2021

Statistical Approaches for Functional Annotation Tree Guided Prioritization of Genome-wide Association Studies (GWAS) Results

Aastha Khatiwada
Medical University of South Carolina

Statistical Approaches for Functional Annotation Tree Guided Prioritization of Genome-wide Association Studies (GWAS) Results

Aastha Khatiwada

A dissertation submitted to the faculty of the Medical University of South Carolina in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the College of Graduate Studies.

Department of Public Health Sciences

July 2021

Approved by:

Dr. Bethany J. Wolf, Chair

Dr. Dongjun Chung, Co-Chair

Dr. Andrew Lawson

Dr. Paula S. Ramos

Dr. Kelly J. Hunt

Dr. Hang J. Kim

Dedication

To my family for their love and support,

To my mentors for their guidance and wisdom, and

To Sulabh for his encouragement and companionship.

Acknowledgments

I would like to extend my thanks and appreciation to those who have supported my dissertation research. This includes my dissertation committee members, MUSC Department of Public Health Sciences, the MUSC Core Center for Clinical Research (CCCR), collaborators I have worked with during my time at MUSC, and all my friends. I would also like to acknowledge my funding sources. This dissertation work was funded by NIH/NIGMS grant R01-GM122078, NIH/NCI grant R21-CA209848, NIH/NIDA grant U01-DA045300, and NIH/NIAMS grants P30-AR072582 and R01-AR071947.

Table of Contents

List of Figures

Abstract

Genome-wide association studies (GWAS) have successfully identified over two hundred thousand trait risk-associated genetic variants; however, several challenges remain. First, a complex trait is associated with many single nucleotide polymorphisms (SNPs), each with small or moderate effect sizes that are hard to detect with limited sample size due to a phenomenon called polygenicity. Additionally, currently available statistical methods are limited in explaining the functional mechanisms through which genetic variants are associated with complex traits.

In the first dissertation aim, we address these challenges by proposing a statistical approach called GPA-Tree. GPA-Tree integrates GWAS summary statistics and functional annotation information for a single trait within a unified framework. Specifically, by combining a decision tree algorithm with a hierarchical modeling framework, GPA-Tree simultaneously implements association mapping and identifies key combinations of functional annotations related to the trait risk-associated SNPs. We evaluate the proposed GPA-Tree approach using simulation studies and demonstrate that, in most scenarios, GPA-Tree shows greater area under the curve (AUC) and power relative to existing statistical approaches in detecting risk-associated SNPs and greater accuracy in identifying the true combinations of functional annotations. We applied GPA-Tree to a systemic lupus erythematosus (SLE) GWAS and functional annotation data including GenoSkyline and GenoSkylinePlus. The results from GPA-Tree highlight the dysregulation of blood immune cells, including but not limited to primary B, memory helper T, regulatory T, neutrophils and CD8+ memory T cells.

The second dissertation aim exploits the phenomenon called pleiotropy, shared genetic basis among multiple traits, to improve statistical power to detect SNPs associated with one or more traits. We extend GPA-Tree to develop Multi-GPA-Tree so that GWAS summary statistics for multiple traits and functional annotation information can be integrated within a unified framework. Specifically, by combining a multivariate decision tree algorithm with a hierarchical modeling framework, Multi-GPA-Tree simultaneously implements association mapping and identifies key combinations of functional annotations related to the SNPs associated with one or more traits. We evaluate the proposed Multi-GPA-Tree approach using simulation studies and demonstrate that, in most scenarios, Multi-GPA-Tree outperforms existing statistical approaches in detecting SNPs associated with one or more traits and identifying the true combinations of functional annotations with high accuracy. We utilize Multi-GPA-Tree to integrate GWAS from two rheumatic diseases, SLE and Rheumatoid Arthritis (RA), and GWAS from two inflammatory bowel diseases, Crohn's trait (CD) and ulcerative colitis (UC), with GenoSkyline and GenoSkylinePlus annotations. The results from Multi-GPA-Tree highlight the dysregulation of blood immune cells for both joint analysis, including dysregulation of primary B cells for SLE and RA, and dysregulation of primary T regulatory cells for UC and CD.

In the third dissertation aim, we develop the R package GPATree and the R Shiny app ShinyGPATree. The R package and Shiny app facilitate users' convenience and make the GPA-Tree and Multi-GPA-Tree approach easily accessible. The package includes an example data and a vignette to facilitate seamless step-by-step implementation of the proposed methods. In addition, the Shiny app allows interactive and dynamic investigation of association mapping results and functional annotation trees.

1. Introduction

1.1 Overview

Single nucleotide polymorphism or SNP is an alteration in a single nucleotide at a specific position in the genome. For example, if the adenine (A) base pair is commonly observed at a specific position in most genomes, the base pair A maybe altered and replaced by another base pair (e.g., guanine (G)) in some genome. This alteration in a single nucleotide indicates the presence of a SNP at that position. When polymorphisms as described here are associated with a trait, they are called trait risk-associated SNPs.

In the past decade, genome wide association studies (GWAS) have been implemented to identify over two hundred thousand trait risk-associated SNPs [1]. However, there are challenges associated with these findings. First, including all GWAS identified SNPs only explains a small proportion of the variation in the heritability of a complex trait [2] ('heritability' is defined as the proportion of variation in a trait that is attributable to genetic variation within a population). This phenomenon is called 'missing heritability'. Missing heritability can be demonstrated using human height as an example. Human height, a highly heritable trait, has an estimated heritability of approximately 80% [3]. Including genome-wide significant and validated SNPs explain about 10% of the variation in human height [4], while including all commonly genotyped SNPs explains about 45% of the variation in human height [5]. This shows that a large number of SNPs that can explain the variation in human height still remain unidentified. Second, a trait can be associated with multiple SNPs with small or moderate effect sizes that do not meet the genome-wide p-value cutoff of 5×10^{-8} through a phenomenon called 'polygenicity' [6]. As a result, many SNPs with small or moderate effect sizes remain unidentified. Increasing the sample size

in the GWAS can potentially improve statistical power to detect SNPs with small and moderate effect sizes. However, recruiting a larger sample size often requires more resources and may not always be feasible due to limited prevalence of trait in the population. Therefore, it is desirable to find alternate ways to increase statistical power to detect SNPs with small effect sizes. Third, SNPs identified by GWAS can be in the coding, non-coding and intergenic regions of the DNA. Although it is easier to understand the functional potential of SNPs in the coding regions, over 85% of the GWAS identified SNPs are located in the non-coding regions [7]. Therefore, their functional role in the trait etiology may not straightforward to understand.

Functional annotations can provide valuable information regarding the different mechanisms through which SNPs may be associated with traits by incorporating information related to tissue- and cell-type specific functions, transcription factor binding, histone modifications, enhancer activity through chromatin architecture, DNA methylation, alternative splicing, and more. Utilizing functional information related to SNPs in the form of functional annotations also evidently improves statistical power to detect SNPs with small or moderate effect sizes while simultaneously elucidating the mechanisms by which SNPs are associated with the traits [8–11]. The general hypothesis is that a set of functional roles that are observed for SNPs may influence the distribution of the GWAS association p-values for the SNPs that are associated with the trait. For example, in the case of auto-immune traits like systemic lupus erythematosus (SLE) and multiple sclerosis (MS), SNPs related to the immune-system might be more enriched while for psychiatric disorders like bipolar disorder (BPD) and schizophrenia (SCZ), SNPs related to the central nervous system might be more enriched.

Another advantage of GWAS has been in recognizing a phenomenon called 'pleiotropy' which shows that distinct traits can share a common genetic basis, i.e., multiple traits can be associated with the same set of SNPs [12]. For example, using graph-GPA (a graphi-

cal model for prioritizing GWAS results and investigating pleiotropic architecture), Chung and colleagues established a shared genetic architecture between several autoimmune and psychiatric disorders [13]. graph-GPA also showed that utilizing multiple GWAS simultaneously to leverages pleiotropy increases statistical power to detect trait risk-associated SNPs for one or more GWAS traits. In addition, there is evidence in literature that simultaneously integrating GWAS for multiple traits to leverage pleitropy alongside SNP related functional annotation data can considerably increase statistical power to detect all relevant SNPs that are shared between different traits as well as for individual traits [8–10, 14–17].

We note that although statistical methodologies to discover trait risk-associated SNPs can be based on individual-level genotype data, it is often extremely difficult to procure individual-level genotype data for a relatively large sample size. Therefore, in this dissertation work, we focus our methodologies on GWAS summary statistics or p-values.

1.2 Gaps in the Current Literature

Several statistical methodologies are available to integrate GWAS summary statistics and functional annotation data to prioritize SNPs and identify relevant functional annotations [8–11, 14–17]. Schork et al. [8] developed the stratified False Discovery Rate (sFDR) method to integrate linkage disequlibrium-weighted genic annotation information and GWAS summary statistic for each SNP, and showed improvement in true discovery rates for SNPs by stratifying them based on their genic position while also uncovering patterns of polygenic effects in specific annotation categories across multiple traits. In the covariate modulated False Discovery Date (cm-FDR) method, Zablocki et al. [14] utilized genic functional annotations as covariates in a two-group mixture model to prioritize trait risk-associated SNPs and to discover enriched categories of functional annotations. Similarly, Ming et al. [9] developed a Latent Sparse Mixed Model (LSMM) method using a genelized linear

mixed model framework where genic- and cell-type specific annotations are assumed to have fixed and random effects, respectively. The GenoCanyon (GC) approach [15] utilizes an unsupervised machine learning algorithm to measure the overall functional potential for a SNP by assigning it a 'GC functional score' that is computed using 22 computational and experimental annotations from the ENCODE project [18]. GC functional scores were utilized in GenoWAP [16] to partition SNPs into functional and non-functional subgroups with the final goal to prioritize trait risk-associated SNPs by computing a posterior functional score for all SNPs using mixture model within an Expectation Maximization (EM) algorithm. Although these methods successfully integrate a single GWAS trait and functional annotation, they fail to leverage pleiotropic relationship between multiple GWAS traits.

On the other hand are methods that integrate GWAS association p-values for multiple traits by leveraging pleiotropy. For example, the pleiotropy-informed FDR method by Andreassen et al. [19] uses a model-free approach to prioritize SNPs that are associated with single as well as multiple GWAS traits. Likewise, graph-GPA by Chung et al. [13] is a useful tool to construct clusters of genetically correlated traits by integrating multiple GWAS. However, these methods do not integrate functional annotation information in their analysis.

Overcoming the shortcomings of the groups of methods that integrate functional annotations or leverage pleiotropy are methods that integrate functional annotations while also leveraging pleiotropy between multiple traits. For instance, Liu et al. [17] combined SNP level information to obtain gene-level information and then used a two-group model within the empirical Bayes framework to integrate pleiotropy and tissue-specific functional annotation to prioritize risk-associated SNPs. Similarly, in the Latent Probit Model (LPM) method [11] and the Genetic Analysis incorporating Functional Annotation and leveraging Pleiotropy (GPA) method [10], pleiotropy was leveraged and functional annotation

information was integrated with GWAS association p-values to characterize the genetic architecture shared by complex traits and to identify enriched functional annotations.

Although the methods described above can efficiently integrate functional annotations, or integrate functional annotation while leveraging pleiotropy to prioritize trat risk-associated SNPs, and to identify individual functional annotations that are related to one or more traits, there is scarcity of statistical methodologies that identify the combinations of functional annotations that act in unison to influence traits. Complex traits are often caused by an amalgamation of functional mechanisms that can be described by multiple functional annotations rather than a single functional annotation. Therefore, identifying the combinations of functional annotations that are associated with the traits can provide valuable insight into trait etiology. Theoretically, some of the existing methods can be extended to include interaction terms. However, scientific knowledge is often lacking to know which interactions to include in the model, especially when large number of functional annotations are considered. While it is possible to include all possible interaction terms, this can quickly become computationally taxing with the existing methods, specially when large number of functional annotations are considered. Therefore, a method that can perform automatic interaction selection needs to be developed.

1.3 Overall Goal and Specific Aims

The goal of this dissertation is to develop statistical methodologies that prioritize trait-risk associated SNPs while identifying the combinations of functional annotations related to the mechanisms through which risk-associated SNPs influence complex traits. In addition, an R package and an R Shiny App will be developed to implement the statistical methodologies. Specifically, the objectives of this dissertation are to:

- Aim 1: Develop a method called 'GPA-Tree' that utilizes a hierarchical architecture

to integrate GWAS summary statistics and functional annotation information within a unified framework for a complex trait by combining an iterative procedure (EM algorithm) and a decision tree algorithm (CART). GPA-Tree will simultaneously prioritize trait risk-associated SNPs and identify combinations of functional annotations that can potentially explain the mechanisms through which risk-associated SNPs are associated with the trait. The application of GPA-Tree will be shown using a SLE GWAS, and GenoSkyline and GenoSkylinePlus annotations.

- Aim 2: Extend Aim 1 to develop a method called 'Multi-GPA-Tree' that utilizes a hierarchical architecture to integrate GWAS summary statistics for multiple complex traits to leverage pleiotropy, and to integrate functional annotation information within a unified framework. Multi-GPA-Tree will combine an iterative procedure (EM algorithm) and a multivariate decision tree algorithm to simultaneously prioritize one or more trait risk-associated SNPs and identify the combinations of functional annotations that can potentially explain the mechanisms through which risk-associated SNPs are associated with one or more traits. The application of Multi-GPA-Tree will be shown using a SLE and rheumatoid arthritis (RA) GWAS, and ulcerative colitis (CD) and Crohn's disease (CD) GWAS, and GenoSkyline and GenoSkylinePlus annotations.

- Aim 3: Develop an R package and R Shiny App to implement the statistical methodologies developed in Aims 1 and 2.

2. Statistical Background

In this section, we will review statistical methodologies currently available for post-GWAS analysis. Noting differences in methodologies in regard to the type of data that can be integrated with each method, this section is divided into three subsections. In the first subsection, we will focus on statistical methodologies that integrate GWAS summary statistics and functional annotation information for single complex traits (no pleiotropy). In the second subsection, we will focus on statistical methodologies that integrate multiple GWAS traits by leveraging pleiotropy, but not including functional annotations. In the final subsection, we will focus on statistical methodologies that integrate GWAS summary statistics for multiple traits with functional annotation information (leveraging pleiotropy).

2.1 Statistical Methods Integrating GWAS Summary Statistics and Functional Annotations for a Single Trait

Several statistical methodologies are available to integrate GWAS association p-values and functional annotation data. LSMM [9] is a recently developed statistical methodology that integrates functional annotation data and GWAS association p-values by assuming a two-group model where SNPs either belong to a non-null (trait-associated) or null (not associated with the trait) group, and the GWAS association p-values for SNPs in the non-null and null groups come from a Beta-Uniform mixture. In LSMM, functional annotations are integrated using a logistic mixed effects model where genic- and cell-type specific functional annotations are assumed to have fixed and random effects, respectively. A sparse structure is imposed on the random effects to adaptively select relevant cell-type specific functional annotations. LSMM progresses in four stages. In the first stage, a two-group

model that sets coefficients to 0 for both tissue-specific fixed effects and cell-type specific random effects is employed within an EM framework to obtain initial parameter estimates for the proportion of non-null SNPs (π) and GWAS signal parameter (α) along with the posterior probabilities of association for each SNP. The estimates from the first stage are used to initialize the second stage in which tissue-specific functional annotations are incorporated to obtain the fixed effects estimates and to further update the parameters and the posterior probabilities of association for each SNP. In the third stage, a logistic sparse mixed model within a variational EM framework is fitted where the posterior probabilities of association from the second stage, and all tissue and cell-type specific functional annotations are used as response and predictor variables, respectively. In the fourth and final stage, the estimates from the third stage are utilized as initial values for all parameter estimates to fit the final variational EM algorithm. Through application of LSMM, Ming et al. discovered substantial enrichment of blood-related cell-type specific annotations for SLE, RA, UC and CD, among others [9]. LSMM also identified several new schizophrenia (SCZ) associated SNPs that were unidentified prior to utilizing functional annotation data, indicating increased statistical power in prioritizing risk-associated SNPs with integration of functional annotation information with GWAS summary statistics data.

Similar to LSMM, the covariate modulated false discovery rate (cmfdr) method by Zablocki et al. is a parametric method that integrates GWAS summary statistics and functional annotation information where functional annotation information provide 'prior information' in a parametric two-group mixture model [14]. Also, the GWAS association test statistic (z) for the null group are assumed to have a folded normal distribution with probability mass to the right of $z = 0$ such that $f_0(z) = 2\phi_{\sigma_0}(z)I_{z\geq 0}$ where $\phi(z)$ is $N(0, \sigma_0)$ and $I_{z\geq 0}$ is an indicator that $z \geq 0$. The GWAS association p-values for the non-null group are assumed to have a gamma distribution with a shape parameter (a), modulated by the covariates/functional annotations (x) as $a(x) = exp(x^T\alpha)$ and a scalar rate parameter (β)

not dependent on functional annotations. The prior probability for the latent indicator denoting whether or not a SNP is non-null ($\delta_i = 1$) is modeled using a logistic regression framework, i.e., $\pi_1(x_i) = Pr(\delta_i = 1|x_i) = \frac{exp(x_i^T\gamma)}{1+exp(x_i^T\gamma)}$. Following this, cmfdr is defined as the posterior probability that a SNP is null given its GWAS association p-value and functional annotation (covariate) information i.e., $cmfdr = \frac{\pi_0(x_i)f_0(z_i)}{\pi_0(x_i)f_0(z_i)+\pi_1(x_i)f_1(z_i|x_i)}$. The cmfdr method assumes that compared to SNPs that are not functionally relevant, SNPs that are functionally relevant have a lower false discovery rate, and are associated with the trait. The cmfdr method was similar at controlling false discovery proportion, but superior in terms of power compared to the local FDR method of Efron [20]. In its application to Crohn's trait, the number of significant loci increased by over five fold at the nominal fdr level of 0.05.

Theoretically, both LSMM and cmfdr can include interaction terms between different functional annotations. In addition, LSMM can perform enrichment analysis to test the importance of functional annotations or the interactions between the functional annotations. However, knowing which interactions to include in these methods is a challenge since there is limited guidance in the clinical literature regarding the constitution of interactions among functional annotations for complex traits. Considering all possible interactions between functional annotations to mitigate variable and interaction selection problem is also not feasible because of high computational burden, especially when large number of functional annotations are involved. Therefore, a more adaptable method that can integrate large number of functional annotations and make inferences related to interactions between those functional annotations will be beneficial.

The method of stratified False Discovery Rate (sFDR) by Sun et al. [21] can also be used to combine GWAS summary statistics and functional annotations. The overarching idea behind using sFDR is that we can calculate the false discovery rate (FDR) for each strata by dividing the GWAS association test statistics into multiple stratas with varying

probability that the null hypothesis is true. The estimates of FDR within each stratum can then be combined to explore changes in sensitivity and specificity. By using stratified analysis, sensitivity and specificity to detect true associations is expected to improve if there truly is variability in the proportion of null tests across strata. Schork et al. applied sFDR to integrate linkage disequilibrium (LD) weighted functional annotations for each SNP and the GWAS association p-values to estimate the true discovery rate (TDR = 1-FDR) for stratas composed of different genic categories [8]. To construct LD-weighted annotation categories that are not mutually exclusive, Schork et al. exploited the presence of naturally mutually exclusive stratification of GWAS SNPs based on its genomic position with respect to the first gene transcript listed in the UCSC known genes database. Eight genic positional categories were scored 0 or 1 based on a SNP's positional presence in: 1) 10, 000 to 1, 001 base pairs (bp) upstream, 2) 1,000 to 1 bp upstream, 3) 5' untranslated region (5'UTR), 4) exon, 5) intron, 6) 3' untranslated region (3'UTR), 7) 1 to 1,000 bp downstream, and 8) 1,001 to 10,000 bp downstream, all with reference to protein coding genes only. To obtain the LD-weighted annotation categories that are not mutually exclusive, for each tag SNP, a pairwise correlation coefficient approximation to LD (r^2) was calculated for all SNPs within one million base pairs (1 Mb) of the tag SNP. The sum of r^2 LD between the tag SNP and all SNPs positioned in a particular category was used as the LD-Weighted annotation scores. Tag SNPs were assigned to every LD-weighted annotation category for which its annotation score was greater than or equal to 1. The resulting LD-weighted annotation categories were not mutually exclusive since the same GWAS tag SNP could be annotated for multiple categories. Using sFDR, Schork et al. observed patterns of enrichment for SNPs in different genomic positions, increased true discovery rates for a given p-value and improved power to detect associations in complex traits. However, when multiple functional annotations are considered for post-GWAS analysis, we cannot easily create stratas, so this method may not be applicable for post-GWAS analysis including

large number of functional annotations.

Other avenues to explore the functional potential of a genomic position is to employ methods like GenoCanyon [15] and GenoWAP [16]. GenoCanyon is a statistical framework that predicts the functional role of each position in the human genome by integrating functional annotations using an unsupervised statistical learning procedure. This method is considered 'unsupervised' because GWAS association p-values are not integrated in its application. GenoCanyon assumes that the joint probability distribution of the functional annotations will vary based depending on the functional potential of a genomic position. This method calculates the functional measure score for a genomic position as the posterior probability that the genomic position is functional given its functional annotation information. GenoWAP integrates the functional measure scores from GenoCanyon and GWAS association p-values to prioritize trait risk-associated SNPs. In GenoWAP, the mean GenoCanyon functional score of the surrounding 10K base pairs is calculated for each SNP in a GWAS dataset such that SNPs with a calculated mean GenoCanyon score higher than 0.1 are assumed to be functional. SNPs are prioritized by GenoWAP by assigning each SNP a score that measures its importance. The importance score is calculated as the posterior probability that a SNP is trait-specific functional given its GWAS association p-value using an EM algorithm. By utilizing the GenoCanyon functional scores, GenoWAP demonstrated its effectiveness in its application to CD and SCZ by identifying new trait risk-associated loci. Although GenoCanyon is a convenient tool to obtain functional potential of genomic positions, its predictive ability is still limited by the annotations that are included in its computation and can be improved by a built-in variable selection procedure. Similarly, given that GenoWAP is a region based tool that can identify regions that are more likely to be functional within LD blocks, determining conclusive functionality for a SNP still requires allele-specific analysis. Also, its performance capability has not been investigated when cell- and tissue-specific epigenetic annotations are incorporated. Therefore, a method that

can integrate different types of functional annotation and adaptively select the combinations of functional annotations is needed.

2.2 Statistical Methods Integrating GWAS Summary Statistics for Multiple Traits by Leveraging Pleiotropy

This section is focused on statistical methodologies that integrate multiple GWAS traits by leveraging pleiotropy. The two methods that are discussed in detail are the pleiotropy-informed conditional FDR by Andreassen et al. [19] and graph-GPA by Chung et al. [13]. The unifying goal of these methods is to show improved statistical power to prioritize one or more trait risk-associated SNPs.

The pleiotropy-informed conditional FDR method was motivated by the sFDR method by Sun et al. [21] described in the previous section and the weighted FDR method by Roeder et al. [22]. The weighted FDR method adds weights to GWAS association p-values using linkage scores from genome-wide linkage studies. In contrast to the sFDR method that uses stratified empirical cdfs, the pleiotropy-informed FDR method uses empicial cdfs for the first trait conditional on the fact that the nominal p-values for the second trait is less than or equal to some predetermined threshold. Finally, pleiotropy-informed conditional FDR is defined as the posterior probability that a given SNP is null for the first trait given that the p-values for both traits are as small or smaller the observed p-values, i.e., $FDR(p_1|p_2) = \pi_0(p_2)p_1/F(p_1|p_2)$, where p_1 and p_2 are the p-values for the first and second traits, $F(p_1|p_2)$ is the conditional cdf of the p-values for the first trait given the p-values for the second trait, $\pi_0(p_2)$ is the conditional proportion of null SNPs for the first trait given that the p-values for the second traits are p_2 or smaller. For a conservative estimate of $FDR(p_1|p_2)$, $\pi_0(p_2)$ is set to 0 and the conditional FDR for trait 1 given trait 2 and vice versa is computed for all SNPs. SNPs with $FDR(p_1|p_2)$ < 0.05 are deemed associated

with the first trait given the second trait and SNPs with $FDR(p_2|p_1)$ < 0.05 are deemed associated with the second trait given the first trait. To identify SNPs associated with both traits, a conjunction FDR value is computed for each SNP. Conjunction FDR is defined as the posterior probability that a SNP is null for one or both traits when the p-values for both traits are as small or smaller than the observed p-values. A conservative, model-free estimate for conjunction FDR is computed as $F(p_1, p_2) = max\{FDR_{p_1|p_2},\ FDR_{p_2|p_1}\}$,where SNPs with conjunction FDR value < 0.05 are assumed to be associated with both traits. Utilizing this methodology, Andreassen et al. showed improved detection of risk-associated SNPs for two psychiatric disorders, SCZ and bipolar disorder (BPD). Despite easy implementation, the model-free approach used in this method imposes several limitations. The lack of a model-based approach in estimating conditional FDR compromises the power to detect non-null associations and also to infer the properties of the non-null distribution. Also, this integration method is limited to a small number (mostly a pair) of traits and cannot inform about functional relevance of risk-associated SNPs as functional annotations cannot be integrated using this method.

In contrast to the the pleiotropy-informed FDR method, graph-GPA can integrate large number of GWAS traits using a hidden Markov random field (MRF) approach. In graph-GPA, GWAS association p-values ($\boldsymbol{p}$) are initially transformed using a cumulative distribution function (CDF) of a standard normal distribution as $\boldsymbol{y} = \Phi^{-1}(1 - \boldsymbol{p})$ and a latent association indicator ($\boldsymbol{e}_i$) denoting the association of SNPs with the i^{th} trait is introduced. Also, the density of $\boldsymbol{y}_i$ given the latent association status $\boldsymbol{e}_i$ is assumed to come from normal mixtures as $p(\boldsymbol{y}_i|\boldsymbol{e}_i,\ \mu,\ \sigma^2) = \boldsymbol{e}_i\, LN(\boldsymbol{y}_i; \mu,\ \sigma^2) + (1 - \boldsymbol{e}_i)N(\boldsymbol{y}_i; 0, 1)$, where $LN(\boldsymbol{y};\ \mu,\ \sigma^2)$ is the log-normal density with mean $\boldsymbol{e}^{\mu+\sigma^2/2}$ and $N(\boldsymbol{y};\ 0, 1)$ is the standard normal distribution. To integrate multiple GWAS traits, a graphical model based on MRF and assuming an auto-logistic spatial scheme is used. Utilizing this scheme, the conditional distribution

of $\boldsymbol{e}_i$ is written as $p(\boldsymbol{e}_i|\boldsymbol{\alpha},\boldsymbol{\beta},G) = C(\boldsymbol{e}|\boldsymbol{\alpha},\boldsymbol{\beta},G)\exp(\sum_{i=1}^{n}\alpha_i\boldsymbol{e}_{it} + \sum_{i\sim j}\beta_{ij}\,\boldsymbol{e}_{it}\boldsymbol{e}_{jt})$, where t is the index for the t^{th} SNP, $C(\boldsymbol{\alpha},\boldsymbol{\beta},G)^{-1} = \sum_{e^*\in\mathcal{E}}\exp(\sum_{i=1}^{n}\alpha_i\boldsymbol{e^*}_{it} + \sum_{i\sim j}\beta_{ij}\,\boldsymbol{e^*}_{it}\boldsymbol{e^*}_{jt})$, $\mathcal{E}$ is the set of all possible values of $\boldsymbol{e}^* = (e_1^*,...,e_n^*)$, β_{ij} is the MRF coefficient for the pair of traits i where j, $i \sim j$ denotes two traits that are adjacent to one another in the graphical representation. Conjugate prior distributions are used for $\mu_i \sim N(\theta_\mu,\tau_\mu^2)$ and $\sigma_i^2 \sim IG(a_\sigma,b_\sigma)$, where IG denotes a inverse-gamma distribution. Prior distributions are also assumed for $\alpha_i \sim N(\theta_\alpha,\tau_\alpha^2)$ and $\beta_{ij} \sim E(i,j)\Gamma(\beta_{ij};a_\beta,b_\beta) + \{1-E(i,j)\}\delta_0(\beta_{ij})$, where $\Gamma(a,b)$ denotes a gamma distribution with mean a/b, δ_0 denotes the Direc delta function at 0, and $E(i,j)$ represents if there exists an edge between traits i and j. In the prior for $\beta_{ij}, \beta_{ij} = 0$ if there is no edge between traits i and j in the graphical representation. Given the model, parameters are estimated using a Markov Chain Monte Carlo (MCMC) sampler. Finally, a graph relating the different traits are plotted using the posterior probability that two traits are genetically correlated as denoted by $p(E(i,j)|\mathbf{Y})$, and the posterior summary of $\beta_{i,j.}$, where traits i and j are deemed to be correlated (and have a edge between them) if $p(E(i,j)|\mathbf{Y}) > 0.5$ and $\beta_{i,j.}$ is significant. Using this approach graph-GPA provides a parsimonious representation of genetic relationship among traits and shows improved statistical power to identify risk-associated SNPs. Its usefulness is demonstrated by integrating 12 traits (five psychiatric disorders, three autoimmune traits, two lipid-related traits and two cardiovascular traits) where clinically related traits are observed to form closely connected clusters. Despite some benefits, graph-GPA has certain significant limitations. The graph-GPA method does not integrate functional annotation information in its implementation, therefore the impact of functional annotations on the different traits can not be investigated. Also, the MCMC implementation can be time consuming as the number of GWAS traits to be integrated increases.

While both pleiotropy-informed FDR and graph-GPA provide useful methodological

improvements in utilizing multiple GWAS simultaneously, both methods do not integrate functional annotation information in their application. A method that efficiently integrates functional annotation while also leveraging pleiotropy between multiple traits can further improve statistical power to detect risk-associated SNPs and be beneficial.

2.3 Statistical Methods Integrating GWAS Summary Statistics for Multiple Traits by Leveraging Pleiotropy and Functional Annotations

In this section, we will focus on statistical methodologies that integrate GWAS association-p-values for multiple traits by leveraging pleiotropy and functional annotation information. The methods that are discussed here in detail include the genetic analysis incorporating pleiotropy and annotation (GPA) method by Chung et al. [10], EPS or the empirical Bayes approach by Liu et al. [17], the latent probit model (LPM) by Ming et al. [11] and the risk variant inference using epigenomic reference annotation (RiVIERA) method by Li et al. [23].

GPA employs a unified statistical framework to integrate genetically correlated GWAS traits by leveraging pleiotropy and functional annotation data to perform joint analysis. When two genetically correlated GWAS traits are considered by GPA, the binary latent status of association for SNPs are represented as $\mathbf{Z} = \{Z_{00}, Z_{10}, Z_{01}, Z_{11}\}$ forming four groups of SNPs (null for both traits, non-null for the first and null for the second trait, null for the first and non-null for the second trait, and non-null for both traits, respectively). The latent association status is assumed to follow a $Multinomial(1, (\pi_{00}, \pi_{10}, \pi_{01}, \pi_{11}))$ distribution, where π_{00}, π_{10}, π_{01} and π_{11} are the proportion of SNPs in the four groups described previously. The p-values for non-null and null SNPs given their latent status of associations are assumed to come from a Beta-Uniform mixture, where α_1, and α_2 $(0 < \alpha_i < 0, i = 1, 2)$ are the parameters for the $Beta(\alpha_i, 1)$ distribution for the SNPs associated with

the i^{th} trait. Also, given the latent association status of a SNP, its functional annotations are assumed to come from a Bernoulli distribution where the parameters $(q_{00}, q_{10}, q_{01}, q_{11})$ represent the proportion of functional SNPs in each of the four groups. GPA utilizies an EM algorithm to estimate the model parameter estimates and their standard errors (SE). Finally, SNPs can be prioritized using their estimated local FDR and by controlling the global FDR using the direct posterior probability approach [24]. GPA can be efficiently used to integrate a few GWAS traits by leveraging pleiotropy while also integrating functional annotation information. However, the number of parameters that are included in the model increases significantly as the number of GWAS traits and functional annotations increase in a model, making implementation of the method statistically and computationally challenging. Also, the functional annotation enrichment analysis that are performed with GPA only evaluates enrichment of one annotation at a time. Thus, interactions that may be present between functional annotations can not be evaluated, limiting the utility of the method.

Similar to GPA, EPS integrates tissue specific functional annotations while leveraging pleiotropy between multiple traits to prioritize risk genes shared by multiple traits [17]. In EPS, SNP level p-values are grouped together to obtain gene level p-values using VEGAS [25], a tool that corrects for LD while combining the effects of all SNPs in a gene for a gene-based test-statistic or p-value. Then a two-group Beta-Uniform mixture model is assumed for the distribution of G genic p-values $(P_1, ..., P_G)$ where $\mathbf{Z}_g = (Z_{g0}, Z_{g1})$ are the latent binary variables indicating whether the p-value for the g^{th} gene is from the null or non-null group such that $Z_{g0}, Z_{g1} \in 0, 1$ and $Z_{g0} + Z_{g1} = 1$. Using the same principle as the two-group model described earlier for GPA, a four-group model represented by $L = \{00, 01, 10, 11\}$ can be formed for two GWAS traits $(k = 1, 2)$ where the latent variable $\mathbf{Z}_g$ can be denoted as $\mathbf{Z}_g = (Z_{g00}, Z_{g10}, Z_{g01}, Z_{g11})$ indicating that the g^{th} gene is not associated with either trait, associated with only the first trait, associated with only the sec-

ond trait and associated with both traits, respectively with probability $\pi_l = Pr(Z_{gl} = 1)$. To incorporate gene expression data ($\mathbf{E}$) as functional annotations from T tissues where $\mathbf{E} \in \mathcal{R}^{G \times T}$, this model assumes that conditional on the latent status, the expression data is normally distributed such that $\mathbf{E}_g | Z_{gl} = 1 \sim N(\boldsymbol{\mu}_l, \Sigma)$, where $\mathbf{E}_g$ is the vector of gene expression across multiple tissues for the g^{th} gene, $\boldsymbol{\mu}_l$ is a vector of length T where μ_{lt} is the mean of gene expression for the t^{th} tissue in the l^{th} group, and Σ is a covariance matrix for T tissues. The joint distribution can be written as $Pr(\mathbf{P}, \mathbf{E}) = \prod_{g=1}^{G} \Big(\sum_{l \in L} Pr(\mathbf{Z}_{gl} = 1) Pr(\mathbf{P}_g, \mathbf{E}_g | Z_{gl} = 1) \Big) = \prod_{g=1}^{G} \Big(\sum_{l \in L} \pi_l Pr(\mathbf{P}_g | Z_{gl} = 1) Pr(\mathbf{E}_g | Z_{gl} = 1) \Big)$. The complete log-likelihood is derived which is then maximized to estimate the parameters by implementing an EM algorithm. After the parameters are estimated, genes are prioritized based on their local FDRs using the direct posterior probability approach by Newton et al. [24]. Likelihood ratio tests (LRT) are used to test if risk genes are differentially expressed for the t^{th} tissue ($H_0^{(t)} : \mu_{00,t} = \mu_{10,t} = \mu_{01,t} = \mu_{11,t}$) and also to test for pleiotropy between traits ($H_0 : \pi_{11} = (\pi_{10} + \pi_{11})(\pi_{01} + \pi_{11})$).

The third method, LPM by Ming et. al, can also be utilized to integrate multiple GWAS traits to leverage pleiotropy and functional annotation information. The three main goals of LPM is to identify the pleiotropic relationship between multiple traits by estimating the correlation between the traits, to identify the effect of functional annotations, and to improve the power to identify risk-associated SNPs for single and also multiple traits. This method also assumes a Beta-Uniform mixture for the distribution of GWAS p-values (P_{jk}) for j SNPs in the k traits. A latent binary variable η_{jk} is used to indicate the association group that SNP j belongs to for the k^{th} trait. The LPM model is given by $\mathbf{Z_j} = \boldsymbol{\beta} \mathbf{X}_j + \epsilon_j,\ \epsilon_j \sim N(0, \mathbf{R})$, where $\mathbf{Z} \in \mathbb{R}^{M \times K}$ is the matrix of latent variable for M SNPs such that $\eta_{jk} = 1$ when $Z_{jk} > 0,$ or 0 otherwise, $\mathbf{X} \in \mathbb{R}^{M \times (D+1)}$ is the design matrix of the intercept and D functional annotations, $\boldsymbol{\beta} \in \mathbb{R}^{K \times (D+1)}$ is the matrix of

coefficients, ϵ is the unmeasured error term and $\mathbf{R} \in \mathbb{R}^{K \times K}$ is the correlation matrix for the K traits. But, rather than integrating with K GWAS traits simultaneously, this method analyzes the traits in a pair-wise manner based on a composite likelihood approach denoted as bivariate LPM or bLPM. If $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{R}\}$ are the parameters for LPM, then $\tilde{\boldsymbol{\theta}} = \{\tilde{\boldsymbol{\alpha}}, \tilde{\boldsymbol{\beta}}, \tilde{\mathbf{R}}\}$ are the parameters for bLPM. The scalability of this method is improved by using a parameter expanded EM (PX-EM) algorithm for pairwise analysis of traits. The parameters in bLPM can be expanded to $\tilde{\mathbf{Z}}_j = \boldsymbol{\gamma}\mathbf{X}_j + \tilde{\boldsymbol{\epsilon}}_j, \tilde{\boldsymbol{\epsilon}}_j \sim N(0, \Sigma)$, where $\gamma = \mathbf{D}\tilde{\boldsymbol{\beta}}, \Sigma = \mathbf{D}\tilde{\mathbf{R}}\mathbf{D} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$ and $D = \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{pmatrix}$. Once all pair-wise computations and the corresponding parameter estimates $\hat{\tilde{\theta}}$ are obtained using PX-EM, the estimates for $\hat{\alpha}_k$ and $\hat{\beta}_k$ in LPM are obtained by averaging over the pairs that include the k^{th} GWAS. The $\hat{\mathbf{R}}$ matrix for correlation is also formed using the estimates of $\hat{\rho}$ from the pair-wise analysis. Finally, SNPs associated with k and k' traits are inferred using $Pr(\eta_{jk} = 1, \eta_{jk'} = 1 | P_{jk}, P_{jk'}, \mathbf{X}; \boldsymbol{\theta})$ and SNPs associated with only the k^{th} trait are inferred using $Pr(\eta_{jk} = 1 | P_{jk}, \mathbf{X}; \boldsymbol{\theta})$, controlling the global FDR using the direct posterior probability approach by Newton et al. [24]. Relationship between two traits in a pair is evaluated using a likelihood ratio test as $\lambda = 2\, log(\frac{Pr(\tilde{\mathbf{P}}|\mathbf{X};\tilde{\boldsymbol{\theta}})}{Pr(\tilde{\mathbf{P}}|\mathbf{X};\tilde{\boldsymbol{\theta}}_0)})$, where $\tilde{\boldsymbol{\theta}}$ is estimated under the alternate hypothesis ($\rho \neq 0$), $\tilde{\boldsymbol{\theta}}_\mathbf{0}$ is estimated under the null hypothesis ($\rho = 0$), and $\lambda \sim \chi_1^2$ asymptotically under the null. Finally, annotation enrichment on each trait is evaluated using the Wald test statistic (W) such that $W \sim \chi_1^2$ asymptotically under the null ($\beta_{kd} = 0$). LPM was efficiently used to analyze 44 GWAS traits with 136 functional annotations.

The risk variant inference using epigenomic reference annotation (RiVIERA) method by Li et al. [23] is a Bayesian method that integrates functional annotations and performs joint analyses of multiple GWAS traits. In this method, the empirical prior of a SNP being

associated with trait d is defined using a logistic function as $\pi_d = [1 + \exp(-[\sum_k w_{0d} + w_{kd}e_{vk})]^{-1}$, where w_{0d} denotes the linear bias and w_{kd} is interpreted as the enrichment coefficient for the k^{th} annotation in the d^{th} trait. A non-negative value is enforced for this parameter during estimation. The effects for annotations are assumed to follow a multivariate normal distribution, $w_{kd} \sim N(0, \Lambda_w^{-1})$, where Λ_w captures the pairwise annotation correlation among D traits. Also, $w_{0d} \sim N(\text{logit}(\pi_0), \lambda_{0d}^{-1})$, where $\lambda_{w_{0d}} \sim \Gamma(\alpha_0, \beta_0)$ and $\text{logit}(\pi_0) = log\frac{\pi_0}{1-\pi_0}$. Also the p-values ($a_{vd}$) are modeled using a re-parameterized Beta distribution with mean μ_d and unknown precision ϕ_d as $a_{vd} \sim \mathcal{B}(\mu_d, \phi_d)$. A joint posterior distribution function is derived using the assumed model distributions. Gibbs sampling is then used to sample parameter estimates from the joint posterior distribution. Causal variants are inferred using a_{vd} and fold enrichment for all annotations are evaluated using the full prior model over the alternative prior where the effect of annotation k for trait d is removed.

All methods described in this section integrate GWAS summary statistics for multiple traits by leveraging pleiotropy and functional annotation information. However, the number of parameters to be estimated increases significantly in all discussed methods as more annotations and GWAS traits are integrated together which can make these methods statistically complex and computationally challenging. The complexity of the models can increase even further when all possible interactions between annotations are included. Additionally, while these methods can test individual effects of functional annotations on a trait etiology, it can be computationally and biologically taxing to investigate the impact that possible combinations or interactions of functional annotations have on a trait etiology. Therefore, a method that can automatically select the combinations of functional annotations can be beneficial in understanding the functional complexity of a trait.

3. Specific Aim 1

For Aim 1, our goal is to develop statistical methodology to prioritize SNPs that are associated with a single trait and to identify combinations of functional annotations that can explain the mechanisms through which risk-associated SNPs are associated with a single trait.

3.1 Introduction

In this aim, we address the challenges posed by missing heritability, polygenicity and missing functional information about trait risk-associated SNPs that are discussed in Section 1.1 by integrating GWAS summary statistics and functional annotation data for a single trait. Several Bayesian and mixed model methods have been employed to integrate GWAS summary statistics and functional annotation data as discussed in Section 2.1. Although these methods are successful in prioritizing trait risk-associated SNPs and also in identifying relevant functional annotations, they do not provide knowledge about interactions between different functional annotations. Even in methods that can include interactions, interactions need to be user-specified. However, this requires strong prior scientific knowledge, which is often lacking, especially when a large number of functional annotations is considered in the analysis.

Our goal in Aim 1 is to address the shortcomings described above by developing a novel statistical approach called GPA-Tree that simultaneously prioritizes trait-associated SNPs and identifies key combinations of functional annotations related to the mechanisms through which trait-associated SNPs influence the trait, within a unified framework. Specifically, GPA-Tree is based on a hierarchical modeling approach integrated with a decision

tree algorithm and facilitates easy interpretation of findings. GPA-Tree takes GWAS summary statistics as input, which allows wide applications and adaptations. Our comprehensive simulation studies and real data applications show that GPA-Tree consistently improves statistical power to detect trait-associated SNPs and also effectively identifies biologically important combinations of functional annotations.

The chapter is structured as follows. In Section 3.2, we present background information about classification and regression tree (CART) and EM algorithm. In Section 3.3, we introduce our method for prioritizing trait risk-associated SNPs and identifying combinations of functional annotations that are associated with the risk SNPs. In Sections 3.4 and 3.5, we describe our simulation settings and simulation results, respectively. In section 3.6, we describe the results of the application of our method to real data. Finally, in Section 3.7 we discuss the implications of using our method.

3.2 Background

In the implementation of our novel method, GPA-Tree, we use a unified framework combining a decision tree algorithm and an EM algorithm. Although many decision tree procedures are available, we employ the classification and regression tree (CART) framework by Breiman et al. [26] for specific Aim 1 because of its suitability for the type of response considered, ease of use and intuitive interpretation of results that are presented in a tree like structure. CART is a suitable choice in comparison to other decision tree methods like ID3 (Iterative Dichotomiser 3) [27], CHAID (Chi-Squared Automatic Interaction Detection) [28] and QUEST (Quick, Unbiased, Efficient, Statistical Tree) [29] as these methods do not allow continuous response variables. Also, GWAS association data for complex traits do not usually provide good signal strength and therefore are not a good fit to be analyzed using other decision tree methods like Random Forest [30] and Logic Regres-

sions [31] as randomness is introduced in the implementation of these methods which can potentially miss the limited signal strength present in GWAS for complex traits. CART also offers several advantages to other commonly used methods like generalized linear mixed models (GLMM). CART can perform automatic variable selection and identification of interactions between the selected variables without *a priori* specification. In contrast, GLMM requires *a priori* specification of variables and interactions of interest. Also, GLMM assumes linearity between the response and predictor variables through a link function. However, CART models do not require such assumptions. CART models can be easily implemented using the $rpart$ package in `R` [32].

CART models are flexible and can be used with binary, discrete or continuous response and predictor variables. Trees obtained from a CART model with binary or discrete response variable are called classification trees and trees obtained from a CART model with continuous response variable are called regression trees. In the implementation of GPA-Tree, we will employ the regression framework of CART as we utilize continuous response and binary predictor variables. In both regression and classification trees, the location for each predictor variable in the tree is called a node. Each node in the tree can have two or zero sub-node. The two sub-nodes of a node are its children and the sub-nodes are each other's siblings. The node without a parent node in the tree is called a root node and nodes without children are called leaves or terminal nodes.

The regression framework of CART uses a greedy approach to identify all predictor variables to be included at the nodes of the regression tree. Greedy search is performed by evaluating all predictor variables at all possible split points for binary space partitioning of the data, and then selecting the predictor variable at the split point that minimizes the cost function. As we utilize binary functional annotations as predictor variables in the implementation of the GPA-Tree approach, there is only one possible split to be considered for all predictor variables. That is, for each predictor variable, we can have two rectangular

sub-space (0 vs 1) where observations can fall. We can calculate the predicted values for the subset of data in the two rectangular sub-space and use it in the formula for the cost function. The cost function for the regression framework is defined as the sum of squared error for all observations i.e., $cost = \sum_{i=1}^{M}(y_i - \hat{y}_i)^2$, where y_i is the observed response, and $\hat{y}_i$ is the predicted response for the rectangular sub-space in which the i^{th} observation lies. The first predictor variable that is identified by CART using greedy search is the root of the regression tree. Other predictor variables are added to the regression tree by recursively partitioning the rectangular sub-spaces even further, creating smaller subsets of data that are used to calculate the cost function at sub-nodes, allowing only the predictor variables that minimize the cost function at each sub-node. Data partitioning is continued and child nodes are added to all nodes of the tree until some stopping criteria is satisfied by the subset of data at each node. Some examples of stopping criteria that are used when building CART models are: setting a threshold for the minimum number of observations required to create a sub-node, setting a threshold for the minimum required improvement in the cost function, and pre-specifying the maximum number of nodes allowed in a tree.

Based on the stopping rules used to build a CART model, we can have complex or simple trees. Complex trees are large trees with many splits and may contain uninformative splits that are not worthwhile. Likewise, simple trees are small trees with few or no splits. We can alter the size of a tree by pruning or growing the tree using a complexity parameter (cp). When pruning a regression tree, a child node is removed if the value of the cost function at the child node is lower than cp. When growing a regression tree, a node can grow children nodes if the cost function for a child node improves by at least cp. However, determining the appropriate value of cp to be used for a given type of data can be a challenge and needs some investigating.

We combine CART and EM algorithm to create a unified framework for the implementation of the proposed GPA-Tree approach. EM algorithm is appropriate to use when latent

variable, observed data and unknown parameters that need to be estimated are involved. Since the goal of the GPA-Tree approach is to integrate GWAS association p-values (observed data) and functional annotation data to prioritize the SNPs that are associated with a trait of interest (using a latent variable) and also to predict the parameter estimates (related to the distribution of the null and non-null groups of association), EM algorithm is an appropriate choice.

EM algorithm is an iterative, two-step procedure that was first explained in 1977 by Dempster et al. [33]. To implement the EM algorithm, we being by writing the complete and incomplete data likelihoods. Given some observed data ($\mathbf{Y}$), an unobserved latent variable ($\mathbf{Z}$) and a vector of unknown parameters (θ) related to the null and non-null groups that generate the data, we can define the complete-data likelihood function as $L_C(\theta|\mathbf{Y},\mathbf{Z}) = p(\mathbf{Y},\mathbf{Z}|\theta)$. Similarly, we can define the incomplete-data likelihood function as $L_{IC}(\theta|\mathbf{Y}) = \int p(\mathbf{Y},\mathbf{Z}|\theta)\, d\mathbf{Z}$ such that L_{IC} is monotone increasing (a notable feature of the EM algorithm). Next, in the E-step of the EM algorithm, we compute the expected value of the complete-data log likelihood function of θ, $log(\ p(\mathbf{Y},\mathbf{Z}|\theta))$, with respect to the conditional distribution of the latent variable $\mathbf{Z}$ given the observed data $\mathbf{Y}$ and the parameter estimates from the previous iteration $\theta^{(t)}$ as defined by $z_i^{(t)} = E(Z_i|\mathbf{Y},\theta^{(t)}) = E_{\mathbf{Z}|\mathbf{Y},\,\theta^{(t)}}\left[log\left(p(\mathbf{Y},\mathbf{Z}|\theta^{(t)})\right)\right]$. In the M-step, we find the parameter estimates that maximize the $z_i^{(t)}$ function from the E-step as $\theta^{(t+1)} = arg\ \max_{\theta^{(t)}} E(Z_i|\mathbf{Y},\theta^{(t)})$. We repeat the iterative process of computing E and M steps until the algorithm converges. However, given the complexity of the data likelihoods and absence of closed form solutions to maximize in the M-step, computations involved in the different steps of EM can be gruesome and convergence of the algorithm can be slow.

Several variations of EM have been proposed and implemented over time to overcome convergence and computational issues pertaining to the original EM algorithm [34–37]. A variant of the EM algorithm that was utilized in the implementation of the GPA-Tree

approach is called generalized EM or GEM [33]. The E-step of GEM remains the same as the E-step of a conventional EM. However, in the M-step, we find $\theta^{(t+1)}$ that increases L_{IC} in the $(t+1)^{th}$ iteration, i.e., we choose $\theta^{(t+1)}$ such that $L_{IC}^{(t+1)}|\theta^{(t+1)} > L_{IC}^{(t)}|\theta^{(t)}$ while also maximizing $L_C^{(t+1)}$. This is in contrast to a conventional EM algorithm in which we find $\theta^{(t+1)}$ that maximizes $E(Z_i|\mathbf{Y}, \theta^{(t)})$ in the M-step. Utilizing GEM is useful in maintaining the monotone increasing property of L_{IC} that may not always be achieved when a conventional EM algorithm is implemented with CART.

3.3 GPA-Tree Method

3.3.1 Model

Let $\mathbf{Y}_{M\times 1} = (Y_1, Y_2, \ldots, Y_M)'$ be a vector of genotype-trait association *p*-values for $i = 1, 2, \cdots, M$ SNPs such that y_i denotes the *p*-value for the association of the i^{th} SNP with the trait. We also assume that we have K binary annotations ($\mathbf{A}$).

$$\mathbf{A} = (\mathbf{A}_{.1}, \ldots, \mathbf{A}_{.K}) = \begin{pmatrix} a_{11} & \ldots & a_{1K} \\ \vdots & \ddots & \vdots \\ a_{M1} & \ldots & a_{MK} \end{pmatrix}_{M\times K}, \quad \text{where}$$

$$a_{ik} = \begin{cases} 0, & \text{if } i^{th} \text{ SNP is not annotated in the } k^{th} \text{ annotation} \\ 1, & \text{if } i^{th} \text{ SNP is annotated in the } k^{th} \text{ annotation} \end{cases}$$

For example, if $\mathbf{A}_{.k}$ is the annotation for an open chromatin region in blood samples, the i^{th} SNP is said to be 'annotated' for the k^{th} annotation if it belongs to the open chromatin region and $A_{ik} = 1$. If the i^{th} SNP does not belong to the open chromatin region, it is considered to be 'not annotated' and $A_{ik} = 0$.

Here our ultimate goal is association mapping, i.e., identifying SNPs associated with the trait given both GWAS and functional annotation data. To accomplish this, we introduce the latent variable $\mathbf{Z}$, where z_i indicates association of i^{th} SNP with the trait.

$$z_i = \begin{cases} 0, & i^{th} \text{ SNP is not associated with the trait; null group} \\ 1, & i^{th} \text{ SNP is associated with the trait; non-null group} \end{cases}$$

Then, the GWAS association *p*-values (y_i) are assumed to come from a mixture of non-risk-associated ($z_i = 0$) and risk-associated groups ($z_i = 1$). As previously proposed by Chung and colleagues [10], if the i^{th} SNP belongs to the non-risk-associated group ($z_i = 0$), then its *p*-value is assumed to come from the Uniform distribution on $[0,\ 1]$. This is based on the rationale that $U[0,\ 1]$ provides a *p*-value density corresponding to equal probability of all values on the interval $[0,\ 1]$ signifying 'no signal' in the data from the non-risk-associated group [38]. If the i^{th} SNP belongs to the risk-associated group ($z_i = 1$), then its *p*-value is assumed to come from the Beta distribution with parameters ($\alpha,\ 1$), where $0 < \alpha < 1$. We restrict α in the Beta distribution to be between 0 and 1 because the smaller α value corresponds to the higher density at lower *p*-values and the lower density at higher *p*-values, while the α value closer to one resembles a $Unif[0,\ 1]$ distribution, making signal in the data from $Beta(\alpha, 1)$ closer to those from the non-risk-associated group.

$$\begin{aligned} (y_i|z_i = 0) &\sim U[0,1], \\ (y_i|z_i = 1) &\sim Beta(\alpha, 1), \quad 0 < \alpha < 1 \end{aligned}$$

We further integrate functional annotation data with the GWAS data by modeling the latent $\mathbf{Z}$ as a function of the functional annotation data $\mathbf{A}$. Specifically, we define a function f that is a combination of functional annotations $\mathbf{A}$ and relate it to the expectation of latent

Z as given in Equation (3.1).

$$P(Z_i = 1; a_{i1}, ..., a_{iK}) = f(a_{i1}, ..., a_{iK}) \tag{3.1}$$

The flowchart in Fig 3.1 provides a complete graphical representation for these data.

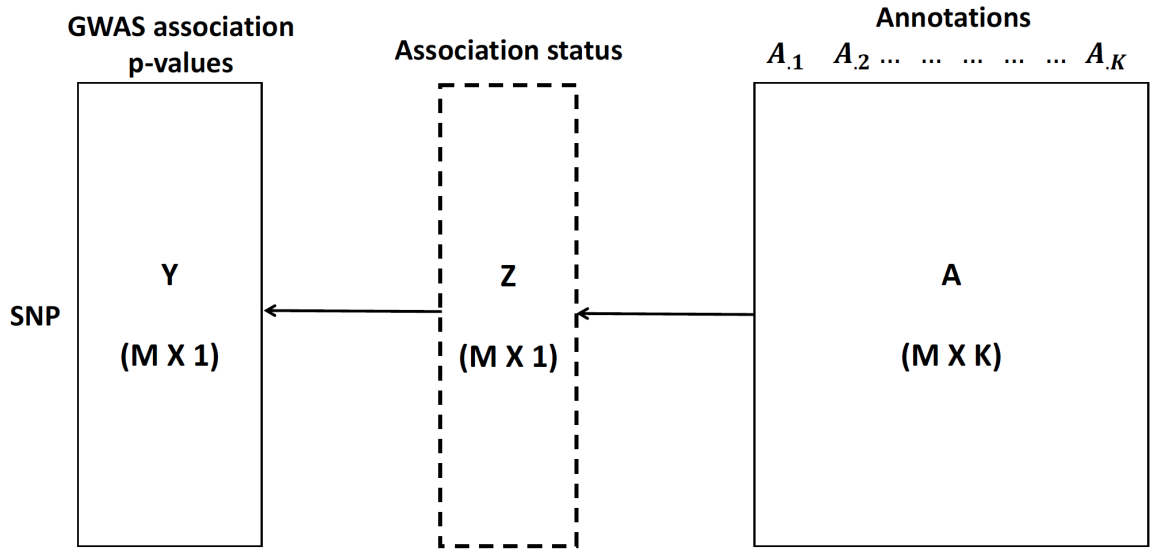


Figure 3.1: Association framework that links the GWAS association p-values ($\mathbf{Y}$), the association status as given by the latent variable ($\mathbf{Z}$) and annotation data ($\mathbf{A}$).

Let $\theta = (\alpha, \boldsymbol{\pi})$, where $\boldsymbol{\pi} = \{\pi_1, \pi_2, ..., \pi_M\}$ is a function of $\mathbf{A}$ and represents the prior probabilities that the SNPs belong to the risk-associated group, i.e., $\pi_i = P(Z_i = 1)$. Assuming that the SNPs are independent, we can write the joint distribution of the observed data $Pr(\mathbf{y}, \mathbf{A})$ as:

$$\begin{aligned} Pr(\mathbf{y}, \mathbf{A}) &= \prod_{i=1}^{M} [P(Z_i = 1)P(y_i|Z_i = 1) + P(Z_i = 0)P(y_i|Z_i = 0)] \\ &= \prod_{i=1}^{M} \left[\pi_i \, \alpha \, y_i^{\alpha-1} + (1 - \pi_i)\right] \end{aligned}$$

The 'incomplete' data log-likelihood is written as:

$$\begin{aligned}\ell_{IC} &= \sum_{i=1}^{M} log\left[P(Z_i = 1)P(y_i|Z_i = 1) + P(Z_i = 0)P(y_i|Z_i = 0)\right] \\ &= \sum_{i=1}^{M} log\left[\pi_i\, \alpha\, y_i^{\alpha-1} + (1 - \pi_i)\right]\end{aligned}$$

We can write the 'complete' data likelihood as:

$$L_C = \prod_{i=1}^{M} \left[\pi_i\, \alpha\, y_i^{\alpha-1}\right]^{Z_i} \left[(1 - \pi_i)\right]^{1-Z_i}$$

Similarly, the 'complete' data log-likelihood can be written as:

$$\ell_C = \sum_{i=1}^{M} Z_i\, (log\, \pi_i + log\, \alpha + (\alpha - 1)\, log\, y_i) + (1 - Z_i)\, log(1 - \pi_i)$$

3.3.2 Algorithm

Given the approach described in Section 3.3.1, we implemented parameter estimation using an EM algorithm. The function f in Equation (3.1) is estimated by a decision tree algorithm and it allows to identify combinations of functional annotations related to risk-associated SNPs. To improve stability, we employed a two-stage approach for parameter estimation. Specifically, in Stage 1, we first estimate the parameter α without identifying a combination of functional annotations. Then, in Stage 2, we identify key combinations of functional annotations ($f(\mathbf{A})$) while the parameter α is kept fixed as the value obtained in the first step. We illustrate more detailed calculation steps below.

Stage 1:

In Stage 1, we initialize $\alpha^{(0)} = 0.1$ and $\pi_i^{(0)}$ as given below.

$$\pi_i^{(0)} = \begin{cases} 0.9, \ y_i \leq 10^{-4} \\ 0.1, \ y_i > 10^{-4} \end{cases}$$

For the i^{th} SNP, the t^{th} iteration of the E-step can be written as:

$$\begin{aligned} \mathbf{E-step}: \mathbf{z_i^{(t)}} &= \mathbf{E}[\mathbf{Z_i};\ \mathbf{Y}, \mathbf{A}, \boldsymbol{\theta}^{(\mathbf{t-1})}] \\ &= Pr(Z_i = 1;\ \mathbf{Y}, \mathbf{A}, \boldsymbol{\theta}^{(t-1)}) \\ &= \frac{P(Y_i;\, Z_i{=}1, \boldsymbol{\theta}^{(t-1)}) P(Z_i{=}1; \mathbf{A}_{i.}, \boldsymbol{\theta}^{(t-1)})}{\sum\limits_{d \in \{1,0\}} P(Y_i;\, Z_i{=}d, \boldsymbol{\theta}^{(t-1)}) P(Z_i{=}d;\, \mathbf{A}_{i.}, \boldsymbol{\theta}^{(t-1)})} \\ &= \frac{\alpha^{(t-1)} y_i^{\alpha^{(t-1)}-1} \pi_i^{(t-1)}}{1 - \pi_i{}^{(t-1)} + \alpha^{(t-1)} y_i^{\alpha^{(t-1)}-1} \pi_i^{(t-1)}} \end{aligned}$$

In the t^{th} iteration of the M-step, π_i and α are updated as:

$$\begin{aligned} \mathbf{M-step}: &\text{ Fit a linear regression model as} \\ &z_i^{(t)} = \beta_0^{(t)} + \beta_1^{(t)} a_{i1} + \cdots + \beta_K^{(t)} a_{iK} + \epsilon_i^{(t)} \\ &\text{Update } \pi_i^{(t)} \text{as the predicted value from the linear} \\ &\text{regression model.} \\ &\text{Update } \alpha^{(t)} = -\frac{\sum\limits_{i=1}^{M} z_i{}^{(t)}}{\sum\limits_{i=1}^{M} z_i{}^{(t)} log(y_i)}, \end{aligned}$$

where $\beta_k^{(t)}, k = 0, \cdots, K$ are the regression coefficients and $\epsilon_i^{(t)}$ is the error term. The E and M steps are repeated until both the incomplete log-likelihood and the α estimate converge. The α and $\boldsymbol{\pi}$ estimated in this stage are used to fix α and initialize $\boldsymbol{\pi}$, respectively, in Stage 2.

Stage 2:

In this stage, we implement another EM algorithm employing a decision tree algorithm

(CART [26]), which allows to identify union, intersection, and complement relationships between functional annotations in estimating π_i.

For the i^{th} SNP, the t^{th} iteration of the E-step can be written as:

$$\mathbf{E-step}: z_i^{(t)} = \frac{\hat{\alpha} y_i^{\hat{\alpha}-1} \pi_i^{(t-1)}}{1-\pi_i{}^{(t-1)} + \hat{\alpha} y_i^{\hat{\alpha}-1} \pi_i^{(t-1)}}$$

Note that here α is fixed as $\hat{\alpha}$, which is the final estimate of α obtained from Stage 1. In the t^{th} iteration of the M-step, π_i is updated as:

$$\begin{aligned} &\mathbf{M-step}: \text{Fit a CART model as} \\ &\qquad z_i^{(t)} = f^{(t)}(a_{i1}, \cdots, a_{iK}) + \epsilon_i^{(t)} \qquad (3.2) \\ &\qquad \text{Update } \pi_i^{(t)} \text{as the predicted value from the CART model,} \end{aligned}$$

where ϵ_i is the error term. In the M-step, the complexity parameter (cp) is the key tuning parameter and defined as the minimum improvement that is required at each node of the tree. Specifically, in the CART model, the largest possible tree (i.e., a full-sized tree) is first constructed and then pruned using cp. The pruned regression tree structure identified by the CART model upon convergence of the EM algorithm (Equation (3.2)) is used as f in Equation (3.1). This approach allows for the construction of the accurate yet interpretable regression tree that can explain relationships between functional annotations and genotype-trait associations. The E and M steps are repeated until the incomplete log-likelihood converges.

We note that unlike the standard EM algorithm, the incomplete log-likelihood in Stage 2 is not guaranteed to be monotonically increasing. Therefore, we implement Stage 2 as a generalized EM algorithm by retaining only the iterations in which the incomplete log-likelihood increases compared to the previous iteration.

3.3.3 Prioritization of Risk-associated SNPs and Identification of Relevant Combinations of Functional Annotations

Once the parameters are estimated as described in Section 3.3.2, we can now prioritize risk-associated SNPs and identify combinations of functional annotations relevant to these SNPs. First, SNPs are prioritized using the local false discovery rate, fdr, which is defined as the posterior probability that the i^{th} SNP belongs to the non-risk-associated group given its GWAS *p*-value and functional annotation information, i.e., $fdr(Y_i, \mathbf{A}_{i.}) = P(Z_i = 0;\ Y_i, \mathbf{A}_{i.}) = 1 - P(Z_i = 1;\ Y_i, \mathbf{A}_{i.})$. When using the fdr control, SNPs with $fdr(Y_i; \mathbf{A}_{i.}) \leq \tau$, where τ is the predetermined fdr control level, are mapped to be associated with the trait. When using the global false discovery rate control, FDR, we utilize the 'direct posterior probability' approach [24]. FDR is defined as the expected ratio of the number of SNPs that are incorrectly predicted to be risk-associated SNPs (false positives) compared to the number of SNPs that are predicted to be risk-associated SNPs (positives). In this approach, SNPs are first sorted by their fdr in an ascending order, denoted as h_i. The threshold for fdr, κ, is then increased from 0 to 1 until

$$FDR = \frac{\sum_{i=1}^{M} h_i \, 1\{h_i \leq \kappa\}}{\sum_{i=1}^{M} 1\{h_i \leq \kappa\}} \leq \tau,$$

where τ is the predetermined level of FDR (e.g., $\tau \leq 0.05$). Finally, SNPs with $h_i \leq \kappa$ are considered to be risk-associated SNPs. Second, relevant combinations of annotations are inferred based on the combination of functional annotations selected by the CART model upon convergence of the EM algorithm in Stage 2.

3.4 Simulation Study Design

We conducted a simulation study to evaluate the performance of the proposed GPA-Tree approach. Parameters considered in the simulation study included the number of SNPs (M), the number of functional annotations (K), the relevant combination of annotations, the percentage of SNPs annotated for each functional annotation (u), and the percentage of the annotated SNPs shared between functional annotations (v). For all the simulation data, the number of SNPs was set to $M = 100,000$, the number of annotations was set to $K = 75$, and risk-associated SNPs were assumed to be characterized with the combinations of functional annotations defined by $L = (A_1 \cap A_2) \cup (A_3 \cap A_4)$; all the remaining functional annotations ($A_k, k = 5, .., 75$) were considered to be noise annotations. The percentage of annotated SNPs (u) for annotations $A_1 - A_4$ was set to $2\%, 6\%, 10\%, 14\%$ and 20%, while the percentage of overlap between the true combinations of functional annotations (v) was set to $12.5\%, 25\%, 50\%, 75\%$ and 87.5%. For example, when $M = 100,000$, $K = 75$, $u = 20\%$ and $v = 50\%$, the simulated data include $20,000$ SNPs that are annotated for functional annotation $A_1 - A_4$. Among these $20,000$ SNPs, $10,000$ SNPs are annotated for both A_1 and A_2 $(A_1 \cap A_2)$ and another $10,000$ SNPs are annotated for both A_3 and A_4 $(A_3 \cap A_4)$ resulting in $20,000$ SNPs that are annotated altogether for the defined combination L. For noise annotations $A_5 - A_{75}$, approximately 20% of SNPs were annotated by first generating the proportion of annotated SNPs from $Unif[0.1, 0.3]$ and then randomly setting this proportion of SNPs to one. The SNPs that satisfy the functional annotation combination L were assumed to be risk-associated SNPs and their *p*-values were simulated from $Beta(\alpha,\ 1)$ with $\alpha = 0.7$. The SNPs that do not satisfy L were assumed to be non-risk SNPs and their *p*-values were simulated from $U[0,\ 1]$. Note that here the signal-to-noise ratio is affected by u and v. Figure 3.2 provides a graphical depiction of the simulation setting.

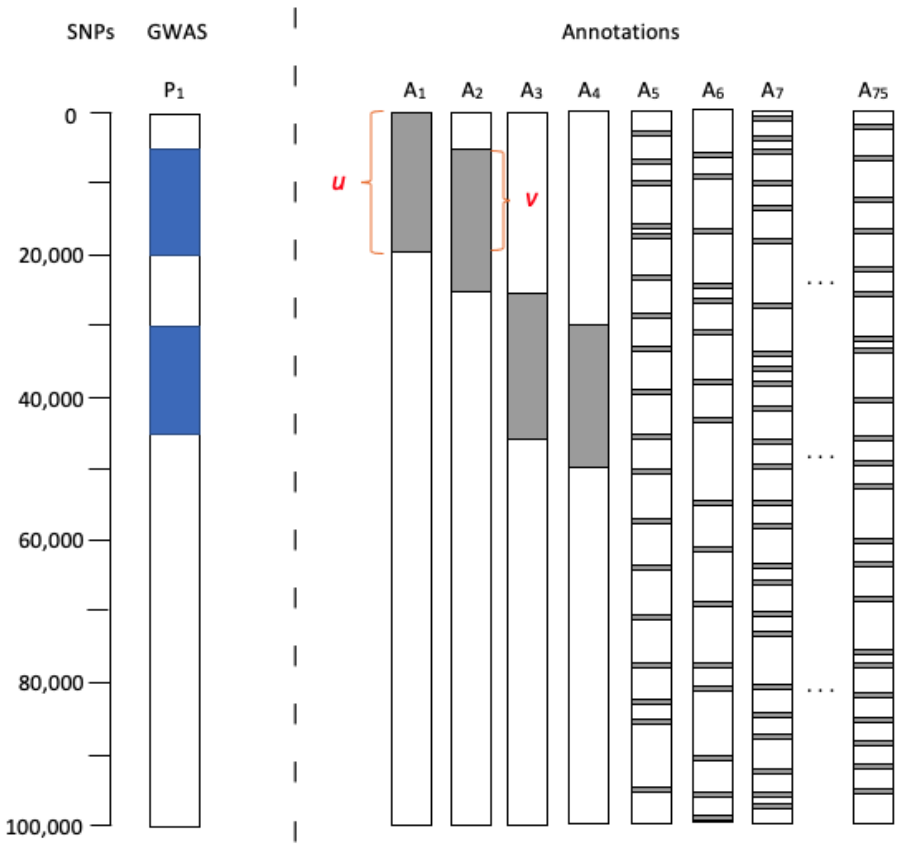


Figure 3.2: Simulation setting with $K = 75$ functional annotations ($A_1 - A_{75}$). The functional annotations $A_1 - A_4$ are assumed to be related to risk-associated SNPs. For each of $A_1 - A_4$, $u\%$ SNPs are assumed to be annotated. In addition, $v\%$ of the annotated SNPs are assumed to be shared between A_1 and A_2, and also between A_3 and A_4. The remaining functional annotations ($A_5 - A_{75}$) are assumed to be unrelated to risk-associated SNPs and approximately 20% of the SNPs are annotated at random. SNPs that satisfy $L = (A_1 \cap A_2) \cup (A_3 \cap A_4)$ (blue SNPs) are assumed to be risk-associated SNPs and their *p*-values were simulated from $Beta(\alpha,\ 1)$ with $\alpha = 0.7$. Remaining SNPs were assumed to be non-risk SNPs and their *p*-values were simulated from $U[0,\ 1]$.

3.5 Simulation Study Results

For each combination of the simulation parameters described in Section 3.4, we simulated 100 datasets and compared the performance of GPA-Tree with LPM [11] and LSMM [9]. The metrics for comparing the methods include (1) area under the curve (AUC), where the curve was created by plotting the true positive rate (sensitivity) against the false positive rate (1-specificity) to detect risk-associated SNPs when global FDR was controlled at various levels; (2) statistical power to identify risk-associated SNPs when global FDR was controlled at the nominal level of 0.05; and (3) estimation accuracy for α parameter in the $Beta(\alpha,\ 1)$ distribution used to generate the *p*-values of risk-associated group. For GPA-Tree, we also examined the accuracy of detecting the correct functional annotation tree, based on the proportion of simulation data for which all relevant functional annotations in L ($A_1 - A_4$) were identified simultaneously, and the average proportion of noise functional annotations ($A_5 - A_{75}$) among the functional annotations identified by GPA-Tree. Here we especially investigate how the percentage of SNPs annotated in $A_1 - A_4$ (u) and the overlap between SNPs annotated in $A_1 - A_2$ and $A_3 - A_4$ (v) impact GPA-Tree's ability to separate functional annotations relevant to the risk-associated SNPs from noise annotations.

AUC: Figure 3.3A shows the AUC comparison between GPA-Tree, LPM, and LSMM. For all the combinations of u and v, GPA-Tree showed the consistently higher AUC relative to LSMM while performing comparably or better than LPM. The performance of LPM and LSMM improved as signal-to-noise ratio increases (i.e., as u and v increase), demonstrating performance closer to GPA-Tree.

Statistical power: Figure 3.3B compares the power to detect true risk-associated SNPs when global FDR is controlled at 0.05 for the three methods. GPA-Tree showed higher statistical power to detect true risk-associated SNPs relative to LPM and LSMM for almost all combinations of u and v. The estimated power for GPA-

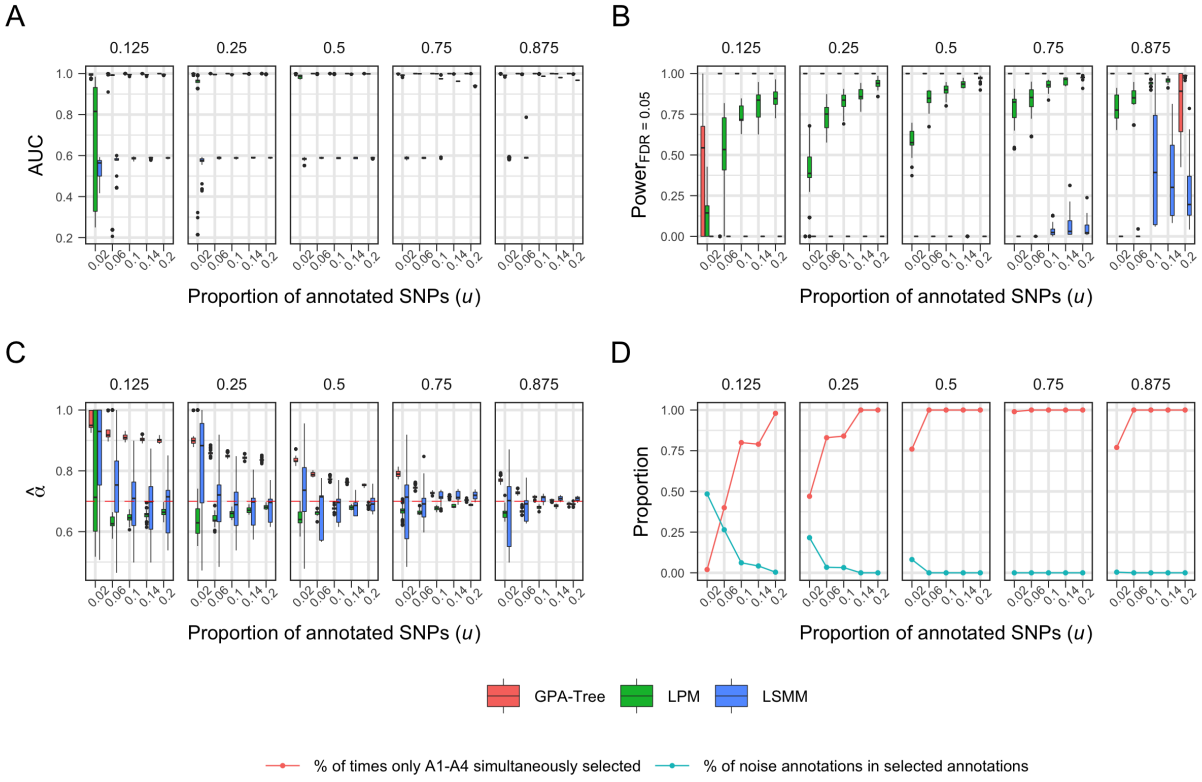


Figure 3.3: Comparison of (A) AUC, (B) statistical power to detect true risk-associated SNPs when global FDR is controlled at the nominal level of 0.05, (C) estimated α parameter, and (D) proportion of times only true functional annotations $A_1 - A_4$ are simultaneously identified by GPA-Tree (red line) and the average proportion of noise annotations ($A_5 - A_{75}$) among the functional annotations identified by GPA-Tree (blue line). The results are presented for different proportions of SNPs annotated in $A_1 - A_4$ (u; x-axis) and proportions of the overlap between SNPs annotated in $A_1 - A_2$ and $A_3 - A_4$ (v; panel). $M = 100,000$, $K = 75$, and $\alpha = 0.7$ in $Beta(\alpha, 1)$ and results are summarized from 100 replications.

Tree was relatively more variable for $u = 2\%$ and $v = 12.5\%$ but it still outperformed LPM and LSMM. The statistical power of LPM increased as a function of u for all v, and the statistical power of LSMM increased as u increases for higher v. However, both LPM and LSMM showed greater variability in statistical power compared to GPA-Tree and on average they showed lower statistical power compared to GPA-Tree.

Estimation of parameter α: Figure 3.3C shows the α parameter estimates obtained from the three methods. GPA-Tree showed less variability in the α estimates compared to LPM and LSMM. LPM was on average more accurate than GPA-Tree in estimating α, however it still often underestimated α. LSMM showed decreased variability in estimation of α as u increases, and estimated α well for higher u and v levels. GPA-Tree generally overestimated α and this was most notable when u and v are small. As u and v increase, α estimates from GPA-Tree became closer to the true value. When u and v are large ($u \geq 10\%$ and $v \geq 75\%$), GPA-Tree estimated α accurately. We note that overestimation of α by GPA-Tree did not impact the method's ability to identify the true combinations of functional annotations or the risk-associated SNPs, which are the main objectives of GPA-Tree.

Selection of relevant and noise annotations: The red line in Figure 3.3D shows the proportion of times only functional annotations in the true combination L ($A_1 - A_4$) were simultaneously identified by GPA-Tree while the blue line shows the proportion of noise annotations ($A_5 - A_{75}$) that were also selected. Excluding instances when signal in the data is really weak ($u \leq 6\%$ and $v \leq 25\%$), GPA-Tree successfully identified all functional annotations included in the true combination L more than 75% of the time. Moreover, GPA-Tree could identify all functional annotations included in the true combination approximately 100% of the time as u or v get

larger (Figure 3.3D, red line). These results demonstrate the potential of GPA-Tree to correctly identify true annotations as long as signal in the data is not too weak. In instances where GPA-Tree did not identify all functional annotations included in L, it either identified one or more noise annotations in addition to the true annotations (false positives), or failed to identify one or more annotations in L (false negative) (Figure 3.3D, blue line).

3.6 Real Data Application: Systemic Lupus Erythematosus

Systemic lupus erythematosus (SLE) is an autoimmune trait caused due to the immune system attacking its own tissue. Various environmental, hormonal, and genetic factors are attributed to SLE [39]. It is also known to disproportionately affect women of childbearing age and individuals of non-white racial groups [40]. SLE can impact a patient's joints, blood cells, and internal organs like heart, lungs and kidneys. According to the Lupus Foundation (www.lupus.org) the symptoms for SLE include but is not limited to inflammation, fatigue, pain or swelling in the joints, headaches, and chest pain when breathing deeply. SLE can be challenging because its symptoms can not be permanently cured, but only be minimized through therapeutics and lifestyle changes.

To understand the complex genetic architecture of SLE, multiple whole genome GWAS studies have been proposed and implemented [41, 42]. We applied the GPA-Tree approach to the SLE GWAS data [42] sourced from the GWAS Catalog [1] . Summary statistics were originally obtained using the genotyped and imputed Immunochip, profiled for $18,264$ individuals ($6,748$ cases and $11,516$ controls) of European ancestry. Altogether $336,745$ SNPs with measure of the observed statistical information associated with the allele frequency estimate between 0.9 and 1 (i.e., $0.9 \leq info \leq 1$) and the average certainty of best-guess genotypes equaling 1 (i.e., $certainty = 1$) passed quality control criteria. Af-

ter excluding SNPs located in the MHC region, $293,976$ SNPs were included in the final analysis and integrated with functional annotation data from GenoSkyline (GS) [43] and GenoSkylinePlus (GSP) [44]. By providing tissue- and cell-type specific functionality, GS and GSP annotations can potentially add useful insights to understanding the etiology of SLE. The Manhattan plot and *p*-value histogram for SLE GWAS data are presented in Figure 3.4.

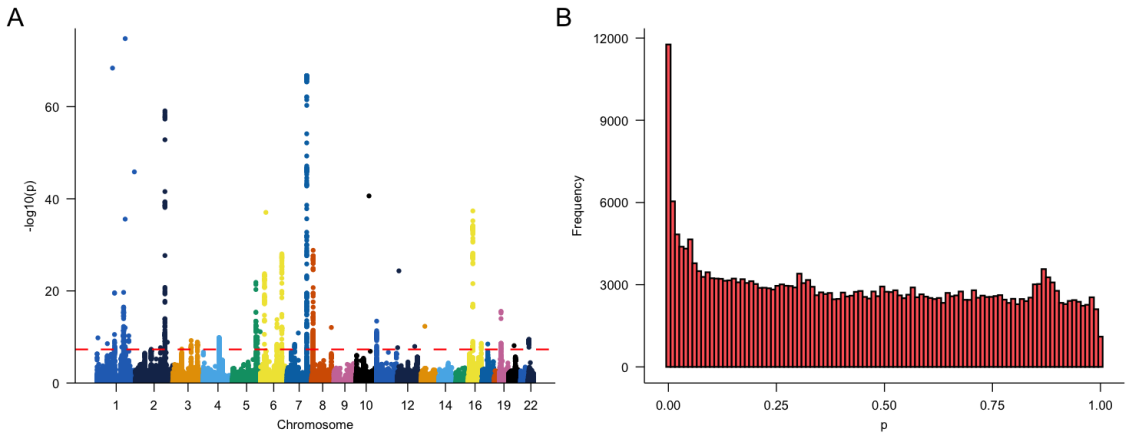


Figure 3.4: Characteristics of the SLE GWAS data. (A) Manhattan plot. Genome-wide significance level (5×10^{-8}) is indicated by the dashed red line. (B) GWAS association *p*-value histogram.

3.6.1 Tissue-level Investigation using GenoSkyline (GS) Annotations

GS utilizes an unsupervised machine learning framework to integrate epigenetic annotations to predict tissue-specific regions that are functionally relevant by assigning each SNP a tissue-specific GS score. GS score, a value between 0 and 1, represents the posterior probability that a SNP is functional given the tissue-specific functional annotation data. The epigenetic functional annotation data utilized in the GS framework were selected from the Roadmap Epigenomics Consortium [45] based on anatomy type and mark availability where relevant samples from at least one of H3k4me1, H3k4me3, H3k36me3, H3k27me3, H3k9me3, H3k27ac, H3k9ac, and DNase I Hypersensitivity are selected to form seven

tissue cluster (Brain, Gastrointestinal/GI, Lung, Heart, Blood, Muscle and Epithelium Tissues) to represent tissue-specific functionality. Using a GS cutoff of 0.5, approximately 22 percent of human genome was predicted to be functional for at least one of the seven tissue-specific region and 1.7 percent was functional for all seven tissue-specific region.

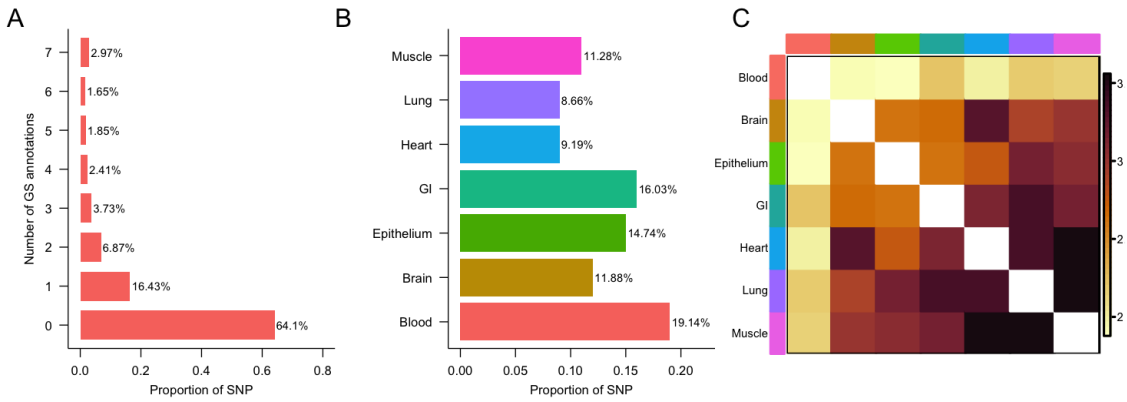


Figure 3.5: Characteristics of $293,976$ SNPs when integrated with seven GenoSkyline (GS) annotations. (A) Number of GS tissues in which SNPs are annotated. (B) Proportion of SNPs that are annotated for each GS tissue type. (C) Overlap of SNPs annotated by seven GS tissue types, calculated using log odds ratio.

We initially investigated the functional potential of all SNPs using seven tissue-specific GS annotations. With a GS score cutoff of 0.5, 35.90% of SNPs were annotated in at least one of the seven tissue types (Figure 3.5A) and the percentage of annotated SNPs ranged from 8.66% for lung tissue to 19.14% for blood tissue (Figure 3.5B). We also measured the overlap in SNPs annotated in different tissue types using log odds ratio (Figure 3.5C). While the highest proportion of SNPs is annotated for blood tissue, SNPs annotated for blood tissue overlap less with other tissue types. On the contrary, SNPs annotated for heart, lung and muscle tissues overlap more with other tissue types. This is consistent with the literature indicating that blood shows the lowest levels of eQTL sharing with other tissue types while muscle and lung tissues show higher levels of eQTL sharing [43, 46].

Next, we applied GPA-Tree to the SLE GWAS and GS annotation data for associa-

tion mapping and characterization of relevant functional annotations. GPA-Tree identified $8,962$ SLE-associated SNPs at the nominal global FDR level of 0.05. Among SLE-associated SNPs, 46.40% were annotated for at least one of the seven GS tissue type (Figure 3.6A), and the percentage of annotated SNPs ranged from 9.89% for lung tissue to 30.22% for blood tissue (Figure 3.6B). We also measured relative enrichment (RE), the ratio of the proportion of SLE-associated SNPs annotated for a specific tissue type, relative to the proportion of non-SLE-associated SNPs annotated for the same tissue type. RE was again highest for the blood tissue with the value of 1.61 (Figure 3.6C). These results are consistent with the dysregulation of blood immune cells that characterizes SLE and other autoimmune diseases like Crohn's disease, ulcerative colitis and rheumatoid arthritis [43].

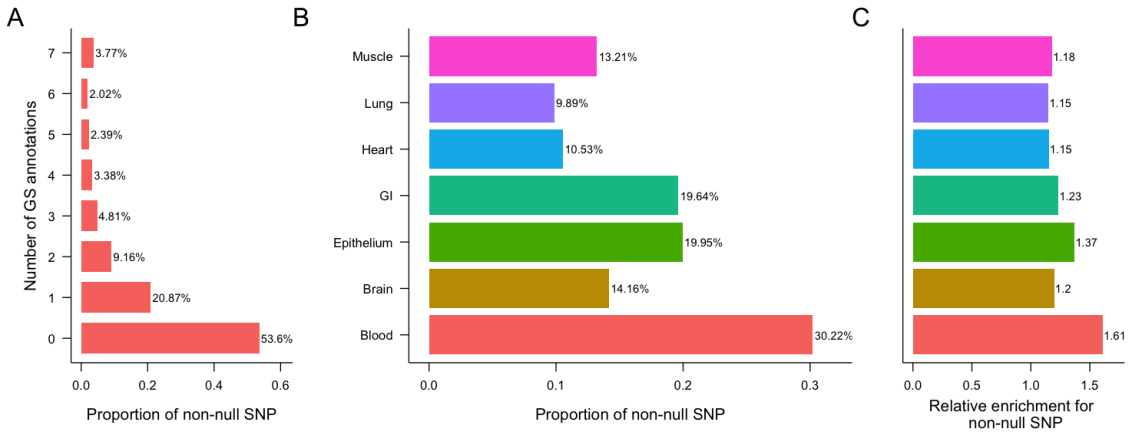


Figure 3.6: Characteristics of the $8,962$ GPA-Tree identified SLE-associated SNPs when integrated with seven GenoSkyline (GS) annotations. (A) Number of GS tissues in which SLE-associated SNPs are annotated. (B) Proportion of SLE-associated SNPs annotated in each GS tissue type. (C) Relative enrichment (RE) of GS tissue types for SLE-associated SNPs. RE is defined as the ratio of the proportion of SLE-associated SNPs that are annotated for a specific GS tissue type, relative to the the proportion of non-SLE-associated SNPs that are annotated for the same GS tissue type.

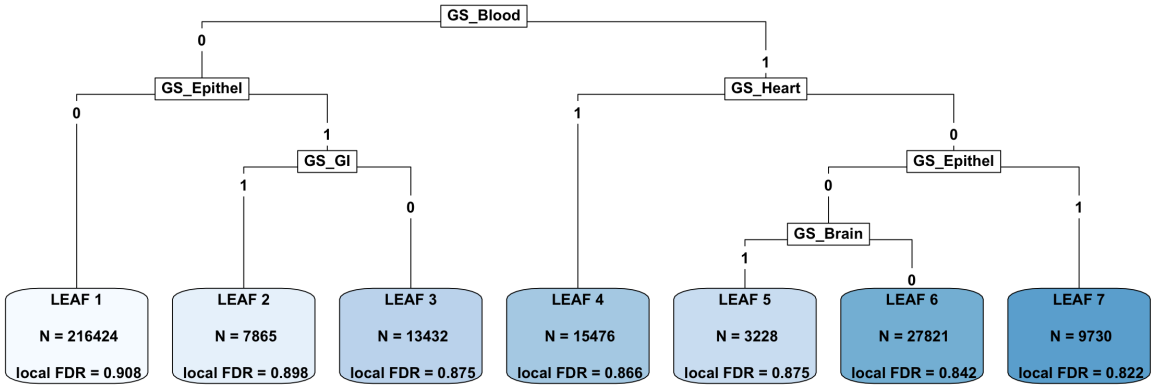


Figure 3.7: Functional annotation tree identified by GPA-Tree approach when seven tissue-level GenoSkyline (GS) annotations are considered. The tree is generated by pruning the GPA-Tree model fit using $cp = 2.5 \times 10^{-4}$. Each leaf (terminal node) in the tree shows the total number of SNPs in the leaf and the mean local FDR for the SNPs in the leaf.

The original GPA-Tree model fit contained blood tissue at the root node and included 28 leaves. For easier interpretation, we used ShinyGPATree app to prune the tree so that it includes 7 leaf nodes (Figure 3.7). We note that although it is occasionally possible to obtain a large functional annotation tree that can be cumbersome to visualize and interpret, the ShinyGPATree app can be utilized to manage such cases as it allows users to investigate different layers of functional annotation trees in an interactive and dynamic manner. For example, the annotation combination for SNPs in leaf 7 can be written as Blood $\cap$!Heart $\cap$ Epithelium, i.e., leaf 7 includes SNPs that are annotated for blood and epithelium tissues but not for heart tissue. The number of SNPs that are located in each leaf node, and the combination of functional annotations that describe SNPs in each leaf node are displayed in Figure 3.7. Further investigation of the GPA-Tree model fitting results showed that, among the $8,962$ SLE-associated SNPs, 578 are concurrently annotated for blood and epithelium tissues while not being annotated for heart tissue as represented in leaf 7; 609 are concurrently annotated for both blood and heart tissues as represented in leaf 4; and 230 are concurrently annotated for epithelium and GI tissues while not being annotated for blood tissue as represented in leaf 2. Blood, epithelium, GI and heart also have the largest RE

(Figure 3.6C). In general, our results are consistent with the literature indicating relevance of blood tissue in SLE, and further add genomic-level support to the relevance of other tissues concurrently with blood [47–50].

3.6.2 Cell-type-level Investigation using GenoSkylinePlus (GSP) Annotations

Utilizing the same statistical framework as GS, GSP functional annotations add another layer of information to SNPs in the form of epigenomic and transcriptomic annotations by providing GSP scores for 127 annotation tracks generated by integrating RNA sequencing and DNA methylation data [44]. Chromatin and DNA methylation data used in GSP were obtained from the Roadmap Epigenomics Project's consolidated reference epigenomic database. Using a GSP score cutoff of 0.5, 3 percent of the human genome, on average, were predicted to be functional across all 127 annotation tracks. Similarly, 26 percent of exonic region, 11 percent of intronic or UTR region, and 6 percent of intergenic regions were predicted to be functional in more than 10 GSP annotation tracks. GSP also identified H3K4me3 and K3K9ac as the most influential predictor of genomic functionality.

Based on the observed relationship between GS annotation for blood tissue and SLE, in the second phase of the real data analysis, we considered 10 blood-related GSP functional annotations. With a GSP score cutoff of 0.5, 25.29% were annotated in at least one of the 10 GSP blood annotations (Figure 3.8A) and the highest enrichment was observed for primary regulatory T cells (12.13%) (Figure 3.8B). The highest overlaps were observed between SNPs annotated with primary memory helper T, effector memory T and $CD8^+$ memory T cells (Figure 3.8C).

Next, we applied GPA-Tree to the SLE GWAS and GSP blood annotations. At the nominal global FDR level of 0.05, GPA-Tree identified $8,993$ SLE-associated SNPs, where

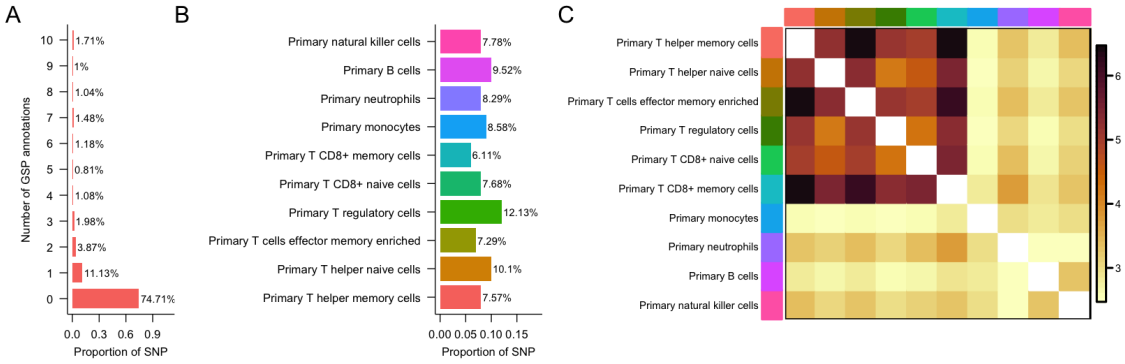


Figure 3.8: Characteristics of the $293,976$ SNPs when integrated with 10 GenoSkyline-Plus (GSP) blood-related annotations. (A) Number of blood-related GSP annotation type in which SNPs are annotated. (B) Proportion of SNPs annotated for each blood-related GSP annotation type. (C) Overlap of SNPs annotated by 10 blood-related GSP cell types, calculated using log odds ratio.

$8,723$ among those overlapped with the SNPs prioritized in the first phase using GS annotations. Among the SLE-associated SNPs prioritized in the second phase, 37.54% were annotated for at least one of the 10 GSP blood annotations (Figure 3.9A). The largest proportion of SLE-associated SNPs was annotated for primary B cells (19.47%), followed by primary regulatory T cells (18.45%) (Figure 3.9B). Primary B cells also showed the highest RE with the value of 2.12 (Figure 3.9C). Since SLE is characterized by the production of autoantibodies, the involvement of B cells, which produce antibodies, is consistent with disease pathology.

The original GPA-Tree model with GSP blood annotations identified primary B cells at the root node and included 172 leaves. Again, to improve interpretability and visualization, we used ShinyGPATree to prune the tree so that it includes 10 leaf nodes (Figure 3.10). In addition to primary B cells, other blood-related GSP functional annotations identified as important included primary memory helper T, regulatory T, neutrophils, natural killer, effector memory T, and $CD8^+$ memory T cells. Among the $8,993$ SLE-associated SNPs, 613 are concurrently annotated for primary B and helper memory T cells as represented in leaf

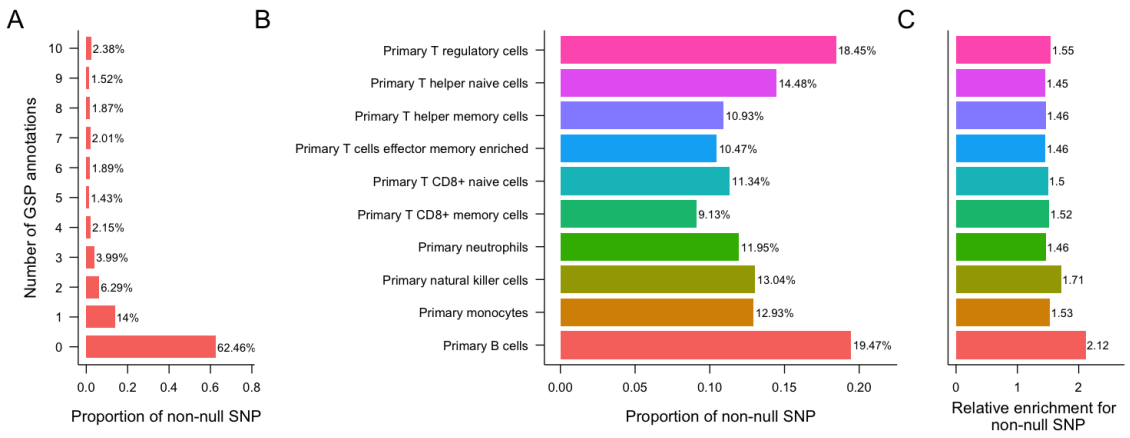


Figure 3.9: Characteristics of the $8,993$ GPA-Tree identified SLE-associated SNPs when integrated with 10 blood-related GSP annotations. (A) Number of blood-related GSP annotations in which SLE-associated SNPs are annotated. (B) Proportion of SLE-associated SNPs annotated in each of the blood-related GSP annotation type. (C) Relative enrichment (RE) of blood-related GSP cell type for SLE-associated SNPs. RE is defined as the ratio of the proportion of SLE-associated SNPs that are annotated for a specific blood-related GSP cell type, relative to the the proportion of non-SLE-associated SNPs that are annotated for the same blood-related GSP cell type.

8; 68 are concurrently annotated for primary B and CD8$^+$ memory T cells while not being annotated for memory helper T cells as represented in leaf 10; and 108 are concurrently annotated for primary regulatory T, neutrophils and effector memory T cells while not being annotated for primary B cells as represented in leaf 4. Overall, these results are consistent with previous literature indicating connections between SLE and B cells, regulatory T cells, neutrophils and CD8$^+$ memory T cells [51–55].

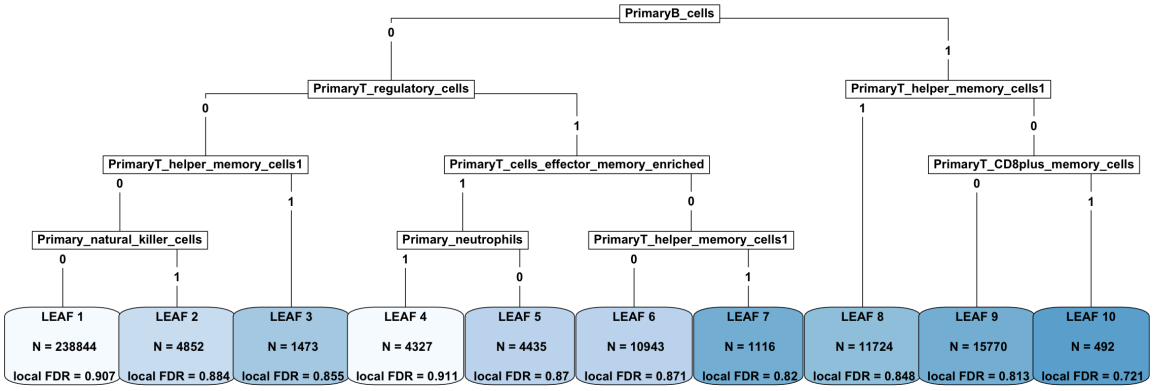


Figure 3.10: Functional annotation tree identified by GPA-Tree approach when 10 blood related cell-type-level GenoSkylinePlus (GSP) annotations are considered. The tree is generated by pruning the GPA-Tree model fit using $cp = 2.5 \times 10^{-4}$. Each leaf (terminal node) in the tree shows the total number of SNPs in the leaf and the mean local FDR for the SNPs in the leaf.

These results also provide several new insights for future investigations. For instance, among the SLE-associated SNPs, 43 SNPs located in the $CLEC16A$ gene and 41 SNPs located in the $IKZF3$ gene are in leaf 8 and concurrently regulate primary B and memory helper T cells; however, an additional 16 SNPs in the $CLEC16A$ gene are in leaf 4 and concurrently regulate primary regulatory T, neutrophils and effector memory T cells while not regulating B cells. These results provide further evidence that multiple independent SNPs in the same gene locus can have different effects on the levels of different immune cell subtypes [56], and can be utilized to investigate a variant's functional role in previously defined associations between SLE, $CLEC16A$ and $IKZF3$ [57–62], among others.

3.7 Conclusions

Several statistical methodologies that efficiently integrate GWAS summary statistics and functional annotation data already exists. However, these methods are not able to identify the combinations of functional annotations that act in unison to influence phenotypic traits. We propose a novel statistical methodology, GPA-Tree, to integrate GWAS summary statis-

tics and functional annotation data to identify trait risk-associated SNPs, and to identify the combinations of functional annotations to potentially explain the mechanisms through with risk-associated SNPs are associated with traits.

GPA-Tree is a hierarchical model, and is implemented by combining an iterative procedure (EM algorithm) and a decision tree algorithm (CART). GPA-Tree assumes that given the latent status (null vs non-null) of the SNPs, their GWAS association p-values come from a Beta-Uniform mixture distribution. Additionally, SNPs are assumed to be conditionally independent given their functional annotation information.

We evaluate the performance of GPA-Tree using simulated data and compare its performance with existing statistical approaches. GPA-Tree showed the higher AUC and statistical power to detect risk-associated SNPs compared to existing approaches. GPA-Tree also successfully identified the true combinations of functional annotations in most cases, facilitating understanding of potential biological mechanisms linking risk-associated SNPs with complex traits. Overall, the ability of GPA-Tree to improve SNP prioritization and attribute functional characteristics to risk-associated SNPs or gene locus can be powerful in facilitating our understanding of genetic susceptibility factors related to complex traits.

4. Specific Aim 2

For Aim 2, our goal is to extend Aim 1 to jointly analyze GWAS summary statistics for multiple complex traits by leveraging pleiotropy and integrating functional annotation information within a unified framework. We implement an iterative procedure (EM algorithm) and a multivariate decision tree algorithm to prioritize SNPs associated with the risk of one or more traits. We also simultaneously identify combinations of functional annotations that can explain the mechanisms through which the risk-associated SNPs influence one or more traits.

4.1 Introduction

Increasing interest in identifying genomic regions associated with different traits has resulted in a substantial increase in the number of reported GWAS studies in the GWAS catalog (https://www.ebi.ac.uk/gwas/). Utilizing the vast number of available GWAS studies to integrate multiple traits has also been useful in demonstrating pleiotropy, a phenomenon in which a SNP is associated with more than one trait. For instance, the human leukocyte antigen (HLA) region is known to be associated with several autoimmune diseases like SLE, multiple sclerosis (MS), Crohn's disease (CD), rheumatoid arthritis (RA), Type-I diabetes (T1D), etc, illustrating the pleiotropic relationship between different autoimmune traits [63, 64]. A joint multivariate analysis of seven autoimmune traits (celiac trait (CEL), inflammatory bowel trait (IBD), which included CD and ulcerative colitis (UC), MS, primary biliary cirrhosis (PBC), RA, SLE and T1D) also identified 67 pleiotropic genes of which 40 were novel [65]. Similar pleiotropic effects are also observed when jointly analyzing different psychiatric disorders [66] as well as other complex traits [67].

Incorporating functional annotation data while leveraging pleiotropy can also improve estimation of heritability of a trait. SNPs that are functionally relevant are more likely to be associated with traits, and can potentially explain larger proportion of variation in the occurrence of a trait attributable to genetic variation. For example, SNPs in the HLA region are more likely to be associated with autoimmune traits [42, 63, 64]. Additionally, SNPs that lie in the blood, brain and liver tissues are significantly associated with autoimmune traits, psychiatric disorders, and lipid-related traits, respectively [44]. Identifying pleiotropic SNPs that are related to multiple traits via some functional process can improve our understanding of the mechanisms of the pleiotropic relationship between traits. In addition to improving our understanding of the biological processes that are shared by multiple traits and pleiotropic SNPs, integrating GWAS summary statistics and functional annotations for multiple traits improves statistical power to detect SNPs with weak or moderate effect sizes. This means that SNPs that are not detected when traits are analyzed separately are potentially detectable when analyzed jointly. For instance, using GPA, Chung et al. showed that joint analysis of BPD and SCZ identified substantially more risk-associated SNPs with the two traits compared to when the traits were separately analyzed [10]. Similar results were observed for low-density lipoprotein (LDL) and total cholesterol (TC) by Ming et al. in LPM [11].

As described in Sections 2.2 and 2.3, several Bayesian and mixture models that integrate GWAS association statistics for multiple traits and functional annotation data exist. Most of these methods prioritize SNPs associated with individual and multiple traits while providing some metric to identify relevant functional annotations. However, they do not provide a method for identifying interactions between functional annotations, a limitation we propose to address with Specific Aim 2. Our goal for Aim 2 is two-fold; we want to (1) prioritize SNPs that are associated with one or more traits by leveraging pleiotropy, and (2) identify the combinations of functional annotations that are related to one or more

trait risk-associated SNPs. To achieve these goals, we propose to use a hierarchical model to extend GPA-Tree. Our method allows for integration of GWAS summary statistics for multiple traits and functional annotation data within a unified framework. This extension of GPA-Tree combines an EM algorithm with a multivariate decision tree algorithm. Utilizing the EM algorithm allows for iterative adjustment of the parameter estimates in the hierarchical model while gradually leading us towards the most appropriate tree structure to identify the combinations of functional annotations that are related to one or more trait risk-associated SNPs.

This chapter is structured as follows. In Section 4.2, we present background information related to different multivariate decision tree approaches. In Section 4.3, we introduce the Multi-GPA-Tree method for prioritizing one or more trait risk-associated SNPs, and identifying combinations of functional annotations that are associated with one or more trait risk-associated SNPs. In Sections 4.4 and 4.5, we describe our simulation settings and simulation results, respectively. In section 4.6, we describe the results of the application of our method to real data. Finally, in Section 4.7 we discuss the implications of using our method, including its limitations and some direction for extending the work.

4.2 Background

Integrating GWAS summary statistics for multiple traits can induce some correlation structure in the multivariate outcome. For instance, the GWAS association p-values for a SNP with two or more traits may be correlated. Ignoring such correlation can lead to inaccurate parameter estimation and loss of statistical power. We can mitigate such problems by utilizing methodologies that account for the covariance between multiple response variables.

Several methodologies have been proposed for developing decision and classification trees while accounting for covariance structure between multiple response variables. These

methodologies focus on response variables that are all binary/multiclass [68, 69], ordinal [69–71], continuous [70] or a combination of binary, multiclass, ordinal and continuous [70, 72, 73]. The first multivariate decision tree methodology, developed by Segal in 1992, was implemented by modifying a regression tree algorithm to allow the split function to accommodate multiple ordinal or continuous longitudinal response variables [70]. Segal considered two types of split functions: one focusing on the mean structure where covariance is treated as a nuisance parameter, and another focusing primarily on the covariance structure. Given T responses, the split function for evaluating any split s at a node g into g_L (left daughter node) and g_R (right daughter node) is given by $\phi_m(s,g) = SS(g) - SS(g_L) - SS(g_R)$, where $SS(g) = \sum_{i \in g} (y_i - \mu(g))' V(\theta, g)^{-1} (y_i - \mu(g))$, $y_i' = (y_{i1}, y_{i2}, ..., y_{iT})$ are the T responses for the i^{th} individual, $V(\theta, g)$ is the model covariance matrix of the responses for node g depending on parameters θ, and $\mu(g)$ is the $T \times 1$ vector of response means for individuals within a given node g. To ensure that ϕ_m is positive, the covariance estimates of the parent node g are used to determine the covariance for each candidate split such that $V(\theta, g) = V(\theta_L, g_L) = V(\theta_R, g_R)$. Similar to comparing within node measures of loss, a function that assesses how closely the sample covariance matrix conforms to the hypothesized covariance matrix is considered by using a likelihood ratio test type statistic. The value of the likelihood ratio test for equality of covariance matrices, maximized over all candidate splits, is used to split the data into subgroups that are most distinct in terms of covariance structure.

A tree-based method for analyzing multiple binary response variables proposed by Zhang [68] selects splits in the tree to ensure that there is homogeneity in the distribution of the response class (y_k) between the sub-nodes. The parametric implementation involves using exponential families of distributions where the joint distribution of $\mathbf{y}$ depends on the linear terms of the individual components in $\mathbf{y}$ and the sum of the second-order product of the components of $\mathbf{y}$ only, and the generalized node entropy or node homogeneity is defined

as the maximum of the log-likelihood of the used distribution such that a split is selected at the point that maximizes the weighted node homogeneity [68]. Zhang and Yu extend the tree-based method to analyze multivariate ordinal response, where each ordinal response with K classes are transformed into $K-1$ binary indicator variables used to develop the trees and the covariance matrix is computed at each node to determine the split [71]. As the number of response variables increases considerably when using this method, its implementation can be computationally burdensome and the results can be difficult to interpret.

Another variation of the approach by Segal [70] was proposed by Larsen and Speckman [74]. They developed the multivariate regression tree (MRT) methodology to evaluate the relationship between the different response variables and predictors. They proposed using the sample covariance matrix for the full data as the covariance matrix V, and used Mahalanobis distance to measure node impurity. In the MRT setting, multiple response variables are explained or can be predicted by explanatory variables. A second method for MRT was proposed by De'Ath in 2002 [75] that extends the univariate regression tree by replacing the univariate response by a multivariate response, and redefines the impurity of a node as the sum of squares around the multivariate means, i.e., $impurity = \sum_{i,j}(x_{ij} - \bar{x}_j)^2$, where i represents the i^{th} subject and j represents the j^{th} response. Similar to univariate regression trees, MRT can be grown and pruned based on the impurity of a node, the rule for splitting nodes and the prediction error for a new observation [26], where the primary objective is to minimize the total impurity for observations in any nodes. MRTs are deemed useful when interactions between predictor variables and nonlinear relationships between response and predictor variables are prevalent. The MRT methodology by De'Ath can be easily implemented for continuous response variables using the $mvpart$ package in R. Yu and Lambert explored two other ways of implementing MRT to multiple response variables [76]. The first approach is based on the assumption that when response variables are smooth, they can be treated as a curve and are approximated by using a linear combination

of basis functions (e.g., B-splines or natural splines), where the derived coefficients of the linear combinations for each individual can be used as responses for a multivariate tree. When constructing the MRT, prediction error for each response in the node is measured using the Mahalanobis distance and a split that minimizes this distance is retained. In the second approach, rather than reducing the dimension of the response variables by treating it as a curve, the dimension of the response variables is reduced by using principal component analysis and the first several principal component scores are used as response variables when fitting the MRT. While both these approaches are easy to implement and can remove correlated response variables, the results are not easy to interpret. Loss of information can also occur when coefficients of linear combinations for splines or principal components, rather than original response variables, are used.

An alternative method that utilizes the idea of residual sign vector was proposed by Loh and Zheng in 2013 [69]. This method also incorporates the unbiased variable selection feature attributable to GUIDE, an algorithm for univariate regression tree construction [77]. At each node, the sample mean vector for the response variables are calculated. A sign vector is then defined such that the observed responses that are less than or equal to the corresponding mean response are given a negative sign while observed responses that are greater than the corresponding mean response are given a positive sign. A chi-squared test for the difference between the two groups is performed. Selecting the interval or split point for a continuous predictor variable can be difficult as it has to be user-defined. Although this method lacks a predetermined algorithm to find the split point for continuous predictor variables, its performance accuracy seems to be on par with $mvpart$, the MRT method by De'Ath [75].

Many of the multivariate tree-based methods presented here are not easy to use due to lack of available statistical software. In the implementation of our novel method for Aim 2, we will combine the MRT methodology proposed by De'Ath [75] and EM algorithm within

a unified framework. This MRT method is easy to implement using the $mvpart$ package in R and circumvents the problem related to bias in variable selection towards predictor variables with the maximum number of splitting option as all predictor variables used in our data application are binary, making the issue of variable selection bias irrelevant. Additionally, interpretation of the MRT is made easier as the R function $mvpart$ can identify the response variables that most strongly influence the splits of the multivariate tree. The function provides tree biplots to represent response group means, and also identifies response variables that best characterize the predicted groups. MRT is a powerful tool for exploration, and can be useful in identifying the combination of functional annotations that are related to multiple response variables.

4.3 Multi-GPA-Tree Method

4.3.1 Model

Let $\mathbf{Y}_{M\times D}$ be a matrix of genotype-trait association *p*-values for $i = 1, 2, \cdots, M$ SNPs and $d = 1, 2, \cdots, D$ traits, where Y_{id} denotes the *p*-value for the association of the i^{th} SNP with the d^{th} trait:

$$\mathbf{Y} = (\mathbf{Y}_{.1}, \ldots, \mathbf{Y}_{.D}) = \begin{pmatrix} y_{11} & \cdots & y_{1D} \\ \vdots & \ddots & \vdots \\ y_{M1} & \cdots & y_{MD} \end{pmatrix}_{M\times D} .$$

We also assume K binary annotations ($\mathbf{A}$) for each SNP:

$$\mathbf{A} = (\mathbf{A}_{.1}, \ldots, \mathbf{A}_{.K}) = \begin{pmatrix} a_{11} & \ldots & a_{1K} \\ \vdots & \ddots & \vdots \\ a_{M1} & \ldots & a_{MK} \end{pmatrix}_{M\times K} , \text{ where}$$

$$a_{ik} = \begin{cases} 0, & \text{if } i^{th} \text{ SNP is not annotated in the } k^{th} \text{ annotation} \\ 1, & \text{if } i^{th} \text{ SNP is annotated in the } k^{th} \text{ annotation} \end{cases}$$

To improve the power to identify risk-associated SNPs for one or more traits, we integrate GWAS association *p*-values for D traits ($\mathbf{Y}$) with functional annotations data ($\mathbf{A}$). We characterize the effect of functional annotations in modeling the relationship between GWAS traits and SNPs by defining a matrix $\mathbf{Z}_{M\times 2^D} \in \{0,1\}$ of latent binary variables where $\mathbf{Z}_i$ is a vector of length 2^D and indicates whether the i^{th} SNP is null or non-null for the D traits. We present the model for the case of two GWAS traits ($D = 2$) to simplify notations.

Let $Y \in \mathbb{R}^{M\times 2}$ be the matrix of GWAS association *p*-values for two traits where Y_{i1} and Y_{i2} are the *p*-values for the association of the i^{th} SNP with traits 1 and 2, respectively. We define the latent binary vector $\mathbf{Z}_i = \{Z_{i00}, Z_{i10}, Z_{i01}, Z_{i11}\}$ for the i^{th} SNP, where $Z_{i00} = 1$ indicates the i^{th} SNP is null for both traits, $Z_{i10} = 1$ indicates the i^{th} SNP is non-null for trait 1 and null for trait 2, $Z_{i01} = 1$ indicates the i^{th} SNP is null for trait 1 and non-null for trait 2, and $Z_{i11} = 1$ indicates the i^{th} SNP is non-null for both traits. We assume that a SNP can only be in one of the four states such that $\sum_{l\in\{00,10,01,11\}} Z_{il} = 1$. The densities for SNPs in the null and non-null groups for both traits are assumed to come from $U[0,1]$ and $Beta(\alpha_d, 1)$ distributions, where $0 < \alpha_d < 1$ and $d = 1, 2$, as proposed by Chung and colleagues [10]. The GWAS association p-value density distribution for SNPs in the different groups are defined as shown below:

$$\begin{aligned}
&(Y_{i1}|Z_{i00} = 1) \sim U[0,1] && (Y_{i2}|Z_{i00} = 1) \sim U[0,1] \\
&(Y_{i1}|Z_{i10} = 1) \sim Beta(\alpha_1, 1) && (Y_{i2}|Z_{i10} = 1) \sim U[0,1] \\
&(Y_{i1}|Z_{i01} = 1) \sim U[0,1] && (Y_{i2}|Z_{i01} = 1) \sim Beta(\alpha_2, 1) \\
&(Y_{i1}|Z_{i11} = 1) \sim Beta(\alpha_1, 1) && (Y_{i2}|Z_{i11} = 1) \sim Beta(\alpha_2, 1),
\end{aligned}$$

where $0 < \alpha_1, \alpha_2 < 1$. We integrate functional annotation data $\mathbf{A}$ with the GWAS summary statistics data $\mathbf{Y}$ by defining a function f that is a combination of functional annotations $\mathbf{A}$ and relating it to the multivariate expectation of latent $\mathbf{Z}$ as given in Equation (4.1).

$$P(Z_{il} = 1; a_{i1}, \ldots, a_{iK}) = f(a_{i1}, \ldots, a_{iK}), \text{where } l \in \{00, 10, 01, 11\} \tag{4.1}$$

For notational convenience we denote $P(Z_{il} = 1; a_{i1}, \ldots, a_{iK})$ as $\pi_{il}, l \in \{00, 10, 01, 11\}$, where $\boldsymbol{\pi}_{i00}$ is the prior probability that the i^{th} SNP is null for both traits, $\boldsymbol{\pi}_{i10}$ is the prior probability that the i^{th} SNP is non-null for trait 1 and null for trait 2, $\boldsymbol{\pi}_{i01}$ is the prior probability that the i^{th} SNP is null for trait 1 and non-null for trait 2, and $\boldsymbol{\pi}_{i11}$ is the prior probability that the i^{th} SNP is non-null for both traits. The flowchart in Fig 4.1 provides a complete graphical representation for these data.

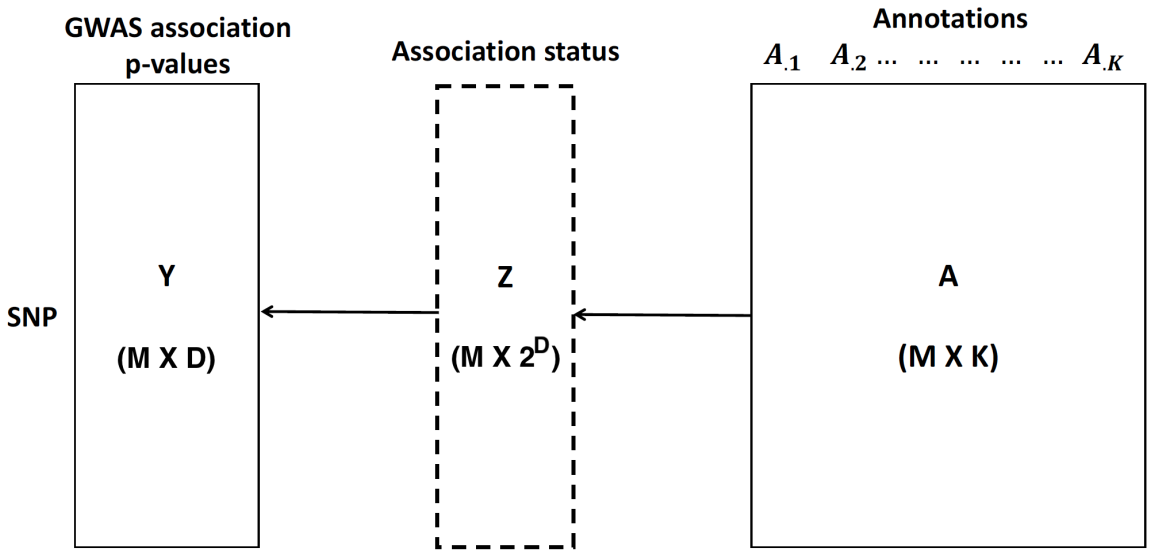


Figure 4.1: Association framework that links the GWAS association p-values for D traits ($\mathbf{Y}$), the association status for the D trait as given by the latent variable ($\mathbf{Z}$) and annotation data ($\mathbf{A}$).

Let $\boldsymbol{\theta} = (\alpha_1, \alpha_2)$. Assuming that the SNPs are independent, we can write the joint

distribution of the observed data $Pr(\mathbf{Y}, \mathbf{A})$ as:

$$\begin{aligned} Pr(\mathbf{Y}, \mathbf{A}) &= \prod_{i=1}^{M} \left[\sum_{l \in \{00,10,01,11\}} P(Z_{il} = 1) P(Y_{i1}, Y_{i2} | Z_{il} = 1) \right] \\ &= \prod_{i=1}^{M} \left[\sum_{l \in \{00,10,01,11\}} \pi_{il} \, P(Y_{i1}, Y_{i2} | Z_{il} = 1) \right] \end{aligned}$$

The 'incomplete' data log-likelihood is written as:

$$\ell_{IC} = \sum_{i=1}^{M} log \left[\sum_{l \in \{00,10,01,11\}} \pi_{il} \, P(Y_{i1}, Y_{i2} | Z_{il} = 1) \right]$$

We can write the 'complete' data likelihood as:

$$L_C = \prod_{i=1}^{M} \prod_{l \in \{00,10,01,11\}} \left[\pi_{il} \, P(Y_{i1}, Y_{i2} | Z_{il} = 1) \right]^{Z_{il}}$$

Similarly, the 'complete' data log-likelihood can be written as:

$$\ell_C = \sum_{i=1}^{M} \sum_{l \in \{00,10,01,11\}} Z_{il} \, log \left[\pi_{il} \, P(Y_{i1}, Y_{i2} | Z_{il} = 1) \right]$$

4.3.2 Algorithm

Given the approach described in Section 4.3.1, we implemented parameter estimation using an EM algorithm. The function f in Equation (4.1) is estimated by using the MRT method by De'Ath. This method allows to identify combinations of functional annotations related to one or more trait risk-associated SNPs. To improve stability, we employed a two-stage approach for parameter estimation. Specifically, in Stage 1, we first estimate the parameters α_1 and α_2 without identifying a combination of functional annotations. Then, in Stage 2, we identify key combinations of functional annotations ($f(\mathbf{A})$) while the parameters α_1 and α_2 are kept fixed as the value obtained in Stage 1. We illustrate more detailed calculation

steps below.

Stage 1:

In Stage 1, we initialize $\alpha_d^{(0)} = 0.1$, $d = 1, 2$ and $\pi_{il}^{(0)} = \frac{1}{2^D}$, $D = 2$ (the number of traits). In the t^{th} iteration of the E-step, define $Z_{il}^{(t)},\ l \in \{00, 10, 01, 11\}$ for the i^{th} SNP as:

$$\begin{aligned} \mathbf{E-step}: z_{il}^{(t)} &= P(Z_{il} = 1|\mathbf{Y}, \mathbf{A};\ \boldsymbol{\theta}^{(t-1)}) \\ &= \frac{\pi_{il}^{(t-1)}\ P(Y_{i1}, Y_{i2}|Z_{il}=1; \boldsymbol{\theta}^{(t-1)})}{\sum\limits_{l' \in \{00,10,01,11\}} \pi_{il'}^{(t-1)}\ P(Y_{i1}, Y_{i2}|Z_{il'}=1; \boldsymbol{\theta}^{(t-1)})} \end{aligned} \tag{4.2}$$

In the t^{th} iteration of the M-step, $\boldsymbol{\pi}_{i.}$, α_1 and α_2 are updated as:

$\mathbf{M-step}$: Fit a multivariate linear regression model as

$$\mathbf{Z}_{i.}^{(t)} = \beta_0^{(t)} + \beta_1^{(t)} a_{i1} + \cdots + \beta_K^{(t)} a_{iK} + \epsilon_i{}^{(t)}$$

Update $\boldsymbol{\pi}_{i.}$as the predicted value from the multivariate linear regression model.

Update $\alpha_1^{(t)} = -\frac{\sum\limits_{i=1}^{M}(z_{i10}^{(t)} + z_{i11}^{(t)})}{\sum\limits_{i=1}^{M}(z_{i10}^{(t)} + z_{i11}^{(t)})(logY_{i1})}$ and $\alpha_2^{(t)} = -\frac{\sum\limits_{i=1}^{M}(z_{i01}^{(t)} + z_{i11}^{(t)})}{\sum\limits_{i=1}^{M}(z_{i01}^{(t)} + z_{i11}^{(t)})(logY_{i2})}$

where $\beta_k^{(t)}, k = 0, \cdots, K$ are the regression coefficients and $\epsilon_i^{(t)}$ is the error term. The E and M steps are repeated until the incomplete log-likelihood, α_1 and α_2 estimates converge. Then, α_1, α_2 and $\boldsymbol{\pi}_{i.}$ estimated in this stage are used to fix α_1, α_2 and initialize $\boldsymbol{\pi}_{i.}$, respectively, in Stage 2.

Stage 2:

In this stage, we implement another EM algorithm employing the MRT algorithm by De'Ath ($mvpart$ [75]), which allows to identify union, intersection, and complement relationships between functional annotations in estimating $\boldsymbol{\pi}_{i.}$.

In the t^{th} iteration of the E-step, define $Z_{il}^{(t)},\ l \in \{00, 10, 01, 11\}$ for the i^{th} SNP as shown in Equation 4.2, except α_1 and α_2 are fixed as $\hat{\alpha_1}$ and $\hat{\alpha_2}$, which are the final estimates of α_1 and α_2 obtained from Stage 1.

$$\mathbf{E-step}: \text{Define } Z_{il}^{(t)},\ l \in \{00, 10, 01, 11\} \text{ as in Equation 4.2, except } \alpha_1 \text{ and } \alpha_2 \text{ are fixed as } \hat{\alpha_1} \text{ and } \hat{\alpha_2}, \text{ the final estimates of } \alpha_1 \text{ and } \alpha_2 \text{ from Stage 1.}$$

In the t^{th} iteration of the M-step, $\boldsymbol{\pi}_{i.}$ is updated as:

$$\begin{aligned} \mathbf{M-step}: &\text{ Fit a MRT model as } \mathbf{Z}_{i.}{}^{(t)} = f^{(t)}(a_{i1}, \cdots, a_{iK}) + \epsilon_i{}^{(t)} \\ &\text{Update } \boldsymbol{\pi}_{i.}{}^{(t)} \text{ as the predicted values from the MRT model,} \end{aligned} \tag{4.3}$$

where ϵ_i is the error term. In the M-step, the complexity parameter (cp) is the key tuning parameter and defined as the minimum improvement that is required at each node of the tree. Specifically, in the MRT model, the largest possible tree (i.e., a full-sized tree) is first constructed and then pruned using cp. The pruned tree structure identified by the MRT model upon convergence of the EM algorithm (Equation (4.3)) is used as f in Equation (4.1). This approach allows for the construction of the accurate yet interpretable MRT that can explain relationships between functional annotations, and risk-associated SNPs for one or more traits. The E and M steps are repeated until the incomplete log-likelihood converges.

We note that unlike the standard EM algorithm, the incomplete log-likelihood in Stage 2 is not guaranteed to be monotonically increasing. Therefore, we implement Stage 2 as a generalized EM algorithm by retaining only the iterations in which the incomplete log-likelihood increases compared to the previous iteration.

4.3.3 Prioritization of Risk-associated SNPs for One or More Traits and Identification of Relevant Combinations of Functional Annotations

Once the parameters are estimated as described in Section 4.3.2, we can prioritize risk SNPs that are associated with one or more traits using local false discovery rate or fdr. For marginal associations with a specific trait, we can define $fdr_d, d = 1, 2$ as the marginal posterior probability that the i^{th} SNP belongs to the null group for the specific trait given its GWAS p-values for all traits and functional annotation information. For joint associations between traits, we can define $fdr_{1,2}$ as the joint posterior probability that the i^{th} SNP belongs to the null group for the traits given its GWAS p-values for all traits and functional annotation information.

$$fdr_1(\mathbf{Y}_{i.}, \mathbf{A}_{i.}) = P(Z_{i00} + Z_{i01} = 1 | \mathbf{Y}_{i.}, \mathbf{A}_{i.}, \hat{\boldsymbol{\theta}}) = \frac{P(Y_{i1}, Y_{i2}, Z_{i00}+Z_{i01}=1; \hat{\boldsymbol{\theta}})}{P(Y_{i1}, Y_{i2}; \hat{\boldsymbol{\theta}})},$$
$$fdr_2(\mathbf{Y}_{i.}, \mathbf{A}_{i.}) = P(Z_{i00} + Z_{i10} = 1 | \mathbf{Y}_{i.}, \mathbf{A}_{i.}, \hat{\boldsymbol{\theta}}) = \frac{P(Y_{i1}, Y_{i2}, Z_{i00}+Z_{i10}=1; \hat{\boldsymbol{\theta}})}{P(Y_{i1}, Y_{i2}; \hat{\boldsymbol{\theta}})},$$
$$fdr_{1,2}(\mathbf{Y}_{i.}, \mathbf{A}_{i.}) = P(Z_{i00} + Z_{i10} + Z_{i01} = 1 | \mathbf{Y}_{i.}, \mathbf{A}_{i.}) = \frac{P(Y_{i1}, Y_{i2}, Z_{i00}+Z_{i10}+Z_{i01}=1; \hat{\boldsymbol{\theta}})}{P(Y_{i1}, Y_{i2}; \hat{\boldsymbol{\theta}})},$$

where

$$P(Y_{i1}, Y_{i2}; \hat{\boldsymbol{\theta}}) = \sum_{l \in \{00,10,01,11\}} \hat{\pi}_{il} \, P(Y_{i1}, Y_{i2} | Z_{il}, \mathbf{A}_{i.}; \hat{\boldsymbol{\theta}}),$$
$$P(Y_{i1}, Y_{i2}, Z_{i00} + Z_{i01} = 1; \hat{\boldsymbol{\theta}}) = \sum_{l \in \{00,01\}} \hat{\pi}_{il} \, P(Y_{i1}, Y_{i2} | Z_{il}, \mathbf{A}_{i.}; \hat{\boldsymbol{\theta}}),$$
$$P(Y_{i1}, Y_{i2}, Z_{i00} + Z_{i10} = 1; \hat{\boldsymbol{\theta}}) = \sum_{l \in \{00,10\}} \hat{\pi}_{il} \, P(Y_{i1}, Y_{i2} | Z_{il}, \mathbf{A}_{i.}; \hat{\boldsymbol{\theta}}),$$
$$P(Y_{i1}, Y_{i2}, Z_{i00} + Z_{i10} + Z_{i01} = 1; \hat{\boldsymbol{\theta}}) = \sum_{l \in \{00,10,01\}} \hat{\pi}_{il} \, P(Y_{i1}, Y_{i2} | Z_{il}, \mathbf{A}_{i.}; \hat{\boldsymbol{\theta}}),$$

We can perform association mapping using fdr control. In this approach, SNPs with $fdr(\mathbf{Y}_{i.}, \mathbf{A}_{i.}) \leq \tau$, where τ is the predetermined fdr control level, are mapped to be associated with the trait. Finally, relevant combinations of functional annotations are inferred

based on the combination of functional annotations selected by the MRT model upon convergence of the EM algorithm.

4.4 Simulation Study Design

We conducted a simulation study to evaluate the performance of the proposed Multi-GPA-Tree approach as depicted in Figure 4.2. For all the simulation data, the number of SNPs was set to $M = 10,000$, the number of annotations was set to $K = 15$, SNPs that are marginally associated with the first trait (P_1) were assumed to be characterized with the combinations of functional annotations defined by $L_1 = A_1 \cap A_2$, SNPs that are marginally associated with the second trait (P_2) were assumed to be characterized with the combinations of functional annotations defined by $L_2 = A_3 \cap A_4$, SNPs that are jointly associated with traits P_1 and P_2 were assumed to be characterized with the combinations of functional annotations defined by $L_3 = A_5 \cap A_6$, and all the remaining functional annotations ($A_k, k = 7, \ldots, 15$) were considered to be noise annotations. The percentage of annotated SNPs (u) for annotations $A_1 - A_6$ was set to $5\%, 10\%, 15\%$ and 20%, and the percentage of overlap between the true combinations of functional annotations (v) was set to 50% . For noise annotations $A_7 - A_{15}$, approximately 20% of SNPs were annotated by first generating the proportion of annotated SNPs from $Unif[0.1, 0.3]$ and then randomly setting this proportion of SNPs to one. The SNPs that satisfy the functional annotation combination L_1 or L_3 were assumed to be risk-associated SNPs for trait P_1 and their *p*-values were simulated from $Beta(\alpha_1,\ 1)$ with $\alpha_1 = 0.4$. Similarly, the SNPs that satisfy the functional annotation combination L_2 or L_3 were assumed to be risk-associated SNPs for trait P_2 and their *p*-values were simulated from $Beta(\alpha_2,\ 1)$ with $\alpha_2 = 0.4$. The SNPs that do not satisfy the required condition for association with P_1 or P_2 were assumed to be non-risk SNPs and their *p*-values were simulated from $U[0,\ 1]$. Note that here the signal-to-noise ratio is

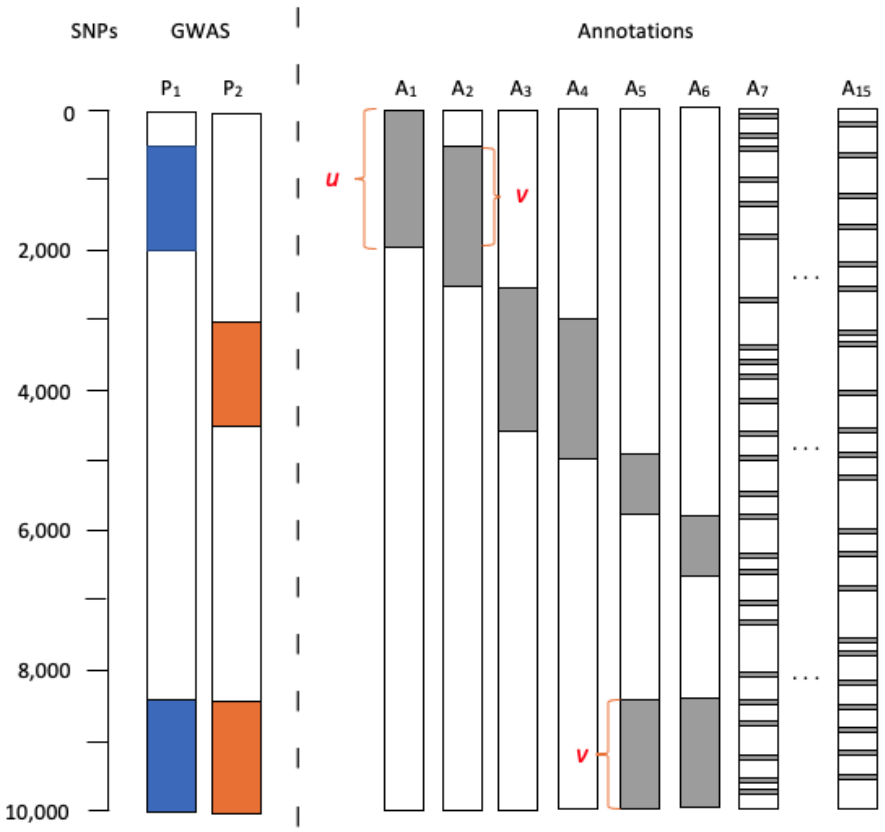


Figure 4.2: Simulation setting with $K = 15$ functional annotations ($A_1 - A_{15}$). Annotations $A_1 - A_2$ are assumed to be related to SNPs marginally associated with trait P_1, annotations $A_3 - A_4$ are assumed to be related to SNPs marginally associated with trait P_2, and annotations $A_5 - A_6$ are assumed to be related to SNPs jointly associated with both traits P_1 and P_2. For each of $A_1 - A_6$, $u\%$ SNPs are assumed to be annotated and $v = 50\%$ of the annotated SNPs are assumed to be shared between A_1 and A_2, A_3 and A_4, and A_5 and A_6. The remaining functional annotations ($A_7 - A_{15}$) are assumed to be unrelated to risk-associated SNPs and approximately 20% of the SNPs are annotated at random.

affected by u.

4.5 Simulation Study Results

For each combination of the simulation parameters defined above, we simulated 20 datasets and compared the performance of Multi-GPA-Tree with LPM [11]. For marginal and joint associations, the metrics for comparing the methods include (1) area under the curve (AUC)

where the curve was created by plotting the true positive rate (sensitivity) against the false positive rate (1-specificity) to detect one or more trait risk-associated SNPs when fdr is controlled at various nominal levels, and (2) statistical power to identify marginal and joint risk-associated SNPs when fdr is controlled at the nominal fdr level of 0.05. We also compared the estimation accuracy for α_d parameters in the $Beta(\alpha_d,\ 1), d = 1, 2$ distribution for the *p*-values of risk-associated groups for traits P_1 and P_2. Lastly, for Multi-GPA-Tree, we examined the accuracy of detecting the correct functional annotation tree based on (1) the proportion of simulation data for which all relevant functional annotations in L_1, L_2 and L_3, i.e, annotations $A_1 - A_6$, were identified simultaneously; (2) the average proportion of true functional annotations ($A_1 - A_6$) among the functional annotations identified by multi-GPA-Tree; and (3) the average proportion of noise annotations ($A_7 - A_{15}$) among all annotations identified by Multi-GPA-Tree. Here we especially investigate how the percentage of SNPs annotated in $A_1 - A_6$ (u) impact Multi-GPA-Tree's ability to separate relevant functional annotations from noise annotations for one or more trait risk-associated SNPs when the overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) is set at 50%.

4.5.1 Marginal Association Results

AUC: Figure 4.3 A and B show the AUC comparison between Multi-GPA-Tree and LPM for traits P_1 and P_2, respectively. For all u, Multi-GPA-Tree showed consistently higher AUC relative to LPM to detect SNPs that are marginally associated with both traits P_1 and P_2. The performance of LPM improved as signal-to-noise ratio increases (i.e., as u increases), demonstrating performance closer to Multi-GPA-Tree.

Statistical power: Figures 4.4 A and B compare the statistical power to detect true risk-associated SNPs when fdr is controlled at the nominal level of 0.05 for

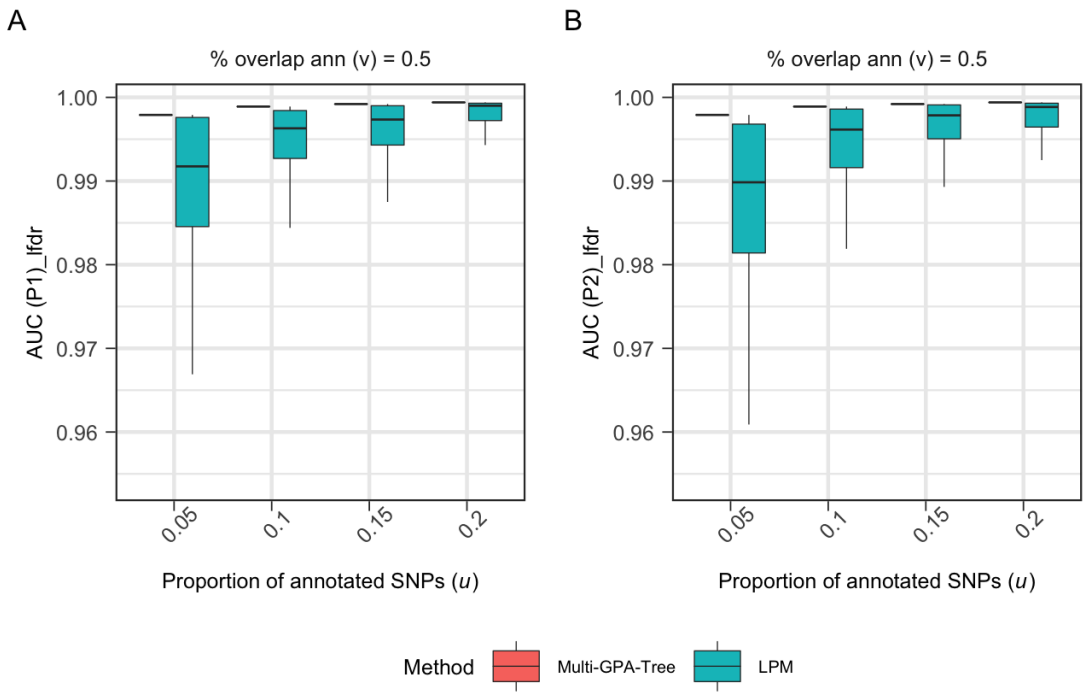


Figure 4.3: Comparison of AUC between Multi-GPA-Tree and LPM for traits (A) P_1, and (B) P_2, respectively. The results are presented for different proportions of SNPs annotated in $A_1 - A_6$ (u; x-axis) when the proportion of overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) equals 50%. $M = 10,000$, $K = 15$, $\alpha_1 = \alpha_2 = 0.4$ in $Beta(\alpha_d, 1), d = 1, 2$. Results are summarized from 100 replications. Outliers are not displayed in the plots.

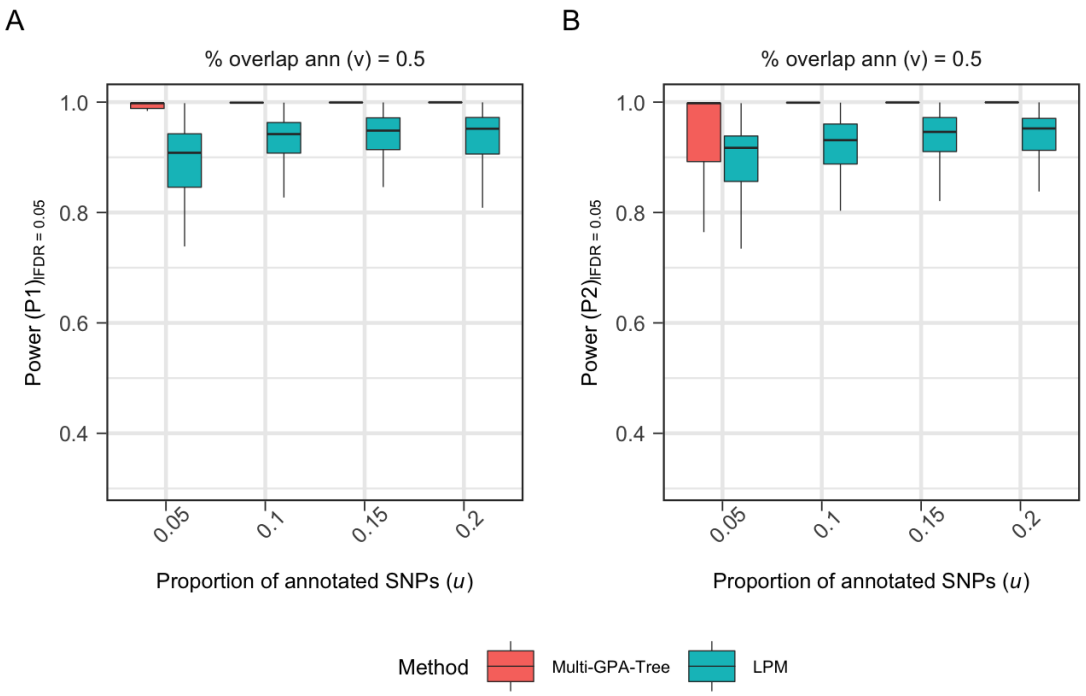


Figure 4.4: Comparison of statistical power between Multi-GPA-Tree and LPM to detect true risk-associated SNPs when fdr is controlled at the nominal level of 0.05 for traits (A) P_1, and (B) P_2, respectively . The results are presented for different proportions of SNPs annotated in $A_1 - A_6$ (u; x-axis) when the proportion of overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) equals 50%. $M = 10,000$, $K = 15$, $\alpha_1 = \alpha_2 = 0.4$ in $Beta(\alpha_d, 1), d = 1, 2$. Results are summarized from 100 replications. Outliers are not displayed the plots.

traits P_1 and P_2, respectively, using Multi-GPA-Tree and LPM. Compared to LPM, Multi-GPA-Tree showed relatively higher statistical power and lower variability in its estimates for almost all u when $v = 50\%$. The performance of LPM improved as signal-to-noise ratio increased (i.e., as u increased), demonstrating statistical power closer to Multi-GPA-Tree.

Predicted fdr control: Figure 4.5 A and B compare the predicted fdr when true fdr is controlled at the nominal level of 0.05 for traits P_1 and P_2, respectively, using Multi-GPA-Tree and LPM. For both Multi-GPA-Tree and LPM, the predicted fdr is

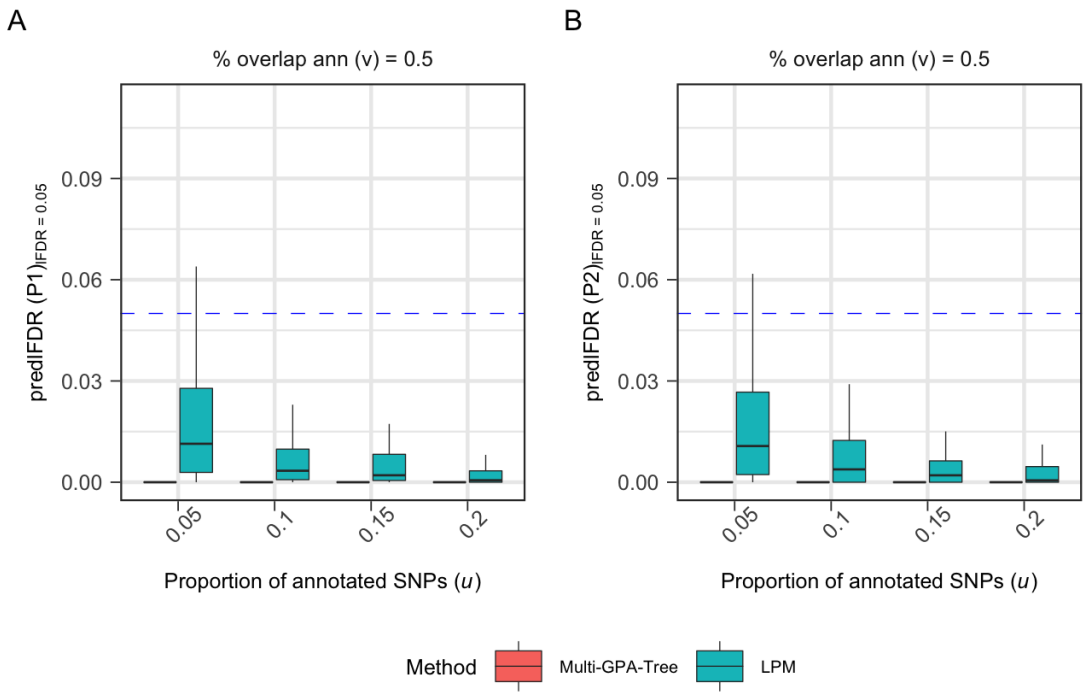


Figure 4.5: Comparison of predicted fdr when fdr is controlled at the nominal level of 0.05 between Multi-GPA-Tree and LPM for traits (A) P_1, and (B) P_2, respectively. The results are presented for different proportions of SNPs annotated in $A_1 - A_6$ (u; x-axis) when the proportion of overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) equals 50%. $M = 10,000$, $K = 15$, $\alpha_1 = \alpha_2 = 0.4$ in $Beta(\alpha_d, 1), d = 1, 2$. Results are summarized from 100 replications. Outliers are not displayed in the plots.

controlled under the nominal level.

4.5.2 Joint Association Results

AUC: Figure 4.6 shows the AUC comparison between Multi-GPA-Tree and LPM for identification of SNPs that are jointly associated with traits P_1 and P_2. Similar to the marginal case, Multi-GPA-Tree showed relatively higher AUC and lower variability in AUC compared to LPM for all u when $v = 50\%$. AUC for both Multi-GPA-Tree and LPM improved as signal-to-noise ratio increased (i.e., as u increased). When

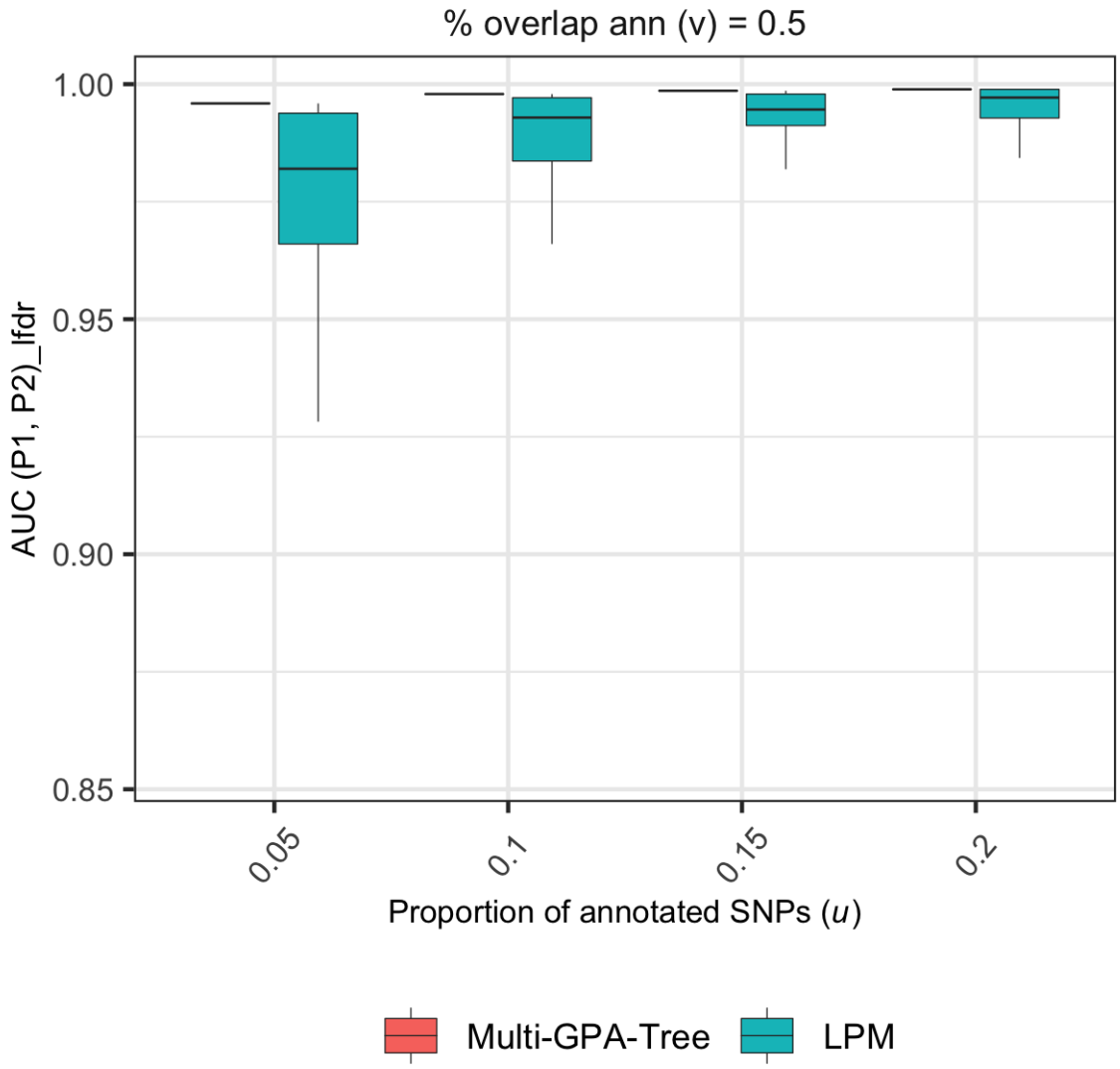


Figure 4.6: Comparison of AUC between Multi-GPA-Tree and LPM to detect SNPs that are jointly associated with traits P_1 andP_2. The results are presented for different proportions of SNPs annotated in $A_1 - A_6$ (u; x-axis) when the proportion of overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) equals 50%. $M = 10,000$, $K = 15$, $\alpha_1 = \alpha_2 = 0.4$ in $Beta(\alpha_d, 1), d = 1, 2$. Results are summarized from 100 replications. Outliers are not displayed in the plots.

signal in the data is highest (i.e., when $u = 20\%$) LPM demonstrated performance closer to Multi-GPA-Tree.

Statistical power: Figure 4.7 compares the statistical power to detect risk SNPs

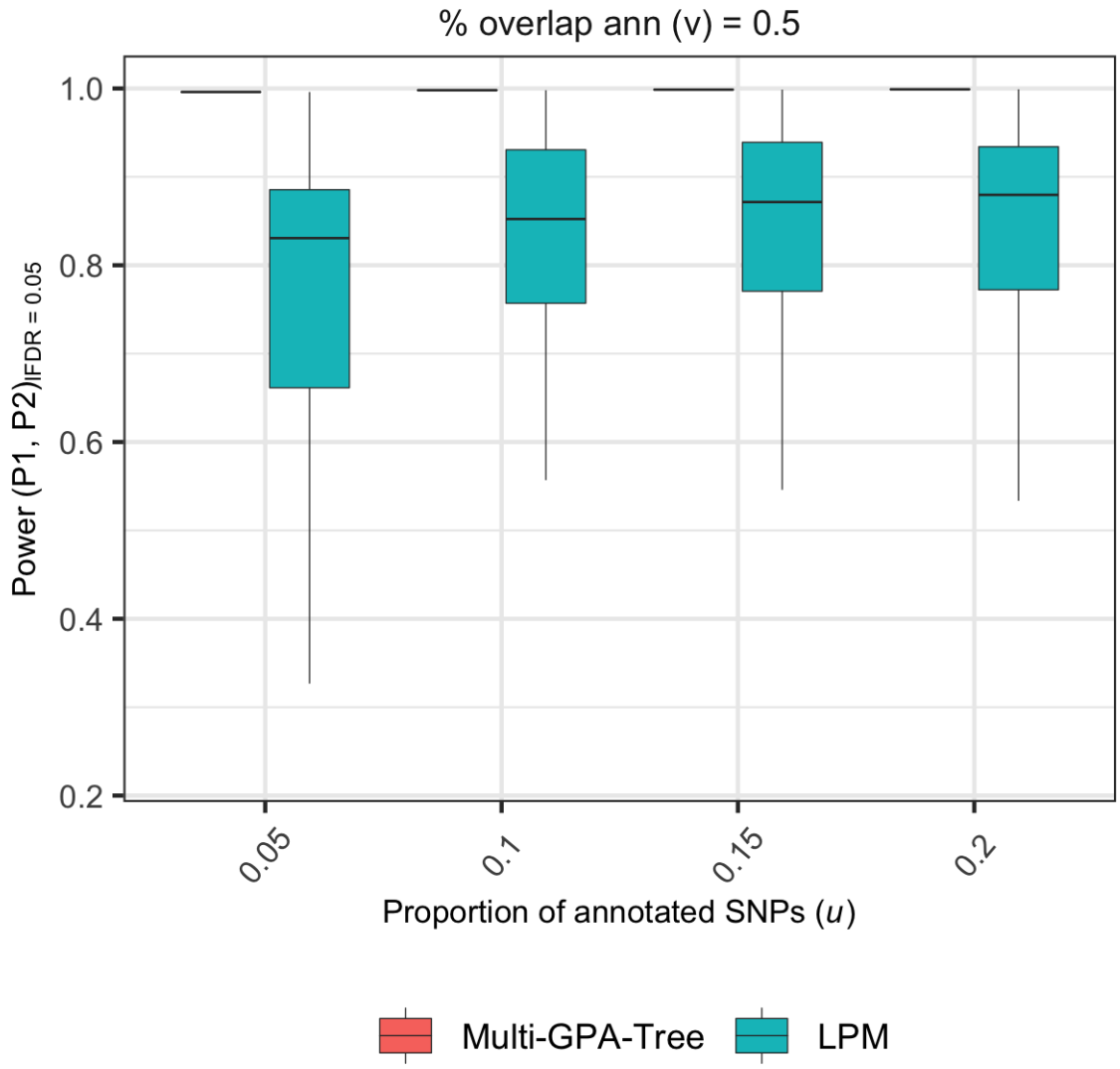


Figure 4.7: Comparison of statistical power between Multi-GPA-Tree and LPM to detect SNPs that are jointly associated with traits P_1 and P_2 when fdr is controlled at the nominal level of 0.05. The results are presented for different proportions of SNPs annotated in $A_1 - A_6$ (u; x-axis) when the proportion of overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) equals 50%. $M = 10,000$, $K = 15$, $\alpha_1 = \alpha_2 = 0.4$ in $Beta(\alpha_d, 1), d = 1, 2$. Results are summarized from 100 replications. Outliers are not displayed in the plots.

that are jointly associated with traits P_1 and P_2 when fdr is controlled at the nominal level of 0.05, using Multi-GPA-Tree and LPM. Compared to LPM, Multi-GPA-

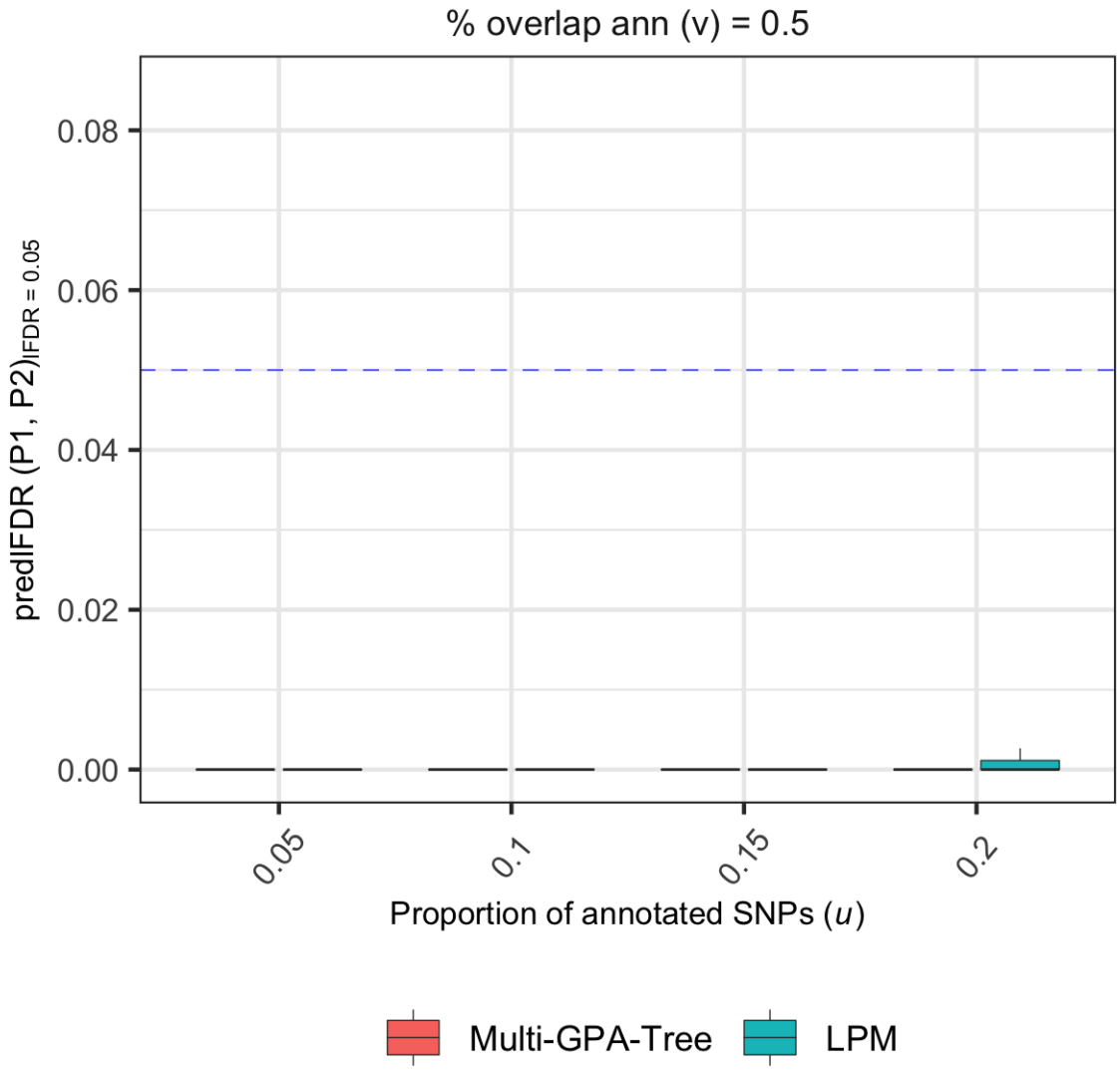


Figure 4.8: Comparison of predicted fdr between Multi-GPA-Tree and LPM when fdr is controlled at the nominal level of 0.05 to detect SNPs that are jointly associated with traits P_1 and P_2. The results are presented for different proportions of SNPs annotated in $A_1 - A_6$ (u; x-axis) when the proportion of overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) equals 50%. $M = 10,000$, $K = 15$, $\alpha_1 = \alpha_2 = 0.4$ in $Beta(\alpha_d, 1), d = 1, 2$. Results are summarized from 100 replications. Outliers are not displayed in the plots.

Tree showed higher statistical power and lower variability in power for all u when $v = 50\%$. The performance of LPM improved in most cases as signal-to-noise ratio increased (i.e., as u increased).

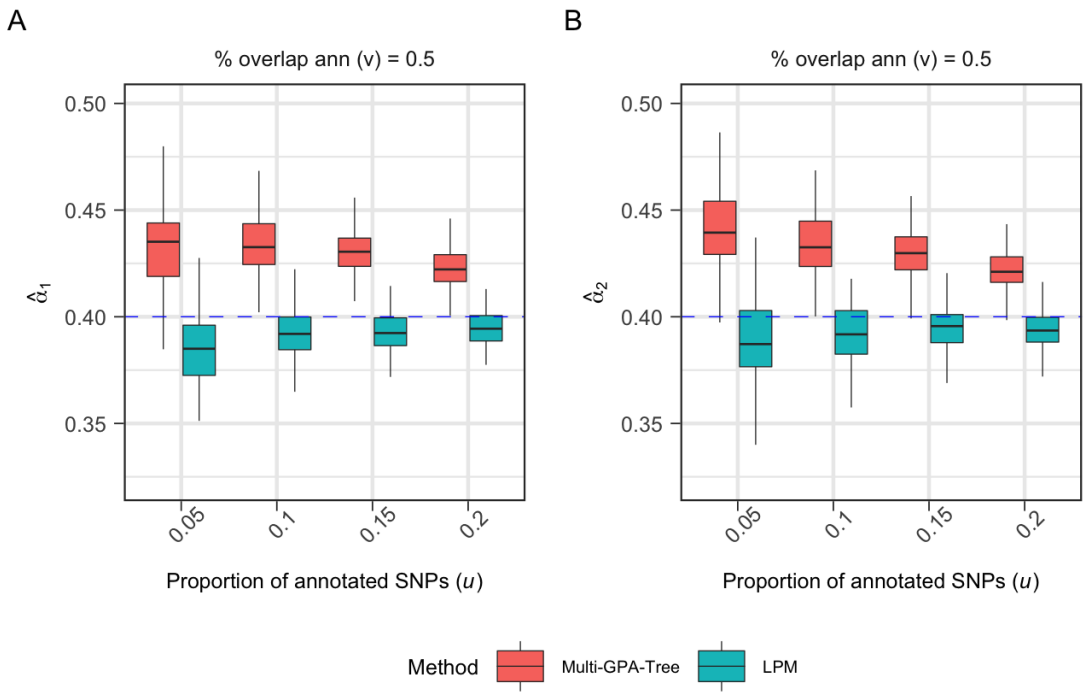


Figure 4.9: Comparison of estimated (A) α_1 and (B) α_2 parameters between Multi-GPA-Tree and LPM for traits P_1 and P_2, respectively. The results are presented for different proportions of SNPs annotated in $A_1 - A_6$ (u; x-axis) when the proportion of overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) equals 50%. $M = 10,000$, $K = 15$, $\alpha_1 = \alpha_2 = 0.4$ in $Beta(\alpha_d, 1), d = 1, 2$. Results are summarized from 100 replications. Outliers are not displayed in the plots.

Predicted fdr control: Figure 4.8 compares the predicted fdr when true fdr is controlled at the nominal level of 0.05 to detect SNPs that are jointly associated with traits P_1 and P_2, using Multi-GPA-Tree and LPM. For both Multi-GPA-Tree and LPM, the predicted fdr is controlled under the nominal level when performing joint association analysis.

4.5.3 Other Results

Estimation of α parameters: Figure 4.9 A and B show the α_1 and α_2 parameter estimates obtained using Multi-GPA-Tree and LPM, respectively. LPM was on average

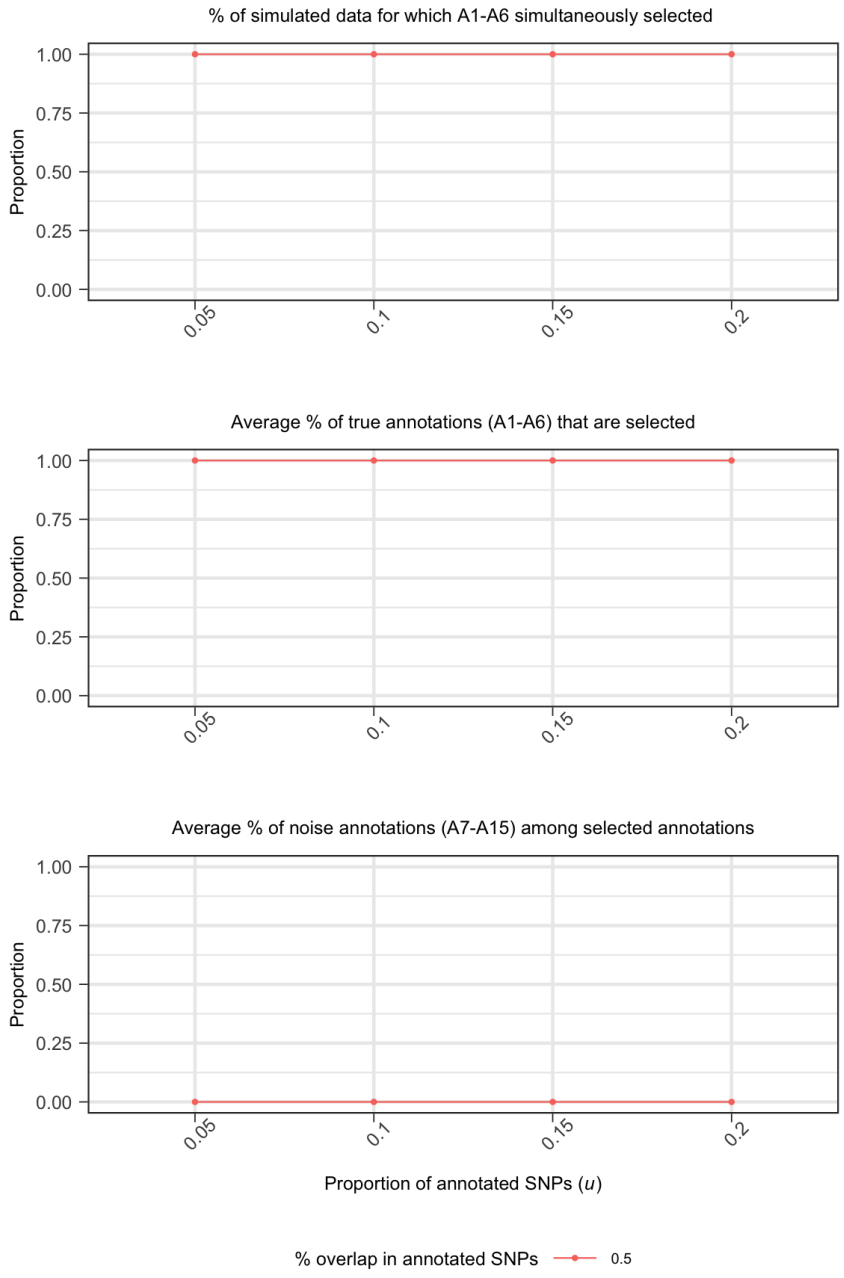


Figure 4.10: Evaluation of accuracy of detecting the correct functional annotation tree based on (A) the proportion of simulation data for which all relevant functional annotations in L_1, L_2 and L_3, i.e, annotations $A_1 - A_6$ were identified simultaneously; (B) the average proportion of true functional annotations $(A_1 - A_6)$ among the functional annotations identified by multi-GPA-Tree; and (C) the average proportion of noise annotations $(A_7 - A_{15})$ among all annotations identified by Multi-GPA-Tree. The results are presented for different proportions of SNPs annotated in $A_1 - A_6$ (u; x-axis) when the proportion of overlap between SNPs annotated in $A_1 - A_2$, $A_3 - A_4$ and $A_5 - A_6$ (v) equals 50%. $M = 10,000$, $K = 15$, $\alpha_1 = \alpha_2 = 0.4$ in $Beta(\alpha_d, 1), d = 1, 2$. Results are summarized from 100 replications.

more accurate than Multi-GPA-Tree in estimating α_1 and α_2. Multi-GPA-Tree generally overestimated both α parameters and this was most notable when u is small. As u increases, the α_1 and α_2 estimates from Multi-GPA-Tree became closer to the true value. We note that overestimation of the α parameters by Multi-GPA-Tree did not impact the method's ability to identify the true combinations of functional annotations or the risk-associated SNPs, which are the main objectives of Multi-GPA-Tree.

Selection of relevant and noise annotations: Figure 4.10A shows the proportion of times only functional annotations in the true combinations L_1, L_2 and L_3 ($A_1 - A_6$) were simultaneously identified by Multi-GPA-Tree and Figure 4.10B shows the average proportion of true functional annotations ($A_1 - A_6$) among the functional annotations identified by multi-GPA-Tree. Multi-GPA-Tree successfully identified all functional annotations included in the true combinations L_1, L_2 and L_3 ($A_1 - A_6$) for all u when $v = 50\%$ (Figure 4.10A-B). Multi-GPA-Tree also never identified any noise annotations ($A_7 - A_{15}$) (Figure 4.10C). These results demonstrate the potential of Multi-GPA-Tree to conservatively identify true annotations.

4.6 Real Data Application

We first obtained a combined dataset including the SLE [42], RA [78], and CD and UC [79] GWAS. Summary statistics in the SLE and RA GWAS was profiled for $18,264$ ($6,748$ cases and $11,516$ controls) and $58,284$ ($14,361$ cases and $43,923$ controls) individuals of European ancestry, respectively. Summary statistics in the CD and UC GWAS was profiled from $8,467$ ($4,686$ cases and $3,781$ controls) individuals of European ancestry. Following quality control and exclusion of SNPs in the MHC region, approximately $375,269$ SNPs were utilized in the final analysis and integrated with functional annotation data from GenoSkyline (GS) [43] and GenoSkylinePlus (GSP) [44]. The Manhattan plots and *p*-value

histogram plots for the four GWAS data are presented in Figure 4.11 and Figure 4.12, respectively.

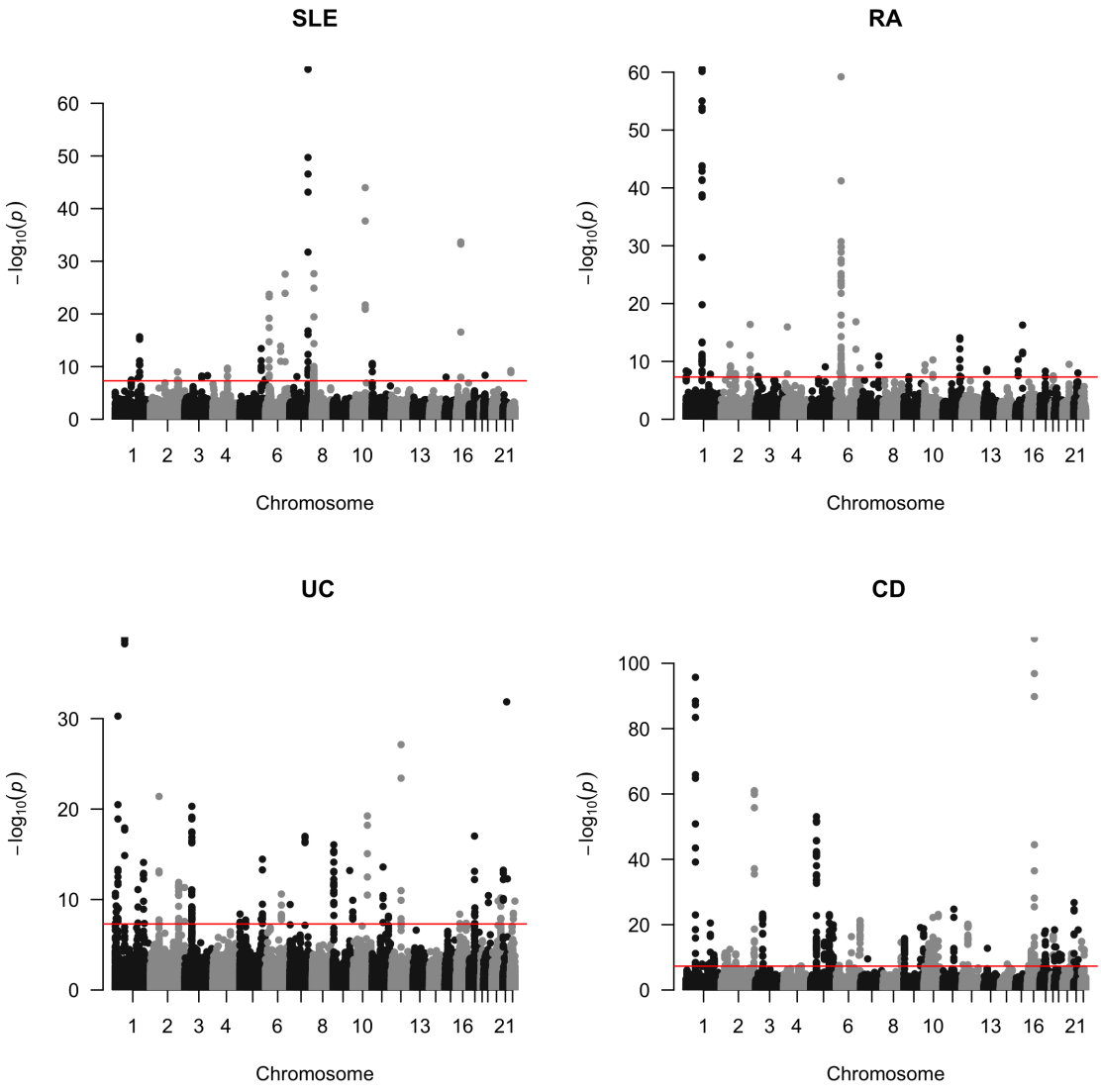


Figure 4.11: Manhattan plot for the four GWAS. Genome-wide significance level ($-log_{10}(5 \times 10^{-8})$) is indicated by the red line.

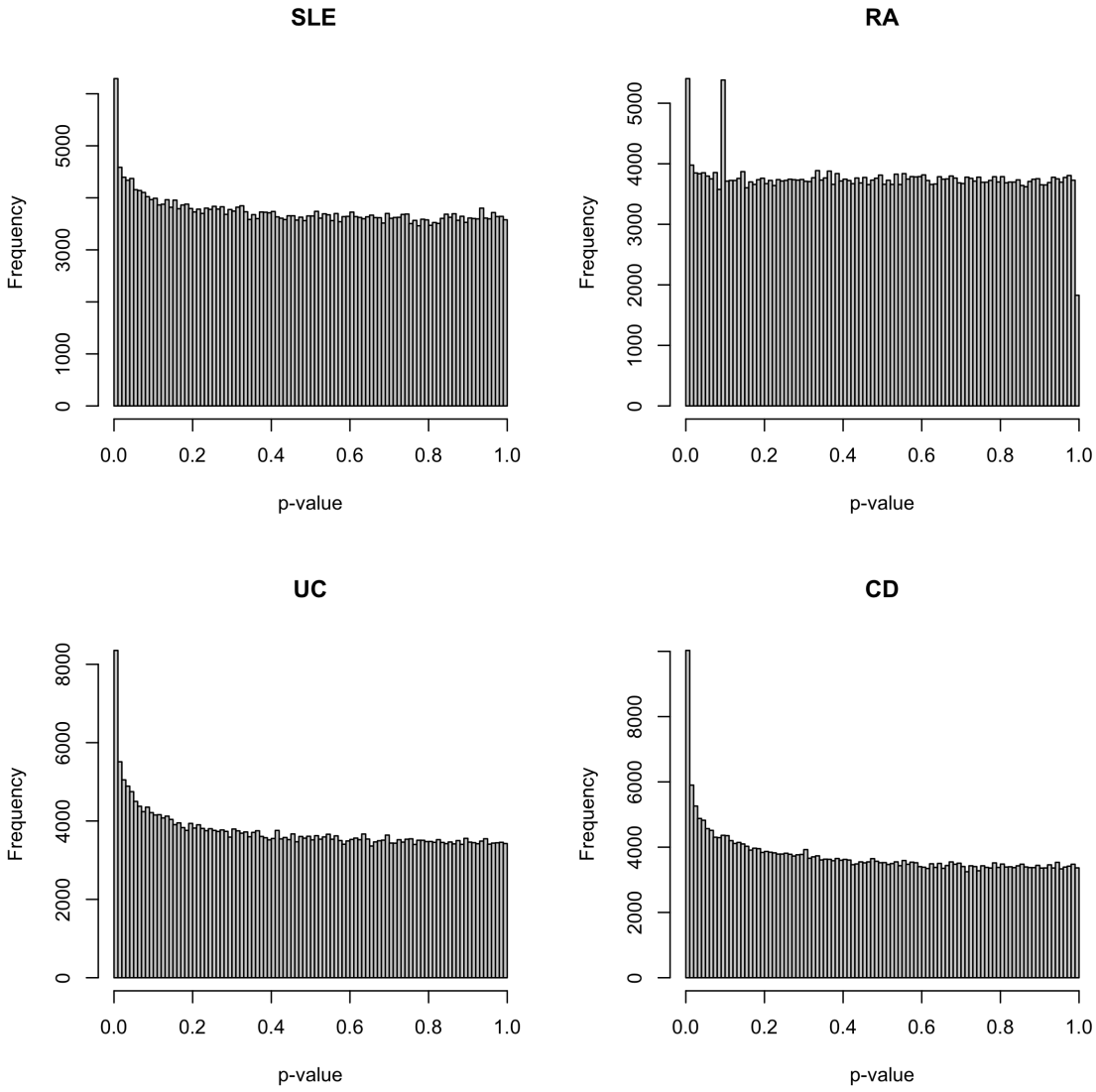


Figure 4.12: GWAS association *p*-value histograms for the four GWAS.

We initially investigated the functional potential of the $375,269$ SNPs using seven tissue-specific GS annotations. With a GS score cutoff of 0.5, 24.15% of SNPs were annotated in at least one of the seven tissue types (Figure 4.13 A) and the percentage of annotated SNPs ranged from 5.72% for lung tissue to 10.44% for GI tissue (Figure 4.13 B). We also measured the overlap in SNPs annotated in different tissue types using log odds ratio (Figure 4.13 C). SNPs annotated for blood tissue overlap less with other tissue types.

On the contrary, SNPs annotated for GI, heart, lung and muscle tissues overlap more with other tissue types. This is consistent with the literature indicating that blood shows the lowest levels of eQTL sharing with other tissue types while muscle and lung tissues show higher levels of eQTL sharing [43, 46].

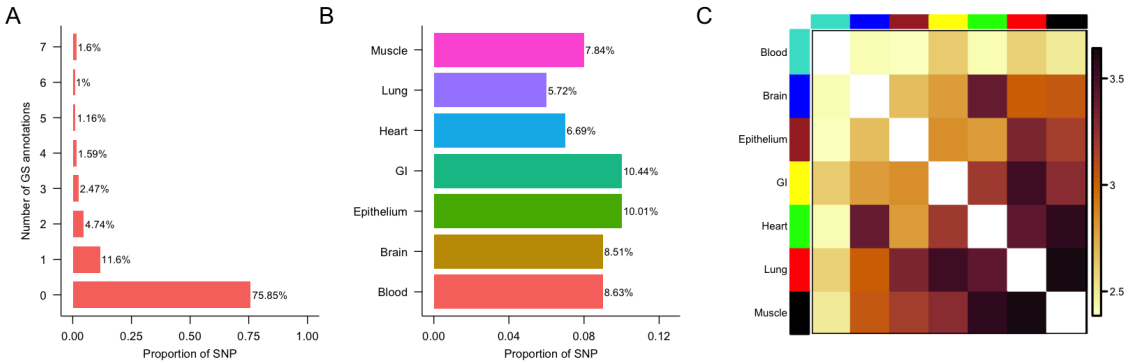


Figure 4.13: Characteristics of $375,269$ SNPs when integrated with seven GenoSkyline (GS) annotations. (A) Number of GS tissues in which SNPs are annotated. (B) Proportion of SNPs that are annotated for each GS tissue type. (C) Overlap of SNPs annotated by seven GS tissue types, calculated using log odds ratio.

Next, we investigated the functional potential of the $375,269$ SNPs using 10 blood-related cell-type specific GSP annotations. With a GSP score cutoff of 0.5, 15.4% of SNPs were annotated in at least one of the 10 blood related cell-type specific annotations (Figure 4.14 A) and the percentage of annotated SNPs ranged from 3.43% for primary T CD8$^+$ memory cells to 6.98% for Primary T regulatory cells (Figure 4.14 B). We also measured the overlap in SNPs annotated in the 10 blood related cell-type specific annotations using log odds ratio (Figure 4.14 C). SNPs annotated for the different types of T cells (Primary helper memory, helper naive, effector/memory enriched, regulatory, CD8$^+$ naive and CD8$^+$ memory T cells) overlap more with each other.

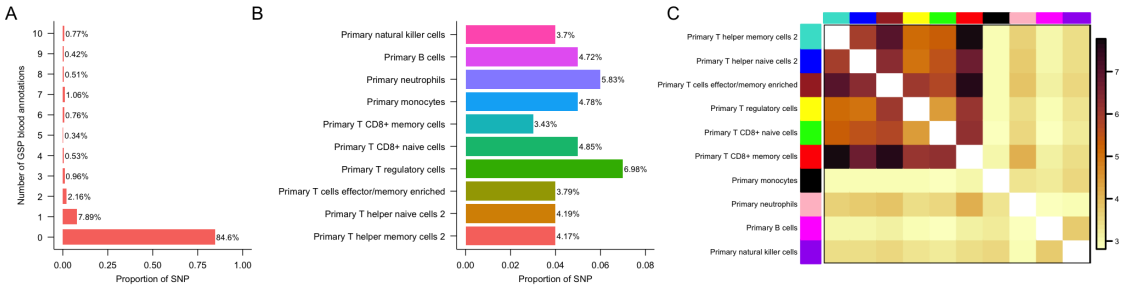


Figure 4.14: Characteristics of $375,269$ SNPs when integrated with 10 blood related GenoSkylinePlus (GSP) annotations. (A) Number of GSP tissues in which SNPs are annotated. (B) Proportion of SNPs that are annotated for each blood related GSP annotations. (C) Overlap of SNPs annotated by 10 blood related GPS annotations, calculated using log odds ratio.

4.6.1 Integration of Systemic Lupus Erythematosus (SLE) and Rheumatoid Arthritis (RA) GWAS

Tissue-level Investigation using GenoSkyline (GS) annotations

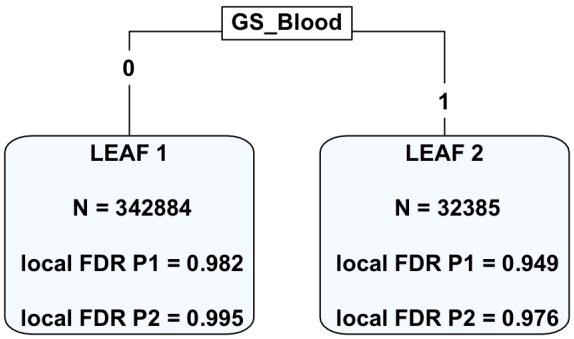


Figure 4.15: Functional annotation tree identified by Multi-GPA-Tree approach when seven tissue-level GenoSkyline (GS) annotations are integrated with SLE and RA GWAS. Each leaf (terminal node) in the tree shows the total number of SNPs in the leaf and the mean local FDR for SLE (P1) and RA (P2) for the SNPs in the leaf.

We applied the Multi-GPA-Tree approach to the SLE and RA GWAS and tissue-specific GS annotations to identify SNPs that are marginally and jointly associated with SLE and

RA, and to characterize the functional annotations relevant to single and multiple trait risk-associated SNPs. At the nominal local FDR level of 0.05, Multi-GPA-Tree identified 519 SNPs that are marginally associated with SLE, 388 SNPs that are marginally associated with RA, and 202 SNPs that are jointly associated with both SLE and RA.

In the joint analysis of SLE and RA with tissue-specific GS annotations, the original Multi-GPA-Tree model fit identified blood tissue at the root node and included 2 leaves. Further investigation of the Multi-GPA-Tree model results showed that, among the 519 SNPs that are marginally associated with SLE, 180 are annotated for blood tissue, among the 388 SNPs that are marginally associated with RA, 129 are annotated for blood tissue, and among the 202 SNPs that are jointly associated with both SLE and RA, 94 are annotated for blood tissue.

Cell-type-level Investigation using GenoSkylinePlus (GSP) annotations

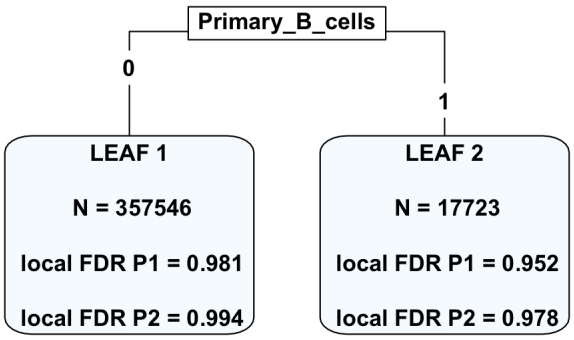


Figure 4.16: Functional annotation tree identified by Multi-GPA-Tree approach when 10 blood related GenoSkylinePlus (GSP) annotations are integrated with SLE and RA GWAS. Each leaf (terminal node) in the tree shows the total number of SNPs in the leaf and the mean local FDR for SLE (P1) and RA (P2) for the SNPs in the leaf.

Based on the observed relationship between GS annotation for blood tissue and SLE and RA, in the second phase of the analysis, we applied the Multi-GPA-Tree approach to the SLE and RA GWAS and 10 blood related cell-specific GSP annotations to identify SNPs

that are marginally and jointly associated with SLE and RA, and to characterize the blood related GSP functional annotations relevant to single and multiple trait risk-associated SNPs. At the nominal local FDR level of 0.05, Multi-GPA-Tree identified 485 SNPs that are marginally associated with SLE, 381 SNPs that are marginally associated with RA, 177 SNPs that are jointly associated with SLE and RA.

In the joint analysis of SLE and RA with 10 blood related cell-type specific GSP annotations, the original Multi-GPA-Tree model fit identified primary B cells at the root node and included 2 leaves. Further investigation model results showed that, among the 485 SNPs that are marginally associated with SLE, 90 are annotated for primary B cells, among the 381 SNPs that are marginally associated with RA, 75 are annotated for primary B cells, and among the 177 SNPs that are jointly associated with both SLE and RA, 42 are annotated for primary B cells.

4.6.2 Integration of Ulcerative Colitis (UC) and Crohn's Disease (CD) GWAS

Tissue-level Investigation using GenoSkyline (GS) annotations

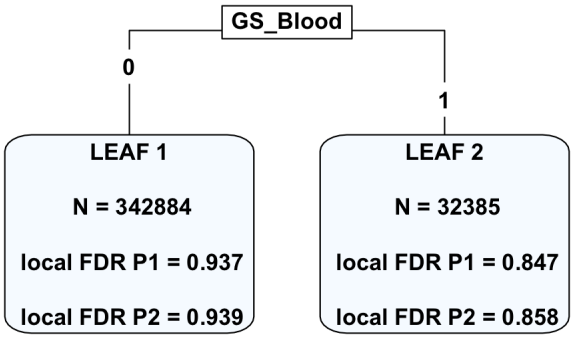


Figure 4.17: Functional annotation tree identified by Multi-GPA-Tree approach when seven tissue-level GenoSkyline (GS) annotations are integrated with UC and CD GWAS. Each leaf (terminal node) in the tree shows the total number of SNPs in the leaf and the mean local FDR for UC (P1) and CD (P2) for the SNPs in the leaf.

We also applied the Multi-GPA-Tree approach to the UC and CD GWAS and tissue-specific GS annotations to identify SNPs that are marginally and jointly associated with UC and CD, and to characterize the functional annotations relevant to one or more trait risk-associated SNPs. At the nominal local FDR level of 0.05, Multi-GPA-Tree identified $2,485$ SNPs that are marginally associated with UC, $2,304$ SNPs that are marginally associated with CD, and $2,304$ SNPs that are jointly associated with both UC and CD.

In the joint analysis of UC and CD with tissue-specific GS annotations, the original Multi-GPA-Tree model fit identified blood tissue at the root node and included 2 leaves. Further investigation of the Multi-GPA-Tree model results showed that, among the $2,485$ SNPs that are marginally associated with UC, 629 are annotated for blood tissue, among the $2,304$ SNPs that are marginally associated with CD, 561 are annotated for blood tissue, and

among the $2,304$ SNPs that are jointly associated with both UC and CD, 561 are annotated for blood tissue.

Cell-type-level Investigation using GenoSkylinePlus (GSP) annotations

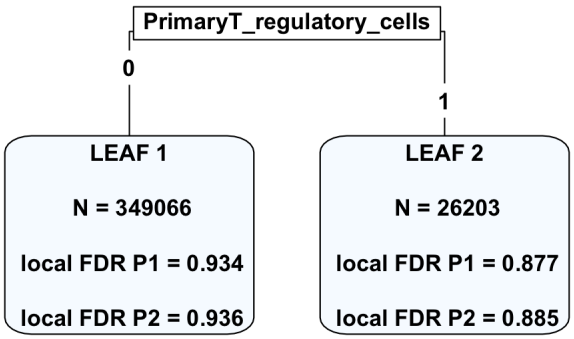


Figure 4.18: Functional annotation tree identified by Multi-GPA-Tree approach when 10 blood related GenoSkylinePlus (GSP) annotations are integrated with UC and CD GWAS. Each leaf (terminal node) in the tree shows the total number of SNPs in the leaf and the mean local FDR for UC (P1) and CD (P2) for the SNPs in the leaf.

Based on the observed relationship between GS annotation for blood tissue, and UC and CD, in the second phase of the analysis, we applied the Multi-GPA-Tree approach to the UC and CD GWAS and 10 blood related cell-specific GSP annotations to identify SNPs that are marginally and jointly associated with UC and CD, and to characterize the blood related GSP functional annotations relevant to single and multiple trait risk-associated SNPs. At the nominal local FDR level of 0.05, Multi-GPA-Tree identified $2,176$ SNPs that are marginally associated with UC, $2,217$ SNPs that are marginally associated with CD, $1,959$ SNPs that are jointly associated with UC and CD.

In the joint analysis of UC and CD with 10 blood related cell-type specific GSP annotations, the original Multi-GPA-Tree model fit identified primary T regulatory cells at the root node and included 2 leaves. Further investigation of the model results showed that, among the $2,176$ SNPs that are marginally associated with UC, 294 are annotated for pri-

mary T regulatory cells, among the $2,217$ SNPs that are marginally associated with CD, 324 are annotated for primary T regulatory cells, and among the $1,959$ SNPs that are jointly associated with both UC and CD, 222 are annotated for primary T regulatory cells.

4.7 Conclusions

Several statistical methodologies that efficiently integrate GWAS summary statistics and functional annotation data for multiple traits already exists. However, these methods are not able to identify the combinations of functional annotations that act in unison to influence one or more traits. We propose a novel statistical methodology, Multi-GPA-Tree, to integrate GWAS summary statistics and functional annotation data by leveraging pleiotropy with the goal to identify risk-associated SNPs and the combinations of functional annotations related to one or more trait risk-associated SNPs.

Multi-GPA-Tree is a hierarchical model, and is implemented by combining an iterative procedure (EM algorithm) and a multivariate decision tree algorithm (MRT). Multi-GPA-Tree assumes that given the latent status of the SNPs that define their association status with one or more traits, their GWAS association p-values come from a Beta-Uniform mixture distribution. Additionally, SNPs are assumed to be conditionally independent given their functional annotation information.

We evaluate the performance of Multi-GPA-Tree using simulated data and compare its performance with existing statistical approaches. Multi-GPA-Tree showed the higher AUC, higher statistical power and controlled fdr to detect risk-associated SNPs for one or more traits compared to existing approaches. Multi-GPA-Tree also successfully identified the true combinations of functional annotations in most cases, facilitating understanding of potential biological mechanisms linking risk-associated SNPs with one more more complex traits. Overall, the ability of Multi-GPA-Tree to improve SNP prioritization and attribute

functional characteristics to one or more trait risk-associated SNPs or gene locus can be powerful in facilitating our understanding of genetic susceptibility factors related to complex traits.

5. Specific Aim 3

5.1 Introduction

Aim 3 will focus on developing an R package and an R Shiny App to implement the statistical methodologies developed in Aims 1 and 2. The goal of this aim is to make the methods easily accessible to basic and clinical science researchers. The R package is called 'GPATree' and the R Shiny App is called 'ShinyGPATree'. GPATree can be utilized to integrate single as well as multiple GWAS association p-values with binary functional annotation data as described in Aims 1 and 2. Implementation of the GPATree package is useful in characterizing the relationship between one or more traits and in obtaining the combinations of functional annotations that are relevant to one or more trait risk-associated SNPs. Several functions are implemented as part of the GPATree package. These functions along with their usage are described in the sections below.

5.2 The R Package 'GPATree'

Package 'GPATree'

June 11, 2021

Title A package to implement the GPA-Tree method

Version 0.0.0.9000

Depends R (>= 3.5.0)

Description This package implements the GPA-Tree methodology for post GWAS analysis.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports readr, rpart, rpart.plot, quantreg, rpart.utils, gtools, dplyr, stringr, base, shiny, DT, graphics, pracma, methods, mvpart

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Aastha Khatiwada [aut, cre] (<https://orcid.org/0000-0002-3565-451X>)

Maintainer Aastha Khatiwada <asthakhatiwada@gmail.com>

R topics documented:

GPATree-package *GPATree: A package to implement the GPA-Tree method*

Description

This package provides functions for fitting GPA-Tree, a statistical approach for integrative analysis of genome wide association studies (GWAS) data and functional annotation information within a unified framework. GPA-Tree simultaneously identifies disease risk-associated SNPs and combinations of functional annotations that potentially explain the mechanisms through which risk-associated SNPs are related with phenotypes.

Details

- Package: GPATree
- Type: Package
- Version: 0.0.0.9000
- Date: 2021-02-16
- License: GPL(>=3)
- LazyLoad: yes

This package contains a main class, GPATree, which represents GPATree model fit. This package contains five main methods for the GPATree framework, GPATree, plot, assoc, prune, leaf. GPA-Tree method fits the GPATree model and assoc method implements association mapping. leaf provided information regarding functional annotations that are enriched for the leaves in the GPATree model results. plot allows plotting the GPATree model result, and prune allows further pruning the GPATree. This package also contains a methods for the ShinyGPATree visualization, association mapping and functional annotation tree selection toolkit. ShinyGPATree opens the ShinyGPATree interface, which takes the results generated from GPATree method as input.

Author(s)

Aastha Khatiwada <asthakhatiwada@gmail.com>

See Also

GPATree, assoc, leaf, prune, plot, ShinyGPATree

Examples

```
## Not run:
library(GPATree)

# load GPATree example data
data(GPATreeExampleData)

# fitting the GPATree model
fit <- GPATree(GPATreeExampleData$gwasPval, GPATreeExampleData$annMat)

# get functional annotation information
leaf(fit)
```

```
# association mapping
assoc.gpatree <- assoc(fit, FDR = 0.01, fdrControl = 'global')

# pruning the GPATree model fit
pruned.fit <- prune(fit, cp = 0.005)

# plotting the GPATree model results
plot(fit)
plot(pruned.fit)

# run the ShinyGPATree app using output from the GPATree method
ShinyGPATree(fit)

## End(Not run)
```

---

assoc *Association mapping*

Description

This function will implement association mapping for the GPA-Tree model.

Usage

```
## S4 method for signature 'GPATree'
assoc(object, FDR = 0.05, fdrControl = "global")
```

Arguments

object	An object of class GPATree.
FDR	FDR level. Value has to be between 0 and 1.
fdrControl	Method to control FDR. Possible values are "global" (global FDR control) and "local" (local FDR control).

Value

Returns a MX2 matrix where the row represents SNPs, the fist column indicates the association between each SNP and phenotype, and the second column indicates the leaf in which the SNP falls

Author(s)

Aastha Khatiwada

Examples

```
## Not run:
library(GPATree)

# load GPATree example data
data(GPATreeExampleData)

#fitting the GPATree model
fit <- GPATree(GPATreeExampleData$gwasPval, GPATreeExampleData$annMat)
```

```
# pruning the GPATree model fit
assoc.fit <- assoc(fit, FDR = 0.05, fdrControl = "global")

## End(Not run)
```

`decTree` *GPATree selected decision tree*

Description

This function will extract the combinations of functional annotations selected by the decision tree in the stage 2 of GPATree method that meets the provided threshold in minPredictedProb.

Usage

```
## S4 method for signature 'GPATree'
decTree(object)
```

Arguments

`object`	An object of class GPATree.

Value

A list containing variables in combinations selected by the decision tree (CART_PIs), the combination (CART_PIs_comb) and the predicted proportions for the selected PIs(assoc_pred) meet the provided threshold in minPredictedProb.

Author(s)

Aastha Khatiwada

`GPATree` *Fit GPA-Tree model*

Description

This function will implement the GPA-Tree and the Multi-GPA-Tree approach for integrative analysis of GWAS and functional annotation data.

Usage

```
GPATree(gwasPval, annMat, initAlpha = 0.1, cpTry = 0.001)
```

Arguments

gwasPval	A matrix of M X 1 dimension, where M is the number of SNPs. The matrix includes the GWAS association p-values for the phenotype. P-values must be between 0 and 1.
annMat	A matrix of binary annotations, where rows and columns correspond to SNPs and annotations, respectively.
initAlpha	Initial value for alpha estimate. Default is 0.1.
cpTry	Complexity parameter (cp) value to be used. cpTry can be between 0 and 1 or NULL. Default is 0.001. When cpTry is NULL, GPATree will select the optimal cp to be used.

Details

The GPATree() function fits the GPATree model. It requires to provide GWAS p-value to gwasPval and binary annotation data to annMat. It is assumed that number of rows of matrix in gwasPval and annMat are equal and correspond to the same SNP.

The assoc() function implements association mapping.

The plot() function takes in an object of class GPATree and will plot the functional annotation tree from the GPATree model.

The leaf() function takes in an object of class GPATree and will provide information regarding the functional annotations that are enriched (1) or not enriched (0) for SNPs in any leaf of the GPATree model plot.

The prune() function takes in an object of class GPATree and a cp parameter and will prune the GPATree model result. This function can be useful when the tree obtained from GPATree model is huge.

The ShinyGPATree app provides visualization of the GPA-Tree model, identifies risk-associated SNPs, and characterizes the combinations of functional annotations that can describe the risk-associated SNPs. The app can also be utilized to improve the visualization of the GPA-Tree model fit to collate or separate layers of the model (add or remove leaves).

Value

Contructs a GPATree class object

Author(s)

Aastha Khatiwada

Examples

```
## Not run:
library(GPATree)

# load GPATree example data
data(GPATreeExampleData)

# fitting the GPATree model
fit <- GPATree(GPATreeExampleData$gwasPval, GPATreeExampleData$annMat)

# get functional annotation information
leaf(fit)
```

```
# association mapping
assoc.gpatree <- assoc(fit, FDR = 0.01, fdrControl = 'global')

# pruning the GPATree model fit
pruned.fit <- prune(fit, cp = 0.005)

# plotting the GPATree model results
plot(fit)
plot(pruned.fit)

# run the ShinyGPATree app using output from the GPATree method
ShinyGPATree(fit)

## End(Not run)
```

GPATreeExampleData *GPATreeExampleData*

Description

Simulated data for the GPATree package

Usage

```
GPATreeExampleData
```

Format

A List that contains 2 data frames.

The first data frame contains 10,000 rows and 1 column. The rows represent SNPs and the columns represent the GWAS association p-value for the association between the SNPs and phenotype

The second dataframe contains 10,000 rows and 10 columns. The rows represent SNPs and the columns represent 10 binary functional annotations.

Source

Simulated Data

GPATreeStage1 *Implement Stage 1 of the GPA-Tree Method.*

Description

This function will implement stage 1 of the GPA-Tree method.

Usage

```
GPATreeStage1(gwasPval, annMat, initAlpha = 0.1)
```

Arguments

gwasPval	A matrix of M X 1 dimension where M is the number of SNPs. The matrix contains GWAS association p-values. Values must be between 0 and 1.
annMat	A matrix of binary annotations, where row and column correspond to SNPs and annotations, respectively.
initAlpha	Initial value for alpha estimate. Default is 0.1.

Value

This function returns a List including:

- numIterConvergence: number of iterations taken for Stage 1 of GPA-Tree to converge.
- pi: predicted posterior probability of being a non-null SNP in Stage 1 of GPA-Tree Method.
- alpha: estimated alpha of GPA-Tree Method.
- beta: beta parameters from the linear model fitted at convergence of Stage 1 of GPA-Tree Method.

Author(s)

Aastha Khatiwada

GPATreeStage2 *Implement Stage 2 of the GPA-Tree approach.*

Description

This function will implement Stage 2 of the GPA-Tree approach.

Usage

```
GPATreeStage2(gwasPval, annMat, alphaStage1, initPi, cpTry)
```

Arguments

gwasPval	A matrix of M X 1 dimension, where M is the number of SNPs. The first column is labeled 'SNPid' and contains the SNPid. The second column contains the GWAS association p-values and is called 'P1'. The values in P1 must be between 0 and 1.
annMat	A matrix of binary annotations, where row and column correspond to SNPs and annotations, respectively.
alphaStage1	alpha estimated in stage 1 of the GPA-Tree approach.
initPi	pi estimated in stage 1 of the GPA-Tree approach.
cpTry	Complexity parameter (cp) value to be used to build annotation decision tree. cpTry can be between 0 and 1 or NULL. Default is 0.001. When cpTry is NULL, GPATree will select the optimal cp to be used.

Value

This function returns a `List` including:

- numIterConvergence: number of iterations taken for GPA-Tree to converge.
- licVec: incomplete log-likelihood from Stage 1 of Multi-GPA-Tree Method.
- lcVec: complete log-likelihood from Stage 1 of Multi-GPA-Tree Method.
- Z: posterior probability of being in the 4 groups using Multi-GPA-Tree method.
- Zmarg: marginal probability of being non-null for the phenotypes.
- pi: predicted posterior probability of being a non-null SNP in Stage 1 of Multi-GPA-Tree Method.
- fit: CART model selected by GPATree.
- fitSelectVar: annotations included in CART tree.

Author(s)

Aastha Khatiwada

`leaf` *Functional annotation tree.*

Description

This function will provide the annotation combinations relevant to risk-associated SNPs.

Usage

```
## S4 method for signature 'GPATree'
leaf(object)
```

Arguments

`object`	An object of class GPATree.

Value

Returns a matrix where each row corresponds to a leaf from the GPA-Tree model fit and contains information regarding the local FDR for SNPs in the leaf, and also information regarding annotations that are enriched (1) or not enriched (0) for the leaf.

Author(s)

Aastha Khatiwada

Examples

```
## Not run:
library(GPATree)

# load GPATree example data
data(GPATreeExampleData)

#fitting the GPATree model
fit <- GPATree(GPATreeExampleData$gwasPval, GPATreeExampleData$annMat)
leaf(fit)

## End(Not run)
```

`mGPATreeStage1` *Implement Stage 1 of the Multi-GPA-Tree Method*

Description

This function will implement stage 1 of the Multi-GPA-Tree method.

Usage

```
mGPATreeStage1(gwasPval, annMat, initAlpha = 0.1)
```

Arguments

`gwasPval`	A matrix of M X 2 dimension where M is the number of SNPs. The first column contains the SNP id and is labeled 'SNPid' and the second column contains the GWAS association p-values and is called P1. Values in P1 must be between 0 and 1.
`annMat`	A matrix of binary annotations, where row and column correspond to SNPs and annotations, respectively.
`initAlpha`	Initial value for alpha estimate. Default is 0.1.

Value

This function returns a `List` including:

- numIterConvergence: number of iterations taken for Stage 1 of Multi-GPA-Tree to converge.
- alpha: estimated alpha parameters using Multi-GPA-Tree Method.
- beta: beta parameters from the linear model fitted at convergence of Stage 1 of Multi-GPA-Tree Method.
- Z: posterior probability of being in the 4 groups using Multi-GPA-Tree method.
- Zmarg: marginal probability of being non-null for the phenotypes.
- pi: predicted posterior probability of being a non-null SNP in Stage 1 of Multi-GPA-Tree Method.
- licVec: incomplete log-likelihood from Stage 1 of Multi-GPA-Tree Method.
- lcVec: complete log-likelihood from Stage 1 of Multi-GPA-Tree Method.
- annMat: annotation data matrix
- gwasPval: GWAS p-value matrix

Author(s)

Aastha Khatiwada

`mGPATreeStage2` *Implement stage 2 of the Multi-GPA-Tree Method.*

Description

This function will implement the Multi-GPA-Tree method for multiple phenotypes while leveraging pleiotropy.

Usage

```
mGPATreeStage2(gwasPval, annMat, alphaStage1, initPi, cpTry)
```

Arguments

`gwasPval`	A matrix of M X 2 dimension, where M is the number of SNPs and 2 is the number of traits. The columns contains the GWAS association p-values for the respective traits The pvalues must be between 0 and 1.
`annMat`	A matrix of binary annotations, where row and column correspond to SNPs and annotations, respectively.
`alphaStage1`	Alpha estimated in stage 1
`initPi`	final alpha at convergence of stage 1
`cpTry`	Complexity parameter (cp) value to be used to build multivaraite CART model. cpTry can be between 0 and 1 or NULL. Default is 0.001. When cpTry is NULL, GPATree will select the optimal cp to be used.

Value

This function returns a `List` including:

- numIterConvergence: number of iterations taken for GPA-Tree to converge.
- fit: CART model selected by GPATree.
- fitSelectVar: annotations included in CART tree.
- Z: posterior probability of being in the 4 groups using Multi-GPA-Tree method.
- Zmarg: marginal probability of being non-null for the phenotypes.
- pi: predicted posterior probability of being a non-null SNP in Stage 1 of Multi-GPA-Tree Method.
- licVec: incomplete log-likelihood from Stage 1 of Multi-GPA-Tree Method.
- lcVec: complete log-likelihood from Stage 1 of Multi-GPA-Tree Method.
- annMat: annotation data matrix
- gwasPval: GWAS p-value matrix

Author(s)

Aastha Khatiwada

plot *Plot the functional annotation tree*

Description

This function will plot the functional annotation tree for the GPA-Tree model fit.

Usage

```
## S4 method for signature 'GPATree,missing'
plot(x, y, ...)
```

Arguments

x	An object of class GPATree.
y	missing (not required).
...	...

Value

Returns a plot for the functional annotation tree from the GPA-Tree model fit.

Author(s)

Aastha Khatiwada

Examples

```
## Not run:
library(GPATree)

# load GPATree example data
data(GPATreeExampleData)

#fitting the GPATree model
fit <- GPATree(GPATreeExampleData$gwasPval, GPATreeExampleData$annMat)

# plotting the GPATree model fit
plot(fit)

## End(Not run)
```

prune *Prune GPA-Tree model fit*

Description

This function will prune the GPA-Tree model fit using the given cp value.

Usage

```
## S4 method for signature 'GPATree'
prune(object, cp = 0.001)
```

Arguments

object	An object of class GPATree.
cp	The cp parameter to be used for pruning. cp must be between 0 and 1.

Value

GPA-Tree model output.

Author(s)

Aastha Khatiwada

Examples

```
## Not run:
library(GPATree)

# load GPATree example data
data(GPATreeExampleData)

#fitting the GPATree model
fit <- GPATree(GPATreeExampleData$gwasPval, GPATreeExampleData$annMat)

# pruning the GPATree model fit
pruned.fit <- prune(fit, cp = 0.005)

## End(Not run)
```

quantile_reg_model *quantile_reg_model result*

Description

Quantile regression model to predict complexity parameter in Stage 2 of the GPA-Tree method

Usage

```
quantile_reg_model
```

Format

An object of class 'rq'. Contains 14 elements:

- coefficients: coefficient of the quantile regression model
- x: provides the x side of the regression model
- y: provides the y side of the regression model
- residuals: the residuals from the fit.
- dual: the vector dual variables from the fit
- fitted.values:
- formula: formula used to fit the quantile regression model.
- terms: terms of the model
- xlevels:
- call: function call
- tau: percentile used in the quantile regression
- rho: The value(s) of objective function at the solution.
- method: the algorithmic method used to compute the fit. There are several options: The default method is the modified version of the Barrodale and Roberts algorithm for l1-regression, used by l1fit in S, and is described in detail in Koenker and d'Orey(1987, 1994), default = "br". This is quite efficient for problems up to several thousand observations, and may be used to compute the full quantile regression process. It also implements a scheme for computing confidence intervals for the estimated parameters, based on inversion of a rank test described in Koenker(1994). For larger problems it is advantageous to use the Frisch–Newton interior point method "fn". And for very large problems one can use the Frisch–Newton approach after preprocessing "pfn". Both of the latter methods are described in detail in Portnoy and Koenker(1997), this method is primarily well-suited for large n, small p problems where the parametric dimension of the model is modest. For large problems with large parametric dimension it is usually advantageous to use method "sfn" which uses the Frisch-Newton algorithm, but exploits sparse algebra to compute iterates. This is typically helpful when the model includes factor variables that, when expanded, generate design matrices that are very sparse. A sixth option "fnc" that enables the user to specify linear inequality constraints on the fitted coefficients; in this case one needs to specify the matrix R and the vector r representing the constraints in the form Rb q r. See the examples. Finally, there are two penalized methods: "lasso" and "scad" that implement the lasso penalty and Fan and Li's smoothly clipped absolute deviation penalty, respectively. These methods should probably be regarded as experimental.
- model: optionally the model frame, if model=TRUE.

Source

Quantile Regression Model fitted using simulated data

See Also

rq

ShinyGPATree *Run ShinyGPATree App*

Description

This function will initialize the ShinyGPATree App for dynamic and interactive visualization of GPA-Tree model results.

Usage

```
ShinyGPATree(object)
```

Arguments

object	An object of class GPATree.

Value

Output of GPA-Tree model.

Author(s)

Aastha Khatiwada

Examples

```
## Not run:
library(GPATree)

# load GPATree example data
data(GPATreeExampleData)

#fitting the GPATree model
fit <- GPATree(GPATreeExampleData$gwasPval, GPATreeExampleData$annMat)

# initialize the ShinyGPATree app
ShinyGPATree(fit)

## End(Not run)
```

Index

5.3 The R Shiny App 'ShinyGPATree'

We implemented the forementioned GPA-Tree algorithm as an R package `'GPATree'`. To further facilitate user's convenience, we developed `'ShinyGPATree'`, a Shiny app for interactive analysis of risk-associated SNPs and the functional annotation tree (Fig 5.1). This Shiny app can be open by sequentially running `GPATree()` and `ShinyGPATree()` functions. First, the `GPATree()` function takes 4 arguments: `gwasPval, annMat, initAlpha and cpTry`. `gwasPval` is a $M \times 1$ matrix of GWAS association *p*-values for M SNPs, `annMat` is a $M \times K$ matrix of K binary functional annotations for M SNPs, `initAlpha` is the initial alpha value to be used to fit the GPA-Tree model (default value = 0.1), and `cpTry` is the cp parameter to be used to fit the GPA-Tree model (default value = 0.001). The `GPATree()` function generates a GPA-Tree model fit required for the ShinyGPATree app. The `ShinyGPATree()` function takes the output of `GPATree()` as an input and opens the ShinyGPATree app using the R code below.

```
R> fit <- GPATree(gwasPval, annMat, initAlpha, cpTry)
R> ShinyGPATree(fit)
```

The ShinyGPATree app provides visualization of the GPA-Tree model fit, identifies risk-associated SNPs, and characterizes the combinations of functional annotations that can describe the risk-associated SNPs. The app also allows to improve the visualization of the GPA-Tree model fit by collating or separating layers of the model using the cp parameter. The numbers of non-risk-associated and risk-associated SNPs that can be characterized by combinations of functional annotations are also automatically updated based on user-selected cp, FDR type (global vs. local) and FDR level values. The interactive nature of the app allows users to effortlessly interact with the GPA-Tree model results to generate plots, prioritize risk SNPs, and make inferences about relevant combinations of functional annotations for the risk-associated SNPs. ShinyGPATree consists of two main tabs, namely

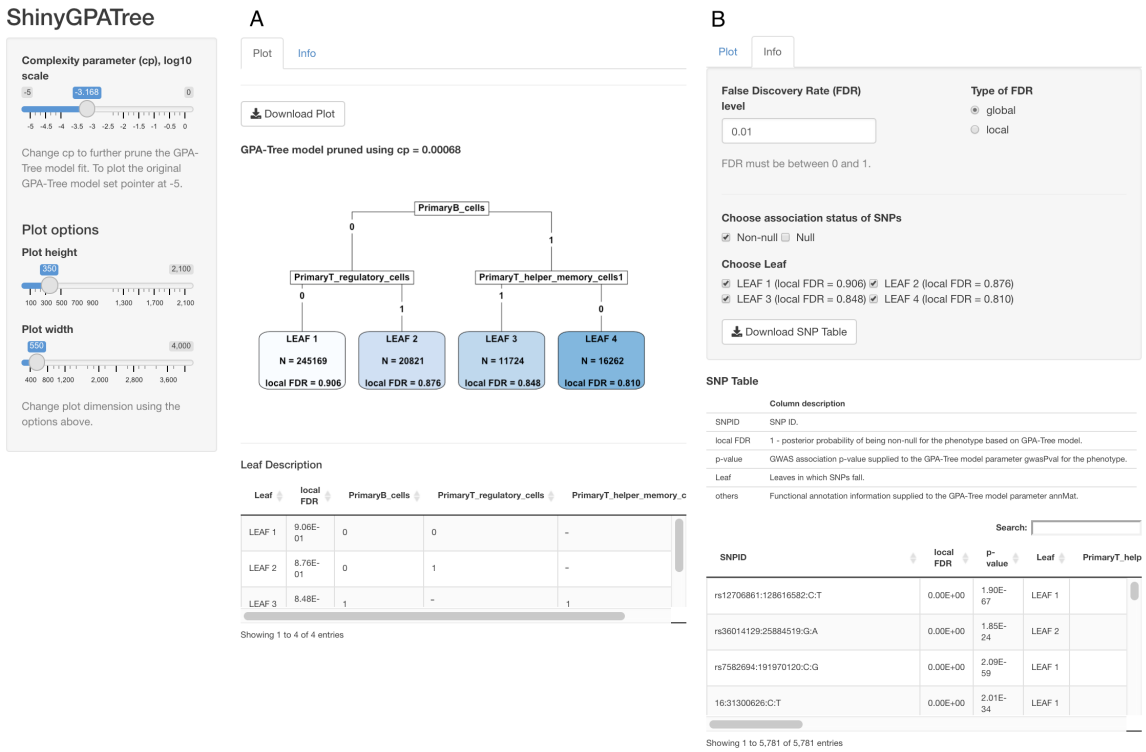


Figure 5.1: Screenshot of the ShinyGPATree app with (A) the 'Plot' tab and (B) the 'Info' tab open.

'Plot' and 'Info', which are explained in detail below.

5.3.1 Plot Tab: Visualization of the GPA-Tree Model

Fig 5.1A shows the layout of the ShinyGPATree app, where the 'Plot' tab opens by default. In the displayed plot, each leaf (terminal node) is characterized by combinations of the functional annotations that are encountered as users move from the root node to the leaf. The summary information is provided for each leaf, including the number of SNPs that satisfy the combination of functional annotations specific to the leaf and the mean local FDR for these SNPs. The summary information displayed in each leaf is automatically updated as the user modifies the `cp` value on the left panel. Users can also improve visualization of the functional annotation tree plot using the `'Plot width'` and `'Plot height'` options

on the left panel. The `'Download Plot'` button on the top allows users to download the functional annotation tree plot as a PNG format file. Finally, a table titled 'Leaf Description' underneath the plot characterizes the functional annotations that are 0 or 1 for SNPs in specific leaves.

5.3.2 Info Tab: Association Mapping and Annotation Selection

The 'Info' tab opens the user interface for association mapping and functional annotation characterization for SNPs as seen in Fig 5.1B. Under this tab, users can find more information on specific SNPs driving the visualization. The top of the panel provides multiple options to control association mapping, including FDR level and FDR type (global vs. local). It also provides options to select which SNPs to display, e.g., choosing SNPs that fall on specific leaves of the GPA-Tree model and/or selecting SNPs with specific association status (non-risk-associated vs. risk-associated SNPs). The 'SNP Table' at the bottom of the 'Info' tab panel shows information about the SNPs that satisfy these options. Each row of the table represents a SNP, where columns include SNP ID, local FDR value, GWAS association *p*-value, the leaf ID in which the SNP is located, and the corresponding complete functional annotation information. The `'Download SNP Table'` button allows users to download the 'SNP Table' as a CSV format file.

5.4 Vignette: Using the GPATree Package and the ShinyGPATree App

Prioritizing GWAS Results and Identifying Risk SNP-Associated Functional Annotation Tree with '**GPATree**' Package

Aastha Khatiwada[1], Bethany J. Wolf[1], Ayse Selen Yilmaz[2], Paula S. Ramos[1,3], Maciej Pietrzak[2], Andrew Lawson[1], Kelly J. Hunt[1], Hang J. Kim[4], Dongjun Chung[2]

[1]Department of Public Health Sciences, Medical University of South Carolina, Charleston, South Carolina, USA
[2]Department of Biomedical Informatics, The Ohio State University, Columbus, Ohio, USA
[3]Department of Medicine, Medical University of South Carolina, Charleston, South Carolina, USA
[4]Divison of Statistics and Data Science, University of Cincinnati, Cincinnati, Ohio, USA

05/15/2021

1 Overview

This vignette provides an introduction to the `GPATree` package. R package `GPATree` implements GPA-Tree, a novel statistical approach to prioritize genome-wide association studies (GWAS) results while simultaneously identifying the combinations of functional annotations associated with risk-associated genetic variants. GPA-Tree integrates GWAS summary statistics and functional annotation data within a unified framework, by combining a decision tree algorithm (CART)(Leo et al. 1984) within the hierarchical model.

The package can be loaded with the command:

```
> library(GPATree)
```

This vignette is organized as follows. Sections 2.1 and 2.2 illustrate the recommended `GPATree-ShinyGPATree` workflow, which provides convenient and interactive genetic data analysis interface. Advanced users might also find Sections 2.3.1 – 2.3.3 useful as the command lines can be used for integrating GPA-Tree as part of the more comprehensive genetic data analysis workflow, for example.

Please feel free to contact Dongjun Chung at `chung.911@osu.edu` for any questions or suggestions regarding the '`GPATree`' package.

2 Workflow

In this vignette, we illustrate the GPA-Tree analysis workflow, using the simulated data provided as the `GPATreeExampleData` in the `GPATree` package. In the simulated data, the number of SNPs is set to $M = 10,000$ and the number of functional annotations is set to $K = 10$. The GWAS association p-values and the binary functional annotation information are stored in `GPATreeExampleData$gwasPval` and `GPATreeExampleData$annMat`, respectively. The number of rows in `GPATreeExampleData$gwasPval`

is assumed to be the same as the number of rows in `GPATreeExampleData$annMat`, where the i-th ($i = 1, ..., M$) row of `gwasPval` and `annMat` correspond to the same SNP.

```
> data(GPATreeExampleData)
> dim(GPATreeExampleData$gwasPval)
[1] 10000     1
> head(GPATreeExampleData$gwasPval)
          P1
SNP_1 0.7454
SNP_2 0.4894
SNP_3 0.6026
SNP_4 0.1496
SNP_5 0.2538
SNP_6 0.3161
> dim(GPATreeExampleData$annMat)
[1] 10000    10
> head(GPATreeExampleData$annMat)
      A1 A2 A3 A4 A5 A6 A7 A8 A9 A10
SNP_1  1  0  0  0  0  1  0  0  0   1
SNP_2  1  0  0  0  0  0  0  0  0   0
SNP_3  1  0  0  0  0  0  0  0  0   1
SNP_4  1  0  0  0  0  0  0  0  0   0
SNP_5  1  0  0  0  1  1  0  0  0   0
SNP_6  1  0  0  0  0  1  0  0  0   0
```

2.1 Fitting the GPA-Tree Model

We can fit the GPA-Tree model using the GWAS association p-values (`GPATreeExampleData$gwasPval`) and functional annotation data (`GPATreeExampleData$annMat`) described above, using the code shown below.

```
> fit.GPATree <- GPATree(gwasPval = GPATreeExampleData$gwasPval,
+                        annMat = GPATreeExampleData$annMat,
+                        initAlpha = 0.1,
+                        cpTry = 0.005)
```

```
> fit.GPATree
Summary: GPATree model results (class: GPATree)
-------------------------------------------------
Data summary:
    Number of GWAS data: 1
    Number of Annotations: 10
    Number of SNPs: 10000
    Alpha estimate: 0.4999
Functional annotation tree description:
       local FDR A4 A2 A1 A3
LEAF 1    0.9849  0  0  -  -
LEAF 2    0.9834  0  1  0  -
LEAF 3    0.0203  0  1  1  -
LEAF 4    0.9850  1  -  -  0
LEAF 5    0.0154  1  -  -  1
-------------------------------------------------
```

2.2 ShinyGPATree

The following command can be used to initialize the ShinyGPATree app. ShinyGPATree allows for interactive and dynamic investigation of disease-risk-associated SNPs and functional annotation trees using R Shiny.

```
> ShinyGPATree(fit.GPATree)
```

Figure 1 shows the layout of the ShinyGPATree app, where the 'Plot' tab opens by default. The summary statistics displayed in the plot are automatically updated as the user input option for cp (in the log10 scale) on the left panel of the screen is modified. Users can also improve visualization of the functional annotation tree plot using the plot width and height options on the left panel. The 'Download Plot' button on the top allows users to download the functional annotation tree plot as a Portable Network Graphics (png) format file. Finally, a table titled 'Leaf Description' underneath the plot characterizes the functional annotations that are 0 or 1 for SNPs in specific leaves.

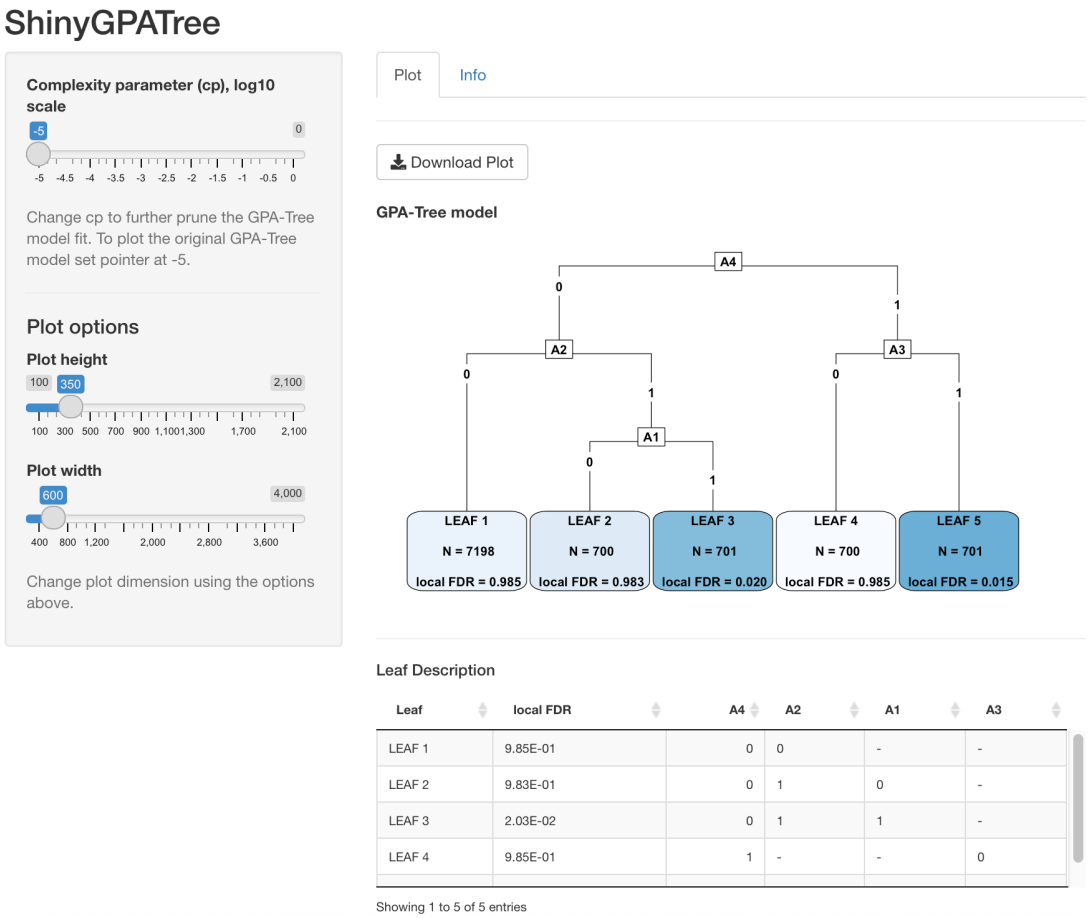


Figure 1: Screenshot of the ShinyGPATree app with the 'Plot' tab open.

As seen in Figure 2, the 'Info' tab in the ShinyGPATree app opens the user interface for association mapping and functional annotation characterization for SNPs. Under this tab, users can find more information on specific SNPs driving the visualization. At the top of the panel, user input options for FDR level and FDR type (global vs. local) are located, followed by options to select SNPs that fall on specific leaves of the

GPA-Tree model or have specific association status (non-risk-associated vs. risk-associated SNPs). The 'SNP Table' at the bottom of the 'Info' tab panel shows information for SNPs that satisfy all user-specified input options. Each row of the table represents a SNP and includes its ID, local FDR value, GWAS association p-value, the leaf in which it is located, and its complete functional annotation information. The 'Download SNP Table' button allows users to download the 'SNP Table' as a Microsoft Excel comma separated values (CSV) file format.

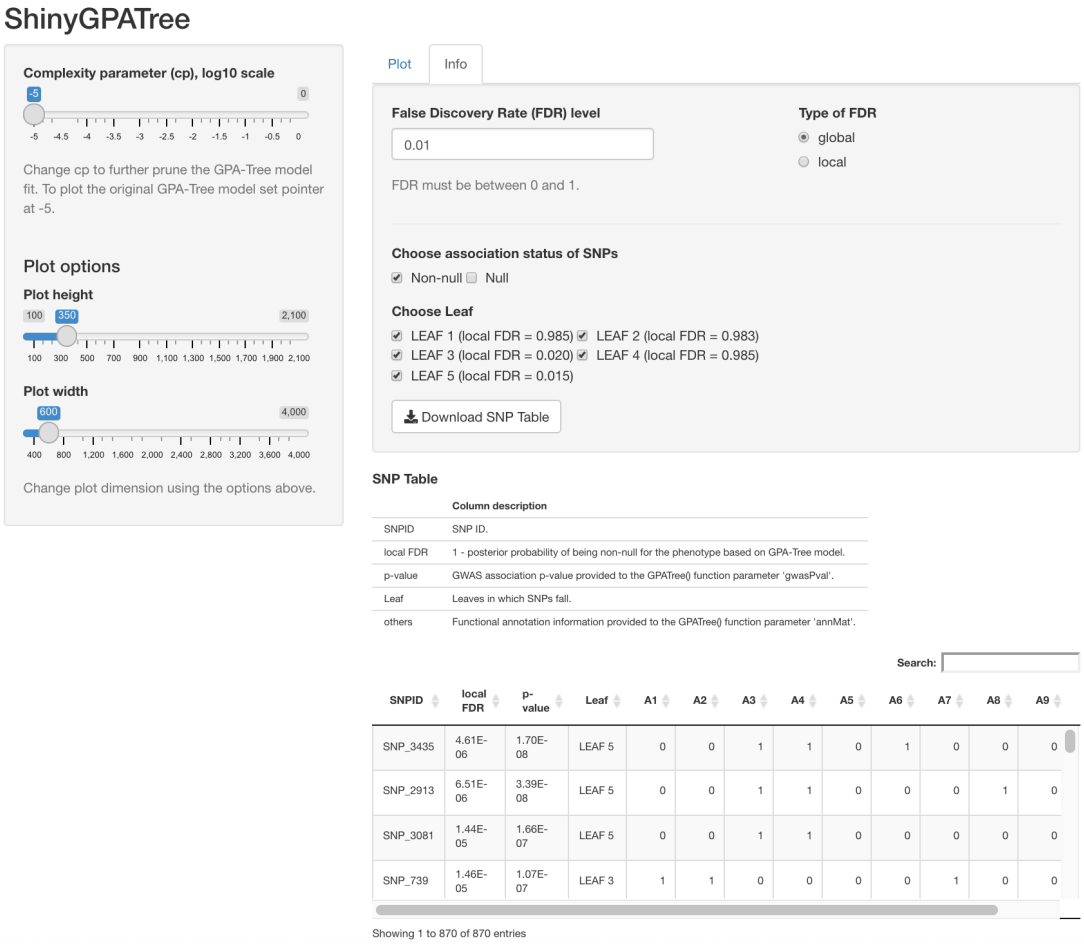


Figure 2: Screenshot of the ShinyGPATree app with the 'Info' tab open.

2.3 Advanced use

2.3.1 Prunning GPA-Tree model fit

The `prune()` function will prune the GPA-Tree model using any cp value between 0 and 1 as shown below.

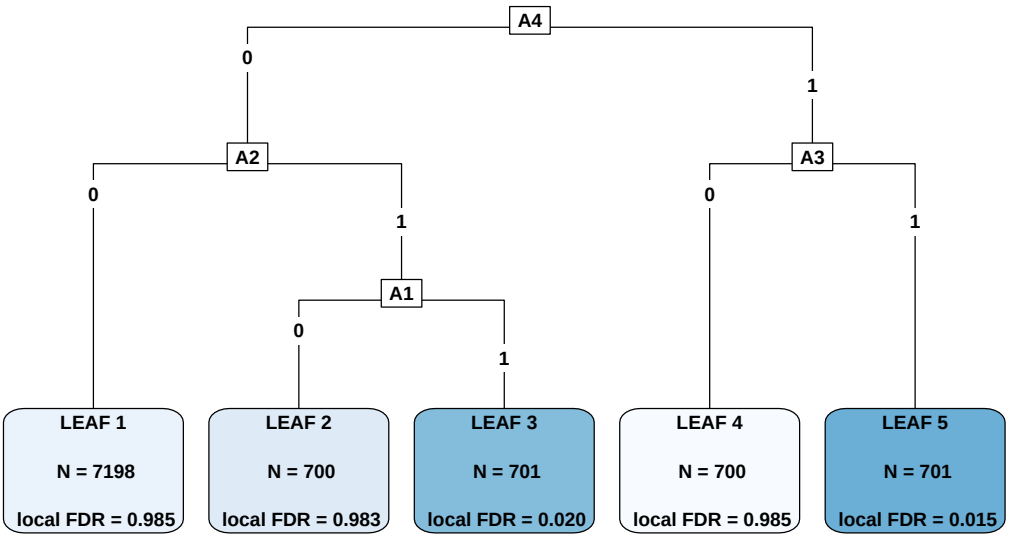
```
> fit.GPATree.pruned <- prune(fit.GPATree, cp = 0.3)
> fit.GPATree.pruned
Summary: GPATree model results (class: GPATree)
-----------------------------------------------
Data summary:
```

```
    Number of GWAS data: 1
    Number of Annotations: 10
    Number of SNPs: 10000
    Alpha estimate: 0.4999
Functional annotation tree description:
       local FDR                    Note
LEAF 1    0.8492 No annotations selected
-------------------------------------------------
```

2.3.2 Functional annotation tree

The `plot()` and `leaf()` functions will plot the GPA-Tree functional annotation tree and provide information about the leaves (terminal nodes) in the tree as shown below.

```
> plot(fit.GPATree)
```




```
> leaf(fit.GPATree)
       local FDR A4 A2 A1 A3
LEAF 1    0.9849  0  0  -  -
LEAF 2    0.9834  0  1  0  -
LEAF 3    0.0203  0  1  1  -
LEAF 4    0.9850  1  -  -  0
LEAF 5    0.0154  1  -  -  1
```

2.3.3 Association mapping

For the fitted GPA-Tree model, we can make inferences about SNPs using the `assoc()` function by: (1) prioritizing risk-associated SNPs, and (2) identifying the leaves of the GPA-Tree model in which the risk-associated SNPs are located. The `assoc()` function returns two columns. The first column contains binary values where 1 indicates that the SNP is associated with the trait and 0 indicates otherwise. The second column provides information regarding the leaf in which the SNP is located in the GPA-Tree plot. The `assoc()` function allows both local (`fdrControl="local"`) and global FDR controls (`fdrControl="global"`) and users can set the threshold to be between 0 and 1 using the 'FDR' argument. For `GPATreeExampleData`, GPA-Tree model identified 870 risk SNPs at the nominal global FDR level

of 0.01. 371 and 499 of the 870 risk-associated SNPs are located in leaf 3 and leaf 5, respectively. The following lines of code can be used to investigate association mapping and functional annotation tree.

```
> assoc.SNP.GPATree <- assoc(fit.GPATree,
+                            FDR = 0.01,
+                            fdrControl="global")
> head(assoc.SNP.GPATree)
      P1   leaf
SNP_1  0 LEAF 1
SNP_2  0 LEAF 1
SNP_3  0 LEAF 1
SNP_4  0 LEAF 1
SNP_5  0 LEAF 1
SNP_6  0 LEAF 1
> table(assoc.SNP.GPATree$P1)

   0    1
9130  870
> table(assoc.SNP.GPATree$leaf)

LEAF 1 LEAF 2 LEAF 3 LEAF 4 LEAF 5
  7198    700    701    700    701
> table(assoc.SNP.GPATree$P1, assoc.SNP.GPATree$leaf)

    LEAF 1 LEAF 2 LEAF 3 LEAF 4 LEAF 5
  0   7198    700    330    700    202
  1      0      0    371      0    499
```

References

Leo, Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. 1984. *Classification and regression trees*. CRC press.

5.5 Conclusions

The R package 'GPATree' and the R Shiny App 'ShinyGPATree' can be used to implement the GPA-Tree and Multi-GPA-Tree approach described in the preceding chapters and is a valuable tool for post-GWAS analysis. The 'GPATree' package also includes an example data and a vignette for step-by-step implementation of the two methods. Availability of the textttShinyGPATree app makes it convenient for users with limited R skills to efficiently implement the described methodologies.

6. Conclusion

6.1 Summary

This dissertation presents two novel statistical methods, GPA-Tree and Multi-GPA-Tree, for integration of genetic data from GWAS with genomic functional annotation data. These two methods fill in an important gap in the current literature for post-GWAS analysis, and are useful tools to identify the combination of functional annotations related to SNPs associated with one of more traits. The GPA-Tree and Multi-GPA-Tree approaches outperform existing statistical approaches in detecting SNPs associated with one or more traits and identifying the true combinations of functional annotations with high accuracy. To facilitate the application of the methods described here, we also developed the `R` package `'GPATree'` and the `R` Shiny App `'ShinyGPATree'`. GPA-Tree and Multi-GPA-Tree are valuable tools that can be utilized to identify genomic regions that are potentially associated with complex traits, and thus, represent an important advancement in the field of post-GWAS analysis.

6.2 Limitations and Extensions

The GPA-Tree and Multi-GPA-Tree approaches have some limitations. Both approaches include binary genomic functional annotation data, and therefore are limited in the type of functional annotation information that is integrated with GWAS data. One avenue for future work could extend the GPA-Tree and Multi-GPA-Tree approaches to include continous as well as other types of genomic annotation data. Future research could also extend GPA-Tree and Multi-GPA-Tree to integrate genetic data from GWAS with other types of 'omics' data like proteomics and metabolomics, among others.

Finally, the current Multi-GPA-Tree approach can be utilized to investigate pleiotropic relationship between two traits only. Expanding Multi-GPA-Tree to efficiently integrate GWAS for more than two traits with functional annotation data to form networks of complex traits that are informed by functional annotation data could provide valuable insights to understand the relationships between different complex traits.

References

[1] Annalisa Buniello, Jacqueline A L MacArthur, Maria Cerezo, Laura W Harris, James Hayhurst, Cinzia Malangone, Aoife McMahon, Joannella Morales, Edward Mountjoy, Elliot Sollis, et al. The NHGRI-EBI GWAS Catalog of published genome-wide association studies, targeted arrays and summary statistics 2019. Nucleic acids research, 47(D1):D1005–D1012, 2019.

[2] Teri A Manolio, Francis S Collins, Nancy J Cox, David B Goldstein, Lucia A Hindorff, David J Hunter, Mark I McCarthy, Erin M Ramos, Lon R Cardon, Aravinda Chakravarti, et al. Finding the missing heritability of complex diseases. Nature, 461(7265):747–753, 2009.

[3] Peter M Visscher, Sarah E Medland, Manuel AR Ferreira, Katherine I Morley, Gu Zhu, Belinda K Cornes, Grant W Montgomery, and Nicholas G Martin. Assumption-free estimation of heritability from genome-wide identity-by-descent sharing between full siblings. PLoS Genet, 2(3):e41, 2006.

[4] Andrew R Wood, Tonu Esko, Jian Yang, Sailaja Vedantam, Tune H Pers, Stefan Gustafsson, Audrey Y Chu, Karol Estrada, Zoltán Kutalik, Najaf Amin, et al. Defining the role of common variation in the genomic and biological architecture of adult human height. Nature genetics, 46(11):1173–1186, 2014.

[5] Jian Yang, Beben Benyamin, Brian P McEvoy, Scott Gordon, Anjali K Henders,

Dale R Nyholt, Pamela A Madden, Andrew C Heath, Nicholas G Martin, Grant W Montgomery, et al. Common SNPs explain a large proportion of the heritability for human height. Nature genetics, 42(7):565–569, 2010.

[6] Alkes L Price, Chris CA Spencer, and Peter Donnelly. Progress and promise in understanding the genetic basis of common diseases. Proceedings of the Royal Society B: Biological Sciences, 282(1821):20151684, 2015.

[7] Hector Giral, Ulf Landmesser, and Adelheid Kratzer. Into the wild: GWAS exploration of non-coding RNAs. Frontiers in cardiovascular medicine, 5:181, 2018.

[8] Andrew J Schork, Wesley K Thompson, Phillip Pham, Ali Torkamani, J Cooper Roddey, Patrick F Sullivan, John R Kelsoe, Michael C O'Donovan, Helena Furberg, Tobacco, Genetics Consortium, Bipolar Disorder Psychiatric Genomics Consortium, Schizophrenia Psychiatric Genomics Consortium, Nicholas J Schork, Ole A Andreassen, and Anders M Dale. All SNPs are not created equal: genome-wide association studies reveal a consistent pattern of enrichment among functionally annotated SNPs. PLoS genetics, 9:e1003449, April 2013.

[9] Jingsi Ming, Mingwei Dai, Mingxuan Cai, Xiang Wan, Jin Liu, and Can Yang. LSMM: a statistical approach to integrating functional annotations with genome-wide association studies. Bioinformatics (Oxford, England), 34:2788–2796, August 2018.

[10] Dongjun Chung, Can Yang, Cong Li, Joel Gelernter, and Hongyu Zhao. GPA: a statistical approach to prioritizing GWAS results by integrating pleiotropy and annotation. PLoS genetics, 10:e1004787, November 2014.

[11] Jingsi Ming, Tao Wang, and Can Yang. LPM: a latent probit model to characterize the relationship among complex traits using summary statistics from multiple GWASs and functional annotations. Bioinformatics, 12 2019. btz947.

[12] Frank W Stearns. One hundred years of pleiotropy: a retrospective. Genetics, 186(3):767–773, 2010.

[13] Dongjun Chung, Hang J Kim, and Hongyu Zhao. graph-GPA: A graphical model for prioritizing GWAS results and investigating pleiotropic architecture. PLoS computational biology, 13:e1005388, February 2017.

[14] Rong W Zablocki, Andrew J Schork, Richard A Levine, Ole A Andreassen, Anders M Dale, and Wesley K Thompson. Covariate-modulated local false discovery rate for genome-wide association studies. Bioinformatics (Oxford, England), 30:2098–2104, August 2014.

[15] Qiongshi Lu, Yiming Hu, Jiehuan Sun, Yuwei Cheng, Kei-Hoi Cheung, and Hongyu Zhao. A statistical framework to predict functional non-coding regions in the human genome through integrated analysis of annotation data. Scientific reports, 5:10576, 2015.

[16] Qiongshi Lu, Xinwei Yao, Yiming Hu, and Hongyu Zhao. GenoWAP: GWAS signal prioritization through integrated analysis of genomic functional annotation. Bioinformatics, 32(4):542–548, 10 2015.

[17] Jin Liu, Xiang Wan, Shuangge Ma, and Can Yang. EPS: an empirical Bayes approach to integrating pleiotropy and tissue-specific information for prioritizing risk genes. Bioinformatics (Oxford, England), 32:1856–1864, June 2016.

[18] The ENCODE (ENCyclopedia of DNA elements) project, author=ENCODE Project Consortium and others. Science, 306(5696):636–640, 2004.

[19] Ole A Andreassen, Wesley K Thompson, Andrew J Schork, Stephan Ripke, Morten Mattingsdal, John R Kelsoe, Kenneth S Kendler, Michael C O'Donovan, Dan Ru-

jescu, Thomas Werge, et al. Improved detection of common variants associated with schizophrenia and bipolar disorder using pleiotropy-informed conditional false discovery rate. PLoS genetics, 9(4), 2013.

[20] Bradley Efron et al. Size, power and false discovery rates. The Annals of Statistics, 35(4):1351–1377, 2007.

[21] Lei Sun, Radu V Craiu, Andrew D Paterson, and Shelley B Bull. Stratified false discovery control for large-scale hypothesis testing with application to genome-wide association studies. Genetic Epidemiology: The Official Publication of the International Genetic Epidemiology Society, 30(6):519–530, 2006.

[22] Kathryn Roeder, Silvi-Alin Bacanu, Larry Wasserman, and B Devlin. Using linkage genome scans to improve power of association in genome scans. The American Journal of Human Genetics, 78(2):243–252, 2006.

[23] Yue Li and Manolis Kellis. Joint bayesian inference of risk variants and tissue-specific epigenomic enrichments across multiple complex human diseases. Nucleic acids research, 44(18):e144–e144, 2016.

[24] Michael A Newton, Amine Noueiry, Deepayan Sarkar, and Paul Ahlquist. Detecting differential gene expression with a semiparametric hierarchical mixture method. Biostatistics, 5(2):155–176, 2004.

[25] Jimmy Z Liu, Allan F Mcrae, Dale R Nyholt, Sarah E Medland, Naomi R Wray, Kevin M Brown, Nicholas K Hayward, Grant W Montgomery, Peter M Visscher, Nicholas G Martin, et al. A versatile gene-based test for genome-wide association studies. The American Journal of Human Genetics, 87(1):139–145, 2010.

[26] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. Classification and regression trees. CRC press, 1984.

[27] JR Quinlan. Induction of decision trees. mach. learn. 1986.

[28] Gordon V Kass. An exploratory technique for investigating large quantities of categorical data. Journal of the Royal Statistical Society: Series C (Applied Statistics), 29(2):119–127, 1980.

[29] Wei-Yin Loh and Yu-Shan Shih. Split selection methods for classification trees. Statistica sinica, pages 815–840, 1997.

[30] Leo Breiman. Random forests. Machine learning, 45(1):5–32, 2001.

[31] Ingo Ruczinski, Charles Kooperberg, and Michael LeBlanc. Logic regression. Journal of Computational and graphical Statistics, 12(3):475–511, 2003.

[32] Terry M Therneau, Elizabeth J Atkinson, et al. An introduction to recursive partitioning using the rpart routines, 1997.

[33] 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), 39(1):1–22, 1977.

[34] Mortaza Jamshidian and Robert I Jennrich. Acceleration of the em algorithm by using quasi-newton methods. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 59(3):569–587, 1997.

[35] Chuanhai Liu, Donald B Rubin, and Ying Nian Wu. Parameter expansion to accelerate em: the px-em algorithm. Biometrika, 85(4):755–770, 1998.

[36] Chuanhai Liu and Donald B Rubin. The ecme algorithm: a simple extension of em and ecm with faster monotone convergence. Biometrika, 81(4):633–648, 1994.

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

[38] Stan Pounds and Stephan W Morris. Estimating the occurrence of false positives and false negatives in microarray studies by approximating and partitioning the empirical distribution of p-values. Bioinformatics, 19(10):1236–1242, 2003.

[39] CC Mok and CS Lau. Pathogenesis of systemic lupus erythematosus. Journal of clinical pathology, 56(7):481–490, 2003.

[40] N Danchenko, JA Satia, and MS Anthony. Epidemiology of systemic lupus erythematosus: a comparison of worldwide disease burden. Lupus, 15(5):308–318, 2006.

[41] Geoffrey Hom, Robert R Graham, Barmak Modrek, Kimberly E Taylor, Ward Ortmann, Sophie Garnier, Annette T Lee, Sharon A Chung, Ricardo C Ferreira, PV Krishna Pant, et al. Association of systemic lupus erythematosus with c8orf13–blk and itgam–itgax. New England Journal of Medicine, 358(9):900–909, 2008.

[42] Carl D Langefeld, Hannah C Ainsworth, Deborah S Cunninghame Graham, Jennifer A Kelly, Mary E Comeau, Miranda C Marion, Timothy D Howard, Paula S Ramos, Jennifer A Croker, David L Morris, et al. Transancestral mapping and genetic load in systemic lupus erythematosus. Nature communications, 8(1):1–18, 2017.

[43] Qiongshi Lu, Ryan Lee Powles, Qian Wang, Beixin Julie He, and Hongyu Zhao. Integrative tissue-specific functional annotations in the human genome provide novel insights on many complex traits and improve signal prioritization in genome wide association studies. PLoS genetics, 12(4):e1005947, 2016.

[44] Qiongshi Lu, Ryan L Powles, Sarah Abdallah, Derek Ou, Qian Wang, Yiming Hu, Yisi Lu, Wei Liu, Boyang Li, Shubhabrata Mukherjee, et al. Systematic tissue-specific functional annotation of the human genome highlights immune-related dna elements for late-onset alzheimer's disease. PLoS genetics, 13(7):e1006933, 2017.

[45] Anshul Kundaje, Wouter Meuleman, Jason Ernst, Misha Bilenky, Angela Yen, Alireza Heravi-Moussavi, Pouya Kheradpour, Zhizhuo Zhang, Jianrong Wang, Michael J Ziller, et al. Integrative analysis of 111 reference human epigenomes. Nature, 518(7539):317–330, 2015.

[46] The GTEx Consortium. The Genotype-Tissue Expression (GTEx) pilot analysis: Multitissue gene regulation in humans. Science, 348(6235):648–660, 2015.

[47] Giuseppe Castellano, Cesira Cafiero, Chiara Divella, Fabio Sallustio, Margherita Gigante, Paola Pontrelli, Giuseppe De Palma, Michele Rossini, Giuseppe Grandaliano, and Loreto Gesualdo. Local synthesis of interferon-alpha in lupus nephritis is associated with type I interferons signature and LMP7 induction in renal tubular epithelial cells. Arthritis research & therapy, 17(1):1–13, 2015.

[48] Keishi Fujio, Yusuke Takeshima, Masahiro Nakano, and Yukiko Iwasaki. Transcriptome and trans-omics analysis of systemic lupus erythematosus. Inflammation and regeneration, 40(1):1–6, 2020.

[49] Bruce I Hoffman and Warren A Katz. The gastrointestinal manifestations of systemic lupus erythematosus: a review of the literature. In Seminars in arthritis and rheumatism, volume 9, pages 237–247. Elsevier, 1980.

[50] Ellen C Ebert and Klaus D Hagspiel. Gastrointestinal and hepatic manifestations of systemic lupus erythematosus. Journal of clinical gastroenterology, 45(5):436–441, 2011.

[51] Denis Comte, Maria P Karampetsou, and George C Tsokos. T cells as a therapeutic target in SLE. Lupus, 24(4-5):351–363, 2015.

[52] Iñaki Sanz and F Eun-Hyung Lee. B cells as therapeutic targets in SLE. Nature Reviews Rheumatology, 6(6):326, 2010.

[53] Mariana J Kaplan. Neutrophils in the pathogenesis and manifestations of SLE. Nature Reviews Rheumatology, 7(12):691–699, 2011.

[54] Patrick Blanco, Vincent Pitard, Jean-François Viallard, Jean-Luc Taupin, Jean-Luc Pellegrin, and Jean-François Moreau. Increase in activated CD8+ T lymphocytes expressing perforin and granzyme B correlates with disease activity in patients with systemic lupus erythematosus. Arthritis & Rheumatism: Official Journal of the American College of Rheumatology, 52(1):201–211, 2005.

[55] Gilberto Filaci, Sabrina Bacilieri, Marco Fravega, Monia Monetti, Paola Contini, Massimo Ghio, Maurizio Setti, Francesco Puppo, and Francesco Indiveri. Impairment of CD8+ T suppressor cell function in patients with active systemic lupus erythematosus. The Journal of Immunology, 166(10):6452–6457, 2001.

[56] Unravelling the complex genetic regulation of immune cells, author=Ramos, Paula S. Nature Reviews Rheumatology, pages 1–2, 2020.

[57] Rachel CY Tam, Alfred LH Lee, Wanling Yang, Chak Sing Lau, and Vera SF Chan. Systemic lupus erythematosus patients exhibit reduced expression of CLEC16A isoforms in peripheral leukocytes. International journal of molecular sciences, 16(7):14428–14440, 2015.

[58] Yong Cui, Yujun Sheng, and Xuejun Zhang. Genetic susceptibility to SLE: recent progress from GWAS. Journal of autoimmunity, 41:25–33, 2013.

[59] Vesela Gateva, Johanna K Sandling, Geoff Hom, Kimberly E Taylor, Sharon A Chung, Xin Sun, Ward Ortmann, Roman Kosoy, Ricardo C Ferreira, Gunnel Nordmark, et al. A large-scale replication study identifies TNIP1, PRDM1, JAZF1, UHRF1BP1 and IL10 as risk loci for systemic lupus erythematosus. Nature genetics, 41(11):1228–1233, 2009.

[60] Christopher J Lessard, Indra Adrianto, John A Ice, Graham B Wiley, Jennifer A Kelly, Stuart B Glenn, Adam J Adler, He Li, Astrid Rasmussen, Adrienne H Williams, et al. Identification of IRF8, TMEM39A, and IKZF3-ZPBP2 as susceptibility loci for systemic lupus erythematosus in a large-scale multiracial replication study. The American Journal of Human Genetics, 90(4):648–660, 2012.

[61] Sotiria Manou-Stathopoulou, Felice Rivellese, Daniele Mauro, Katriona Goldmann, Debasish Pyne, Peter Schafer, Michele Bombardieri, Costantino Pitzalis, and Myles Lewis. 235 IKZF1 and IKZF3 inhibition impairs B cell differentiation and modulates gene expression in systemic lupus erythematosus, 2019.

[62] Xinze Cai, Ying Qiao, Cheng Diao, Xiaoxue Xu, Yang Chen, Shuyan Du, Xudong Liu, Nan Liu, Shuang Yu, Dong Chen, et al. Association between polymorphisms of the IKZF3 gene and systemic lupus erythematosus in a Chinese Han population. PloS one, 9(10):e108661, 2014.

[63] Erik Thorsby and Benedicte A Lie. Hla associated genetic predisposition to autoimmune diseases: Genes involved and possible mechanisms. Transplant immunology, 14(3-4):175–182, 2005.

[64] E Zanelli, FC Breedveld, and RRP De Vries. Hla association with autoimmune disease: a failure to protect? Rheumatology, 39(10):1060–1066, 2000.

[65] Xiaocan Jia, Nian Shi, Yu Feng, Yifan Li, Jiebing Tan, Fei Xu, Wei Wang, Changqing Sun, Hongwen Deng, Yongli Yang, et al. Identification of 67 pleiotropic genes associated with seven autoimmune/autoinflammatory diseases using multivariate statistical analysis. Frontiers in Immunology, 11:30, 2020.

[66] Phil H Lee, Verneri Anttila, Hyejung Won, Yen-Chen A Feng, Jacob Rosenthal, Zhaozhong Zhu, Elliot M Tucker-Drob, Michel G Nivard, Andrew D Grotzinger, Danielle Posthuma, et al. Genomic relationships, novel loci, and pleiotropic mechanisms across eight psychiatric disorders. Cell, 179(7):1469–1482, 2019.

[67] Shanya Sivakumaran, Felix Agakov, Evropi Theodoratou, James G Prendergast, Lina Zgaga, Teri Manolio, Igor Rudan, Paul McKeigue, James F Wilson, and Harry Campbell. Abundant pleiotropy in human complex diseases and traits. The American Journal of Human Genetics, 89(5):607–618, 2011.

[68] Heping Zhang. Classification trees for multiple binary responses. Journal of the American Statistical Association, 93(441):180–193, 1998.

[69] Wei-Yin Loh, Wei Zheng, et al. Regression trees for longitudinal and multiresponse data. The Annals of Applied Statistics, 7(1):495–522, 2013.

[70] Mark Robert Segal. Tree-structured methods for longitudinal data. Journal of the American Statistical Association, 87(418):407–418, 1992.

[71] Heping Zhang and Yuanqing Ye. A tree-based method for modeling a multivariate ordinal response. Statistics and its interface, 1(1):169, 2008.

[72] Seong Keon Lee. On generalized multivariate decision tree by using gee. Computational Statistics & Data Analysis, 49(4):1105–1119, 2005.

[73] Abdessamad Dine, Denis Larocque, and François Bellavance. Multivariate trees for mixed outcomes. Computational Statistics & Data Analysis, 53(11):3795–3804, 2009.

[74] David R Larsen and Paul L Speckman. Multivariate regression trees for analysis of abundance data. Biometrics, 60(2):543–549, 2004.

[75] Glenn De'Ath. Multivariate regression trees: a new technique for modeling species–environment relationships. Ecology, 83(4):1105–1117, 2002.

[76] Yan Yu and Diane Lambert. Fitting trees to functional data, with an application to time-of-day patterns. Journal of Computational and graphical Statistics, 8(4):749–762, 1999.

[77] Wei-Yin Loh. Regression tress with unbiased variable selection and interaction detection. Statistica sinica, pages 361–386, 2002.

[78] Yukinori Okada, Di Wu, Gosia Trynka, Towfique Raj, Chikashi Terao, Katsunori Ikari, Yuta Kochi, Koichiro Ohmura, Akari Suzuki, Shinji Yoshida, et al. Genetics of rheumatoid arthritis contributes to biology and drug discovery. Nature, 506(7488):376–381, 2014.

[79] Katrina M De Lange, Loukas Moutsianas, James C Lee, Christopher A Lamb, Yang Luo, Nicholas A Kennedy, Luke Jostins, Daniel L Rice, Javier Gutierrez-Achury, Sun-Gou Ji, et al. Genome-wide association study implicates immune activation of multiple integrin genes in inflammatory bowel disease. Nature genetics, 49(2):256–261, 2017.