ABSTRACT

PONGPANICH, MONNAT. On the SNP-based and Sequence-based Whole Genome Studies for Complex Traits. (Under the direction of Dr. Jung-Ying Tzeng.)

Quality Control (QC) of the single nucleotide polymorphism (SNPs) used in genome-wide association studies (GWAS) is essential to minimize potential false findings. SNP QC commonly uses expert-guided filters to exclude low-quality SNPs. Expert filters aim to remove SNPs that fall into the extremes of QC variables including Hardy–Weinberg equilibrium, missing proportion (MSP) and minor allele frequency (MAF). However, implementations of these filters require arbitrary thresholds and do not jointly consider all QC features. We propose an algorithm that is based on principal component analysis and clustering analysis to detect low-quality SNPs. The method minimizes the use of arbitrary cutoff values, allows a collective consideration of the QC features and provides conditional thresholds contingent on other QC variables. We compare the performance of our method to expert filters on datasets from the Wellcome Trust Case Control Consortium and the Genetic Association Information Network. Our results suggest that with the same or fewer SNPs excluded, the proposed algorithm tends to give a similar or lower inflation factor of the test statistics ($\lambda$), gives a reduced number of false associations, and retains all true associations.

GWAS methods that collapse information across genetic markers when searching for association signals are gaining momentum in the literature due to their usefulness in marker set analysis and identifying rare variants. Collapsing information can be done at the genotype level, which focuses on the mean of genetic information or the similarity level, which focuses on the variance of genetic information. We seek to understand the strengths and weaknesses of these two collapsing paradigms. Our results show that neither collapsing strategy outperforms the other across all simulated scenarios. The signal-to-noise ratio and the
underlying genetic architecture of the causal variants are the two factors that dominate their performance. Genotype collapsing is more sensitive to the marker set being contaminated by noise loci than similarity collapsing. It performs best when the genetic architecture of the causal variants is not complex. Similarity collapsing is more robust and outperforms genotype collapsing when the genetic architecture of the markerset becomes more sophisticated such as causal loci with various effect sizes or frequencies. In addition, we consider a two-stage analysis that combines the two top-performing methods from different collapsing strategies and find that it is reasonably robust across all simulated scenarios.

RNA-Seq is a promising approach for understanding transcriptomes due to its accuracy, large dynamic range of expression level, and ability to detect novel transcripts. The first step prior to any data analysis is to map reads against a reference genome or transcript set using an alignment tool e.g., TopHat. In many experiments, a non-trivial number of reads are unmapped and excluded from down-stream analyses. To maximize the potential utility of sequenced reads, we propose a method of incorporating these unmapped reads in testing for differentially expressed (DE) genes. Specifically, we use BLAST to align the unmapped reads and assign a weight to each mapped read that reflects the mapping confidence. Gene expression is estimated from the summation of weights of the reads mapped to a gene. To test the general utility of the proposed approach, we construct a simple statistical method and show that using weights improves the power to detect DE genes while still controlling the false discovery rate. In addition, we examine the characteristics of the reads not mapped by TopHat and find that not only the beginning region of the reads, but the tail region of the reads also causes problems with the alignment.
On the SNP-based and Sequence-based Whole Genome Studies for Complex Traits

by
Monnat Pongpanich

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Bioinformatics

Raleigh, North Carolina

2012

APPROVED BY:

_______________________________  ______________________________
Dr. Jung-Ying Tzeng     Dr. Dahlia Nielsen
Chair of Advisory Committee

_______________________________  ______________________________
Dr. Steffen Heber     Dr. Trudy F. C. Mackay
DEDICATION

To my parents, for their unconditional love, encouragement and support.

To Panitan Patrayunyong, for your patience, love and care.
BIOGRAPHY

Monnat Pongpanich was born in Bangkok, Thailand. She received her Bachelor of Science in Computer Science from Chulalongkorn University in March 2005. Shortly after she joined ExxonMobil Limited in April 2005, she received a prestigious scholarship from Anandamahidol Foundation under the Royal Patronage of His Majesty the King of Thailand to pursue a Ph.D. in Bioinformatics. She began her graduate work in Bioinformatics program at North Carolina State University in the summer of 2006. After receiving her doctoral degree, she will return to Thailand to participate in improving the field of Bioinformatics in Thailand.
ACKNOWLEDGMENTS

I would like to express my gratitude to my advisor, Dr. Jung-Ying Tzeng, for her guidance, patience and generous support. I am so thankful having her as my advisor. I can never thank her enough for all she has done for me. I would like to thank Dr. Dahlia Nielsen for her support and giving me the opportunity to work on RNA-Sequencing project. She has always been helpful and understanding. I would also like to thank Dr. Steffen Heber for his time, advice and contribution on my first project. I would also like to thank Dr. Trudy Mackay for her comments and taking the time to serve in my committee. My sincere appreciation is extended to Dr. Patrick Sullivan. I have learned many things from his lab. I would also like to thank Dr. John Pierre Mertz for correcting grammar in my first research paper.

I am also extremely grateful for the financial support from Anandamahidol Foundation. My graduate study would not have begun without Anandamahidol Foundation. I would like to thank Dr. Nalin Nilubol for being my caring mentor. Thank you to Ms. Komsorn Aukayapisudhi for all the administrative help. I also gratefully acknowledge financial support from the National Institutes of Health (grant R01 MH084022-01A1).

Thank you to Sirote Chaiwattanapong for being by my side and making me happy. Thank you to Santi Sanglestsawai for taking good care of me and helping me in ST521 and ST522. Thank you to Plawut Wongwiwat for making me feel that everything will be okay when I am panic. Thank you to Adisri Charoenpanich for being my girlfriend. Thank you to Warakorn Bunkanokwong, Sitthichai Tanthasith, Sirapun Yongwattananunth, Piyada Thongjaem, Jasinee Janjirawatana and Borworn Khuhirun for their friendship. Thank you to
Sasitorn Srisawadi, Manida Swangnetr and Karnjana Sanglimsuwan for settling me down here. Thank you to Ganokon Urkasemsin for always listening to me. Thank you to Worawut Wangwatcharakul and Raywat Tanadkithirun for helping me with statistics.

I also appreciate the support from the faculty and staffs at the Bioinformatics Research Center. Thank you to Chris Smith for all your help from technical support to correcting my English grammar. Thanks to Julibeth Briseno and Siarra Dickey for all the administrative help and Tina Chen for handling all my financial documents.

A special thanks to Noffisat Oyindamola Oki, Gunjan Hariani, Nicholas Hardison, Alexander Griffing, Xin Wang, Zhi Wang, Youfang Liu, Megan Neely, Jihye Kim, Ronglin Che, Chen-yen Lin and Na Cai for all their help and wonderful friendship.

Lastly and most importantly, I would like to thank my parents, Tanong and Gridsana Pongpanich, and Panitan Patrayunyong. You are there for me 24/7. You give me strength, unconditional love and support and encourage me whenever I am down. I am very fortunate to have them in my life.
# TABLE OF CONTENTS

LIST OF TABLES ................................................................................................................... ix  
LIST OF FIGURES .................................................................................................................. x  
Chapter 1: Introduction ............................................................................................................. 1  
  Association Studies ............................................................................................................... 1  
    Obtaining genotypes. ........................................................................................................ 2  
    SNP array. ..................................................................................................................... 2  
    DNA sequencing. .......................................................................................................... 4  
    Quality control. ................................................................................................................. 6  
  Detecting association. ....................................................................................................... 7  
Gene Expression ................................................................................................................... 8  
Overview of the Dissertation .............................................................................................. 11  
References ........................................................................................................................... 13  
Chapter 2: A Quality Control Algorithm for Filtering SNPs in Genome-wide Association  
  Studies ..................................................................................................................................... 19  
  Abstract ............................................................................................................................... 19  
  Introduction ......................................................................................................................... 20  
  Methods ............................................................................................................................... 23  
    The proposed QC algorithm............................................................................................ 23  
    Performance evaluations using real datasets. .................................................................. 26  
    Implementation. .............................................................................................................. 27  
  Results ................................................................................................................................. 28  
  Discussion ........................................................................................................................... 30  
  Table ................................................................................................................................... 34  
  Supplementary tables .......................................................................................................... 35  
  Figures ................................................................................................................................. 43  
  Supplementary Figures ....................................................................................................... 47  
  References ........................................................................................................................... 61  
Chapter 3: On the Aggregation of Multimarker Information for Marker-set and Sequencing  
  Data Analysis: Genotype Collapsing vs. Similarity Collapsing ............................................. 64  
  Abstract ............................................................................................................................... 64  
  Introduction ........................................................................................................................... 65
Chapter 4: Assessing RNA-seq Differential Expression Levels with Low-confidence Mapped Reads

Introduction ......................................................................................................................... 100

Methods............................................................................................................................... 102
LIST OF TABLES

Chapter 2: A Quality Control Algorithm for Filtering SNPs in Genome-wide Association Studies

Table 1. Agreement and disagreement in SNP classifications (good SNPs versus bad SNPs) by the proposed filter and the WTCCC expert filter ........................................ 34
Supplementary Table 1. Proportion of SNPs excluded and λ value of the proposed filter and the WTCCC expert filters ........................................ 35
Supplementary Table 2. Counts of false positives (FP), true positives (TP) of the proposed filter and the WTCCC expert filters ......................... 37
Supplementary Table 3. Agreement and disagreement in SNP classifications (good SNPs vs. bad SNPs) by the proposed filter and the WTCCC expert filter .................................................. 39
Supplementary Table 4. Agreement and disagreement in SNP classifications (good SNPs vs. bad SNPs) by the proposed filter and the WTCCC expert-no-MAF filter (i.e., the expert filter without removing SNPs with MAF < 0.01) ...................................................... 41

Chapter 3: On the Aggregation of Multimarker Information for Marker-set and Sequencing Data Analysis: Genotype Collapsing vs. Similarity Collapsing

Table 1. Minor allele frequency (MAF) of markers resulting from sequencing data from the CHB sample of 1000 Genomes Project ........................................ 88
Table 2. Type I error rates averaged over the 495 possible scenarios for 4 causal markers out of 12 and 500 replicate data sets .................................................. 88

Chapter 4: Assessing RNA-seq Differential Expression Levels with Low-confidence Mapped Reads

Table 1. Percentage of reads mapped by TopHat and BLAST ......................... 118
Table 2. Characteristics of reads unmapped by TopHat but mapped by BLAST categorized into 16 types ................................................................. 118
Table 3. Proportion of reads from each characteristic in human, M. truncatula, mouse and corn ...................................................................................... 119
Table 4. Estimated FDRs based on an average across 100 simulated datasets at five nominal levels ................................................................. 119
Table 5. Power based on an average TPR across 100 simulated datasets at five nominal levels ........................................................................ 120
LIST OF FIGURES

Chapter 2: A Quality Control Algorithm for Filtering SNPs in Genome-wide Association Studies

Figure 1. Projections of SNPs on the two-dimensional PC plane for the WTCCC CAD study ................................................................. 43
Figure 2. Projections of SNPs on the two-dimensional PC plane for the GAIN MDD study ................................................................. 44
Figure 3. Performance comparisons of different filters based on the four criteria defined in the text ................................................................. 45
Figure 4. Characteristics of SNPs with disagreeing classification results between the proposed filter and WTCCC expert filter in CAD ................................................................. 46
Supplementary Figure 1. Projections of SNPs on the 2-dimensional PC plane for BD, CD, HT, RA, T1D, T2D and MDD ................................................................. 48
Supplementary Figure 2. Characteristics of SNPs with disagreed classification results between the proposed filter and WTCCC expert filter for BD, CD, HT, RA, T1D and T2D ................................................................. 54

Chapter 3: On the Aggregation of Multimarker Information for Marker-set and Sequencing Data Analysis: Genotype Collapsing vs. Similarity Collapsing

Figure 1. Power results when casual loci have same effects under 4 causal loci out of 12 markers setting for binary trait ................................................................. 89
Figure 2. Power results when casual loci have different risk effect sizes under 4 causal loci out of 12 markers setting for binary trait ................................................................. 90
Figure 3. Power results for multiple causal alleles in a locus with same effects under 4 loci out of 12 markers setting for binary trait ................................................................. 91
Figure 4. Power results when casual loci have same effects under 4 causal loci out of 4 markers setting for binary trait ................................................................. 92
Figure 5. Power results when casual loci have same effects under 2 loci out of 12 markers setting for binary trait ................................................................. 92
Figure 6. Power results when casual loci have same effects under 2 loci out of 30 markers setting for binary trait ................................................................. 93
Figure 7. Power results when casual loci have different risk effect sizes under 2 loci out of 30 markers setting for binary trait ................................................................. 94
Figure 8. Power results when casual loci have same effects under 2 loci out of 12 markers setting for quantitative trait ................................................................. 95
Chapter 4: Assessing RNA-seq Differential Expression Levels with Low-confidence Mapped Reads

Figure 1. An illustration of how differentially expressed genes are created. .......... 121
Figure 2. Characteristics of reads mapped by BLAST in human ......................... 122
Figure 3. Characteristics of reads mapped by BLAST in M. truncatula ............... 123
Figure 4. Characteristics of reads mapped by BLAST in mouse ..................... 124
Figure 5. Characteristics of reads mapped by BLAST in mouse ..................... 125
Figure 6. Testing for differential expression using weighted counts................... 126
Figure 7. Testing for differential expression using unweighted counts.............. 127
Figure 8. log_2 of foldchange between weighted and unweighted counts ......... 128
Figure 9. log_2 of mean normalized expression between weighted and unweighted counts. .......................................................... 129

Appendix A

Figure A1. The sorted fourth NN distance for CAD. ................................. 139
Chapter 1

Introduction

Association Studies

A genetic association study aims to find associations between genetic polymorphism(s) and phenotypes (Lunetta, 2008). In family-based association studies, the transmission frequency for each allele from heterozygous parent to affected offspring is estimated and is 50% under the null. In contrast, disease associated alleles will be transmitted in excess to the affected individual. The challenge in this design is its sensitivity to genotyping error which can distort transmission proportions between parents and offspring (Pearson & Manolio, 2008). In case-control association studies, the allele or genotypes frequencies between cases and controls are compared (Hirschhorn & Daly, 2005). If the variant does not relate to the disease, the alleles/genotypes frequencies between cases and controls should not significantly differ. This design is susceptible to population substructure (McCarthy et al., 2008) and additional procedures must be used to avoid spurious association, such as computing association statistics that take subpopulation clusters into account (Pritchard et al., 2000), adjusting for the inflation factor caused by the substructure (Devlin & Roeder, 1999), and using major principal components as covariates to correct for stratification (Price et al., 2006). Prior to next generation sequencing technologies, association studies used linkage disequilibrium (LD) information; that is, researchers genotyped markers that are in LD with the disease locus (Dick, 2008). These genotyped single nucleotide polymorphisms (SNPs) on a SNP chip capture most common variants
(Neale & Purcell, 2008) and association studies have identified many common alleles associated with disease (Donnelly, 2008; Hardy & Singleton, 2009). Nevertheless, proportion of heritability explained by the loci found in association studies is small. For example, in a GWAS study of a highly heritable trait, height, 180 loci have been identified, yet proportion of heritability explained is only 10% (Allen et al., 2010). Much of the speculation is that rare variants contribute to the missing heritability (Bogardus, 2009; Manolio et al., 2009). The individually trivial contributions of rare variants to the proportion of the variance in a trait, and their low frequency, make it difficult to detect rare variants by association studies based on the use of tag SNPs (Bodmer & Bonilla, 2008). Direct sequencing of candidate gene or the entire genome allows us to identify rare variants. In the near future, sequencing technologies will become more cost-effective. This starts to shift the first wave of association studies to re-sequencing association studies or DNA sequence-based association studies. The voluminosity and new type of data (sequence reads) having a wide spectrum of allele frequencies coming from sequencing poses many challenges: (a) storing, accessing, and handling data; (b) mapping short reads to a reference genome or de novo assembly of the genome; (c) calling true variants; (d) analyzing common and low frequency loci that may exhibit allelic heterogeneity with respect to the trait (Pop & Salzberg, 2008; Day-Williams & Zeggini, 2011).

**Obtaining genotypes.**

**SNP array.** In high throughput SNP arrays, millions of SNPs are genotyped in a single assay. The genotyping process for the Affymetrix platform can be briefly summarized as follows: total genomic DNA is digested with restriction enzymes resulting in various sizes
of fragments, adaptors are then ligated to fragments regardless of size, fragments with selected size are amplified by PCR, and the amplified target is fragmented, denatured, end labeled and hybridized to the array. The array contains the probes for each of the two SNP alleles (A, B) (Affymetrix, Inc., 2009).

For each SNP and each individual, probe hybridization intensities are summarized for each allele. That is, a pair of coordinates - an “A” signal and a “B” signal, which correspond to the quantities of the “A” and “B” alleles, are obtained (Affymetrix Inc., 2007). Automated procedures have been developed to call genotypes. Typically, they employ information across samples at each SNP. This usually results in three clouds of fluorescent signals where the cluster of points with high “A” (“B”) signal but close to zero “B” (“A”) signal is expected to correspond to AA (BB) genotype samples and the cluster of points with similar “A” and “B” signal is expected to correspond to AB genotype samples (Teo, 2008). The genotype clusters are used to call genotypes for each sample. Generally, the genotype calling algorithms calculate probabilities of belonging to each of the three clusters given the observed data point across all samples and assign the most likely genotype to a sample if the corresponding probability passes the threshold (e.g., WTCCC, 2007; Affymetrix Inc., 2007).

Under ideal conditions, obtaining genotypes should be straightforward. However, various factors can affect genotyping accuracy. For some SNPs, the three clusters overlap and thus cause ambiguity in calling the genotype, resulting in assigning “missing call” to individuals in the overlap regions (Anney et al., 2008). In some SNPs, homozygote-heterozygote miscall occurs, yet with high-confidence assignments of genotypes (Teo et al., 2007). Allelic dropout could also yield a false homozygote in heterozygous individuals.
(Pompanon et al., 2005). The confidence score threshold is another factor that provides a tradeoff between call rate and accuracy (Teo, 2008). In addition, monomorphic SNPs can be erroneously assessed as polymorphic (Pettersson et al., 2008). The calling accuracy can also depend on batch composition (combining or separating cases and controls) and batch size (the number of samples called at the same time) especially for SNPs with low minor allele frequency (MAF) as calling more people at the same time will help in forming the minor allele homozygote cluster (Teo, 2008; Miclaus et al., 2010).

**DNA sequencing.** Sequencing technologies differ in the specific protocols they combine. Major steps are template preparation, sequencing and imaging, and genome alignment and assembly. There are two types of templates: clonally amplified and single-molecule. In clonally amplified templates, the two most common methods, emulsion PCR (one DNA molecule per bead; beads are deposited into wells) and solid-phase amplification (resulting in hundreds of millions of molecular clusters; one DNA molecule per cluster), can be used to amplify the DNA molecule. Amplification is necessary because the imaging systems cannot detect single fluorescent events (Metzker, 2010). In single-molecule templates, the template molecules are immobilized on a solid support either via primer immobilization, template immobilization (Helicos BioSciences Corporation, 2008, 2010) or polymerase immobilization (Pacific Biosciences, 2009).

Sequencing strategies can be divided into 4 types: cyclic reversible termination (CRT), sequencing by ligation (SBL), single nucleotide addition (SNA; pyrosequencing) and real time sequencing. The key steps are incorporating dye-labelled nucleotides/probes that complement the template base, and fluorescent imaging (Metzker, 2010). In four-color CRT,
a series of color images from each cluster are translated into a sequence as one color represents one base (Illumina, Inc., 2010). In one-color CRT, a series of light on and off images from each cluster are translated into a sequence as nucleotides are flowed sequentially in a specific order (Helicos BioSciences Corporation, 2010). In SBL, two-base-encoded probes are used to interrogate two bases at a time. Five primer rounds are performed (shifting primer one position to the left between each round). Color calls from five ligation rounds are compiled and decoded into DNA sequence (Life Technologies Corporation, 2011). In SNA, the emitted light is recorded as a series of peaks called a flowgram, which reveal the DNA sequence (Roche Diagnostics Gmb, 2006). In real time sequencing, incorporation of nucleotides is continuous and sequential bursts of light are recorded (Pacific Biosciences, 2009).

The output from sequencing is (potentially) millions of reads and associated quality scores. The next step is to map these short reads to the reference genome or de novo assembly. The alignment step is important, as wrongly mapped reads will propagate error into downstream analysis (Day-Williams & Zeggini, 2011). There are numerous challenges in the alignment step: reads may have sequencing errors, reads might map to multiple locations in the genome, reads may be from repetitive regions or might correspond to a region that does not exist in the reference genome (Marguerat&Bährler, 2010; Metzker, 2010; Day-Williams &Zeggini, 2011). Once reads are mapped, the variants are called. However, complications in variant calling include alignment artifacts, PCR artifacts, and location of the variants. Reads containing small indels can be misaligned and lead to false positive SNPs with high confidence. If a single piece of DNA gives rise to non-independent reads and the
PCR reaction introduces reads supporting both the correct and erroneous bases, confident but spurious SNP calls could be generated. The ends of the reads can lead to false SNP calls as error rates are higher for the bases toward the end of the reads (Day-Williams & Zeggini, 2011).

**Quality control.** In order that association studies have reliable and reproducible results, good quality control is essential. Genotyping error can alter the magnitude of the difference between allele frequencies in cases and controls, thus affecting the power (Gordon & Ott, 2001). Differential genotyping error rates between cases and controls can result in type I error (Moskvina et al., 2006; Clayton et al., 2005; Morris & Cardon, 2007).

Quality control is usually done in two levels: sample quality control and SNP quality control. For sample quality control, samples with high rates of missingness or heterozygosity are excluded since high a missing proportion implies hybridization problems, and excess heterozygosity is a sign of sample contamination. Related samples inferred through identity-by-state (IBS) are removed. Duplicate samples and samples with external discordance with genotype or phenotype data are also removed (Teo 2008; WTCCC, 2007). For SNP quality control, expert filters are applied to SNPs in order to remove SNPs that fall into the extremes of QC variables including Hardy-Weinberg equilibrium (HWE), missing proportion (MSP), and minor allele frequency (MAF). The rationale is clear: extreme deviation from HWE is typically used to identify gross genotyping error (Teo et al., 2007); a high MSP indicates poor genotype probe performance and low genotyping accuracy (Neale and Purcell, 2008; WTCCC, 2007); SNPs with low MAF are more prone to error, as fewer samples would be within a genotype cluster and most clustering based calling algorithms do not perform well
with rare alleles (Neale and Purcell, 2008; Teo, 2008). After removing low quality SNPs, the presence of population structure in the data needs to be assessed and accounted for if present in case-control studies. Once quality control is performed, researchers can then move on to association evaluation.

**Detecting association.** Methods for detecting marker association can be single-marker or multi-marker. In single-marker analysis, individual markers are tested for association. In multi-marker analysis, markers are grouped together into a marker-set and association tests are performed for each marker-set. Marker sets can be formed based on functional annotations of the genomic regions e.g., grouping variants in regulatory regions or conserved regions, or grouping markers in the same gene or pathway, functionality of variants e.g., grouping the coding variants, non-synonymous variants, or simply based on moving window or haplotype block (Bansal et al., 2010).

Although single-marker analysis has been successful in identifying many associated variants, marker-set analysis has an advantage over single-marker analysis (Wu et al., 2010). In single-marker analysis, the genome-wide significance threshold can be difficult to reach due to the large number of tests. Marker-set analysis alleviates multiple testing problems (Wu et al., 2010, Tzeng et al., 2011). Marker-set analysis also enhances power to detect variants as it accumulates small effects across multiple loci for common SNPs and enriches the signal in the case of rare variants, which are hard to detect via single-marker analysis. Marker-set analysis can also consider the joint effect of multiple markers that are potentially interacting (Tzeng et al., 2011). In high throughput SNP arrays, marker-set analysis could capture the
true effect more effectively as it is plausible that causal SNPs are in LD with many of markers in the set (Wu et al., 2010).

In a marker-set analysis, marker information in the set is aggregated and assessed for the collective effect of the markers on the phenotype. The information among markers can be collapsed at the genotype level or similarity level. Genotype collapsing methods focus on the mean level of the genetic information, while similarity-collapsing methods focus on the variance level of the genetic information. At the genotype level, information can be collapsed by calculating a weighted sum of the genotypes across all markers e.g., Li & Leal (2008), Madsen & Browning (2009), Han & Pan (2010), Price et al. (2010). At the similarity level, information can be collapsed by quantifying the genetic similarity across all markers for each pair of unrelated individuals e.g., Wessel & Schork (2006), Tzeng et al. (2009, 2011), Mukhopadhyay et al. (2010), Wu et al. (2010, 2011). The two collapsing paradigms have their advantages and disadvantages. Researchers have to use their judgment to pick an appropriate method for their analysis.

**Gene Expression**

Genotypes give rise to phenotypes through gene expression. Previously, microarray allowed researchers to study gene expression for thousands of genes at once. However, the introduction of massively parallel sequencing platforms has revolutionized the field. RNA-Seq is one of the applications of next generation sequencing. Microarrays are a hybridization-based approach and thus, suffer from background and cross-hybridization issues (Costa et al., 2010). In contrast, RNA-Seq, based on the principles of DNA sequencing, has many advantages over microarrays. RNA-seq is not limited to detecting known transcripts, and can
therefore detect novel transcribed regions. It does not have an upper limit for quantification, thus allows a large dynamic range of expression level (Wang et al., 2009). RNA-Seq results show high levels of reproducibility (Marioni et al., 2008). Resolution of RNA-Seq data is a single base pair, consequently allowing researchers to generate annotation at single-base resolution e.g., transcript boundary (Wang et al., 2009; Costa et al., 2010). In addition, it allows detection of antisense transcripts, SNPs and mutations (Wilhelm & Landry, 2009).

In microarrays, RNA is extracted and labeled with a fluorescent dye. In two channel microarrays, the treatment and control mRNA are labeled with two different dyes, mixed then hybridized on the same array. The array is scanned to acquire two images: treatment and control sample. In single channel microarrays, treatment and control are labeled with the same dye but hybridized on different arrays (Tarca et al., 2006). The image is analyzed to obtain raw intensity data for every spot. The extracted data from the image is preprocessed to filter poor quality data and then normalized. Finally, downstream analysis can be performed (Leung & Cavalieri, 2003).

In RNA-Seq, RNAs are isolated from cells and ribosomal RNAs (rRNA) are removed. A library of cDNA fragments is prepared through either RNA fragmentation or DNA fragmentation. Sequencing adaptors are added to each cDNA fragment. The remaining steps are as described in DNA sequencing: clonal amplification of cDNA fragments, sequencing and imaging the amplified fragments. The final output is millions of short sequence reads (Costa et al., 2010). The first step of any data analysis is either to map sequence reads to a reference genome or to perform de novo assembly. Challenges in RNA-Seq alignment are the same as in DNA-Seq alignment. In addition, reads that contain post-
transcriptionally modified e.g. alternative splicing (read spans exon junction),
polyadenylation and RNA editing, add more challenges (Marguerat & Bähler, 2010). At the end of the alignment step, reads are either uniquely mapped, mapped to multiple locations (multi-match reads) or are not mapped (Costa et al., 2010). Most studies limit their attention to only uniquely mapped reads e.g., Nagalakshmi et al. (2008), Marioni et al. (2008). However, there have been some efforts to address multi-match reads e.g., Cloonan et al. (2008), Faulkner et al. (2008), Mortazavi et al. (2008), Li et al. (2010).

To detect differentially expressed (DE) genes, gene expression must be quantified. Read counts are used to estimate gene expression and need to be properly normalized for two reasons: (1) longer transcripts generate more reads (2) number of fragment across samples fluctuate for each run. Therefore, read counts are usually normalized by the gene’s length and total number of mapped reads in the sample. The expression of a gene is defined as the sum of the expression level from all of its isoforms. Two schemes are most commonly used to obtain gene expression: (a) exon intersection method, which uses only constitutive exons and (b) exon union method, which uses exons from all isoforms (Garber et al., 2011). Regardless of whether the exon intersection or union method is used, two approaches can be used to obtain the expression scores. One approach is to count the number of reads that cover each nucleotide position within the included exons and then sum those counts. The other approach is to sum the number of reads that fall within the included exons. Then the sum is divided by the length of the feature and total number of reads mapped (Wilhelm & Landry, 2009).

Once gene expression is obtained, statistical analysis of differential expression can be performed. Many tests were developed for detecting DE genes in microarrays. However, in
microarrays, the output is fluorescent intensity, which can be effectively modeled as a continuous variable, whereas the count-based nature of RNA-Seq data needs to be modeled as a discrete variable. Therefore, new methodologies have been developed to handle RNA-Seq data (Costa et al., 2010). In early attempts, the Poisson distribution was considered (Marioni et al., 2008), but it was found that the variation observed in RNA-Seq data are larger than that predicted by Poisson (which is referred to as the overdispersion problem). More recent progress has focused on tackling this overdispersion problem based on modeling counts using the negative binomial or the beta binomial, such as Robinson et al., (2010), Anders and Huber, (2010), Hardcastle and Kelly, (2010), Di et al., (2011) and Zhou et al., (2011).

**Overview of the Dissertation**

Genotype data quality is of paramount importance for association studies, thus performing quality control is a preliminary step before testing for an association. The arbitrary threshold and non-simultaneous consideration of all QC features has motivated us to develop an unsupervised filter based on the rationale of expert filters. In chapter 2, we propose an algorithm that is based on principal component analysis and clustering analysis to identify low-quality SNPs. The method minimizes the use of arbitrary cutoff values, allows a collective consideration of the QC features, and provides conditional thresholds contingent on other QC variables (e.g., different missing proportion thresholds for different minor allele frequencies).

In detecting association, collapsing methods are gaining momentum due to their usefulness in marker-set analysis and rare variant detection. Understanding the strengths and
weakness of the two collapsing paradigms: genotype collapsing vs. similarity collapsing, helps researchers select a suitable approach for their analysis. In chapter 3, we seek to understand the advantages and drawbacks of the two collapsing paradigms over a wide range of plausible scenarios. We investigate the implications of employing these different collapsing strategies when performing multi-marker association analysis.

In RNA-Seq analysis, read mapping is the first step prior to any analysis. Depending on the transcriptome and read length, the fraction of unmappable reads varies. Based on analyses of a dataset from the Nielsen Lab, and many other experiments, this fraction is not trivial. Discarding these unmapped reads is wasteful in terms of cost and data loss. This has motivated us to minimize the amount of discarded data. In chapter 4, we incorporated reads that an aligner tailored to short reads fails to map to detect DE genes by using a more general alignment algorithm i.e., BLAST, to align these unmapped reads and assign a weight to each mapped read as a function of aligned chunk length, gaps and mismatches.
References


Chapter 2

A Quality Control Algorithm for Filtering SNPs in Genome-wide Association Studies¹

Abstract

Motivation. The quality control (QC) filtering of single nucleotide polymorphisms (SNPs) is an important step in genome-wide association studies to minimize potential false findings. SNP QC commonly uses expert-guided filters based on QC variables [e.g. Hardy–Weinberg equilibrium, missing proportion (MSP) and minor allele frequency (MAF)] to remove SNPs with insufficient genotyping quality. The rationale of the expert filters is sensible and concrete, but its implementation requires arbitrary thresholds and does not jointly consider all QC features.

Results. We propose an algorithm that is based on principal component analysis and clustering analysis to identify low-quality SNPs. The method minimizes the use of arbitrary cutoff values, allows a collective consideration of the QC features and provides conditional thresholds contingent on other QC variables (e.g. different MSP thresholds for different MAFs). We apply our method to the seven studies from the Wellcome Trust Case Control Consortium and the major depressive disorder study from the Genetic Association Information Network. We measured the performance of our method compared to the expert filters based on the following criteria: (i) percentage of SNPs excluded due to low quality;

(ii) inflation factor of the test statistics ($\lambda$); (iii) number of false associations found in the filtered dataset; and (iv) number of true associations missed in the filtered dataset. The results suggest that with the same or fewer SNPs excluded, the proposed algorithm tends to give a similar or lower value of $\lambda$, a reduced number of false associations, and retains all true associations.

**Introduction**

Genome-wide association studies (GWAS) have been shown to be a powerful and successful strategy in identifying genetic variants that influence common and complex diseases. Prior to the advent of GWAS in 2005, there were only a few robust, replicated associations identified, such as NOD2 for Crohn’s disease (CD; Hugot et al., 2001), and PPARG, KCNJ11 and CAPN10 for Type 2 diabetes (T2D) mellitus (McCarthy, 2004). With GWAS, there are now more than 30 loci identified for CD and almost 20 loci for T2D (Barrett et al., 2008; Zeggini et al., 2008). To date (April 2010), there are over 545 published studies reporting genetic variants responsible for more than 340 common diseases (Hindorff et al., 2009; http://www.genome.gov/gwastudies).

GWAS interrogate millions of single nucleotide polymorphisms (SNPs), and the large-scale genotype calling (which translates probe hybridization intensities into actual genotypes) must fully resort to automated clustering procedures (Plagnol et al., 2007; Teo, 2008). Ideally, SNP genotyping yields three clusters of signals, and a subject’s genotype can be assigned according to cluster membership (Ziegler et al., 2008). In reality, the clustering methods are unavoidably prone to error, as imperfect clusters of signal clouds can arise due to experimental variation, DNA quality and nonspecific hybridization issues (Anney et al.,...
2008; Clayton et al., 2005). Common error patterns include missing calls for SNPs with overlapping genotype clusters (Anney et al., 2008), homozygote–heterozygote miscalls (Teo et al., 2007), false homozygote calls in heterozygous individuals due to allelic dropout (Pompanon et al., 2005), and erroneous assessment of monomorphic SNPs as polymorphic (Pettersson et al., 2008).

SNP quality control (QC) is commonly safeguarded by ‘supervised’ (i.e. expert-guided) filters to exclude low-quality SNPs. The ‘supervised’ expert filters aim to remove SNPs that fall into the extremes of QC variables including Hardy–Weinberg equilibrium (HWE), missing proportion (MSP) and minor allele frequency (MAF). The rationale is clear: extreme deviation from HWE is typically used to identify gross genotyping error (Teo et al., 2007); a high MSP indicates poor genotype probe performance and low genotyping accuracy (Neale and Purcell, 2008; WTCCC, 2007); SNPs with low MAF are more prone to error, as fewer samples would be within a genotype cluster and most clustering-based calling algorithms do not perform well with rare alleles (Neale and Purcell, 2008; Teo, 2008). However, the implementation of expert filters tends to require arbitrary determination of cutoff values for the QC variables, and does not jointly consider all QC features. For example, in GAIN studies, the minimum SNP genotyping quality standards are HWE P-value > 0.00033, average MSP < 3%, MSP maximum < 10% and quality score and MAF greater than a pre-determined minimum level, which varies from study to study (GAIN Collaborative Research Group, 2007). For GWAS conducted by the Wellcome Trust Case Control Consortium (WTCCC), the criteria for retaining a SNP are: HWE P-value ≥ 5.7×10⁻⁷, MSP ≤ 5% if MAF ≥ 5%, MSP ≤ 1% if MAF < 5% and MAF > 0.01 (WTCCC,
Sladek et al. (2007) included SNPs when the HWE P-value > 0.001, MSP ≤ 5% and MAF > 0.01. Unoki et al. (2008) included SNPs when the HWE P-value ≥ 10^{-6} and MSP ≤ 10%.

Statistical methods have also been developed to identify, assess or incorporate genotyping errors in association studies (Gordon et al., 2001; Gordon and Ott, 2001; Hao and Wang, 2004; Rice and Holmans, 2003). Recently, Plagnol et al. (2007) introduced a calling algorithm to minimize the biases that occur when case and control DNA samples are from different sources and processed in different laboratories. Miyagawa et al. (2008) investigated appropriate cutoff values for each of the QC variables (MSP, MAF, HWE and confidence score of genotype calls) by dividing and reshuffling healthy samples. Teo et al. (2008) assessed the stability of the assigned genotypes by introducing white noise to the fluorescent intensities of each subject and evaluating the agreement between the calls made with the noise-perturbed and original intensities. Finally, for family-based studies, Fardo et al. (2009) developed a transmission test to measure the genotyping error rate of each proband.

In this work, we take the rationale of the expert filters and propose an ‘unsupervised’ (i.e. algorithm-determined) filter to detect low-quality SNPs. Like ‘supervised’ expert filters, our filter also aims to identify QC outliers. Furthermore, our filter automates the QC threshold determination based on all QC features, and gives conditional cutoffs contingent on the values of other QC variables (e.g. different MSP thresholds for different MAFs). The algorithm is based on the premise that the majority of SNPs have sufficient genotyping quality with QC variable values in certain directions (e.g. low MSP and non-low MAF). SNPs with QC values deviating from the majority are considered outliers and are then
labeled as problematic SNPs. The algorithm first performs principal component analysis (PCA) on the QC variables with an aim to separate good SNPs from problematic SNPs on a two-dimensional plane. It then uses Density Based Spatial Clustering of Applications with Noise (DBSCAN; Ester et al., 1996) to identify the boundaries of good SNPs and define QC thresholds. We evaluate the performance of the proposed algorithm and demonstrate its utility using the seven WTCCC datasets (WTCCC, 2007) and the major depressive disorder (MDD) dataset from Genetic Association Information Network (GAIN) studies (Sullivan et al., 2009).

**Methods**

**The proposed QC algorithm.** We begin with a SNP dataset that has been cleaned using the criteria of quality score and HWE. That is, if an SNP does not reach the desired level of quality score, the genotyping result is specified as ‘missing’. In addition, SNPs that show severe HWE violations in the control group (i.e. the P-value of the HWE test is smaller than a threshold appropriate for multiple testing) are excluded from the dataset. Quality score and HWE are used to pre-clean the dataset because they have clear definitions for good SNPs. While deviation from HWE has relatively low sensitivity in testing for genotyping error (Cox and Kraft, 2006), it has been shown that severe genotyping errors often do cause extreme HWE deviations (Teo et al., 2007).

With this pre-cleaned dataset, our algorithm aims to identify good-quality SNPs based on two basic QC features, MSP and MAF. Specifically, we consider six QC variables including MSP in case samples (denoted by MSPcs), MSP in control samples (MSPcn), MSP in the combined case control samples (MSPall), logMAF in the combined samples
(logMAFall), the ratio of MSPcs to MAFcs and the ratio of MSPcn to MAFcn. MAF is considered on the log scale to ensure a more careful QC examination with a low MAF than a high MAF. The interaction term between MSP and MAF is designed to allow for an adaptive MSP threshold with different MAF values, and is defined as MSP×(1/MAF). The adaptive thresholds ensure that SNPs with smaller MAF have a more stringent MSP threshold, as missing genotypes have a larger impact on frequency when occurring in low MAF than in high MAF. We use the ratio rather than the product of MSP and MAF, so that different low-quality features (e.g. high MSP and low MAF) will be retained rather than being cancelled out in the interaction terms. A higher interaction value indicates lower quality.

There are two main steps involved in the proposed QC algorithm: (i) using PCA to separate the good SNPs from the bad SNPs based on the QC features on a two-dimensional plane; and (ii) using DBSCAN (Ester et al., 1996) to identify the boundaries of good SNPs on the plane. The PCA is performed on the six QC variables to separate good SNPs from bad ones on the plane of the first two principal components (PC1 versus PC2), which usually account for about >80% of the variation in the original QC variables. The use of PCA facilitates the task of modeling all of these QC variables that can be highly correlated. It also projects good SNPs into a concentrated corner on the plane and spreads out bad SNPs in opposite directions along the axes of the original QC variables (e.g. see the biplots shown in Figures 1a and 2a, and the expert SNP classification in Figures 1b and 2b). Different studies may result in different patterns of PC biplots, but the key common feature is that good SNPs are pushed toward a certain corner that represents desirable QC values: low MSP, high MAF and low MSP to MAF ratio.
Given the PCA plots, we use DBSCAN (Ester et al., 1996) to define the boundaries of the good SNPs. DBSCAN is a density-based clustering algorithm, it performs efficiently on large-scale datasets, and most importantly, it can find clusters of arbitrary shape. Given a data space, it defines regions of high-density points as clusters and classifies regions of low-density points as noises (i.e. a noise is a point that does not belong to any clusters). DBSCAN requires that for each point in a cluster, there are at least a minimum number, $K$, of points in the neighborhood of a given radius $r$ of the target point. Ester et al. recommended setting $K$ to four, and to determine $r$ from the data via the following steps. First, calculate the distance of each target point to its $K$-th nearest point. Next, plot the sorted $K$-th nearest neighbor (NN) distance (which is referred to as the sorted $K$-th NN graph). Finally, set $r$ to the Y-axis value where a sharp jump occurs. We follow the suggestion of using $K = 4$. Instead of eyeballing the value for $r$ as suggested by the original work, we solve for $r$ by fitting a change point model as described in the Appendix A1. The $r$ value determined by the change-point method should be viewed as an initial value, and should be further fine tuned until certain criteria are fulfilled. For example, tune $r$ until the resulting ‘good’ SNPs yield a desirable $\lambda$ value (i.e. the inflation factor of the test statistics; Devlin and Roeder, 1999), until maximum MSP is smaller than a desirable level, or until a certain percentage of SNPs are removed. With the suitable $r$ value, we then use the largest cluster identified by DBSCAN to define the boundaries for good SNPs (e.g. see the borders of the blue area in Figures 1c and 2c). The boundaries represent meaningful thresholds with respect to the original QC variables. In the final output of the algorithm, an SNP is labeled as ‘good’ if (i) it is in the largest cluster, or (ii) it passes the identified thresholds of all QC features even if it is not in the largest cluster.
Criterion (ii) ensures the monotonicity of the thresholding. That is, any SNPs located in the ‘good SNP corner’ (i.e. high logMAF, low MSP and low MSP/MAF) will be included even if they are not dense enough to be included in a cluster.

**Performance evaluations using real datasets.** The performance of the proposed QC algorithm was evaluated using the seven GWAS studies conducted by WTCCC, including bipolar disorder (BD), coronary artery disease (CAD), CD, hypertension (HT), rheumatoid arthritis (RA), Type 1 diabetes (T1D), and T2D (WTCCC, 2007). In addition, we also assess the algorithm using the MDD dataset from GAIN studies (Sullivan et al., 2009). In each WTCCC GWAS, there were 490,032 SNPs genotyped on chromosomes 1 to 22 from 2000 cases and 3000 common controls, which included 1500 from the 1958 British Birth Cohort (58C) and another 1500 from blood donors recruited by UK Blood Services. We excluded unreliable individuals as defined in the original studies: poor sample call rate (<97%), extreme overall heterozygosity (>30% or <23%) and high genome-wide IBD values (>0.86), and obtained on average 1887 cases and 2974 controls. We then removed those SNPs with HWE P-value < 5.7×10^{-7} (WTCCC, 2007) and were left with 474,657 SNPs for the QC evaluations. The MDD study contained 556,131 SNPs genotyped on chromosomes 1 to 22 from 1738 MDD cases and 1802 controls. In this dataset, all unreliable samples (e.g. poor sample call rate, extreme heterozygosity, high relatedness and ancestral outliers) have been excluded using the steps described in Sullivan et al. (2009). We removed SNPs with HWE P-value < 5.7×10^{-7} and performed the QC analysis on the remaining 526,740 SNPs.

The results of the proposed QC algorithm are compared with the expert filter defined in WTCCC (WTCCC, 2007), which removed SNPs with MSP > 5% if MAF ≥ 5%, MSP >
1% if MAF < 5% or SNPs with MAF ≤ 1%. The performances are assessed based on the following four criteria: (i) percentage of SNPs excluded due to low quality; (ii) inflation factor of the substructure-adjusted test statistics $\lambda$; (iii) number of false associations found in the filtered dataset [referred to as false positives (FP)]; and (iv) number of true associations missed in the filtered dataset [referred to as true positives (TP)]. For (ii), the inflation factor $\lambda$ is calculated as the median of the observed test statistics of association divided by the median of $\chi^2(1)$ distribution (i.e. 0.456) (Devlin and Roeder, 1999). For the WTCCC datasets, the association statistics were calculated using a stratified Cochran–Mantel–Haenszel test in PLINK (Purcell et al., 2007) to adjust for population substructure. For the MDD dataset, the trend test statistics were used because the samples are ancestrally homogeneous (Sullivan et al. 2009). For (iii), an FP is defined as a significant signal found in the data analyses but not confirmed in the literature (i.e. neither in PubMed database nor the published GWAS catalog at www.genome.gov/gwastudies). For (iv), a TP is defined as a significant signal found in the data analyses and also confirmed in the literature (either in PubMed or the published GWAS catalog). A P-value threshold of $5 \times 10^{-7}$ (following the WTCCC paper) is used to define significance. However, because there were no literature-confirmed signals that survived the $5 \times 10^{-7}$ threshold for BD, HT and MDD, we used a less stringent threshold of $10^{-5}$ for the P-value in our analysis for these three diseases. A threshold of $10^{-5}$ is considered to be a moderate association in the WTCCC studies (WTCCC, 2007).

**Implementation.** We have implemented a command-line-based software package to perform the proposed QC algorithm. The software runs PCA using the ‘prcomp’ function in
R and runs DBSCAN using C++ code written by us for speed improvement. The software and instructions are available for download from the corresponding author’s website.

**Results**

Figure 3 shows the results of our method and the WTCCC expert filter based on the four criteria. The specific numerical results are given in the Supplementary Tables 1 and 2. To illustrate, we report the results using the r value obtained by the change-point model for all eight diseases regardless of whether a further fine-tuning of r was carried out. Overall, the algorithm with change-point r removed from 2.8% to 14.6% fewer SNPs than the expert filters, and yet had either smaller or comparable \(\lambda\) values, contained fewer or comparable FPs, and retained the same TPs (which were all the TPs in the genotyped SNPs). The maximum MSP retained in the datasets ranged from 4.44% to 5.55% for the WTCCC datasets and was 37.63% for MDD.

Carefully examining Supplementary Tables 1 and 2, we saw that there were three diseases where the performance with the initial change-point r value was not as good as expert filters in some of the criteria: BD (having a larger \(\lambda=1.123\) than the 1.122 of the expert filter), RA (having a larger \(\lambda=1.083\) than 1.052 of expert and including four more FPs) and MDD (having two more FPs than expert). Using BD as an example, with the change-point r value, our filter removed 13.2% of the SNPs (versus 18.6% of expert), and the resulting ‘good’ SNPs had a maximum MSP of 4.91%, a not small \(\lambda\) value of 1.123 (versus 1.122 of expert), 19 FPs (versus 27 of expert, out of 912 unfiltered FPs) and retained all 6 TPs (same as expert). In practice, the algorithm should be continued by adjusting r until a desirable \(\lambda\) is reached. However, for comparison purposes, we instead fine-tuned r to a smaller value so
that the two filters removed about the same proportion of SNPs. With a similar removal rate (18.2% versus 18.6% of expert), our algorithm gave a slightly smaller $\lambda$ (1.114 versus 1.122) and kept fewer FPs (18 versus 27). The results of the TPs remained unchanged.

In RA, we removed 9.8% of SNPs (versus 18.7% of expert), which resulted in a maximum MSP of 5.55%, a $\lambda$ of 1.083 (versus 1.052 of expert), 211 FPs (versus 207 of expert, out of 817 unfiltered FPs), and the same number of TPs as the expert filter (6 out of 6 unfiltered TPs). When we removed about the same proportion of SNPs as the expert filter (18.5% versus 18.7%), the remaining good SNPs yielded a slightly smaller $\lambda$ (1.048 versus 1.052), contained 16 fewer FPs (191 versus 207) and the same number of TPs (6). In MDD, with the change-point $r$, the algorithm removed much fewer SNPs (5.26% versus 19.89%), yielded comparable $\lambda$ (1.043 versus 1.044), but kept two more FPs (6 versus 4 out of 6 unfiltered FPs) compared to the expert filter. Because the resulting maximum MSP was too large (i.e. 37.6%) when we used the change-point $r$, we tuned $r$ by decreasing its value till the maximum MSP was <10% (i.e. 9.7%). The $\lambda$ and FPs became 1.044 and 4, respectively, which were the same as the expert filters. The $\lambda$ and FPs stayed unchanged when we continued tuning $r$ until we had removed the same proportion of SNPs as the expert filter.

We also categorized the SNPs into four groups according to whether they were included (i.e. labeled as ‘good SNP’) or excluded (i.e. labeled as ‘bad SNP’) by our filter and the expert filter (Table 1). For all of the diseases, our algorithm and the expert filter have around 80% agreement in inclusion and around 12% agreement in exclusion on average. The majority of the disagreement between the two filters can be attributed to the use of adaptive thresholds in our filter. To illustrate, we show the boundary of good SNPs from our filter, the
expert filter, and the overlay of the two on a two-dimensional PC plane using CAD (Figure 1d) and MDD (Figure 2d) (see Supplementary Figure 1 for other diseases). Instead of the step-like boundary of good SNPs in the expert filter, our filter gives a smoother boundary of good SNPs. Figure 4 further illustrates the disagreements on the axes of MSP versus MAF (instead of PC1 versus PC2) for CAD (see Supplementary Figure 2 for other diseases). The yellow dots represent SNPs that are labeled as ‘good’ by our algorithm but ‘bad’ by expert filters. One group of yellow dots occurred in the extremely low MAF and low MSP range, indicating that our algorithm would keep SNPs of MAF < 0.01 when their MSPs were extremely low. In contrast, the red dots represent the SNPs that are labeled as ‘bad’ by our algorithm but ‘good’ by expert filters. The big red area on the right side indicates that our algorithm gives a more stringent MSP criterion for good SNPs than the expert filter (i.e. MSP < 5%): our criteria ranged from MSP < 2% to MSP < 4%, depending on the MAF. Lastly, the two red and yellow triangles in the upper middle area show the impact of the ‘smoother’ threshold of our algorithm: it has a more stringent MSP threshold when MAF is 0.01–0.025, and a less stringent threshold when MAF is 0.025–0.05.

**Discussion**

Ensuring the quality of genotype data is essential for drawing accurate and replicable conclusions (Donnelly, 2008). In this work, we have introduced a QC algorithm to identify SNPs with low QC features using criteria determined through PCA and DBSCAN. The proposed filter is in essence an ‘unsupervised’ (i.e. algorithm determined) version of the ‘supervised’ expert filter to classify SNPs, and it aims to account for multiple QC variables, provides adaptive cutoff values and automates thresholding decisions. Specifically, we use
PCA to jointly model the potentially highly correlated QC variables, and use DBSCAN to identify the borders of good-SNP clusters that have arbitrary shapes. The boundary of the good-SNP cluster can be translated directly to meaningful thresholds for the original QC variables. The proposed algorithm retains the rationale of the expert filter to identify QC outliers, avoids arbitrary decisions on cutoff values and gives contingent MSP thresholds for different MAF values. The data applications show that with the same or fewer SNPs discarded due to bad quality, the proposed algorithm has comparable or better performance than the expert filter for all diseases.

The underlying rationale of our algorithm is that the majority of genotyped markers have sufficient genotyping quality, and hence low-quality SNPs can be treated as outliers and be identified by looking for SNPs with distinct QC features. To facilitate the implementation of the idea, we use PCA on the original QC variables. PCA consolidates the information from the many correlated QC variables, and projects SNPs onto a two-dimensional PC plane where good SNPs clump together into a corner of desirable QC values, whereas bad SNPs fan out in all directions. It is expected that the PC biplot may differ from one study to another: in our exploration, we have seen different patterns in the biplots for WTCCC datasets and for the MDD dataset. Yet all biplots have good SNPs lumped into a corner that corresponds to good QC features.

We wish to point out that when using the proposed algorithm, it is important to monitor the features of the retained good SNPs and tune the neighborhood radius $r$ to safeguard the basic QC criteria. This is because the thresholds for outliers are determined relative to the majority of the data. The tuning becomes particularly crucial if a big
proportion of data points are of low quality. For example, in the MDD dataset, there were about 11.4% of SNPs with MSP > 10%, and our algorithm with the initial change-point r kept SNPs with MSP up to almost 38%. Tuning of r was thus continued until the maximum MSP dropped to < 10%. In practice, the smaller r is, the more stringent the QC criteria for ‘good’ SNPs will be, as a smaller r makes it harder to form a cluster in DBSCAN. We suggest starting with a value of r determined by fitting a change-point model to the sorted fourth nearest distances, and then further to adjust r until the specific goal is reached, so as to assure the λ value, the maximum MSP, or the percentage of SNPs removed within reasonable ranges. In our explorations, we found that the change-point r often suggests a reasonable value (e.g. in CAD, CD, HT, T1D and T2D) or is at least a good upper bound (e.g. in BD, RA and MDD, judging by the resulting λ values or the retained maximum MSP). Given the change-point r, one can reduce its value if a more stringent filter is needed, and increase its value if one wishes to remove only extreme outlier SNPs.

When selecting which QC variables to include in the algorithm, we intend to avoid using MAFcs and MAFcn because they may obfuscate the true associations. For the rest of the QC variables, it is possible to make other choices for inclusion/exclusion, e.g. to exclude MSPall from the proposed QC variable set (i.e. to use five variables), or to include MSPall/MAFall to the proposed QC variable set (i.e. to use seven variables). While we expected that the performance would not change much, we carried out sensitivity analyses to evaluate the impact of using different QC variables in the proposed algorithm. The results are given in Supplementary Tables 1, 2 and 3. For comparability, we tuned the r values so that each analysis removed a similar proportion of SNPs to the original 6-variable analysis. As
expected, the 7- and 5-variable analyses performed very similarly to the proposed 6-variable analyses, indicating the robustness of the proposed filter to the different choices of QC variables.
### Table

Table 1. Agreement and disagreement in SNP classifications (good SNPs versus bad SNPs) by the proposed filter and the WTCCC expert filter

<table>
<thead>
<tr>
<th>Diseases</th>
<th>r</th>
<th>Agreed</th>
<th>Disagreed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Both good(^a) (%)</td>
<td>Both bad(^b) (%)</td>
</tr>
<tr>
<td>BD</td>
<td>0.0154 (^e)</td>
<td>80.33</td>
<td>12.10</td>
</tr>
<tr>
<td></td>
<td>0.0110 (^f)</td>
<td>78.38</td>
<td>15.10</td>
</tr>
<tr>
<td>CAD</td>
<td>0.0125 (^e)</td>
<td>79.32</td>
<td>13.92</td>
</tr>
<tr>
<td>CD</td>
<td>0.0134 (^e)</td>
<td>79.58</td>
<td>12.57</td>
</tr>
<tr>
<td>HT</td>
<td>0.0132 (^e)</td>
<td>79.69</td>
<td>13.37</td>
</tr>
<tr>
<td>RA</td>
<td>0.0179 (^e)</td>
<td>80.76</td>
<td>9.27</td>
</tr>
<tr>
<td></td>
<td>0.0101 (^f)</td>
<td>77.05</td>
<td>14.23</td>
</tr>
<tr>
<td>T1D</td>
<td>0.0133 (^e)</td>
<td>79.62</td>
<td>12.77</td>
</tr>
<tr>
<td>T2D</td>
<td>0.0136 (^e)</td>
<td>79.93</td>
<td>13.24</td>
</tr>
<tr>
<td>MDD</td>
<td>0.0259 (^e)</td>
<td>80.11</td>
<td>5.26</td>
</tr>
<tr>
<td></td>
<td>0.0072 (^f)</td>
<td>78.95</td>
<td>17.26</td>
</tr>
<tr>
<td></td>
<td>0.0063 (^f)</td>
<td>78.53</td>
<td>18.24</td>
</tr>
</tbody>
</table>

\(^a\)The % of SNPs classified as ‘good’ by both filters.
\(^b\)The % of SNPs classified as ‘bad’ by both filters.
\(^c\)The % of SNPs classified as ‘good’ by the proposed filter but ‘bad’ by the expert filter.
\(^d\)The % of SNPs classified as ‘bad’ by the proposed filter but ‘good’ by the expert filter.
\(^e\)The r value is determined by the change-point model.
\(^f\)In these analyses, the values of r are chosen to make the proportion of SNPs removed by the proposed filter comparable to that of the WTCCC expert filters.
\(^g\)The r value is chosen to make the maximum MSP in the resulting good SNPs < 10%.
Supplementary Table 1. Proportion of SNPs excluded and $\lambda$ value of the proposed filter and the WTCCC expert filters

<table>
<thead>
<tr>
<th>Diseases</th>
<th>$r$</th>
<th>Percentage of SNPs excluded</th>
<th>$\lambda$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Unfiltered</td>
<td>Our algorithm</td>
</tr>
<tr>
<td>BD1$^6$</td>
<td>0.0154$^1$</td>
<td>13.20</td>
<td>18.57</td>
</tr>
<tr>
<td>BD2$^6$</td>
<td>0.0110$^2$</td>
<td>18.16</td>
<td>18.57</td>
</tr>
<tr>
<td>BD$^5$</td>
<td>0.0100$^2$</td>
<td>18.61</td>
<td>18.57</td>
</tr>
<tr>
<td>BD$^7$</td>
<td>0.0105$^2$</td>
<td>18.26</td>
<td>18.57</td>
</tr>
<tr>
<td>CAD$^4$</td>
<td>0.0125$^1$</td>
<td>15.90</td>
<td>18.69</td>
</tr>
<tr>
<td>CAD$^3$</td>
<td>0.0118$^4$</td>
<td>15.84</td>
<td>18.69</td>
</tr>
<tr>
<td>CAD$^7$</td>
<td>0.0130$^4$</td>
<td>16.02</td>
<td>18.69</td>
</tr>
<tr>
<td>CD$^4$</td>
<td>0.0134$^1$</td>
<td>14.70</td>
<td>18.29</td>
</tr>
<tr>
<td>CD$^3$</td>
<td>0.0120$^4$</td>
<td>14.77</td>
<td>18.29</td>
</tr>
<tr>
<td>CD$^7$</td>
<td>0.0133$^4$</td>
<td>15.09</td>
<td>18.29</td>
</tr>
<tr>
<td>HT$^6$</td>
<td>0.0132$^1$</td>
<td>15.11</td>
<td>18.57</td>
</tr>
<tr>
<td>HT$^3$</td>
<td>0.0118$^4$</td>
<td>15.26</td>
<td>18.57</td>
</tr>
<tr>
<td>HT$^7$</td>
<td>0.0135$^4$</td>
<td>15.01</td>
<td>18.57</td>
</tr>
<tr>
<td>RA1$^6$</td>
<td>0.0179$^1$</td>
<td>9.79</td>
<td>18.71</td>
</tr>
<tr>
<td>RA2$^6$</td>
<td>0.0101$^2$</td>
<td>18.47</td>
<td>18.71</td>
</tr>
<tr>
<td>RA$^5$</td>
<td>0.0095$^2$</td>
<td>18.99</td>
<td>18.71</td>
</tr>
<tr>
<td>RA$^7$</td>
<td>0.0100$^2$</td>
<td>18.43</td>
<td>18.71</td>
</tr>
<tr>
<td>T1D$^6$</td>
<td>0.0133$^1$</td>
<td>14.58</td>
<td>18.57</td>
</tr>
<tr>
<td>T1D$^5$</td>
<td>0.0120$^4$</td>
<td>14.70</td>
<td>18.57</td>
</tr>
<tr>
<td>T1D$^7$</td>
<td>0.0140$^4$</td>
<td>14.68</td>
<td>18.57</td>
</tr>
</tbody>
</table>
Supplementary Table 1. Continued

<table>
<thead>
<tr>
<th>Diseases</th>
<th>$r$</th>
<th>Percentage of SNPs excluded</th>
<th>$\lambda$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Our algorithm</td>
<td>WTCCC expert filters</td>
</tr>
<tr>
<td>----------</td>
<td>-----</td>
<td>----------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>T2D&lt;sup&gt;6&lt;/sup&gt;</td>
<td>0.0136&lt;sup&gt;1&lt;/sup&gt;</td>
<td>14.74</td>
<td>18.58</td>
</tr>
<tr>
<td>T2D&lt;sup&gt;3&lt;/sup&gt;</td>
<td>0.0120&lt;sup&gt;4&lt;/sup&gt;</td>
<td>14.89</td>
<td>18.58</td>
</tr>
<tr>
<td>T2D&lt;sup&gt;7&lt;/sup&gt;</td>
<td>0.0140&lt;sup&gt;4&lt;/sup&gt;</td>
<td>14.53</td>
<td>18.58</td>
</tr>
<tr>
<td>MDD&lt;sup&gt;1&lt;/sup&gt;</td>
<td>0.0259&lt;sup&gt;1&lt;/sup&gt;</td>
<td>5.26</td>
<td>19.89</td>
</tr>
<tr>
<td>MDD&lt;sup&gt;6&lt;/sup&gt;</td>
<td>0.0072&lt;sup&gt;3&lt;/sup&gt;</td>
<td>18.42</td>
<td>19.89</td>
</tr>
<tr>
<td>MDD&lt;sup&gt;2&lt;/sup&gt;</td>
<td>0.0063&lt;sup&gt;2&lt;/sup&gt;</td>
<td>19.82</td>
<td>19.89</td>
</tr>
<tr>
<td>MDD&lt;sup&gt;5&lt;/sup&gt;</td>
<td>0.0038&lt;sup&gt;2&lt;/sup&gt;</td>
<td>19.73</td>
<td>19.89</td>
</tr>
<tr>
<td>MDD&lt;sup&gt;7&lt;/sup&gt;</td>
<td>0.0051&lt;sup&gt;2&lt;/sup&gt;</td>
<td>19.83</td>
<td>19.89</td>
</tr>
</tbody>
</table>

<sup>1</sup>WTCCC expert filters without removing SNPs with MAF < 0.01.
<sup>2</sup>The $r$ value is determined by the change-point model.
<sup>3</sup>In these analyses, the values of $r$ are chosen to make the proportion of SNPs removed by the proposed filter compatible to that of the WTCCC expert filter.
<sup>4</sup>The $r$ value is chosen to make the maximum MSP in the resulting good SNPs is < 10%.
<sup>5</sup>The $r$ values are chosen to make the proportion of SNPs removed by the 5-variable and 7-variable filters compatible to the 6-variable proposed filter.
<sup>6</sup>The proposed algorithm, based on 5 QC variables, including MSPcs, MSPcn, logMAFall, MSPcs/MAFcs, and MSPcn/MAFcn.
<sup>7</sup>The proposed algorithm based on 6 QC variables, including MSPcs, MSPcn, logMAFall, MSPcs/MAFcs, and MSPcn/MAFcn. Figure 3 are plotted using these results.
<sup></sup>The proposed algorithm based on 7 QC variables, including MSPcs, MSPcn, logMAFall, MSPcs/MAFcs, MSPcn/MAFcn, and MSPall/MAFall.
Supplementary Table 2. Counts of false positives (FP), true positives (TP) of the proposed filter and the WTCCC expert filters

<table>
<thead>
<tr>
<th>Diseases</th>
<th>$r$</th>
<th>FP</th>
<th>TP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unfiltered</td>
<td>Our</td>
<td>WTCCC</td>
<td>WTCCC</td>
</tr>
<tr>
<td></td>
<td>algorithm</td>
<td>expert filters</td>
<td>expert-no-MAF filters</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BD1$^a$</td>
<td>0.0154$^i$</td>
<td>912</td>
<td>19</td>
</tr>
<tr>
<td>BD2$^a$</td>
<td>0.0110$^i$</td>
<td>912</td>
<td>18</td>
</tr>
<tr>
<td>BD$^5$</td>
<td>0.0100$^i$</td>
<td>912</td>
<td>17</td>
</tr>
<tr>
<td>BD7$^7$</td>
<td>0.0105$^i$</td>
<td>912</td>
<td>17</td>
</tr>
<tr>
<td>CAD$^9$</td>
<td>0.0125$^i$</td>
<td>778</td>
<td>20</td>
</tr>
<tr>
<td>CAD$^3$</td>
<td>0.0118$^i$</td>
<td>778</td>
<td>20</td>
</tr>
<tr>
<td>CAD7$^7$</td>
<td>0.0130$^i$</td>
<td>778</td>
<td>21</td>
</tr>
<tr>
<td>CD$^6$</td>
<td>0.0134$^i$</td>
<td>229</td>
<td>41</td>
</tr>
<tr>
<td>CD$^5$</td>
<td>0.0120$^i$</td>
<td>229</td>
<td>52</td>
</tr>
<tr>
<td>CD7$^7$</td>
<td>0.0133$^i$</td>
<td>229</td>
<td>52</td>
</tr>
<tr>
<td>HT$^a$</td>
<td>0.0132$^i$</td>
<td>1232</td>
<td>7</td>
</tr>
<tr>
<td>HT$^5$</td>
<td>0.0118$^i$</td>
<td>1232</td>
<td>7</td>
</tr>
<tr>
<td>HT7$^7$</td>
<td>0.0135$^i$</td>
<td>1232</td>
<td>8</td>
</tr>
<tr>
<td>RA1$^b$</td>
<td>0.0179$^i$</td>
<td>817</td>
<td>211</td>
</tr>
<tr>
<td>RA2$^b$</td>
<td>0.0101$^i$</td>
<td>817</td>
<td>191</td>
</tr>
<tr>
<td>RA$^5$</td>
<td>0.0095$^i$</td>
<td>817</td>
<td>194</td>
</tr>
<tr>
<td>RA7$^7$</td>
<td>0.0100$^i$</td>
<td>817</td>
<td>193</td>
</tr>
<tr>
<td>T1D$^6$</td>
<td>0.0133$^i$</td>
<td>1291</td>
<td>546</td>
</tr>
<tr>
<td>T1D$^5$</td>
<td>0.0120$^i$</td>
<td>1291</td>
<td>548</td>
</tr>
<tr>
<td>T1D7$^7$</td>
<td>0.0140$^i$</td>
<td>1291</td>
<td>549</td>
</tr>
</tbody>
</table>
Supplementary Table 2. Continued

<table>
<thead>
<tr>
<th>Diseases</th>
<th></th>
<th>Unfiltered</th>
<th>Our algorithm</th>
<th>WTCCC expert filters</th>
<th>WTCCC expert-no-MAF filters</th>
<th>Unfiltered</th>
<th>Our algorithm</th>
<th>WTCCC expert filters</th>
<th>WTCCC expert-no-MAF filters</th>
</tr>
</thead>
<tbody>
<tr>
<td>T2D(^6)</td>
<td>0.0136(^1)</td>
<td>363</td>
<td>8</td>
<td>9</td>
<td>13</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>T2D(^3)</td>
<td>0.0120(^1)</td>
<td>363</td>
<td>8</td>
<td>9</td>
<td>13</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>T2D(^7)</td>
<td>0.0140(^1)</td>
<td>363</td>
<td>8</td>
<td>9</td>
<td>13</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>MDD1(^6)</td>
<td>0.0259(^1)</td>
<td>6</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>MDD3(^6)</td>
<td>0.0072(^3)</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>MDD2(^4)</td>
<td>0.0063(^2)</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>MDD(^5)</td>
<td>0.0038(^3)</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>MDD(^7)</td>
<td>0.0051(^2)</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

\(^0\)WTCCC expert filters without removing SNPs with MAF < 0.01.
\(^1\)The r value is determined by the change-point model.
\(^2\)In these analyses, the values of r are chosen to make the proportion of SNPs removed by the proposed filter compatible to that of the WTCCC expert filter.
\(^3\)The r value is chosen to make the maximum MSP in the resulting good SNPs is < 10%.
\(^4\)The r values are chosen to make the proportion of SNPs removed by the 5-variable and 7-variable filters compatible to the 6-variable proposed filter.
\(^5\)The proposed algorithm, based on 5 QC variables, including MSPcs, MSPcn, logMAFall, MSPcs/MAFcs, and MSPcn/MAFcn.
\(^6\)The proposed algorithm based on 6 QC variables, including MSPcs, MSPcn, MSPall, logMAFall, MSPcs/MAFcs, and MSPcn/MAFcn. Figure 3 are plotted using these results.
\(^7\)The proposed algorithm based on 7 QC variables, including MSPcs, MSPcn, MSPall, logMAFall, MSPcs/MAFcs, MSPcn/MAFcn, and MSPall/MAFall.
Supplementary Table 3. Agreement and disagreement in SNP classifications (good SNPs vs. bad SNPs) by the proposed filter and the WTCCC expert filter

<table>
<thead>
<tr>
<th>Diseases</th>
<th>$r$</th>
<th>Agreed</th>
<th>Disagreed</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Both good (%)</td>
<td>Both bad (%)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BD1</td>
<td>0.01541</td>
<td>80.33</td>
<td>12.10</td>
<td>6.47</td>
<td>1.10</td>
</tr>
<tr>
<td>BD2</td>
<td>0.01102</td>
<td>78.38</td>
<td>15.10</td>
<td>3.47</td>
<td>3.06</td>
</tr>
<tr>
<td>BD</td>
<td>0.01003</td>
<td>78.24</td>
<td>15.42</td>
<td>3.15</td>
<td>3.19</td>
</tr>
<tr>
<td>BD</td>
<td>0.01055</td>
<td>78.55</td>
<td>15.38</td>
<td>3.19</td>
<td>2.88</td>
</tr>
<tr>
<td>CAD</td>
<td>0.01251</td>
<td>79.32</td>
<td>13.92</td>
<td>4.78</td>
<td>1.99</td>
</tr>
<tr>
<td>CAD</td>
<td>0.01184</td>
<td>79.65</td>
<td>14.18</td>
<td>4.51</td>
<td>1.66</td>
</tr>
<tr>
<td>CAD</td>
<td>0.01304</td>
<td>79.60</td>
<td>14.31</td>
<td>4.38</td>
<td>1.71</td>
</tr>
<tr>
<td>CD</td>
<td>0.01344</td>
<td>79.58</td>
<td>12.57</td>
<td>5.72</td>
<td>2.13</td>
</tr>
<tr>
<td>CD</td>
<td>0.01204</td>
<td>79.54</td>
<td>12.61</td>
<td>5.69</td>
<td>2.17</td>
</tr>
<tr>
<td>CD</td>
<td>0.01334</td>
<td>79.96</td>
<td>13.34</td>
<td>4.95</td>
<td>1.75</td>
</tr>
<tr>
<td>HT</td>
<td>0.01321</td>
<td>79.69</td>
<td>13.37</td>
<td>5.20</td>
<td>1.74</td>
</tr>
<tr>
<td>HT</td>
<td>0.01185</td>
<td>79.77</td>
<td>13.60</td>
<td>4.97</td>
<td>1.66</td>
</tr>
<tr>
<td>HT</td>
<td>0.01353</td>
<td>80.07</td>
<td>13.65</td>
<td>4.92</td>
<td>1.36</td>
</tr>
<tr>
<td>RA1</td>
<td>0.01791</td>
<td>80.76</td>
<td>9.27</td>
<td>9.45</td>
<td>0.53</td>
</tr>
<tr>
<td>RA2</td>
<td>0.01012</td>
<td>77.05</td>
<td>14.23</td>
<td>4.48</td>
<td>4.24</td>
</tr>
<tr>
<td>RA</td>
<td>0.00952</td>
<td>77.87</td>
<td>15.58</td>
<td>3.13</td>
<td>3.41</td>
</tr>
<tr>
<td>RA</td>
<td>0.01002</td>
<td>77.75</td>
<td>14.88</td>
<td>3.83</td>
<td>3.54</td>
</tr>
<tr>
<td>T1D</td>
<td>0.01331</td>
<td>79.62</td>
<td>12.77</td>
<td>5.80</td>
<td>1.81</td>
</tr>
<tr>
<td>T1D</td>
<td>0.01204</td>
<td>79.60</td>
<td>12.87</td>
<td>5.70</td>
<td>1.83</td>
</tr>
<tr>
<td>T1D</td>
<td>0.01404</td>
<td>80.03</td>
<td>13.27</td>
<td>5.30</td>
<td>1.41</td>
</tr>
</tbody>
</table>
Supplementary Table 3. Continued

<table>
<thead>
<tr>
<th>Diseases</th>
<th>r</th>
<th>Agreed</th>
<th>Disagreed</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Both good(^6)</td>
<td>Both bad(^9)</td>
<td>Good vs. bad(^{10})</td>
<td>Bad vs. good(^{11})</td>
</tr>
<tr>
<td>T2D(^6)</td>
<td>0.0136(^4)</td>
<td>79.93</td>
<td>13.24</td>
<td>5.34</td>
<td>1.50</td>
</tr>
<tr>
<td>T2D(^5)</td>
<td>0.0120(^4)</td>
<td>79.69</td>
<td>13.15</td>
<td>5.43</td>
<td>1.74</td>
</tr>
<tr>
<td>T2D(^7)</td>
<td>0.0140(^4)</td>
<td>80.22</td>
<td>13.33</td>
<td>5.25</td>
<td>1.20</td>
</tr>
<tr>
<td>MDD1(^6)</td>
<td>0.0259(^1)</td>
<td>80.11</td>
<td>5.26</td>
<td>14.63</td>
<td>0.00</td>
</tr>
<tr>
<td>MDD3(^6)</td>
<td>0.0072(^2)</td>
<td>78.95</td>
<td>17.26</td>
<td>2.63</td>
<td>1.16</td>
</tr>
<tr>
<td>MDD2(^6)</td>
<td>0.0063(^2)</td>
<td>78.53</td>
<td>18.24</td>
<td>1.65</td>
<td>1.58</td>
</tr>
<tr>
<td>MDD(^3)</td>
<td>0.0038(^2)</td>
<td>78.50</td>
<td>18.13</td>
<td>1.76</td>
<td>1.61</td>
</tr>
<tr>
<td>MDD(^7)</td>
<td>0.0051(^2)</td>
<td>78.46</td>
<td>18.17</td>
<td>1.72</td>
<td>1.66</td>
</tr>
</tbody>
</table>

\(^{6}\)WTCCC expert filters without removing SNPs with MAF < 0.01.
\(^{7}\)The r value is determined by the change-point model.
\(^{2}\)In these analyses, the values of r are chosen to make the proportion of SNPs removed by the proposed filter compatible to that of the WTCCC expert filter.
\(^{3}\)The r value is chosen to make the maximum MSP in the resulting good SNPs is < 10%.
\(^{4}\)The r values are chosen to make the proportion of SNPs removed by the 5-variable and 7-variable filters compatible to the 6-variable proposed filter.
\(^{5}\)The proposed algorithm, based on 5 QC variables, including MSPcs, MSPcn, logMAFall, MSPcs/MAFcs, and MSPcn/MAFcn.
\(^{6}\)The proposed algorithm based on 6 QC variables, including MSPcs, MSPcn, MSPall, logMAFall, MSPcs/MAFcs, and MSPcn/MAFcn.
\(^{7}\)These are the results shown in Table 1.
\(^{8}\)The proposed algorithm based on 7 QC variables, including MSPcs, MSPcn, MSPall, logMAFall, MSPcs/MAFcs, MSPcn/MAFcn, and MSPall/MAFall.
\(^{9}\)% of SNPs classified as “good” by both filters.
\(^{10}\)% of SNPs classified as “bad” by both filters.
\(^{11}\)% of SNPs classified as “good” by the proposed filter (but classified as “bad” by expert).
\(^{11}\)% of SNPs classified as “bad” by the proposed filter (but classified as “good” by expert).
Supplementary Table 4. Agreement and disagreement in SNP classifications (good SNPs vs. bad SNPs) by the proposed filter and the WTCCC expert-no-MAF filter (i.e., the expert filter without removing SNPs with MAF < 0.01).

<table>
<thead>
<tr>
<th>Diseases</th>
<th>r</th>
<th>Agreed Both good (%)</th>
<th>Agreed Both bad (%)</th>
<th>Disagreed Good versus bad (%)</th>
<th>Disagreed Bad versus good (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BD1</td>
<td>0.0154</td>
<td>86.23</td>
<td>5.19</td>
<td>0.57</td>
<td>8.01</td>
</tr>
<tr>
<td>BD2</td>
<td>0.0110</td>
<td>81.72</td>
<td>5.64</td>
<td>0.12</td>
<td>12.52</td>
</tr>
<tr>
<td>BD</td>
<td>0.0100</td>
<td>81.22</td>
<td>5.60</td>
<td>0.16</td>
<td>13.01</td>
</tr>
<tr>
<td>BD</td>
<td>0.0105</td>
<td>81.61</td>
<td>5.62</td>
<td>0.14</td>
<td>12.63</td>
</tr>
<tr>
<td>CAD</td>
<td>0.0125</td>
<td>83.77</td>
<td>5.56</td>
<td>0.32</td>
<td>10.34</td>
</tr>
<tr>
<td>CAD</td>
<td>0.0118</td>
<td>83.81</td>
<td>5.53</td>
<td>0.35</td>
<td>10.31</td>
</tr>
<tr>
<td>CAD</td>
<td>0.0130</td>
<td>83.60</td>
<td>5.50</td>
<td>0.38</td>
<td>10.52</td>
</tr>
<tr>
<td>CD</td>
<td>0.0134</td>
<td>85.03</td>
<td>5.27</td>
<td>0.27</td>
<td>9.43</td>
</tr>
<tr>
<td>CD</td>
<td>0.0120</td>
<td>84.93</td>
<td>5.25</td>
<td>0.29</td>
<td>9.52</td>
</tr>
<tr>
<td>CD</td>
<td>0.0133</td>
<td>84.58</td>
<td>5.22</td>
<td>0.33</td>
<td>9.87</td>
</tr>
<tr>
<td>HT</td>
<td>0.0132</td>
<td>84.49</td>
<td>5.31</td>
<td>0.40</td>
<td>9.80</td>
</tr>
<tr>
<td>HT</td>
<td>0.0118</td>
<td>84.42</td>
<td>5.38</td>
<td>0.33</td>
<td>9.87</td>
</tr>
<tr>
<td>HT</td>
<td>0.0135</td>
<td>84.56</td>
<td>5.28</td>
<td>0.44</td>
<td>9.73</td>
</tr>
<tr>
<td>RA1</td>
<td>0.0179</td>
<td>89.28</td>
<td>5.03</td>
<td>0.93</td>
<td>4.76</td>
</tr>
<tr>
<td>RA2</td>
<td>0.0101</td>
<td>81.43</td>
<td>5.86</td>
<td>0.10</td>
<td>12.61</td>
</tr>
<tr>
<td>RA</td>
<td>0.0095</td>
<td>80.89</td>
<td>5.84</td>
<td>0.12</td>
<td>13.15</td>
</tr>
<tr>
<td>RA</td>
<td>0.0100</td>
<td>81.46</td>
<td>5.85</td>
<td>0.11</td>
<td>12.58</td>
</tr>
<tr>
<td>T1D</td>
<td>0.0133</td>
<td>85.11</td>
<td>5.40</td>
<td>0.31</td>
<td>9.18</td>
</tr>
<tr>
<td>T1D</td>
<td>0.0120</td>
<td>84.98</td>
<td>5.39</td>
<td>0.32</td>
<td>9.31</td>
</tr>
<tr>
<td>T1D</td>
<td>0.0140</td>
<td>84.92</td>
<td>5.31</td>
<td>0.40</td>
<td>9.36</td>
</tr>
<tr>
<td>Diseases</td>
<td>r</td>
<td>Both good (%)</td>
<td>Both bad (%)</td>
<td>Good versus bad (%)</td>
<td>Bad versus good (%)</td>
</tr>
<tr>
<td>----------</td>
<td>-------</td>
<td>---------------</td>
<td>--------------</td>
<td>---------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>T2D⁴</td>
<td>0.0136</td>
<td>84.85</td>
<td>5.34</td>
<td>0.42</td>
<td>9.40</td>
</tr>
<tr>
<td>T2D⁵</td>
<td>0.0120</td>
<td>84.72</td>
<td>5.36</td>
<td>0.39</td>
<td>9.52</td>
</tr>
<tr>
<td>T2D⁶</td>
<td>0.0140</td>
<td>85.01</td>
<td>5.30</td>
<td>0.46</td>
<td>9.24</td>
</tr>
<tr>
<td>MDD1⁴</td>
<td>0.0259</td>
<td>80.22</td>
<td>4.92</td>
<td>14.52</td>
<td>0.34</td>
</tr>
<tr>
<td>MDD3⁵</td>
<td>0.0072</td>
<td>78.95</td>
<td>16.81</td>
<td>2.63</td>
<td>1.61</td>
</tr>
<tr>
<td>MDD2⁶</td>
<td>0.0063</td>
<td>78.53</td>
<td>17.79</td>
<td>1.65</td>
<td>2.03</td>
</tr>
<tr>
<td>MDD⁵</td>
<td>0.0038</td>
<td>78.50</td>
<td>17.68</td>
<td>1.76</td>
<td>2.06</td>
</tr>
<tr>
<td>MDD⁷</td>
<td>0.0051</td>
<td>78.46</td>
<td>17.72</td>
<td>1.72</td>
<td>2.10</td>
</tr>
</tbody>
</table>

⁵WTCCC expert filters without removing SNPs with MAF < 0.01.
⁶The r value is determined by the change-point model.
⁷In these analyses, the values of r are chosen to make the proportion of SNPs removed by the proposed filter compatible to that of the WTCCC expert filter.
⁸The r value is chosen to make the maximum MSP in the resulting good SNPs is < 10%.
⁹The r values are chosen to make the proportion of SNPs removed by the 5-variable and 7-variable filters compatible to the 6-variable proposed filter.
⁴The proposed algorithm, based on 5 QC variables, including MSPcs, MSPcn, logMAFAll, MSPcs/MAFc, and MSPcn/MAFc.
⁴The proposed algorithm based on 6 QC variables, including MSPcs, MSPcn, logMAFAll, MSPcs/MAFc, and MSPcn/MAFc.
⁴The proposed algorithm based on 7 QC variables, including MSPcs, MSPcn, MSPall, logMAFAll, MSPcs/MAFc, and MSPcn/MAFc.
⁴The proposed algorithm based on 5 QC variables, including MSPcs, logMAFAll, MSPcs/MAFc, and MSPcn/MAFc.
⁴% of SNPs classified as “good” by both filters.
⁴% of SNPs classified as “bad” by both filters.
⁴% of SNPs classified as “good” by the proposed filter (but classified as “bad” by expert).
⁴% of SNPs classified as “bad” by the proposed filter (but classified as “good” by expert).
Figures

Figure 1. Projections of SNPs on the two-dimensional PC plane for the WTCCC CAD study. (a) PCA biplot, where the red arrows represent directions of the original variables. (b) SNP classification results from the expert filter. (c) SNP classification results from the proposed filter. (d) Overlay of good SNP boundaries defined by both filters. In (b), (c) and (d), the identified signals (green dots) are those SNPs with association test P-values < $5 \times 10^{-7}$.
Figure 2. Projections of SNPs on the two-dimensional PC plane for the GAIN MDD study. (a) PCA biplot, where the red arrows represent directions of the original variables. (b) SNP classification results from the expert filter. (c) SNP classification results from the proposed filter. (d) Overlay of good SNP boundaries defined by both filters. Note in (d) there are no red dots because the good SNP region by the proposed method is a superset of the expert good region. In (b), (c) and (d), the identified signals (green dots) are those SNPs with association test P-values $< 10^{-5}$. 
Figure 3. Performance comparisons of different filters based on the four criteria defined in the text. (a) Percentage of SNPs removed. (b) $\lambda$ from substructure adjusted test statistics. (c) Difference (expert-proposed) in the numbers of FPs retained by different filters. The numbers below each disease code on the X-axis show the counts of FPs in the unfiltered results. (d) Difference (expert-proposed) in the numbers of TPs retained in the dataset by different filters. The numbers below each disease code on the X-axis show the counts of TPs in the unfiltered results. On the X-axis of each figure, '1' indicates the results based on the $r$ value determined by the change-point model, '2' indicates the results based on the $r$ value that makes the proportion of SNPs removed by the proposed filter comparable to the expert filter and '3' indicates the results based on the $r$ value that makes the maximum MSP in the filtered dataset <10%. 
Figure 4. Characteristics of SNPs with disagreeing classification results between the proposed filter and WTCCC expert filter in CAD. In the graph, each dot represents an SNP; the yellow dots indicate SNPs labeled as ‘good’ in the proposed filter but ‘bad’ in the expert filter, and the red dots indicate the opposite scenario. The green ‘×’s indicate FPs. There are no TPs in the disagreement regions because both filters classified all TPs as ‘good’ SNPs.
Supplementary Figures
Supplementary Figure 1. Projections of SNPs on the 2-dimensional PC plane for BD, CD, HT, RA, T1D, T2D and MDD. (a) PCA biplot, where the red arrows represent directions of the original variables. (b) SNP classification results from the expert filter. (c) SNP classification results from the proposed filter. (d) Overlay of good SNP boundaries defined by both filters. In (b), (c) and (d), the identified signals (green dots) are those SNPs with p-values of association tests < 5x10^{-7} in CD, RA, T1D and T2D and <10^{-5} in BD, HT and MDD.
(a) Biplot

\[ T1D, r = 0.0133 \]

(b) Expert filter

(c) The proposed filter

(d) Overlay of both filters (zoom-in)

---

(a) Biplot

\[ T2D, r = 0.0136 \]

(b) Expert filter

(c) The proposed filter

(d) Overlay of both filters (zoom-in)
Supplementary Figure 2. Characteristics of SNPs with disagreed classification results between the proposed filter and WTCCC expert filter for BD, CD, HT, RA, T1D and T2D. In the graph, each dot represents a SNP, the yellow dots indicate SNPs are labeled as “good” in the proposed filter but “bad” in the expert filter, and the red dots indicate the opposite scenario. The green “x”s indicate FPs.
(c) CD, $r = 0.0134$

(d) HT, $r = 0.0132$
(g) T1D, $r = 0.0133$

(h) T2D, $r = 0.0136$
(i) MDD, \( r = 0.0259 \)

(j) MDD, \( r = 0.0072 \)
(k) MDD, $r = 0.0063$
References


Chapter 3

On the Aggregation of Multimarker Information for Marker-set and Sequencing Data Analysis: Genotype Collapsing vs. Similarity Collapsing

Abstract

Methods that collapse information across genetic markers when searching for association signals are gaining momentum in the literature. Although originally developed to achieve a better balance between retaining information and controlling degrees of freedom when performing multimarker association analysis, these methods have recently been proven to be a powerful tool for identifying rare variants that contribute to complex phenotypes. The information among markers can be collapsed at the genotype level, which focuses on the mean of genetic information, or the similarity level, which focuses on the variance of genetic information. The aim of this work is to understand the strengths and weaknesses of these two collapsing strategies. Our results show that neither collapsing strategy outperforms the other across all simulated scenarios. Two factors that dominate the performance of these strategies are the signal-to-noise ratio and the underlying genetic architecture of the causal variants. Genotype collapsing is more sensitive to the marker set being contaminated by noise loci than similarity collapsing. In addition, genotype collapsing performs best when the genetic architecture of the causal variants is not complex (e.g., causal loci with similar effects and

\(^2\)Reprinted with permission from Pongpanich, M., Neely, M.L. & Tzeng, J. Y. (2012). On the aggregation of multimarker information for marker-set and sequencing data analysis: Genotype collapsing vs. similarity collapsing. Frontiers in Genetics, 2
similar frequencies). Similarity collapsing is more robust as the complexity of the genetic architecture increases and outperforms genotype collapsing when the genetic architecture of the marker set becomes more sophisticated (e.g., causal loci with various effect sizes or frequencies and potential non-linear or interactive effects). Because the underlying genetic architecture is not known \textit{a priori}, we also considered a two-stage analysis that combines the two top-performing methods from different collapsing strategies. We find that it is reasonably robust across all simulated scenarios.

\textbf{Introduction}

Methods that collapse information across genetic makers when searching for association are gaining momentum in the literature (e.g., Li and Leal, 2008; Madsen and Browning, 2009; Tzeng et al., 2009, 2011; Bansal et al., 2010; Han and Pan, 2010; Hoffmann et al., 2010; Morris and Zeggini, 2010; Price et al., 2010; Wu et al., 2010, 2011; Zhang et al., 2010; Ionita-Laza et al., 2011; Neale et al., 2011). Rather than assessing the association between a phenotype and each marker individually, these methods aggregate information across several markers and assess their collective effect on the phenotype. These methods were originally developed for multimarker analysis with an aim to find a better balance between retaining information from multiple markers and controlling the degrees of freedom. Recently, they have been extended to become a powerful tool for the detecting rare variants. Due to the moderate or low frequency and the large number of variants in these analyses, pooling information across all markers is advantageous and can enhance association signals that could be missed by using traditional single marker approaches (Morris and Zeggini, 2010; Ionita-Laza et al., 2011).
The information among markers can be collapsed at the genotype level or similarity level. Genotype collapsing methods focus on the mean level of the genetic information, while similarity collapsing methods focus on the variance level of the genetic information. At the genotype level, information can be collapsed by calculating a weighted sum of the genotypes across all markers. Several methods have been developed for determining the weights used to create the combined genotype. Weights can be chosen to maximize the information retained by the combined genotype [e.g., weights based on Fourier transformation (Wang and Elston, 2007), linkage disequilibrium (LD; Li et al., 2009), and PCA (Gauderman et al., 2007; Wang and Abbott, 2008)] or to better target variants of interest [e.g., weights based on the allelic frequency (Li and Leal, 2008; Madsen and Browning, 2009; Han and Pan, 2010), functionality (Price et al., 2010), and estimated effective size (Lin and Tang, 2011)]. At the similarity level, information can be collapsed by quantifying the genetic similarity across all markers for each pair of unrelated individuals. Current developments include the kernel machine approaches where identity-by-state (IBS) is used as a kernel to summarize information (Kwee et al., 2008; Schaid, 2010a,b; Wu et al., 2010, 2011), building regression models that relate trait similarity with genetic similarity (Wessel and Schork, 2006; Tzeng et al., 2009, 2011; Mukhopadhyay et al., 2010), or random effect methods (Goeman et al., 2004; Tzeng and Zhang, 2007) where genetic similarity is used to specify the variance–covariance structure of the multimarker effects.

Many comparative studies are available that investigate the performance of different collapsing methods for detecting rare variants (Bansal et al., 2010; Morris and Zeggini, 2010; Bacanu et al., 2011; Basu and Pan, 2011) and common variants (Chapman and Whittaker,
2008; Lin and Schaid, 2009; Ballard et al., 2010). They provide substantial insight for understanding the strengths and weaknesses of each method and help researchers select the most suitable approach for their analysis. For example, genotype-level collapsing would be the optimal approach if the effects of different loci are additive and of a similar size. On the other hand, similarity-level collapsing are more powerful if interactive or non-linear effects exist among the markers or if the effect sizes vary radically across markers. While collapsing methods can improve the power to identify genetic variants over classic single marker or multimarker approaches, the power gain comes with limitations: Most collapsing methods target either rare or common variants, but not both, and their performance typically suffers when non-causal variants are included in the marker set.

Previous comparative papers also recognized the need for more in depth studies to compare these methods across an exhaustive set of scenarios that can occur when investigating complex phenotypes (Bansal et al., 2010; Basu and Pan, 2011). With this goal in mind, we further investigate the strengths and weaknesses of genotype collapsing and similarity collapsing over a wide range of plausible scenarios. Unlike the previous comparative studies that focused on the relative performance of individual methods, in this work we seek to understand the advantages and drawbacks of the two collapsing paradigms. That is, rather than examining the ability of a set of particular methods for detecting rare variants or common variants solely, we concentrate on the implications of applying the two collapsing strategies. The factors that we examine in this work include (a) the underlying genetic architecture of the causal variants (i.e., effect size, frequency, and number causal alleles within a causal locus), (b) composition of the variant set (i.e., proportion of causal
variants in the set and LD between causal and non-causal loci in the set), and (c) the weighting scheme used in the collapsing method. Our results show that neither collapsing strategy outperforms the other across all simulated scenarios. Genotype collapsing is more sensitive to the marker set being contaminated by noise loci than similarity collapsing. In addition, genotype collapsing performs best when the genetic architecture of the causal variants is not complex (e.g., causal loci with similar effects and similar frequencies). Similarity collapsing is more robust as the complexity of the genetic architecture increases and outperforms genotype collapsing when the genetic architecture of the marker set becomes more sophisticated (e.g., causal loci with various effect sizes or frequencies and potential non-linear or interactive effects). Because the underlying genetic architecture is not known a priori, we also considered a two-stage analysis that combines two top-performing methods from the two collapsing paradigms. The approach is shown to be reasonably robust across all simulation scenarios and provides an attractive comprehensive approach.

In the remaining sections of this paper, we briefly review the representative genotype-level and similarity-level collapsing methods we compared in the simulation study, describe the simulation study used to investigate the performance of the two collapsing paradigms, present and interpret results of the simulation study, and conclude with a discussion of the work’s major findings and connections to the current literature.

**Materials and Methods**

To investigate the strengths and weaknesses of the two collapsing paradigms, we compared the performance of representative methods from each school. We considered two genotype-level collapsing methods, combined multivariate and collapsing (CMC; Li and
Leal, 2008) and variable threshold (VT; Price et al., 2010), and one similarity-level collapsing method, gene-trait similarity regression (SimReg; Tzeng et al., 2009, 2011). As explained in Section “Gene-trait Similarity Regression”, we note that other current similarity-collapsing methods can be viewed as special cases of SimReg, such as the C-alpha test (Neale et al., 2011) and the sequence kernel association test (SKAT; Wu et al., 2011). For each paradigm, we considered methods that target rare variants (VT for genotype level and SKAT for similarity level) and those that use all available variants (CMC for genotype level and SimReg for similarity level). In addition, we considered one standard approach for marker-set analysis that does not employ a collapsing technique, the minimum p-value method (MinP). Each method investigated in this work has been developed and reported previously. Thus, we only briefly review the main components of each method here.

**Methods.**

*Single SNP-based marker-set test: MinP.* One standard approach for examining association between a marker set and a phenotype is to use the best-scoring SNP from the set as a summary measure for the evidence of association for the entire marker set (referred to as MinP). The procedure begins by testing each SNP in the marker set for association individually, and the best-scoring SNP is taken to be the variant with the minimum p-value. Permutation is then used to adjust for multiple comparisons and to account for the LD structure among the SNPs in the marker set. This is achieved by permuting the phenotype R times and recording the p-value of the best-scoring SNP from each permuted dataset. The empirical p-value is then calculated as proportion of minimum p-values from the permuted datasets that are less than the minimum p-value observed in the original data set. In this
work, R was taken to be 1000 and Pearson’s Chi-Square test was used to obtain the
association p-value for each marker.

**Combined multivariate and collapsing method.** The CMC method (Li and Leal, 2008) is a procedure that combines collapsing information across genetic markers and multivariate tests into a single approach. The procedure aims to unify the advantages of both collapsing, which enriches association signals and decreases degrees of freedom by aggregating information across multiple markers, and multimarker tests, which model the association of all variants in a marker set simultaneously. Unlike most rare-variant genotype collapsing methods, the CMC test statistic is computed on all loci in the marker set rather than focusing only on loci with low minor allele frequency (MAF). The procedure begins by dividing the markers into subgroups based on some pre-specified criteria. Then within each group, the information across all markers is collapsed such that an individual is coded as a 1 if they have a rare allele present at any marker within the sub-group and as a 0 otherwise. A multivariate test is then applied to the groups of collapsed markers to examine the association between the marker set and the phenotype. In this work, we used MAF to define subgroups. If the MAF of a marker was greater than $f^*$, the marker created a singleton, otherwise the marker was placed into a group with all other markers with MAF less than $f^*$ and collapsed in the manner described above. The multivariate test used to determine association was Hotelling’s $T^2$ test (Xiong et al., 2002).

**Variable threshold method.** Based on the assumption that variants with MAF less than $T$ are more likely to be functional than variants with MAF greater than $T$, the VT method (Price et al., 2010) focuses only on loci with MAF lower than a certain threshold $T$. 

70
Instead of a fixed MAF threshold that has to be determined \textit{a priori} (e.g., Madsen and Browning, 2009 and CMC), VT allows the threshold \( T \) to vary when assessing the association between a marker set and a phenotype. The procedure begins by calculating a score value \( z(T) \) for each allele frequency threshold \( T \) and finding the maximum score value \( z_{\text{max}} \) over all thresholds. For a given value of \( T \), the score value compares the number of rare variants (i.e., those with MAF less than \( T \)) in a marker set among distinct phenotype states. Permutation is then used to assess the statistical significance of \( z_{\text{max}} \). This is achieved by permuting the phenotype \( R \) times and recording the maximum score value from each permuted dataset. The empirical p-value is then calculated as proportion of maximum score values from the permuted data sets that are greater than the maximum score value observed in the original dataset. In this work, \( R \) was taken to be 1000 and the score value \( z(T) \) was calculated following the procedure outlined by Price et al. (2010).

\textit{Gene-trait similarity regression.} Gene-trait similarity regression (SimReg) quantifies genetic similarity between pairs of individuals at each locus and aggregates multimarker information by summing the similarity scores across all loci. The method regresses trait similarity between individual pairs on their overall genetic similarity, and then evaluates the gene-trait association by testing the significance of the resulting regression coefficient. Typically, the test statistic is computed from all loci in the marker set with locus-specific weights that depend on allele frequencies. These weights are designed to better distinguish between the sharing due to a rare event from that due to a common event. In this work, the weights were taken to be \( f^{-X/4} \) where \( f \) is the allele frequency and \( X \) was taken to be 0, 3, or 4. Thus, as \( X \) increases away from 0 the contribution of rare variants is weighted more
strongly in the test statistic. It has been shown that the SimReg regression coefficient can be expressed as a variance component of a random effects model (Tzeng et al., 2009, 2011). This result unifies gene-trait similarity regression with other variance-component methods, including the kernel machine regression, as well as their special cases that target rare variants only (e.g., the C-alpha and SKAT methods). Specifically, the C-alpha method is SimReg with a thresholding weight based on the MAF, and SKAT is SimReg with weights taken to be \((1 - f)^{24}\). Unlike the weights typically used with SimReg, the C-alpha and SKAT weights are designed to only consider rare variants in the marker set. In this work, trait similarity and genetic similarity were calculated by matching allele proportions as outlined in Tzeng et al. (2009, 2011), and the significance of the regression coefficient from SimReg was assessed using the score test developed by the same authors.

**Simulation studies.** We performed simulation studies to explore the strengths and weaknesses of the two different collapsing paradigms when analyzing case–control data over a wide range of scenarios that could occur when investigating the genetic architecture of a complex phenotype. We compared the powers of representative genotype-based and similarity-based collapsing methods against each other, and the performance was benchmarked against a standard approach for marker-set analysis that does not involve collapsing information across markers. For ease of discussion, let geno-sum refer to genotype-level collapsing and let sim-sum refer to similarity-level collapsing.

**Simulation settings.** Our simulation studies were based on two haplotype distributions derived from aligned sequence data on chromosome 21 of 109 individuals from the CHB sample of the 1000 Genomes Project (The 1000 Genomes Project Consortium,
We performed variant calling using GATK (McKenna et al., 2010; DePristo et al., 2011) and divided the resulting variants into groups by exon. Marker genotypes were phased using BEAGLE (Browning and Browning, 2007). The first haplotype distribution was based on a 12-locus exon that consisted of 11 biallelic SNPs and 1 indel with three alleles. The second haplotype distribution was formed by combining three different exons, each with 10 biallelic SNPs, to create a 30-locus region. The MAFs for each marker based on the genotypes of the 109 sampled individuals are given in Table1 for Haplotype Distributions 1 and 2, respectively.

For Haplotype Distribution 1, case–control samples were generated assuming that 4 out of the 12 markers were causal. All combinations of four causal markers were considered in the simulation studies, resulting in 495 possible scenarios. Data was generated under four simulation settings in order to investigate the performance of genotype-level vs. similarity-level collapsing. (1) Under the first setting (Figure1), all four causal loci increase the disease risk with the same odds ratio of 1.3. (2) Under the second setting (Figure2), we allowed the four causal loci to have various effect sizes on the phenotype. Those with MAF less than 0.01 were set to have an odds ratio of 2 while all others were set to have an odds ratio of 1.3. (3) Under the third setting (Figure3), we took advantage of the triallelic indel in the marker set and allowed two of three alleles from the indel to be causal (i.e., the rarest and second rarest, with frequencies 0.009 and 0.096 respectively). In this setting, we only considered scenarios where this indel was included as one of the four causal loci (which resulted in 165 possible scenarios instead of 495). Each causal variant was set to have the same effect on the phenotype with an odds ratio of 1.3. (4) Under the fourth setting (Figures4–5), we considered
different proportions of causal loci in the marker set – 2 out of 12 and 4 out of 4. In both scenarios, the causal loci were set to have the same effect on the phenotype with an odds ratio of 1.3. When 2 out of 12 loci were assumed to be causal, all combinations of two markers were considered, resulting in 66 possible scenarios. When four out of four loci were assumed to be causal, the same 495 possible scenarios were considered, but the remaining eight loci were not included in the marker set during the analysis.

For Haplotype Distribution 2, case–control samples were generated assuming that 2 out of the 30 markers were causal. All combinations of two causal markers were considered in the simulation studies, resulting in 435 possible scenarios. Data was generated under two simulation settings in order to investigate the two collapsing paradigms’ performance in a larger genomic region with a low proportion of causal variants. (1) The first setting (Figure 6) is analogous to Setting 1 for Haplotype Distribution 1. That is, both causal loci were set to have the same effect size on the phenotype – an odds ratio of 1.3. (2) The second setting (Figure 7) is analogous to Setting 2 for Haplotype Distribution 1. That is, both causal loci were allowed to have different effect sizes with the same direction on the phenotype. When the MAF of the causal loci was less than 0.01, the odds ratio was taken to be 2; otherwise it was taken to be 1.3.

**Data generation.** To create a case–control sample of size n under an additive genetic model, we generated the haplotype pair of an individual conditional on their disease status and then dissolved the haplotype pair into its unphased genotypes. Let \( P(H = h|Y = y) \) denote the probability of having a particular haplotype pair conditional on disease status. This probability can be expressed as
\[ P(H = h|Y = y) = \frac{P(Y = y|H = h) \cdot P(H = h)}{\sum_h P(Y = y|H = h) \cdot P(H = h)}. \]

For a case individual, \( P(Y = 1|H = h) \) was found using the logistic regression model

\[ P(Y = 1|H = h) = \frac{\exp(\beta_0 + \mathcal{Z}(D)^T \beta)}{1 + \exp(\beta_0 + \mathcal{Z}(D)^T \beta)}. \]

For a control individual, \( P(Y = 0|H = h) = 1 - P(Y = 1|H = h) \). The function \( \mathcal{Z}(\cdot) \) depends on the genetic mode of the loci associated with the disease. Under an additive genetic model, \( \mathcal{Z}(D) = D \) where \( D \) is the vector of minor allele counts for each locus in a given haplotype pair. The vector \( \beta \) was taken to be the log of 1.3 or 2 for all causal loci and the log of 1.0 for all non-causal loci in the marker set. The value of \( \beta_0 \) was set to maintain a disease prevalence of 1%. Once \( P(Y = y|H = h) \) was calculated for each haplotype pair formed from the derived haplotype distribution, the vectors \( P_{H|Y=y} = \left( P(H = h_1|Y = y) \cdots P(H = h_q|Y = y) \right) \) were calculated for \( Y = 0 \) and \( Y = 1 \), where \( q \) is total number of haplotype pairs. The sample was generated by taking 1000 draws from the multinomial distribution parameterized by \( P_{H|Y=0} \) to determine the haplotype pairs of the control individuals and by taking 1000 draws from the multinomial distribution parameterized by \( P_{H|Y=1} \) to determine the haplotype pairs of the case individuals. The haplotype pair of each individual was then dissolved into its unphased genotype.

**Computational details.** For each simulation setting, 500 replicate data sets were generated for each possible combination of 4 (or 2) causal loci. Each data set was analyzed using the following methods: (1) MinP; (2) CMC with the MAF collapsing threshold set at 0.01 or 0.05, which will be referred to as CMC01 and CMC05, respectively; (3) VT; (4)
SimRegX, i.e., SimReg based on all loci with weights taken to be $f^{-0/4}$ (SimReg0), $f^{-3/4}$ (SimReg3), and $f^{-4/4}$ (SimReg4); and (5) SKATsr, i.e., SimReg based on rare variants only by using the SKAT weight $(1 - f)^2$. In addition, we also considered a two-stage procedure that combines genotype-level collapsing and similarity-level collapsing. The two-stage procedure, referred to as 2stage, performs both SimReg0 and VT, but assesses the significance of each analysis at $\alpha/2$ instead of $\alpha$ like the other methods, where $\alpha$ is a desired significance level. If either underlying method rejects the null hypothesis, the two-stage procedure rejects the null. The performance of each method was compared by calculating their power to detect the association between the marker set and the phenotype at $\alpha = 0.05$ as well as their Type I error rate.

**Results**

To investigate the performance of each collapsing paradigm, we calculated each representative method’s Type I error rate and power to detect an informative marker set (i.e., one containing causal loci). We present the Type I error rates in Table 2. All methods have desirable and similar performances under a null model. Each had Type I error rates that were around the nominal level being considered (i.e., $\alpha=0.01$, 0.05, or 0.10). We present power results in Figures1–8. Each figure groups the results into categories defined by combinations of two factors – range of average causal allele frequency (across columns) and range of percent-signal (down rows). Percent-signal is calculated as

$$\frac{m_c}{m} + \frac{\overline{R^2}_i - \min R^2}{\max R^2 - \min R^2} \cdot \frac{m_{nc}}{m},$$
where m is the total number of loci in the marker set, \( m_c \) is the number of causal loci in the marker set, \( m_{nc} = m - m_c \) is the number of non-causal loci, and \( \bar{R}_2^2 \) is the average pair-wise \( R^2 \) between causal and non-causal loci for simulation scenario i. The quantities \( \max \bar{R}_2^2 \) and \( \min \bar{R}_2^2 \) are the maximum and minimum, respectively, of \( \bar{R}_2^2 \) across all i. The fraction \( (\bar{R}_2^2 - \min \bar{R}_2^2) / (\max \bar{R}_2^2 - \min \bar{R}_2^2) \) is used to rescale the small range of the observed \( \bar{R}_2^2 \) to range from 0 and 1. Within each figure, boxplots of the power results (listed on the y-axis) are given for the methods under consideration (listed on the x-axis) for each category. Boxplots were created using the power results from the simulated marker-set scenarios that belonged to each average-causal allele frequency by percent-signal category.

**Underlying genetic architecture.**

**Causal allele frequency (Figures 1 and 6).** The average causal allele frequency reflects the ratio of rare causal variants to common causal variants in the analysis set. A low average causal allele frequency results from a high rare to common variant ratio, whereas a moderate or high average causal allele frequency results from a mix of rare and common variants or a low rare to common variant ratio. In Figures 1 and 6, geno-sum and sim-sum methods have comparable performances when comparing rare-variant approaches to rare-variant approaches and all-variant approaches to all-variant approaches. That is, VT and SKATsr have similar power when the causal allele frequencies are low. However, when the causal allele frequencies increase, VT performs slightly better than SKATsr. Similarly, CMC and SimRegX have similar power with the exception SimReg4 at moderate and high frequencies. In these settings, SimReg4 under performs compared to the other versions of SimRegX and tends to perform more like the rare-variant approaches as it most strongly
upweights the contribution from rare variants. As expected, the relative performance between rare-variant and all-variant approaches depends on the underlying causal allele frequencies. When causal allele frequencies are low (i.e., Column 1), methods that target rare variants (i.e., VT and SKATsr) have the best performance. As the frequencies increase to moderate or high (i.e., Columns 2–4 in Figure 1 and Columns 3–4 in Figure 6), methods that use all variants (i.e., CMC and SimRegX) start to outperform the rare-variant approaches. At these elevated causal allele frequencies, the power difference between rare-variant and all-variant approaches is more substantial for sim-sum methods (i.e., SKATsr vs. SimRegX) than for geno-sum methods (i.e., VT vs. CMC) as the power of SKATsr remains relatively constant as the frequencies increase. SKATsr does not take advantage of any information from common variants because it extremely downweights their contributions in the combined genotype. As a result, VT typically outperforms SKATsr when causal allele frequencies are high because it uses variable thresholding that can include common variants in the analysis. The two-stage procedure, which combines VT and SimReg0, does not suffer the same dramatic power switch when the causal allele frequencies change and is able to maintain similar or higher power than the best collapsing approach. MinP never uniformly outperforms or is outperformed by any geno-sum or sim-sum method. However, it often had satisfactory performance when the percent signal is low (e.g., the bottom row in Figures 1, 3, 4, and 6).

The above observations hold regardless of the percent-signal. When investigating the impact of other simulation factors, we will refer back to this scenario as a baseline for comparison.

**Magnitude of causal allele effect (Figures 2 vs. 1 and 7 vs. 6).** When we allow the underlying causal variants to have different effects sizes in the same direction, we see an
overall increase in power for all methods. The largest gain in power is seen for rare-variant sim-sum approaches, as seen in SKATsr which shortens its gap with VT or even has better power when, e.g., comparing Figure2 to Figure1. Substantial power gain is also observed for all-variant sim-sum methods at low causal allele frequencies, where SimReg3 and SimReg4 have comparable or better power than VT and SKATsr. Nevertheless, the general pattern of results observed in the baseline scenario still holds. That is, geno-sum and sim-sum methods have comparable performances when comparing similar approaches (i.e., rare-variant to rare-variant, similarly for all-variant approaches) across all simulation settings.

**Multiple causal alleles in a locus (Figures 3 vs. 1).** When we allow multiple alleles within a locus to be causal with the same effect, sim-sum methods generally perform better than geno-sum methods. That is, SimReg3 is the best or near-best across all simulation settings. When comparing rare-variant approaches, SKATsr performs better than VT at low frequencies and becomes comparable to VT when the frequencies increase (i.e., no longer have power loss). This result is different from the baseline scenario where SKATsr is comparable to VT when frequencies are low and tends to have less power as frequencies increase. Similarly, when comparing all-variant approaches, SimRegX outperforms CMC, with the exception of SimReg4, regardless of the underlying causal allele frequencies. These patterns hold regardless of percent-signal.

**Composition of marker set.**

**Proportion of causal Loci (Figures 1, 4, 5, and 6).** Results of different proportions of causal loci are shown in Figure 4 (for 4 out of 4), Figure 1 (for 4 out of 12), Figure 5 (for 2 out of 12), and Figure 6 (for 2 out of 30). When the proportion of causal loci is high (4/4
scenario; Figure4), geno-sum performs better than or similar to sim-sum across different settings. Specifically, for all-variant methods (i.e., CMC vs. SimRegX), geno-sum and sim-sum methods generally perform comparably, and at low causal allele frequencies, geno-sum has a slight advantage over sim-sum. For rare-variant methods (i.e., VT vs. SKATsr), geno-sum clearly outperforms sim-sum. Furthermore, VT performs comparable to SimRegX even at moderate and high causal allele frequencies.

When the proportion of causal loci drops, all methods suffer a power loss, but the loss suffered by sim-sum is less than that suffered by geno-sum. As a result, sim-sum begins to have similar or more power than its geno-sum counterpart. For example, if we focus on low causal alleles frequencies (i.e., first column), we see that the power gain of VT over SKATsr becomes smaller and smaller when we move from Figure 4 (4/4), to Figure 1 (4/12), to Figure 5 (2/12) and to Figure 6 (2/30). Additionally, when percent-signal is low, SKATsr can outperform VT. The relative performance of the two-stage procedure remains consistent regardless of the proportion of causal loci. That is, the power of 2stage always falls between that of VT and SimReg0, and thus outperforms the SimReg0 at low causal allele frequency when VT is superior and vice versa when the causal allele frequency is moderate or high.

**LD between causal and non-causal loci (Figure 1).** Recall that the calculation of percent-signal involves two components: (1) $R^2$ that reflects the LD between causal and non-causal loci in the marker set, and (2) the proportional of causal loci. As such, percent-signal can be used as a proxy to investigate the effects of the underlying LD on the paradigms’ performances. All methods suffer a power loss as LD decreases (i.e., Figure 1, from top row to bottom row), but sim-sum is less sensitive to the decrease of LD compared to geno-sum.
Therefore, although general trends among the methods’ performances hold regardless of the underlying LD, the magnitude of the power gain or loss between the geno-sum and sim-sum methods is influenced by changes in LD. For rare-variant approaches, SKATsr is quite robust to the decrease of LD, while VT is sensitive to the drop of LD and decreases with LD. As a result, when the causal allele frequencies are low, the relative performance of SKATsr vs. VT flip-flops depending on LD (i.e., Column 1 in Figure 1). For moderate or high allele frequencies, the relative power loss of SKATsr compared to VT increases as LD increases. Because SKATsr does not incorporate information from common variants, it cannot fully benefit from an increase in LD like VT whose power increases with LD as it is able to incorporate information from common variants (i.e., Figures 1, 5, and 6). For all-variant approaches, SimRegX and CMC are fairly robust to LD changes and their performances remain comparable as LD changes, with the exception of SimReg4. CMC is a hybrid of a geno-sum approach and a classic genotype-based multimarker approach; therefore it is not as sensitive to the underlying LD pattern like typical geno-sum approaches such as VT. The power of the two-stage procedure is also fairly robust to changes in LD.

**Weighting schemes used in collapsing methods.**

**SimReg method.**

*SimReg0, SimReg3 vs. SimReg4.* For sim-sum methods based on all loci, when the causal allele frequencies are low, SimReg0, which does not upweight contributions from rare alleles, has the lowest power, and SimReg4, which uses the strongest weights to promote sharing from rare alleles, has the highest power. And as expected, when the causal allele frequencies increase, the relationship flips. This pattern of results holds regardless of percent-
signal. However, the pattern does not hold when multiple alleles within a particular locus are causal (Figure3). Under this scenario, SimReg4 no longer outperforms the other versions of SimRegX at low causal allele frequencies; instead SimReg3 performs the best. Overall, the results show that using strong weights can boost the power to detect rare variants, but it may risk losing power when some causal variants are common. Among the weights studied, SimReg3 appears to achieve a better compromise and exhibited more robustness against the influence of causal allele frequencies.

**Rare variants vs. all variants.** When the causal loci are rare, the results suggest that SimReg4 is not strong enough to surpass VT and more extreme weights such as SKAT weights are needed. At low causal allele frequencies, SKATsr outperforms SimReg4, and SimReg4 performs comparably to SimReg3 which outperforms SimReg0. As the causal allele frequencies increase, SKATsr and SimReg4 suffer a power loss, which is quite severe for SKATsr. This again suggests that it might be advantageous to use a sim-sum method that considers all variants with a moderate weighting scheme (e.g., SimReg3), since it achieves better power at moderate and high causal allele frequencies, and yet the power loss at low causal allele frequencies is not as severe as the power loss observed for SKATsr at moderate and high causal allele frequencies.

**CMC method.** Like SimReg, which version of CMC performs the best appears to depend on the frequency of the causal alleles in the marker set (see Figures1–3). When the causal allele frequencies are low, CMC01 performs better than CMC05. However, when the causal allele frequencies are moderate or high, CMC05 performs better than CMC01. This pattern holds regardless of percent-signal, the magnitude of the causal effect, or the number
of causal alleles at a particular locus. These results suggest using a fixed threshold in geno-
sum methods may be unsatisfactory. When the threshold is set too low, the power of CMC
may suffer due to increased degrees of freedom. However, if the threshold is set too high, the
power of CMC may also suffer as noise loci are introduced into the combined genotype.

**VT method.** The weighting scheme of VT is to triage loci with high MAF. It weights
each locus by an indicator function, i.e., weight equal to 1 if MAF is less than a data-driven
threshold T, and weight equal to 0 otherwise. As a result, it performed the best if all causal
alleles have small frequencies. It suffered non-trivial power loss if some heterogeneity
existed among allele frequencies. Nevertheless, compared to other approaches that target rare
variants (e.g., SKATsr), the advantage of a data-driven threshold becomes apparent when
non- rare variants are present in the marker set: The power loss between VT and its all-
variant counterpart is significantly less severe than that of SKATsr. VT’s adaptive threshold
permits the inclusion of some information from common variants, while SKATsr’s strong
weight against common variants does not allow them to contribute any information to the
combined genotype.

**Discussion**

Collapsing methods are drawing big attention due to their usefulness in marker-set
analysis and rare variant detection. Collapsing information can be done at genotype level or
at similarity level. In this work, we investigated the implications of employing these different
collapsing strategies when performing multimarker association analysis in order to uncover
the strengths and weak- nesses of the two paradigms. Using realistic data based on 1000
Genomes Project, we considered scenarios where the causal alleles can be rare, non-rare, or a
mixture of two, where the causal loci can be biallelic SNPs or multiallelic markers, and where the association signal of a marker set, quantified based on the proportion of causal loci and LD structure, varied from weak to strong. We also considered approaches proposed to better target rare variants and those that use all variant information in the marker set. For genotype-level collapsing, we considered VT, which aggregates and uses information only from loci with MAF below the adaptive threshold, and CMC, which collapses rare-variant information but retains and analyzes information from all loci. For similarity-level collapsing, we considered SimReg which can incorporate many current variance-component based approaches (e.g., C-alpha and SKAT) as special cases. As a result, SimReg can be used as a rare-variant as well as an all-variant approach. We considered weights that upweighted the contribution from rare variants with varying strengths, ranging from the extreme case that placed almost no weight on common variants (SKATsr), to strong-but-not-extreme weights against common variants (SimReg4 and SimReg3), to not promoting rare variants at all (SimReg0).

Our results show that neither collapsing strategy outperforms the other across all simulated scenarios. Nevertheless, employing a collapsing strategy is advantageous across all simulated scenarios. At least one of the two strategies resulted in higher power than the standard approach which does not aggregate information across markers. Two factors that dominate the performance of the collapsing strategies are the signal-to-noise ratio and the underlying genetic architecture of the causal variants. We found that similarity-level collapsing tends to be more robust to changes in the signal-to-noise ratio. That is, the power loss due to the inclusion of non-causal variants in the marker was much less substantial for
similarity-level collapsing than for genotype-level collapsing. This can be seen by comparing the power loss from Figure 4 (i.e., 4 out of 4) to Figure 1 (4 out of 12), Figure 5 (2 out of 12), and Figure 6 (2 out of 30), as well as the similar or higher power of similarity collapsing than genotype collapsing when the proportion of the functional loci is moderate or low (e.g., Figures 5–7).

The performance of these collapsing strategies was also heavily influenced by the underlying genetic architecture of the causal variants, which we refer to as their effect patterns (e.g., same or varying effect sizes/directions, linear vs. non-linear, additive vs. interactive) and the variant frequencies. Genotype-level collapsing generally performs best when the genetic architecture of the causal variants is not complex. That is, the causal variants have similar, additive, linear effects with similar frequencies. When collapsing at the genotype level, the underlying philosophy is that all loci share the same effect size (and hence can be well detected by a common regression coefficient). Therefore the approach lends itself to scenarios where the proportion of functional loci in a marker set is high and each locus exhibits similar influence on traits. In contrast, collapsing at the similarity level can be viewed as test of the variation among regression coefficients and allows each locus to have a different effect size. As a result, this approach can accommodate more complex genetic architectures such as a mixture of rare and non-rare variants, different effect sizes and directions, and multiple causal alleles within a locus. This notion is supported by our results which show that similarity-level collapsing is more robust as the complexity of the genetic architecture increases and outperforms genotype collapsing when the genetic architecture of the marker set becomes more sophisticated.
The underlying causal allele frequencies impact the choice of the weighting scheme (i.e., approaches based on all variants vs. rare variants only) more than the choice of collapsing paradigm. As expected, when the causal variants are rare, approaches that target rare variants will be the best, but when there is a mixture of rare and common, approaches that use all variant information will be the best. The power lost by using a rare-variant approach when common causal variants are present in the marker set is much more severe than the power lost by using an all-variant approach when the causal variants are all rare. Based on this observation and because the frequency of causal variants is not known \textit{a priori}, a reasonable strategy would be to use an all-variant approach with a moderate weight against common variants, such as SimReg3 or CMC with a suitable threshold. Indeed, our results also show that using an adaptive threshold can gain robustness against the unknown frequency distributions of the causal variants (i.e., the relatively small power loss of VT compared to SKATsr for high allele frequencies). This suggests that CMC with variable threshold holds good potential.

Because the optimal statistical methods depend on the unknown architecture of the causal variants and the marker set, we also considered a two-stage analysis. The two-stage procedure performs both VT, which generally performs the best for rare variants, and SimReg0, which generally performs the best or near-best for common variants. Like CMC, this hybrid strategy uses two different strategies to detect rare and non-rare variants. However, the two-stage approach can gain efficiency by using fewer degrees of freedom when modeling multiple common variants and is applicable to quantitative traits. By combining the top method from each scenario, the two-stage approach is reasonably robust.
and yields comparable though not necessarily the highest power across all simulation
scenarios. It provides an attractive alternative to SimReg3 and CMC with variable threshold.

We focused on binary phenotypes in our simulation studies. However, most of the
methods considered here are applicable to quantitative phenotypes (except CMC). We
simulated data under one setting (2 causal loci out of 12 markers with same effects; see
Figure 8) to compare the performance of these methods for binary and quantitative traits. The
general pattern of our findings typically holds between the two trait types. In short, genotype
collapsing is more sensitive to the marker set being contaminated by noise loci than
similarity collapsing. In addition, genotype collapsing performs best when the genetic
architecture of the marker set is not complex (e.g., causal loci with similar effects and similar
frequencies). Similarity collapsing is more robust as the complexity of the genetic
architecture increases and outperforms genotype collapsing when the genetic architecture of
the marker set becomes more sophisticated (e.g., causal loci with various effect sizes or
frequencies and potential non-linear or interactive effects). We expect the same trends of
results to occur under the other simulation settings.
Tables

Table 1. Minor allele frequency (MAF) of markers resulting from sequencing data from the CHB sample of 1000 Genomes Project.

<table>
<thead>
<tr>
<th>Marker ID&lt;sup&gt;*&lt;/sup&gt;</th>
<th>MAF</th>
<th>Marker ID</th>
<th>MAF</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3400</td>
<td>1</td>
<td>0.1881</td>
</tr>
<tr>
<td>B</td>
<td>0.1697</td>
<td>2</td>
<td>0.1330</td>
</tr>
<tr>
<td>C</td>
<td>0.0275</td>
<td>3</td>
<td>0.1101</td>
</tr>
<tr>
<td>D</td>
<td>0.0229</td>
<td>4</td>
<td>0.0734</td>
</tr>
<tr>
<td>E&lt;sup&gt;**&lt;/sup&gt;</td>
<td>0.0092</td>
<td>5</td>
<td>0.0596</td>
</tr>
<tr>
<td>F – L</td>
<td>0.0046</td>
<td>6</td>
<td>0.0459</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7</td>
<td>0.0367</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
<td>0.0321</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9 – 10</td>
<td>0.0275</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11</td>
<td>0.0229</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12</td>
<td>0.0138</td>
</tr>
<tr>
<td></td>
<td></td>
<td>13 – 18</td>
<td>0.0092</td>
</tr>
<tr>
<td></td>
<td></td>
<td>19 – 30</td>
<td>0.0046</td>
</tr>
</tbody>
</table>

<sup>*</sup> Marker ID s were assigned according to sorted MAFs rather than genomic position.

<sup>**</sup> Marker E is a one indel with three alleles while the other markers are biallelic SNPs.

Table 2. Type I error rates averaged over the 495 possible scenarios for 4 causal markers out of 12 and 500 replicate data sets.

<table>
<thead>
<tr>
<th>Sig Level (g)</th>
<th>CMC01</th>
<th>CMC05</th>
<th>SimReg0</th>
<th>SimReg3</th>
<th>SimReg4</th>
<th>SKATsr</th>
<th>VT</th>
<th>2-Stage</th>
<th>MinP</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>0.0098</td>
<td>0.0101</td>
<td>0.0132</td>
<td>0.0111</td>
<td>0.0120</td>
<td>0.0105</td>
<td>0.0097</td>
<td>0.0126</td>
<td>0.0101</td>
</tr>
<tr>
<td>0.05</td>
<td>0.0498</td>
<td>0.0499</td>
<td>0.0496</td>
<td>0.0499</td>
<td>0.0487</td>
<td>0.0492</td>
<td>0.0502</td>
<td>0.0515</td>
<td>0.0504</td>
</tr>
<tr>
<td>0.10</td>
<td>0.0999</td>
<td>0.0997</td>
<td>0.0926</td>
<td>0.0969</td>
<td>0.0924</td>
<td>0.0966</td>
<td>0.1007</td>
<td>0.0965</td>
<td>0.1002</td>
</tr>
</tbody>
</table>
Figures

Figure 1. Power results when casual loci have same effects under 4 causal loci out of 12 markers setting for binary trait. Power is calculated over 500 replicate data sets with $\alpha = 0.05$; there are 495 possible combinations of 4 causal loci out of 12 markers; boxplots summarize the power results for marker-combinations belonging to each average functional allele frequency [(A) is 0–0.02, (B) is 0.04–0.06, (C) is 0.08–0.11, (D) is 0.13–0.15] and percent-signal category; 1 = CMC01, 2 = CMC05, 3 = SimReg0, 4 = SimReg3, 5 = SimReg4, 6 = SKATsr, 7 = VT, 8 = 2Stage, 9 = MinP.
Figure 2. Power results when casual loci have different risk effect sizes under 4 causal loci out of 12 markers setting for binary trait. Power is calculated over 500 replicate data sets with $\alpha = 0.05$; there are 495 possible combinations of 4 causal loci out of 12 markers; boxplots summarize the power results for marker-combinations belonging to each average functional allele frequency [(A) is 0–0.02, (B) is 0.04–0.06, (C) is 0.08–0.11, (D) is 0.13–0.15] and percent-signal category; 1 = CMC01, 2 = CMC05, 3 = SimReg0, 4 = SimReg3, 5 = SimReg4, 6 = SKATsr, 7 = VT, 8 = 2Stage, 9 = MinP.
Figure 3. Power results for multiple causal alleles in a locus with same effects under 4 loci out of 12 markers setting for binary trait. Power is calculated over 500 replicate datasets with $\alpha = 0.05$; there are 165 possible combinations of 4 causal loci out of 12 markers; boxplots summarize the power results for marker-combinations belonging to each average functional allele frequency [(A) is 0.02–0.06, (B) is 0.06–0.10, (C) is 0.10–0.14, (D) is 0.14–0.16] and percent-signal category; 1 = CMC01, 2 = CMC05, 3 = SimReg0, 4 = SimReg3, 5 = SimReg4, 6 = SKATsr, 7 = VT, 8 = 2Stage, 9 = MinP.
Figure 4. Power results when casual loci have same effects under 4 causal loci out of 4 markers setting for binary trait. Power is calculated over 500 replicate data sets with $\alpha = 0.05$; there are 495 possible combinations of 4 causal loci out of 12 markers; boxplots summarize the power results for marker-combinations belonging to each average functional allele frequency [(A) is 0–0.02, (B) is 0.04–0.06, (C) is 0.08–0.11, (D) is 0.13–0.15] and percent-signal category; 1 = CMC01, 2 = CMC05, 3 = SimReg0, 4 = SimReg3, 5 = SimReg4, 6 = SKATsr, 7 = VT, 8 = 2Stage, 9 = MinP.

Figure 5. Power results when casual loci have same effects under 2 loci out of 12 markers setting for binary trait. Power is calculated over 500 replicate data sets with $\alpha = 0.05$; there are 66 possible combinations of 2 causal loci out of 12 markers; boxplots summarize the power results for marker-combinations belonging to each average functional allele frequency [(A) is 0–0.01, (B) is 0.01–0.03, (C) is 0.08–0.10, (D) is 0.16–0.26] and percent-signal category; 1 = CMC01, 2 = CMC05, 3 = SimReg0, 4 = SimReg3, 5 = SimReg4, 6 = SKATsr, 7 = VT, 8 = 2Stage, 9 = MinP.
Figure 6. Power results when casual loci have same effects under 2 loci out of 30 markers setting for binary trait. Power is calculated over 500 replicate data sets with \( \alpha = 0.05 \); there are 435 possible combinations of 2 causal loci out of 30 markers; boxplots summarize the power results for marker-combinations belonging to each average function allele frequency [(A) is 0–0.01, (B) is 0.01–0.05, (C) is 0.05–0.10, (D) is 0.10–0.20] and percent-signal category; 1 = CMC01, 2 = CMC05, 3 = SimReg0, 4 = SimReg3, 5 = SimReg4, 6 = SKATsr, 7 = VT, 8 = 2Stage, 9 = MinP.
Figure 7. Power results when casual loci have different risk effect sizes under 2 loci out of 30 markers setting for binary trait. Power is calculated over 500 replicate data sets with $\alpha = 0.05$; there are 435 possible combinations of 2 causal loci out of 30 markers; boxplots summarize the power results for marker-combinations belonging to each average functional allele frequency [(A) is 0–0.01, (B) is 0.01–0.05, (C) is 0.05–0.10, (D) is 0.10–0.20] and percent-signal category; 1 = CMC01, 2 = CMC05, 3 = SimReg0, 4 = SimReg3, 5 = SimReg4, 6 = SKATsr, 7 = VT, 8 = 2Stage, 9 = MinP.
Figure 8. Power results when casual loci have same effects under 2 loci out of 12 markers setting for quantitative trait. Power is calculated over 500 replicate data sets with $\alpha=0.05$; there are 66 possible combinations of 2 causal loci out of 12 markers; boxplots summarize the power results for marker-combinations belonging to each average functional allele frequency [(A) is 0–0.01, (B) is 0.01–0.03, (C) is 0.08–0.10, (D) is 0.16–0.26] and percent-signal category; 3 = SimReg0, 4 = SimReg3, 5 = SimReg4, 6 = SKATsr, 7 = VT, 8 = 2Stage, 9 = MinP (Note: 1 = CMC01 and 2 = CMC05 are not applicable to quantitative traits).
References


Chapter 4

Assessing RNA-seq Differential Expression Levels with Low-confidence Mapped Reads

Introduction

RNA-Seq is a high-throughput sequencing assay that can be used to identify de novo exons, genes and splicing events (e.g. Sultan et al., 2008, Trapnell et al., 2010, Filichkin et al., 2010), determine exon boundaries (e.g. Nagalakshmi, 2008), quantify gene expression level (e.g. Cloonan et al., 2008, Marioni et al. 2008, Mortazavi et al., 2008), interrogate allele-specific expression (e.g. Heap et al., 2010, Montgomery et al., 2010), identify sequence variation such as SNPs in transcribed regions (e.g. Cloonan et al., 2008), and investigate RNA editing (e.g. Rosenberg et al., 2011). It offers key advantages over other technologies including large dynamic range, low background noise (Wang et al., 2009) and highly reproducible results (Cloonan et al., 2008; Marioni et al., 2008; Mortazavi et al., 2008).

Briefly, a typical RNA-Seq experiment involves isolating messenger RNA (mRNA) from cells and converting it into a library of cDNA fragments either via DNA or RNA fragmentation. Fragments that meet a certain size specification are amplified and then sequenced by a high-throughput sequencer resulting in millions of short reads. These reads are then aligned to a reference genome. The number of reads that aligned to a gene (gene counts) can serve as a measure of expression of that gene. However, RNA-Seq technology is not without drawbacks. It is currently expensive, thus, resulting in small sample sizes (Cumbie et al., 2011; Zhou et al., 2011). There is also a transcript length bias for calling
differentially expressed genes: the number of reads for a particular transcript is proportional not only to the expression level but also the transcript length (Oshlack and Wakefield, 2009).

Our interest here lies in detecting gene-level differential expression (DE) between samples measured under two different conditions. The typical alignment tools used to generate these counts are developed specifically for the short reads produced by ultra-high-throughput sequencing, such as Bowtie and TopHat (Trapnell et al., 2009), BWA (Li and Durbin, 2009), and others. These methods tend to gain efficiency in mapping speed compared to BLAST. However, often a fair number of reads remain unaligned and are excluded from downstream analyses. For example, Li et al. (2010) observed about 47% of reads were unaligned, to a reference, both in real and simulated datasets. The percent of reads found to be unaligned in experiments summarized by Hansen et al. (2010, table S1) was less than 50%. There are many reasons that may cause reads to be unaligned including sequencing errors, polymorphisms (Wang et al., 2009), low complexity or paralogous gene families (Li et al., 2010), reads containing post-transcriptionally modified sequences (e.g., exon junction), or poor quality reference genomes (Marguerat and Bahler, 2010). Therefore, we are interested in recovering unaligned reads so that they can be included in downstream tests of DE.

We propose a method of incorporating reads that aligners tailored to short reads (we used TopHat) fail to map. We use a more general alignment algorithm, i.e., BLAST, to align reads unmapped by TopHat. Since BLAST allows a read to be aligned if only a sub-segment of that read is consistent with the reference genome, reads mapped by BLAST tend to have lower mapping stringency, possibly leading to lower confidence in the alignment of those
reads. To account for this, we assign a weight to each aligned read. The weight is a function of the aligned segment length and the number of gaps and mismatches in the alignment, and is designed to reflect the mapping confidence. This weight is then incorporated into the total counts for the expression level of a gene: each mapped read contributes a “dosage” towards the total count, determined by the weights. We construct a simple testing procedure to detect DE genes based on the weighted counts. We show that using weights improves the power to detect DE while still controlling the false discovery rate (FDR). Our results suggest that incorporating weights into existing tests of DE using RNA-seq data is likely to improve the power of those tests as well.

As part of this work, we were curious to determine why there appear to be such large proportions of reads that do not align when using aligners tailored to short reads, as such we examined the characteristics of reads that were unaligned by TopHat but were able to be successfully mapped using BLAST. We confirm a generally known observation that with current high-throughput sequencing technology, the tail regions of sequence reads tend to display lower quality base calls which can lead to problems with alignment. In addition, we also find that for a substantial number of reads, it is the beginning region of the reads that causes problems with alignment, in spite of overall high base quality scores. This phenomenon exists across several species we examined with data generated from different sequencing labs using the Illumina platform.

Methods

**Imposing quality weights to TopHap unmapped reads.** The first step in RNA-seq analysis is to map each read to either a reference transcriptome or a full genome sequence.
Then given a gene of interest, one can count the number of reads mapped to the gene for each subject. In the current practice, each read contributes to the total mapped count of a gene as 1 if it is mapped to the gene by the aligner, and as 0 otherwise (i.e., mapped to other genes or unmapped). In this paper, we refer such 0-1 counts as “unweighted count”. We propose to take the unmapped reads from the first step and align them by a less stringent alignment algorithm. In this work, we use TopHat (Trapnell et al., 2009) in the first step and use BLAST (Altschul et al., 1990) in the second step. To obtain the expression count, each mapped read, regardless from TopHat or BLAST, is assigned a weight that reflects the confidence of the mapped read. Unlike TopHat which limits the maximum number of mismatches (either over the entire read length or in the first certain bases), BLAST allows a read to be aligned even when only a sub-segment of the read is perfectly or partially matched with the reference sequence. The two aligners thus result in different levels of confidence in mapping, which motivate us to assign a quality weight to each read. Specifically, the weight is given by

\[
\frac{L_{aligned} - n_{mismatch} - n_{gap}}{L_{total}},
\]

where \(L_{total}\) is the total length of a read, \(L_{aligned}\) is the length of the aligned segment of a read, \(n_{mismatch}\) is the number of mismatches in the aligned segment, and \(n_{gap}\) is the number of gaps in the aligned segment. When a read is perfectly aligned to the reference, the weight equals 1. When the aligned segment is shorter or contains more mismatches and gaps, the weight becomes smaller.
For subject $i$, define $w_{il}$ to be the quality weight for read $l$ as defined above, and $l = 1, 2, ..., N_l$ with $N_l$ the total number of mapped reads (by either TopHat or BLAST). Also define $z_{ilj}$ to be an indicator for whether a read is mapped to gene $j$. That is, $z_{ilj}=1$ if read $l$ of subject $i$ is mapped to gene $j$, and $z_{ilj}=0$ otherwise. We then calculate the expression count of gene $j$ in subject $i$ as $y_{ij} \equiv \sum_{l=1}^{N_l} w_{il} z_{ilj}$. We refer to the proposed expression count as the “weighted count”. In contrast, the unweighted count defined earlier is given as $\sum_{l=1}^{N_l} \delta_{il} z_{ilj}$, where $\delta_{il} = 1$ if read $l$ is mapped by TopHap and 0 otherwise. To test for DE, we construct below a simple test to illustrate the utility of weighted counts.

**Characteristics of unmapped reads.** We investigated the characteristics of reads not mapped by TopHat but mapped by BLAST using four datasets. The first was a human RNA-Seq dataset obtained from the Sequence Read Archive (SRA, http://www.ncbi.nlm.nih.gov/sra) accession number SRX042195, and was sequenced using Illumina GAII (75 bp single end reads). The second dataset was based on the plant *Medicago truncatula* (*M. truncatula*) infected with *Meloidogyne hapla* (*M. hapla*) obtained from the Nielsen lab (72 bp single end reads). The third dataset was based on blood samples collected from mice infected with *Plasmodium chabaudi*, from the Nielsen lab. The final dataset was from corn infected with the fungus *Aspergillus flavus* from the Payne lab at North Carolina State University. The second, third and forth datasets were sequenced using the Illumina GAIIx. We used TopHat with the default parameters to map the reads of each dataset to the full genome sequences. We used BLAST with expectation value cutoff of $10^{-4}$ to align those reads that were not mapped by TopHat.
For each mapped read, we consider (1) the starting and (2) ending position of the aligned segment, (3) number of mismatches and (4) number of gaps in the aligned segment. We used the four factors to categorize reads (details described in the results section). We then look at the average number of unaligned bases, mismatches and gaps in each category and what percentage of reads falls into each category.

**Statistical tests to detect DE genes.** We assume $z_{lij}$ follows a Bernoulli distribution with parameter $p_{ij}$. That is, for subject $i$, read $l$ is mapped to gene $j$ with probability $p_{ij}$. To account for overdispersion, we let $p_{ij}$ follow Beta($\alpha_j, \beta_j$). When all weights are equal to one, this model posits a beta-binomial distribution on $y_{ij}$, i.e., the expression count of gene $j$, which is a common model to consider for overdispersed count data.

For each gene $j$, it is of interest to determine if there is sufficient evidence to reject the null hypothesis of no differential gene expression between the two conditions. Define $\mu_{1j}$ and $\mu_{2j}$ the mean expression levels of gene $j$ under group 1 and group 2 respectively. The DE genes can be detected by testing for the null hypothesis of $H_0: \mu_{1j} = \mu_{2j}$ vs. $H_A: \mu_{1j} - \mu_{2j} \neq 0$.

We construct the test statistics, $T_j$, by contrasting the sample mean of the expression level between the two different groups:

$$T_j = \frac{\sum_{l=1}^{n_1} \left( \frac{y_{lj}}{\sum_{l=1}^{N_i} w_{ll}} \right)}{n_1} - \frac{\sum_{l=1}^{n_2} \left( \frac{y_{lj}}{\sum_{l=1}^{N_i} w_{ll}} \right)}{n_2},$$

where $n_1$ and $n_2$ are the number of samples in the first and second treatment group respectively. $\sum_{l=1}^{N_i} w_{ll} z_{lij}$ is normalized by $\sum_{l=1}^{N_i} w_{ll}$ to account for the fact that each sample
has a different number of reads. We then standardize \( T_j \) to \( Z_j = \frac{T_j}{\sqrt{\text{Var}(T_j)}} \) so that each gene has variance 1. This is crucial to our procedure because we later pool the test statistics across all genes to obtain the p-values. The quantity \( \text{Var}(T_j) \) is obtained by:

\[
\text{Var}(T_j) = \text{Var}\left(\frac{\sum_{i=1}^{n_1} \frac{y_{ij}}{N_i} w_{il}}{n_1}\right) + \frac{\sum_{i=1}^{n_2} \frac{y_{ij}}{N_i} w_{il}}{n_2},
\]

where

\[
\text{Var}\left(\frac{\sum_{i=1}^{n_1} \frac{y_{ij}}{N_i} w_{il}}{n_1}\right) = \frac{1}{n_1^2} \sum_{l=1}^{N_i} \left(\frac{\sum_{i=1}^{n_1} \frac{y_{ij}}{N_i} w_{il}}{n_1}\right)^2 - \frac{\left(\sum_{l=1}^{N_i} w_{il}^2\right) E(p_{ij}) - \sum_{l=1}^{N_i} w_{il}^2 \left( E(p_{ij}) \right)^2}{n_1} + \left(\sum_{l=1}^{N_i} w_{il}\right)^2 \text{Var}(p_{ij}).
\]

The variance of the second term can be obtained in a similar fashion. The derivation can be found in appendix B.1. The calculation of variance term involves \( E(p_{ij}) \) and \( \text{Var}(p_{ij}) \).

Recall that \( p_{ij} \) follows Beta(\( \alpha_j, \beta_j \)); hence,

\[
E(p_{ij}) = \frac{\alpha_j}{\alpha_j + \beta_j},
\]

\[
\text{Var}(p_{ij}) = \frac{\alpha_j \beta_j}{(\alpha_j + \beta_j)^2(\alpha_j + \beta_j + 1)}.
\]

To estimate \( \alpha_j \) and \( \beta_j \), we use the following moment estimates derived in Appendix B.2:
\[
\tilde{\alpha}_j = \frac{m'}{(1 + m')^3B} - \frac{1}{(1 + m')},
\]

(1)

\[
\tilde{\beta}_j = \tilde{\alpha}_jm',
\]

(2)

where

\[
m' = \frac{1}{A} - 1,
\]

\[
A = \frac{1}{n} \sum_{i=1}^{n} \frac{y_{ij}}{\sum_{l=1}^{N_i} w_{il}},
\]

\[
B = A - \left( \frac{1}{n} \sum_{i=1}^{n} \frac{y_{ij}}{\sum_{l=1}^{N_i} w_{il}} \right)^2.
\]

Obtaining the null distribution of the test statistics. Since our test statistic has an unknown distribution, it is natural to assess the significance of the statistic by permutation. However, due to small sample size, the number of possible distinct permutations is limited. As a result, there are only restricted possible p-values and the smallest p-values can still be large. Therefore, we obtained p-values using Yang and Churchill’s approach (2007) which proposed to pool test statistics in permuted datasets across all genes. In the procedure, one first enumerates all possible permutations of the sample labels and calculates the test statistic in the permuted data. Then one calculates the first-stage p-value by counting the number of permuted test statistics pooled across all genes that are greater than or equal to the observed test statistic. This number is then divided by the total number of permutations times the number of genes. The \( \alpha \)-level critical value is then determined as the inflection point in the graph plotted between the observed test statistics (x-axis) and their first-stage p-values (y-axis). The inflection point is the point at which the growth rate of the curve is at its
maximum. Subsequently, the t-statistic is calculated for each gene, and those genes that have an absolute value of the test statistic bigger than the $\alpha$-level critical value of the t-distribution are removed. The removal of these genes is so that the pooled null distribution from experiments contains as few DE genes as possible. Based on the permuted test statistics pooled across the genes that are retained, the final p-value is the proportion of permuted statistics that are greater than or equal to the observed test statistic.

**Simulation design.** We conducted a simulation study to compare the performances of using the weighted counts and unweighted counts to detect DE genes. To capture the complexities of real data yet still be able to know the truth of differential expression, we simulated the data based on the second RNA-Seq dataset, i.e., the plant *M. truncatula* infected with *M. hapla* obtained from the Nielsen lab. Specifically, the study considered two different strains of *M. hapla* infecting *M. truncatula*, i.e., LM and VW9. Five *M. truncatula* samples were infected by strain LM and the other five samples were infected by strain VW9. The cDNA samples collected from these ten infected plants were sequenced using Illumina GAIIx. We then mapped the sequence reads to both genomes of *M. truncatula* and *M. hapla* using TopHat with the default parameters. We obtained the unweighted expression counts for each of the *M. truncatula* samples.

For the simulation study, we further filtered out genes with low expression, leaving 14,760 genes. Based on the unweighted expression counts from the real dataset, we then created a null dataset in which no systematic differential expression was expected to be present. The null dataset was created by permuting the sample labels across the two treatment groups (two samples from treatment group one were labeled as treatment group two, and
vice-versa). This effectively randomized treatment labels to samples. We then incorporated
differential gene expression signals into the null dataset in such a way as to keep the overall
mapped read count per sample identical to the original dataset. This created the power
dataset. Next, 700 genes were chosen that represented a range of expression levels from high
to low. These genes were then paired with 700 other genes with means differing from their
own by varying amounts, and expression counts were swapped within pairs for one of the
two treatment groups (Figure 1). This created a set of 1,400 genes that displayed a range of
differential expression patterns across treatment groups, but that also maintained the total
number of mapped reads per sample to be the same as the real dataset.

We also used BLAST to map those reads that were unmapped by TopHat. Then we
calculated the quality weights for all mapped reads from either TopHat or BLAST and these
weights formed a “weight distribution”. For a given gene of a given sample, we binned the
weight distributions of the mapped reads to that gene in that sample into 100 bins of bin size
0.01. In each bin, we know which read got mapped by TopHat or BLAST, so we calculate

\[ P_{ijk}^* = \frac{\text{# of read mapped by TopHat in bin}_k}{\text{total # of read in bin}_k}. \]

In addition, we count the total number of reads mapped by TopHat or BLAST in each sample (denoted by \( N_i^* \)). We also calculated the weighted expression counts in this dataset. Based on all of this information, we generated
100 replicated datasets as described below for power analysis.

In each simulation replication, we generated the read counts of gene \( j \) for subject \( i \)
from a beta-binomial distribution \( (\alpha_j^*, \beta_j^*, N_i^*) \). Specifically, for subject \( i \) we first generated
\( p_{ij}^* \) from Beta \( (\alpha_j^*, \beta_j^*) \) distribution as the probability that read \( l \) mapped to gene \( j \). We then
generated \(z_{ilj}\) from Bernoulli\((p_{ilj}^*)\) for \(l = 1\) to \(N_i^*\), i.e., \(z_{ilj} = 1\) if it is mapped to gene \(j\) and \(z_{ilj} = 0\) otherwise. We also denoted the total number of reads mapped to gene \(j\) in sample \(i\) as \(R_j^* \equiv \sum_{l=1}^{N_i^*} z_{ilj}\). We then generated \(w_{il}\) by sampling with replacement \(R_j^*\) times from the distribution of weights of reads that mapped to gene \(j\) in sample \(i\). Each read is then determined to be mapped by TopHap (i.e., \(\delta_{il} = 1\)) or BLAST (i.e., \(\delta_{il} = 0\)) using the following procedure. First, each read was assigned to a bin based on its weight value. If a read is assigned to bin \(k\), then it has probability \(P_{ijk}^*\) to be mapped by TopHat and probability \(1 - P_{ijk}^*\) to be mapped by BLAST. Such procedure is designed so that if the sampled weight has low value, it is more likely to come from BLAST. The unweighted counts are then obtained by \(\sum_{l=1}^{N_i^*} \delta_{il}z_{ilj}\), and the weighted counts were obtained by \(\sum_{l=1}^{N_i^*} w_{il}z_{ilj}\).

The values of \(\alpha_j^*\) and \(\beta_j^*\) are obtained based on the observed weighted expression counts using equations (1) and (2). Specifically, for DE genes, we obtained \((\alpha_{j, group 1}^*, \beta_{j, group 1}^*)\) by estimates from LM samples only and obtained \((\alpha_{j, group 2}^*, \beta_{j, group 2}^*)\) by estimates from VW9 samples only. To generate EE genes, we set \((\alpha_{j, group 1}^*, \beta_{j, group 1}^*) = (\alpha_{j, group 2}^*, \beta_{j, group 2}^*) \equiv (\alpha_j^*, \beta_j^*)\) which was obtained based on combined LM and VW9 samples.

**Results**

**Characteristics of unmapped reads by TopHat.** From Table 1, TopHat resulted in around 45% uniquely mapped reads with an exception for corn whose reads are mapped around 66%. BLAST uniquely mapped additional \(~12%\) of the reads, except in human where BLAST mapped as high as 43% more reads.
To understand the features of reads mapped by BLAST, we inspect those reads in four aspects: (1) Is the beginning region of the read unaligned? (2) Is the tail region of the read unaligned? (3) Are there any mismatches in the aligned segment? (4) Are there any gaps in the aligned segment? Table 2 described how we refer to all possible 16 combinations of the four aspects. For example, Characteristic 7 refers to the case that the beginning region of the read is unaligned, the tail region of the read is aligned, there are mismatches in the aligned segment, and no gap found in the aligned segment. Characteristics 1 to 4 are the cases where the beginning and tail regions are aligned. Characteristics 5-8 are those cases where the beginning region is unaligned but the tail region is aligned. Characteristics 9-12 are the cases where the beginning region is aligned but the tail region is unaligned. Finally, Characteristics 13-16 are the cases where both the beginning and tail regions are unaligned.

For each characteristic, we look at an average number of unaligned bases, mismatches and gaps. We summarized the results for each dataset (i.e., see Figures 2-5 for human, *M. truncatula*, mouse and corn respectively). In summary, mismatches occurred more frequently than gaps. There are generally 1-2 gaps but 1-6 mismatches in the aligned segment. The number of mismatches are higher when gaps are absent than when gaps are present (Characteristics 3 vs. 4). Among Characteristics 3, 7, 11 and 15 (i.e., mismatches occurred in the aligned segment but no gap), Characteristic 3 has the highest number of mismatches. For Characteristics 5-8, there are 6-26 bases unaligned at the beginning region of a read, mainly depending on datasets whether gaps and mismatches are present in the aligned segment. Generally speaking, the number of unaligned bases in the beginning region is the highest when both gaps and mismatches are absent, is the second to highest when only mismatches
are present, and is the lowest when both gaps and mismatches are present. This pattern is also observed for the number of unaligned bases in the tail region (Characteristics 9-12). Around 7-25 bases are unaligned at the tail region of a read, depending on datasets and the presence of gaps and mismatches. The pattern mentioned above occurred in Characteristics 13-16 as well when summing the number of unaligned bases in the beginning and tail region. Approximately 10-30 bases are unaligned in the beginning and tail region of the read.

Table 3 shows the percentage of reads from each characteristic in each dataset. There were about 13.43%, 31.56%, 31.87% and 30.09% of the reads for human, *M. truncatula*, mouse and corn, respectively, that have the beginning region unaligned but tail region aligned; there were about 71.84%, 34.08%, 35.34%, 32.16% of the reads that have the tail region unaligned but beginning region aligned; and there were 13.42%, 26.83%, 17.05% and 20.48% of the reads that have both beginning and tail region unaligned. This shows that the number of reads that have the beginning region unaligned is not trivial. A much lower percentage of reads have both beginning and tail region aligned: 1.31%, 7.53%, 15.74% and 17.26% in human, *M. truncatula*, mouse and corn respectively. Regardless of whether the beginning or tail region is aligned, the majority of the reads have no mismatches and no gaps, or only have mismatches (e.g., in Characteristics 5-8, where beginning of reads unaligned, reads are mostly from Characteristic 5 (no mismatches/gaps) and 7 (has mismatches only)).

**Simulation results.** We generated data as described in Methods to assess the performance between weighted and unweighted count based approaches. We adjusted for multiple-testing with the procedure of Benjamini and Hochberg (1995) at FDR of 1%, 5%, 10%, 15% and 20% (i.e., *q* = 0.01, 0.05, 0.10, 0.15 and 0.20) and calculated the empirical
FDR and true positive rate (TPR) for each dataset at those mentioned q. Empirical FDR is calculated as the total number of false positives detected divided by the total number of declared significant tests. The empirical TPR is calculated to be the total number of true positive results divided by the total number of DE genes. We averaged empirical FDR and TPR estimates across the 100 simulated datasets. We considered whether estimated FDR was controlled at each nominal FDR for weighted and unweighted counts and compared the power (TPR) between weighted and unweighted counts at each nominal FDR.

**False discovery rate.** An averaged empirical FDR across 100 datasets for both weighted and unweighted approaches at $q = 0.01, 0.05, 0.10, 0.15$ and $0.20$ are shown in table 4. Both methods show control of the FDR although they are slightly conservative. However, the weighted count method tends to be closer to the nominal level than the unweighted method.

**Power to detect DE genes.** We considered an averaged TPR across 100 datasets between using weighted and unweighted counts at $q = 0.01, 0.05, 0.10, 0.15$ and $0.20$ as shown in table 5. The weighted count approach has higher power than unweighted at all $q$ level although the difference in power is decreasing as $q$ level increases which is expected. In summary, using the weighted count approach detected additional DE genes around 9.56%, 8.25%, 7.44%, 7.07% and 6.57% at $q$ stated above.

We further considered per-gene comparisons between the weighted and unweighted methods at $q = 0.05$. The genes were classified into 5 types based on whether they are EE or DE and based on the testing results from the weighted and unweighted approaches. (See Figures 6 and 7 for weighted and unweighted counts respectively.) In this comparison, a
gene is identified as DE if it is significant at $q = 0.05$ in more than 80 out of 100 simulations. Both methods are able to detect most DE genes that either have high foldchanges with moderate to high expression, or moderate foldchanges and high expression levels (yellow dots in Figure 6 and 7). Both methods cannot detect DE genes that have low foldchanges or low expression levels (green dots in Figures 6 and 7). The DE genes that can only be detected by the weighted count approach tend to be those genes with moderate foldchanges and moderate expression levels, or those genes with high foldchanges and low expression levels (blue dots in Figures 6 and 7).

The red dots indicate the DE genes that are missed by the weighted method but detected by the unweighted method. Comparing the position of the red dots in Figures 6 and 7, we can see these DE genes have lower foldchanges when measured by weighted counts than by the unweighted methods (i.e., the red dots shifted towards to the dashed line from unweighted counts (Figure 7) to weighted counts (Figure 6)). The position shift is also observed in some of the blue dots, but the magnitude is not as prominent as the red dots. The blue dots slightly shift downward to the dashed line and to the left from Figure 6 to Figure 7, indicating using weighted counts can increase foldchanges and the mean counts. Figure 8 contrasts the foldchanges obtained by the weighted and unweighted methods. The points fall along the diagonal line indicate similar foldchanges between the weighted and unweighted counts. Corresponding to what we observed in Figures 6 and 7, we see the red dots clustering around the vertical dashed line (i.e., 0 foldchanges by the weighted counts). We also see that quite a few blue dots deviate from the diagonal line and shifting further toward the horizontal
dashed line, indicating the foldchanges are reduced by the unweighted counts compared to weighted counts.

Figure 9 contrasts the mean counts obtained by the weighted and unweighted methods. As expected, most of the dots are along or on the right side of the diagonal line. This matches with the impression that using the weighted counts tends to increase the total number of expression counts. Furthermore, most of the dots falling on the right side of the diagonal line are blue dots. This also agrees with the previous observation that blue dots in Figure 7 slightly shift from left to right in Figure 6. For the blue dots that barely deviate from the diagonal line in both Figures 8 and 9, using weighted counts can detect these DE genes because the weighted expression counts are more homogeneous among replicates within each treatment than unweighted counts.

Gene counts using weighted vs. unweighted based approaches result in different foldchange and mean expressions, and the magnitude of the difference varies from gene to gene. For some DE genes, using unweighted counts increases foldchange while for other DE genes using weighted counts increases foldchange and mean expression. However, using weighted counts resulted in increased foldchange and mean expression much more than using unweighted counts for DE genes. For DE genes for which foldchange and mean expression are similar between weighted and unweighted count procedures, using unweighted counts yields a more heterogeneous count within treatment.

Discussion

A substantial number of unaligned reads observed from the data in our lab, Li et al. (2010) and many experiments summarized by Hansen et al. (2010) in table S1 have
motivated us to attempt to include those unmapped reads as much as possible in testing for DE genes. Specifically, we use BLAST to align those unmapped reads and assign a weight to each uniquely mapped read. Gene counts are the sum of weights of reads that match to a gene (weighted counts). We verify the utility of employing less confidence mapped reads and getting gene counts via weight by constructing a statistical method that considers weight in order to get the proper variance estimate. We showed that using weighted counts resulted in an appreciably higher power than using unweighted counts though simulation. Although, our results are based on simulated data only, we believe these results can be conjectured on real data as we simulated them based on real data. The weighted count approach yields higher power than the unweighted count approach because it increases mean expressed value and foldchange, and creates more homogeneous read counts within the treatment group.

To focus on the contribution from using weight, we used only uniquely mapped reads to test for DE genes. However, multi-match reads can also be included as in e.g., Cloonan et al. (2008), Faulkner et al. (2008) and Mortazavi et al. (2008) in a way that fractionally assigns a weight to genes. Moreover, any aligners, not restricted to TopHat and BLAST, can be used to map reads.

The weight we used was subtracting the number of gaps and mismatches from the length of the aligned chunk and dividing by total read length. Although mismatches could be due to polymorphisms or sequencing errors, in our case, it is solely due to sequencing errors since *M. truncatula* are genetically homogeneous. In other experiments, weights could be calculated in different ways to better suit the data e.g., subtract the number of mismatches - 1 from aligned chunk length to allow for polymorphism. We subtract the number of gaps from
the aligned chunk length; this means we penalized gap opening and gap extension equally. Weights can be changed to have smaller gap extension penalty than gap opening penalty. In addition, if multi-match reads are included, weights can also be computed as a function of the number of locations to which the read mapped, in addition to using mismatches, gaps, aligned chunk length and read length.

In summary, we introduced a weighting scheme that reflects the mapping confidence of each read and demonstrated that incorporating reads unmapped by aligners tailored to short reads can improve the power to detect differentially expressed genes while keeping the FDR at a desired level.
### Tables

**Table 1. Percentage of reads mapped by TopHat and BLAST**

<table>
<thead>
<tr>
<th></th>
<th>Number of reads uniquely mapped by TopHat</th>
<th>Percent of reads uniquely mapped by TopHat</th>
<th>Number of reads uniquely mapped by BLAST</th>
<th>Percent of reads uniquely mapped by BLAST</th>
<th>Total number of reads uniquely mapped by both</th>
<th>Total percent of reads uniquely mapped by both</th>
<th>Total number of reads</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human</td>
<td>14,184,336</td>
<td>43.76</td>
<td>14,019,319</td>
<td>43.25</td>
<td>28,203,655</td>
<td>87.01</td>
<td>32,415,405</td>
</tr>
<tr>
<td>LM5</td>
<td>13,235,971</td>
<td>47.86</td>
<td>3,739,001</td>
<td>13.52</td>
<td>16,974,972</td>
<td>61.38</td>
<td>27,654,736</td>
</tr>
<tr>
<td>Mouse</td>
<td>18,417,445</td>
<td>45.04</td>
<td>4,987,830</td>
<td>12.20</td>
<td>23,405,275</td>
<td>57.23</td>
<td>40,895,799</td>
</tr>
<tr>
<td>Corn</td>
<td>39,474,819</td>
<td>66.17</td>
<td>8,095,351</td>
<td>13.57</td>
<td>47,570,170</td>
<td>79.74</td>
<td>59,656,885</td>
</tr>
</tbody>
</table>

**Table 2. Characteristics of reads unmapped by TopHat but mapped by BLAST categorized into 16 types.**

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>beginning region of read unaligned?</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>tail region of read unaligned?</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>mismatches in the aligned segment?</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>gap in the aligned segment?</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
</tbody>
</table>
Table 3. Proportion of reads from each characteristic in human, *M. truncatula*, mouse and corn

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Percent of reads having each characteristics in</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Human</td>
</tr>
<tr>
<td>1</td>
<td>0.64</td>
</tr>
<tr>
<td>2</td>
<td>0.22</td>
</tr>
<tr>
<td>3</td>
<td>0.37</td>
</tr>
<tr>
<td>4</td>
<td>0.07</td>
</tr>
<tr>
<td>5</td>
<td>11.58</td>
</tr>
<tr>
<td>6</td>
<td>0.03</td>
</tr>
<tr>
<td>7</td>
<td>1.81</td>
</tr>
<tr>
<td>8</td>
<td>0.01</td>
</tr>
<tr>
<td>9</td>
<td>54.25</td>
</tr>
<tr>
<td>10</td>
<td>0.17</td>
</tr>
<tr>
<td>11</td>
<td>17.32</td>
</tr>
<tr>
<td>12</td>
<td>0.11</td>
</tr>
<tr>
<td>13</td>
<td>10.38</td>
</tr>
<tr>
<td>14</td>
<td>0.01</td>
</tr>
<tr>
<td>15</td>
<td>3.01</td>
</tr>
<tr>
<td>16</td>
<td>0.01</td>
</tr>
</tbody>
</table>

Table 4. Estimated FDRs based on an average across 100 simulated datasets at five nominal levels

<table>
<thead>
<tr>
<th>Weighted Count</th>
<th>Unweighted Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>( q = 0.01 )</td>
<td>0.009</td>
</tr>
<tr>
<td>( q = 0.05 )</td>
<td>0.044</td>
</tr>
<tr>
<td>( q = 0.10 )</td>
<td>0.087</td>
</tr>
<tr>
<td>( q = 0.15 )</td>
<td>0.132</td>
</tr>
<tr>
<td>( q = 0.20 )</td>
<td>0.176</td>
</tr>
</tbody>
</table>
Table 5. Power based on an average TPR across 100 simulated datasets at five nominal levels

<table>
<thead>
<tr>
<th>$q$</th>
<th>Weighted Count</th>
<th>Unweighted Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>0.579</td>
<td>0.483</td>
</tr>
<tr>
<td>0.05</td>
<td>0.690</td>
<td>0.607</td>
</tr>
<tr>
<td>0.10</td>
<td>0.736</td>
<td>0.661</td>
</tr>
<tr>
<td>0.15</td>
<td>0.764</td>
<td>0.693</td>
</tr>
<tr>
<td>0.20</td>
<td>0.783</td>
<td>0.718</td>
</tr>
</tbody>
</table>
Figure 1. An illustration of how differentially expressed genes are created. The color gradients represent expression levels. Each row represents genes and each column represents samples where the first five columns are samples in group 1 and the other five columns are samples in group 2. Expression levels of samples in group 1 in gene pairs marked by a circle are swapped. Expression levels of samples in group 2 in gene pairs marked by a square are swapped. This creates 4 DE genes.
Figure 2. Characteristics of reads mapped by BLAST in human
Figure 3. Characteristics of reads mapped by BLAST in *M. truncatula*
Figure 4. Characteristics of reads mapped by BLAST in mouse
Figure 5. Characteristics of reads mapped by BLAST in corn
Figure 6. Testing for differential expression using weighted counts. Y-axis is \( \log_2 \) of fold change and x-axis is \( \log_2 \) of mean normalized expression, averaged across 100 simulated datasets. Each dot represents one gene. Grey dots represent EE genes that neither method calls significant. Yellow are DE genes identified by both methods. Blue are DE genes identified by only the weighted method. Red are DE genes identified by only the unweighted method. Green are DE genes that no method identified. Genes are said to be identified if they are significant in more than 80 out of 100 simulations.
Figure 7. Testing for differential expression using unweighted counts. Y-axis is log$_2$ of foldchange and x-axis is log$_2$ of mean normalized expression, averaged across 100 simulated datasets. Each dot represents one gene. Grey dots represent EE genes that neither method calls significant. Yellow are DE genes identified by both methods. Blue are DE genes identified by only the weighted method. Red are DE genes identified by only the unweighted method. Green are DE genes that no method identified. Genes are said to be identified if they are significant in more than 80 out of 100 simulations.
Figure 8. log$_2$ of fold change between weighted and unweighted counts. Y-axis is log$_2$ of fold change of unweighted count and x-axis is log$_2$ of fold change of weighted count, averaged across 100 simulated datasets. Each dot represents one gene. Grey dots represent EE genes that neither method calls significant. Yellow are DE genes identified by both methods. Blue are DE genes identified by only the weighted method. Red are DE genes identified by only the unweighted method. Green are DE genes that no method identified. Genes are said to be identified if they are significant in more than 80 out of 100 simulations.
Figure 9. log$_2$ of mean normalized expression between weighted and unweighted counts. Y-axis is log$_2$ of mean normalized expression of unweighted counts and x-axis is log$_2$ of mean normalized expression of weighted counts, averaged across 100 simulated datasets. Each dot represents one gene. Grey dots represent EE genes that neither method calls significant. Yellow are DE genes identified by both methods. Blue are DE genes identified by only the weighted method. Red are DE genes identified by only the unweighted method. Green are DE genes that no method identified. Genes are said to be identified if they are significant in more than 80 out of 100 simulations.
References


Chapter 5

Concluding Remarks

The reliability and replicability of the results of association studies depends in part on quality of data. SNP genotyping technologies are not without error. Even if the error rate is modest, with large numbers of SNPs and samples, genotyping error can be influential. Differential bias in genotyping can affect power and inflate type I error. A common practice in SNP quality control is applying expert filters based on Hardy–Weinberg equilibrium, missing proportion, and minor allele frequency to exclude low-quality SNPs. However, the implementation of these filters requires arbitrary thresholds and does not jointly consider all QC features. In chapter 2, we proposed an algorithm to detect low-quality SNPs that automates the QC threshold determination, allows a collective consideration of the QC features and gives conditional thresholds contingent on other QC variables (e.g. different MSP thresholds for different MAFs). The algorithm is based on the premise that the majority of SNPs have sufficient genotyping quality with QC variable values in certain directions (e.g. low MSP and non-low MAF). SNPs with QC values deviating from the majority are considered outliers and are then labeled as problematic SNPs. The algorithm applies PCA on the QC variables in order to separate good SNPs and problematic SNPs on a two-dimensional plane then uses DBSCAN to identify the borders of good SNPs, which can be translated to a QC threshold. We apply our method to the seven studies from the Wellcome Trust Case Control Consortium and the major depressive disorder study from the Genetic Association Information Network in order to evaluate the performance of the proposed algorithm and
demonstrate its utility. The criteria we used to measure our method’s performance against expert filters are: (i) percentage of SNPs excluded due to low quality; (ii) inflation factor of the test statistics ($\lambda$); (iii) number of false associations found in the filtered dataset; and (iv) number of true associations missed in the filtered dataset. The results suggest that with the same or fewer SNPs excluded, the proposed algorithm has comparable or better performance than the expert filter for all diseases. The same QC filtering principle can be generalized from GWAS SNP to DNA sequencing data. For next generation sequencing data, the QC variables include combined depth across samples, strand bias evidence, total count of mapping quality zero reads across all samples, largest contiguous homopolymer run of the variant allele in either direction on the reference, and variant confidence. PCA can be performed based on these QC variables, and then followed by DBSCAN to identify a subset of good quality data. However, further evaluations are needed.

It is speculated that rare variants contribute to the missing heritability from GWAS. With direct sequencing, we can identify rare variants. The single-marker approach is underpowered in detecting rare variants. In contrast, collapsing methods are gaining momentum due to their usefulness in marker-set analysis and rare variant detection. These methods aggregate information across several markers and assess their collective effect on the phenotype. The information among markers can be collapsed at the genotype level or similarity level. At the genotype level, information can be collapsed by calculating a weighted sum of the genotypes across all markers e.g., Li & Leal (2008), Madsen & Browning (2009), Han & Pan (2010), Price et al. (2010). At the similarity level, information can be collapsed by quantifying the genetic similarity across all markers for each pair of
unrelated individuals e.g., Wessel & Schork (2006), Tzeng et al. (2009,2011), Mukhopadhyay et al. (2010), Wu et al. (2010,2011). Understanding the strengths and weaknesses of the two collapsing paradigms helps researchers select a suitable approach for their analysis. In chapter 3, we investigated the strengths and weaknesses of genotype collapsing and similarity collapsing over a wide range of plausible scenarios through simulations. The factors that we examined included (a) the underlying genetic architecture of the causal variants (i.e., effect size, frequency, and number of causal alleles within a causal locus), (b) composition of the variant set (i.e., proportion of causal variants in the set and LD between causal and non-causal loci in the set), and (c) the weighting scheme used in the collapsing method. Our results show that neither collapsing strategy outperforms the other across all simulated scenarios. Similarity-level collapsing has less power loss due to the inclusion of non-causal variants in the marker set than genotype-level collapsing. Genotype-level collapsing generally performs best when the genetic architecture of the causal variants is not complex. That is, the causal variants have similar, additive, linear effects with similar frequencies. Similarity collapsing outperforms genotype collapsing when the genetic architecture of the marker set becomes more complex such as a mixture of rare and non-rare variants, different effect sizes and directions, and multiple causal alleles within a locus.

The arrival of RNA-Seq has transformed gene expression data from analog fluorescence intensities to digital read counts. It generates millions of short reads; thus the first step of any data analysis is mapping reads to a reference genome or de novo assembly. The alignment step has numerous challenges: reads may contain sequencing errors or polymorphisms, reads may be from low complexity regions or from a paralogous gene
family, reads may contain post-transcriptionally modified sequences, or the reference genome may have poor quality. These problems can result in multi-match reads or unmapped reads. Several methods have been proposed to test for differentially expressed genes using RNA-Seq count data. The key thing that they all focus on is how to properly model the count data to tackle the overdispersion problem. However, those methods use only the reads mapped by aligners tailored to short reads e.g. TopHat, BWA, and discard a non-trivial percentage of (unmapped) reads. In chapter 4, we proposed incorporating these unmapped reads in testing for differential gene expression. Specifically, we used BLAST, a more general aligner, to align those unmapped reads and assign each mapped read a weight, which reflects our confidence in the mapping. The gene expression level changes from a count of the number of reads hitting a gene to a sum of the weights of reads hitting a gene. In order to evaluate the utility of the proposed concept, we construct a statistical method and perform simulations based on real data. Our results show that using the proposed concept improves power in detecting differentially expressed genes while still controlling for false discovery rate. Extending the proposed concept to the existing methods is likely to improve the power of those tests as well.
References


Appendix
Appendix A

The Change-Point Model for Estimating r

A typical sorted K-th NN graph is shown in Figure A1a. Given the wide range of the distance, it would be more stable to fit the change-point model to the log transformation of the distance, as shown in Figure A1b. After the log transformation, there are two change points, and our focus is on the right one. The change point model that we consider uses two linear lines to approximate the data points around the change point (Fig. A1c). Let \( y \) be the log of the fourth NN distance, \( x \) be the (distance-sorted) SNP ID, and \( x^* \) be the change point on the X-axis. The two linear models are

\[
\begin{align*}
  y_i &= \alpha_0 + \beta_0 x_i + e_i \quad \text{for } x_i < x^* \quad \text{and} \\
  y_i &= \alpha_1 + \beta_1 x_i + e_i \quad \text{for } x_i > x^*,
\end{align*}
\]

where \( e_i \sim N(0, \sigma^2) \) and \( \alpha_0 + \beta_0 x^* = \alpha_1 + \beta_1 x^* \sigma_0 + \beta_0 \) (or equivalently \( \alpha_1 = \alpha_0 + (\beta_0 - \beta_1) x^* \)). A normal likelihood is then specified and optimized to obtain the maximum likelihood estimates \( \hat{\alpha}_0, \hat{\beta}_0, \hat{\beta}_1, \hat{\sigma} \) and \( \hat{x}^* \). Then the change-point r value is the distance value on the Y-axis corresponding to \( \hat{x}^* \).
Figure A1. The sorted fourth NN distance for CAD. (a) The distance on the original scale. (b) The distance on the log scale. (c) The distance on the log scale (dashed line) superimposed with the fit from the change-point model (solid line) as described in the Appendix A1. The change-point r is indicated by the dotted horizontal line.
Appendix B

B.1 The derivation of

\[
Var\left( \frac{\sum_{l=1}^{n_1} \frac{\sum_{i=1}^{N_i} w_{ij} z_{ilj}}{\sum_{i=1}^{N_i} w_{ij}}}{n_1} \right) = \frac{1}{n_1^2} \sum_{i=1}^{N_i} \left( \frac{1}{(\sum_{i=1}^{N_i} w_{ij})^2} \right) Var\left( \sum_{l=1}^{n_1} w_{ij} z_{ilj} \right)
\]

We first calculate \(Var(\sum_{l=1}^{N_i} w_{ij} z_{ilj})\). We have

\[
Var\left( \sum_{l=1}^{N_i} w_{ij} z_{ilj} \right) = E(Var\left( \sum_{l=1}^{N_i} w_{il} z_{ilj} | p_{ij} \right)) + Var(E(\sum_{l=1}^{N_i} w_{il} z_{ilj} | p_{ij})). \tag{1}
\]

In the first term of equation (1), we have

\[
Var\left( \sum_{l=1}^{N_i} w_{il} z_{ilj} | p_{ij} \right) = \sum_{l=1}^{N_i} Var\left( w_{ij} z_{ilj} \right)\]

\[
\therefore Var\left( w_{ij} z_{ilj} | p_{ij} \right) = w_{il}^2 Var\left( z_{ilj} | p_{ij} \right)
\]

\[
= w_{il}^2 \left( 1 - p_{ij} \right)
\]

\[
= w_{il}^2 p_{ij} - w_{il}^2 p_{ij}^2
\]

\[
\therefore Var\left( \sum_{l=1}^{N_i} w_{il} z_{ilj} | p_{ij} \right) = \sum_{l=1}^{N_i} \left( w_{il}^2 p_{ij} - w_{il}^2 p_{ij}^2 \right).
\]

Hence, the first term of equation (1) becomes,

\[
E(Var\left( \sum_{l=1}^{N_i} w_{il} z_{ilj} | p_{ij} \right)) = E\left( \sum_{l=1}^{N_i} \left( w_{il}^2 p_{ij} - w_{il}^2 p_{ij}^2 \right) \right)
\]

\[
= E\left( \sum_{l=1}^{N_i} w_{il}^2 p_{ij} \right) - \sum_{l=1}^{N_i} w_{il}^2 p_{ij}^2
\]

Hence, the first term of equation (1) becomes,
\[= \sum_{l=1}^{N_i} w_{il}^2 E(p_{ij}) - \sum_{l=1}^{N_i} w_{il}^2 E(p_{ij}^2).\]

In the second term of equation (1), we have

\[E\left( \sum_{l=1}^{N_i} w_{il} z_{ilj} | p_{ij} \right) = \sum_{l=1}^{N_i} E(w_{il} z_{ilj} | p_{ij}) \]
\[= \sum_{l=1}^{N_i} w_{il} E(z_{ilj} | p_{ij}) \]
\[= \sum_{l=1}^{N_i} w_{il} p_{ij}.\]

Hence, the second term of equation (1) becomes,

\[Var\left( \sum_{l=1}^{N_i} w_{il} z_{ilj} | p_{ij} \right) = Var\left( \sum_{l=1}^{N_i} w_{il} p_{ij} \right) \]
\[= \left( \sum_{l=1}^{N_i} w_{il} \right)^2 Var(p_{ij}).\]

Hence equation (1) is

\[Var\left( \sum_{l=1}^{N_i} w_{il} z_{ilj} \right) = \sum_{l=1}^{N_i} w_{il}^2 E(p_{ij}) - \sum_{l=1}^{N_i} w_{il}^2 E(p_{ij}^2) + \left( \sum_{l=1}^{N_i} w_{il} \right)^2 Var(p_{ij}) \]
\[= \sum_{l=1}^{N_i} w_{il}^2 E(p_{ij}) - \sum_{l=1}^{N_i} w_{il}^2 E(p_{ij}^2) + \left( \sum_{l=1}^{N_i} w_{il} \right)^2 Var(p_{ij}) + \sum_{l=1}^{N_i} w_{il}^2 (E(p_{ij}))^2 \]
\[= \sum_{l=1}^{N_i} w_{il}^2 (E(p_{ij}))^2\]
\[
\begin{align*}
&= \sum_{l=1}^{N_i} w_{ll}^2 \mathbb{E}(p_{ij}) - \sum_{l=1}^{N_i} w_{ll}^2 (\mathbb{E}(p_{ij}))^2 + \left( \sum_{l=1}^{N_i} w_{il} \right)^2 \text{Var}(p_{ij}) \\
&- \sum_{l=1}^{N_i} w_{ll}^2 [\mathbb{E}(p_{ij}^2) - (\mathbb{E}(p_{ij}))^2] \\
&= \sum_{l=1}^{N_i} w_{ll}^2 \mathbb{E}(p_{ij}) - \sum_{l=1}^{N_i} w_{ll}^2 (\mathbb{E}(p_{ij}))^2 + \left( \sum_{l=1}^{N_i} w_{il} \right)^2 \text{Var}(p_{ij}) \\
&- \sum_{l=1}^{N_i} w_{ll}^2 \text{Var}(p_{ij}) \\
&= \sum_{l=1}^{N_i} w_{ll}^2 \mathbb{E}(p_{ij}) - \sum_{l=1}^{N_i} w_{ll}^2 (\mathbb{E}(p_{ij}))^2 + \left( \sum_{l=1}^{N_i} w_{il} \right)^2 - \sum_{l=1}^{N_i} w_{ll}^2 \text{Var}(p_{ij}).
\end{align*}
\]

Hence,

\[
\text{Var} \left( \frac{\sum_{l=1}^{N_i} \sum_{l=1}^{N_i} w_{il}^2 w_{ll}}{n_1} \right) = \frac{1}{n_1^2} \sum_{l=1}^{N_i} \frac{1}{\sum_{l=1}^{N_i} w_{ll}} \left[ \sum_{l=1}^{N_i} w_{ll}^2 \mathbb{E}(p_{ij}) - \sum_{l=1}^{N_i} w_{ll}^2 (\mathbb{E}(p_{ij}))^2 + \left( \sum_{l=1}^{N_i} w_{il} \right)^2 - \sum_{l=1}^{N_i} w_{ll}^2 \text{Var}(p_{ij}) \right].
\]
B.2 The derivation of the moment estimates for \( \alpha_j \) and \( \beta_j \).

Define A and B to be the first and second sample moments:

\[
A = \frac{1}{n} \sum_{i=1}^{n} \left( \frac{y_{ij}}{\sum_{l=1}^{N_l} w_{il}} \right),
\]

\[
B = \frac{1}{n} \sum_{i=1}^{n} \left( \frac{y_{ij}}{\sum_{l=1}^{N_l} w_{il}} \right)^2 - A^2.
\]

where \( n \) is number of samples and \( y_{ij} = \sum_{l=1}^{N_l} w_{il} z_{ilj} \). The moment estimates, denoted by \( \hat{\alpha}_j \) and \( \hat{\beta}_j \), were obtained by equating the sample moments and population moments.

Specifically,

\[
A = E \left( \frac{y_{ij}}{\sum_{l=1}^{N_l} w_{il}} \right) = \frac{\hat{\alpha}_j}{\hat{\alpha}_j + \hat{\beta}_j}, \tag{1}
\]

\[
B = Var \left( \frac{y_{ij}}{\sum_{l=1}^{N_l} w_{il}} \right) = \frac{\hat{\alpha}_j \hat{\beta}_j}{(\hat{\alpha}_j + \hat{\beta}_j)^2 (\hat{\alpha}_j + \hat{\beta}_j + 1)}. \tag{2}
\]

Solve for \( \hat{\alpha}_j \) and \( \hat{\beta}_j \) from equations (1) and (2). By rearranging equation (1), we have

\[
A\hat{\alpha}_j + A\hat{\beta}_j = \hat{\alpha}_j
\]

\[
A\hat{\alpha}_j - \hat{\alpha}_j = -A\hat{\beta}_j
\]

\[
\therefore \hat{\beta}_j = \frac{A\hat{\alpha}_j - \hat{\alpha}_j}{-A} = \hat{\alpha}_j \left( \frac{1}{A} - 1 \right) = \hat{\alpha}_j m',
\]

where \( m' = \frac{1}{A} - 1 \).
Rearranging the above equation and we have

\[
(\hat{\alpha}_j (1 + m') + 1) = \frac{m'}{(1 + m')^2 B}
\]

\[
\hat{\alpha}_j (1 + m') = \frac{m'}{(1 + m')^2 B} - 1
\]

\[
\therefore \hat{\alpha}_j = \frac{m'}{(1 + m')^3 B} - \frac{1}{(1 + m')^3}
\]