} \delta_k = 0$, I implicitly obtain the optimal value for $\delta_{|b|}$ when optimizing $\tilde{\delta}$ for each block. Isolating the relevant components, I define a new objective function, $f(\tilde{\delta})$, that, when maximized, provides an optimal value for $\tilde{\delta}$:

$$
\begin{aligned}
f(\tilde{\delta}) = &-\frac{1}{2}\sum_{i=1}^{n}[\mathbf{v}_{ib} - \tilde{\mu}_b]^\top \tilde{\Sigma}_b^{-1}[\mathbf{v}_{ib} - \tilde{\mu}_b] \\
&+ \sum_{i=1}^{n}\sum_{j \in b}\{X_{ij}\log \pi(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta}) + (2 - X_{ij})\log[1 - \pi(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta})]\} \quad (4.21)
\end{aligned}
$$

To optimize the elements of $\tilde{\delta}$ *jointly*, I need to compute the gradiant of 4.21, $\nabla f(\tilde{\delta})$, and the Jacobian matrix for the gradient, $\mathbf{J}$. I start by noting that:

$$
\begin{aligned}
[\nabla f(\tilde{\delta})]_k = &-\frac{n}{2}(\tilde{\delta}_k - \tilde{\mu}_{b_{k+1}})[\tilde{\Sigma}_b^{-1}]_{k+1,k+1} - \frac{1}{2}\sum_{i=1}^{n}\sum_{j \neq k}(\mathbf{v}_{ib_j} - \tilde{\mu}_{b_j})[\tilde{\Sigma}_b^{-1}]_{j,k+1} \\
&+ \sum_{i=1}^{n}\sum_{j \in b}\{X_{ij}[\tilde{C}_b]_{jk} - 2\pi(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta})[\tilde{C}_b]_{jk}\} \quad (4.22)
\end{aligned}
$$

The Jacobian of 4.22 is symmetric so that $[\mathbf{J}]_{jk} = [\mathbf{J}]_{kj}$; moreover, for $j \neq k$:

$$
[\mathbf{J}]_{jk} = -\frac{n}{2}\tilde{\delta}_j[\tilde{\Sigma}_b^{-1}]_{j,k+1} - 2[\tilde{C}_b]_{jk}\sum_{i=1}^{n}\sum_{j' \in b}\pi^{(1)}(\tilde{Z}_{ib}\mathbf{d}_{b_j} + [\tilde{C}_b]_j^\top \tilde{\delta})[\tilde{C}_b]_{j'k}. \quad (4.23)
$$

The diagonal elements of $\mathbf{J}$ each have one additional term: $-\frac{n}{2}[\tilde{\Sigma}_b^{-1}]_{k+1,k+1}$. Combining this fact with equations 4.22 and 4.23 to compute $\nabla f(\tilde{\delta})$ and $\mathbf{J}$, I fit $\tilde{\delta}$ for a given block by repeatedly applying the update equation, $\tilde{\delta}^{(t+1)} = \tilde{\delta}^{(t)} - [\mathbf{J}^{(t)}]^{-1}\nabla f(\tilde{\delta}^{(t)})$, until convergence.

4.3.2 Estimating θ, σ^2 and γ

After updating the $\tilde{Z}$ and δ values, I apply the same type of EM algorithm as described in Chapter 2 for the spatial boost model to fit the remaining model parameters. Just as before, at the t-th iteration of the procedure, for the E-step I need to compute and store $\langle\theta_b\rangle^{(t)} \doteq \mathbb{E}_{\theta \,|\, y;\gamma^{(t)},(\sigma^2)^{(t)}}[\theta_b]$. For the latent genotype model,

$$\langle\theta_b\rangle = \mathbb{P}(\theta_b = 1 \,|\, y, \gamma, \sigma^2) = \frac{\mathbb{P}(\theta_b = 1, \gamma_b \,|\, \sigma^2)}{\mathbb{P}(\theta_b = 0, \gamma_b \,|\, \sigma^2) + \mathbb{P}(\theta_b = 1, \gamma_b \,|\, \sigma^2)},$$

and so

$$\operatorname{logit}\langle\theta_b\rangle = \log \frac{\mathbb{P}(\theta_b = 1, \gamma_b \,|\, \sigma^2)}{\mathbb{P}(\theta_b = 0, \gamma_b \,|\, \sigma^2)} = -\frac{1}{2}\log\kappa - \frac{\gamma_b^2}{2\sigma^2}\left(\frac{1}{\kappa} - 1\right) + \operatorname{logit}(\psi) \tag{4.24}$$

To update γ and σ^2 I once again employ conditional maximization steps; from (4.17) I see that the update for σ^2 follows immediately from the mode of an inverse gamma distribution conditional on $\gamma^{(t)}$:

$$(\sigma^2)^{(t+1)} = \frac{\frac{1}{2}\sum_{b=1}^{K}\left(\gamma_b^{(t)}\right)^2\left(\frac{\langle\theta_b\rangle^{(t)}}{\kappa} + 1 - \langle\theta_b\rangle^{(t)}\right) + \lambda}{\frac{K+1}{2} + \nu + 1}. \tag{4.25}$$

The terms in (2.4) that depend on γ come from the log likelihood of y and from the expected prior on γ, $\gamma \sim N(0, \Sigma^{(t)})$, where

$$\Sigma^{(t)} = \operatorname{Diag}\left(\frac{\sigma^2}{\langle\theta_b\rangle^{(t)}/\kappa + 1 - \langle\theta_b\rangle^{(t)}}\right).$$

Since updating γ is equivalent here to fitting a ridge regularized logistic regression, I exploit the usual iteratively reweighted least squares (IRLS) algorithm [60]. Setting $\varpi^{(t)}$ as the vector of expected responses with $\varpi_i^{(t)} = \operatorname{logit}^{-1}(\tilde{Z}_i^\top \gamma^{(t)})$ and $W^{(t)} = \operatorname{Diag}(\varpi_i^{(t)}(1 - \varpi_i^{(t)}))$

as the variance weights, the update for γ is then

$$\gamma^{(t+1)} = (\tilde{Z}^\top W^{(t)} \tilde{Z} + (\Sigma^{(t)})^{-1})^{-1} \big(\tilde{Z}^\top W^{(t)} \tilde{Z}\gamma^{(t)} + \tilde{Z}^\top (y - \varpi^{(t)})\big), \tag{4.26}$$

where I substitute $(\sigma^2)^{(t)}$ for σ^2 in the definition of $\Sigma^{(t)}$.

4.3.3 Model Selection

I continue to filter out blocks of SNPs at a time in the model fitting procedure outlined above until I reach either a desired number of blocks or the performance of the model deteriorates beyond a certain point as measured by a metric such as the posterior predictive loss. After reaching the final step of the EM filter, I decide on a final model by thresholding the final estimates of each $\langle\theta_b\rangle$. For my simulation and case studies I use a threshold of 0.5 so that any SNP in a block such that $\langle\theta_b\rangle > 0.5$ is included in the final model.

4.4 Empirical Studies

To assess the utility of the latent genotype model and to understand the effect that different choices of ζ and τ^2 have on the model, I conduct a simulation study using a real data set provided by the Genetic Analysis Workshop 18 on hypertension. For simplicity, I focus only on the 31,812 SNPs that remain on chromosome 1 in 141 unrelated individuals after applying the usual data filters described in Section 1.1.1. I describe the full data set in more detail in Section 4.5.

4.4.1 Simulation Study Details

The goal of my study is to compare the performance of the latent genotype model to the simplest, most popular alternative model, i.e. single SNP tests. However, in addition to the single SNP tests, I also consider a simpler alternative to the model fitting procedure described in Section 4.3 where I estimate $\tilde{Z}$ once using X and y, i.e. I stop the EM filtering procedure after the first iteration and retain all blocks, and then run the single SNP tests on the block-wise latent genotypes instead of the raw minor allele frequency counts.

To account for different configurations of SNP positions and LD patterns, for each of 10 batches, I first randomly sample a contiguous subset of 1,000 SNPs on chromosome 1 from a random subset of 100 unrelated individuals in the GAW18 data set. Given a randomly sampled subset of SNP data, I simulate 10 replicates of response variables, y, from a multiple logistic regression model where the only non-zero coefficients occur at the 10 SNPs located at indices $\{100, 200, \ldots, 1{,}000\}$ and are drawn from the standard normal distribution.

4.4.2 Comparison Simulation Study

For each of the 100 simulated data sets, I apply the single SNP tests on X, and then for each configuration where $\zeta \in \{1{,}000, 5{,}000\}$ and $\tau^2 \in \{.01, .25, 1\}$ I apply the single SNP tests on the estimates of $\tilde{Z}$ after the first iteration of the EM filtering procedure (without removing any blocks), and I use the full model fitting procedure described in the previous section to filter the number of blocks down to 5. Figure 4.3 shows a comparison of the distributions of block sizes when using the different choices of ζ; the smaller value of 1,000 results in a majority of blocks that contain only one SNP and no block containing more than 8 SNPs whereas the larger value of 5,000 allows for a larger proportion of blocks to contain multiple SNPs and a maximum size closer to 40.

To assess the performance of each model, I compute the area under the Receiving Operating Characteristic (ROC) curve using my knowledge of the true and false positives in each data set. When applying the single SNP tests, I use the ranking of $-\log_{10}(\text{p-value}_j)$ in decreasing order to generate the points on the ROC curve. For the latent genotypes model, at each step of the EM filtering procedure I use the ranking of the final estimates of $\langle \theta_b \rangle$ in decreasing order to generate the points on the ROC curve; in this case, I treat the selection of a block of SNPs simply as the selection of all of the SNPs inside that block.

As shown in Figure 4.4, the distribution of AUC values for all configurations of the latent genotypes model is comparable and often overall higher to that of the single SNP

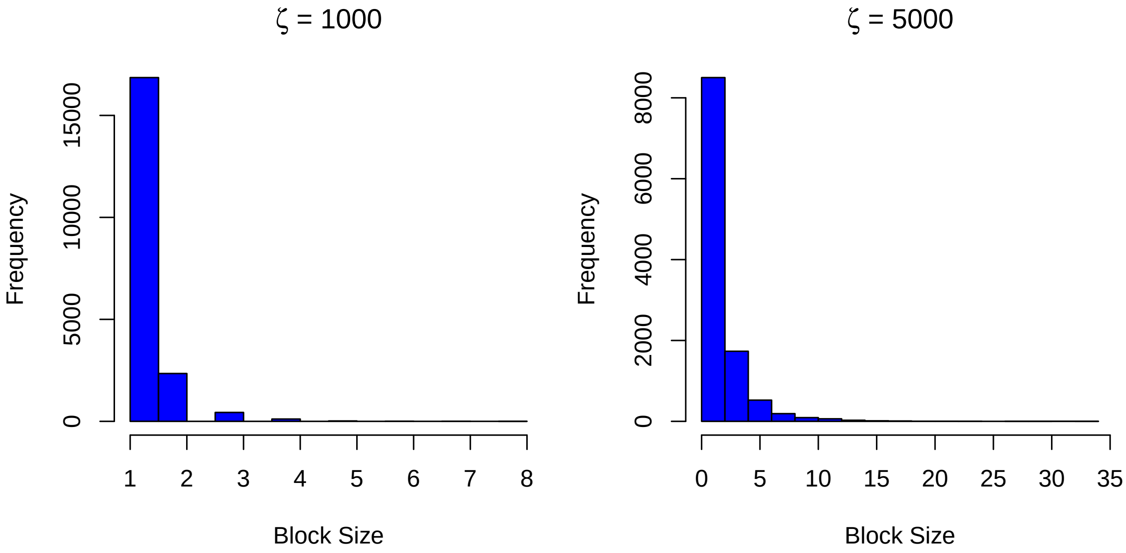

Figure 4.3: Comparison of distributions of block lengths on chromosome 1 for different ζ.

tests model. While the latent genotypes model seems robust to different choices for the underlying variance of the latent genotypes, τ^2, there is a clear improvement in model performance when selecting a relatively larger value for ζ and allowing for a larger proportion of blocks to contain multiple SNPs. This result is promising not only because the crux of my method is to aim at improving model performance by replacing short contiguous blocks of *multiple* correlated SNPs with independent block-wise latent genotypes, but also because I observe that the latent genotypes models outperform the single SNP tests even when the response variables are simulated from a different model entirely based on X instead of $\tilde{Z}$.

4.5 Case Study

In this section I apply the latent genotypes model to analyze the binary trait of hypertension in the GAW18 data set. The raw data set contains measurements on 959 individuals from a longitudinal survey on blood pressure and hypertension. The SNP data for these individuals consists of genotypes assayed at Texas Biomed using the Illumina platform. In order to reduce the size of the data set, genotypic data are provided only for markers on the odd-numbered autosomes. After removing rare variants and SNPs that significantly deviate

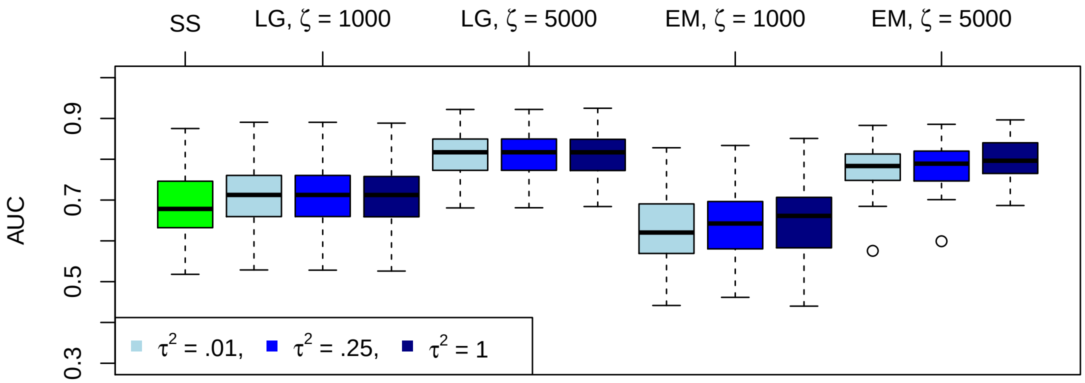

Figure 4.4: Results from the comparison simulation study. From left to right I show the distribution of AUC values for the single SNP tests on X (SS), and under a given configuration of (ζ, τ^2), for the single SNP tests on $\tilde{Z}$ after the first iteration of the EM filtering procedure (LG), and for the full model fitting procedure (EM).

from Hardy-Weinberg Equilibrium, I use CLUSTAG in an initial filtering step with the default $\rho = 0.8$ to obtain a total of 200,561 tag SNPs scattered across the first 11 odd-numbered chromosomes. For the response variable, I consider the union of the hypertension indicator variables measured at each of the four time points so that y_i denotes whether or not the i^{th} individual had hypertension at any point in the study. To abide by the assumption of independence across response variables, I consider only the 157 unrelated individuals in the study. Moreover, I remove a further 16 individuals due to missing genotype or phenotype information. My final filtered data set consists of 141 unrelated individuals, their corresponding y_i's, and their SNPs.

For the real data set, I apply single SNP tests and two variations of my latent genotypes model where $\zeta \in \{0, 5000\}$ and $\tau^2 = 0.5$. For computational convenience when fitting the the latent genotype model, I analyze each chromosome separately and fit all model parameters and latent variables at each step of the EM filtering procedure until the number of blocks is reduced to five. Then I build a final model by combining the thresholded values, I[$\langle\theta_b\rangle \geq 0.5$], from each chromosome where the $\langle\theta_b\rangle$'s are taken from the iteration of that chromosome's EM filter that has the smallest posterior predictive loss (PPL). The trivial choice of $\zeta = 0$ corresponds to running the EM filtering procedure on X instead of $\tilde{Z}$ whereas the larger choice of $\zeta = 5{,}000$ encourages a larger proportion of blocks that contain multiple SNPs as shown in the simulation study.

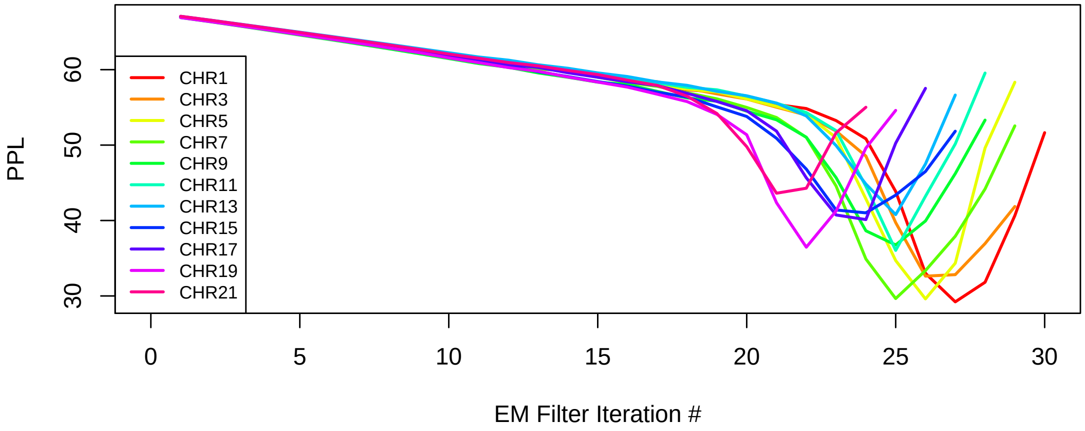

Figure 4.5: Comparison of the different PPL curves for GAW18 when using X.

In Figures 4.5 and 4.6, I show the paths of the PPL curves as a function of the EM filter iteration number. All curves generally exhibit the same behavior observed in Chapter 2's analysis of the WTCCC data set; the PPL decreases uniformly for the first several iterations of the filter and then sharply decreases for a few iterations before changing directions and increasing uniformly for the rest of the procedure. Interestingly, some of the PPL curves for the models where $\zeta = 5{,}000$, e.g. the curves for chromosomes 17, 19 and 21, do not exibit a pronounced global minimum. This can perhaps be attributed to the latent genotype model's flexibility in allowing the re-estimation of both $\tilde{Z}$ and γ at each step of the filter. If there are only a few significant blocks on a chromosome and if the EM filter works well to remove the insignificant blocks at each step of the filter, then it is reasonable to expect the PPL curve to continue to decrease even at the later stages of the procedure. Indeed, the only blocks included in my final model where $\zeta = 5{,}000$ lie on chromosome 19.

Tables 6.17 through 6.18 list the top 10 SNPs obtained by each of the methods. Using a simple Bonferroni correction on a starting significance level of .05 to address the multiple testing problem, the threshold for genome-wide significance on $-\log_{10}(\text{p-value}_j)$ is 6.60. None of the SNPs achieve this value in the single SNP tests; moreover, the problem of multicollinearity causes several SNPs that are highly correlated with each other to appear simultaneously in the top 10 ranking. The simple Bayesian variable selection model on X, i.e. the latent genotype model where $\zeta = 0$, avoids this problem; however using a threshold on $\langle \theta_b \rangle$ of 0.5, no SNPs are selected for inclusion in this model. Nevertheless, the top 10 ranking for this method includes two SNPs that lie in genes with independent associations with hypertension including GABRG3 (rs6606865) [44] and CSK (rs1378942) [93]. I select three blocks of SNPs on chromosome 19 with high conditional posterior probability of association in the latent genotype model where $\zeta = 5{,}000$. Each of these blocks contains only one SNP and lies between the 30 Mbp and 50 Mbp positions of chromosome 19. I do not observe any direct or independent association with hypertension by the genes in these blocks; therefore further experiements are necessary to determine whether or not there is any functional relationship between SLC17A7, DHDH, NUCB1 and hypertension.

4.6 Conclusions

In this chapter I presented a method to de-correlate contiguous blocks of SNPs by replacing them with a block-wise latent genotype after using a SAR modeling framework to relate the correlation structure between markers in a block to the genomic distance between them. In a comparison simulation study using real GWAS data, I found that the method outperformed the usual single SNP tests even when the simulated data were generated from a different model based on real genotypes observed in the GAW18 data set. In a case study on the GAW18 data set, I showed that the method can significantly change the pattern of signals in the standard Manhattan plot and highlight at least one region with a possible

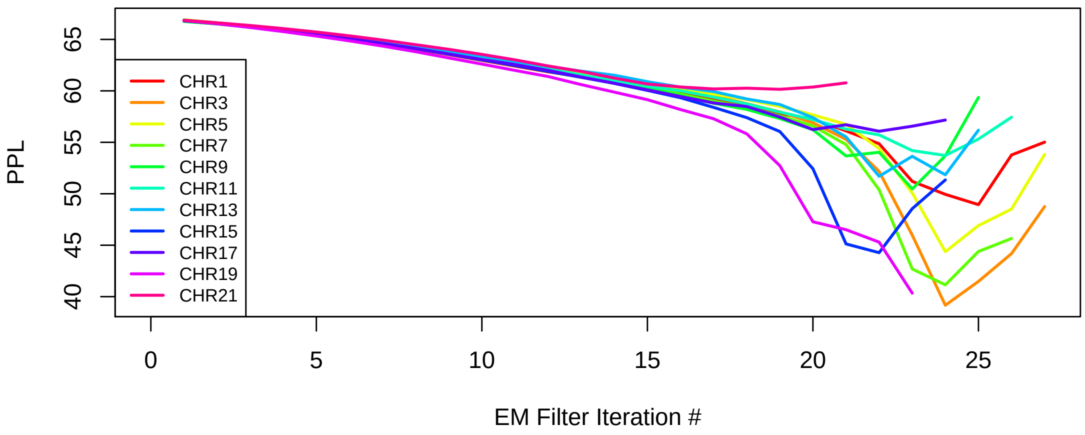

Figure 4.6: Comparison of the different PPL curves for GAW18 when using $\tilde{Z}$.

connection to the trait of interest. Further experiments are therefore needed to validate the markers that I selected for my final model; I could perhaps attribute my null result to a combination of factors such as the small sample size of 141 and the possibly oversimplifying assumption that all blocks of SNPs idependently have the same prior probability of association. In the next and final chapter of my thesis, I explore this direction by incorporating the prior from the spatial boost model on the block-wise latent genotypes.

Chapter 5

Application of Proposed Methods

In this final chapter, I incorporate the ideas from Chapters 2 and 4 and modify the spatial boost model to operate on latent genotype. Then, using all of my proposed methods, I re-analyze the two data sets on binary traits that were discussed in the previous chapters, the rheumatoid arthritis data set from the Wellcome Trust Case Control Consortium and the hypertension data set from the Genetic Analysis Workshop 18.

5.1 Spatial Boost Model on Latent Genotypes

Seeking to evolve the spatial boost model to a level that analyzes blocks of latent genotypes as hinted in Sections 2.6 and 4.6, I now extend the spatial boost model presented in Chapter 2 to include the methods presented in Chapter 4 for de-correlating and analyzing blocks of SNPs. My goal is to update the model in 2.1 to operate on latent genotypes in such a way that preserves the hierarchy between SNPs and genes.

5.1.1 Model Definition

The spatial boost model on latent genotypes is a natural extension of 4.8 where the prior distribution on the indicator variable, θ_b, that indicates whether or not the b^{th} block is associated with the response variable is modified to include a spatial boost term:

$$\theta_b \overset{\text{ind}}{\sim} \mathsf{Bernoulli}(\xi_0 + \xi_1 \tilde{\mathbf{w}}_b(\phi)^\top \mathbf{r}). \tag{5.1}$$

In this prior distribution, $\tilde{\mathbf{w}}_b^\top$ refers to a $G \times 1$ vector of gene weights for the b^{th} block.

The rest of the tuning parameters, ξ_0, ξ_1 and $\mathbf{r}$ retain their original definitions from the spatial boost model, i.e., they represent, respectively, the logit of the prior probability of association of a block that is not close to any relevant genes, the maximum log odds increase in the prior probability of association of a block due to the spatial boost term, and a $G \times 1$ vector of relevance values that quantify the relationship between each gene and the trait of interest.

5.1.2 Defining Block Weights

To control how much a gene can contribute to the prior probability of association for a *block* based on the gene's length and distance to a block, I combine the information contained in the usual gene weights of the SNPs in that block. Recall that given a range parameter $\phi > 0$ for a given window of the genome, I define the gene weight $w_{j,g}$ for a gene g that spans genomic positions g_l to g_r, and the j-th marker at genomic position s_j as

$$w_{j,g} = \int_{g_l}^{g_r} \frac{1}{\sqrt{2\pi\phi^2}} \exp\left\{ -\frac{(x - s_j)^2}{2\phi^2} \right\} \mathrm{d}x.$$

I now set

$$\tilde{w}_{b,g} = \frac{1}{|b|} \sum_{j \in b} w_{j,g}.$$

By taking the simple average of the gene weights of the SNPs in a given block, each SNP contributes equally to the definition of $\tilde{w}_{b,g}$. Thus blocks that have a relatively larger number of SNPs that lie close to genes will have a relatively larger block gene weight. Because my model operates on short contiguous blocks of a chromosome, the mean gene weight of the SNPs in a given block provides a reasonable summary of that block's spatial proximity to the gene. Figure 5.1 shows a visualzation of the process of computing the block-wise gene weights in a mock example with 10 SNPs (green) on a chromosome with five genes (purple). In the top plot, I center a Gaussian curve with standard deviation $\phi = 0.25$ at each SNP's position and shade in the area underneath from each gene's starting and ending positions. The middle plot shows the resulting gene weights from this process as dots (blue) placed at a height equal to their value on the vertical axis. Following the methodology in Chapter 4 and using a threshold of $\zeta = 1$, I define four blocks: $\{1, 2, 3, 4\}$, $\{5\}$, $\{6, 7\}$, and $\{8, 9, 10\}$. The bottom plot shows the final block-wise gene weight for each block. As before, I normalize the gene weight contributions to θ_b in (5.1), $\tilde{\mathbf{w}}_b(\phi)^\top \mathbf{r}$, such that $\max_j \mathbf{w}_j(\phi)^\top \mathbf{r} = 1$. This way, as before, it is possible to compare estimates of ξ_1 across different gene weight and relevance schemes.

5.1.3 Preserving the Distribution of Spatial Boost Terms

During the model fitting procedure, I employ the usual Expectation Maximization filtering algorithm to iteratively remove the bottom quartile of blocks from the model until either I reach a final desired number of blocks or the performance of the model is optimized according to a metric such as the posterior predictive loss. I note that at the beginning of the filter, the distribution of spatial boost terms, $\tilde{\mathbf{w}}_b(\phi)^\top \mathbf{r}$, is right-skewed with the majority of values being close to zero. If the strength of the prior distribution on θ_b is too strong, then the distribution of spatial boost terms for the blocks that remain in the EM filter can become incorrectly left-skewed if the blocks with relatively larger values of the spatial boost term rank higher than they should.

To account for this possibility, I take a precaution to preserve the initial distribution of spatial boost terms throughout the filter. Letting $F_0(\cdot)$ denote the cumulative probability distribution function (CDF) of all of the spatial boost terms before any filtering, letting $[\tilde{\mathbf{w}}_b(\phi)^\top \mathbf{r}]_{(k)}$ denote the k^{th} order statistic of the spatial boost terms, and letting K_t denote the number of blocks in the filter *after* the t^{th} iteration, I consider the following transformation:

$$[\tilde{\mathbf{w}}_b(\phi)^\top \mathbf{r}]_{(k)}^{(t+1)} = F_0^{-1}\left(\frac{k}{K_t}\right). \tag{5.2}$$

In other words, before running the $(t+1)^{\text{th}}$ iteration of the filter, I use the ranking of

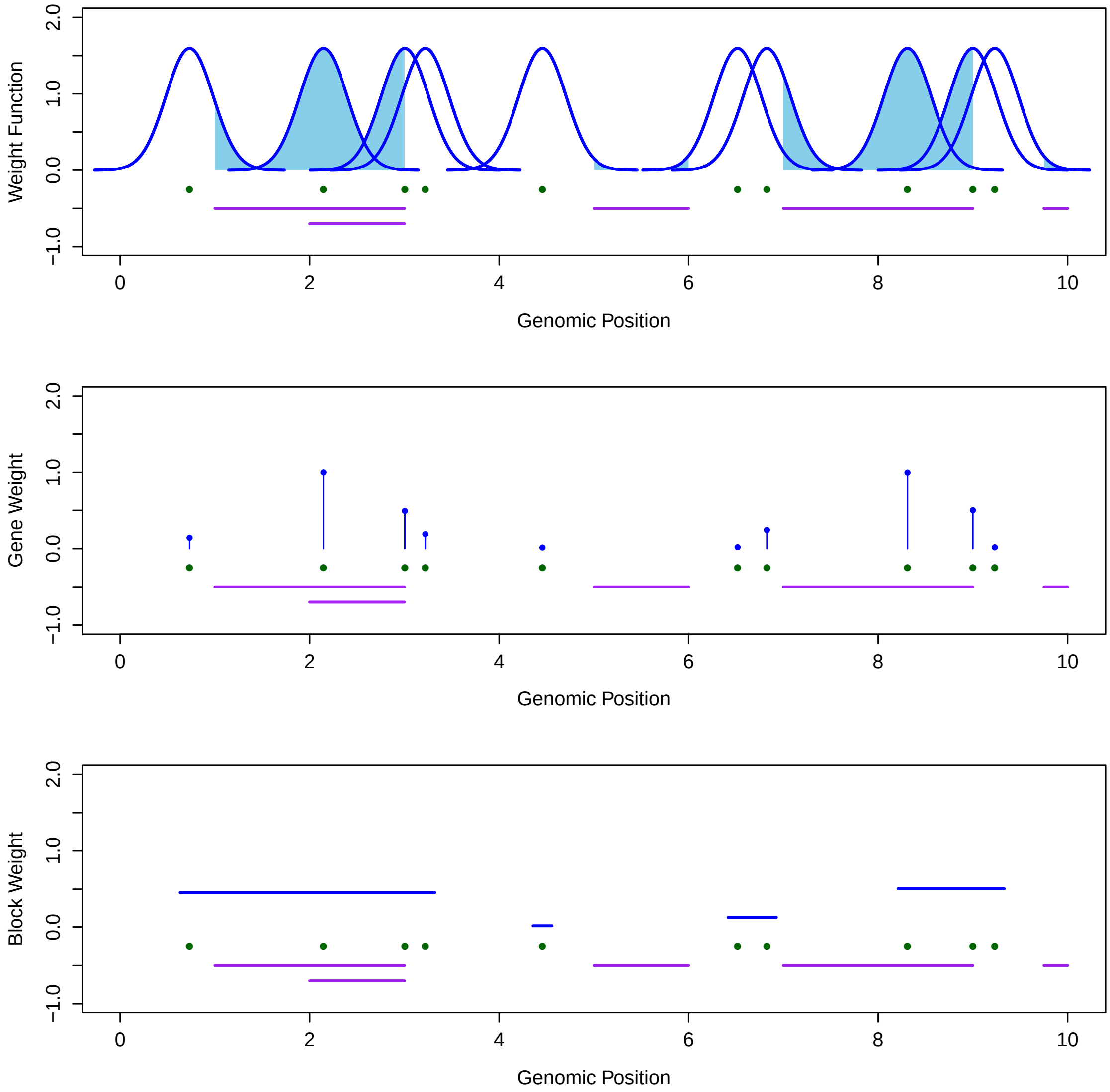

Figure 5.1: Visualization of the computation of block-wise spatial boost terms.

the remaining spatial boost terms from the t^{th} iteration to determine the order statistics. Then I assign a value for the spatial boost term to each remaining block equal to the quantile of F_0 that corresponds to that block's rank in the order statistics.

Figure 5.2 shows a mock example of this problem and its subsequent correction. The probability distribution of all of the spatial boost terms before any filtering, f_0, is shown on the left. The middle plot shows a possible problematic probability distribution of the spatial boost terms after the t^{th} iteration of the filter that exhibits left-skewness as opposed to the desired right-skewness. The transformed distribution based on (5.2) is shown in the right plot; the simple transformation enforces that the shape of the distribution of spatial boost terms remains the same at each step of the EM filter.

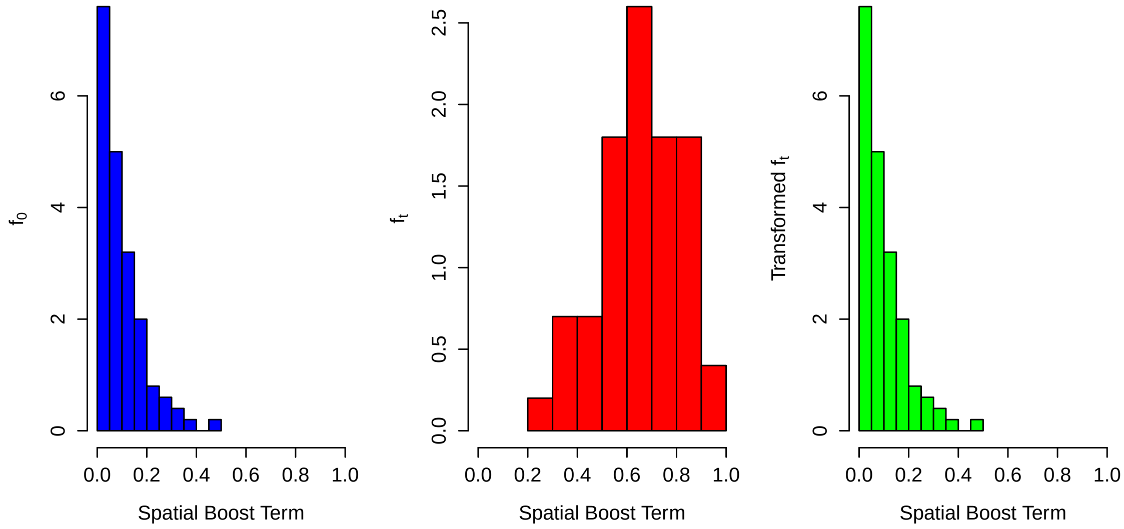

Figure 5.2: Visualization of the spatial boost term correction.

5.2 Case Studies

5.2.1 WTCCC Data Set: Rheumatoid Arthritis

In a re-analysis of the rheumatoid arthritis data set that was first presented in Chapter 2, I use the above block-wise spatial boost prior in the latent genotypes model and set $\zeta = 5{,}000$, $\xi_0 = -8$ and $\xi_1 = 4$. This configuration of model tuning parameters allows for both a relatively larger proportion of blocks that contain more than one SNP as explored in 4.4.2 and a moderate gene boost effect as explored in 2.5. I again use the MalaCards relevance scores for $\mathbf{r}$ and, for computational convenience, I analyze each chromosome separately after first removing the rare variants with MAF $< 5\%$ and then running the CLUSTAG algorithm on the remaining markers with $\rho = 0.8$ to select a subset of tag SNPs that best summarizes each chromosome's observed genotypic variation.

At the initial step of the EM filter for each chromosome the average number of blocks is 5,271 with a standard deviation of 2,485 blocks. Setting $\tau^2 = 0.5$ for the prior variance of the latent genotypes for each individual and setting values for the prior parameters of μ and ϕ (and thus Σ) in accordance with the guidelines established in 4.2, I then apply the model fitting procedure described in 4.3 and iteratively (i) fit the block-wise latent genotypes for each individual and all remaining model parameters, and (ii) rank all blocks in decreasing order according to their final estimates of $\mathbb{E}[\theta_b | \hat{\beta}_{\text{EM}}, \hat{\sigma}^2_{\text{EM}}, y]$ and remove the bottom 25% of blocks. I repeat this iterative process until I either reach a desired final number of blocks or the posterior predictive loss reaches a minimum value.

Keeping track of the PPL at each iteration of the EM filter, I remove the bottom quartile of blocks on each chromosome until the number of remaining blocks is five or less. Unlike the original analysis that used the spatial boost model on the raw genotypes, the PPL curves for each chromosome in the new analysis that included the latent genotypes each have a distinct minimum value that occurs well before the natural end of the EM filter. As shown in Figure 5.3, the location of the natural end of the filter differs depending on the chromosomes; however, the filter usually terminates around the twentieth iteration with the minimum PPL value occurring somewhere between the tenth and fifteenth iterations.

Figures 5.5 and 5.5 respectively depict the signals of association observed in single marker tests and in the spatial boost model on latent genotypes. A full numerical summary of the top results in these plots are provided in Tables 6.1 and 6.20. Although 15 SNPs pass the standard Bonferroni corrected genome-wide significance threshold in the single marker tests, only two SNPs pass the threshold for inclusion in the final spatial boost model on latent genotypes. For my final model I select a block from chromosome 7 that contains one of the genome-wide significant SNPs in the single marker tests, rs10262109, and a block from chromosome 10 that contains another genome-wide significant SNP, rs1733717.

The overall ranking of blocks based on the conditional expected values of θ_b at the optimal EM filtering steps of each chromosome is notably different from the ranking of

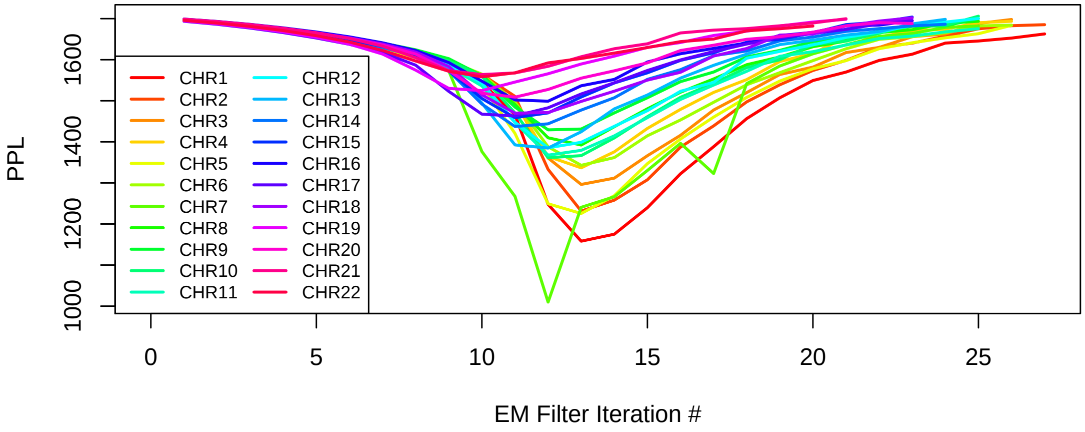

Figure 5.3: Posterior Predictive Loss curves for each chromosome in the WTCCC data set.

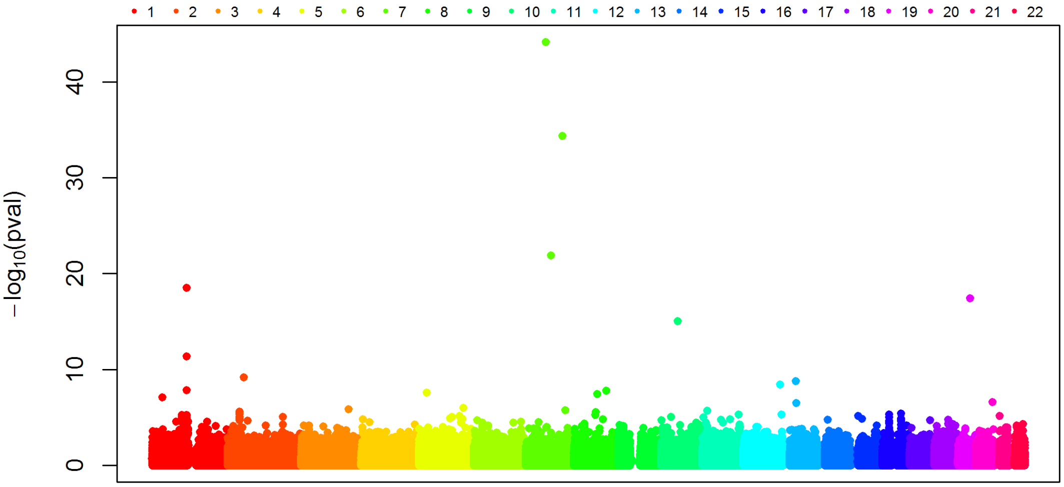

Figure 5.4: Visualization of the observed signal in the single marker tests for the WTCCC data set.

individual SNPs in the analogous spatial boost models from Chapter 2. In particular, as evident in the previous rankings of top SNPs in Tables 6.6 and 6.12, neither the analogous spatial boost model with a non-informative gene relevance vector, $\mathbf{r} = \mathbf{1}$, nor the analogous model with an informative gene relevance vector taken from MalaCards scores highlights the region of chromosome 10 that contains rs1733717. This is especially interesting because evidence of association betweeen this SNP and rheumatoid arthritis has already been replicated in at least two additional studies. In this way, the analysis on block-wise latent genotypes has improved upon the initial sparser signal observed in the spatial boost model on the raw genotypes to further narrow down the regions of interest.

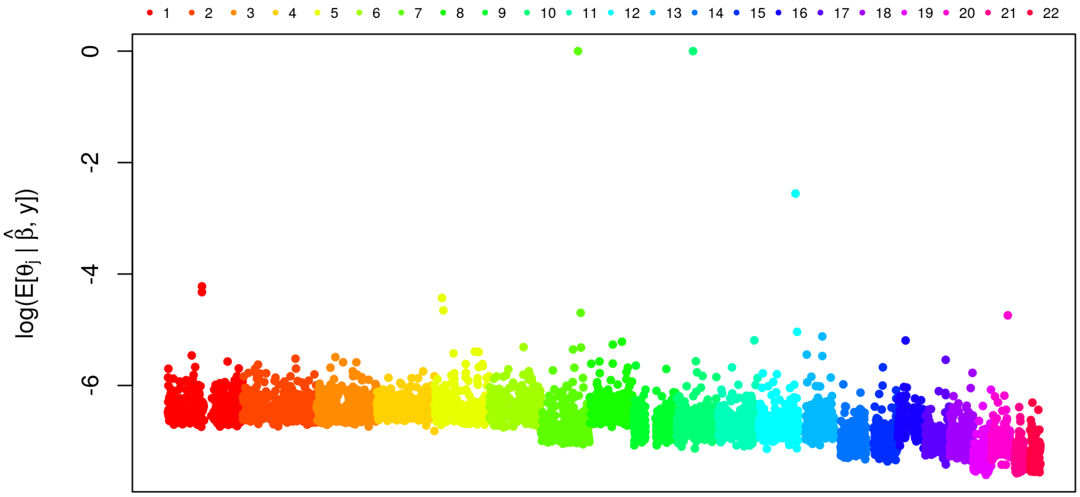

Figure 5.5: Visualization of the observed signal in the spatial boost model on latent genotypes for the WTCCC rheumatoid arthritis data set.

The extraction of an important signal in an region of the genome that was not highlighted by the previous spatial boost models on the raw genotypes in the WTCCC data set shows that my upgraded model that incorporates the latent genotypes is another useful, complementary tool for GWAS.

5.2.2 GAW18 Data Set: Hypertension

In a re-analysis of the hypertension data set that was first presented in Chapter 4, I again use the above block-wise spatial boost prior in the latent genotypes model and set $\zeta = 5{,}000$ but instead set $\xi_0 = -4$ and $\xi_1 = 2$. This configuration of model tuning parameters takes into account the fewer number of chromosomes and markers in the GAW18 data set and again allows for both a relatively larger proportion of blocks that contain more than one SNP and a moderate gene boost effect. I again use the hypertension MalaCards relevance scores for $\mathbf{r}$ and, for computational convenience, I analyze each chromosome separately after first removing the rare variants with MAF $< 5\%$ and then running the CLUSTAG algorithm on the remaining markers with $\rho = 0.8$ to select a subset of tag SNPs that best summarizes each chromosome's observed genotypic variation.

For this data set, at the initial step of the EM filter for each chromosome the average number of blocks is 6,119 with a standard deviation of 2,958 blocks. Setting $\tau^2 = 0.5$ for the prior variance of the latent genotypes for each individual and setting values for the prior parameters of μ and ϕ (and thus Σ) in accordance with the guidelines established in 4.2, I then apply the model fitting procedure described in 4.3 and iteratively (i) fit the block-wise latent genotypes for each individual and all remaining model parameters, and (ii) rank all blocks in decreasing order according to their final estimates of $\mathbb{E}[\theta_b|\hat{\beta}_{\text{EM}}, \hat{\sigma}^2_{\text{EM}}, y]$ and remove the bottom 25% of blocks. I repeat this iterative process until I either reach a desired final number of blocks or the posterior predictive loss reaches a minimum value.

Just as in the re-analysis of the WTCCC data set, I keep track of the PPL at each iteration of the EM filter and remove the bottom quartile of blocks on each chromosome until the number of remaining blocks is five or less. As shown in Figure 5.6, similar to the original analysis that used simple Bayesian variable selection on the latent genotypes, the PPL curves for each chromosome in the new analysis that included the spatial boost prior monotonically decrease until close to the end of the filter where they begin to increase.

When utilizing the spatial boost prior, I observe a new ranking of the top 10 blocks as reported in Table 6.21 where none of the blocks qualify for inclusion in the final model. Interestingly, the spatial boost model on latent genotypes does not select or even highlight the region on chromosome 19 that was picked up by the previous analyses. Instead, the top 10 ranking of blocks in the spatial boost model on latent genotypes for hypertension includes some of the other unselected blocks that appear in the results of essentially an analagous spatial boost model where $\xi_1 = 0$ (see Table 6.19). In the previous analysis,

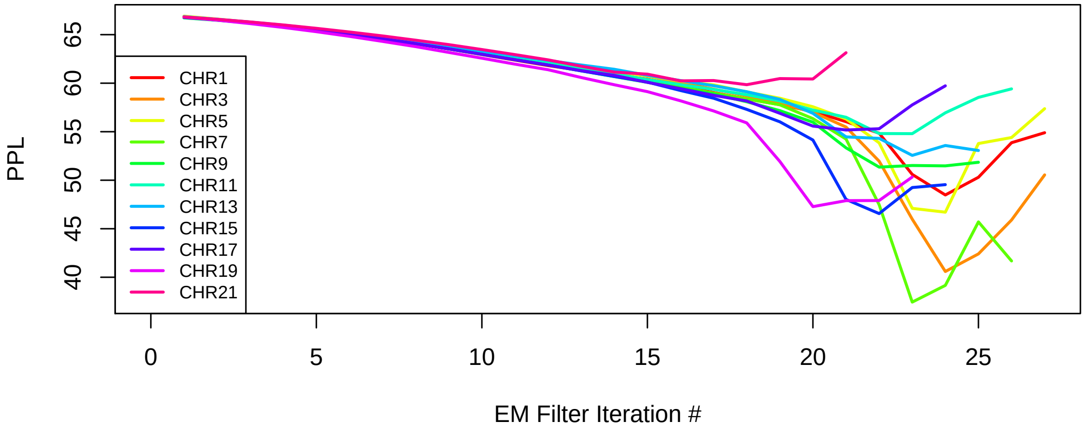

Figure 5.6: Posterior Predictive Loss curves for each chromosome in the GAW18 data set.

although I selected blocks from a region of chromosome 19, I could not find any known connection between the genes in those blocks and hypertension. The absence of these regions in the new final model is due to the spatial boost prior that prioritizes the blocks which are close to relevant genes. Whereas inclusion of the latent genotypes helped to further refine the results in the analysis of the WTCCC data set, the inclusion of the spatial boost prior helped to remove a region of false positives in the GAW18 data set.

5.3 Conclusions

The overarching primary objective in the research that I have conducted for my thesis is to complement and advance the state-of-the-art techniques for analyzing the statistical relationship between a set of genetic markers and a population trait of interest in genome-wide association studies. To that end, I have succeeded in developed a useful series of hierarchical Bayesian models that exploit external biological knowledge to first de-correlate short contiguous blocks of markers and to then analyze the resulting independent block-wise latent genotypes jointly in such a way that prioritizes the blocks that are close in genomic distance to relevant genes or other features on the chromosomes.

Unlike some typical models for this problem that model a trait given the observed genotypes as fixed data, the main methodological contribution of my work is the simultaneous modeling of both a set of observed markers and a trait of interest as functions of unobserved latent genotypes. This contribution makes it possible to first pool information from both X and y in a SAR modeling framework in the estimation of independent latent genotypes and to then use that set of fitted latent genotypes in the selection of regions of the genome that are associated with the trait of interest. Overcoming the typical high computational costs that are required when using Bayesian models, the main computational contribution of my work is a computationally efficient pipeline for fitting my models. Moreover, I observed an interesting and consistent phenomena when fitting the spatial boost model to both quantitative and qualitative traits where the final model selected after running the EM algorithm matched the final model selected after running the Gibbs sampler.

In several simulation studies, I have demonstrated the superior performance of my models relative to other state-of-the-art models in terms of the observed ratio of true positives to false positives, and I have shown that the computational speed-ups that I exploit in my EM algorithms can make my models, depending on the size of the data set, faster to fit than even the single marker tests. In two independent case studies on real GWAS data concerning the presence or absence of rheumatoid arthritis and hypertension, I demonstrated the utility of my method by filtering a set of several hundred thousand genetic markers down to at most two interesting blocks. For rheumatoid arthritis, my final model selects two blocks where one of them contains a SNP that has multiple replicated associations to the trait. For hypertension, my final model does not select any blocks; however, this is not surprising because the data set has a small sample size of 141 individuals and even the single marker tests fail to identify any genome-wide significant SNPs.

I have also shown that even running just one component of my overall method, i.e. the spatial boost model described in Chapter 2, or the block-wise latent genotype model described in Chapter 4, can yield promising results on real GWAS data sets. Altogether my contributions to this problem form a useful complementary set of techniques that can be efficiently used to identify causal genetic markers. To share these techniques with the scientific community, I have developed an R package that implements all the methods described in this thesis. For now, this package is available at the public github repository `http://github.com/ianjstats/spatialboost`, but I plan to submit it to CRAN (the Comprehensive R Archive Network, the main official repository for R packages) as soon as I have a suitable publication to reference.

Chapter 6

Appendix

6.1 Tables of Results in Chapter 2

Table 6.1: Genome-wide significant SNPs obtained by single marker tests when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	-log10(p-value)	Gene
rs4718582	7	66.95	0.08	44.15	—
rs10262109	7	121.44	0.06	34.35	—
rs12670243	7	82.97	0.06	21.88	—
rs6679677	1	114.30	0.14	18.54	—
rs664893	19	39.76	0.12	17.44	—
rs1733717	10	54.29	0.07	15.03	—
rs1230666	1	114.17	0.18	11.36	MAGI3
rs903228	2	53.69	0.06	9.20	—
rs9315704	13	40.14	0.17	8.78	LHFP
rs1169722	12	121.64	0.17	8.44	—
rs2488457	1	114.42	0.24	7.85	AP4B1-AS1
rs16874205	8	107.20	0.06	7.78	—
rs962087	5	24.89	0.16	7.59	—
rs2943570	8	76.51	0.34	7.46	—
rs10914783	1	34.27	0.06	7.12	CSMD2

Table 6.2: Top 15 SNPs at optimal EM filtering step using $\xi_0 = -8$ and $\xi_1 = 0$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\mathbb{E}(\theta_j \vert \cdot)$	Gene	Length (kpb)
rs4718582	7	66.95	0.08	1.00	—	—
rs10262109	7	121.44	0.06	1.00	—	—
rs1028850	13	40.94	0.47	0.26	LINC00598	133.9
rs903228	2	53.69	0.06	0.14	—	—
rs664893	19	39.76	0.12	0.11	—	—
rs765534	11	91.59	0.12	0.05	—	—
rs9371407	6	156.26	0.15	0.05	—	—
rs577483	1	36.21	0.13	0.04	CLSPN	37.8
rs1169722	12	121.64	0.17	0.04	—	—
rs11218078	11	120.84	0.18	0.04	GRIK4	326.0
rs10004440	4	80.27	0.24	0.04	—	—
rs6679677	1	114.30	0.14	0.04	—	—
rs6940680	6	123.34	0.35	0.04	CLVS2	67.5
rs977375	2	56.98	0.45	0.04	—	—
rs17724320	16	84.04	0.35	0.03	—	—

Table 6.3: Top 15 SNPs based on posterior samples using $\xi_0 = -8$ and $\xi_1 = 0$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\hat{\mathbb{P}}(\theta_j = 1 \vert y)$	Gene	Length (kpb)
rs4718582	7	66.95	0.08	1.00	—	—
rs10262109	7	121.44	0.06	1.00	—	—
rs1028850	13	40.94	0.47	0.02	LINC00598	133.9
rs903228	2	53.69	0.06	0.02	—	—
rs12670243	7	82.97	0.06	0.01	—	—
rs664893	19	39.76	0.12	0.01	—	—
rs6679677	1	114.30	0.14	0.01	—	—
rs765534	11	91.59	0.12	0.01	—	—
rs577483	1	36.21	0.13	0.01	CLSPN	37.8
rs9371407	6	156.26	0.15	0.01	—	—
rs11218078	11	120.82	0.18	0.01	GRIK4	326.0
rs4260892	8	34.41	0.14	0.00	—	—
rs10144971	14	30.33	0.16	0.00	PRKD1	351.2
rs10004440	4	80.27	0.24	0.00	—	—
rs1906470	10	63.01	0.10	0.00	—	—

Table 6.4: Top 15 SNPs at optimal EM filtering step using $\mathbf{r} = \mathbf{1}$, $\xi_0 = -8$, and $\xi_1 = 1$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\mathbb{E}(\theta_j\|\cdot)$	Gene	Length (kpb)
rs4718582	7	66.95	0.08	1.00	—	—
rs664893	19	39.76	0.12	1.00	—	—
rs10262109	7	121.44	0.06	1.00	—	—
rs6679677	1	114.30	0.14	1.00	—	—
rs11983481	7	69.97	0.07	0.05	AUTS2	26.6
rs17100164	14	33.59	0.08	0.04	NPAS3	465.9
rs3773050	3	29.55	0.10	0.04	RBMS3	729.1
rs9819844	3	143.44	0.25	0.04	SLC9A9	269.9
rs3848052	13	92.08	0.49	0.03	GPC5	1,468.6
rs7752758	6	88.87	0.12	0.03	CNR1	1.4
rs4545164	9	77.85	0.40	0.03	—	—
rs17671833	16	7.43	0.09	0.03	RBFOX1	380.6
rs10765177	10	129.68	0.22	0.03	CLRN3	15.1
rs17675094	16	82.95	0.40	0.03	CDH13	554.2
rs982932	15	61.06	0.36	0.03	RORA	741.0

Table 6.5: Top 15 SNPs based on posterior samples using $\mathbf{r} = \mathbf{1}$, $\xi_0 = -8$, and $\xi_1 = 1$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\hat{\mathbb{P}}(\theta_j = 1\|y)$	Gene	Length (kpb)
rs4718582	7	66,954,061	0.08	1.00	—	—
rs10262109	7	121,444,199	0.06	1.00	—	—
rs664893	19	39,757,572	0.12	1.00	—	—
rs6679677	1	114,303,808	0.14	1.00	—	—
rs11983481	7	69,973,572	0.07	0.04	AUTS2	26.6
rs17100164	14	33,589,065	0.08	0.02	NPAS3	465.9
rs17671833	16	7,427,842	0.09	0.01	RBFOX1	380.6
rs3773050	3	29,554,121	0.10	0.01	RBMS3	729.1
rs12637323	3	61,868,242	0.11	0.01	PTPRG	733.3
rs7752758	6	88,866,376	0.12	0.01	CNR1	1.4
rs10952495	7	154,261,961	0.11	0.01	DPP6	1,101.6
rs7511741	1	7,145,417	0.14	0.01	CAMTA1	984.3
rs3807218	7	154,461,112	0.10	0.01	DPP6	1,101.6
rs10765177	10	129,682,249	0.22	0.01	CLRN3	15.1
rs6480991	10	54,834,396	0.21	0.01	—	—

Table 6.6: Top 15 SNPs at optimal EM filtering step using $\mathbf{r} = \mathbf{1}$, $\xi_0 = -8$, and $\xi_1 = 4$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\mathbb{E}(\theta_j \mid \cdot)$	Gene	Length (kpb)
rs1982126	7	157.74	0.33	0.08	PTPRN2	1,002.7
rs3773050	3	29.55	0.10	0.06	RBMS3	729.1
rs1279214	14	33.45	0.19	0.06	NPAS3	465.9
rs4971264	1	216.29	0.33	0.04	USH2A	249.4
rs7752758	6	88.87	0.12	0.04	CNR1	1.4
rs6969220	7	157.74	0.44	0.04	PTPRN2	1,002.7
rs4462116	1	215.96	0.24	0.04	USH2A	249.4
rs17326887	8	3.59	0.09	0.04	CSMD1	2,059.5
rs11983481	7	69.97	0.07	0.04	AUTS2	26.6
rs9644354	8	3.58	0.13	0.04	CSMD1	2,059.5
rs17100164	14	33.59	0.08	0.04	NPAS3	465.9
rs8031347	15	33.59	0.25	0.04	—	—
rs7517281	1	3.22	0.22	0.04	PRDM16	369.4
rs2343466	2	45.51	0.23	0.03	—	—
rs4545164	9	77.85	0.40	0.03	—	—

Table 6.7: Top 15 SNPs based on posterior samples using $\mathbf{r} = \mathbf{1}$, $\xi_0 = -8$, and $\xi_1 = 4$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\hat{\mathbb{P}}(\theta_j = 1 \mid y)$	Gene	Length (kpb)
rs11983481	7	69.97	0.07	0.03	AUTS2	26.6
rs1982126	7	157.74	0.33	0.02	PTPRN2	1,002.7
rs1279214	14	33.45	0.19	0.02	NPAS3	465.9
rs3773050	3	29.55	0.10	0.01	RBMS3	729.1
rs17100164	14	33.59	0.08	0.01	NPAS3	465.9
rs16958917	16	82.98	0.06	0.01	CDH13	554.2
rs1403592	8	3.86	0.08	0.01	CSMD1	2,059.5
rs9644354	8	3.58	0.13	0.01	CSMD1	2,059.5
rs6969220	7	157.74	0.44	0.01	PTPRN2	1,002.7
rs17326887	8	3.59	0.09	0.01	CSMD1	2,059.5
rs7752758	6	88.87	0.12	0.01	CNR1	1.4
rs3807218	7	154.46	0.10	0.01	DPP6	1,101.6
rs17185050	14	68.05	0.10	0.01	PLEKHH1	11.0
rs10503246	8	4.13	0.29	0.01	CSMD1	2,059.5
rs2343466	2	45.51	0.23	0.01	—	—

Table 6.8: Top 15 SNPs at optimal EM filtering step using $\mathbf{r} = \mathbf{1}$, $\xi_0 = -8$, and $\xi_1 = 8$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\mathbb{E}(\theta_j \vert \cdot)$	Gene	Length (kpb)
rs1982126	7	157.74	0.33	0.11	PTPRN2	1,002.7
rs1279214	14	33.45	0.19	0.07	NPAS3	465.9
rs3773050	3	29.55	0.10	0.06	RBMS3	729.1
rs4971264	1	216.29	0.33	0.05	USH2A	249.4
rs7752758	6	88.87	0.12	0.05	CNR1	1.4
rs6969220	7	157.74	0.44	0.05	PTPRN2	1,002.7
rs4462116	1	215.96	0.24	0.05	USH2A	249.4
rs17100164	14	33.59	0.08	0.05	NPAS3	465.9
rs11983481	7	69.97	0.07	0.05	AUTS2	26.6
rs17326887	8	3.59	0.09	0.05	CSMD1	2,059.5
rs7517281	1	3.22	0.22	0.04	PRDM16	369.4
rs4545164	9	77.85	0.40	0.04	—	—
rs8031347	15	33.59	0.25	0.04	—	—
rs9644354	8	3.58	0.13	0.04	CSMD1	2,059.5
rs2343466	2	45.51	0.23	0.04	—	—

Table 6.9: Top 15 SNPs based on posterior samples using $\mathbf{r} = \mathbf{1}$, $\xi_0 = -8$, and $\xi_1 = 8$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\hat{\mathbb{P}}(\theta_j = 1 \vert y)$	Gene	Length (kpb)
rs11983481	7	69.97	0.07	0.03	AUTS2	26.6
rs1982126	7	157.74	0.33	0.02	PTPRN2	1,002.7
rs1279214	14	33.45	0.19	0.02	NPAS3	465.9
rs17100164	14	33.59	0.08	0.01	NPAS3	465.9
rs3773050	3	29.55	0.10	0.01	RBMS3	729.1
rs16958917	16	82.98	0.06	0.01	CDH13	554.2
rs6969220	7	157.74	0.44	0.01	PTPRN2	1,002.7
rs1403592	8	3.86	0.08	0.01	CSMD1	2,059.5
rs17326887	8	3.59	0.09	0.01	CSMD1	2,059.5
rs1195693	1	81.58	0.16	0.01	—	—
rs9644354	8	3.58	0.13	0.01	CSMD1	2,059.5
rs2498587	6	118.03	0.15	0.01	NUS1	35.3
rs10503246	8	4.13	0.29	0.01	CSMD1	2,059.5
rs7517281	1	3.22	0.22	0.01	PRDM16	369.4
rs8031347	15	33.59	0.25	0.01	—	—

Table 6.10: Top 15 SNPs at optimal EM filtering step using MalaCards, $\xi_0 = -8$, and $\xi_1 = 1$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\mathbb{E}(\theta_j \vert \cdot)$	Gene	Length (kpb)
rs4718582	7	66.95	0.08	1.00	—	—
rs10262109	7	121.44	0.06	1.00	—	—
rs1028850	13	40.94	0.47	0.25	LINC00598	133.9
rs664893	19	39.76	0.12	0.17	—	—
rs6679677	1	114.30	0.14	0.13	—	—
rs4133002	8	72.72	0.12	0.12	—	—
rs903228	2	53.69	0.06	0.11	—	—
rs1169722	12	121.64	0.17	0.06	—	—
rs11218078	11	120.82	0.18	0.05	GRIK4	326.0
rs10893006	11	123.18	0.36	0.05	—	—
rs947474	10	6.39	0.18	0.05	—	—
rs16881910	8	34.13	0.14	0.04	—	—
rs977375	2	56.98	0.45	0.04	—	—
rs2137862	20	58.01	0.16	0.03	—	—
rs7826601	8	26.41	0.29	0.03	DPYSL2	144.0

Table 6.11: Top 15 SNPs based on posterior samples using MalaCards, $\xi_0 = -8$, and $\xi_1 = 1$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\hat{\mathbb{P}}(\theta_j = 1 \vert y)$	Gene	Length (kpb)
rs4718582	7	66.95	0.08	1.00	—	—
rs10262109	7	121.44	0.06	1.00	—	—
rs903228	2	53.69	0.06	0.03	—	—
rs664893	19	39.76	0.12	0.03	—	—
rs6679677	1	114.30	0.14	0.03	—	—
rs12670243	7	82.97	0.06	0.02	—	—
rs1028850	13	40.94	0.47	0.02	LINC00598	133.9
rs4133002	8	72.72	0.12	0.01	—	—
rs947474	10	6.39	0.18	0.01	—	—
rs10893006	11	123.18	0.36	0.01	—	—
rs11218078	11	120.82	0.18	0.01	GRIK4	326.0
rs2137862	20	58.01	0.16	0.01	—	—
rs1169722	12	121.64	0.17	0.00	—	—
rs7601303	2	40.12	0.12	0.00	—	—
rs6843448	4	129.47	0.31	0.00	—	—

Table 6.12: Top 15 SNPs at optimal EM filtering step using MalaCards, $\xi_0 = -8$, and $\xi_1 = 4$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\mathbb{E}(\theta_j \mid \cdot)$	Gene	Length (kpb)
rs4718582	7	66,954,061	0.08	1.00	—	—
rs10262109	7	121,444,199	0.06	1.00	—	—
rs903228	2	53,692,049	0.06	1.00	—	—
rs664893	19	39,757,572	0.12	0.99	—	—
rs1028850	13	40,941,480	0.47	0.78	LINC00598	133.9
rs1169722	12	121,641,625	0.17	0.26	—	—
rs6679677	1	114,303,808	0.14	0.11	—	—
rs12670243	7	82,969,350	0.06	0.10	—	—
rs11629054	14	70,206,417	0.29	0.07	—	—
rs4133002	8	72,718,581	0.12	0.07	—	—
rs11218078	11	120,824,692	0.18	0.06	GRIK4	326.0
rs9315704	13	40,140,215	0.17	0.05	—	—
rs950776	15	78,926,018	0.35	0.04	CHRNB4	17.0
rs17381815	13	109,015,760	0.19	0.04	—	—
rs2356895	14	51,828,412	0.30	0.04	LINC00640	32.2

Table 6.13: Top 15 SNPs based on posterior samples using MalaCards, $\xi_0 = -8$, and $\xi_1 = 4$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\hat{\mathbb{P}}(\theta_j = 1 \mid y)$	Gene	Length (kpb)
rs4718582	7	66.95	0.08	1.00	—	—
rs10262109	7	121.44	0.06	1.00	—	—
rs903228	2	53.69	0.06	0.29	—	—
rs664893	19	39.76	0.12	0.23	—	—
rs12670243	7	82.97	0.06	0.18	—	—
rs1028850	13	40.94	0.47	0.06	LINC00598	133.9
rs6679677	1	114.30	0.14	0.04	—	—
rs1169722	12	121.64	0.17	0.02	—	—
rs9315704	13	40.14	0.17	0.01	LHFP	260.3
rs3747113	22	24.72	0.27	0.01	SPECC1L	171.5
rs11218078	11	120.82	0.18	0.01	GRIK4	326.0
rs4133002	8	72.72	0.12	0.01	—	—
rs6945822	7	130.36	0.08	0.01	TSGA13	18.8
rs11629054	14	70.21	0.29	0.01	—	—
rs7601303	2	40.12	0.12	0.01	—	—

Table 6.14: Top 15 SNPs at optimal EM filtering step using MalaCards, $\xi_0 = -8$, and $\xi_1 = 8$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\mathbb{E}(\theta_j \mid \cdot)$	Gene	Length (kpb)
rs4718582	7	66.95	0.08	1.00	—	—
rs10262109	7	121.44	0.06	1.00	—	—
rs12670243	7	82.97	0.06	1.00	—	—
rs6679677	1	114.30	0.14	1.00	—	—
rs664893	19	39.76	0.12	1.00	—	—
rs1169722	12	121.64	0.17	0.98	—	—
rs1028850	13	40.94	0.47	0.98	LINC00598	133.9
rs11218078	11	120.82	0.18	0.13	GRIK4	326.0
rs220704	6	46.87	0.12	0.08	GPR116	69.5
rs4133002	8	72.72	0.12	0.08	—	—
rs10088000	8	3.53	0.41	0.08	CSMD1	2,059.5
rs17191596	15	61.04	0.12	0.05	RORA	741.0
rs11629054	14	70.21	0.29	0.05	—	—
rs10892997	11	123.16	0.43	0.05	—	—
rs556560	5	102.62	0.35	0.05	—	—

Table 6.15: Top 15 SNPs based on posterior samples using MalaCards, $\xi_0 = -8$, and $\xi_1 = 8$ when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	MAF	$\hat{\mathbb{P}}(\theta_j = 1 \mid y)$	Gene	Length (kpb)
rs4718582	7	66.95	0.08	1.00	—	—
rs10262109	7	121.44	0.06	1.00	—	—
rs6679677	1	114.30	0.14	0.70	—	—
rs664893	19	39.76	0.12	0.57	—	—
rs12670243	7	82.97	0.06	0.49	—	—
rs1028850	13	40.94	0.47	0.10	LINC00598	133.9
rs1169722	12	121.64	0.17	0.08	—	—
rs220704	6	46.87	0.12	0.01	—	—
rs4133002	8	72.72	0.12	0.01	—	—
rs11218078	11	120.82	0.18	0.01	GRIK4	326.0
rs6959847	7	11.26	0.12	0.01	TSGA13	18.8
rs10088000	8	3.53	0.41	0.01	CSMD1	2,059.5
rs17191596	15	61.04	0.12	0.01	RORA	741.0
rs17100164	14	33.59	0.08	0.01	NPAS3	465.9
rs556560	5	102.62	0.35	0.01	—	—

6.2 Tables of Results in Chapter 3

Table 6.16: I give the mean running times and corresponding standard deviations (in parentheses) in minutes for the SB model and the single SNP tests in R using 10 replicates.

Task (n, p) :	$(10^2, 10^3)$	$(10^2, 10^4)$	$(10^3, 10^3)$	$(10^3, 10^4)$
Compute SVD with irlba [57]	0.35 (0.00)	3.43 (0.08)	1.16 (0.00)	124.90 (4.51)
EM filter on X				
after running the first pass	0.02 (0.00)	10.36 (0.03)	0.12 (0.00)	31.23 (0.25)
after retaining 25% of p	0.04 (0.00)	17.81 (0.03)	0.39 (0.01)	91.26 (0.49)
EM filter on SVD (1% MSE)				
after running the first pass	0.03 (0.00)	1.99 (0.01)	0.13 (0.00)	33.88 (0.12)
after retaining 25% of p	0.15 (0.00)	3.64 (0.04)	1.27 (0.01)	60.95 (1.28)
EM filter on SVD (10% MSE)				
after running the first pass	0.01 (0.00)	0.77 (0.00)	0.01 (0.00)	7.66 (0.00)
after retaining 25% of p	0.02 (0.00)	1.64 (0.01)	0.04 (0.00)	17.57 (0.33)
EM filter on SVD (25% MSE)				
after running the first pass	0.00 (0.00)	0.28 (0.01)	0.00 (0.00)	1.47 (0.01)
after retaining 25% of p	0.01 (0.00)	0.83 (0.02)	0.01 (0.00)	2.48 (0.01)
Gibbs sampler on X with N = 1,500	9.00 (0.03)	6626.99 (33.98)	9.32 (0.04)	7612.43 (94.71)
Single SNP tests	0.05 (0.00)	0.53 (0.01)	0.07 (0.00)	0.73 (0.02)

6.3 Tables of Results in Chapter 4

Table 6.17: Top 10 SNPs after running single SNP tests when analyzing the hypertension dataset provided by the Genetic Analysis Workshop 18.

SNP	CHR	Position (Mbp)	$-\log_{10}(\text{p-value}_j)$	Gene
rs2045732	9	100.19	4.67	TDRD7
rs4557815	9	100.21	4.67	TDRD7
rs11916152	3	127.45	4.31	MGLL
rs9829311	3	81.36	4.29	—
rs2827641	21	24.00	4.20	—
rs3013107	1	13.80	4.18	LRRC38
rs10982745	9	100.32	4.17	TMOD1
rs4743112	9	100.33	4.17	TMOD1
rs360490	1	33.22	4.09	KIAA1522
rs7621379	3	127.46	4.00	MGLL

Table 6.18: Top 10 SNPs after running latent genotype model with $\tau^2 = 0.5$ and $\zeta = 0$ when analyzing the hypertension dataset provided by the Genetic Analysis Workshop 18.

SNP	CHR	Position (Mbp)	$\mathbb{E}(\theta_j\|\cdot)$	Gene
rs1143700	19	5.21	0.19	PTPRS
rs8081951	17	0.84	0.18	NXN
rs6606865	15	27.28	0.17	GABRG3
rs7501812	17	17.75	0.15	TOM1L2
rs1370722	1	80.34	0.15	—
rs10127541	1	10.17	0.14	UBE4B
rs16889068	5	21.21	0.13	—
rs945742	1	146.79	0.13	—
rs1378942	15	75.08	0.13	CSK
rs2826363	21	21.91	0.13	—

Table 6.19: Top 10 SNPs after running latent genotype model with $\tau^2 = 0.5$ and $\zeta = 5{,}000$ when analyzing the hypertension dataset provided by the Genetic Analysis Workshop 18.

SNP	CHR	Position (Mbp)	$\mathbb{E}(\theta_j\|\cdot)$	Gene
rs8112338	19	31.61	1.00	—
rs1320301	19	49.94	1.00	SLC17A7
rs4801783	19	49.43	1.00	NUCB1, DHDH
rs329548	7	35.11	0.17	—
rs9582005	13	28.73	0.15	PAN3
rs11916152	3	127.45	0.14	MGLL
rs11632150	15	46.05	0.13	—
rs16876243	5	5.77	0.12	—
rs7498047	15	92.08	0.11	—
rs3176639	9	100.46	0.11	XPA

6.4 Tables of Results in Chapter 5

Table 6.20: Top 10 SNPs after running the spatial boost model on latent genotypes when analyzing the rheumatoid arthritis dataset provided by the Wellcome Trust Case Control Consortium.

SNP	CHR	Position (Mbp)	$\mathbb{E}(\theta_j \mid \cdot)$	Gene
rs10262109	7	121.44	1.00	—
rs1733717	10	54.29	1.00	—
rs1169722	12	121.64	0.08	—
rs6679677	1	114.30	0.01	—
rs1230666	1	114.17	0.01	MAGI3
rs962087	5	24.87	0.01	—
rs4867173	5	29.34	0.01	—
rs6945822	7	130.36	0.01	TSGA13
rs2011703	20	54.56	0.01	—
rs11058660	12	126.94	0.01	LOC100128554

Table 6.21: Top 10 SNPs after running the spatial boost model on latent genotypes when analyzing the hypertension dataset provided by the Genetic Analysis Workshop 18.

SNP	CHR	Position (Mbp)	$\mathbb{E}(\theta_j \mid \cdot)$	Gene
rs7498047	15	92.08	0.19	—
rs9582005	13	28.73	0.16	PAN3
rs11916152	3	127.45	0.15	MGLL
rs329548	7	35.11	0.14	—
rs2607221	19	28.70	0.13	—
rs17725246	7	44.58	0.13	—
rs6690382	1	39.10	0.12	—
rs12047550	1	33.70	0.12	—
rs6576443	15	25.90	0.12	—
rs1417272	9	85.48	0.11	—

Bibliography

[1] Hisham Al-Mubaid and Rajit K Singh. A text-mining technique for extracting gene-disease associations from the biomedical literature. *International journal of bioinformatics research and applications*, 6(3):270–286, 2010.

[2] Bruce Alberts, Dennis Bray, Julian Lewis, Martin Raff, Keith Roberts, and James D Watson. Molecular biology of the cell, 1994. *Garland, New York*, pages 139–194, 1994.

[3] Carl A Anderson, Fredrik H Pettersson, Geraldine M Clarke, Lon R Cardon, Andrew P Morris, and Krina T Zondervan. Data quality control in genetic case-control association studies. *Nature protocols*, 5(9):1564–1573, 2010.

[4] Sio Iong Ao, Kevin Yip, Michael Ng, David Cheung, Pui-Yee Fong, Ian Melhado, and Pak C Sham. CLUSTAG: hierarchical clustering and graph methods for selecting tag SNPs. *Bioinformatics*, 21(8):1735–1736, 2005.

[5] Kristin G Ardlie, Leonid Kruglyak, and Mark Seielstad. Patterns of linkage disequilibrium in the human genome. *Nature Reviews Genetics*, 3(4):299–309, 2002.

[6] Kristin L Ayers and Heather J Cordell. SNP Selection in genome-wide and candidate gene studies via penalized logistic regression. *Genetic epidemiology*, 34(8):879–891, 2010.

[7] David J Balding. A tutorial on statistical methods for population association studies. *Nature Reviews Genetics*, 7(10):781–791, 2006.

[8] M.M. Barbieri and J.O. Berger. Optimal predictive model selection. *The Annals of Statistics*, 32(3):870–897, 2004.

[9] J.O. Berger. *Statistical decision theory and Bayesian analysis.* Springer, 1985.

[10] Andrew P Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

[11] Broad Institute. SNP, 2015.

[12] Stephen P Brooks and Andrew Gelman. General methods for monitoring convergence of iterative simulations. *Journal of computational and graphical statistics*, 7(4):434–455, 1998.

[13] Paul R Burton, David G Clayton, Lon R Cardon, Nick Craddock, Panos Deloukas, Audrey Duncanson, Dominic P Kwiatkowski, Mark I McCarthy, Willem H Ouwehand, Nilesh J Samani, et al. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*, 447(7145):661–678, 2007.

[14] Christopher S Carlson, Michael A Eberle, Mark J Rieder, Qian Yi, Leonid Kruglyak, and Deborah A Nickerson. Selecting a maximally informative set of single-nucleotide polymorphisms for association analyses using linkage disequilibrium. *The American Journal of Human Genetics*, 74(1):106–120, 2004.

[15] L.E. Carvalho and C.E. Lawrence. Centroid Estimation in Discrete High-Dimensional Spaces with Applications in Biology. *Proceedings of the National Academy of Sciences*, 105(9):3209–3214, 2008.

[16] James M Cheverud. A simple correction for multiple comparisons in interval mapping genome scans. *Heredity*, 87(1):52–58, 2001.

[17] Seoae Cho, Haseong Kim, Sohee Oh, Kyunga Kim, and Taesung Park. Elastic-net regularization approaches for genome-wide association studies of rheumatoid arthritis. In *BMC proceedings*, volume 3, page S25. BioMed Central Ltd, 2009.

[18] 1000 Genomes Project Consortium et al. An integrated map of genetic variation from 1,092 human genomes. *Nature*, 491(7422):56–65, 2012.

[19] A Corvin, N Craddock, and PF Sullivan. Genome-wide association studies: a primer. *Psychological medicine*, 40(07):1063–1077, 2010.

[20] Mary Kathryn Cowles and Bradley P Carlin. Markov chain Monte Carlo convergence diagnostics: a comparative review. *Journal of the American Statistical Association*, 91(434):883–904, 1996.

[21] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[22] Frank Dudbridge and Arief Gusnanto. Estimation of significance thresholds for genomewide association scans. *Genetic epidemiology*, 32(3):227–234, 2008.

[23] Olive Jean Dunn. Multiple comparisons among means. *Journal of the American Statistical Association*, 56(293):52–64, 1961.

[24] Evangelos Evangelou and John PA Ioannidis. Meta-analysis methods for genome-wide association studies and beyond. *Nature Reviews Genetics*, 14(6):379–389, 2013.

[25] Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

[26] J. Friedman, T. Hastie, and R. Tibshirani. A note on the group lasso and a sparse group lasso. Technical Report arXiv:1001.0736, Jan 2010.

[27] Xiaoyi Gao, Lewis C Becker, Diane M Becker, Joshua D Starmer, and Michael A Province. Avoiding the high Bonferroni penalty in genome-wide association studies. *Genetic epidemiology*, 34(1):100–105, 2010.

[28] Alan E Gelfand and Sujit K Ghosh. Model choice: A minimum posterior predictive loss approach. *Biometrika*, 85(1):1–11, 1998.

[29] Edward I George and Robert E McCulloch. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889, 1993.

[30] Richard A Gibbs, John W Belmont, Paul Hardenbol, Thomas D Willis, Fuli Yu, Huanming Yang, Lan-Yang Ch'ang, Wei Huang, Bin Liu, Yan Shen, et al. The international HapMap project. *Nature*, 426(6968):789–796, 2003.

[31] Gene H Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[32] Aleksander M Grabiec, Chiara Angiolilli, Linda M Hartkamp, Lisa GM van Baarsen, Paul P Tak, and Kris A Reedquist. JNK-dependent downregulation of FoxO1 is required to promote the survival of fibroblast-like synoviocytes in rheumatoid arthritis. *Annals of the rheumatic diseases*, pages annrheumdis–2013, 2014.

[33] Yongtao Guan, Matthew Stephens, et al. Bayesian variable selection regression for genome-wide association studies and other large-scale problems. *The Annals of Applied Statistics*, 5(3):1780–1815, 2011.

[34] D. Habier, R. Fernando, K. Kizilkaya, and D. Garric. Extension of the Bayesian alphabet for genomic selection. *BMC bioinformatics*, 12:186, 2011.

[35] Michiaki Hamada and Kiyoshi Asai. A classification of bioinformatics algorithms from the viewpoint of maximizing expected accuracy (MEA). *Journal of Computational Biology*, 19(5):532–549, 2012.

[36] Daniel L Hartl, Andrew G Clark, et al. *Principles of population genetics*, volume 116. Sinauer associates Sunderland, 1997.

[37] Jarvis Haupt, Rui M Castro, and Robert Nowak. Distilled sensing: Adaptive sampling for sparse detection and estimation. *Information Theory, IEEE Transactions on*, 57(9):6222–6235, 2011.

[38] Edith Heard, Sarah Tishkoff, John A Todd, Marc Vidal, Günter P Wagner, Jun Wang, Detlef Weigel, and Richard Young. Ten years of genetics and genomics: what have we achieved and where are we heading? *Nature Reviews Genetics*, 11(10):723–733, 2010.

[39] Joel N Hirschhorn and Mark J Daly. Genome-wide association studies for common diseases and complex traits. *Nature Reviews Genetics*, 6(2):95–108, 2005.

[40] A. Hoerl and R. Kennard. Ridge regression - applications to nonorthogonal problems. *Technometrics*, 12:69–82, 1970.

[41] Gabriel E Hoffman, Benjamin A Logsdon, and Jason G Mezey. PUMA: a unified framework for penalized multiple regression analysis of GWAS data. *PLoS computational biology*, 9(6):e1003101, 2013.

[42] John PA Ioannidis, Gilles Thomas, and Mark J Daly. Validating, augmenting and refining genome-wide association signals. *Nature Reviews Genetics*, 10(5):318–329, 2009.

[43] Hemant Ishwaran and J Sunil Rao. Spike and slab variable selection: frequentist and Bayesian strategies. *Annals of Statistics*, pages 730–773, 2005.

[44] Andrew D Johnson and Christopher J O'Donnell. An open access database of genome-wide association results. *BMC medical genetics*, 10(1):6, 2009.

[45] Gillian CL Johnson, Laura Esposito, Bryan J Barratt, Annabel N Smith, Joanne Heward, Gianfranco Di Genova, Hironori Ueda, Heather J Cordell, Iain A Eaves, Frank Dudbridge, et al. Haplotype tagging for the identification of common disease genes. *Nature genetics*, 29(2):233–237, 2001.

[46] Ian Johnston and Luis E Carvalho. A Bayesian hierarchical gene model on latent genotypes for genome-wide association studies. In *BMC proceedings*, volume 8, page S45. BioMed Central Ltd, 2014.

[47] Ian Jolliffe. *Principal component analysis*. Wiley Online Library, 2005.

[48] LB Jorde. Linkage disequilibrium and the search for complex disease genes. *Genome research*, 10(10):1435–1444, 2000.

[49] Eric Jorgenson and John S Witte. A gene-centric approach to genome-wide association studies. *Nature Reviews Genetics*, 7(11):885–891, 2006.

[50] Omid Kohannim, Derrek P Hibar, Jason L Stein, Neda Jahanshad, Clifford R Jack, Michael W Weiner, Arthur W Toga, and Paul M Thompson. Boosting power to detect genetic associations in imaging using multi-locus, genome-wide scans and ridge regression. In *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*, pages 1855–1859. IEEE, 2011.

[51] Charles Kooperberg, Michael LeBlanc, and Valerie Obenchain. Risk prediction using genome-wide association studies. *Genetic epidemiology*, 34(7):643–652, 2010.

[52] Leonid Kruglyak. Prospects for whole-genome linkage disequilibrium mapping of common disease genes. *Nature genetics*, 22(2):139–144, 1999.

[53] Thomas LaFramboise. Single nucleotide polymorphism arrays: a decade of biological, computational and technological advances. *Nucleic acids research*, page gkp552, 2009.

[54] Eric S Lander, Lauren M Linton, Bruce Birren, Chad Nusbaum, Michael C Zody, Jennifer Baldwin, Keri Devon, Ken Dewar, Michael Doyle, William FitzHugh, et al. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001.

[55] Robert Lawrence, Aaron G Day-Williams, Katherine S Elliott, Andrew P Morris, and Eleftheria Zeggini. CCRaVAT and QuTie-enabling analysis of rare variants in large-scale case control and quantitative trait association studies. *BMC bioinformatics*, 11(1):527, 2010.

[56] Juan Pablo Lewinger, David V Conti, James W Baurley, Timothy J Triche, and Duncan C Thomas. Hierarchical Bayes prioritization of marker associations from a genome-wide association scan for further investigation. *Genetic epidemiology*, 31(8):871–882, 2007.

[57] B Lewis. irlba: Fast Partial SVD by Implicitly-Restarted Lanczos Bidiagonalization. *R package version 0.1*, 1:1520, 2009.

[58] Chun Li and Mingyao Li. GWAsimulator: a rapid whole-genome simulation program. *Bioinformatics*, 24(1):140–142, 2008.

[59] Jin Liu, Kai Wang, Shuangge Ma, and Jian Huang. Regularized regression method for genome-wide association studies. In *BMC proceedings*, volume 5, page S67. BioMed Central Ltd, 2011.

[60] Peter MacCullagh and John Ashworth Nelder. *Generalized linear models*, volume 37. CRC press, 1989.

[61] MalaCards. MalaCards Scores, 2014.

[62] Peter McCullagh and John A Nelder. Generalized linear models. 1989.

[63] Xiao-Li Meng and Donald B Rubin. Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika*, 80(2):267–278, 1993.

[64] Laëtitia Michou, Sandra Lasbleiz, Anne-Christine Rat, Paola Migliorini, Alejandro Balsa, René Westhovens, Pilar Barrera, Helena Alves, Céline Pierlot, Elodie Glikmans, et al. Linkage proof for PTPN22, a rheumatoid arthritis susceptibility gene and a human autoimmunity gene. *Proceedings of the National Academy of Sciences*, 104(5):1649–1654, 2007.

[65] Andrew P Morris and Eleftheria Zeggini. An evaluation of statistical approaches to rare variant analysis in genetic association studies. *Genetic epidemiology*, 34(2):188–193, 2010.

[66] John Neter, William Wasserman, and Michael H Kutner. Applied linear regression models. 1989.

[67] Dale R Nyholt. A simple correction for multiple testing for single-nucleotide polymorphisms in linkage disequilibrium with each other. *The American Journal of Human Genetics*, 74(4):765–769, 2004.

[68] Roman Pahl and Helmut Schäfer. PERMORY: an LD-exploiting permutation test algorithm for powerful genome-wide association testing. *Bioinformatics*, 26(17):2093–2100, 2010.

[69] Bin Peng, Dianwen Zhu, Bradley P Ander, Xiaoshuai Zhang, Fuzhong Xue, Frank R Sharp, and Xiaowei Yang. An Integrative Framework for Bayesian variable selection with informative priors for identifying genes and pathways. *PloS one*, 8(7):e67672, 2013.

[70] Kaare Brandt Petersen and Michael Syskind Pedersen. The matrix cookbook, 2008.

[71] Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian Inference for Logistic Models Using Pólya–Gamma Latent Variables. *Journal of the American Statistical Association*, 108(504):1339–1349, 2013.

[72] Alkes L Price, Nick J Patterson, Robert M Plenge, Michael E Weinblatt, Nancy A Shadick, and David Reich. Principal components analysis corrects for stratification in genome-wide association studies. *Nature genetics*, 38(8):904–909, 2006.

[73] Alkes L Price, Noah A Zaitlen, David Reich, and Nick Patterson. New approaches to population stratification in genome-wide association studies. *Nature Reviews Genetics*, 11(7):459–463, 2010.

[74] Shaun Purcell, Benjamin Neale, Kathe Todd-Brown, Lori Thomas, Manuel AR Ferreira, David Bender, Julian Maller, Pamela Sklar, Paul IW De Bakker, Mark J Daly, et al. PLINK: a tool set for whole-genome association and population-based linkage analyses. *The American Journal of Human Genetics*, 81(3):559–575, 2007.

[75] Brian D Ripley. *Spatial statistics*, volume 575. John Wiley & Sons, 2005.

[76] Veronika Ročková and Edward I George. EMVS: The EM approach to Bayesian Variable Selection. *Journal of the American Statistical Association*, (just-accepted), 2013.

[77] Xia Shen, Moudud Alam, Freddy Fikse, and Lars Rönnegård. A novel generalized ridge regression method for quantitative genetics. *Genetics*, 193(4):1255–1268, 2013.

[78] Daniel O Stram, Christopher A Haiman, Joel N Hirschhorn, David Altshuler, Laurence N Kolonel, Brian E Henderson, and Malcolm C Pike. Choosing haplotype-tagging SNPS based on unphased genotype data using a preliminary sample of unrelated subjects with an example from the Multiethnic Cohort Study. *Human heredity*, 55(1):27–36, 2003.

[79] Silke Szymczak, Joanna M Biernacka, Heather J Cordell, Oscar González-Recio, Inke R König, Heping Zhang, and Yan V Sun. Machine learning in genome-wide association studies. *Genetic Epidemiology*, 33(S1):S51–S57, 2009.

[80] Technology Department Carnegie Library of Pittsburgh. *The Handy Science Answer Book*. Visible Ink Press, 2002.

[81] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 58:267–288, 1996.

[82] R. Tibshirani and M. Saunders. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society*, 67:91–108, 2005.

[83] J Craig Venter, Mark D Adams, Eugene W Myers, Peter W Li, Richard J Mural, Granger G Sutton, Hamilton O Smith, Mark Yandell, Cheryl A Evans, Robert A Holt, et al. The sequence of the human genome. *science*, 291(5507):1304–1351, 2001.

[84] Lily Wang, Peilin Jia, Russell D Wolfinger, Xi Chen, and Zhongming Zhao. Gene set analysis of genome-wide association studies: methodological issues and perspectives. *Genomics*, 98(1):1–8, 2011.

[85] William YS Wang, Bryan J Barratt, David G Clayton, and John A Todd. Genome-wide association studies: theoretical and practical concerns. *Nature Reviews Genetics*, 6(2):109–118, 2005.

[86] Mike E Weale, Chantal Depondt, Stuart J Macdonald, Alice Smith, Poh San Lai, Simon D Shorvon, Nicholas W Wood, and David B Goldstein. Selection and Evaluation of Tagging SNPs in the Neuronal-Sodium-Channel Gene SCN1A: Implications for Linkage-Disequilibrium Gene Mapping. *The American Journal of Human Genetics*, 73(3):551–565, 2003.

[87] Lingjie Weng, Fabio Macciardi, Aravind Subramanian, Guia Guffanti, Steven G Potkin, Zhaoxia Yu, and Xiaohui Xie. SNP-based pathway enrichment analysis for genome-wide association studies. *BMC bioinformatics*, 12(1):99, 2011.

[88] Alice S Whittemore. A Bayesian false discovery rate for multiple testing. *Journal of Applied Statistics*, 34(1):1–9, 2007.

[89] Janis E Wigginton, David J Cutler, and Gonçalo R Abecasis. A note on exact tests of Hardy-Weinberg equilibrium. *The American Journal of Human Genetics*, 76(5):887–893, 2005.

[90] Michael C Wu, Seunggeun Lee, Tianxi Cai, Yun Li, Michael Boehnke, and Xihong Lin. Rare-variant association testing for sequencing data with the sequence kernel association test. *The American Journal of Human Genetics*, 89(1):82–93, 2011.

[91] Tong Tong Wu, Yi Fang Chen, Trevor Hastie, Eric Sobel, and Kenneth Lange. Genome-wide association analysis by lasso penalized logistic regression. *Bioinformatics*, 25(6):714–721, 2009.

[92] Tong Tong Wu and Kenneth Lange. Coordinate descent algorithms for lasso penalized regression. *The Annals of Applied Statistics*, pages 224–244, 2008.

[93] Bo Xi, Yue Shen, Kathleen Heather Reilly, Xia Wang, and Jie Mi. Recapitulation of four hypertension susceptibility genes (CSK, CYP17A1, MTHFR, and FGF5) in East Asians. *Metabolism*, 62(2):196–203, 2013.

[94] Xiang Zhou, Peter Carbonetto, and Matthew Stephens. Polygenic modeling with Bayesian sparse linear mixed models. *PLoS genetics*, 9(2):e1003264, 2013.

[95] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society*, 67:301–320, 2005.

Curriculum Vitae

Contact Ian Johnston

Department of Mathematics and Statistics, Boston University, 111 Cummington Street, Boston, MA 02215, USA

Education **Drexel University**, B.Sc., Mathematics, 9/2005 – 9/2010.

Boston University, M.A., Mathematics, 9/2010 – 1/2013.

Boston University PhD candidate, Mathematics, 9/2010 – present. Thesis advisor: Luis Carvalho.

Publications

1. Ian Johnston and Luis Carvalho, *A Bayesian hierarchical gene model on latent genotypes for genome-wide association studies.* BMC proceedings. Vol. 8. No. Suppl 1. BioMed Central Ltd, 2014.
2. Ian Johnston, Yang Jin, and Luis Carvalho, *Assessing a Spatial Boost Model for Quantitative Trait GWAS.* In Interdisciplinary Bayesian Statistics, pp. 337-346. Springer International Publishing, 2015.
3. Ian Johnston, Timothy Hancock, Hiroshi Mamitsuka, and Luis Carvalho, *Hierarchical Gene-Proximity Models For Genome-Wide Association Studies.* arXiv preprint arXiv:1311.0431 (2013).