ABSTRACT

Title of the Dissertation: ALGORITHMS FOR GERMLINE GENOTYPING, METHYLATION DECONVOLUTION, AND SOMATIC PHYLOGENY RECONSTRUCTION

Ananth Hari
Doctor of Philosophy, 2025

Dissertation Directed by: Dr. Uzi Vishkin
Department of Electrical and Computer Engineering
Dr. S. Cenk Sahinalp
National Cancer Institute

One of the fundamental problems in nature is to assemble small blocks into known patterns, using a large reference set of known patterns. The complexity of the problem is at least an order of magnitude higher if the reference set is not fully known.

Genotyping germline genes involves assembling sequencing reads into known sequences of genes known as alleles and determining their number of copies in the genomes of individuals. The first part of the thesis presents methods to genotype complex and highly polymorphic germline loci. First, we present ImmunoTyper-SR, an algorithmic approach to genotype and analyze copy numbers of the germline immunoglobulin and T cell receptor genes using Illumina whole genome sequencing (WGS) data. ImmunoTyper-SR is based on a novel combinatorial optimization formulation that aims to minimize the total edit distance between reads and their assigned IGH alleles from a given database, with constraints on the number and distribution of

reads across each called allele. We also present Aldy 4, a fast and highly accurate combinatorial optimization method, to genotype a large set of genes responsible for drug metabolism.

The second part of the dissertation focuses on inferring and interpreting data originating from evolutionary processes in somatic cells. DNA methylation is the biological process by which methyl groups are added to certain nucleotides in the DNA that dictate the activity of genes. We present Qombucha, a method for deconvolving bulk DNA methylation data from patient samples into constituent proportions of known cell types. Since the reference cell profiles are only partially known, we also simultaneously fill in the gaps in the methylation values of the representatives of the cell types.

Finally, we present a work in progress of the analysis of phylogenetic evolution of B cell receptor sequences. When the innate immune system fails to destroy pathogenic invaders, various immunoglobulin genes combine to form naive B cell receptors and rapidly accumulate mutations to greatly increase the binding affinity of the antibodies. We survey the existing literature and describe currently available computational tools, their assumptions and limitations and propose various novel combinatorial optimization formulations to solve distinct variants of the B cell receptor phylogeny inference problem.

ALGORITHMS FOR GERMLINE GENOTYPING, METHYLATION DECONVOLUTION, AND SOMATIC PHYLOGENY RECONSTRUCTION

by

Ananth Hari

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2025

Advisory Committee:
Professor Uzi Vishkin, Chair/Advisor
Dr. S. Cenk Sahinalp, Co-Advisor
Professor Behtash Babadi
Professor Richard La
Professor Philip Johnson
Professor Can Firtina

Acknowledgments

I owe my sincere thanks to all those who have made this dissertation possible and it would not have been possible without even a single cog in this well-oiled machine, it would not have been possible.

First and foremost I'd like to thank my advisor, Professor Uzi Vishkin for giving me an invaluable opportunity to work on challenging and interesting projects over the past four years. He has been extremely quick to respond to requests even at odd hours. It has been a pleasure to work with and learn from such an extraordinary individual who has gathered large swathes of knowledge in multiple areas of research.

I would also like to thank my co-advisor, Dr. Cenk Sahinalp. His ability to match patterns and suggest established methods to solve novel problems is second to none. His ability to explain complex theoretical ideas in simple terms is exceptional. He has been a great mentor through all of years as a Fellow at the NIH.

Thanks are due to Professor Steve Mount, Professor Richard La, Professor Behtash Babadi, Professor Philip Johnson, and Professor Can Firtina for agreeing to serve on my dissertation committee and for sparing their invaluable time reviewing the manuscript and suggesting edits.

My colleagues and friends at the Cancer Data Science Laboratory (CDSL) have enriched my graduate life in many ways and deserve a special mention. My co-authors, Mike Ford, Cindy Li, and Alejandro Schäffer, have been brilliant assets to all projects and their inputs, ideas, and

problem solving abilities have been invaluable. Chih Hao Wu has been my closest friend through my tenure at CDSL and I wish we had worked on projects together. His extremely high level of knowledge on various biological mechanisms, intuitiveness at solving complex problems with noisy data, and software engineering skills are great assets to have in a project partner. Special mentions to Chih Hao, John Bridgers, and Ataberk who have tirelessly ferried me and other colleagues to the NIH campus. Friends such as John, Yuelin, MG, and Ataberk have definitely made life outside of the office very rewarding. Finally, an honorable mention to Simit, our lab mascot, for keeping the morale high and being one's literal best friend (if you have food!).

I would also like to acknowledge help and support from all the staff members in the graduate studies office, International Students Services, and CDSL, NCI. Emily, Souad, Maria, Tiffany, Ashley, Catha, and Bijoya have helped me carefully navigate the system to ensure all i's are dotted and all t's are crossed.

I owe my deepest thanks to my mother and father, who have always stood by me and guided me through my career. I owe a huge debt of gratitude to my brother and sister-in-law, who have been my closest confidants and mentors.

I would like to acknowledge financial support from the NCI-UMD Partnership Program and the Office of Intramural Research at the NIH that are responsible for all projects discussed in this dissertation.

Lastly, I want to thank my wife, Alekhya, for trusting me, putting faith in me, and supporting me in every way she can just to see me succeed. I love you so so much.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AIRR	Adaptive Immune Receptor Repertoire
AIRR-seq	Adaptive Immune Receptor Repertoire Sequencing
BCR	B cell Receptor
BCR-seq	B cell Receptor Sequencing
IG	Immunoglobulin (germline locus)
IGH	Immunoglobulin Heavy Chain
IGHC	Immunoglobulin Heavy Chain Constant region
IGHD	Immunoglobulin Heavy Chain Diversity region
IGHJ	Immunoglobulin Heavy Chain Joining region
IGHV	Immunoglobulin Heavy Chain Variable region
TCR	T cell Receptor
TR	T cell Receptor (germline locus)
WGS	Whole Genome Sequencing

Glossary

Allele	A variant form of a gene that differs from other alleles in sequence composition
Antigen	Molecule/allergen that can bind to an antibody or TCR
B cell	Produces antibody molecules
B cell receptor (BCR) (Antibody/Immunoglobulin)	A molecule translated from rearranged/recombined sequence of IG genes that binds to antigens. Attached to the cell surface of a B cell
BCR-seq	Sequencing technology that outputs pieces of all expressed BCR sequences in the form of reads
Clonotype	A specific germline V(D)J recombination that results in multiple evolved BCR sequences
Gene	Substring of the genome sequence. Basic unit of heredity.
Genome	3 billion character-long string (from $\{A, C, G, T\}$) containing all genetic information
Germline	Genes/alleles/traits inherited from parents
Immunoglobulin locus (IG) (Germline)	Germline locus that contains all genes that code for BCRs
IGH	Immunoglobulin Heavy Chain region
IGHV	Immunoglobulin Heavy Chain region Variable gene
IGHD	Immunoglobulin Heavy Chain region Diversity gene
IGHJ	Immunoglobulin Heavy Chain region Joining gene
Multiple Sequence Alignment (MSA)	Alignment of > 2 sequences that puts all characters of all sequences in a matrix, where each row has a constant length
Psuedogene	Genomic substring that looks like a gene but non-functional
Somatic mutations	Changes to genetics that are not passed through germline
T cell	Many subtypes exist that perform various immune-related tasks from activating B cells to killing off infected cells
T cell receptor (TCR)	Molecule attached to the cell surface of a T cell
V(D)J recombination	The process by which an unmutated BCR sequence emerges, by choosing one gene each from all *IGHV*, *IGHD*, and *IGHJ* genes

Chapter 1: Introduction

Over several billions of years, life on earth evolved from single-cell organisms to multicellular organisms, including vertebrates such as humans. A part of this evolution includes development of robust immune systems that not only fight against invading pathogens but also regulate cells within the organism by inducing apoptosis (programmed cell death) [3].

The mammalian immune system protects the organism from abundant infectious microbes (high *sensitivity*), while avoiding responses that produce excessive damage of host's tissues (high *specificity*) [4]. The immune system is complex and is divided in two categories: (i) the **innate** (or nonspecific) immunity, which consists of the activation and participation of preexistent mechanisms; and (ii) the **adaptive** (or specific) immunity, which is targeted against a previously recognized specific microorganism or pathogen. Thus, when a given pathogen is new to the host, it is initially recognized by the innate immune system and then the adaptive immune response is activated. The former acts very fast (minutes to hours) while the latter takes longer to evolve providing a more effective long-term response (within days to weeks) [3]. Our research proposal focuses mostly on the adaptive immune system, specific to humans. We explore interesting problems in this space and offers novel algorithmic techniques to solve them.

1.1 The Adaptive Immune System

Adaptive immunity is the product of a subset of white blood cells known as *lymphocytes*. Lymphocytes are divided into two kinds: B cells and T cells. These cells, along with their receptors—B cell receptors and T cell receptors, respectively—form the major component of the adaptive immune system. This immune system exists to provide immunity against numerous specific *antigens*, the protein molecules that elicit an adaptive immune response. All pathogenic microbes such as viruses and bacteria have an antigen subpart. There are also self-antigens—antigens produced by host cells—which help shape the immune repertoire [5]. Self-antigens exist to make sure immune responses do not target the host itself.

The adaptive or acquired immune system has three major functions [6]:

1. Recognize *non-self* antigens (negatively select for responses that affect host's self-antigens).

2. Generate efficient and responses specific to recognized antigens by creating and releasing antibodies that bind well to the antigen.

3. Maintain *immunological memory*. When the immune system encounters a known antigen, the response would be speedier and more effective.

Broadly speaking, the main function of B cells is to produce protein complexes called *immunoglobulins*, *B cell receptors (BCRs)* or *antibodies*[1] that are used to identify antigens [4]. Once trained against specific antigens, a lot of the trained B cells and the corresponding antibodies proliferate, giving rise to the BCR/antibody repertoire. Human antibody repertoires are

[1] Some authors use the term "antibodies" only for immunoglobulins that circulate freely in the bloodstream; we make no such distinction.

capable of recognizing more than 5×10^{13} different antigens [5]. T cells perform various important immune-related functions such as: helping B cells produce antibodies, control T cell population to prevent attacks against host tissue, and most importantly, destroy cells presenting antigens on their surface (such as virus-infected cells) [7].

Chapters 2 and 5 in this manuscript detail problems from two areas of the adaptive immune system, specifically, B cells and their receptors: (i) inferring the germline sequence composition of the immunoglobulin locus of the human genome from short-read sequencing data, and (ii) characterizing the evolutionary process underlying the BCR repertoire formation. The germline is the set of traits passed down to a child from their biological parents.

1.2 Pharmacogenes

Pharmacogenes are the set of genes that determine how fast or slow an individual responds to drugs [8]. For instance, the genotype of the gene *CYP2D6* is known to impact up to 25% of clinically prescribed drugs [9]. Chapter 3 details a robust method on genotyping a representative (but not exhaustive) set of pharmacogenes, using both short-read and long-read sequencing data.

1.3 DNA methylation

Chapter 4 explores a unique problem in deconvolution of a mixture of contributions from various cells into its constituent cell types. Methyl groups when added to certain nucleotides in the genome are known to enhance or suppress the activity of genes, in a process called DNA methylation. We explore the problem of inferring the contributions of methylation values of certain known cell types to the overall methylation features obtained from bulk data. Since the

reference methylation values are not fully known for each cell type, we also simultaneously infer the unknown values, in a block coordinate descent-like approach.

Chapter 2: Germline Genotyping of Adaptive Immune System Genes

The following section gives necessary information for the problem to be defined in subsequent sections.

2.1 Genes and Alleles: A Primer

The human genome consists of all genetic information for the organism and is present in its every living cell. It is made up of nucleotide sequences of DNA, whose combined length is around 3 billion, from the DNA alphabet, $\{A, C, G, T\}$, and is presented in 23 chromosome pairs in the cell nuclei.

When originally introduced, a "gene" was defined as a simple unit of heredity, an inheritable factor responsible for a physical trait [10]. The definition of the gene has evolved over the years to include additional details as the field of molecular biology matured. The updated definition of the gene proposed by Gerstein et al. [11] includes genomic sequences encoding all functional products (RNA and proteins), and a coherent union of overlapping genomic regions coding for overlapping functional products. A concise definition for the gene is given as "a union of genomic sequences encoding a coherent set of potentially overlapping functional products". Regulatory elements such as the promoter, enhancer, and insulator, that influence expression of genes, are not included in the definition. A gene whose final functional product is a protein is

first *transcribed* into RNA, which is later *translated* into a protein end-product.

An *allele* is one of multiple versions of DNA sequence (a single base or a potentially non-contiguous segment of bases) at a given genomic location. In this manuscript, an allele refers to a version of a gene, unless stated otherwise. Generally, an individual inherits two alleles, one from each parent, for any given genomic location. This difference is characterized by a set of *variant nucleotides* (or simply, *variants*). A variant can be *synonymous* or *non-synonymous*. Two individuals from the same species, each with a distinct allele of a protein-coding gene which differs from the other by a non-synonymous variant, will have this gene code for different proteins. There are about 20,000 protein-coding genes (out of a possible $\sim$60,000 genes) in the human genome [12].

A pseudogene is a DNA sequence that looks like a gene but does not produce a functional product. They are thought to be derived from functional genes through duplication but they have lost the original functions of their parental genes over many generations of evolution [13]. However, studies have shown that up to 20% of them are transcriptionally active, i.e., they produce RNA transcripts [14]. In a few cases, a pseudogene RNA was found to be spliced with the transcript of its neighboring gene to form a gene–pseudogene chimeric transcript. In the next chapter, we study an adjacent case of this phenomenon for a very important drug metabolism-influencing gene, *CYP2D6*, and its pseudogene neighbor, *CYP2D7*, where they are sometimes present as a gene-pseudogene (functional) fusion allele in the germline.

The definition of the gene also encompasses discontinuous genes such as the genes from germline immunoglobulin locus, where rearranged genes code for B cell receptor (BCR) surface proteins (also called "antibodies" when freely available in blood). Analogously, the germline T cell receptor locus contains genes that encode T cell receptor (TCR) surface proteins. Genes from

both of these genomic loci consist of the following gene segments: variable (V), diversity (D), joining (J), and constant (C), which are rearranged in B and T lymphocytes to form BCR and TCR genes, respectively [15]. The following chapter focuses on characterizing the coding region of each gene segment of the *variable* region of gene segments from various germline immunoglobulin and T cell receptor loci. In terms of labeling created by The International Immunogenetics Information System (IMGT), the coding region is annotated as "V-REGION" [16]. More details are presented in subsequent sections.

2.2 Germline Loci of the Adaptive Immune System

As mentioned in Chapter 1, the adaptive immune system consists of B and T cells and their receptors, BCRs and TCRs, respectively. Our first problem involves finding out the sequence composition for all genes of the immunoglobulin locus (which eventually form BCRs). Of all genes/gene families present in this locus, genotyping immunoglobulin heavy chain variable region (*IGHV*) genes is the hardest problem due to high degree of both intra-gene and inter-gene similarities, large number of total alleles present, and a significant number of psuedogenes. Our focus for the rest of the chapter is on genotyping *IGHV* genes.

The following text has been adapted from our article that was published in the Cell Systems journal [17].

2.3 Motivation for the Genotyping Problem

The vast majority of available genome sequencing data, especially clinical sequence data is, and, at least in the next 3-5 years, will be generated with short-read technology. Leveraging the

enormous quantity of short-read Whole Genome Sequencing (WGS) data available for genomic locus characterization provides an opportunity to gain biological insights into population-level diversity and disease associations. Together, such insights can help inform the development of low-cost clinical genotyping approaches that can be used to guide healthcare decisions. For example, the NIH's NIAID COVID Consortium[1] is a large collaborative effort that has generated and analyzed Illumina short-read WGS data from COVID-19 patients "with the aim of discovering monogenetic and multigenic variants that contribute to disease susceptibility, severity and treatment outcomes"[2]. Unfortunately the characteristics of short-read sequencing technologies make genotyping difficult for repetitive and homogeneous loci across the human genome.

One such challenging but biologically important region is the immunoglobulin heavy chain (*IGH*) locus on chromosome 14. The *IGH* locus harbors the variable (*IGHV*), diversity (*IGHD*), joining (*IGHJ*), and constant (*IGHC*) genes (or gene segments) that encode the building blocks of expressed B cell receptors (BCRs) and antibodies (Abs). The variable domain of BCRs and Abs is composed of *IGHV*, *IGHD*, and *IGHJ* gene segments, and is responsible for engaging directly with antigen, playing a critical role in identifying and neutralizing pathogenic viruses and bacteria. Despite the importance of BCRs and Abs in the adaptive immune system, our understanding of population-level genetic diversity in the *IGH* locus and its contribution to Ab function in disease remains limited [18, 19, 20]. The severe impact of the COVID-19 global pandemic stresses the need for development of tools and approaches to more fully characterize immune system genes, especially those involved in the development of neutralizing Abs. Unfortunately, due to its intricate sequence composition, *IGH* has remained stubbornly resistant to large scale, high

[1] `https://www.niaid.nih.gov/research/host-genetics-severe-covid-19-infection`

[2] A complementary international effort, the COVID-19 host genetics initiative (`https://www.covid19hg.org`) has similar goals.

resolution characterization.

IGH is one of the most complex and dynamic regions of the human genome [18] - it is known to contain many large structural variants (SVs), including segmental duplications, large insertions and deletions, and other copy number variants (CNVs) [21, 22, 23]. Of the *IGH* gene segments, *IGHV* is the most extensive gene group, and the ∼800–1000 Kb region of *IGH* in which this gene family resides has proven particularly challenging to genotype using short-read WGS data, due to its repetitive and homogeneous nature [24, 25]. A given individual is likely to have upwards of 40 functional and ORF *IGHV* gene copies on each haplotype, and at least twice as many non-functional pseudogene copies, collectively residing across the primary *IGH* locus on chromosome 14 and two *orphon loci* located on chromosomes 15 and 16 [23, 21]. These gene segments are short - between 165 bp and 305 bp (with a mean of 291 bp) - and are highly similar to others in terms of sequence composition, with 40% of the known functional alleles having a *sequence similarity* of $> 98\%$ with at least one other allele in a different gene[3].

While germline *IGH* genotypes have generally been overlooked in GWAS and other disease association studies, partially due to genotyping challenges, there is increasing interest in investigating the effect of *IGH* germline genetic variation on the adaptive immune system and disease [18, 20]. In fact *IGHV* germline genetic variation has recently been associated with clinical phenotypes including infectious disease and vaccine response [26, 27, 28, 29], autoimmune/inflammatory conditions [30, 31, 32], and cancer [33].

Of particular interest is COVID-19, which is primarily modulated by the innate immune system [34]. However, there is increasing evidence for an association between the adaptive immune system and COVID-19 severity [35, 36, 37], via the presence of anti-type I interferon (IFN)

[3] i.e., the edit distance between such a pair of alleles is no more than 2% of the length of the shorter allele

autoantibodies. This presents a potential biological mechanism for an association between *IGHV* germline genetic variation and COVID-19 disease severity, driven by *IGHV* germline alleles coding anti-type I IFN antibodies.

To date, there has been only one published method/computational pipeline for germline *IGHV* genotyping using short-read WGS data [38, 39]. Due to the genotyping challenges encountered with *IGHV*, the authors intended their pipeline to be applied on a population level, stating that it "is not intended to be used to accurately genotype individual genomes". The authors indeed applied their pipeline to a cohort of 109 individuals with the purpose of compiling aggregate measures of variation, and performed gene-level genotyping, with allele-level granularity on 11 (of the 56) *IGHV* genes.

One other previously published method, ImmunoTyper [40], performs *IGHV* genotype and copy number analysis using standard single molecule real-time (SMRT) Pacific Biosciences (PacBio) long read sequence data. As a result, its application has been limited by the paucity of publicly available long read WGS datasets. ImmunoTyper can not be easily modified to handle short-read WGS data because it relies on extracting full length *IGHV* segments from long PacBio reads (which are then clustered using a facility location formulation, similar to balanced k-means clustering). Short reads, on the other hand, only partially cover *IGHV* segments.

More recently [23] have published a technique called IGenotyper, which can haplotype the entire *IGH* region at high resolution using targeted ultra-deep *long read sequencing* and *de novo* assembly. This represents a critical step towards characterizing *IGH* haplotype diversity and heterogeneity, however since it requires a custom sequencing approach, it is expensive to scale and as a result is more suitable for high-resolution haplotype characterization rather than high-throughput *IGHV* genotyping.

There are several published methods for inferring germline *IGHV* genotypes using adaptive immune receptor repertoire sequencing (AIRR-seq) data as input [41, 42]. This approach benefits from not having to contend with pseudogene sequence, however it introduces noise that arises from somatic hypermutation. Furthermore, these methods can only provide calls for germline *IGHV* alleles that are expressed in the B cell population at the time of sampling, and thus are susceptible to missing *IGHV* alleles with lower expression.

We present ImmunoTyper-SR, a computational approach for genotype and CNV analysis of functional germline *IGHV* genes using short-read WGS data, using a database of known *IGHV* sequences as a reference. ImmunoTyper-SR is based on a combinatorial optimization formulation that aims to minimize the total edit distance between the reads and their assigned alleles while maintaining additional constraints on the number and distribution of reads across each allele identified (see Figure 2.1). This approach extends our previous work on long read based *IGHV* genotyping by addressing additional challenges introduced through the use of short reads. As a result, ImmunoTyper-SR is the one of the first short-read based germline *IGHV* genotyping tool that offers allele-level resolution.

We have validated ImmunoTyper-SR on 12 individuals with diverse genetic backgrounds from the 1000 Genomes Project [43] (1kGP), by comparing ImmunoTyper-SR genotype calls on Illumina WGS data from the NYGC [44] against targeted long read-based *IGH* assemblies generated using IGenotyper [23]. We then applied ImmunoTyper-SR to WGS data from a cohort of 585 individuals from the NIAID COVID Consortium (from here on the "NIAID cohort") to investigate associations between *IGHV* genotypes, type I IFN autoantibodies and COVID-19 disease severity. The cohort includes nine individuals who have been independently sequenced twice; this subcohort provides additional means to assess the robustness of ImmunoTyper-SR on

clinical sequencing data. We observed that ImmunoTyper-SR is able to produce *IGHV* accurate allele and CNV calls on both data sets, demonstrating its feasibility on Illumina WGS data with read lengths of 150 bp and moderate genome coverage. We finally employed a permutation test on the alleles identified by ImmunoTyper-SR to determine those strongly associated with anti-type I IFN autoantibody activity – within the limitations of the size and demographic composition of the NIAID cohort.

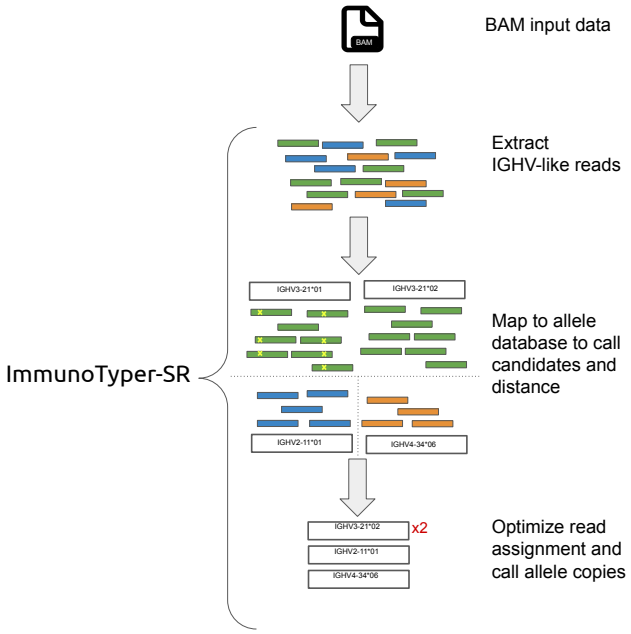


Figure 2.1: ImmunoTyper-SR workflow: ImmunoTyper-SR maps IGHV-relevant reads to the allele database and resolves mapping ambiguities using an ILP optimization approach, resulting in IGHV allele presence and CNV genotype calls.

2.4 Methods

2.4.0.1 Code availability

ImmunoTyper-SR is open source and it is available at `https://github.com/algo-cancer/ImmunoTyper-SR`.

2.4.1 Method Details

2.4.1.1 WGS Read Recruitment

A mapped WGS sample is provided to ImmunoTyper-SR in the BAM file format. Reads are extracted if they share any mapping to *IGHV*, the *IGHV* orphons on chromosome 15 and 16, or any other loci that share sequences similar to *IGHV*.

2.4.1.2 Allele Candidate Assignment

The extracted reads are then mapped to a database of *IGHV* allele reference sequences using BWA-MEM [45], with the `-a` option. The allele reference database was created by combining the current IMGT allele database [46] (including pseudogene and orphon alleles) with additional germline alleles (currently unpublished), resolved using IGenotyper [23]. While these novel alleles are as-yet unpublished, they are resolved using PacBio sequencing on an independent dataset that is not subject to the noise present in the 1kGP samples (see subsection 2.4.1.4 *Annotation of 1kGP Sample Assemblies*).

A read is putatively assigned to a single candidate allele or set of candidate alleles if it has a mapping of at least 50 bp to the allele reference sequence. A read can have a truncated mapping to an *IGHV* allele if and only if the truncation occurs either at the beginning or the end of the allele, such that truncated mappings that do not start on the first base of the allele, or end on the last base, are removed.

The edit distance between a given read and candidate allele is taken from the `NM` tag of the mapping, which is used below.

2.4.1.3 Allele Assignment

Reads are assigned to one of their candidate alleles through an Integer Linear Programming (ILP) approach, which aims to minimize the total sequence variation (i.e. edit distance) between assigned reads and alleles, while matching the sequence depth and variance of the given WGS sample. The ILP is solved using the Gurobi package [47].

As previously noted by [38, 39], a correct read assignment for a given allele will have a read depth in the shape of a trapezoid, as the number of reads having the minimum 50 bp of mapped sequence will decrease towards the ends of the allele reference sequence. This problem could be avoided by including the non-coding flanking sequence in section 2.4.1.3, however this technique may be complicated by the well known lack of characterized intergenic *IGH* sequences and haplotypes. As a result, we have opted to omit the flanking sequences in order to minimize any chances of genotype biases caused by reference haplotype sequences.

Sampling Sequence Depth and Variance: To ensure that our read assignment matches the underlying sample's sequencing characteristics, we empirically calculate the sequencing depth and variance from each sample by examining the WGS mapping at a representative locus. We use exon 327 of the TTN gene, located on chromosome 2. This region was selected for being one of the longest exons in the genome, and thus provides a $\sim$17 kbp sampling region that is likely to be relatively stable [48]. We further confirmed that there are no other loci in the genome with a similar sequence by simulating 200X 150 bp reads from the exon using ART [49] and mapping them back to GRCh38 using BWA-MEM [45] with the `-a` parameter; indeed, all reads mapped back to the exon region.

The sequencing depth and variance are calculated as the mean and variance of the mapped read depth across exon 327. To obtain the expected sequencing depth variance over the 'sloped' regions of the trapezoid shape found in a valid *IGHV* read assignment as defined in section 2.4.1.3, we calculate the read depth variance for each position of window of size $read\_length - 50$ across all bases of exon 327.

Coverage Constraints: In order to match the assigned read depth and variance to the TTN-sampled values, we must monitor the read depth for each possible assignment that arises during the optimization process defined below. We only monitor the read depth at equally-spaced 'landmark' positions, which are chosen for every candidate allele in advance, to reduce optimization time. We can then constrain the mean assigned read depth across landmark positions to be within one standard deviation of the expected read depth using the sequencing depth and variance values calculated in section 2.4.1.3. However, this constraint may allow for a read assignment with outlier read depth values at consecutive landmark positions that retain a mean coverage within the bounds. To combat this problem, we group landmarks (in a round-robin process) and set the constraint defined above for each landmark group independently. This reduces the likelihood that two such balancing outlier landmarks are in the same group.

Discarding reads: The read assignment optimization strategy allows for reads to be discarded, to contend with reads that come from non-*IGH* loci which do not have their source sequence characterized in the IMGT allele database, but nonetheless have a mapping due to their similar sequence content. A given read can be discarded with a penalty equal to the expected number of errors for the read, multiplied by a constant factor, which is set to 2, by default. This implies that

to be discarded, a read needs twice as many errors as expected relative to to the closest matching allele in the database which allows for a small number of novel variants to exist without the read being discarded.

ILP Model Definitions: Let $\mathbf{R} = \{r_i\}_{i=1}^{n}$ be the set of all WGS reads in the sample.

Let $\mathbf{A} = \{a_j\}_{j=1}^{m}$ be the set of database alleles. Let $\mathbf{C}_i$ be the set of candidate alleles of read r_i.

Let $\mathbf{L}_{j,g}$ be a set of landmark nucleotide positions in a_j that are part of landmark group g, where g is a subset of all positions in a_j. $\mathbf{L}_j$ is the set of all landmark positions in allele a_j.

Let $covers(r_i, a_j, l)$ be the set of reads that have a_j as candidate/mapping and cover landmark l.

Let $\mu_{l,j}$ and $\sigma_{l,j}$ be the expected mean and standard deviation (*SD*) in read coverage of allele a_j for a given landmark $l \in \mathbf{L}_j$ for a single copy of the allele. This is computed by sampling read coverage within TTN gene's exon 327 and dividing by two to account for the diploid nature of the exon to estimate empirical mean and SD for the read coverage of landmark l.

Let $\lambda =$ user-provided upper bound on the number of standard deviations allowed on the deviation of mean read coverage of any reference allele. Default 1.5.

Let $min\_cov =$ user-defined proportion of the estimated mean read coverage of any landmark position that the data needs to satisfy. Default 0.3.

Let $e(r_i, a_j) =$ edit distance for the best mapping/alignment of r_i on a_j.

Let $expected\_errors(r_i) =$ sequencing error rate $\times$ alignment length of primary mapping for r_i.

Let $discard\_penalty\_multiplier =$ user-provided penalty for discarding a read. Default 2.

ILP Model Variables:

$$\text{Let } D_i^j = \begin{cases} 1 & \text{if } r_i \text{ has been assigned to } a_j \\ 0 & \text{otherwise} \end{cases}$$

$$\text{Let } X_i = \begin{cases} 1 & \text{if } r_i \text{ is discarded} \\ 0 & \text{otherwise} \end{cases}$$

$$\text{Let } C_{c,j} = \begin{cases} 1 & \text{if } a_j \text{ has } c \text{ copies called} \\ 0 & \text{otherwise,} \end{cases}$$

where $c \in \{0, \ldots, K\}$ where K is the maximum allowed number of copies per allele.

ILP Model Constraints:

$$\forall a_j, \quad \sum_c C_{c,j} = 1 \tag{2.1}$$

$$\forall r_i, \quad \sum_{a_j} D_i^j = 1 - X_i \tag{2.2}$$

$$\forall a_j, \quad n \cdot \sum_c C_{c,j} \geq \sum_{r_i} D_i^j \text{ where } n \text{ is the total number of reads} \tag{2.3}$$

$$\forall a_j, \quad \sum_{r_i} D_i^j \geq \sum_c C_{c,j} \tag{2.4}$$

$$\forall a_j, \forall l \in \mathbf{L}_{j,g}, \sum_{r_i \in covers(r_i, a_j, l)} D_i^j \geq min\_cov \cdot \mu_{l,j} \cdot \sum_c c \cdot C_{c,j} \tag{2.5}$$

$$\forall \mathbf{L}_{j,g}, \sum_{l \in \mathbf{L}_{j,g}} \left| \sum_c \mu_{l,j} \cdot c \cdot C_{c,j} - \sum_{r_i \in covers(r_i, a_j, l)} D_i^j \right| \leq \sum_{l \in \mathbf{L}_{j,g}} \sum_c \lambda \cdot \sigma_{l,j} \cdot \sqrt{c} \cdot C_{c,j} \tag{2.6}$$

1. One-hot encoding of all possible copy numbers for each allele.

2. Prevent discarded reads from being assigned to any allele and enforce assignment to at most one copy.

3. If at least one of the reads is assigned to allele a_j, then *at least* one copy must be called.

4. If $C_j == c(> 0)$ (allele a_j is called and has c copies), there must be *at least* c reads assigned to a_j, to ensure there is no allele copy with zero reads assigned.

5. Each landmark position in $\mathbf{L}_{j,g}$ should have a minimum read coverage proportional to the expected coverage, if allele a_j is called.

6. If c copies of allele a_j are called, the deviation of read coverage of a group of landmark positions away from the estimated mean is bounded.

ILP Model Objective Function: Minimize:

$$\sum_{r_i \in \mathbf{R}, a_j \in \mathbf{A}} D_i^j \cdot e(r_i, a_j) + \sum_{r_i \in \mathbf{R}} \Big(X_i \cdot expected\_errors(r_i) \cdot discard\_penalty\_multiplier \Big)$$

2.4.1.4 Annotation of 1kGP Sample Assemblies

We created *IGHV* gene and pseudogene annotations for each haplotype from the 12 1kGP *IGH* assemblies to act as ground truth in our evaluation of ImmunoTyper-SR allele calls. The complete allele database was mapped against each assembly using Bowtie2 [50] (with `-af --end-to-end --very-fast` parameters). The mapping results are then grouped by target location, and the best mapped allele for each target location is assigned to that gene. Any assignment with an edit distance > 1 is manually reviewed.

It is possible for an assembly to have *IGHV* gene copies that have considerably divergent sequence from any allele in the database. This can happen either through a true novel allele, as a result of a sequencing or assembly error, or due to large structural alterations present in the sample haplotype. This latter case is of particular concern as the 1kGP samples employed in this study are derived from lymphoblastoid B cell lines (LCL), which have undergone some degree of V(D)J rearrangement. This can introduce noise in the sequencing dataset, and can result in somatic deletions in a haplotype that may affect one or more *IGHV* genes.

To identify any considerably divergent *IGHV* genes, we generated 150 bp error-free in-silico reads from all sequences in the allele database and mapped them to the haplotypes using BWA-MEM [45] (`-a` parameter). Any contiguous target region not identified above was extracted and mapped back to the allele database using BWA-MEM [45] (`-a` parameter) to identify the most similar allele and edit distance.

2.4.1.5 Comparison With the Luo et al. Pipeline

We implemented the pipeline as described in "Worldwide genetic variation of the *IGHV* and *TRBV* immune receptor gene families in humans" [39]. *IGHV* gene calls and CNVs were compared against *IGH* assembly annotations as described in section 2.4.1.4.

Luo et al. [39] limit allele calling to the following 11 genes that they determine are two-copy genes: *IGHV1-18, IGHV1-24, IGHV1-45, IGHV1-58, IGHV2-26, IGHV3-20, IGHV3-72, IGHV3-73, IGHV3-74, IGHV5-51,* and *IGHV6-1*. Since not all these genes are present in two copies in the 1kGP samples, we further limited allele calls only to those genes that are present in the above list, and for which the pipeline gave CNV calls of two.

As a result, when comparing the allele calls against a sample's ground truth and ImmunoTyper-SR, we only used those genes that met the pipeline's allele-calling criteria in *IGHV* assembly.

2.4.1.6 Measuring Concordance between Doubly-sequenced Samples

To calculate genotype call concordance between two WGS sequences from the same subject of the NIAID subcohort that has been doubly-sequenced, we used the weighted Jaccard similarity coefficient (J_w), defined as follows:

Let s be a subject who was sequenced twice and $g_{s,i}$ be the genotype vector corresponding to i-th WGS sample of the subject ($i \in \{1, 2\}$), where k-th element of the vector ($g_{s,i}[k]$) is the number of copies of allele A_k, as called by ImmunoTyper-SR. Then, the weighted Jaccard similarity coefficient for the subject s is:

$$J_w(s) = \frac{\sum_{k=1}^{m} min(g_{s,1}[k], g_{s,2}[k])}{\sum_{k=1}^{m} max(g_{s,1}[k], g_{s,2}[k])}$$

2.4.2 Quantification and Statistical Analysis

2.4.2.1 Statistical Association between *IGHV* Genotype and anti-Type I IFN Autoantibody Presence

First, the 585 samples from the NIAID cohort were filtered for the presence of α, β, ω IFN autoantibody labels. Any label with a "Yes, partially" value was modified to "Yes", and we combined the individual α, β, ω labels into a single binary label indicating presence of any one of the autoantibodies. *IGHV* alleles that were present in at least one of sample's genotype were

chosen as the independent variable, in the form of binary presence/absence labels. A candidate selection logistic regression model was then performed between all such *IGHV* alleles and anti-type I IFN autoantibody presence, and those alleles with a significant p-value ($>= 0.01$) were then selected as top candidates.

The candidate alleles were applied in a separate logistic regression model to determine the effect size. For significance and multiple test correction, we performed a permutation test, by applying a logistic regression model to the dataset 100,000 times, where the autoantibody labels were randomly shuffled in each iteration. The resulting p-value was calculated for each allele as the proportion of the shuffled models that had a more extreme p-value than that in the original, unshuffled model.

All logistic regression was performed using the `glm` function in R with the `family="binomial"` parameter.

2.5 Results

A primary challenge in germline *IGHV* genotyping is the validation of computationally identified alleles. To date there has been only one *IGHV* genotyping protocol published that uses short-read data [38, 39]. A recently published *IGH* haplotyping tool [23] has opened the door for high quality *IGH* assemblies which can be used for validation.

IGHV genotyping through inference using AIRR-seq (Adaptive Immune Receptor Repertoire sequencing) is common-place and well established [41, 42], however, we were unable to find any publicly available paired AIRR-seq and WGS datasets that would be suitable for comparison.

2.5.1 Validation using 1kGP Samples

We ran ImmunoTyper-SR on 12 publicly available high-coverage WGS of 1kGP samples, sequenced at ∼30X on the Illumina NovaSeq 6000 Platform [44]. These samples have had independent *de novo IGH* haplotyping performed by Rodriguez et al. using a targeted long read sequencing and assembly protocol called IGenotyper [23] tailor-made for the *IGH* region. While this technique represents the most accurate *IGH* haplotyping tool published to date, it is likely that these assemblies are not 100% accurate, as the samples are derived from LCL cell lines. It has been noted that the usage of LCL cell lines can impact *IGH* genotype accuracy due to the presence of somatic V(D)J rearrangement [51]. This may result in somatic deletions or inconsistent read coverage, which can affect the assembly quality, and ultimately ImmunoTyper-SR's genotype accuracy.

Distribution of genotype call results are shown in Figure 2.2; as can be seen, the mean precision and recall values are respectively 83.7% and 80% for identification of each allele sequence and its copy number exactly. While ImmunoTyper-SR generally provided accurate genotype calls, the ranges of precision and recall values were large, with a minimum and maximum F-score of 63% and 88% respectively. These figures improve substantially if some limited noise can be tolerated in the sequence composition or copy number of the allele calls, as discussed in the remainder of the section.

2.5.1.1 Accuracy is Impacted by Novel Alleles

As ImmunoTyper-SR is designed to find the closest allele in the database and not call novel alleles and variants, we investigated the relationship between the presence of putative novel

Figure 2.2: ImmunoTyper-SR's functional allele call accuracy on 1kGP dataset ($n = 12$). No CNV indicates measures accuracy for presence/absence of alleles. 3 bp allowance counts a false positive as a true positive if there is a false negative allele within 3 bp edit distance.

variants as a driver of the variation seen in genotype accuracy. It is important to distinguish these putative novel alleles, which are derived from the 1kGP sample *IGH* assemblies, from those validated novel alleles that were added to the allele database as described in Section 2.4.1.2, which were sourced from an independent, high-quality dataset.

There is significant correlation between the proportional divergence of a given sample's *IGHV* alleles from the allele database and ImmunoTyper-SR's genotype call accuracy.

As previously mentioned, it is likely that some of these novel variants are in fact due to sequencing or assembly errors, or noise from *IGH* rearrangement due to the LCL-sourced samples. Since it is impossible to determine which of the putative novel alleles are valid, we chose not to add these novel sequences to the allele database for fear of increasing noise and model complexity.

2.5.1.2 High Genotype Accuracy for Distinct Genes

To investigate the genotype accuracy on a gene-by-gene basis, we grouped allele call results by genes across all samples. *IGHV* genes that have high sequence similarity to many alleles in the database (regardless of gene "source" of each such allele) have lower allele call accuracy.

To reduce the impact of this effect, we recalculated allele call accuracy statistics by not penalizing a miscalled (false positive) allele as long as it is within 3 bp edit distance of a ground truth (false negative) allele (see Figure 2.2). This increases the mean F-score by 6.6%, from 81.5% to 87.9%. The tolerance threshold is chosen to be 3 bp because it is about 1% of the average length of all *IGHV* allele sequences.

2.5.1.3 ImmunoTyper-SR Reports Genotypes with Higher Granularity and Accuracy than Comparable Methods

We compared ImmunoTyper-SR's genotype results on the 1kGP samples against the only other published short-read *IGHV* genotyping method, [39]. With the exception of at most 11 two-copy genes (out of 56 *IGHV* genes with at least one functional allele), the pipeline reports only gene-level genotypes and CNV calls (see Section 2.4.1.5). We implemented the pipeline in-house and compared gene call results. We found that ImmunoTyper-SR had much higher precision (median 84.7% vs 93.1%), and moderately improved recall (mean 87.4% vs 90.1%).

For allele calls, Luo et al. pipeline calls alleles for only a small proportion of the *IGHV* genes present, an average of 5.75 out of 44 functional genes. For those genes that do have allele calls, ImmunoTyper-SR is much more accurate, with mean precision and recall values more than

double the Luo et al. results (see Table 2.1). Note that the 11 genes listed in Section 2.4.1.5 are those genes that the authors deemed two copy genes with high confidence, however in many cases not all of those genes were called with two copies and thus not selected for allele calling, despite having two copies in the ground truth.

Sample	Luo et al. Precision	ImmunoTyper-SR Precision	Luo et al. Recall	ImmunoTyper-SR Recall	Proportion of Functional Genes with Allele Calls by Luo et al.
HG00512	50.0	94.7	50.0	100.0	9 / 44
HG00513	20.0	100.0	20.0	90.0	5 / 45
HG00514	37.5	87.5	37.5	87.5	4 / 34
HG00731	56.2	100.0	56.2	93.8	8 / 41
HG00732	16.7	100.0	16.7	100.0	3 / 49
HG00733	37.5	88.9	37.5	100.0	4 / 48
HG01109	41.7	100.0	45.5	100.0	6 / 44
HG01243	50.0	100.0	50.0	100.0	4 / 44
HG02723	35.0	77.8	38.9	77.8	10 / 44
HG03098	50.0	76.9	50.0	71.4	7 / 46
NA12878	31.2	100.0	31.2	100.0	8 / 40
NA19240	62.5	100.0	62.5	100.0	4 / 49
Mean	40.7	**93.8**	41.3	**93.4**	5.75 / 44
Median	39.6	**100.0**	42.2	**100.0**	5.5 / 44

Table 2.1: Allele call comparison between Luo et al. (2019) pipeline and ImmunoTyper-SR. The final column shows the number of ground truth genes in the sample that met the allele-calling criteria of the Luo et al. pipeline (i.e. that the gene is among the 11 least ambiguous genes and its number of copies is predicted to be ≤ 2); their precision and recall values are calculated only on those genes. In contrast, ImmunoTyper-SR calls alleles for all ground truth genes and its precision and recall values are thus calculated on the entire set of genes.

2.5.2 Validation and Association with anti-Type I IFN Autoantibodies with a Cohort of 585 COVID-19 Patient WGS Data

As part of the NIAID COVID Consortium we analyzed health records and genomic data from a cohort of 585 COVID patients. The dataset includes various health information for each patient, as well as whole genome sequences performed at The American Genome Center,

USUHS, and results from anti-type I IFN autoantibody (aAb) assays.

2.5.2.1 Genotype Validation Through Doubly-Sequenced Concordance

To further validate ImmunoTyper-SR on non-LCL sourced WGS samples, we compared genotype calls for nine individuals who had been independently sequenced twice as part of the NIAID cohort. The WGS were generated at the same sequencing center, which reduces the effect of systematic sequencing bias. We compared the complete set of allele calls and CNVs between the matched WGS. The mean weighted Jaccard similarity coefficient across all nine doubly-sequenced samples was 0.696. If the CNV calls are ignored, the mean score increases to 0.923, indicating many of the miscalls are due to CNVs, rather than calling the wrong allele (see Figure 2.3).

Figure 4: Distribution of Jaccard similarity coefficients for doubly sequenced samples in the NIAID cohort: (left) no error tolerance in sequence composition or copy number of called alleles; (center) no error tolerance in copy number but an edit distance of ≤ 3 tolerated on called alleles; (right) no error tolerance in sequence composition of the alleles with copy number differences ignored.

2.5.2.2 Association with anti-Type I IFN Autoantibodies

Removing samples that did not have WGS or aAb assay results produced 542 samples, of which 32 tested positive for the presence of anti-type I IFN aAb. The association, effect size, and p-values are provided in Table 2.2 for the top most significant alleles. It is important to note that three out of four of the top most significant alleles are rare alleles, present in at most 10

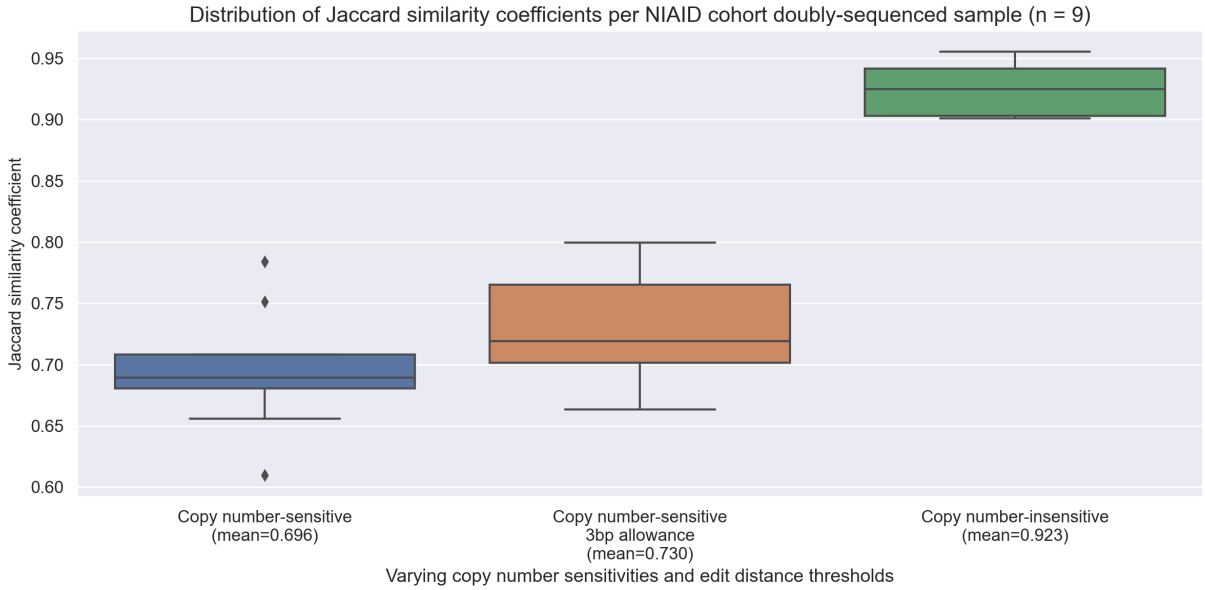


Figure 2.3: Distribution of Jaccard similarity coefficients for doubly sequenced samples in the NIAID cohort: (left) no error tolerance in sequence composition or copy number of called alleles; (center) no error tolerance in copy number but an edit distance of ≤ 3 tolerated on called alleles; (right) no error tolerance in sequence composition of the alleles with copy number differences ignored.

Allele	*IGHV*1-46*Novel-168	*IGHV*5-51*Novel-501	*IGHV*3-49*04	*IGHV*3-9*Novel-205
Samples with genotype	3 / 542	10 / 542	237 / 542	9 / 542
Samples with IFN aAb	1	2	10	2
Samples without IFN aAb	2	8	227	7
Effect probability	37.5	24.6	3.7	26.6
P-value	0.037	0.018	0.11	0.013

Table 2.2: Summary statistics of *IGHV* alleles that are associated with Type I IFN aAb in the 542 samples from the NIAID cohort that have been tested. Top row: the proportion of the samples with each allele. Second row: the number of samples with the allele as well as the Type I IFN aAb. Third row: Number of samples with the allele but not the IFN aAb. Bottom two rows: Association between the allele and the Type I IFN aAb.

individuals. In addition, this analysis suffers from a very low number of anti-type I IFN aAb cases present in this dataset. While two of the selected alleles have relatively low p-values, they remain higher than 0.01, partially due to their rarity in the cohort.

2.6 Discussion and future work

ImmunoTyper-SR successfully raises the bar for short-read-based *IGHV* germline genotyping. While the tool has demonstrated the greatest accuracy of published Illumina-based tools to date, it is important to note that there is still room for improvement, particularly when distinguishing highly-similar alleles.

The challenges presented by novel alleles are another opportunity for improving *IGHV* genotype calls. ImmunoTyper-SR clearly performs best when a sample's *IGHV* alleles minimally diverge from the known allele database. As the *IGHV* allele database is updated and improved, genotype call accuracy will likely increase. Ongoing projects such as the OGRDB [52], a curated database of germline *IGHV* alleles inferred from AIRR-seq data, as well as those being curated from high-quality long read assemblies, may help improve WGS short-read-based germline *IGHV* calling. In addition, our future work includes expanding ImmunoTyper-SR to call novel alleles and variants, and further validating ImmunoTyper-SR calls using trio data.

Our article represents one of the first investigations into germline *IGHV* genotype-phenotype association using short-read WGS data. Despite there being a hypothetical biological mechanism for the association between *IGHV* genotype and anti-type I IFN autoantibodies, the dataset used is very limited for this application. The low number of autoantibody-positive patients and rarity of the associated alleles severely reduces statistical power resulting in a low-confidence statistical analysis. As a result, we must judge the association results as inconclusive, despite finding two associated alleles with relatively low p-values. However, our future work includes repeating this analysis on a cohort of $>$ 1200 COVID-19 patient WGS data, which we believe will improve our confidence.

While our investigation of association was inconclusive, there are many other potential association targets for future work. It is important to acknowledge a possibility that disease effects of germline *IGHV* variants can be overcome by somatic hypermutation during antibody repertoire proliferation. However, the (currently limited) body of evidence demonstrating *IGHV* variant effects on phenotype suggests there are many *IGHV* alleles that have a considerable effect on a variety of diseases. In the near future, comprehensively investigating these associations is likely only possible using short-read WGS data; there are numerous WGS datasets in existence which have remained untouched with respect to *IGHV*, and alternative data types have additional costs and scaling challenges.

It is possible that using alleles as the explanatory variables in phenotype association is less biologically relevant than single nucleotide variants. If a single nucleotide variant is the root driver of a phenotype effect, perhaps by affecting development of a given protective or detrimental antibody, then using variants as an explanatory variable would be more suitable. However, if the effect is driven by the combination of variants, using alleles may be more suitable. Given a sufficient sized dataset, joint effects, either of variants or alleles, could also be investigated. It is also possible that other variant types, such as structural variants or CNVs, could have an effect on phenotype. Our future work includes improving ImmunoTyper-SR to provide variant calls, which will improve the ability to investigate phenotype and disease associations.

There is great opportunity for future studies that combine WGS with complete BCR repertoire data. The immediate application would be to validate WGS and inference-based germline *IGHV* genotype calls using orthogonal datasets. Unfortunately we were unable to find any such paired datasets for this study.

There are also open questions regarding the direct effects of germline *IGHV* genotype

on the BCR repertoire. This effect has been investigated with repertoire data alone, however inference-based germline genotype calls are impacted by temporal effects of repertoire sampling, and as a result may be incomplete. By combining germline sequence data with repertoire data, it will be possible to create a complete picture of the interaction between germline *IGHV* genotype and the BCR repertoire.

By enabling short-read WGS-based *IGHV* genotyping with allele-level granularity, ImmunoTyper-SR opens the door to applying the power of the largest WGS datasets available to uncover the mysteries of one of the least understood loci in the human genome.

Chapter 3: An efficient genotyper and star-allele caller for pharmacogenomics

Following the theme of genotyping complex germline loci, we present our work on genotyping another important set of genes known as *pharmacogenes*. These genes encode proteins/enzymes that are responsible for determining an individual's rate of metabolism of various pharmaceutical drugs [53]. Inter-individual heterogeneity in drug metabolism may be attributed to both (a) genotype of these pharmacogenes and (b) their copy number polymorphisms. In order to fully characterize an individual's response to specific drugs, the following information needs to be known:

1. The genotypes and copy numbers of an appropriate set of pharmacogenes of the individual, and

2. The relation between the genotypes and phenotypes (here, rate and other characteristics of drug metabolism).

The genotype-phenotype correlation information is obtained through large cohort studies with candidate gene and whole genome analyses [53]. This chapter focuses on characterizing the genotypes of an individual's pharmacogene set of interest, from both short-read and long-read data. The rest of the chapter is adapted from our published work [54].

3.1 Motivation for pharmacogene genotyping problem and previous work

The rapid development of high-throughput sequencing (HTS) technologies has ushered in the era of precision medicine that aims to tailor medical decisions at the individual level [55]. A key component of precision medicine is pharmacogenomics which studies the associations between the individual genotypes of clinically important *pharmacogenes*) and individual variation in drug response [8]. While the HTS data theoretically provides sufficient means to accurately genotype any gene in a given individual, genotyping of many pharmacogenes remains challenging [56]. One of the key challenges is the fact that many pharmacogenes of vital clinical importance—most notably the *CYP2D6* gene, whose genotype impacts up to 25% of clinically prescribed drugs [9]—are highly polymorphic, and furthermore located next to the highly similar pseudogenes due to being located within segmental duplications [9]. Many of these pharmacogenes are also subject to various copy number and structural changes, e.g. through a fusion event between a gene and its pseudogene, possibly due to the instability of the segmental duplication region wherein they reside [57]. These issues need to be carefully and comprehensively accounted for before the genotyping process, in order to obtain accurate results. Lastly, alleles of many pharmacogenes are not defined through a single nucleotide variant (SNV), but through a complete gene haplotype. Thus, the exact functional impact of an allele can only be determined through phasing, or haplotyping, of the whole genic region. In the pharmacogenomics community, haplotyping is commonly known as *star-allele calling* [58], owing to the fact that most of the known pharmacogenetic haplotypes are assigned a unique star-allele identifier.

Standard tools for genotyping HTS datasets, such as Genome Analysis Toolkit (GATK) [59, 60], cannot be used for star-allele calling because they are unable to haplotype the whole genic

regions and assign correct star-alleles. General-purpose computational phasing tools, such as HapCUT2 [61] and HapTree-X [62], are also inadequate for calling star-alleles: Either these tools are designed for phasing diploid organisms and thus cannot phase regions that underwent significant copy number changes or cannot handle fusions and other structural variations. Furthermore, the distance between allele-defining variants is often too large and, as a result, many alleles cannot be phased with short-read sequencing data. On the other hand, statistical phasing tools, such as Beagle [63] or Eagle [64], also do not handle the presence of fusions and structural variations. Thus, most of the star-allele calling was— and still is— being done by various custom primer-specific PCR and array assays [65], mostly due to their price and speed, despite calls generated by these assays being often limited in breadth and scope [66, 67].

Several tools have been recently developed to address the challenge of accurate star-allele calling [68]. Cypiripi [65], the first tool specifically designed for this purpose, supported calling *CYP2D6* star-alleles from Illumina WGS data. Cypiripi was followed by Aldy [69], Stargazer [70], Astrolabe [71], StellarPGx [72], PharmCAT [73] and Cyrius [74]. These tools aggregate the data from the existing star-allele databases, such as PharmVar [75], and use it to call star-alleles directly from HTS data. Some of these tools, such as Aldy and Stargazer, are also able to detect copy number changes and fusions with a high level of accuracy. However, a majority of these tools target only a small set of pharmacogenes (typically *CYP2D6* and other cytochrome P450 genes) and are tuned for short-read HTS data generated by the Illumina whole-genome sequencing (WGS), whole-exome sequencing (WES) and (in some cases) targeted sequencing panels such as PGRNseq [76].

In recent years, there is a slow but steady shift toward third-generation HTS technologies such as Pacific Biosciences (PacBio) and Oxford Nanopore [77]. These technologies produce

significantly longer reads (typically measured in tens of kilobases) than Illumina reads (measured in tens of basepairs). While they were initially dismissed in clinical settings due to the high cost of sequencing and high error rates, these technologies are making a resurgence thanks to the recent improvements in terms of accuracy and cost. For example, PacBio HiFi sequencing offers up to 25 kbp-long reads with a 99.5% accuracy rate [78]. Unfortunately, not many tools are able to correctly use the data generated by these technologies for calling pharmacogenomic star-alleles due to the different assumptions and biases as compared to the standard Illumina short-read data. Star-allele callers are also unable to make use of the long-range information within long reads for better phasing of allele-defining variants.

Here we present Aldy 4, the next iteration of Aldy software that addresses the aforementioned challenges. Aldy 4 completely revamps its original star-allele calling pipeline and adds support for long-read technologies such as PacBio HiFi, while extending support for short-read technologies (whole-genome and targeted capture data). Other updates also include extensive support for genotyping whole-exome [79] and VCF data. The changes include an alignment correction module that addresses various biases and errors common during the alignment of long reads to the pharmacogenomic regions. It also provides a novel star-allele calling pipeline that incorporates the long-range phasing information from long reads into the star-allele calling model. Finally, Aldy 4 brings support for 19 new pharmacogenes, provides an easy interface for adding the support for other pharmacogenes, adds an application programming interface (API) for easy incorporation of pharmacogenomic calling within the existing pipelines, and brings various other improvements to the original pipeline. As such, we envision Aldy to be a single-stop tool for calling star-alleles from diverse sequencing technology datasets, and hope that it will remain a crucial tool in the pharmacogenomics toolbox even with the advent of long-read sequencing

technologies.

3.2 Methods

The goal of Aldy is to reconstruct the exact sequence content (or haplotype) of each gene copy of a given pharmacogene from a high-throughput sequencing (HTS) data sample and assign a star-allele to each reconstructed haplotype present in the dataset. This process is subsequently referred to as *star-allele calling*.

In order to accurately call star-alleles, it is necessary to consult a database of known star-alleles that contains the exact sequence content of each pharmacogene allele. Suppose that a pharmacogene G harbors variants $\mathcal{M} = \{m_1, ..., m_{|\mathcal{M}|}\}$, where any $m \in \mathcal{M}$ is a single nucleotide substitution or a small indel. Depending on their impact on the gene G, these variants are either deemed functional (core) variants or silent variants (subvariants); while core variants are typically non-synonymous, they might also include UTR and intronic variants that affect the drug metabolism. The reference allele of G is an allele that harbors no variants at all. It is commonly known as the *1 star-allele. Notable exceptions to this rule include *CYP2C19* (where *CYP2C19*1* was recently renamed to *CYP2C19*38*) and wild-type alleles in *IFNL3* and *DPYD*. Any other star-allele S_i is defined by the subset of known variants $\mathcal{M}$ that distinguish its sequence content from the reference *1 allele.

In some genes, such as *CYP2D6*, star-allele identifiers are also assigned to fusions and other pseudogene-induced structural changes that affect the pharmacogene. For this reason, the definition of star-alleles is extended to also include their structural configuration. This configuration describes whether a pharmacogene is wholly present in the genome, deleted, or is a gene-

pseudogene hybrid. The set of valid configurations is denoted as $\mathcal{G}$. Note that each structural configuration can induce many distinct star-alleles depending on the choice of variants from $\mathcal{M}$. Thus, we can formally define a *star-allele* S_i as a tuple $(\mathbf{g}_i, A_i)$ where $\mathbf{g}_i \in \mathcal{G}$ and $A_i \subseteq \mathcal{M}$. The *star-allele database* is formally a collection of all known structural configurations, variants and known star-alleles $(\mathcal{G}, \mathcal{M}, \{S_1, S_2, \dots\})$ where $S_i = (\mathbf{g}_i, A_i)$ such that $\mathbf{g}_i \in \mathcal{G}$ and $A_i \subseteq \mathcal{M}$.

To call star-alleles of a given pharmacogene from the given sequencing sample, Aldy needs to perform the following steps:

- analyze the aligned HTS reads in BAM/CRAM format and resolve incorrectly aligned reads;
- detect structural configurations by calling copy number changes and gene-pseudogene fusions; and
- use the read alignments from BAM/CRAM to call star-alleles and phase the gene.

3.2.1 BAM/CRAM analysis and alignment correction

Aldy begins by taking a SAM, BAM or CRAM file [80] generated by a read aligner (e.g. `BWA` [81], `pbbm2`[1] or `minimap2` [82]). It is recommended to post-process these files with the GATK's "Best Practices" pipeline [83] (the local indel realignment step is especially helpful for the subsequent variant calling). Aldy extracts the relevant variants that are present in a given pharmacogene from the alignment file, as well as coverage information needed for the copy number and structural variation detection step. It also collects phasing information from long reads, barcoded fragments and paired-end fragments where available.

[1] `https://github.com/PacificBiosciences/pbmm2`

The original version of Aldy relied on the assumption that read alignments produced by the off-the-shelf aligner are mostly correct. While this assumption holds for short paired-end Illumina reads, it breaks for long reads such as PacBio HiFi reads. For example, if a sample harbors a gene duplication and if the highly similar pseudogene is located immediately next to this gene, any long read spanning two duplicated copies of the gene will get its second half incorrectly aligned to the pseudogene because the reference genome does not contain two copies of the gene in question (Integrative Genome Viewer (IGV)[84] coverage plot in Figure 3.1). The correct alignment would perform a split mapping and align the second half again to the pharmacogene. These incorrect alignments are even more problematic in the presence of gene fusions: any read that spans a gene-pseudogene fusion breakpoint will not be split-mapped but incorrectly aligned to either pharmacogene or its pseudogene.

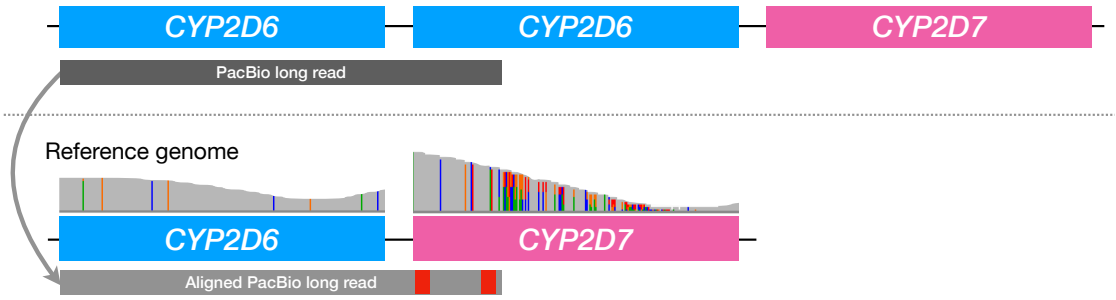


Figure 3.1: An example of an incorrect long read alignment to the reference genome and its correction. If a donor genome (above) contains two copies of *CYP2D6* pharmacogene, any long read (gray rectangle) that spans both copies will get aligned to the reference genome (below) that contains only a single *CYP2D6* copy. However, this read will get its second half (containing *CYP2D6* sequence) incorrectly aligned to the *CYP2D7* pseudogene due to the high sequence similarity between these genes. The final result is the over-abundance of coverage in the pseudogene region as compared to the *CYP2D6* region (an IGV coverage plot is shown above the reference genome).

Aldy 4 corrects such alignments by splitting any long read that spans the gene-pseudogene boundary into shorter gene-level segments and aligning each segment independently. Each segment is guaranteed to span only one gene (either pharmacogene or a pseudogene) and thus avoid

being misaligned in the manner described above. The size of each such segment is at most the size of the gene. Aldy 4 performs a further split-mapping of each segment that spans a potential fusion breakpoint to determine whether a read originates from a fusion event or not (a read is said to originate from a fusion event if its split-mapping alignment score is better than the original alignment score).

Unlike previous versions, Aldy 4 considers base quality scores and read mapping qualities when calling the allele-specific variants. This ensures that the low-quality variants in noisy and low-coverage samples are filtered out before the star-allele calling. Finally, Aldy 4 performs the indel-realignment step through indelpost [85] to correct misalignments that often happen when aligning short-reads to small indels [86].

3.2.2 Copy number and structural variation analysis

In a typical scenario, a sample contains two parental copies of a pharmacogene of interest for which star-alleles need to be called. This is generally true for most of the pharmacogenes of interest. However, a few major pharmacogenes do not follow this pattern and are prone to various copy number changes and structural events. The most notable example is that of the *CYP2D6* gene, one of the most important pharmacogenes [9], whose copies can undergo whole gene deletions, duplications and hybrid fusions (where a copy begins with the *CYP2D6* sequence but switches to the pseudogene *CYP2D7* sequence at a given breakpoint, or vice versa) [87]. Other prominent examples include *CYP2A6*, *G6PD* and so on. Each copy—fusions included—yields its own star-allele. Thus, in order to correctly call star-alleles of such genes, it is necessary to correctly detect the total number of available gene copies as well as the configuration (i.e.,

structure) of each copy.

Each gene copy can be described by its structural configuration represented as a binary vector $\mathbf{g} \in \mathcal{G}$ that indicates the presence or absence of genic regions in a given configuration (Figure 3.2). Because each star-allele is defined by a matching structural configuration, such configurations must be found before the star-alleles can be accurately called. The size of the configuration vector depends on the number of gene segments that define various structural configurations. For example, the *CYP2D6* gene is divided into $r = 20$ segments that correspond to its exons, introns and flanking regions, because all structural variations are described at the level of whole exons and introns [87]. The total length of the *CYP2D6* configuration vector is $2r$ (i.e. 40) because the vector also includes segments from the neighboring *CYP2D7* pseudogene. This vector can encode any known *CYP2D6* structural configuration: for example, a single *CYP2D6* copy (r ones followed by r zeros), a single *CYP2D7* copy (r zeros followed by r ones), *CYP2D6–2D7* fusion in intron 1 (one followed by $r-1$ zeros, in turn followed by a 0 and $r-1$ ones) and so on. Once these vectors are established, any complex configuration within CYP2D locus can be represented as an aggregate of individual configuration vectors (see Figure 3.2A for an example). Note that valid structural configuration vectors are obtained from the corresponding allele databases, and that each such vector is typically assigned a star-allele identifier (e.g., *CYP2D6*13A* represents a *CYP2D6* fusion with the breakpoint in exon 1).

In a sequenced sample, we only observe the aggregate coverage vector $\mathbf{cn}$ that describes the number of reads covering each genomic loci of interest within the sample (the formation of this vector is described in Supplemental Note S2 of [54]; an example is shown in Figure 3.2B). The goal of Aldy is to find a set of configuration vectors $\{\mathbf{g}_1, \ldots, \mathbf{g}_n\} \subseteq \mathcal{G}$ whose sum is closest to the observed aggregate coverage, where each structural configuration can be selected only

once (an assumption made for the sake of clarity; in practice, Aldy allows selecting the same configurations multiple times). As there might be many such sets, Aldy only looks for the most parsimonious solution: a solution that selects the minimal number of such vectors. This problem, previously dubbed as the Copy Number Estimation Problem (CNEP) [69], can be efficiently solved via integer linear programming (ILP) as follows.

Assume that a gene G is segmented into $2r$ regions. Let $\mathcal{G} = \{\mathbf{g}_1, \ldots, \mathbf{g}_{|\mathcal{G}|}\}$ stand for the set of the available configuration vectors, where $\mathbf{g}_i = [\mathbf{g}_{i,1}, \ldots, \mathbf{g}_{i,2r}]$ and $\mathbf{g}_{i,j} \in \{0, 1\}$ for any i and j. Let $\mathbf{cn}$ be the aggregate coverage vector observed from HTS data, and let us introduce a binary variable z_i for each $\mathbf{g}_i$ that indicates if $\mathbf{g}_i$ is a part of the solution or not. The objective—minimization of difference between the observed aggregate coverage and predicted solution—can be modeled as follows:

$$\min \sum_{j=1}^{2r} \left| \mathbf{cn}_j - \sum_{i=1}^{|\mathcal{G}|} z_i \mathbf{g}_{i,j} \right| .$$

While this model performs well on WGS and targeted data [69], it is rather sensitive to deviations from the expected coverage distribution. It can also not properly handle the cases where the normalized aggregate coverage is not stable or uniform. For targeted panels with non-uniform coverage distributions, aggregate coverage can be "normalized" by dividing it by the coverage of the control sample if it is stable across different samples. Aldy does this automatically for known targeted panels. Thus, Aldy 4 improves the original CNEP formulation by introducing additional optimization terms. This is done by modifying the original objective term and extending it with two additional terms, resulting in a three-term optimization objective.

The first term is the same as the original CNEP objective, but focuses only on the regions

associated with the pharmacogene (and not its pseudogene):

$$o_1 = \sum_{j=1}^{r} \left| \mathbf{cn}_j - \sum_{i=1}^{|\mathcal{G}|} z_i \mathbf{g}_{i,j} \right|.$$

The second term of the objective function considers the interaction between the pharmacogene and the corresponding pseudogene region by considering the changes between their respective region coverage. For example, if the coverage of the exon 2 in *CYP2D6* is 3 and in *CYP2D7* is 2, the resulting region coverage difference would be 1. This difference can be further normalized (in this case, divided by 3). Using normalized differences allows us to handle samples in which the observed aggregate coverage ($\mathbf{cn}$) varies between the regions due to various sequencing and alignment biases. Despite region-specific coverage variation, the relative abundances between the matching gene-pseudogene regions remain constant. This term can formally be expressed as:

$$o_2 = \sum_{j=1}^{r} \left| \frac{\mathbf{cn}_j - \mathbf{cn}_{j+r}}{\nu_j} - \sum_{i=1}^{|\mathcal{G}|} z_i \frac{\mathbf{g}_{i,j} - \mathbf{g}_{i,j+r}}{\nu_j} \right|.$$

Here $\nu_j = \max\{\mathbf{cn}_j, \mathbf{cn}_{j+r}\} + 1$ is the normalization factor.

The final term of the objective function ensures that the ILP solver selects the most parsimonious solution:

$$o_3 = \sum_{i=1}^{|\mathcal{G}|} \mu_i z_i.$$

μ_i is parsimony parameter (by default set to $1/|\mathcal{G}|$). However, some unlikely configurations, such as left fusions, will have higher parsimony scores to reflect the observation that such configurations are rare [88].

Aldy 4's modified CNEP model uses an ILP solver to minimize sum of these three terms

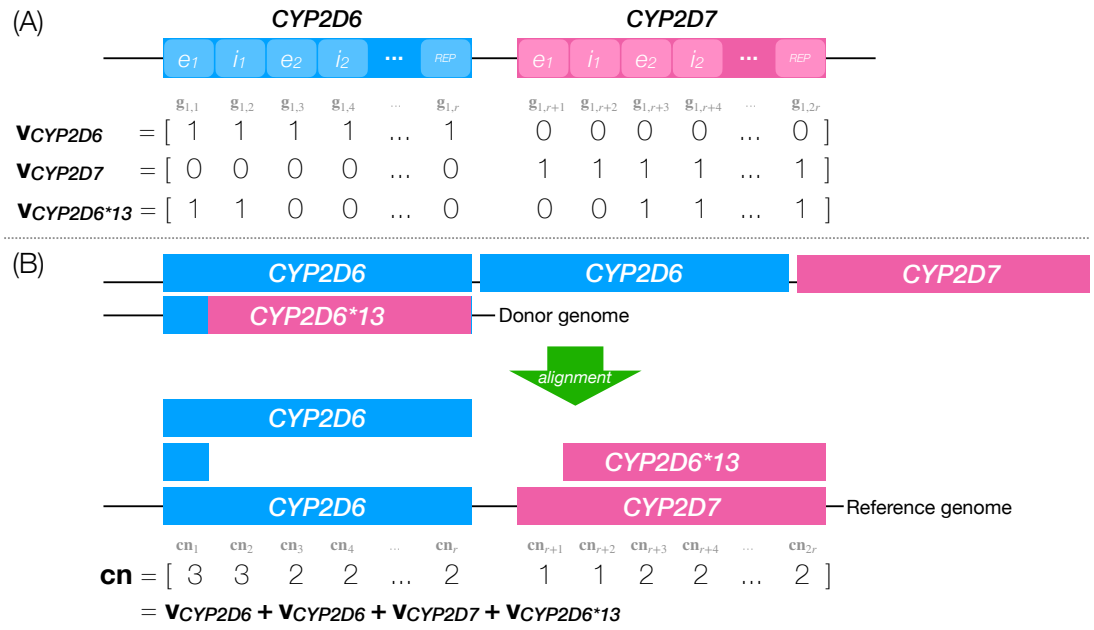


Figure 3.2: A sample decomposition of aggregate coverage into individual structural configurations. (A) An example database of *CYP2D6* structural configurations containing three such configurations ($\mathbf{v}_{\text{CYP2D6}}$, $\mathbf{v}_{\text{CYP2D7}}$ and $\mathbf{v}_{\text{CYP2D6*13}}$). Regions on *top* of the configurations that were defined (i.e., e_1, i_1 etc.) are shaded with lighter color. In this example, $\mathbf{v}_{\text{CYP2D6}}$ corresponds to the $\mathbf{g}_1$. (B) Sample decomposition of the aggregate coverage vector $\mathbf{cn}$, observed after aligning the reads originating from the donor genome (above) to the reference genome (below). As can be seen, $\mathbf{cn}$ can be expressed as the sum of 4 structural configuration vectors from the database.

$o_1 + o_2 + o_3$. These solutions are passed to the later steps that will decide the best overall solution.

3.2.3 Star-allele calling

Aldy now proceeds by assigning the exact star-allele identifier to each of the n structural configurations obtained in the previous step. As stated in Section Methods, a star-allele S_i is defined as a tuple $(\mathbf{g}_i, A_i)$, where $\mathbf{g}_i \in \mathcal{G}$ and $A_i \subseteq \mathcal{M}$. The star-allele assignment problem can also be modelled through the ILP as follows.

Let us indicate the presence of star-allele S_i with a binary variable a_i. Our goal is to select a set of star-alleles $S_1, \ldots, S_n$ such that (1) the set of the structural configurations that describes selected star-alleles is identical to the set the structural configurations from the previous step,

and (2) the difference between predicted and observed coverage for each variant m (denoted as $\mathrm{cov}(m)$) is minimized. In other words, we want to minimize

$$\sum_{m \in \mathcal{M}} \left| \mathrm{cov}(m) - \sum_{i:\, m \in A_i} a_i \right|.$$

While conceptually simple, this model does not account for cases where database definitions are incomplete or incorrect. To account for these cases, the model must be allowed to alter star-allele definitions if needed. Aldy thus introduces new binary variables $p_{i,m}$ and $q_{j,m}$ that indicate if a variant m is to be "removed" from the star-allele S_i (while being present in the database definition A_i), or "added" to it (while being absent in A_i). Then it attempts to minimize the following expression for each variant m:

$$e_m = \left| \mathrm{cov}(m) - \left(\sum_{i: m \in A_i} a_i p_{i,m} + \sum_{i: m \notin A_i} a_i q_{i,m} \right) \right|.$$

As a_i, $p_{i,m}$ and $q_{i,m}$ are all binary variables, their product can be expressed as a set of linear constraints.

The minimization objective can be expressed as:

$$\min \sum_{m \in \mathcal{M}} e_m + \sum_i a_i \left[\sum_m \alpha_{i,m}(1 - p_{i,m}) + \sum_m \beta_{i,m} q_{i,m} \right].$$

Parameters $\alpha_{i,m}$ and $\beta_{i,m}$ are penalties for adding or removing the variant m from allele S_i. Adding a variant is less common than missing a variant, so generally we use $\alpha_{i,m} = 2$ and $\beta_{i,m} = 1$ for any i and m [69]. Note that not all variants are the same: as functional (core) variants can fundamentally alter the behavior of a star-allele (and thus change its designation),

Aldy disallows removing such variants from any star-allele and allows adding novel functional (core) variants to the allele if and only if no other assignment is possible. This is done by setting the corresponding $\alpha_{i,m}$ to a very large value. Note that novel core variants are added only if no other decision can be made.

The star-allele calling model also enforces other constraints: each functional (core) variant must be expressed by at least one allele, and each structural configuration must be expressed by at least one allele compatible with it. Finally, Aldy performs two rounds of star-allele calling for improved accuracy. In the first round, Aldy only considers functional (core) variants and identifies all major (core) star-allele combinations that explain the present functional variants [69]. Being restricted solely to core variants, this step alone often produces multiple candidate solutions. Thus, Aldy then uses the second round to refine the candidate calls from the first round and break the ties by considering the silent variants (subvariants) as well. It finally selects the star-allele with the best second-round objective score as the final call.

The formulation Aldy 4 uses for this step remains similar to the original model used in the older versions of Aldy. The single major difference is the change in the first (functional star-allele calling) round: Aldy 4 can now call star-alleles that contain novel functional (core) variants—a not uncommon event if a gene database is incomplete—if no other call can be made.

3.2.4 Read-based phasing

The above-described model essentially performs a variant of statistical phasing: it utilizes the database knowledge to select the most likely haplotypes that best explain the given observations from the data. While performing well in practice [69], there are nevertheless cases when the

aforementioned model produces multiple equally-likely calls. It is also unable to assign a novel variant to a particular star-allele unambiguously. Finally, in sporadic cases, the above model can produce incorrect results. These challenges can be resolved with long reads that provide long-range phasing information. Aldy 4 newly incorporates the handling of long-range phasing information to the star-allele calling model as follows.

Suppose that there are z fragments $R_1, \ldots, R_z$, each fragment being defined by a set of variants that it spans: $R_j = \{m_1, ...,\} \subseteq \mathcal{M}$. Each sequenced fragment originated from a single star-allele, and can thus be assigned to one of the star-alleles in the dataset. This assignment can be controlled by introducing a binary variable $f_{i,j}$ that is set if and only if a fragment R_j is assigned to S_i. Clearly, $\sum_i f_{i,j}$ must be one for every R_j because each fragment originates from a single allele.

Ideally, we want to assign a R_j to S_i only if such an assignment agrees with the star-allele sequence as much as possible. In other words, we want to minimize the number of disagreements between allele S_i and fragment R_j. Thus, the total disagreement of an assignment can be expressed as follows:

$$e_{i,j} = \sum_{m \in R_j} (1 - p_{i,m} - q_{i,m}) + \sum_{m \in \bar{R}_j} (p_{i,m} + q_{i,m}),$$

where $\bar{R}_j$ denotes the set of variants that are not present in read R_j but are spanned by it.

The total phasing error can be expressed as $\sum_{i,r} f_{r,i} e_{r,i}$. This expression can be added to the objective function of the star-allele calling model. Although the expanded version of this expression contains quadratic terms, each quadratic term is a product of two binary variables and, as such, can be trivially linearized.

As a final remark, note that the number of binary variables in the phasing model is dependent on the total number of present reads and alleles. In some cases, it can exceed half a million variables, making the overall model very costly to solve. The model can be significantly improved by using a smaller random sample of fragments, where the size of the random sample depends on the number of present reads and alleles.

3.2.5 Limitations

Aldy uses ILP solvers to solve the presented models. While ILP solving is NP-hard even when restricted to the models mentioned above [69], all these models are solvable in practice in less than a minute thanks to the state-of-the-art integer programming solvers utilized by Gurobi [47] or CBC [89] solvers.

In some rare instances, Aldy cannot unambiguously call star-alleles from short-read datasets due to the read length limitations and lack of strand information. In these cases, Aldy will report all possible solutions. In some cases, this might be misleading; for example, a *68+*4/*5 call can be reported as *68/*4 (where *5 stands for deletion allele). However, both calls are functionally identical and should be treated as equal (as is done here). Aldy also makes heavy use of the existing star-allele databases to call star-alleles and fusion breakpoints. While it can handle cases where the database is incomplete or lacking, it can theoretically report incorrect results if a present allele is wildly divergent from any allele in the database.

Aldy 4's detection of structural configurations is highly dependent on the stability of coverage across different sequencing runs. While this is not a significant issue for short-read WGS and targeted sequencing panels, the coverage might vary more than expected in PacBio samples. For

this reason, Aldy 4 brings support for the exploration of a broader solution space when needed to account for potential noise.

Finally, note that Aldy 4 does not cover all existing pharmacogenes: genes from the IG and HLA regions, ABC gene families (e.g., *ABCG2*), and the UGT1/2 gene clusters are prominently not included. While some genes, such as *ABCG2*, can be easily supported with a corresponding database file (something that is planned for the future releases), more complex clusters such as HLA or IGH require major changes to the core algorithm to account for challenges posed by those regions, and are better left to the specialized tools such as ImmunoTyper-SR [90].

3.3 Results

We have compared Aldy 4.3[2] (with PharmVar v5.2.3) against Astrolabe v0.8.7.2 [71], StellarPGx v1.2.5 [72], Stargazer v1.0.8 [70] and Cyrius v1.1.1 [74]. Aldy 4 was also compared against Aldy v3.3 [69], the previous version of Aldy. The comparisons were done on a sizeable GeT-RM set of publicly available samples and genes for which genotyping panel validations were available [66, 91, 92]. These samples were sequenced with three technologies: (1) PGRNseq v.3 Illumina-based pharmacogene-targeted panel ([76]; 137 samples), (2) Illumina whole-genome sequencing (WGS; 70 samples), and (3) 10× Genomics sequencing (95 samples). In addition to these samples, Aldy 4 was also run on the set of 45 Coriell samples sequenced by PacBio HiFi pharmacogene-targeted panel [93, 94] and validated by [95]. The percentage of known alleles covered by these datasets for each evaluated gene is available in Supplemental Table S1 of [54].

Aldy 4 and other tools were run on the following 19 genes: *CYP1A1*, *CYP1A2*, *CYP2A13*, *CYP2A6*, *CYP2B6*, *CYP2C8*, *CYP2C9*, *CYP2C19*, *CYP2D6*, *CYP2J2*, *CYP2S1*, *CYP3A4*, *CYP3A5*,

[2] `https://github.com/0xTCG/aldy`

CYP3A7, *CYP3A43*, *CYP4F2*, *DPYD*, *SLCO1B1*, and *TPMT*. While Aldy 4 also supports additional 15 genes, their evaluation was omitted because we did not have the ground truth panel data for these genes. Note that not every tool supports all of these genes: as a rule of thumb, Stargazer, Aldy 3 and Aldy 4 provide the broadest support, while the other tools are geared towards a small subset of these genes (typically CYP genes, such as *CYP2D6* and *CYP2C19*).

In an ideal world, we would have a "perfect" phase for each sample and would be able to evaluate each tool against such phase. Unfortunately, the available ground truth data is obtained through genotyping panels and assays designed to detect only the common *major star-alleles* (or *core alleles*; alleles defined solely by functional, or core, variants). These panels often cannot call minor star-alleles (or *sub-alleles*; alleles defined by non-functional variants, or subvariants, and functionally indistinguishable from the major star-alleles), as well as less common alleles. The low resolution of the available ground-truth data and the differences in database specifications between the different tools necessitated a few accommodations within the evaluation process for the sake of fairness. First, we updated ground truth calls that missed the presence of less common variants and alleles. Updates were only done if there was a consensus between the star-allele calling tools that differed from the ground truth data and if an updated call extended the validated allele definition (i.e., if the variants defining the validated allele also form a part of the consensus definition). Note that a similar approach was used in [69]. Each updated call was further manually inspected to ensure that the variants missing from the ground truth calls are indeed present and not sequencing artifacts. In rare instances, it was hard to precisely distinguish the presence of the variant, especially if the variant allele frequency (VAF) was too low (alleles with lower VAFs are sometimes caused by the sequencing or read alignment bias, especially in the presence of pseudogenes, and are typically validated through Sanger sequencing). Samples with

such variants were marked as "Need Validation" (Table 3.1). For such samples, calls that either used or ignored such ambiguous variants were deemed "correct". Second, we have followed the common strategy employed in clinical studies [79] by only comparing the major (core) star-allele calls and ignoring the minor star-allele (sub-allele) designations. In other words, only the phasing of functional (core) variants was considered; subvariants and silent variants that do not alter the functionality of an allele were ignored (i.e., a *1A/*2B minor star-allele call was treated as a functionally equivalent *1/*2 major star-allele call. Note that major (core) star-alleles are typically distinguished by the number (e.g., *1 functionally differs from *2), while minor star-alleles (sub-alleles) are traditionally distinguished by a letter (e.g., *2A and *2B harbor different silent variants despite sharing common core variants), or by a numerical suffix in the recent PharmVar definitions (e.g., *2.001 instead of *2A).

The further complication lies in the discrepancies among the databases themselves: different tools ship with different databases and often augment such databases with custom entries. For that reason, identical alleles with differing names were treated equally, and ambiguities were resolved in the individual tool's favor (e.g., if a tool's call could be interpreted as correct, we called it as correct; this way, we ensured that custom database entries were accounted for). The exact criteria used for allele updates and allele comparisons is listed in the Supplemental Note S1 of [54].

Where possible, the *CYP2D8* region was used as the copy number neutral region; exceptions include Aldy 4 using *F1* region for the PacBio data. Some tools, such as Astrolabe and Stargazer, required VCF files; where needed, VCFs were generated by `BCFtools` [80].

All results were obtained on machines with Intel Xeon E5-2680v4 and 8260 CPUs. Each evaluated tool genotypes a single gene in a single sample within a few minutes, regardless of the

sequencing technology used. However, note that Aldy 4 only needs BAM/CRAM to run; other tools often require VCF or GDF files that can take significant time to generate.

Overall, the best accuracy on short-read datasets (PGRNseq v.3, Illumina WGS and 10× Genomics) was achieved by Aldy 4 (98.42%), followed by Aldy 3 (96.78%), StellarPGx (90.40%), Astrolabe (86.50%), Cyrius (82.63%) and Stargazer (76.47%).

3.3.1 PGRNseq v.3

Aldy 4, Aldy 3, Stargazer and Astrolabe were run on 137 PGRNseq v.3 targeted sequencing [76] samples from the GeT-RM collection. PGRNseq v.3 targets common pharmacogenes and sequences them at high depths (up to 1000× per loci).

Note that we could not get either Stargazer or Astrolabe to run on targeted sequencing data natively; thus, VCF files were provided as an input for these tools. Because of the limited nature of VCF files, these tools were unable to call copy number changes and fusions on this dataset. While Stargazer has a mode for targeted data, we were unable to get good results with it. Comparison with StellarPGx was omitted as it does not support targeted sequencing data.

As can be seen in Table 3.1, Aldy 4 identifies nearly all of the alleles in all genes correctly—more than the other two tools—with a total accuracy of 98.45%. In some cases (e.g., failed cases in the genes *CYP1A1* and *CYP2B6*), no caller was able to call correct star-alleles because the PGRNseq panel did not sequence the variant of interest (e.g., a non-exonic downstream variant rs4646903 that defines *CYP1A1*2A* was not covered by the panel at all).

On this dataset, Aldy 4's performance is only marginally better than Aldy 3. This is expected as neither of the model updates unique to Aldy 4 applies to the high-quality PGRNseq

Table 3.1: Summary of the star-allele calls generated by six tools (Aldy 4, Aldy 3, StellarPGx, Stargazer, Astrolabe and Cyrius) on 137 GeT-RM samples sequenced by three different technologies. Bold results indicate the best tool for a given genotyping platform. Some tools do not support all genes; those cases are indicated with a dash (—). Checkmark (✔) indicates the call that matches the updated validation panel star-allele call; a cross-mark (✖) indicates the mismatch. Number of updated panel calls, as well as the calls that need further validation (marked with N.V.), is indicated at the beginning. Note that the total number of samples varies across genes and technologies due to the availability of sequencing data and ground truth validation. Detailed results are available at `https://github.com/0xTCG/aldy`.

			PGRNseq v.3							
			Aldy 4		**Aldy 3**		**Stargazer**		**Astrolabe**	
Gene	Updated	N.V.	✔	✖	✔	✖	✔	✖	✔	✖
CYP2D6	0	0	134	3	135	2	68	69	71	66
CYP2A6	12	1	137	0	133	4	117	20	—	
CYP1A1	31	0	78	18	79	17	78	18	—	
CYP1A2	1	0	96	0	96	0	96	0	—	
CYP2A13	20	0	96	0	96	0	96	0	—	
CYP2B6	36	5	135	2	134	3	118	19	—	
CYP2C8	5	0	137	0	137	0	137	0	135	2
CYP2C9	1	0	137	0	137	0	120	17	128	9
CYP2C19	3	0	137	0	137	0	128	9	137	0
CYP2J2	1	0	96	0	96	0	85	11	—	
CYP2S1	4	0	93	3	94	2	93	3	—	
CYP3A4	0	0	137	0	137	0	128	9	—	
CYP3A5	0	0	137	0	137	0	107	30	—	
CYP3A7	7	0	96	0	96	0	45	51	—	
CYP3A43	10	0	87	9	87	9	87	9	—	
CYP4F2	11	0	137	0	137	0	124	13	137	0
DPYD	46	0	137	0	136	1	70	67	—	
SLCO1B1	42	1	136	1	114	23	133	4	135	2
TPMT	0	0	137	0	137	0	137	0	137	0
Accuracy			**98.45%**		97.37%		84.93%		91.76%	

	Illumina WGS											
	Aldy 4		**Aldy 3**		**StellarPGx**		**Stargazer**		**Astrolabe**		**Cyrius**	
Gene	✔	✖	✔	✖	✔	✖	✔	✖	✔	✖	✔	✖
CYP2D6	69	1	63	7	65	5	50	20	43	27	70	0
CYP2A6	70	0	68	2	67	3	58	12	—		—	
CYP1A1	66	0	66	0	—		64	2	—		—	
CYP1A2	66	0	66	0	66	0	66	0	—		—	
CYP2A13	66	0	66	0	—		59	7	—		—	
CYP2B6	69	1	69	1	68	2	56	14	—		—	
CYP2C8	70	0	70	0	—		58	12	68	2	—	
CYP2C9	70	0	70	0	70	0	54	16	67	3	—	
CYP2C19	69	1	69	1	69	1	67	3	70	0	—	
CYP2J2	66	0	66	0	—		51	15	—		—	
CYP2S1	66	0	66	0	—		65	1	—		—	
CYP3A4	70	0	70	0	70	0	49	21	—		—	
CYP3A5	69	1	69	1	68	2	46	24	—		—	
CYP3A7	66	0	66	0	—		54	12	—		—	
CYP3A43	66	0	66	0	—		55	11	—		—	
CYP4F2	70	0	70	0	66	4	57	13	26	44	—	
DPYD	70	0	70	0	—		28	42	—		—	
SLCO1B1	70	0	68	2	—		65	5	69	1	—	
TPMT	70	0	70	0	—		63	7	70	0	—	
Accuracy	99.69%		98.92%		97.28%		81.80%		84.29%		**100.00%**	

dataset with stable coverage. Minor changes are mostly due to the differences in the variant calling (e.g., Aldy 4's incorporation of quality scores and mapping qualities).

Table 3.1: (Table continued from the previous page)

	10× Genomics											
	Aldy 4		**Aldy 3**		**StellarPGx**		**Stargazer**		**Astrolabe**		**Cyrius**	
Gene	✔	✖	✔	✖	✔	✖	✔	✖	✔	✖	✔	✖
CYP2D6	78	17	57	38	54	41	44	51	51	44	62	33
CYP2A6	69	26	42	53	81	14	34	61		—		—
CYP1A1	88	1	89	0		—	56	33		—		—
CYP1A2	89	0	89	0	89	0	77	12		—		—
CYP2A13	89	0	89	0		—	64	25		—		—
CYP2B6	94	1	94	1	63	32	61	34		—		—
CYP2C8	94	1	95	0		—	64	31	93	2		—
CYP2C9	95	0	95	0	94	1	54	41	92	3		—
CYP2C19	94	1	95	0	62	33	65	30	95	0		—
CYP2J2	89	0	89	0		—	58	31		—		—
CYP2S1	89	0	89	0		—	76	13		—		—
CYP3A4	95	0	95	0	91	4	51	44		—		—
CYP3A5	95	0	95	0	87	8	53	42		—		—
CYP3A7	89	0	88	1		—	49	40		—		—
CYP3A43	88	1	88	1		—	66	23		—		—
CYP4F2	94	1	95	0	88	7	49	46	36	59		—
DPYD	95	0	94	1		—	30	65		—		—
SLCO1B1	93	2	85	10		—	74	21	93	2		—
TPMT	95	0	95	0		—	80	15	95	0		—
Accuracy	**97.11%**		94.04%		83.51%		62.68%		83.46%		65.26%	

3.3.2 Illumina WGS

We have run all tools on 70 Illumina HiSeq-sequenced WGS samples from the GeT-RM sample collection. These samples were sequenced with an average depth of roughly 30×. The details are also available in Table 3.1.

Here, Aldy 4 again calls nearly all star-alleles correctly and genotypes more samples than the competition for every considered gene. The only exception is *CYP2D6*, where Cyrius genotypes two samples (NA21781) more than Aldy 4. In this case, Aldy 4 misses the non-functional *68 and identifies the *2 allele as *65; however, the *65 allele extends the *2 allele with a single variant (rs1065852), and it is unclear if this allele is indeed a *2 or a *65.

Aldy 4 and other tools were able to correctly call alleles defined by intronic and downstream variants across the genes on this data. Note that the main reason behind the Stargazer's lower

accuracy on this dataset was copy number calling: while Stargazer often identified the star-allele correctly, it would often call them more times than needed (e.g., *1/*2+*2 instead of *1/*2). Note that Aldy 4 only calls copy numbers and fusions on genes that are known to harbor such changes; otherwise, it assumes that two copies are present.

Note that Astrolabe used a modified *CYP4F2* database whose allele nomenclature differed from the other databases. Thus, the comparison with Astrolabe on *CYP4F2* was omitted for the sake of consistency. We also observed a large number of mismatches in *SLCO1B1* across all tools due to the incomplete panel validation and inconsistent database specifications used by various tools.

Finally, the improvements in the copy number model and more sensitive variant calling in Aldy 4 account for a few improved calls on more complex *CYP2D6* and *CYP2A6* samples.

3.3.3 10× Genomics

We have run all tools on 95 GeT-RM samples sequenced by a 10× Genomics WGS sequencer. The average depth of sequencing was roughly 40×. Because several important pharmacogenes reside within repeated regions of the human genome, EMA aligner [96] with density-based optimization mode was used for improved alignment of the 10× reads to the reference genome (hg19 at the time of alignment; the same results should be expected when aligning the data to hg38 as well because the gene regions of interest did not undergo major changes between the two releases). The comparison details are available in Table 3.1.

Although the 10× Genomics protocol uses Illumina HiSeq for sequencing, the read coverage is not as uniform as it is in an average Illumina WGS sample. 10×-specific biases also result

in quite a few misaligned reads compared to the WGS data. For this reason, the overall allele calling accuracy is lower than the WGS dataset; this is especially evident in Stargazer, where the accuracy of its copy number detection module is even lower than in WGS data.

However, Aldy 4 still correctly calls the majority of alleles (with 97.11%) accuracy, especially when compared to the other tools. The most challenging genes for all tools were *CYP2A6* and *CYP2D6*. Aldy's accuracy is lower in these genes, primarily due to the occasional copy number mismatch (due to the coverage unevenness) and sequencing artifacts (where many misidentified variants were either an artifact or were under-sequenced). Note that Aldy 4 benefited from the novel phasing module that was able to successfully utilize 10× Genomics barcodes to link long-distance variants together. Finally, we observe significant improvements over Aldy 3 in *CYP2D6* and *CYP2A6* samples on this dataset due to an improved copy-number model that better handles noisy coverage and ambiguous variants (a common case in 10× Genomics samples), and is as such able to improve the calling accuracy up to 30% in these genes.

3.3.4 PacBio HiFi

Finally, Aldy 4 was run on two sets of PacBio HiFi samples sequenced by a custom targeted pharmacogenomics panel [93, 94]. The first set contained 24 samples, while the second set was comprised of 21 samples. The coverage of these datasets varies—it can be as low as 10×—and at times exceed even 200×. Aldy's calls were compared with those of Astrolabe. While none of the other tools support PacBio long reads natively, we were able to at least run Astrolabe in VCF mode. The validation data was obtained from [95] and [91]. The call details are available in Table 3.2.

Star-allele calls generated by Aldy 4 agree with the ground truth in all genes except for a few *CYP2D6* calls and one *CYP2C9* call. Furthermore, its calls augmented and phased many ground-truth calls generated by panels with limited variant coverage with additional variants observed by PacBio data (Table 3.2). Aldy was also able to find and phase alleles that have not been cataloged in *CYP2B6*, *CYP2C19*, *CYP3A4*, *DPYD* and *SLCO1B1*. Further validation of such novel calls, as well as of the calls that were deemed ambiguous, is needed to fully confirm and understand such alleles.

When it comes to *CYP2D6*, Aldy 4's calls disagree with the ground-truth data due to the difference in predicted copy number. In two instances, Aldy 4 called an additional copy (i.e., *1+*1 instead of *1, and *4+*4 instead of *4), while in the other two instances, Aldy 4 did not call an existing copy (i.e., it called *2 instead of *2+*2 and *10 instead of *10+*10). In one instance, Aldy called *36 instead of *10 (note that these alleles are nearly identical, with the only difference being a conversion of exon 9 in *36); in the final instance Aldy did not call the non-functional *68 fusion allele. In all these cases, the observed coverage was noisy, and further validation is needed to ascertain the exact copy number of these samples. Let us also point out that Astrolabe's calls in genes *CYP2C19* and *SLCO1B1*, as well as on *CYP2D6* on the second dataset, were highly ambiguous, often containing more than ten functionally different solutions.

3.3.5 Other remarks

Many tools often confuse *CYP2B6*6* and *CYP2B6*9* alleles that differ only in the variant rs2279343. This variant is often either under-sequenced or covered by ambiguous reads that potentially originate from the neighboring *CYP2B7* pseudogene, and is thus hard to call with

Table 3.2: Overview of the star-allele calls generated by Aldy 4 and Astrolabe on PacBio HiFi targeted data. Some tools do not support all genes; those cases are indicated with a dash (—). Check mark (✔) indicates the call that matches the updated validation panel star-allele call; a cross-mark (✖) indicates the mismatch. Number of updated panel calls, as well as the calls that need further validation (marked with N.V.), is indicated at the beginning.

Dataset 1						
			Aldy 4		**Astrolabe**	
Gene	Updated	N.V.	✔	✖	✔	✖
CYP2D6	6	1	22	2	9	15
CYP1A2	1	0	24	0	—	
CYP2B6	8	0	24	0	—	
CYP2C8	2	0	24	0	15	9
CYP2C9	3	1	24	0	23	1
CYP2C19	4	0	24	0	9	15
CYP3A4	14	0	24	0	—	
CYP3A5	0	0	24	0	—	
CYP4F2	0	0	24	0	13	11
DPYD	20	0	24	0	—	
NUDT15	0	1	24	0	—	
SLCO1B1	8	0	24	0	24	0
TPMT	1	0	24	0	18	6
Accuracy			**99.36%**		66.07%	
Dataset 2						
			Aldy 4		**Astrolabe**	
Gene	Updated	N.V.	✔	✖	✔	✖
CYP2D6	0	0	18	3	14	7
CYP1A2	0	0	21	0	—	
CYP2B6	6	0	21	0	—	
CYP2C8	0	0	21	0	12	9
CYP2C9	0	0	20	1	20	1
CYP2C19	2	0	21	0	19	2
CYP3A4	0	0	21	0	—	
CYP3A5	0	0	21	0	—	
CYP4F2	1	0	21	0	12	9
DPYD	8	0	21	0	—	
SLCO1B1	6	0	20	1	21	0
TPMT	0	0	21	0	19	2
Accuracy			**98.02%**		79.59%	

high confidence in some technologies (e.g., PGRNseq v.3). When the true call was ambiguous, both possible calls were deemed "correct". Similar cases were also observed with *CYP2A6*1*

and *CYP2A6*35* alleles. Further validation is needed to properly ascertain the true existence of these alleles in problematic samples.

If multiple allele calls were generated by a tool for a given sample and gene combination, the call was deemed "correct" if at least one such multi-call matched the ground truth. Note that the prevalence of multiple calls was overall low: around 1.1% for Aldy 4, 2.7% for Aldy 3, 1.9% for Stargazer, 1.9% for StellarPGx and 15.7% for Astrolabe. Aldy 4's new phasing model was a significant factor for a low multi-call rate: while the rate was 1.6% on PGRNseq v.3 and 1.8% on WGS samples due to the short read lengths of such samples, it decreased to 0.5% on 10× and PacBio samples that allowed better phasing. The vast majority of ambiguous calls were observed when genotyping *CYP4F2* and *SLCO1B1*. Finally, note that Aldy 4 generated no more than three diplotypes for each ambiguous call.

3.4 Discussion

Pharmacogenomics is becoming a key component of evidence-based medicine [97]. Genes like *CYP2D6* and *CYP2C19* regulate a large portion of clinically prescribed drugs; other genes, such as *HLA* or *IGH* gene cluster, are vital for understanding the immune response [98, 90]. As their function is dependent on their haplotype, it is of vital importance to genotype and haplotype these genes prior to administering medical treatment [99]. High-throughput sequencing technologies are a natural candidate for this process, especially when considered that the currently available clinical genotyping panels are often restricted only to the most common genotypes and struggle to detect more complex structural alterations within pharmacogenes.

In this work, we have presented Aldy 4, the first tool that can accurately and consistently

call star-alleles in data from various sequencing technologies, including but not limited to long-read PacBio data, Illumina short-read sequencing in all of its flavors (i.e., whole-genome, whole-exome, and targeted capture data), as well as the 10× Genomics barcoded data. Aldy 4 achieves this by employing combinatorial optimization models to solve various challenges associated with calling pharmacogenetic haplotypes from sequencing data, such as copy number and structural variation detection, variant calling and variant phasing, ultimately resulting in a star-allele decomposition of a gene of interest. We have shown the strength of Aldy 4's approach through a series of comparisons against the current state-of-the-art star-allele callers, in which Aldy 4 performed the best. We hope that Aldy 4 will be of vital importance to clinicians in tailoring prescription recommendations, thus leading to improved medical care.

There are still some open questions left that need to be answered in future work. Most importantly, the panel-validated calls improved by the star-allele callers through the use of HTS data—often containing novel alleles not previously cataloged in the existing databases—need to be validated in a wet-lab environment for all genes presented, as was done recently for a selection of CYP2C genes [100]. More tests are also needed on larger cohorts to accurately evaluate the precision of these tools, Aldy 4 included, on rare fusions. The incorporation of other highly polymorphic pharmacogenomics regions, such as *HLA* or *IGH*, should also be considered, as Aldy (and other evaluated pharmacogenomics tools) is currently unable to handle the complexities of such regions. Finally, the complete characterization of minor star-alleles, accompanied by the careful characterization of non-coding variants, is also needed to understand the full effect of pharmacogenes on the treatment and drug dosage decisions.

Chapter 4: Identifying distinct developmental methylation programs and cell type compositions in Glioblastoma using `Qombucha`

The conversion of genes into their functional products, typically proteins, is called gene expression. It's a tightly regulated process that enables a cell to respond to its changing environment by making more or less of a specific protein [101]. While many processes control the expression of selected genes in specific environments, a biological process called *DNA methylation* is used by organisms to enhance or suppress gene expression. DNA methylation involves adding "methyl" groups to the DNA molecule either within genes or regions that regulate gene expression such as the *promoter* region. It is a mechanism by which the activity of a gene can be controlled without changing its sequence composition.

4.1 DNA methylation at CpG sites

DNA methylation typically occurs at the nucleotide cystine (the character *C* among $\{A, C, G, T\}$ nucleotides). Nucleotides *C* and *G* occurring next to each other in the DNA is called a "CpG" site (phosphate links any two nucleotides in the DNA). "CpG islands" [102] are the regions in the genome that have a high frequency of CpG sites.

DNA methylation is largely preserved under mitosis, i.e., daughter cells of the same cell type inherit methylation of the same sites as their parent cells. However, various pathways seem

to contribute to excessive methylation/demethylation of somatic cells, whose effect is more pronounced in cancer [103]. The following (yet) unpublished work explores the problem of simultaneously characterizing the types of brain cells and their methylation patterns in patients with an aggressive form of brain cancer.

4.2 Motivation for the cell type inference problem and previous work

Glioblastoma (GBM), one of the most aggressive and common malignant brain tumors, has a poor prognosis due to its invasive nature and resistance to conventional therapies [104, 105]. Treatment failures in GBM are attributed to the limitations of doing brain surgery [106] and to high heterogeneity both within and across GBM tumors [107, 108, 109]. The intertumor heterogeneity of GBM can be characterized by differences in molecular markers, including DNA, mRNA, and methylation [110]. [111] pioneered mRNA-based classification of GBM and established the Classical, Mesenchymal, Proneural, and Neural subtypes. These transcriptomic subtypes exhibit different responses to therapies [111], although genomic analyses showed that multiple subtypes are often present in the same patient [107, 112]. To an extent, prognosis can also be predicted via the presence/absence of somatic alterations in genes such as *IDH1*, *IDH2*, *PTEN*, and *EGFR* [113, 110].

There has been a growing focus on DNA methylation-based classification and subtyping [114, 110, 115]. [116] characterized six DNA methylation-based GBM subtypes: K27, G34, IDH1, RTK (receptor tyrosine kinase) I, RTK II, and MES (mesenchymal), among which RTK I, RTK II, and MES are most frequent in adult GBM [117]. These three DNA methylation-defined subtypes have been associated with differences in incidence of seizures, spatial location

within the brain, genomic features, and tumor purity. RTK I tumors are mainly found in the right frontal lobe of the brain [118] and are enriched for platelet-derived growth factor receptor-alpha (*PDGFRA*) gene amplifications. RTK II tumors are often found in the left-hemispheric regions [118] and are associated with higher seizure incidence [119] and frequent epidermal growth factor receptor (*EGFR*) gene amplifications. MES tumors are commonly observed in the left-hemispheric regions [118], tend to have lower tumor purity and neurofibromin-1 (*NF1*) aberrations and increased level of immune cell infiltration [120]. Using a subset of 32k methylation probes with high variance across subtypoes, [121] extended this line of work to obtain a classifier for 98 brain and central nervous system (CNS) tumor subtypes, including GBM subtypes RTK I, RTK II, and MES.

Single-cell RNA-seq data have improved genomic understanding of GBM [108, 122, 123, 112, 124], but all the larger data sets rely on bulk measurements. In this study, we introduce and analyze a new data set of 896 IDH-wildtype (no somatic mutation in *IDH1* or *IDH2*) tumors with bulk methylation data. For comparison between IDH-wildtype and IDH-mutant tumors, we also introduce a new dataset of 961 IDH-mutant tumors.

For analyzing such bulk methylation data from a large cohort, we introduce a new method `Qombucha` to address: how can one estimate the proportion of each cell type from among a defined set (next paragraph and Methods) of cell types and their unseen progenitors? Cell type estimation problems/solutions are generally called "deconvolution". For bulk transcriptomics data, there are various deconvolution methods some of which jointly estimate the proportion of each cell type and the proportion of total expression of a gene attributable to that cell type [125, 126, 127, 128, 129, 130, 131]. The deconvolution of bulk methylation data has also been studied

by others using the following equation, where B is the observed bulk data.

$$B \approx PC \tag{4.1}$$

The prior work on deconvolution falls into three categories: (i) supervised (reference-based), (ii) unsupervised (reference-free), and (iii) semi-supervised (hybrid) [132]. The supervised methods rely on the known methylation profiles of cells of interest, represented by known matrix C, to estimate the cell type compositions, represented by unknown matrix P in Equation (4.1). [133] conducted pioneering work in methylation deconvolution and developed a linear constrained projection-based algorithm that estimates cell type proportions in bulk sample data by minimizing the differences between the observed bulk sample data and the projection of a reference matrix onto the cell-type proportion matrix. Other methods, such as MethAtlas [134] and MethylResolver [135], have employed least squares regression. In addition, machine learning approaches have been developed, including the support vector regression-based MethylCIBERSORT [136] and the deep neural network-based cfSort [137]. The unsupervised methods, in contrast, have been developed for situations where both P and C are completely unknown and need to be inferred simultaneously. The unsupervised methods are typically based on non-negative matrix factorization (NMF), including MeDeCom [138] and EDec [139], or on expectation–maximization (EM) algorithms, such as MethylPurify [140] and PRISM [141]. The semi-supervised methods combine unsupervised techniques and prior knowledge to deconvolute. In these approaches, C includes fully known reference panels but leaves room for unknown cell types. CelFiE [142], for example, uses a block coordinate descent-like algorithm to infer the compositions for known cell types and a user-defined number of unknown cell types. PRMeth [143] employs an iteratively op-

timized NMF framework to simultaneously infer the methylation profiles of unknown cell types and the proportions of both known and unknown cell types.

Methylation deconvolution has been applied in GBM to elucidate the tumor's epigenetic landscape. Previously, [144] used a reference-free deconvolution method developed by [145] to elucidate cell types and cell states in bulk GBM methylation data. [146] used the R package InfiniumPurify [140] to deconvolve GBM tumor methylation data and assess tumor purity. However, none of these prior studies have specifically addressed the question of deconvolving GBM methylation data to explore GBM subtype-specific brain cell development processes. Moreover, the previous applications of methylation-based deconvolution generally consider observable cell types but not the unobservable progenitors.

In contrast, our analysis of cell types in GBM is focused on the six *non-neuronal differentiated cell types* studied by [147] as well as their *progenitors*; the differentiated cell subsets of most interest include (in alphabetic order): astrocytes (ASC), endothelial cells (EC), microglia (MGC), oligodendrocytes (ODC), oligodendrocyte precursors (OPC), and (the union of) vascular and leptomeningeal cells (VLMC). Studies of various tissue types and organs support the notion that developmental programs, which differentiate unobservable progenitor cells to observable mature cells are branched processes [148, 149, 150]. Therefore, we consider the observable and progenitor cell types as a tree which can substantially aid in the deconvolution because methylation status is typically inherited from parent cell to child cell. Previous research has shown that GBM has complex cell origins, including neural stem cells (NSC) and non-neuronal cells like ASC and OPC [151]. GBM tumors also display diverse cell states with transcriptomic signatures that are ASC-like, neural progenitor cell(NPC)-like, OPC-like, and MES-like [112].

After we estimate cell type proportions, we investigate associations between cell types

and tumor types using the methylation-based classification method of Capper et al. [121], IDH mutation status, and overall survival. We also investigate which methylation sites give the most signal for inferring cell type proportions and whether the methylation status of those sites is stable in cell differentiation as explained below.

Previously, [147] performed single-cell whole genome bisulfite sequencing (scWGBS-seq) for 40 major brain cell types in 46 brain regions across samples from three healthy males and reported that 800 features grouped from 156k CpG sites are sufficient to differentiate mature brain cell types - whose developmental program was represented as a tree. It is possible to trace the origin of each cell by identifying those CpG sites that go through methylation status change in distinct stages of development/differentiation [152]. Therefore, we used a subtree of the developmental tree and features in [147] to define our tree-based deconvolution problem below. The individuals from whom the samples were used for the model in [147] are healthy and thus, we avoid the potential ciruclarity of using GBM-based modeling to define the developmental tree-based based deconvolution problem below.

Analysis of a normal development tree process in cancer data could be confounded by the established theory of [153] that subclones within solid tumors such as GBM evolve in a tree-like manner. The study of inferring the tree history of tumors from static snapshots of tumor omics data is called "tumor phylogenetics"; most tumor phylogenetics studies to date have been based on somatic mutations or copy-number variants [154, 155, 156, 157]. Some tumor phylogenetics studies have instead used methylation data as the input and found that evolution of "tumor lineage informative" CpG sites also forms a tree [158, 159, 160, 161]. The overlay of a methylation-based tree structure for normal development and another methylation-based tree structure for tumor progression could be problematic if the same (un)methylated CpG sites are most informative for

both trees. We show early in Results that this overlap is not substantial in a statistical sense.

Because the methylation profiles of progenitor cell types in the developmental tree are not available [147], `Qombucha` employs the empirical observation that, for many features, a progenitor's methylation state matches that of its mature descendants when those descendants agree. We show that the tree inheritance property enables the deconvolution of the bulk GBM methylation data by jointly estimating each sample's cell-type composition and the unknown progenitor methylation values across the cohort. Intuitively, this allows one to attribute portions of the bulk signal to specific developmental paths (cell-of-origin/trajectory) while preserving tumor-evolutionary information carried by lineage-informative CpG sites [158, 121, 162, 160, 161, 147].

4.3 Methods

4.3.1 Simultaneous inference of cell type compositions and unknown methylation values in progenitor profiles

We formally define the problem of simultaneous inference of cell compositions and unknown methylation values in progenitor cells as a Quadratic Program (QP) below. The input to the QP is: (i) a matrix of bulk methylation data B, where each row corresponds to a vector of methylation *features* of a patient tumor sample, and (ii) a partially completed reference panel matrix C_p informed by the tree of brain development history with mature brain cells as leaves.

In the framework introduced by [147], a *feature* is the average methylation value between 0 and 1 among a set of individual methylation sites. To incorporate information from brain development, we first determine the methylation profiles of certain features in non-neuronal pro-

genitors using the methylation profiles of mature brain cell types and the tree structure reflecting the history of brain development. The features of each progenitor whose methylation values can be determined are those that have consistent methylation values among all mature descendant cell types of the progenitor. The values of the remaining undetermined features in the progenitor profiles can be simultaneously inferred along with the cell composition using a quadratic programming (QP) formulation summarized in Figure 4.1. As described below, the aim of the QP is to deconvolve the input matrix B into the following two matrices: (i) P - the cell type proportion matrix, and (ii) C_f - the completed reference panel matrix (i.e., the completed version of C_p).

4.3.1.1 Formulation of the Quadratic Program

The inputs to the QP-based deconvolution method include a bulk methylation profile matrix B of size $n \times m$, where n is the number of patients and m the number of methylation features, and a partially filled reference panel matrix C_p of size $r \times m$, where $r = 13$ (for the GBM data) is the number of cell types. The reference panel matrix $C_p = \{c_{k,j}\}$ consists of a stable methylation value per feature, for each mature and progenitor brain cell in the tree of brain development history. The elements of matrix C_p are fully known for mature cell types, while those for progenitor cell types they may not be fully determined. The exact construction of C_p is provided in the Section 4.3.1.2.

Let $P = \{p_{i,k}\}$ be the matrix of size $n \times r$ that stores the inferred fraction of each cell type for each patient, with $p_{i,k}$ being the inferred fraction of cell type k in patient sample i. In each patient sample i, the sum of cell fractions across all r cell types must be equal to 1. Let U be the set of indices of unknown entries in matrix C_p. The value of every unknown entry

$c_{x,y}|_{(x,y)\in U}$ must be between 0 and 1. With $\|X\|$ representing the Frobenius norm of matrix X, the simultaneous inference problem can be formulated as a QP as follows:

$$\begin{array}{lll} \text{minimize:} & \|B - PC_p\|^2 & \\ \text{subject to:} & \forall i \in \{1 \dots n\} : & \sum_{k=1}^{r} p_{i,k} = 1 \\ & \forall i \in \{1 \dots n\}, \forall k \in \{1 \dots r\} : & 0 \leq p_{i,k} \leq 1 \\ & \forall (x,y) \in U : & 0 \leq c_{x,y} \leq 1 \end{array} \tag{4.2}$$

We use Gurobi [47] (through its Python API) to solve the QP to simultaneously infer the estimated cell type compositions and the unknown entries in the partially complete progenitor profiles.

4.3.1.2 Cell hierarchy-informed partial inference of progenitor cell type profiles

This section details the construction of the partial progenitor cell profiles from the methylation values of mature cells. Based on prior knowledge that GBM mostly originates from non-neuronal cells [151], we used the CpG sites and single-cell methylation reported by [147] to construct a rooted binary neighbor-joining [163] tree for the 6 non-neuronal cell types reported in the same study, including astrocyte (ASC), oligodendrocyte (ODC), oligodendrocyte precursor (OPC), microglia (MGC), epithelial cell (EC), and vascular and leptomeningeal cell (VLMC) to reflect their development history (Figure 4.5b). In this tree, which we denote by T, the mature non-neuronal cell types are represented as leaves, and their presumptive progenitors are represented as internal nodes. The representation of the differentiation of mature cells as a branched process, i.e., a tree has been presented in [147]. The tree T we obtained has topologically differ-

ences from the cell hierarchy tree from [147] and in particular it has a clade consisting of oligodendrocytes (ODC) and oligodendrocyte progenitors (OPC) which is biologically more plausible than the tree in [147] in which ODC and OPC are apart. A description of how T is constructed from the non-neuronal brain cell single-cell methylation profiles is in Section 4.3.4.1, and how methylation feature values of the mature cell types, i.e., the leaves of T are determined can be found in Section 4.3.4.3.

Next we infer the unknown methylation feature value for a progenitor, i.e. an internal node of T, so that it is "consistent" with the known values of the said feature across the progenitor's descendants in T. Specifically, we identify for each internal node, the "range" of possible methylation values of the feature, from the node's children, in a bottom-up manner (see Section 4.3.4.3 for details). We start by assigning to each feature of each leaf a range that includes only the known methylation value for that cell type. Then, for each internal node with children whose range of values for that feature has already been determined, we identify the smallest contiguous range that fully includes these two ranges; if this range is larger than a user defined threshold δ, we assign this range to the feature of the internal node - otherwise, we calculate the median value of this range and assign to the feature of the internal node (the range that only includes) this median value. The pseudocode for this process is given in the "bottom-up phase" of Algorithm 2 (until Line 22). When the process terminates at the root of the tree, if any given feature of an internal node is assigned a single value, this value will be fixed. If instead, the feature is assigned a range of values for this internal node, a specific value from this range will be determined by `Qombucha`. The input to `Qombucha` is thus these partially determined feature values, represented as reference panel C_p of size $r \times m$, where r is the number of cell types, m is the number of methylation features, and thus row c_k is the methylation profile of cell type k. C_p has a fixed

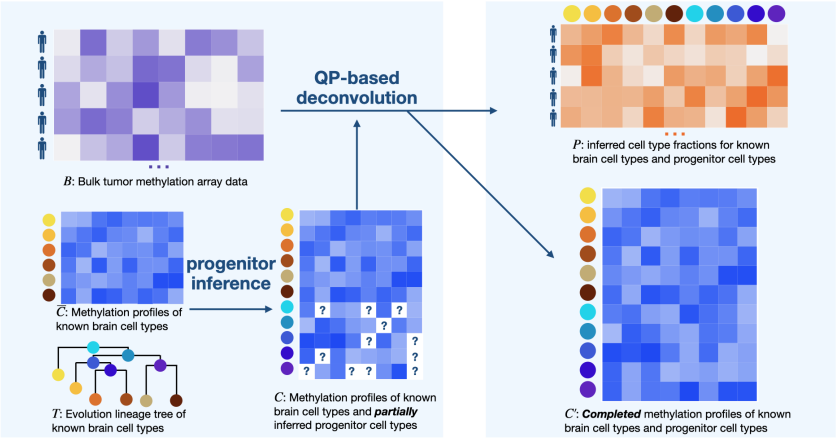


Figure 4.1: **The key computation problem solved by `Qombucha`** formulated as Quadratic Programming (QP). The input to the QP consists of: (i) a partially complete reference panel C_p which includes the complete methylation profiles for known developed brain cell types and partially complete profiles for progenitors, and (ii) the bulk methylation data matrix B. The QP asks to infer full reference panel C_f and the cell type composition matrix P with the objective of minimizing the sum of the squared differences between B and $P \times C_f$.

value for each feature of a mature cell type, and some of the features of each progenitor; it also specifies the range of values that an unknown feature of a progenitor can take.

4.3.2 `Qombucha`: Solving the QP Iteratively

A direct simultaneous inference of the cell composition matrix P and unknown entries in C_p in the quadratic program (4.2) requires substantial time and computing resources to find the optimal solution. To improve computational tractability, we introduce `Qombucha`, a block coordinate descent-like approach based on iteratively updating cell type composition and unknown entries in progenitor profiles, as summarized in Figure 4.2.

4.3.2.1 Iterative updating of P and unknown entries in C

Instead of trying to solve unknown entries in C_p and the cell composition matrix P simultaneously, `Qombucha` initializes the unknown entries guided by T to obtain C_0, and then uses

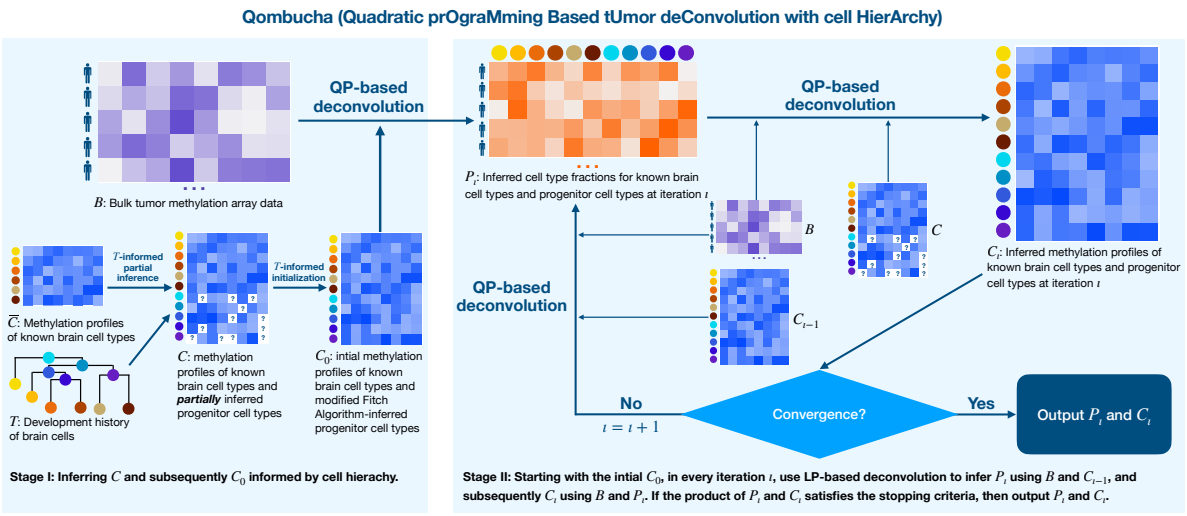


Figure 4.2: **Graphical abstract of `Qombucha`'s block coordinate descent-like approach.** `Qombucha`first builds the initial reference panel C_0 by imputing the unknown entries of C_p—i.e., the methylation features of the progenitor cell types for which the descendant mature cell methylation values do not agree—to minimize progenitor–descendant differences under the developmental tree T. Then, for iterations $\iota = 1, 2, \ldots$, `Qombucha`alternates QP subproblems: infer cell proportions P_ι from $C_{\iota-1}$, then update the unknown entries of C_p to obtain C_ι using P_ι. Upon meeting the stopping criterion, `Qombucha` outputs $P = P_\iota$ and $C_f = C_\iota$. Because P_ι is inferred from fully determined $C_{\iota-1}$ and C_ι is subsequently inferred from then fully determined P_ι, the QP steps are more efficient than the QP to infer P and C_f simultaneously.

C_0 as a starting point in an iterative procedure to alternate between updating P and the unknown entries in C_p until convergence. The construction of C_0 is described in Section 4.3.2.2. In each iteration ι, we first solve a restricted version of the QP described in Section 4.3.1.1 for P_ι with B *and* fully known $C_{\iota-1}$ as inputs:

$$\begin{aligned} P_\iota := \quad & \arg\min_P \|B - PC_{\iota-1}\|^2 \\ \text{subject to:} \quad & \forall i \in \{1, \ldots, n\} : \textstyle\sum_{k=1}^{r} p_{i,k} = 1 \\ & \forall i \in \{1, \ldots, n\}, \forall k \in \{1, \ldots, r\} : 0 \leq p_{i,k} \leq 1 \end{aligned} \tag{4.3}$$

Then, we solve another restricted version of the QP described in Section 4.3.1.1 for the unknown entries in C_p (whose indices form the set U) to obtain C_ι with B and P_ι (computed above) as

inputs:

$$\begin{aligned} C_\iota \coloneqq \ & \arg\min_{C_p} \|B - P_\iota C_p\|^2 \\ \text{subject to: } \forall (x,y) \in U : \min(C_p[Clade(x), y]) \ & \leq c_{x,y} \leq \max(C_p[Clade(x), y]), \end{aligned} \tag{4.4}$$

where $Clade(v)$ is the set of all leaf labels in the subtree of T rooted at vertex v and $C_p[Clade(v), j] = \{C_p[i, j] \mid \forall i \in Clade(v)\}$.

`Qombucha` outputs P_{ι^*} and C_{ι^*} provided it "converges" at some iteration ι^*. The algorithm is said to have converged when any of the following conditions is satisfied: (i) the bulk methylation matrix reconstruction error is "small" and the change in objective function value between successive iterations is "small", or (ii) the algorithm has run long enough. Specifically, in each iteration ι, `Qombucha` minimizes the squared error between the observed bulk methylation data matrix B and the reconstructed matrix $B_\iota = P_\iota C_\iota$. At the end of iteration ι, the normalized squared difference, Δ_ι, is calculated between B and B_ι as:

$$\Delta_\iota = \frac{\|B - B_\iota\|^2}{\|B\|^2} \tag{4.5}$$

`Qombucha` is said to have converged at iteration ι if Δ_ι falls below a user-defined δ and $\Lambda = \Delta_\iota - \Delta_{\iota-1}$ is smaller than a user-defined λ; this is based on the expectation that another iteration would be unlikely to reduce the error substantially. Additionally, `Qombucha`has a user-defined maximum iteration limit max_iter, which is set to bound the running time. The pseudocode for the iterative procedure is given in Algorithm 1.

Algorithm 1 Iterative optimization for P and unknown entries in C

$B \leftarrow$ bulk sample methylation data
$C \leftarrow$ partially complete reference panel
$C_0 \leftarrow$ initial complete reference panel, whose values are imputed using Algorithm 2
$\delta, \lambda \leftarrow$ user-defined parameters indicating convergence tolerance
$(\Delta, \Lambda) \leftarrow (\infty, \infty)$
$max\_iter \leftarrow$ maximum iterations
$\iota \leftarrow 1$, iteration index
while $(\Delta > \delta$ or $\Lambda > \lambda)$ and $(\iota \leq max\_iter)$ **do**
    $P_\iota \leftarrow$ Solution to QP (4.3) ▷ Step 1: Solve for P_ι with B and $C_{\iota-1}$ as inputs
    $C_\iota \leftarrow$ Solution to QP (4.4) ▷ Step 2: Solve for unknown entries in C with B and P_ι as inputs
    $\Delta \leftarrow \Delta_\iota$ ▷ Step 3: Compute variable values that are used to check for convergence
    $\Lambda \leftarrow \Delta_\iota - \Delta_{\iota-1}$
    $\iota \leftarrow \iota + 1$
end while
return $P_{\iota^*}, C_{\iota^*} \leftarrow$ values of P_ι and C_ι at convergence

4.3.2.2 Cell hierarchy-informed initialization of unknown entries in the progenitor profiles

The partially complete reference matrix C_p is obtained as described in Section 4.3.1.2. We utilize the brain cell development tree T to inform the initialization of "unknown entries" in C_p to obtain the initial complete reference matrix C_0, which serves as the starting point in the iterative procedure of `Qombucha`. The *unknown entries* in matrix C_p are those entries corresponding to the progenitor cells whose feature values are left as ranges at the end of the bottom-up phase of Algorithm 2. If, for any methylation feature y and progenitor cell type x, the methylation range of each unknown entry in C_p is given by $[m_{low}[x, y], m_{high}[x, y]]$, then the optimal initial methylation values are simultaneously determined by minimizing the sum of squared methylation

changes between parent-child pairs across the entire tree, using the following quadratic program:

$$\begin{aligned} M \coloneqq \quad & \arg\min_{\forall (x,y)\in U} \sum_{(x,v)\in E} (q_{x,y} - q_{v,y})^2 \\ \text{subject to:} \quad & m_{low}[x,y] \le q_{x,y} \le m_{high}[x,y], \qquad \forall (x,y) \in U \end{aligned} \tag{4.6}$$

where $q_{x,y}$ represents the methylation level at the parent node for feature y and $q_{v,y}$ represents the methylation level at the child node for the same feature. The complete pseudocode for building C_p and obtaining C_0, is given in Algorithm 2.

Algorithm 2 Initializing unknown entries in C for a given methylation feature s

$T(V,E) \leftarrow$ The binary tree representing brain developmental history with vertex set V and directed edge set E
The indexing of feature s is suppressed in the rest of the description to keep it from becoming too verbose
$M[v] \leftarrow$ The final methylation value of vertex v for feature s. It is initially known only for leaf vertices
$\forall v \in V, \quad m_{low}[v] \leftarrow 0$ ▷ Represents the lower bound of methylation value that can be assigned to vertex v for feature s
$\forall v \in V, \quad m_{high}[v] \leftarrow 0$ ▷ Represents the upper bound of methylation value that can be assigned to vertex v for feature s
$\delta \leftarrow$ A small user-defined threshold value
function MODIFIEDFITCH(T, δ, M)
    for all v is a leaf of T **do**
        $(m_{low}[v], m_{high}[v]) \leftarrow (M[v], M[v])$
    end for
    for each internal vertex u in postorder traversal of T **do** ▷ Bottom-Up Phase (Postorder Traversal)
        $v_0, v_1 \leftarrow$ children of u
        $\left(intersect_{low}, intersect_{high}\right) \leftarrow \left(\max\left(m_{low}[v_0], m_{low}[v_1]\right), \min\left(m_{high}[v_0], m_{high}[v_1]\right)\right)$
        $\left(union_{low}, union_{high}\right) \leftarrow \left(\min\left(m_{low}[v_0], m_{low}[v_1]\right), \max\left(m_{high}[v_0], m_{high}[v_1]\right)\right)$
        if $intersect_{low} \le intersect_{high}$ **then** ▷ Choose the intersection of the ranges of children's m-values, if it is valid.
            $(m_{low}[u], m_{high}[u]) \leftarrow (intersect_{low}, intersect_{high})$;
        else ▷ Otherwise, choose the maximal range that includes both the children's m-value ranges.
            $(m_{low}[u], m_{high}[u]) \leftarrow (union_{low}, union_{high})$;
        end if
        if $m_{high}[u] - m_{low}[u] \le \delta$ at all methylation sites for vertex u **then** ▷ Optional step to speed up computation
            $(m_{low}[u], m_{high}[u]) \leftarrow \left(\frac{m_{low}[u]+m_{high}[u]}{2}, \frac{m_{low}[u]+m_{high}[u]}{2}\right)$
        end if
    end for
    $M \leftarrow$ Solution to QP (4.6) ▷ Solve a quadratic program to simultaneously assign optimal initial methylation values to the unknown entries in C
    Return the final methylation values $M[v]$, for all vertices $v \in V$, from the optimal solution to the quadratic program.
end function

4.3.3 Evaluating tightness of the known GBM subtype clusters

To evaluate how well different types of information (i.e., `Qombucha`-imputed cell fractions, the methylation values of [121]-reported 32k sites etc.) recapitulate the known GBM subtype clusters, we use the McClain–Rao score [164], i.e., the ratio between the average in-class distance and the average out-of-class distance (D_{in}/D_{out}) as well as the silhouette score [165] to assess the tightness of clusters.

4.3.3.1 McClain–Rao score (D_{in}/D_{out})

This score corresponds to the ratio between the mean in-class distance and mean out-of-class distance for each patient to evaluate the tightness of GBM subtype clusters. Let v_i be the feature vector representing patient sample i, and assume that sample i belongs to class θ. Let α denote the number of samples in class θ, and β denote the number of samples outside of class θ. The mean L^1 distance between sample i and other samples within the same class θ is defined as:

$$D_{\text{in}}(i) = \frac{1}{\alpha - 1} \sum_{a \in C_\theta, a \neq i} |v_i - v_a| \tag{4.7}$$

where C_θ represents the set of samples belonging to class θ, and a indexes over all samples in C_θ excluding sample i. The mean L^1 distance between sample i and samples outside of class θ is defined as:

$$D_{\text{out}}(i) = \frac{1}{\beta} \sum_{b \notin C_\theta} |v_i - v_b| \tag{4.8}$$

where b indexes over all samples outside class θ. The clustering goodness for sample i is then defined as the ratio of the in-class distance to the out-of-class distance:

$$R_i = \frac{D_{\text{in}}(i)}{D_{\text{out}}(i)} \tag{4.9}$$

R_i is the ratio that indicates the tightness of the clusters: a value close to 0 indicates tighter clustering (i.e., the mean in-class distance is much smaller than the mean out-of-class distance), while a value close to 1 indicates poor clustering and a value larger than 1 indicates that sample i has a closer affinity with samples outside of its assigned class θ than the samples within class θ.

4.3.3.2 The silhouette score

A silhouette value $s(i)$, as defined by [165], can be computed for each item i in any given cluster as follows: Let $a(i)$ denote the average dissimilarity of the item to all other items in its own cluster C_I, and $b(i)$ denote the average dissimilarity of item i to all items in its closest neighboring cluster C_J. That is,

$$\begin{gathered} a(i) = \frac{1}{|C_I| - 1} \sum_{j \in C_I, i \neq j} d(i,j); \qquad b(i) = \min_{J \neq I} \frac{1}{|C_J|} \sum_{j \in C_J} d(i,j) \\ \text{Then, } s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \end{gathered} \tag{4.10}$$

Here, $d(i,j)$ could be any distance metric. We use L^1 norm in our evaluation (our observations hold true for L^2 norm as well). By definition, $s(i)$ lies in $[-1, 1]$; the closer the value is to 1, the better clustered item i is. In contrast, if $s(i)$ is closer to -1, it would be more appropriate for item i to be taken out from its current cluster and assigned its closest neighboring cluster. We

compute $s(i)$ value for each patient i using the corresponding row vectors from P_{ι^*} as features.

4.3.4 Input data preparation

4.3.4.1 Reconstructing brain development history using single-cell methylation data

We randomly select 30 cells per each non-neuronal cell type from the Tian et al. (2023) [147] data set. The sampling ensures that each cell type has equal representation in the cohort to avoid biasing the tree. As explained above, the value of a feature in a sample is the average methylation value for sites used in that feature group for each feature group reported in [147]. The pairwise distances of the cells are calculated based on the values of the 800 feature groups. The tree representing the branched process underlying brain development history is then constructed using neighbor-joining [163] based on the pairwise distances. The resulting tree structure is shown in Figure 4.3. As can be observed from the tree, the cells of the same type cluster together with only a few exceptions. We then infer the brain developmental history by contracting the clades of the same cell types. Additionally, we removed the PC cells from the brain development history as this cell type is not clearly defined in either [147] or in the Cell Ontology [166, 167, 168]. The resulting brain development history is depicted in Figure 4.5b.

4.3.4.2 Processing bulk sample methylation data

For the GBM samples, we started with bulk methylation data for 1,160 samples sequenced using the Infinium HumanMethylation450 BeadChip (450k array), which were collected and curated from various sources, including the TCGA database and clinical samples. The samples

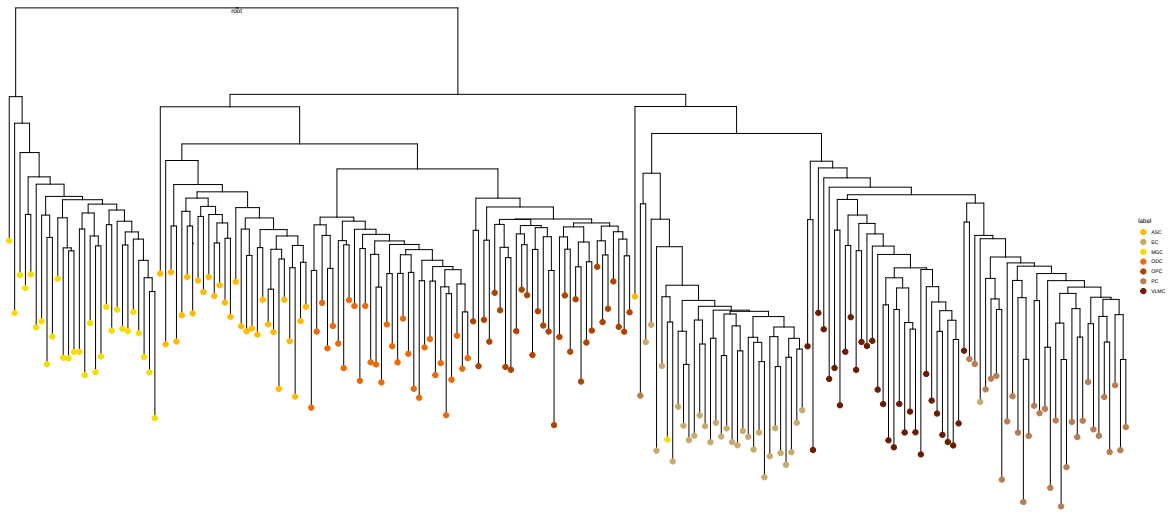


Figure 4.3: The neighbor-joining tree of mature non-neuronal cells. As described in Section 4.3.2.2, the tree is constructed using the methylation profiles of cells randomly sampled from the Tian et al. (2023) data set.

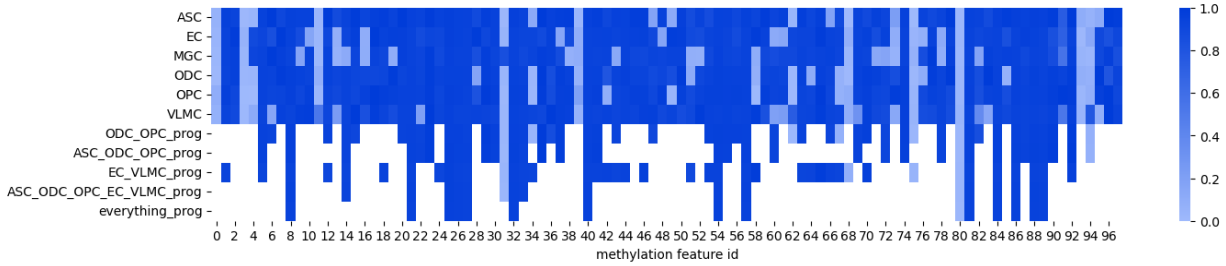
were classified as RTK I, RTK II, or MES TYP subtypes using the [121] brain and CNS tumor classifier. To eliminate potential biases stemming from tumor impurity and subtype classification errors, we removed samples with lower than 0.5 tumor purity and subtype assignment scores lower than 1, resulting in 896 patient samples passing quality control. Among the 896 samples, there are 216 RTK I samples, 431 RTK II samples, and 249 MES TYP samples. The 450k methylation array data cover 485,512 sites, among which 2,484 sites overlap with the 156k CpG sites reported by [147]. The 2,484 sites were then grouped according to the features reported in [147], resulting in 98 brain development-associated features with at least one site among the original 800 features. The other 702 features reported in [147] are excluded, as none of the 485k methylation array sites overlap with the brain cell type-differentiating CpG sites in these feature groups. We similarly processed the 961 IDH-mut glioma 450k array samples.

4.3.4.3 Obtaining the partially complete reference panel C_p and initial reference panel C_0

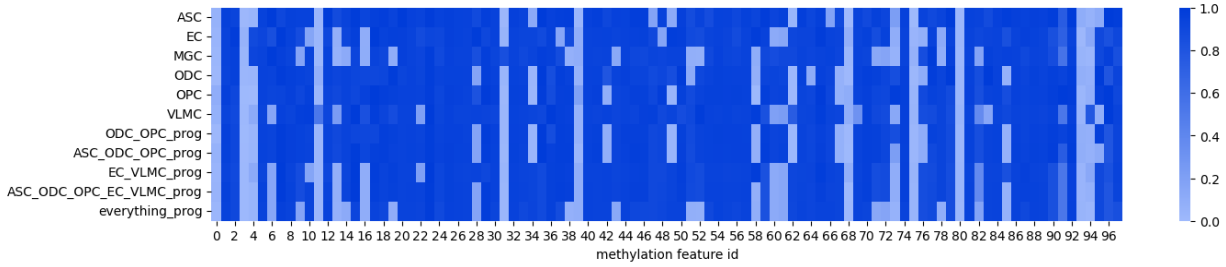
We combine the single-cell methylation profiles of 30 cells randomly sampled from each non-neuronal cell type (ASC, ODC, OPC, MGC, EC, and VLMC) and compute the average methylation value for sites in each of the 98 methylation feature groups selected by [147]. This results in a reference panel of size 6×98 for mature non-neuronal cells. Then, using a threshold of $\delta = 0.01$ and cell hierarchy-informed inference of progenitor cell profiles as described in Section 4.3.1.2, we obtained the partially complete profile of non-neuronal progenitors. Stacking together the profiles of mature non-neuronal cells and the partially inferred progenitors, we obtained the partially complete reference panel C_p of size 11×98 (see Figure 4.4a). C_p contains 327 unknown entries. To obtain the initial reference panel C_0, which serves as the starting point in the iterative procedure of `Qombucha`, we initialized the 327 unknown entries in C using the method described in Section 4.3.2.2 guided by the brain development history T constructed as described in Section 4.3.4.1 (see Figure 4.4b).

4.3.4.4 Expanding C_p and C_0 with neuronal cell types

While GBM is a cancer of non-neuronal cells, GBM samples may include impurities consisting of neuronal cells, as non-neuronal cells like astrocytes (ASC) and oligodendrocytes (ODC) are in close proximity of neuronal cells [169]. Thus we also add to C_p and C_0 the partially complete excitatory and inhibitory neuronal cell consensus methylation profiles. In the [147] data set, 15 cell types are in the excitatory neuronal cell category, and 18 in the inhibitory neuronal cell



(a) Partially complete reference panel C obtained as described in Section 4.3.4.3.



(b) Initial reference panel C_0 with values imputed using Algorithm 2 obtained as described in Section 4.3.4.3.

Figure 4.4: Reference panels.

category. Similar to the processing of non-neuronal cells, we randomly sampled and combined the single-cell methylation profiles of 30 cells of each neuronal cell types. Then, respectively for the excitatory and inhibitory neuronal cell category, we computed the partially complete consensus profiles by averaging values of methylation features consistent across cell types within the same category (standard deviation ≤ 0.01), and leave inconsistent methylation features (19 in excitatory category and 52 in inhibitory category) empty. As a result, after including the excitatory and inhibitory neuronal cell consensus profiles, C_p has size 13×98. The empty entries of the neuronal consensus profiles are then initialized by randomly drawing values in the range $[0, 1]$ and the resulting profile is included in C_0, making it also of size 13×98.

4.3.5 Experiments for robustness and benchmarking

4.3.5.1 Simulations

We generated P_{sim} by randomly drawing 100 samples, each with 13 dimensions, the same number of cell types included in this manuscript, from a Dirichlet distribution. The randomly generated P_{sim} is then multiplied with C_0, which was generated as described in Sections 4.3.4.3 and 4.3.4.4, followed by introducing Gaussian noise ($\mu = 0, \sigma = 0.01$) to obtain B_{sim}. The entries in B_{sim} are corrected to 0 if < 0 and 1 if > 1.

4.3.5.2 Alternative initialization of C_0

To test the usefulness of the algorithm-guided initialization of matrix C, i.e., the matrix C_0 obtained through Algorithm 2, we compare the quality of solutions to `Qombucha` instances against two other modes of initialization: (i) random initialization - the unknown entries of the matrix C_0 are filled by drawing real numbers uniformly within range $[0, 1]$; (ii) "mean of descendants" initialization - the unknown entries of C_0 are filled by setting the methylation values of each internal node in the tree T to be the mean of its direct descendants' values.

4.3.5.3 Subsampling patient samples

From the bulk methylation matrix of 896 GBM patients we processed - as described in Section 4.3.4.2, we generated 10 patient subsets, each by randomly subsampling 100 patient samples from each of the three subtypes (RTK I, RTK II, and MES TYP). Each subset of 300 patient samples are then used as input to `Qombucha` to test its robustness.

4.3.5.4 Benchmarking quality of clustering

We benchmark the quality of clustering of `Qombucha`-inferred cell fractions by computing McCain-Rao scores (D_{in}/D_{out}) and silhouette scores (described in Section 4.3.3) using the following information: (i) the methylation values of 2484 sites randomly sampled from the 450k methylation array, (ii) the methylation values of [121] 32k sites, (iii) the methylation values of 98 features containing cell type-differentiating sites included in the 450k array (see Section 4.3.4 for detailed explanation), (iv) QP-inferred non-neuronal mature brain cell fractions, excluding progenitor profiles, (v) MethylCIBERSORT[136]-inferred non-neuronal mature brain cell fractions, excluding progenitor profiles, and (vi) `Qombucha`-inferred non-neuronal mature and progenitor cell fractions with alternative initialization of C_0 as described in Section 4.3.5.2.

4.3.6 Stable inheritance of methylation features

We declare a methylation feature to be *stably inherited* in a given path along the brain development tree, if the overall standard deviation of the methylation values on the path is smaller than or equal to 0.01, and the methylation value is strictly monotonic along the path.

4.4 Results

4.4.1 Our Main Findings

We applied `Qombucha` to bulk methylation data from 896 GBM patients spanning multiple subtypes and an additional 961 IDH-MUT patients and showed that `Qombucha` successfully recapitulated the established biological insight that GBM MES TYP patients exhibit higher im-

mune cell infiltration [120]. Beyond recovering known biology, `Qombucha` identified GBM subtype-specific brain developmental programs and demonstrated that these programs are significantly associated with patient outcomes. Importantly, the `Qombucha`-imputed mature and progenitor cell fractions provide a set of low-dimensional and interpretable features yielding subtype clusters that are "as tight" or "tighter—than" those clusters derived from high-variance methylation features commonly used in clinical classifiers [121]. Our work offers a framework for understanding subtype-specific GBM progression at a granular level and provides insights that may help refine future precision medicine for GBM.

4.4.2 Methylation Sites Associated with Brain Cell Development and Brain/CNS Cancer Subtyping

Our `Qombucha`-based analysis for GBM overlays a methylation-based tree structure for normal development and that for abnormal tumor progression. Therefore, we quantitatively evaluated whether the CpG sites that are associated with brain cell development/differentiation are distinct from those that are associated with specific brain/CNS cancer subtypes.

We first show that the genomic regions of the brain development/differentiation associated methylation sites identified by [147] and those of the brain/CNS cancer associated methylation sites identified by [121] are distinct: the development-associated sites are less concentrated in CpG islands and shores and more concentrated in inter-CGI than the brain-cancer subtype associated sites (Figure 4.5a).

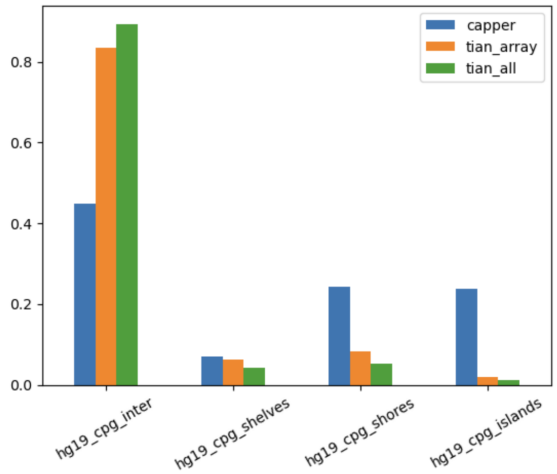
Next, we demonstrate that the brain development/differentiation associated methylation sites identified by [147] are statistically distinct from the brain/CNS cancer subtype associated

sites used by [121]. Of the 485,512 sites in the 450k methylation array, 32,000 have been reported by [121] to be associated with brain/CNS cancers. Among the 156,000 sites identified by [147] as mature brain cell type-differentiating, 2484 overlap with the 450k methylation array. Among these 2484 sites, only 108 overlap with Capper et al.'s 32,000 sites; in constrast, a random selection of 2484 sites from the 450k array would have an expected overlap of $32000 \times 2484/485512 = 163.72$ with the Capper et al.s sites. The p-value of observing an overlap of ≤ 108 sites between a set of 2484 sites and another set of $32,000$ sites is $\Pr(X \leq 108) = \sum_{k=0}^{108} \binom{n}{k} p^k (1-p)^{n-k} = 1.136 \times 10^{-6}$, where X is the binomial random variable representing the number of overlaps, $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ is the binomial coefficient for k successes in n trials, and $p = 32000/485512 \approx 0.06588$ is the probability of a randomly selected 450k methylation array site overlaps with [121] 32,000 sites.

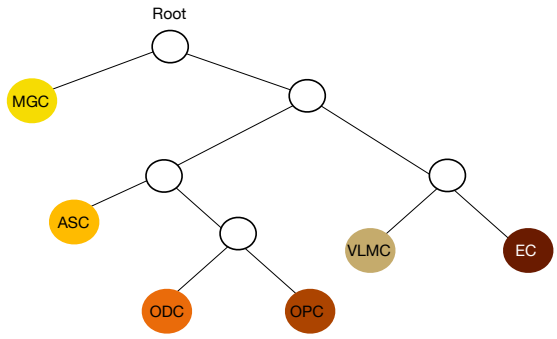
The above observations suggest that the methylation signals for brain cell differentiation are distinct from those for brain/CNS tumor progression.

4.4.3 Biological and Clinical Insights

In this section, we demonstrate our key biological and clinical findings derived from `Qombucha`-imputed cell fractions and progenitor methylation profiles. Our methodological contribution is to utilize the prior knowledge of brain cell developmental history to guide the inference of progenitor cell methylation profiles, which, to the best of our knowledge, has not been employed by previous attempts at methylation deconvolution. Our work addresses a different question compared to prior deconvolution efforts. We aim to identify whether distinct GBM subtypes are associated with distinct mature or progenitor cell types, as well as distinct developmental trajectories, i.e.



(a) Comparison of the genomic regions of brain cell type-differentiating sites and brain cancer subtype-differentiating sites. Four categories of genomic regions are present in the bar plot: CpG islands (CGI), CpG shores (2kbp spanning CGI), CpG shelves (2kbp spanning the shores), and inter-CGI regions. The blue bars indicate the fractions of the 32k CpG sites differentiating brain cancer subtypes identified in [121] in each genomic region category. The green bars indicate the 156k CpG sites differentiating normal brain cell types reported in [147], and orange bars indicate a subset of the 156k CpG sites overlapping with the Illumina 450k methylation array.



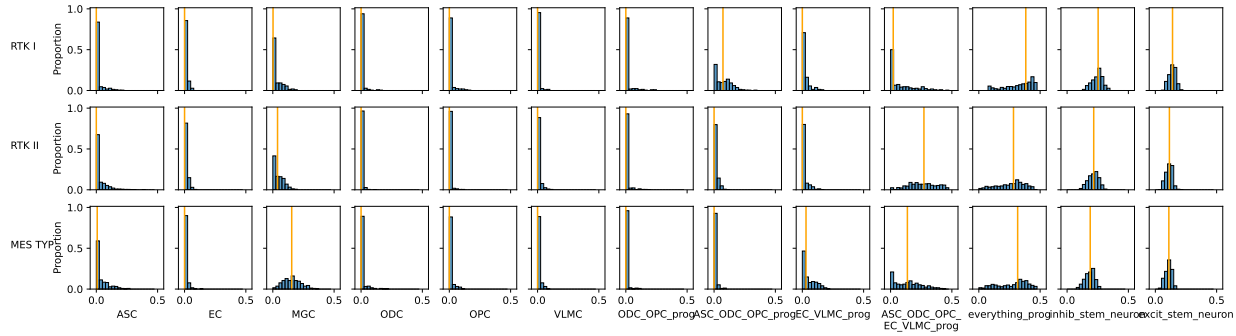
(b) Brain development history represented as a tree. The methylation sites associated with brain development - distinct from those associated with brain cancer subtypes, suggest a tree structure for representing the brain development history, as described in Section 4.3.4.1. The colored leaf nodes represent the developed brain cell types whose methylation profiles are fully known. The internal nodes (with no labels) represent the progenitor cells whose methylation profiles are partially derived by `Qombucha` from their descendant brain cell types.

Figure 4.5: Methylation sites associated with brain cell development/differentiation vs. brain/CNS cancer subtypes.

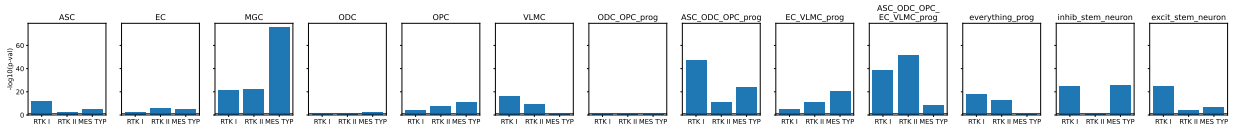
specific paths in the tree representing the developmental program of brain cells.

4.4.3.1 `Qombucha` recapitulates known GBM subtype characteristics

We first obtained the brain development history (Section 4.3.4.1) and subsequently reference panels C_p and C_0 (Section 4.3.4.3). Using C_0, C_p, and B, the bulk methylation array data of 896 GBM patient samples (processed as described in Section 4.3.4.2) as input, we employed `Qombucha` to obtain deconvolved cell type compositions per sample. As can be observed in the



(a) Cell type compositions inferred by `Qombucha` in each GBM subtype. Each row represents a specific GBM subtype: RTK I, RTK II, or MES TYP, and each column represents a specific cell type. The orange vertical line in each plot indicates the median inferred fraction (across all samples for the associated GBM subtype represented by that row) of the cell type represented by that column.



(b) Bar plots showing $-\log_{10}$ p-values from KS tests (comparing each GBM subtype against the combination of the remaining two subtypes) on cell fractions inferred by `Qombucha`. In each plot, the red horizontal line represents a p-value threshold of 0.05. Taller bars indicate more significant KS test results. With the exception of MGC (which are known to be enriched in GBM MES subtype [120]) the GBM subtype differentiating cell types are mostly progenitors.

Figure 4.6: GBM subtypes are characterized by distinct cell type compositions.

third column of Figure 4.6a, compared to RTK I and RTK II, the patient samples in the MES TYP subtype demonstrate a higher presentation of microglia, which are the resident immune cells in the brain [170]. This result recapitulates the known association with higher immune activity and inflammation in the tumor microenvironment and increased levels of immune cell infiltration in the MES TYP subtype [120], showing that `Qombucha` is able to confirm prior biological insights regarding GBM subtypes.

4.4.3.2 `Qombucha` reveals that brain developmental programs are GBM subtype-specific

We show in Figure 4.6a that different brain developmental programs, represented by cell type fractions, appear to be associated with different GBM subtypes. To assess whether the differences in imputed cell type fractions across GBM subtypes are significant, we performed two-sided Kolmogorov-Smirnov (KS) tests [171, 172] to assess the statistical significance of these variations. Specifically, we compared the fractions imputed for patients of one GBM subtype against the imputed fractions of patients of the other two subtypes. The p-values obtained from these tests in Figure 4.6b show that MGC is the most significantly different cell type across all GBM subtypes and distinguishes the MES TYP subtype from other subtypes, reinforcing our observations described in Section 4.4.3.1.

We also compared the estimated proportion of the everything progenitor as a proxy for stemness. By this measure, RTK I has a significantly higher stemness than either MES TYP ($p <$ 9e-11, Wilcoxon test) and RTK II ($p <$ 9e-21); MES TYP has more stemness than RTK II ($p <$ 0.013). This order of stemness among the three subtypes is consistent with a recent methylation-based analysis of [173]. An earlier expression-based analysis of stemness by [174] ranked the subtypes as RTK I $>$ RTK II $>$ MES TYP in decreasing order of stemness.

4.4.3.3 Qombucha-imputed cell type fractions improve GBM subtype clustering compared to state-of-the-art methods

We evaluated the tightness of known GBM subtype clusters using both the D_{in}/D_{out} score and the silhouette score, as described in Section 4.3.3. Briefly, both the D_{in}/D_{out} score and the silhouette score $s(i)$ of sample i denote how well the sample is clustered. Smaller D_{in}/D_{out} scores indicate better clustering, whereas larger silhouette scores indicate better clustering. We used six distinct types of information to calculate (L^1) distances between all pairs of patients: (i) the methylation values of 2484 sites randomly sampled from the 450k methylation array, (ii) the methylation values of the 32k sites used in [121], (iii) the methylation values of 98 features containing cell type-differentiating sites included in the 450k array (see Section 4.3.4), (iv) QP-inferred mature brain cell fractions, excluding progenitor profiles, (v) MethylCIBERSORT[136]-inferred mature brain cell fractions, excluding progenitor profiles, and (vi) Qombucha-inferred mature and progenitor cell fractions with a number of distinct starting points. The (L^1) distances are then used to calculate the D_{in}/D_{out} score and the silhouette score $s(i)$ for each sample i for assessing how close the sample is to the other samples are in the same GBM subtype, against those samples in the other two subtypes. Evaluated by both scores, Qombucha-inferred mature and progenitor cell fractions for all GBM samples outperform all other types of values in tightness of the known GBM subtype clusters (Figure 4.7, row VIII, column "All").

These results show that Qombucha-imputed cell fractions, despite being only 13-dimensional vectors, achieve better clustering of GBM subtypes than higher-dimensional methods, such as the [121]-selected 32k sites (experiment VIII vs. II). Moreover, the Qombucha-imputed cell fractions outperform GBM subtype clustering based on the original bulk methylation data B (ex-

periment VIII vs. III). Strictly speaking, `Qombucha` is not performing dimensionality reduction since the CpG sites it considers are associated with the developmental program and have minimal overlap with the CpG sites used by [121] for brain tumor classification. We show that it is essential to include progenitor cell profiles in the reference panel C, as clustering performance greatly improves when progenitor profiles are included (experiment VIII vs. IV).

Furthermore, we identified a subset of cell types that best distinguish GBM subtypes. We enumerated all possible quadruplets of cell types and calculated the D_{in}/D_{out} using the `Qombucha`-imputed fractions of cell types in each quadruplet. The quadruplets best distinguishing each GBM subtype (i.e., quadruplets with the lowest D_{in}/D_{out} scores for each subtype) are the following: RTK I is best distinguished by MGC, ODC, VLMC, and ASC-ODC-OPC-EC-VLMC progenitor; RTK II by ODC, OPC, ODC-OPC progenitor, and ASC-ODC-OPC progenitor; MES TYP by EC, VLMC, ODC-OPC progenitor, and ASC-ODC-OPC progenitor. Combining the quadruplets that best distinguished each GBM subtypes results in 8 cell types: EC, MGC, ODC, OPC, VLMC, ODC-OPC progenitor, ASC-ODC-OPC progenitor, as well as ASC-ODC-OPC-EC-VLMC progenitor. We calculated the D_{in}/D_{out} using these 8 cell types and demonstrate that the `Qombucha`-inferred fractions of 8 cell types have the best performance in improving the clustering tightness of the GBM subtypes (Figure 4.7, row IX).

4.4.3.4 `Qombucha` recapitulates known GBM IDH-WT and IDH-mut glioma differences and cell-of-origins of IDH-mut glioma

One of the most important advances in GBM genomics in the 21st century was the recognition that some tumors have recurrent somatic mutations in either of the paralogous genes *IDH1*,

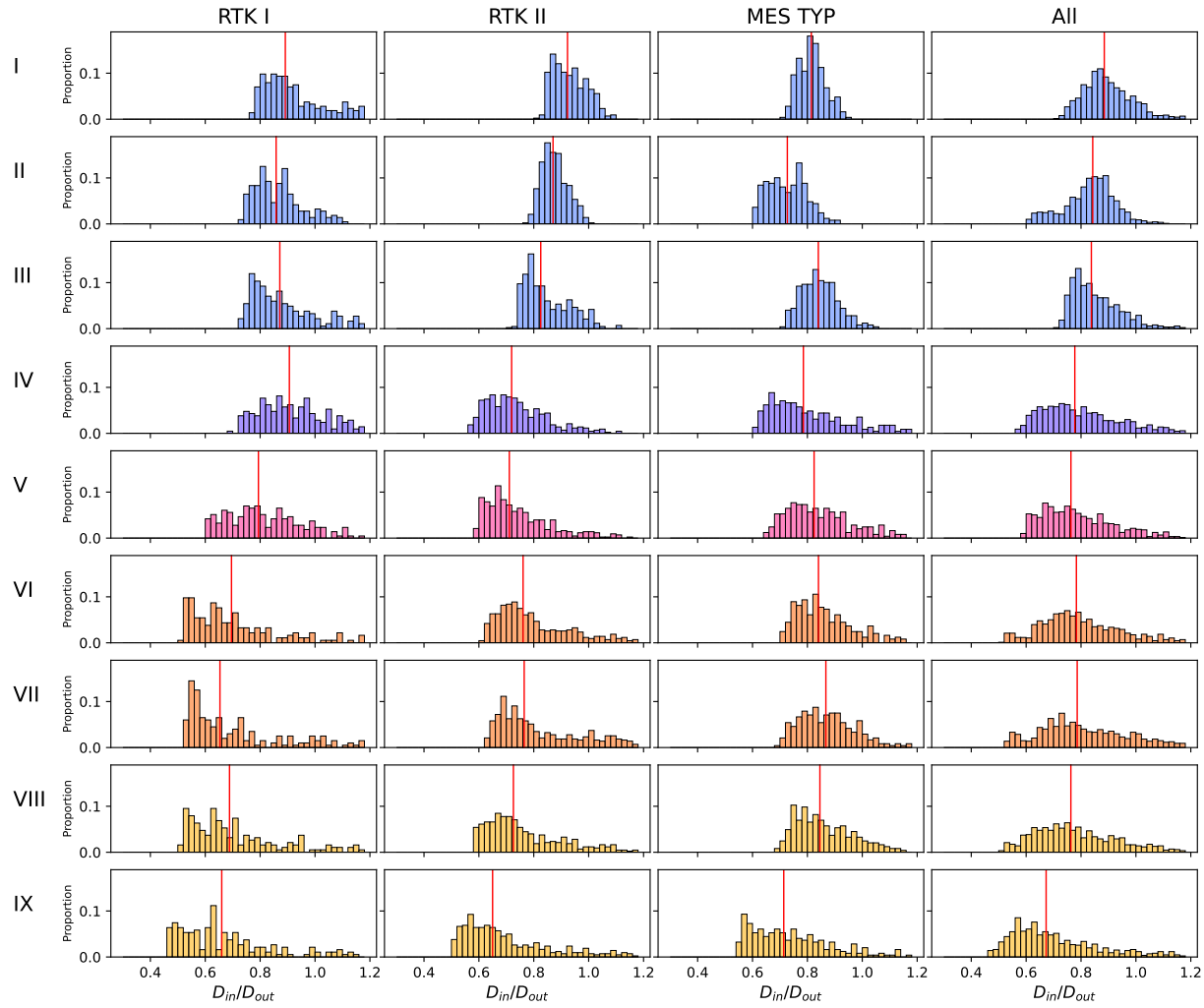


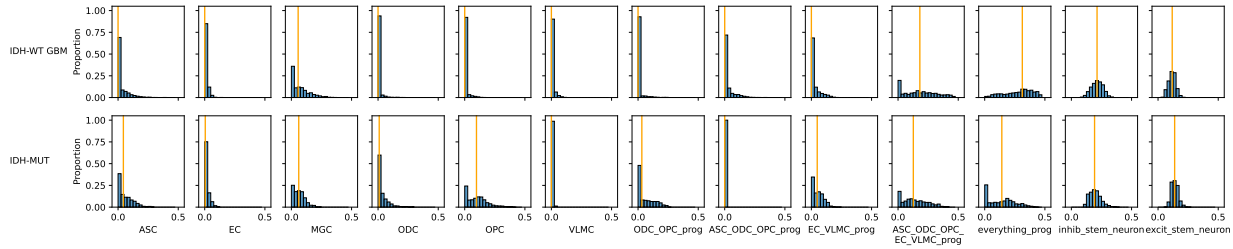
Figure 4.7: **Tightness of GBM subtype clusters computed using D_{in}/D_{out}.** Here D_{in}/D_{out} is the ratio between the mean L_1 distance among sample pairs within a specific GBM subtype and that between the samples of this GBM subtype and the remaining samples. Each row uses distinct information to compute the distances for assessing the tightness of each GBM subtype cluster (a smaller D_{in}/D_{out} indicates a better clustering): (I) the methylation values of 2484 sites randomly sampled from 450k methylation array; (II) the methylation values of Capper et al. 32k sites; (III) the methylation values of 98 features; (IV) QP-inferred cell fractions, excluding progenitors; (V) MethylCIBERSORT-inferred cell fractions; (VI) `Qombucha`-inferred cell fractions with progenitor profiles initialized as the mean of direct descendants; (VII) `Qombucha`-inferred cell fractions with random initialization; (VIII) `Qombucha`-inferred cell fractions with development tree-informed starting point; (IX) `Qombucha`-inferred cell fractions with development tree-informed starting point, score calculated from the union of cell type quadraplets best distinguishing each subtype. The scores of IV-VIII are calculated of all 11 cell types including mature non-neuronal cell type and their progenitors as listed in the brain development tree, excluding the impurities consisting of neuronal cells. The red vertical lines indicate the median for each subtype in each experiment.

IDH2 and the patients with IDH mutations have a better prognosis [175, 113]. We applied `Qombucha` separately on IDH-WT GBM and IDH-MUT patient samples. Because IDH-MUT cancers at any site are known to have disruptions in methylation [176, 177] and there are also

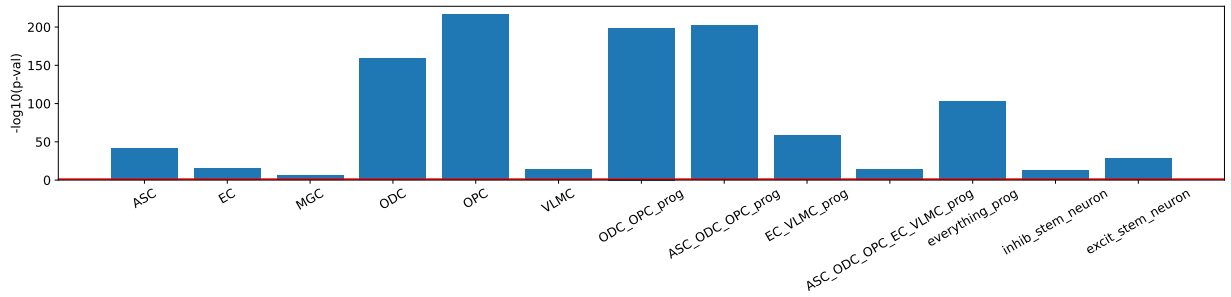
brain cancer-specific IDH-related methylation changes [178], we choose to employ the completed reference panel C_f obtained from the deconvolution of all 896 IDH-WT GBM samples, as described in Section 4.4.3.1, to perform one-step QP-based simultaneous deconvolution of 896 IDH-WT GBM samples and 959 IDH-MUT samples. As can be observed from Figure 4.8a and 4.8b, IDH-WT GBM and IDH-MUT have different cell compositions and different favored developmental programs. Interestingly, IDH-WT GBM have higher overall presentation of the ASC-ODC-OPC-EC-VLMC progenitor and the "everything progenitor", which are the two progenitors at the highest levels in the brain developmental tree. This suggests that IDH-WT GBM have greater stem-like characteristics than IDH-MUT, which potentially explains that IDH-WT GBM tends to have worse prognosis than IDH-MUT. Additionally, the D_{in}/D_{ou} scores presented in Figure 4.8c indicate that `Qombucha`-imputed cell fractions are able to distinguish IDH-WT GBM and IDH-MUT patients. Notably, IDH-MUT gliomas are known to originate from OPC [179], and as can be observed in Figure 4.8a, our results mirror this prior finding.

4.4.3.5 `Qombucha`-inferred progenitor methylation values are stably inherited

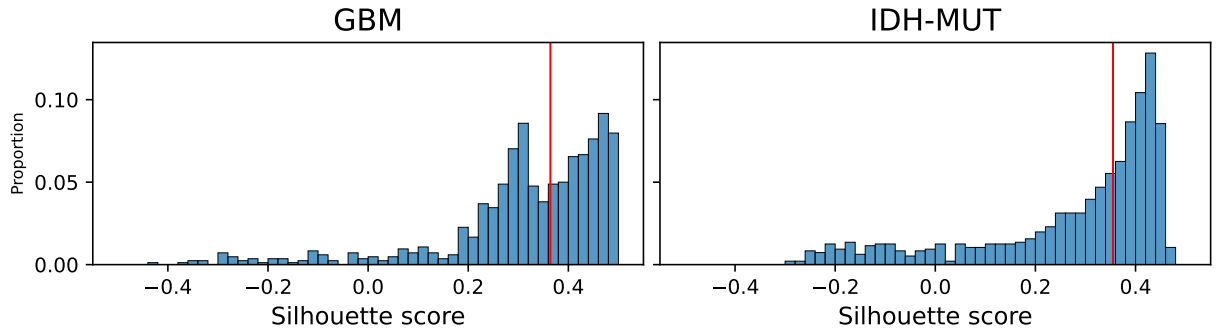
From the complete reference panel C_ι output by `Qombucha`, we identified 39 features that are stably inherited (Section 4.3.6) on at least one of the five paths from the root of the brain development tree to a leaf with two or more edges. We performed the following simulation to show that this number is higher than expected: we generated 1000 instances of randomly assigning methylation values (ranging between 0 and 1) to the unknown entries in the partially complete reference panel C, then evaluated how many methylation sites are stably inherited on at least one of the five paths. We repeated this process 5 times, and the number of instances



(a) Cell type compositions inferred by `Qombucha` in IDH-WT GBM and IDH-MUT. Orange vertical lines indicate the median inferred fraction for each cell type.Among the developed cell types, the primary difference between the IDH-WT GBM and IDH-MUT samples is in the fraction of OPC cells they habor, as reported earlier [179].



(b) Bar plot showing $-\log_{10}$ p-values from KS tests on cell fractions inferred by `Qombucha` for IDH-WT GBM and IDH-MUT patients. The red horizontal line represents a p-value threshold of 0.05. Taller bars indicate more significant KS test results.



(c) Tightness of IDH-WT GBM and IDH-MUT clusters evaluated using the Silhouette Score. The scores are calculated using the `Qombucha`-imputed cell fractions. The left panel shows the scores of the IDH-WI GBM patients, and the right shows that of the IDH-MUT patients.

Figure 4.8: IDH-WT GBM and IDH-MUT gliomas have distinct cell type compositions.

with 39 or more stably inherited sites on at least one path is the following: 42, 38, 36, 41, 44. See Figure 4.9a for one of the five experiments of the simulation. These results show that identifying 39 features stably inherited on at least one of the five paths is higher than expected, with an average empirical p-value of 0.0402. This demonstrates that on the brain development

tree derived from the normal mature cell type methylation profiles by [147], the progenitor cell types whose methylation profiles were identified on our GBM cohort by `Qombucha`, appear to stably inherit and transmit a high fraction (39 of the 98) of GBM-associated methylation features. Thus, from a methylation point of view, the progression of the GBM cells are concordant with brain cell development trajectories.

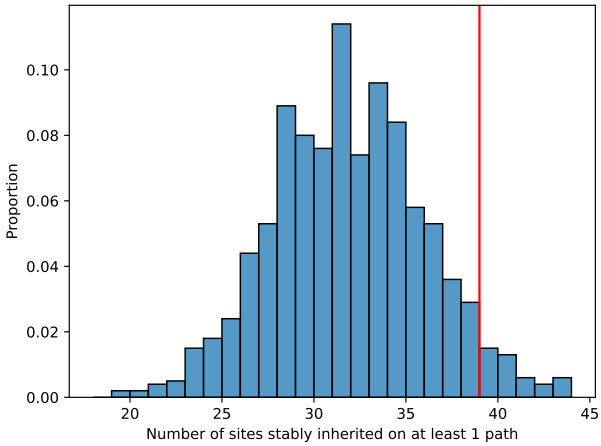
We then split the `Qombucha`-inferred reference panel C into two, one with 39 stably inherited features, the other with 59 not stably inherited features. We then inferred the cell fractions with one-step deconvolution using the two panels, respectively. Interestingly, the cell fractions inferred using only the stably inherited features demonstrate clustering tightness of the GBM subtypes comparable to when using all 98 features, whereas the cell fractions inferred using features not stably inherited show worse clustering of the subtypes, as can be seen in Figure 4.9b. These results demonstrate that stably inherited methylation changes in brain development are dominating the distinctions among different GBM subtypes.

4.4.4 Efficiency and robustness of `Qombucha`

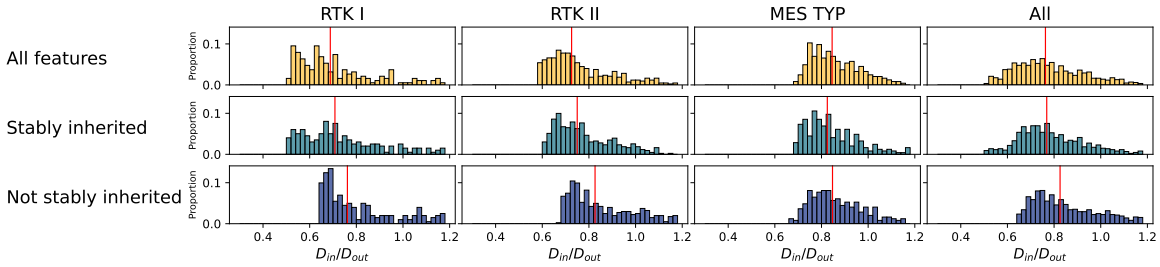
We performed the following analyses to evaluate and validate the robustness and efficiency of `Qombucha`.

4.4.4.1 `Qombucha` recovers simulated ground truth

We generated 5 random instances of the cell proportion matrix P_{sim} and added a small Gaussiance noise to it, as described in Section 4.3.5.1. We then multiplied the randomly generated P_{sim} with C_0, the tree-informed intialization of reference panel, to obtain the simulated



(a) Random assignment of methylation value to evaluate the number of stably inherited sites obtained by chance. We generated 1000 instances of assigning random methylation values (between 0 and 1) to the unknown entries in the partially complete reference panel C_p, then evaluated how many methylation sites are stably inherited on at least one of the five paths. The simulation was repeated 5 times, and one instance of the simulation is demonstrated in the distribution plot. The red vertical line marks 39, the number of stably inherited sites obtained from real data.



(b) Tightness of GBM subtype clusters computed using D_{in}/D_{out}. Every row represents the tightness of known GBM subtype clusters computed using different information: all 98 features; 39 stably inherited features; 59 not stably inherited features. The red vertical lines indicate the median for each subtype in each experiment.

Figure 4.9: `Qombucha` inference identifies stably inherited methylation changes in brain development.

bulk mixture B_{sim}. The goal is to evaluate whether `Qombucha` is able to recover the simulated ground truth cell proportions in P_{sim} as well as the held out entries in C_0. As can be observed in Figure 4.10a, the simulated cell fractions in P_{sim} and `Qombucha`-imputed cell fractions, as well as the held-out entries of C_0 and the `Qombucha`-imputed held-out entries, are both highly correlated. These results demonstrate the ability of `Qombucha` to solve for unknown entries in both C and P via the iterative process.

4.4.4.2 Qombucha executes numerous iterations rapidly and achieves quick convergence

The iterative update of matrices converges when the matrix reconstruction error is small and the change in objective function value between two consecutive iterations of Qombucha is below a user-defined threshold. The user-defined threshold parameters for the normalized matrix reconstruction error and the difference in objective value are denoted by δ and λ, respectively.

To evaluate the running time of Qombucha, we tested it with maximum iteration values of 100, 200, 300, 400, and 500 while setting the convergence tolerance parameters, δ and λ, to $-\infty$. Using the bulk methylation matrix of 896 patients, processed as outlined in Section 4.3.4.2, as well as the partial reference panel C and the initial reference panel C_0 obtained as described in Section 4.3.4.3, we observed that Qombucha completed 100 iterations in 23 minutes and 500 iterations in 114 minutes on a single CPU. Unsurprisingly, the runtime scales linearly with the number of iterations, as shown in Figure 4.11.

The choice of δ and λ may be critical in balancing runtime and solution accuracy. In our application of Qombucha to the GBM patient data, we observe that Δ decreases and stabilizes rapidly, as shown in Figure 4.10b(i) and 4.10b(iii), suggesting that running too many iterations may be unnecessary. After performing a parameter sweep, we noticed that setting δ to 0.0125 and λ to 10^{-9} offers a good trade-off between accuracy and runtime. With this setting, Qombucha converged within 86 iterations with the previously described bulk methylation matrix of 896 patients, partial C and initial C_0 as input. For all Qombucha deconvolution of real patient data in this study, we set max_iter, the user-defined parameter specifying the maximum number of allowed Qombucha iterations to 100 while continuing to set δ to 0.0125 and λ to 10^{-9}.

4.4.4.3 `Qombucha` is robust to different starting points

We evaluated two alternative ways to fill in unknown entries in the initial value of the matrix C_p to obtain C_0 to test the robustness of iterative process of `Qombucha`, as descibred in Section 4.3.5.2: (i) random initialization, which is drawing values uniformly at random in the range $[0, 1]$, and (ii) "mean of descendants" initialization, which is deterministically setting the methylation values of each row corresponding to a progenitor cell to be the average of its direct descendants in the tree of brain history development.

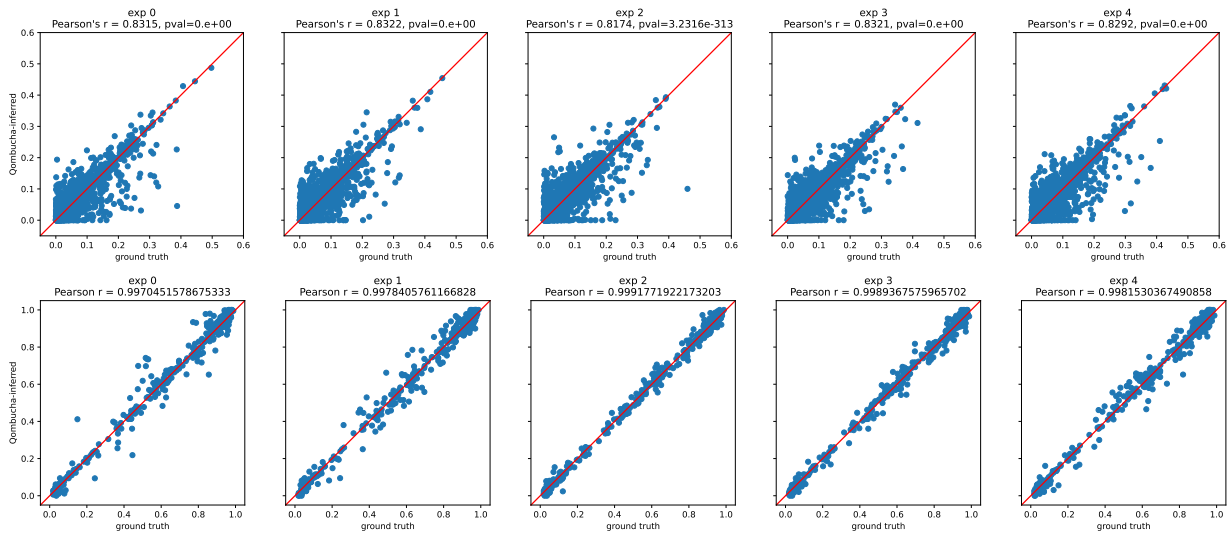
Random and mean starting points give comparable yet slightly higher objective value compared to the tree-informed QP-based method. The tree-informed QP-based initialization method is preferable because (i) it still has the best objective value and (ii) $\Lambda = |\Delta_\iota - \Delta_{\iota-1}|$ converges faster, as can be observed in Figure 4.10b.

4.4.4.4 `Qombucha` yields robust inferred cell compositions despite subsampling B

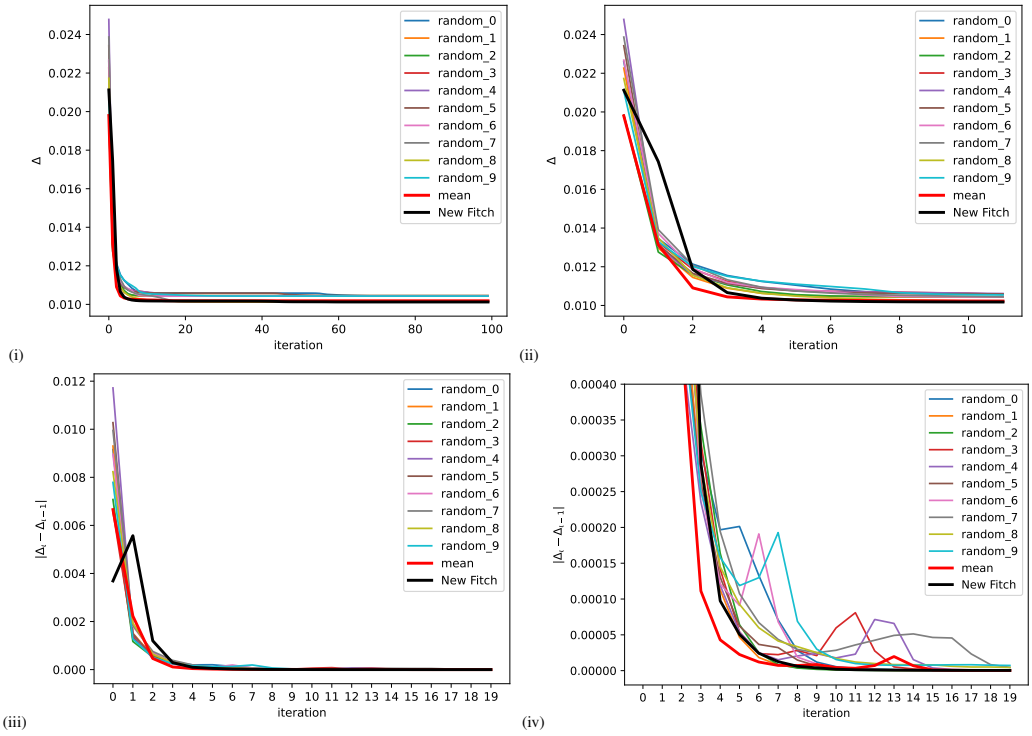
From the bulk methylation matrix of 896 patients we processed as described in Section 4.3.4.2, we generated 10 instances of randomly subsampling 100 patient samples from each of the three subtypes and used each set of 300 patient samples as input. As shown in Figure 4.12, the median inferred cell fractions of each subtype are largely consistent across the random instances. This result shows that `Qombucha`-imputed cell compositions are mostly robust to selection bias in input patient samples.

4.5 Discussion

In this study, we introduce `Qombucha`, a combinatorial optimization-based framework that leverages known brain developmental history to infer simultaneously methylation profiles of progenitor brain cells and cell type compositions from the methylation data of bulk GBM samples. We show that `Qombucha` quickly produces optimal/near-optimal solutions and yields robust inference of cell type compositions despite added input noise or data subsampling. The application of `Qombucha` to new GBM data provided valuable biological insights, revealing that different brain developmental programs are associated with different GBM subtypes. Furthermore, we demonstrate that `Qombucha`-inferred cell fractions per patient sample provide tighter clustering of GBM subtypes compared to other types of information, including the methylation values of 32k brain cancer subtype distinguishing methylation sites identified by [121]. In future work, we plan to explore how these developmental programs are linked to patient outcomes on other cohorts and to extend the application of `Qombucha` to other brain cancer subtypes.



(a) Correlation between values of simulated ground truth and `Qombucha`-inferred values. (**Top**) Correlation of simulated cell fractions and `Qombucha`-inferred cell fractions. (**Bottom**) Correlation of held-out entries in C_0 and `Qombucha`-inferred values of the held-out entries.



(b) Informed starting point gives smaller objective value Δ and less volatile changes in Λ. $\Lambda = \Delta_\iota - \Delta_{\iota-1}$. Random starting points are marked in thin lines. Starting point informed by averaging the mean of ancestors is marked in thick red lines. Starting points informed by the modified Fitch algorithm followed by the QP-based optimization is marked in thick black lines. (i) Change in Δ across iterations. (ii) Change in Δ across the first 10 iterations (zoomed-in version of (i)). (iii) Change in Λ across iterations. (iv) Change in Δ across the first 20 iterations (zoomed-in version of (iii)).

Figure 4.10: `Qombucha` is robust and efficient.

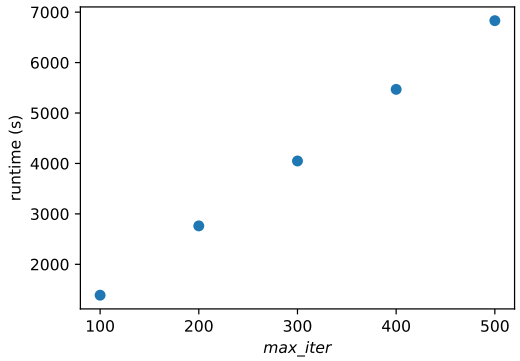


Figure 4.11: Runtime for different numbers of maximum iterations.

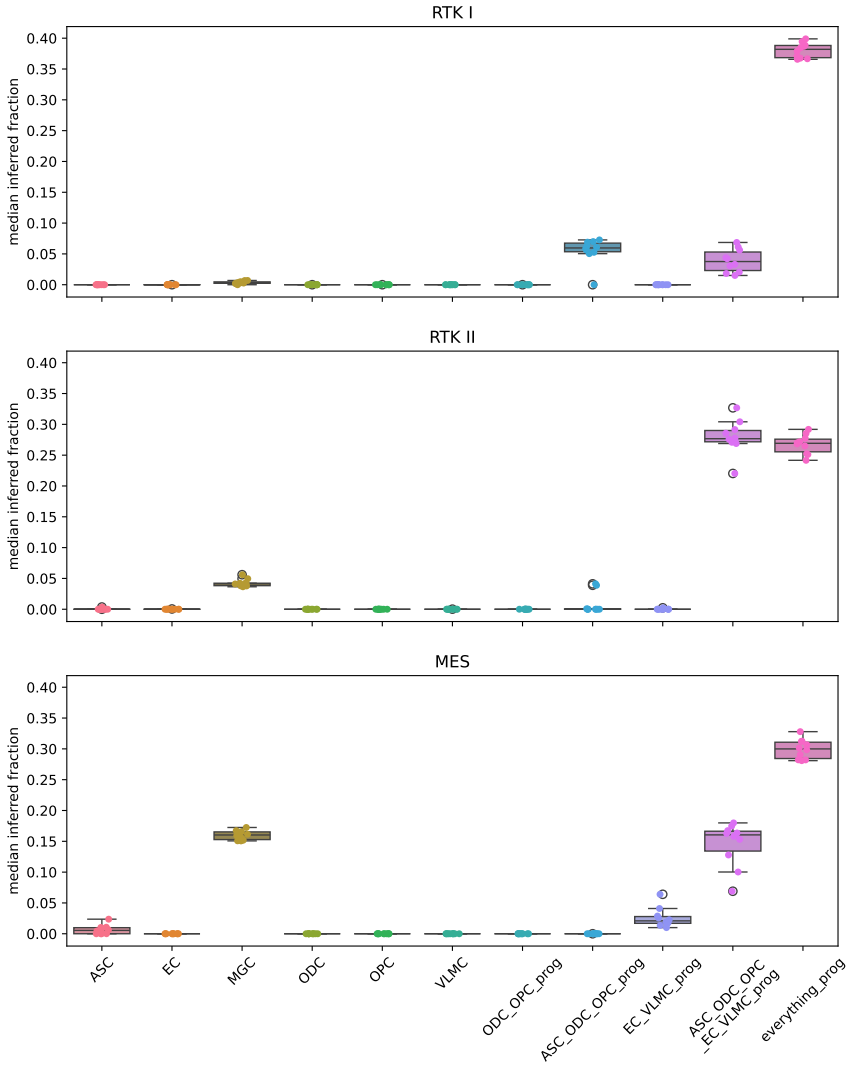


Figure 4.12: The distribution of median (across patients) inferred cell type fraction for each subtype across 10 random instances. Each instance involves randomly selecting 100 patient samples from each of the three subtypes to be used as the input matrix B.

Chapter 5: Phylogenetic Reconstruction of the Adaptive Immune Receptor Repertoire

As discussed in Chapter 1, proliferation of trained or activated B and T cells give rise to the adaptive immune receptor repertoire (*AIRR*). Sequencing and characterizing the immune repertoire through AIRR-seq is critical for basic and clinical immunology research, including vaccine design, therapeutic antibody discovery, minimal-residual disease detection, and monitoring of responses to therapy [180]. Our research proposal focuses on the BCR—rather than TCR—repertoire since the problem is harder. The following sections detail the mechanism behind the evolution of the BCR repertoire within an individual, the definition of the main problem we are interested in, previous literature related to the problem, and finally, we lay down the framework for our approach.

5.1 Evolution of the B cell Receptor Repertoire

The antibody or the B cell receptor molecule sticks out on the surface of a *mature* B cell (free-floating antibodies in blood are these molecules repeatedly produced by activated B cells). The molecule, as depicted in Figure 5.1, comprises arms made of immunoglobulin *heavy* and *light* chains [1]. We ignore the "constant" genes in this section for the sake of simplicity.

A B cell receptor or an antibody molecule is coded by a combination of heavy and light

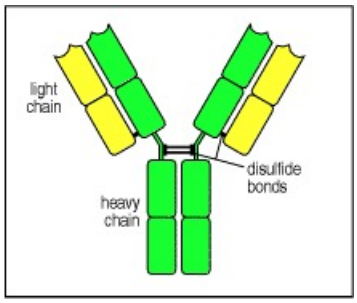


Figure 5.1: An antibody molecule. The stem of the "Y" is coded by the constant region of the heavy chain. The yellow arms are identical light chain regions while the green arms are identical heavy chain regions. Figure from [1].

chain regions of the immunoglobulin locus. Each heavy chain region of the BCR sequence is composed of a "variable" gene (*IGHV*), "diversity" gene (*IGHD*), and a "joining" gene (*IGHJ*), which are excised and brought together by a process named *V(D)J recombination*. Similarly, the light chain region is composed of a "variable" gene and a "joining" gene, both from either the *Immunoglobulin Kappa* or the *Immunoglobulin Lambda* locus. An illustration of the V(D)J recombination is present in Figure 5.2[1].

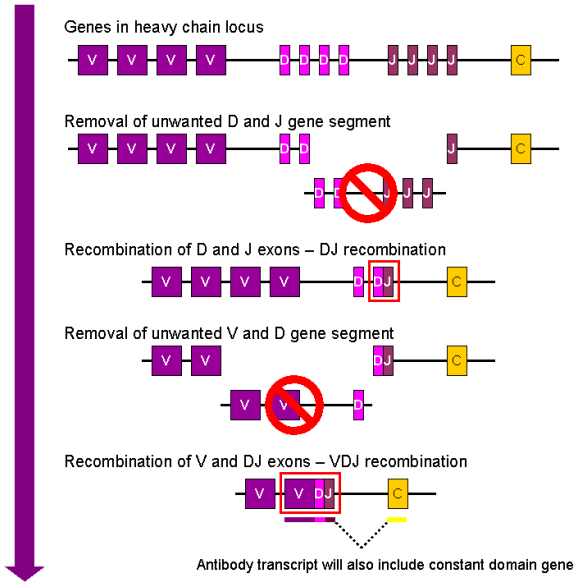


Figure 5.2: V(D)J recombination.

[1]Image taken from `https://en.wikipedia.org/wiki/V(D)J_recombination#/media/File:VDJ_recombination.png`

5.1.1 Affinity Maturation via Somatic Hypermutation

When a human body encounters unknown pathogens, the innate immune system mounts a defense as soon as it detects non-self antigens. This defense is non-antigen specific and not always very effective. The adaptive immune system begins working towards an antigen-specific defense. The first step is to obtain a variety of naive B cells that are negatively selected against self-antigens through V(D)J recombination, as described above. While these B cells can produce BCRs and antibodies right away, the force with which they attract (or bind to), namely the *binding affinity*, against the antigen would likely be low. Therefore, they enter secondary lymphoid organs such as spleen and lymph form a special structure known as the *germinal center* [181], to increase their binding affinity through a process known as *somatic hypermutation* (SHM).

This process accumulates mutations, mostly in the binding domains of the BCR molecule, at a rate of about 10^5–10^6 higher than the normal mutation rate across the whole genome in a short amount of time (less than a week). Within the germinal center, there is approximately a constant flow of antigens which the currently available set of B cells bind to. As more and more mutations are introduced, the binding affinity of the BCR increases. There are usually only single nucleotide variants (or *point mutations*) introduced with very few (or no) insertions or deletions. Most of the mutations are introduced in *complementarity-determining regions* (CDRs); the sites involved in antigen recognition on the immunoglobulin.

In a BCR repertoire, all BCR sequences that originate from the same naive (unmutated) V(D)J recombination form a *clonotype* (a tuple of (V, D, J) genes for a heavy chain clonotype and a tuple of (V, J) genes for a light chain clonotype). Usually, there are millions of clonotypes in an individual's BCR repertoire [182]. However, when actively responding to a pathogenic

attack, it is not uncommon for the BCR repertoire to be dominated by only a few clonotypes, such as in the response to SARS-CoV-2 infection and/or vaccine [183]. BCR repertoire analyses of a patient cohort are often limited to a few dominant clonotypes.

5.2 Building a BCR Phylogeny Forest from AIRR-seq Data

The most basic form of the BCR phylogeny reconstruction problem involves inferring an evolutionary tree for each clonotype, given extant B cells and their assembled BCR sequences (*leaves*). The *root* node of each tree describing the BCR evolution is the unmutated V(D)J-recombined germline alleles of *IGHV*, *IGHD*, and *IGHJ* genes. Ignoring duplicates of the extant BCR sequences, there exists a unique path from the root to each BCR sequence, which constitutes a *lineage*. The problem is unique for different input abstraction levels and also for different sequencing technologies.

1. Sequencing technologies: The input to the problem could be either:

    - Bulk RNA-seq
    - Single-cell RNA-seq (scRNA-seq)
    - Bulk AIRR-seq (or more specifically, BCR-seq)
    - Single-cell AIRR/BCR-seq

    in increasing levels of resolution in capturing the evolution of B cell receptor sequences.

2. Input abstraction levels:

    - Since building phylogenetic trees all the way from RNA/BCR sequencing reads is too cumbersome most of the time, available tools may assume that the BCR sequences

have already been assembled from the reads, using BCR germline inference tools like TIgGER [22], TRUST-4 [184] etc.

- We aim to directly use the sequencing reads for our methods.

We are choosing to directly use the sequencing reads because we are able to actually infer the germline immunoglobulin genes, thanks to our previously published tool ImmunoTyper-SR [17], bypassing the need for the germline inference tools. However, an important caveat is that ImmunoTyper-SR needs whole-genome sequencing (WGS) reads, apart from the AIRR-seq reads needed for building the phylogenies for BCRs. It is to be noted that cohort datasets that sequence both WGS and AIRR (or even RNA-seq) are extremely rare to obtain.

5.3 Previous Literature

A recent review article [185] lists many state-of-the-art tools for building BCR phylogenies using single-cell RNA sequencing technologies. Of these, the most popular methods/tools that are being widely used are: IgPhyML [186], GCtree [187], SAMM [188], TRIBAL [189] and maximum likelihood- and maximum parsimony-based methods that are not uniquely designed for BCR phylogenies such as dnapars/dnaml [190] and IQ-TREE [191].

All of these methods use either Maximum Likelihood (ML), Maximum Parsimony (MP), Bayesian Inference, or a combination of two or more of these methods. While most of these methods solve the BCR phylogeny problem to varying levels of accuracy, they vary wildly in the assumptions made and the constraints of model of BCR evolution. [188] provides a fantastic benchmarking tool to compare various phylogenetic methods for BCR evolution, while also providing a simulator to generate BCR sequences for validation. We intend to make full use of this

simulator/benchmarking platform to validate any algorithmic method that we develop.

5.4 Our Proposed Approaches

The problem, at its most basic version, is NP-hard, since it reduces to the large parsimony problem [192], even for a single clonotype. We are currently attacking this problem from multiple angles. While our current focus is on solving the problem using serial algorithms, we also aim to design parallel algorithms as a part of our research. The ILP/QIP solver [47] (Integer Linear Program/Quadratic Integer Program) we use also inherently uses parallel algorithms/heuristics to quickly get close to the optimal solution. The following sections detail our proposed methods.

5.5 Constrained Parsimony Models and their Quadratic Integer Programming Formulations

A mathematical formulation of the BCR phylogeny reconstruction problem requires an evolutionary model for BCRs. These "constrained parsimony models" specify what kind of changes can occur to each character of the BCR sequence during its evolution. A mathematical formulation of the BCR phylogeny reconstruction problem formulates this model as a linear or a quadratic integer program.

We have considered several constrained parsimony models for the BCR phylogeny reconstruction problem. For each of these models we will be using quadratic integer programming (QIP) to solve reconstructing the BCR phylogeny through the use of the combinatorial optimization toolkit, Gurobi [47]. Since we do not have access to data that has both the whole genome and transcriptome (including AIRR) sequenced, we will assume that our input consists of assembled

BCR sequences and inferred germline sequences.

The BCR phylogeny reconstruction problem requires a solution for all clonotypes at once. As a result the BCR phylogeny reconstruction problem can be thought as a generalized version of the standard phylogeny reconstruction problem. In terms of the number of constraints it is an order of magnitude larger than the standard phylogeny reconstruction problems encountered in evolutionary studies and thus requires substantial computing resources. The QIP formulations we have been developing are currently intractable; as a result, we now aim to break them down into smaller independent components and solve them in parallel.

Each of our QIP formulations are based on a set of constraints on the evolutionary model, which need to be translated into equality and/or inequality constraints, as well as an explicit objective function that the formulation asks to maximize or minimize within these constraints. We are currently considering only unmutated (germline) *IGHV* allele sequences, even though each clonotype evolves from V(D)J rearranged sequences at the root. The input to each QIP formulation is a connected component, consisting of assembled BCR sequences and a set of candidate alleles to which these sequences have high sequence similarity/low sequence distance, as measured through their multiple sequence alignment (MSA). These connected components are a result of a distance-based clustering of all available BCR sequences and the IGHV allele database [46]. The output is a conflict-free matrix per clonotype and the implied phylogenetic tree [193]. We use variants of the integer linear programming formulation from [193], which reconstructs clonal lineages of tumor cells assuming a restricted parsimony model. As will be described in the next two subsections, we use two types of restricted parsimony models in our work: “Perfect Phylogeny” and “Camin-Sokal phylogeny”.

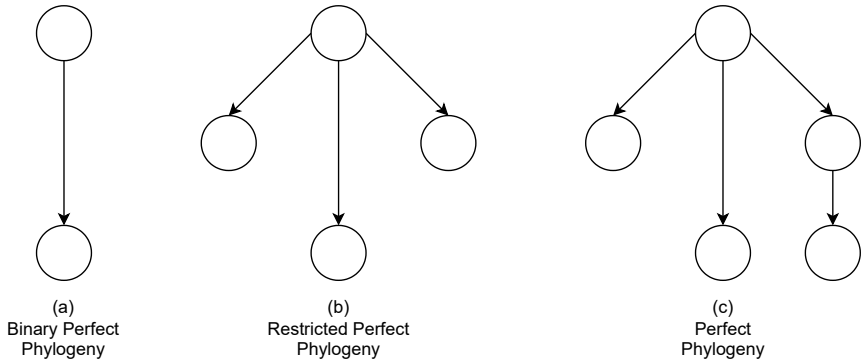


Figure 5.3: Models of perfect phylogeny of character states under various models of parsimony.

5.5.1 Perfect Phylogeny Models

In this subsection, we assume that all clonotypes evolve either as a *perfect phylogeny* (PP), a *restricted perfect phylogeny* (RPP), or a *binary perfect phylogeny* (BPP). This key assumption makes the problem more tractable. Within the realm of perfect phylogenies, we define four distinct evolutionary models for BCRs below. In each of these models, the root of a BCR phylogeny represents the sequence of a distinct germline V(D)J recombination. A character of the root sequence is said to evolve with *no homoplasy* within the BCR phylogeny if that character undergoes each possible state change *at most* once. The PP assumption stipulates that a specific state change of any character can happen only once in the entire evolutionary tree. A PP is BPP if the allowed character states are binary (0 and 1). In a restricted perfect phylogeny (RPP) a character cannot change states more than once in a lineage. These three variants of the perfect phylogeny assumption are illustrated in Figure 5.3.

Below we describe four distinct models of BCR evolution, each based on the PP assumption, as well as the QIP formulations to solve them.

5.5.1.1 Evolutionary Model 1 and its QIP Formulation

Assumptions about the data in each connected component:

- The data has no sequencing errors, i.e., all assembled BCR sequences, including the set of candidate alleles, need no flips/corrections to fit in phylogenetic trees.
- Only one allowed mutation per character from root along any path to any leaf in a tree aka the evolution model is the restricted perfect phylogeny (RPP).
- We have all nodes necessary to build the forest and we don't have false positive leaves.

Objective: Minimize the number of roots called per cluster of root and leaf nodes.

Limitation: There is no way to discard BCR sequences that do not belong to a particular BCR clonotype.

5.5.1.2 Evolutionary Model 2 and its QIP Formulation

Assumptions about the data in each connected component:

- Sequencing errors are allowed in the assembled sequences.
- Each substitution in a character in the root is treated as the same kind of mutation.
- There are at most k roots in the dataset, where k is a user-specified constant.
- We have all nodes necessary to build the forest and we don't have false positive leaves.

Objective: Minimize the number of flips in the character states of the leaf nodes to fit a binary perfect phylogeny.

Limitation: There is no way to discard BCR sequences that do not belong to a particular BCR clonotype.

5.5.1.3 Evolutionary Model 3 and its QIP Formulation

Assumptions about the data in each connected component:

- Sequencing errors are allowed in the assembled sequences.
- Each substitution in a character in the root is treated as the same kind of mutation.
- Choose only **one** root node among all available root nodes for the connected component.
- Nodes/BCR sequences can be discarded, if they do not fit the parsimonious model of the phylogeny.

Objective: Minimize the number of discarded BCR sequences (potential nodes in the tree) such that the set of remaining nodes fit a binary perfect phylogeny.

Limitation: All available BCR sequences are forced to fit exactly one (most parsimonious) phylogenetic tree, per connected component.

5.5.1.4 Evolutionary Model 4 and its QIP Formulation

This model extends Model 5.5.1.3 so that the number of roots that can be chosen from the possible set is at most k, instead of 1. The rest of the assumptions and the objective are identical to that of

Model 5.5.1.3.

Limitation: Too hard to solve as a QIP even for a modest value of k.

Therefore, we propose an **iterative** heuristic method to solve this model, which is based on repeatedly applying Model 5.5.1.3, as follows:

1. Choose the root node, which, among all root nodes, is the nearest to the largest number of BCR sequences, and parsimoniously assign all these BCR sequences/nodes to it.

2. Iteratively choose the next root node for the remaining (unassigned) BCR sequences/nodes.

3. Stop when the remaining set of sequences/nodes differ too much from any root node left in the pool.

Since this iterative approach is manageable in terms of memory, number of CPUs and runtime it is highly promising. Whether it can achieve a level of practical accuracy that is permissible needs to be seen.

5.5.2 Camin-Sokal Parsimony Model

Highly restricted parsimony models for BCR phylogeny reconstruction problem, such as the perfect phylogeny model do not fully capture the biological mechanism of BCR evolution through somatic hypermutations. Alternative models for BCR evolution via less restrictive parsimony models are preferable. Unfortunately, even the BCR phylogeny reconstruction problem based on our general perfect parsimony model is intractable. As mentioned earlier, we aim to address this challenge by introducing a preprocessing step for assigning each BCR sequence to a distinct clonotype. This preprocessing step reduces the problem to a number of distinct but

relatively straightforward "phylogeny reconstruction problems", in each of which we are given a specific root node (the germline V(D)J-recombination sequence) and a set of extant BCR sequences (assembled from AIRR-seq) that presumably have evolved from the root sequence in a clonotype. The goal of each such problem would thus be the construction of the most parsimonious tree for the BCR sequences, given the root sequence, within the chosen evolutionary model. Possible less restrictive alternatives to the *perfect phylogeny* model include the *k-star homoplasy* model (for which a combinatorial optimization formulation has recently been developed [2]) and the *bounded Camin-Sokal parsimony* model. In what follows, we describe the key features of these models and discuss their potential use in BCR phylogeny reconstruction.

Various parsimony models for evolving character states have been studied in the phylogenetics literature based on the number of state transitions allowed, reversibility of mutations acquired, and bounds on number of mutational losses/gains; see Figure 5.4. Among them, the least restrictive parsimony model allows a character to transition to any other state in any node (i.e. without any restriction). The star homoplasy restricts this model by allowing a character to transition to any other state only once in a lineage. Once a character has changed states, the subtree rooted at this node has the same state for the character for all of its nodes.

A better alternative to the star homoplasy for BCR phylogeny reconstruction is the *Camin-Sokal* parsimony model, where state changes are "irreversible". This means that, a character can switch from state $A \rightarrow T \rightarrow G$ along a lineage but it can never go back to a previous state (here, once the character is in state G, it cannot mutate to A or T). Since somatic hypermutation for B cells are biased towards those that have greater affinity towards specific antigens, we can hypothesize that there is some evolutionary pressure to gain mutations until an "ideal" BCR sequence/antibody that binds well with the targeted antigen is reached. Unfortunately, while

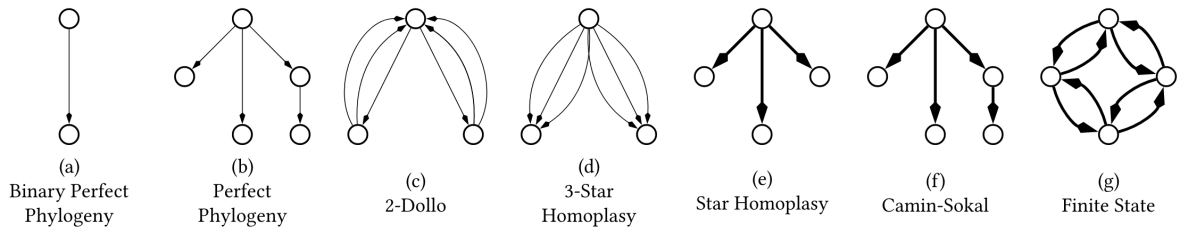


Figure 5.4: Models of evolution of character states under various models of parsimony. Models decrease in the number of evolutionary constraints as one goes from left to right. Thick lines indicate edges with no restrictions on the number of times a transition can occur, i.e. infinite multiplicity. Figure from [2].

Camin-Sokal parsimony on binary character states has been extensively studied, the multi-state version of the problem has not been solved [2]. We aim to study a bounded version of this model of parsimony, where the number of state transitions of each character in each root of a clonotype is upper-bounded, and apply it to the BCR phylogeny problem. [2] also gives a method to solve the multi-state "k-star homoplasy" (bounded version of star homoplasy) phylogeny problem, which we intend to extend to phylogenies based on multi-state Camin-Sokal parsimony.

Bibliography

[1] Charles A Janeway Jr, Paul Travers, Mark Walport, and Mark J Shlomchik. The structure of a typical antibody molecule. In *Immunobiology: The Immune System in Health and Disease. 5th edition*. Garland Science, 2001.

[2] Palash Sashittal, Henri Schmidt, Michelle Chan, and Benjamin J Raphael. Startle: a star homoplasy approach for crispr-cas9 lineage tracing. *Cell Systems*, 14(12):1113–1121, 2023.

[3] Juan-Manuel Anaya, Yehuda Shoenfeld, Adriana Rojas-Villarraga, Roger A Levy, and Ricard Cervera. *Autoimmunity: from bench to bedside [Internet]*. El Rosario University Press, 2013.

[4] David D Chaplin. 1. overview of the immune response. *Journal of Allergy and Clinical Immunology*, 111(2):S442–S459, 2003.

[5] Kathrin Pieper, Bodo Grimbacher, and Hermann Eibel. B-cell biology and development. *Journal of Allergy and Clinical Immunology*, 131(4):959–971, 2013.

[6] Charles A Janeway Jr, Paul Travers, Mark Walport, and Mark J Shlomchik. Principles of innate and adaptive immunity. In *Immunobiology: The Immune System in Health and Disease. 5th edition*. Garland Science, 2001.

[7] Brahma V Kumar, Thomas J Connors, and Donna L Farber. Human t cell development, localization, and function throughout life. *Immunity*, 48(2):202–213, 2018.

[8] Richard M Weinshilboum and Liewei Wang. Pharmacogenomics: precision medicine and drug response. In *Mayo Clinic Proceedings*, volume 92, pages 1711–1722. Elsevier, 2017.

[9] Magnus Ingelman-Sundberg. Genetic polymorphisms of cytochrome P450 2D6 (CYP2D6): Clinical consequences, evolutionary aspects and functional diversity. *The pharmacogenomics journal*, 5(1):6–13, 2004.

[10] Gregor Mendel. Experiments in plant hybridization (1865). *Verhandlungen des naturforschenden Vereins Brünn.) Available online: www. mendelweb. org/Mendel. html (accessed on 1 January 2013)*, 1996.

[11] Mark B Gerstein, Can Bruce, Joel S Rozowsky, Deyou Zheng, Jiang Du, Jan O Korbel, Olof Emanuelsson, Zhengdong D Zhang, Sherman Weissman, and Michael Snyder. What

is a gene, post-encode? history and updated definition. *Genome research*, 17(6):669–681, 2007.

[12] Iakes Ezkurdia, David Juan, Jose Manuel Rodriguez, Adam Frankish, Mark Diekhans, Jennifer Harrow, Jesus Vazquez, Alfonso Valencia, and Michael L Tress. Multiple evidence strands suggest that there may be as few as 19 000 human protein-coding genes. *Human molecular genetics*, 23(22):5866–5878, 2014.

[13] Evgeniy S Balakirev and Francisco J Ayala. Pseudogenes: are they "junk" or functional dna? *Annual review of genetics*, 37(1):123–151, 2003.

[14] Paul M Harrison, Deyou Zheng, Zhaolei Zhang, Nicholas Carriero, and Mark Gerstein. Transcribed processed pseudogenes in the human genome: an intermediate form of expressed retrosequence lacking protein-coding ability. *Nucleic acids research*, 33(8):2374–2383, 2005.

[15] Eleonora Market and F Nina Papavasiliou. V (d) j recombination and the evolution of the adaptive immune system. *PLoS biology*, 1(1):e16, 2003.

[16] Veronique Giudicelli, Patrice Duroux, Chantal Ginestoux, Geraldine Folch, Joumana Jabado-Michaloud, Denys Chaume, and Marie-Paule Lefranc. Imgt/ligm-db, the imgt® comprehensive database of immunoglobulin and t cell receptor nucleotide sequences. *Nucleic acids research*, 34(suppl_1):D781–D784, 2006.

[17] Michael K.B. Ford, Ananth Hari, Oscar Rodriguez, Junyan Xu, Justin Lack, Cihan Oguz, Yu Zhang, Andrew J. Oler, Ottavia M. Delmonte, Sarah E. Weber, Mary Magliocco, Jason Barnett, Sandhya Xirasagar, Smilee Samuel, Luisa Imberti, Paolo Bonfanti, Andrea Biondi, Clifton L. Dalgard, Stephen Chanock, Lindsey B. Rosen, Steven M. Holland, Helen C. Su, Luigi D. Notarangelo, Uzi Vishkin, Corey T. Watson, S. Cenk Sahinalp, Kerry Dobbs, Elana Shaw, Miranda F. Tompkins, Camille Alba, Adelani Adeleye, Samuel Li, and Jingwen Gu. ImmunoTyper-SR: A computational approach for genotyping immunoglobulin heavy chain variable genes using short-read data. *Cell Systems*, 13(10):808–816.e5, October 2022.

[18] C T Watson and F Breden. The immunoglobulin heavy chain locus: genetic variation, missing data, and implications for human disease. *Genes And Immunity*, 13:363, 5 2012.

[19] Corey T Watson, Jacob Glanville, and Wayne A Marasco. The individual and population genetics of antibody immunity. *Trends in immunology*, 38(7):459–470, 2017.

[20] Andrew M. Collins, Gur Yaari, Adrian J. Shepherd, William Lees, and Corey T. Watson. Germline immunoglobulin genes: Disease susceptibility genes hidden in plain sight? *Current Opinion in Systems Biology*, 24:100–108, 2020.

[21] Corey T. Watson, Karyn M. Steinberg, John Huddleston, Rene L. Warren, Maika Malig, Jacqueline Schein, A. Jeremy Willsey, Jeffrey B. Joy, Jamie K. Scott, Tina A. Graves, Richard K. Wilson, Robert A. Holt, Evan E. Eichler, and Felix Breden. Complete haplotype sequence of the human immunoglobulin heavy-chain variable, diversity, and joining

genes and characterization of allelic and copy-number variation. *American Journal of Human Genetics*, 92(4):530–546, 2013.

[22] Daniel Gadala-Maria, Moriah Gidoni, Susanna Marquez, Jason A Vander Heiden, Justin T Kos, Corey T Watson, Kevin C O'Connor, Gur Yaari, and Steven H Kleinstein. Identification of subject-specific immunoglobulin alleles from expressed repertoire sequencing data. *Frontiers in immunology*, 10:129, 2019.

[23] Oscar L. Rodriguez, William S. Gibson, Tom Parks, Matthew Emery, James Powell, Maya Strahl, Gintaras Deikus, Kathryn Auckland, Evan E. Eichler, Wayne A. Marasco, Robert Sebra, Andrew J. Sharp, Melissa L. Smith, Ali Bashir, and Corey T. Watson. A Novel Framework for Characterizing Genomic Haplotype Diversity in the Human Immunoglobulin Heavy Chain Locus. *Frontiers in Immunology*, 11(September):1–16, 2020.

[24] Andrew M Collins, Ayelet Peres, Martin M Corcoran, Corey T Watson, Gur Yaari, William D Lees, and Mats Ohlin. Commentary on population matched (pm) germline allelic variants of immunoglobulin (ig) loci: relevance in infectious diseases and vaccination studies in human populations. *Genes & Immunity*, pages 1–4, 2021.

[25] Corey T Watson, Frederick A Matsen, Katherine J L Jackson, Ali Bashir, Melissa Laird Smith, Jacob Glanville, Felix Breden, Steven H Kleinstein, Andrew M Collins, and Christian E Busse. Comment on "A Database of Human Immune Receptor Alleles Recovered from Population Sequencing Data". *Journal of immunology (Baltimore, Md. : 1950)*, 198(9):3371, 2017.

[26] Yuval Avnir, Corey T. Watson, Jacob Glanville, Eric C. Peterson, Aimee S. Tallarico, Andrew S. Bennett, Kun Qin, Ying Fu, Chiung Yu Huang, John H. Beigel, Felix Breden, Quan Zhu, and Wayne A. Marasco. IGHV1-69 polymorphism modulates anti-influenza antibody repertoires, correlates with IGHV utilization shifts and varies by ethnicity. *Scientific Reports*, 6(February):1–11, 2016.

[27] Yik Andy Yeung, Davide Foletti, Xiaodi Deng, Yasmina Abdiche, Pavel Strop, Jacob Glanville, Steven Pitts, Kevin Lindquist, Purnima D Sundar, Marina Sirota, et al. Germline-encoded neutralization of a staphylococcus aureus virulence factor by the human antibody repertoire. *Nature communications*, 7(1):1–14, 2016.

[28] Christina Yacoob, Marie Pancera, Vladimir Vigdorovich, Brian G. Oliver, Jolene A. Glenn, Junli Feng, D. Noah Sather, Andrew T. McGuire, and Leonidas Stamatatos. Differences in Allelic Frequency and CDRH3 Region Limit the Engagement of HIV Env Immunogens by Putative VRC01 Neutralizing Antibody Precursors. *Cell Reports*, 17(6):1560–1570, 11 2016.

[29] Jeong Hyun Lee, Laura Toy, Justin T Kos, Yana Safonova, William R Schief, Colin Havenar-Daughton, Corey T Watson, and Shane Crotty. Vaccine genetics of ighv1-2 vrc01-class broadly neutralizing antibody precursor naïve human b cells. *NPJ vaccines*, 6(1):1–12, 2021.

[30] Mi La Cho, Pojen P. Chen, Young Il Seo, Sue Yun Hwang, Wan Uk Kim, Do June Min, Sung Hwan Park, and Chul Soo Cho. Association of homozygous deletion of the Humhv3005 and the VH3-30.3 genes with renal involvement in systemic lupus erythematosus. *Lupus*, 12(5):400–405, 2003.

[31] Todd A Johnson, Yoichi Mashimo, Jer-Yuarn Wu, Dankyu Yoon, Akira Hata, Michiaki Kubo, Atsushi Takahashi, Tatsuhiko Tsunoda, Kouichi Ozaki, Toshihiro Tanaka, et al. Association of an ighv3-66 gene variant with kawasaki disease. *Journal of human genetics*, 66(5):475–489, 2021.

[32] Tom Parks, Mariana M Mirabel, Joseph Kado, Kathryn Auckland, Jaroslaw Nowak, Anna Rautanen, Alexander J Mentzer, Eloi Marijon, Xavier Jouven, Mai Ling Perman, et al. Association between a common immunoglobulin heavy chain allele and rheumatic heart disease risk in oceania. *Nature communications*, 8(1):1–10, 2017.

[33] Ming Cui, Jing Huang, Shenghua Zhang, Qiaofei Liu, Quan Liao, and Xiaoyan Qiu. Immunoglobulin expression in cancer cells and its critical roles in tumorigenesis. *Frontiers in immunology*, 12:893, 2021.

[34] Joachim L. Schultze and Anna C. Aschenbrenner. COVID-19 and the human innate immune system. *Cell*, 184(7):1671–1692, 2021.

[35] Paul Bastard, Lindsey B Rosen, Qian Zhang, Eleftherios Michailidis, Hans-Heinrich Hoffmann, Yu Zhang, Karim Dorgham, Quentin Philippot, Jérémie Rosain, Vivien Béziat, et al. Autoantibodies against type i ifns in patients with life-threatening covid-19. *Science*, 370(6515):eabd4585, 2020.

[36] Eric Y Wang, Tianyang Mao, Jon Klein, Yile Dai, John D Huck, Jillian R Jaycox, Feimei Liu, Ting Zhou, Benjamin Israelow, Patrick Wong, et al. Diverse functional autoantibodies in patients with covid-19. *Nature*, 595(7866):283–288, 2021.

[37] Monique GP van der Wijst, Sara E Vazquez, George C Hartoularos, Paul Bastard, Tianna Grant, Raymund Bueno, David S Lee, John R Greenland, Yang Sun, Richard Perez, et al. Type i interferon autoantibodies are associated with systemic immune alterations in patients with covid-19. *Science translational medicine*, 13(612):eabh2624, 2021.

[38] Shishi Luo, Jane A. Yu, and Yun S. Song. Estimating Copy Number and Allelic Variation at the Immunoglobulin Heavy Chain Locus Using Short Reads. *PLoS Computational Biology*, 12(9):1–21, 2016.

[39] Shishi Luo, A Yu Jane, Heng Li, and Yun S Song. Worldwide genetic variation of the IGHV and TRBV immune receptor gene families in humans. *Life science alliance*, 2(2), 2019.

[40] Michael Ford, Ehsan Haghshenas, Corey T Watson, and S Cenk Sahinalp. Genotyping and Copy Number Analysis of Immunoglobulin Heavy Chain Variable Genes Using Long Reads. *iScience*, 23(9):101508, 9 2020.

[41] Ayelet Peres, Moriah Gidoni, Pazit Polak, and Gur Yaari. RAbHIT: R Antibody Haplotype Inference Tool. *Bioinformatics*, 35(22):4840–4842, 2019.

[42] Daniel Gadala-Maria, Moriah Gidoni, Susanna Marquez, Jason Anthony Vanderheiden, Justin T Kos, Corey Watson, Kevin C O'Connor, Gur Yaari, and Steven H Kleinstein. Identification of subject-specific immunoglobulin alleles from expressed repertoire sequencing data. *Frontiers in Immunology*, 10(February):1–12, 1 2019.

[43] 1000-Genomes-Project-Consortium. A global reference for human genetic variation. *Nature*, 526(7571):68, 2015.

[44] Marta Byrska-Bishop, Uday S. Evani, Xuefang Zhao, Anna O. Basile, Haley J. Abel, Allison A. Regier, André Corvelo, Wayne E. Clarke, Rajeeva Musunuri, Kshithija Nagulapalli, Susan Fairley, Alexi Runnels, Lara Winterkorn, Ernesto Lowy-Gallego, The Human Genome Structural Variation Consortium, Paul Flicek, Soren Germer, Harrison Brand, Ira M. Hall, Michael E. Talkowski, Giuseppe Narzisi, and Michael C. Zody. High coverage whole genome sequencing of the expanded 1000 genomes project cohort including 602 trios. *bioRxiv*, 2021.

[45] Heng Li. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM, 2013.

[46] MP Lefranc. Imgt® databases, web resources and tools for immunoglobulin and t cell receptor sequence analysis, http://www.imgt.org. *Leukemia*, 17(1):260–266, 2003.

[47] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2021.

[48] Marie Louise Bang, Thomas Centner, Friderike Fornoff, Adam J. Geach, Michael Gotthardt, Mark McNabb, Christian C. Witt, Dietmar Labeit, Carol C. Gregorio, Henk Granzier, and Siegfried Labeit. The complete gene sequence of titin, expression of an unusual 700-kDa titin isoform, and its interaction with obscurin identify a novel Z-line to I-band linking system. *Circulation Research*, 89(11):1065–1072, 2001.

[49] Weichun Huang, Leping Li, Jason R. Myers, and Gabor T. Marth. ART: A next-generation sequencing read simulator. *Bioinformatics*, 28(4):593–594, 2012.

[50] Ben Langmead and Steven L. Salzberg. Fast gapped-read alignment with Bowtie 2. *Nature Methods*, 9(4):357–359, 2012.

[51] Oscar L Rodriguez, Andrew J Sharp, and Corey T Watson. Limitations of lymphoblastoid cell lines for establishing genetic reference datasets in the immunoglobulin loci. *bioRxiv*, 2021.

[52] William Lees, Christian E. Busse, Martin Corcoran, Mats Ohlin, Cathrine Scheepers, Frederick A. Matsen, Gur Yaari, Corey T. Watson, Andrew Collins, and Adrian J. Shepherd. OGRDB: A reference database of inferred immune receptor genes. *Nucleic Acids Research*, 48(D1):D964–D970, 2020.

[53] Michael H Court. A pharmacogenomics primer. *The Journal of Clinical Pharmacology*, 47(9):1087–1103, 2007.

[54] Ananth Hari, Qinghui Zhou, Nina Gonzaludo, John Harting, Stuart A Scott, Xiang Qin, Steve Scherer, S Cenk Sahinalp, and Ibrahim Numanagić. An efficient genotyper and star-allele caller for pharmacogenomics. *Genome Research*, 33(1):61–70, 2023.

[55] Margaret A. Hamburg and Francis S. Collins. The Path to Personalized Medicine. *New England Journal of Medicine*, 363(4):301–304, July 2010.

[56] David Twesigomwe, Galen EB Wright, Britt I Drögemöller, Jorge da Rocha, Zané Lombard, and Scott Hazelhurst. A systematic comparison of pharmacogene star allele calling bioinformatics algorithms: a focus on cyp2d6 genotyping. *NPJ genomic medicine*, 5(1):1–11, 2020.

[57] Hideki Sezutsu, Gaëlle Le Goff, and René Feyereisen. Origins of P450 diversity. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 368(1612):20120428, 2013.

[58] J. D. Robarge, L. Li, Z. Desta, A. Nguyen, and D. A. Flockhart. The star-allele nomenclature: Retooling for translational genomics. *Clinical Pharmacology and Therapeutics*, 82(3):244–248, September 2007.

[59] Aaron McKenna, Matthew Hanna, Eric Banks, Andrey Sivachenko, Kristian Cibulskis, Andrew Kernytsky, Kiran Garimella, David Altshuler, Stacey Gabriel, Mark Daly, and Mark A. DePristo. The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Research*, 20(9):1297–1303, September 2010.

[60] Ryan Poplin, Valentin Ruano-Rubio, Mark A DePristo, Tim J Fennell, Mauricio O Carneiro, Geraldine A Van der Auwera, David E Kling, Laura D Gauthier, Ami Levy-Moonshine, David Roazen, et al. Scaling accurate genetic variant discovery to tens of thousands of samples. *BioRxiv*, page 201178, 2017.

[61] Peter Edge, Vineet Bafna, and Vikas Bansal. HapCUT2: robust and accurate haplotype assembly for diverse sequencing technologies. *Genome research*, 27(5):801–812, 2017.

[62] Emily Berger, Deniz Yorukoglu, Lillian Zhang, Sarah K Nyquist, Alex K Shalek, Manolis Kellis, Ibrahim Numanagić, and Bonnie Berger. Improved haplotype inference by exploiting long-range linking and allelic imbalance in rna-seq datasets. *Nature communications*, 11(1):1–9, 2020.

[63] Brian L Browning, Xiaowen Tian, Ying Zhou, and Sharon R Browning. Fast two-stage phasing of large-scale sequence data. *The American Journal of Human Genetics*, 108(10):1880–1890, 2021.

[64] Po-Ru Loh, Pier Francesco Palamara, and Alkes L Price. Fast and accurate long-range phasing in a UK Biobank cohort. *Nature genetics*, 48(7):811–816, 2016.

[65] Ibrahim Numanagić, Salem Malikić, Victoria M. Pratt, Todd C. Skaar, David A. Flockhart, and S. Cenk Sahinalp. Cypiripi: Exact Genotyping of CYP2D6 Using High-Throughput Sequencing Data. *Bioinformatics*, 31(12):i27–i34, June 2015.

[66] Victoria M Pratt, Barbara Zehnbauer, Jean Amos Wilson, Ruth Baak, Nikolina Babic, Maria Bettinotti, Arlene Buller, Ken Butz, Matthew Campbell, Chris Civalier, and others. Characterization of 107 Genomic DNA Reference Materials for CYP2D6, CYP2C19, CYP2C9, VKORC1, and UGT1A1: A GeT-RM and Association for Molecular Pathology Collaborative Project. *The Journal of Molecular Diagnostics*, 12(6):835–846, 2010.

[67] H Fang, X Liu, J Ramírez, N Choudhury, M Kubo, HK Im, A Konkashbaev, NJ Cox, MJ Ratain, Y Nakamura, and others. Establishment of CYP2D6 reference samples by multiple validated genotyping platforms. *The pharmacogenomics journal*, 14(6):564–572, 2014.

[68] Sylvan M Caspar, Timo Schneider, Janine Meienberg, and Gabor Matyas. Added value of clinical sequencing: Wgs-based profiling of pharmacogenes. *International journal of molecular sciences*, 21(7):2308, 2020.

[69] Ibrahim Numanagić, Salem Malikić, Michael Ford, Xiang Qin, Lorraine Toji, Milan Radovich, Todd C Skaar, Victoria M Pratt, Bonnie Berger, Steve Scherer, et al. Allelic decomposition and exact genotyping of highly polymorphic and structurally variant genes. *Nature communications*, 9(1):1–11, 2018.

[70] Seung-been Lee, Marsha M Wheeler, Karynne Patterson, Sean McGee, Rachel Dalton, Erica L Woodahl, Andrea Gaedigk, Kenneth E Thummel, and Deborah A Nickerson. Stargazer: a software tool for calling star alleles from next-generation sequencing data using CYP2D6 as a model. *Genetics in Medicine*, 21(2):361–372, 2019.

[71] Greyson P Twist, Andrea Gaedigk, Neil A Miller, Emily G Farrow, Laurel K Willig, Darrell L Dinwiddie, Josh E Petrikin, Sarah E Soden, Suzanne Herd, Margaret Gibson, Julie A Cakici, Amanda K Riffel, J Steven Leeder, Deendayal Dinakarpandian, and Stephen F Kingsmore. Constellation: A tool for rapid, automated phenotype assignment of a highly polymorphic pharmacogene, CYP2D6, from whole-genome sequences. *npj Genomic Medicine*, 1:15007, January 2016.

[72] David Twesigomwe, Britt I Drögemöller, Galen EB Wright, Azra Siddiqui, Jorge da Rocha, Zané Lombard, and Scott Hazelhurst. StellarPGx: A Nextflow Pipeline for Calling Star Alleles in Cytochrome P450 Genes. *Clinical Pharmacology & Therapeutics*, 110(3):741–749, 2021.

[73] Katrin Sangkuhl, Michelle Whirl-Carrillo, Ryan M Whaley, Mark Woon, Adam Lavertu, Russ B Altman, Lester Carter, Anurag Verma, Marylyn D Ritchie, and Teri E Klein. Pharmacogenomics clinical annotation tool (pharm cat). *Clinical Pharmacology & Therapeutics*, 107(1):203–210, 2020.

[74] Xiao Chen, Fei Shen, Nina Gonzaludo, Alka Malhotra, Cande Rogert, Ryan J Taft, David R Bentley, and Michael A Eberle. Cyrius: accurate CYP2D6 genotyping using whole-genome sequencing data. *The pharmacogenomics journal*, 21(2):251–261, 2021.

[75] Andrea Gaedigk, Magnus Ingelman-Sundberg, Neil A Miller, J Steven Leeder, Michelle Whirl-Carrillo, Teri E Klein, and PharmVar Steering Committee. The Pharmacogene Variation (PharmVar) Consortium: incorporation of the human cytochrome P450 (CYP) allele nomenclature database. *Clinical Pharmacology & Therapeutics*, 103(3):399–401, 2018.

[76] Adam S. Gordon, Robert S. Fulton, Xiang Qin, Elaine R. Mardis, Deborah A. Nickerson, and Steve Scherer. PGRNseq: A Targeted Capture Sequencing Panel for Pharmacogenetic Research and Implementation. *Pharmacogenetics and Genomics*, January 2016.

[77] Wouter De Coster, Matthias H Weissensteiner, and Fritz J Sedlazeck. Towards population-scale long-read sequencing. *Nature Reviews Genetics*, 22(9):572–587, 2021.

[78] Ting Hon, Kristin Mars, Greg Young, Yu-Chih Tsai, Joseph W Karalius, Jane M Landolin, Nicholas Maurer, David Kudrna, Michael A Hardigan, Cynthia C Steiner, et al. Highly accurate long-read HiFi sequencing data for five complex genomes. *Scientific data*, 7(1):1–11, 2020.

[79] Reynold C Ly, Tyler Shugg, Ryan Ratcliff, Wilberforce Osei, Ty C Lynnes, Victoria M Pratt, Bryan P Schneider, Milan Radovich, Steven M Bray, Benjamin A Salisbury, et al. Analytical validation of a computational method for pharmacogenetic genotyping from clinical whole exome sequencing. *The Journal of Molecular Diagnostics*, 2022.

[80] Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, Richard Durbin, and 1000 Genome Project Data Processing Subgroup. The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16):2078–2079, 2009.

[81] Heng Li and Richard Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.

[82] Heng Li. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*, 34(18):3094–3100, 2018.

[83] Geraldine A. Van der Auwera, Mauricio O. Carneiro, Chris Hartl, Ryan Poplin, Guillermo del Angel, Ami Levy-Moonshine, Tadeusz Jordan, Khalid Shakir, David Roazen, Joel Thibault, Eric Banks, Kiran V. Garimella, David Altshuler, Stacey Gabriel, and Mark A. DePristo. From FastQ data to high confidence variant calls: The Genome Analysis Toolkit best practices pipeline. *Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.]*, 11(1110):11.10.1–11.10.33, October 2013.

[84] James T Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S Lander, Gad Getz, and Jill P Mesirov. Integrative genomics viewer. *Nature biotechnology*, 29(1):24–26, 2011.

[85] Kohei Hagiwara, Michael N Edmonson, David A Wheeler, and Jinghui Zhang. indelpost: harmonizing ambiguities in simple and complex indel alignments. *Bioinformatics*, 38(2):549–551, 2022.

[86] Heng Li. Toward better understanding of artifacts in variant calling from high-coverage samples. *Bioinformatics*, 30(20):2843–2851, 2014.

[87] Whitney E Kramer, Denise L Walker, Dennis J O'Kane, David A Mrazek, Pamela K Fisher, Brian A Dukek, Jamie K Bruflat, and John L Black. CYP2D6: Novel genomic structures and alleles. *Pharmacogenetics and genomics*, 19(10):813, 2009.

[88] Sarah C Sim, Ann K Daly, and Andrea Gaedigk. CYP2D6 update: revised nomenclature for CYP2D7/2D6 hybrid genes. *Pharmacogenetics and genomics*, 22(9):692–694, 2012.

[89] John Forrest, Ted Ralphs, Stefan Vigerske, LouHafer, Bjarni Kristjansson, jpfasano, EdwinStraver, Miles Lubin, Haroldo Gambini Santos, rlougee, and Matthew Saltzman. coin-or/cbc: Version 2.9.9, July 2018.

[90] Michael Ford, Ananth Hari, Oscar Rodriguez, Junyan Xu, Justin Lack, Cihan Oguz, Yu Zhang, Sarah Weber, Mary Magliocco, Jason Barnett, et al. Immunotyper-sr: A novel computational approach for genotyping immunoglobulin heavy chain variable genes using short read data. In *International Conference on Research in Computational Molecular Biology*, pages 382–384. Springer, 2022.

[91] Victoria M. Pratt, Robin E. Everts, Praful Aggarwal, Brittany N. Beyer, Ulrich Broeckel, Ruth Epstein-Baak, Paul Hujsak, Ruth Kornreich, Jun Liao, Rachel Lorier, Stuart A. Scott, Chingying Huang Smith, Lorraine H. Toji, Amy Turner, and Lisa V. Kalman. Characterization of 137 Genomic DNA Reference Materials for 28 Pharmacogenetic Genes: A GeT-RM Collaborative Project. *The Journal of molecular diagnostics: JMD*, 18(1):109–123, January 2016.

[92] Andrea Gaedigk, Amy Turner, Robin E Everts, Stuart A Scott, Praful Aggarwal, Ulrich Broeckel, Gwendolyn A McMillin, Roberta Melis, Erin C Boone, Victoria M Pratt, et al. Characterization of reference materials for genetic testing of CYP2D6 alleles: a GeT-RM collaborative project. *The Journal of Molecular Diagnostics*, 21(6):1034–1052, 2019.

[93] Daniel Portik, Ting Hon, Josiah Wilcots, Nina Gonzaludo, Yao Yang, Nathan A. Hammond, Zev Kronenberg, Nathaniel Watson, John Harting, Euan Ashley, Janet Ziegle, Stuart A. Scott, and Sarah Kingan. Abstract: Development and Optimization of a 43 Gene Pharmacogenomic Panel Using Enrichment-Based Capture and PacBio HiFi Sequencing. In *ASHG 2021 Virtual Conference*, 2021.

[94] S Kingan, J Harting, T Hon, YC Tsai, I McLaughlin, J Ziegle, T Han, L Arbiza, S Kloet, L Busscher, G Henno, E Lee, and N Gonzaludo. Poster: Enablement of long-read targeted sequencing panels using Twist hybrid capture and PacBio HiFi sequencing. In *ESHG 2022*, 2022.

[95] Stuart A Scott, Erick R Scott, Yoshinori Seki, Annette J Chen, Richard Wallsten, Aniwaa Owusu Obeng, Mariana R Botton, Neal Cody, Huanzhi Shi, Geping Zhao, et al. Development and analytical validation of a 29 gene clinical pharmacogenetic genotyping panel: Multi-ethnic allele and copy number variant detection. *Clinical and translational science*, 14(1):204–213, 2021.

[96] Ariya Shajii, Ibrahim Numanagic, and Bonnie Berger. Latent variable model for aligning barcoded short-reads improves downstream analyses. In *RECOMB*, pages 280–282. Springer, 2018.

[97] Mary V Relling and William E Evans. Pharmacogenomics in the clinic. *Nature*, 526(7573):343–350, 2015.

[98] Michael Ford, Ehsan Haghshenas, Corey T Watson, and S Cenk Sahinalp. Genotyping and copy number analysis of immunoglobin heavy chain variable genes using long reads. *Iscience*, 23(3):100883, 2020.

[99] Kristine R Crews, Andrea Gaedigk, Henry M Dunnenberger, J Steve Leeder, Teri E Klein, Kelly E Caudle, Cyrine E Haidar, Danny D Shen, John T Callaghan, Senthilkumar Sadhasivam, et al. Clinical Pharmacogenetics Implementation Consortium guidelines for cytochrome P450 2D6 genotype and codeine therapy: 2014 update. *Clinical Pharmacology & Therapeutics*, 95(4):376–382, 2014.

[100] Andrea Gaedigk, Erin C Boone, Steven E Scherer, Seung-been Lee, Ibrahim Numanagić, Cenk Sahinalp, Joshua D Smith, Sean McGee, Aparna Radhakrishnan, Xiang Qin, et al. CYP2C8, CYP2C9, and CYP2C19 characterization using next-generation sequencing and haplotype analysis: A get-rm collaborative project. *The Journal of Molecular Diagnostics*, 2022.

[101] Michael Karin. *Gene expression: general and cell-type-specific*. Springer Science & Business Media, 2013.

[102] Aimée M Deaton and Adrian Bird. Cpg islands and the regulation of transcription. *Genes & development*, 25(10):1010–1022, 2011.

[103] Geneviève P Delcuve, Mojgan Rastegar, and James R Davie. Epigenetic control. *Journal of cellular physiology*, 219(2):243–250, 2009.

[104] Christopher G. Hubert and Justin D. Lathia. Seeing the GBM diversity spectrum. *Nature Cancer*, 2(2):135–137, 2021.

[105] Quinn T. Ostrom, Nirav Patil, Gino Cioffi, Kristin Waite, Carol Kruchko, and Jill S. Barnholtz-Sloan. CBTRUS statistical report: primary brain and other central nervous system tumors diagnosed in the United States in 2013–2017. *Neuro-Oncology*, 22(Supplement_1):iv1–iv96, 2020.

[106] David J. Cote, Naci Balak, Jannick Brennum, Daniel T. Holsgrove, Neil Kitchen, Herbert Kolenda, Wouter A. Mojjen, Karl Schaller, Pierre A. Robe, Tiit Mathiesen, et al. Ethical difficulties in the innovative surgical treatment of patients with recurrent glioblastoma multiforme. *Journal of Neurosurgery*, 126(6):2045–2050, 2017.

[107] Andrea Sottoriva, Immaculada Spiteri, Sara G. M. Piccirillo, Anestis Touloumis, V. Peter Collins, John C. Marioni, Christina Curtis, Colin Watts, and Simon Tavaré'. Intratumor heterogeneity in human glioblastoma reflects cancer evolutionary dynamics. *Proceedings of the National Academy of Sciences USA*, 110(10):4009–4014, 2013.

[108] Anoop P. Patel, Itay Tirosh, John J. Trombetta, Alex K. Shalek, Shawn M. Gillespie, Hiroaki Wakimoto, Daniel P. Cahill, Brian V. Nahed, William T. Curry, Robert L. Martuza, et al. Single-cell rna-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science*, 344(6190):1396–1401, 2014.

[109] Laura M. Richards, Owen K.N. Whitley, Graham MacLeod, Florence MG Cavalli, Fiona J. Coutinho, Julia E. Jaramillo, Nataliia Svergun, Mazdak Riverin, Danielle C. Croucher, Michelle Kushida, et al. Gradient of developmental and injury response transcriptional states defines functional vulnerabilities underpinning glioblastoma heterogeneity. *Nature Cancer*, 2(2):157–173, 2021.

[110] Pei Zhang, Qin Xia, Liqun Liu, Shouwei Li, and Lei Dong. Current opinion on molecular characterization for GBM classification in guiding clinical diagnosis, prognosis, and therapy. *Frontiers in Molecular Biosciences*, 7:562798, 2020.

[111] Roel G. W. Verhaak, Katherine A. Hoadley, Elizabeth Purdom, Victoria Wang, Yuan Qi, Matthew .D Wilkerson, C. Ryan Miller, Li Ding, Todd Golub, Jill P. Mesirov, et al. Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in *PDGFRA*, *IDH1*, *EGFR*, and *NF1*. *Cancer Cell*, 17(1):98–110, 2010.

[112] Cyril Neftel, Julie Laffy, Mariella G. Filbin, Toshiro Hara, Marni E. Shore, Gilbert J. Rahme, Alyssa R. Richman, Dana Silverbush, McKenzie L. Shaw, Christine M. Hebert, et al. An integrative model of cellular states, plasticity, and genetics for glioblastoma. *Cell*, 178(4):835–849, 2019.

[113] C. Houllier, X. Wang, G. Kaloshi, K. Mokhtari, R. Gillevin, J. Laffaire, S. Paris, B. Boisselier, A. Idbaih, F. Laigle-Donadey, K. Hoang-Xuan, M. Sanson, and J.-Y. Delattre. *IDH1* or *IDH2* mutations predict longer survival and response to temozolomide in low-grade gliomas. *Neurology*, 75(17):1560–1565, 2010.

[114] Rahul Kumar, Anthony P. Y. Liu, Brent A. Orr, Paul A. Northcott, and Robinson Giles W. Advances in the classification of pediatric brain tumors through DNA methylation profiling: From research tool to frontline diagnostic. *Cancer*, 124(21):4168–4180, 2018.

[115] Christian Koelsche and Andreas von Deimling. Methylation classifiers: Brain tumors, sarcomas, and what's next . *Genes, Chromosomes & Cancer*, 61(6):346–355, 2022.

[116] Dominik Sturm, Hendrik Witt, Volker Hovestadt, Dong-Anh Khuong-Quang, David T. W. Jones, Carolin Konermann, Elke Pfaff, Martje Tönjes, Martin Sill, Sebastian Bender, et al. Hotspot mutations in *H3F3A* and *IDH1* define distinct epigenetic and biological subgroups of glioblastoma. *Cancer Cell*, 22(4):425–437, 2012.

[117] Richard Drexler, Ulrich Schüller, Alicia Eckhardt, Katharina Filipski, Tabea I. Hartung, Patrick N. Harter, Iris Divé, Marie-Therese Forster, Marcus Czabanka, Claudius Jelgersma, et al. DNA methylation subclasses predict the benefit from gross total tumor resection in IDH-wildtype glioblastoma patients. *Neuro-Oncology*, 25(2):315–325, 2023.

[118] Martha Foltyn-Dumitru, Haidar Alzaid, Aditya Rastogi, Ulf Neuberger, Felix Sahm, Tobias Kessler, Wolfgang Wick, Martin Bendszus, Philipp Vollmuth, and Marianne Schell. Unraveling glioblastoma diversity: Insights into methylation subtypes and spatial relationships. *Neuro-Oncology Advances*, 6(1):vdae112, 2024.

[119] Franz L. Ricklefs, Richard Drexler, Kathrin Wollmann, Alicia Eckhardt, Dieter H. Heiland, Thomas Sauvigny, Cecile Maire, Katrin Lamszus, Manfred Westphal, Ulrich Schüller, et al. DNA methylation subclass receptor tyrosine kinase II (RTK II) is predictive for seizure development in glioblastoma patients. *Neuro-Oncology*, 24(11):1886–1897, 2022.

[120] Zhihong Chen and Dolores Hambardzumyan. Immune microenvironment in glioblastoma subtypes. *Frontiers in Immunology*, 9:1004, 2018.

[121] David Capper, David T. W. Jones, Martin Sill, Volker Hovestadt, Daniel Schrimpf, Dominik Sturm, Christian Koelsche, Felix Sahm, Lukas Chavez, David E. Reuss, Annekathrin Kratz, Annika K. Wefers, Kristin Huang, Kristian W. Pajtler, Leonille Schweizer, Damian Stichel, Adriana Olar, Nils W. Engel, Kerstin Lindenberg, Patrick N. Harter, Anne K. Braczynski, Karl H. Plate, Hildegard Dohmen, Boyan K. Garvalov, Roland Coras, Annett Hölsken, Ekkehard Hewer, Melanie Bewerunge-Hudler, Matthias Schick, Roger Fischer, Rudi Beschorner, Jens Schittenhelm, Ori Staszewski, Khalida Wani, Pascale Varlet, Melanie Pages, Petra Temming, Dietmar Lohmann, Florian Selt, Hendrik Witt, Till Milde, Olaf Witt, Eleonora Aronica, Felice Giangaspero, Elisabeth Rushing, Wolfram Scheurlen, Christoph Geisenberger, Fausto J. Rodriguez, Albert Becker, Matthias Preusser, Christine Haberler, Rolf Bjerkvig, Jane Cryan, Michael Farrell, Martina Deckert, Jürgen Hench, Stephan Frank, Jonathan Serrano, Kasthuri Kannan, Aristotelis Tsirigos, Wolfgang Brück, Silvia Hofer, Stefanie Brehmer, Marcel Seiz-Rosenhagen, Daniel Hänggi, Volkmar Hans, Stephanie Rozsnoki, Jordan R. Hansford, Patricia Kohlhof, Bjarne W. Kristensen, Matt Lechner, Beatriz Lopes, Christian Mawrin, Ralf Ketter, Andreas Kulozik, Ziad Khatib, Frank Heppner, Arend Koch, Anne Jouvet, Catherine Keohane, Helmut Mühleisen, Wolf Mueller, Ute Pohl, Marco Prinz, Axel Benner, Marc Zapatka, Nicholas G. Gottardo, Pablo Hernáiz Driever, Christof M. Kramm, Hermann L. Müller, Stefan Rutkowski, Katja von Hoff, Michael C. Frühwald, Astrid Gnekow, Gudrun Fleischhack, Stephan Tippelt, Gabriele Calaminus, Camelia-Maria Monoranu, Arie Perry, Chris Jones, Thomas S. Jacques, Bernhard Radlwimmer, Marco Gessi, Torsten Pietsch, Johannes Schramm, Gabriele Schackert, Manfred Westphal, Guido Reifenberger, Pieter

Wesseling, Michael Weller, Vincent Peter Collins, Ingmar Blümcke, Martin Bendszus, Jürgen Debus, Annie Huang, Nada Jabado, Paul A. Northcott, Werner Paulus, Amar Gajjar, Giles W. Robinson, Michael D. Taylor, Zane Jaunmuktane, Marina Ryzhova, Michael Platten, Andreas Unterberg, Wolfgang Wick, Matthias A. Karajannis, Michel Mittelbronn, Till Acker, Christian Hartmann, Kenneth Aldape, Ulrich Schüller, Rolf Buslei, Peter Lichter, Marcel Kool, Christel Herold-Mende, David W. Ellison, Martin Hasselblatt, Matija Snuderl, Sebastian Brandner, Andrey Korshunov, Andreas von Deimling, and Stefan M. Pfister. DNA methylation-based classification of central nervous system tumours. *Nature*, 555(7697):469–474, 2018.

[122] Andrew S Venteicher, Itay Tirosh, Christine Hebert, Keren Yizhak, Cyril Neftel, Mariella G. Filbin, Volker Hovestadt, Leah E Escalante, McKenzie L. Shaw, Christopher Rodman, et al. Decoupling genetics, lineages, and microenvironment in idh-mutant gliomas by single-cell RNA-seq. *Science*, 355(6332):eaai8478, 2017.

[123] Qinjie Weng, Jincheng Wang, Jiajia Wang, Danyang He, Zuolin Cheng, Feng Zhang, Ravinder Verma, Lingli Xu, Xinrang Dong, Yunfei Liao, et al. Single-cell methylation sequencing data reveal succinct metastatic migration histories and tumor progression models. *Cell Stem Cell*, 24(5):707–723, 2019.

[124] Charles P. Couturier, Shamini Ayyadhury, Phuong U. Le, Javad Nadaf, Jean Monlong, Gabriele Riva, Redouane Allache, Salma Baig, Xiaohua Yan, Mathieu Bourgey, et al. Single-cell rna-seq reveals that glioblastoma recapitulates a normal neurodevelopmental hierarchy. *Nature Communications*, 11:3406, 2020.

[125] Aaron M. Newman, Chih Long Liu, Michael R. Green, Andrew J. Gentles, Weiguo Feng, Yue Xu, Chuong D. Hoang, Maximilian Diehn, and Ash A. Alizadeh. Robust enumeration of cell subsets from tissue expression profiles. *Nature Methods*, 12:453–457, 2015.

[126] Etienne Becht, Nicolas A. Giraldo, Laetitia Lacroix, Bénédicte Buttard, Nabila Elarouci, Florent Petitprez, Janick Selves, Pierre Laurent-Puig, Catherine Sautès-Fridman, Wolf H. Fridman, et al. Estimating the population abundance of tissue-infiltrating immune and stromal cell populations using gene expression. *Genome Biology*, 17:218, 2016.

[127] Dvir Aran, Zicheng Hu, and Atul J. Butte. xCell: digitally portraying the tissue cellular heterogeneity landscape. *Genome Biology*, 18:220, 2017.

[128] Francesca Finotello, Clemens Mayer, Christina Plattner, Gerhard Laschober, Dietmar Rieder, Hubert Hackl, Anne Krogsdam, Zuzana Loncova, Wilfried Posch, Doris Wilflingseder, et al. Molecular and pharmacological modulators of the tumor immune contexture revealed by deconvolution of rna-seq data. *Genome Medicine*, 11:34, 2019.

[129] Aaron M. Newman, Chloé B. Steen, Chih Long Liu, Andrew J. Gentles, Aadel A. Chaudhuri, Florian Scherer, Michael S. Khodadoust, Mohammad S. Esfahani, Bogdan A. Luca, David Steiner, et al. Determining cell type abundance and expression from bulk tissues with digital cytometry. *Nature Biotechnology*, 37(7):773–782, 2019.

[130] Kun Wang, Sushant Patkar, Joo Sang Lee, E. Michael Gertz, Welles Robinson, Fiorella Schischlik, David R. Crawford, Alejandro A. Schäffer, and Eytan Ruppin. Deconvolving clinically relevant cellular immune cross-talk from bulk gene expression using CODEFACS and LIRICS stratifies patients with melanoma to anti-pd-1 therapy. *Cancer Discovery*, 12(4):1088–1105, 2022.

[131] Aleksandr Zaitsev, Maksim Chelushkin, Daniiar Dyikanov, Ilya Cheremushkin, Boris Shpak, Krystle Nomie, Vladimir Zyrin, Ekaterina Nuzhdina, Yaroslav Lozinsky, Anastasia Zotova, et al. Precise reconstruction of the tme using bulk rna-seq and a machine learning algorithm trained on artificial transcriptomes. *Cancer Cell*, 40(8):879–894, 2022.

[132] Maísa R. Ferro dos Santos, Edoardo Giuili, Andries De Koker, Celine Everaert, and Katleen De Preter. Computational deconvolution of DNA methylation data from mixed DNA samples. *Briefings in Bioinformatics*, 25(3):bbae234, 2024.

[133] Eugene Andres Houseman, William P. Accomando, Devin C. Koestler, Brock C. Christensen, Carmen J. Marsit, Heather H. Nelson, John K. Wiencke, and Karl T. Kelsey. DNA methylation arrays as surrogate measures of cell mixture distribution. *BMC Bioinformatics*, 13:86, 2012.

[134] Joshua Moss, Judith Magenheim, Daniel Neiman, Hai Zemmour, Netanel Loyfer, Amit Korach, Yaacov Samet, Myriam Maoz, Henrik Druid, Peter Arner, et al. Comprehensive human cell-type methylation atlas reveals origins of circulating cell-free DNA in health and disease. *Nature Communications*, 9:5068, 2018.

[135] Douglas Arneson, Xia Yang, and Kai Wang. MethylResolver—a method for deconvoluting bulk DNA methylation profiles into known and unknown cell contents. *Communications Biology*, 3:422, 2020.

[136] Ankur Chakravarthy, Andrew Furness, Kroopa Joshi, Ehsan Ghorani, Kirsty Ford, Matthew J. Ward, Emma V. King, Matt Lechner, Teresa Marafioti, Sergio A. Quezada, et al. Pan-cancer deconvolution of tumour composition using DNA methylation. *Nature Communications*, 9:3220, 2018.

[137] Shuo Li, Weihua Zeng, Xiaohui Ni, Qiao Liu, Wenyuan Li, Mary L Stackpole, Yonggang Zhou, Arjan Gower, Kostyantyn Krysan, Preeti Ahuja, et al. Comprehensive tissue deconvolution of cell-free DNA by deep learning for disease diagnosis and monitoring. *Proceedings of the National Academy of Sciences USA*, 120(28):e2305236120, 2023.

[138] Pavlo Lutsik, Martin Slawski, Gilles Gasparoni, Nikita Vedeneev, Matthias Hein, and Jörn Walter. MeDeCom: discovery and quantification of latent components of heterogeneous methylomes. *Genome Biology*, 18:55, 2017.

[139] Vitor Onuchic, Ryan J. Hartmaier, David N. Boone, Michael L. Samuels, Ronak Y. Patel, Wendy M. White, Vesna D. Garovic, Steffi Oesterreich, Matt E. Roth, Adrian V. Lee, et al. Epigenomic deconvolution of breast tumors reveals metabolic coupling between constituent cell types. *Cell Reports*, 17(8):2075–2086, 2016.

[140] Xiaoqi Zheng, Qian Zhao, Hua-Jun Wu, Wei Li, Haiyun Wang, Clifford A. Meyer, Qian Alvin Qin, Han Xu, Chongzhi Zang, Peng Jiang, et al. MethylPurify: tumor purity deconvolution and differential methylation detection from single tumor DNA methylomes. *Genome Biology*, 15:419, 2014.

[141] Dohoon Lee, Sangseon Lee, and Sun Kim. PRISM: methylation pattern-based, reference-free inference of subclonal makeup. *Bioinformatics*, 35(14):i520–i529, 2019.

[142] Christa Caggiano, Barbara Celona, Fleur Garton, Joel Mefford, Brian L. Black, Robert Henderson, Catherine Lomen-Hoerth, Andrew Dahl, and Noah Zaitlen. Comprehensive cell type decomposition of circulating cell-free DNA with CelFiE. *Nature Communications*, 12:2717, 2021.

[143] Dingqin He, Ming Chen, Wenjuan Wang, Chunhui Song, and Yufang Qin. Deconvolution of tumor composition using partially available DNA methylation data. *BMC Bioinformatics*, 23:355, 2022.

[144] Richard Drexler, Robin Khatri, Thomas Sauvigny, Malte Mohme, Cecile L. Maire, Alice Ryba, Yahya Zghaibeh, Lasse Dührsen, Amanda Salviano-Silva, Katrin Lamszus, et al. A prognostic neural epigenetic signature in high-grade glioma. *Nature Medicine*, 30(6):1622–1635, 2024.

[145] Dana Silverbush, Mario Suva, and Volker Hovestadt. LTBK-08. Inferring cell type and cell state composition in glioblastoma from bulk DNA methylation profiles using multi-omic single-cell analyses. *Neuro-Oncology*, 24(Supplement_7):vii300–vii300, 2022.

[146] Anna Wenger, Sandra Ferreyra Vega, Teresia Kling, Thomas Olsson Bontell, Asgeir Store Jakola, and Helena Carén. Intratumor DNA methylation heterogeneity in glioblastoma: implications for DNA methylation-based classification. *Neuro-Oncology*, 21(5):616–627, 2019.

[147] Wei Tian, Jingtian Zhou, Anna Bartlett, Qiurui Zeng, Hanqing Liu, Rosa G. Castanon, Mia Kenworthy, Jordan Altshul, Cynthia Valadon, Andrew Aldridge, et al. Single-cell DNA methylation and 3D genome architecture in the human brain. *Science*, 382(6667):eadf5357, 2023.

[148] Ayushi S. Patel and Itai Yanai. A developmental constraint model of cancer cell states and tumor heterogeneity. *Cell*, 187(12):2907–2918, 2024.

[149] Lanqi Gong, Jie Luo, Emily Yang, Beibei Ru, Ziyang Qi, Yuma Yang, Anshu Rani, Abhilasha Purohit, Yu Zhang, Grace Guan, et al. Cancer immunology data engine reveals secreted AOAH as a potential immunotherapy. *Cell*, 188(18):5062–5080, 2025.

[150] Silvia Domcke and Jay Shendure. A reference cell tree will serve science better than a reference cell atlas. *Cell*, 186(6):P1103–1114, 2023.

[151] Hui Zong, Luis F. Parada, and Suzanne J. Baker. Cell of origin for malignant gliomas and its implication in therapeutic development. *Cold Spring Harbor Perspectives in Biology*, 7(5):a020610, 2015.

[152] Calum Gabutt, Ryan O. Schenck, Daniel J. Weisenberger, Christopher Kimberley, Alison Berner, Jacob Househam, Eszter Lakatos, Mark Robertson-Tessi, Isabel Martin, Roshani Patel, et al. Fluctuating methylation clocks for cell lineage tracing at high temporal resolution in human tissues. *Nature Biotechnology*, 40(5):720–730, 2022.

[153] Peter C. Nowell. The clonal evolution of tumor cell populations. *Science*, 194(4260):23–28, 1976.

[154] Salem Malikic, Andrew W. McPherson, Nilgun Donmez, and S. Cenk Sahinalp. Clonality inference in multiple tumor samples using phylogeny. *Bioinformatics*, 31(9):1349–1356, 2015.

[155] Niko Beerenwinkel, Rolad F. Schwartz, Morits Gerstung, and Florian Markowetz. Cancer evolution: mathematical models and computational inference. *Systematic Biology*, 64(1):e1–25, 2015.

[156] Russell Schwartz and Alejandro A. Schäffer. The evolution of tumour phylogenetics: principles and practice. *Nature Reviews Genetics*, 18(4):213–229, 2017.

[157] Lin Li, Wenqin Xie, Li Zhan, Shaodi Wen, Xiao Luo, Shuangbin Xu, Yantong Cai, Wenli Tang, Qianwen Wang, and Ming Li. Resolving tumor evolution: a phylogenetic approach. *Journal of the National Cencer Center*, 4(2):97–106, 2024.

[158] Shuhui Bian, Yu Hou, Xin Zhou, Xianlong Li, Jun Yong, Yicheng Wang, Wendong Wang, Jia Yan, Boqiang Hu, Hongshan Guo, Jilian Wang, Shuai Gao, Yunuo Mao, Ji Dong, Ping Zhu, Dianrong Xiu, Liying Yan, Lu Wen, Jie Qiao, Fuchou Tang, and Wei Fu. Single-cell multiomics sequencing and analyses of human colorectal cancer. *Science*, 362(6418):1060–1063, 2018.

[159] George D. Cresswell, Daniel Nichol, Inmaculada Spiteri, Haider Tari, Luis Zapata, Timon Heide, Carlo C. Maley, Luca Magnani, Gaia Schiavon, Alan Ashworth, et al. Mapping the breast cancer metastatic cascade onto ctDNA using genetic and epigenetic clonal tracking. *Nature Communications*, 11:1446, 2020.

[160] Yuelin Liu, Xuan Cindy Li, Farid Rashidi Mehrabadi, Alejandro A. Schäffer, Drew Pratt, David R. Crawford, Salem Malikic, Erin K. Molloy, Vishaka Gopalan, Stephen M. Mount, Eytan Ruppin, Kenneth D. Aldape, and S. Cenk Sahinalp. Single-cell methylation sequencing data reveal succinct metastatic migration histories and tumor progression models. *Genome Research*, 33(7):1089–1100, 2023.

[161] Calum Gabutt, Martí Duran-Ferrer, Heather E. Grant, Diego Mallo, Ferran Nadeu, Jacob Househam, Neus Villamor, Madlen Müller, Simon Heath, Emanuele Raineri, et al. Fluctuating dna methylation tracks cancer evolution at clinical scalem. *Nature*, page in press, 2025.

[162] Rafael A Irizarry, Christine Ladd-Acosta, Bo Wen, Zhijin Wu, Carolina Montano, Patrick Onyango, Hengmi Cui, Kevin Gabo, Michael Rongione, Maree Webster, Hong Ji, James B Potash, Sarven Sabunciyan, and Andrew P Feinberg. The human colon cancer methylome

shows similar hypo- and hypermethylation at conserved tissue-specific cpg island shores. *Nature Genetics*, 41(2):178–186, 2009.

[163] N. Saitou and M. Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, 4(4):406–425, 07 1987.

[164] J. O. McClain and V. R. Rao. Clustisz: A program to test for the quality of clustering of a set of objects. *Journal of Marketing Research*, 12(4):456–460, 1975.

[165] Peter J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[166] Alexander D. Diehl, Terrence F. Meehan, Yvonne M. Bradford, Matthew H. Brush, Wasila M. Dahdul, David S. Dougall, Yongqun He, David Osumi-Sutherland, Alan Ruttenberg, Sirarat Sarntivijai, et al. The Cell Ontology 2016: enhanced content, modularization, and ontology interoperability. *Journal of Biomedical Semantics*, 7:44, 2016.

[167] Terrence F. Meehan, Anna Maria Masci, Amina Abdulla, Lindsay G. Cowell, Judith A. Blake, Christopher J. Mungall, and Alexander D. Diehl. Logical development of the cell ontology. *BMC Bioinformatics*, 12:6, 2011.

[168] Jonathan Bard, Seung Y. Rhee, and Michael Ashburner. An ontology for cell types. *Genome Biology*, 6:R21, 2005.

[169] Daniel S Auld and Richard Robitaille. Glial cells and neurotransmission: an inclusive view of synaptic function. *Neuron*, 40(2):389–400, 2003.

[170] Denis Soulet and Serge Rivest. Microglia. *Current Biology*, 18(12):R506–R508, 2008.

[171] Andrey Nikolaevich Kolmogorov. Sulla determinazione empirica di una legge didistribuzione. *Giorn Dell'inst Ital Degli Att*, 4:89–91, 1933.

[172] Richard Simard and Pierre L'Ecuyer. Computing the two-sided Kolmogorov-Smirnov distribution. *Journal of Statistical Software*, 39(11):1–18, 2011.

[173] Yuji Matsumoto, Omkar Singh, Jose Garcia, Zied Abdullaev, Nelson F. Freeburg, Fanyang Yu, Hamed Akbari, Kyunglok Baik, Jun Guo, Natali N. C. Shih, et al. Intratumoral preservation of the stem-like malignant cell proportion in glioblastoma, prognostic impact, and its radiomethylomic signatures. *Nero-Oncology*, in press:noaf175, 2025.

[174] Zihao Wang, Yaning Wang, Tianrui Yang, Hao Xing, Yuekun Wang, Lu Gao, Xiaopeng Guo, Bing Xing, Yu Wang, and Wenbin Ma. Machine learning revealed stemness features and a novel stemness-based classification with appealing implications in discriminating the prognosis, immunotherapy and temozolomide responses of 906 glioblastoma patients. *Briefings in Bioinformatics*, 22(5):bbab032, 2021.

[175] Hai Yan, William Parsons, Genglin Jin, Roger McLendon, Ahmed Rasheed, Weishi Yuan, Ivan Kos, and Darrel D. Bigner. *IDH1* and *IDH2* mutations in gliomas. *New England Journal of Medicine*, 360(8):765–773, 2009.

[176] Houtan Nousmehr, Daniel J. Weisenberger, Kristin Diefes, Phillips Heidi S., Kanan Pujara, Benjamin P. Berman, Fei Pan, Christopher E. Pelloski, Erik P. Sulman, Krishna P. Bhat, et al. Identification of a CpG island methylator phenotype that defines a distinct subgroup of glioma. *Cancer Cell*, 17(5):510–522, 2010.

[177] Sevin Turcan, Daniel Rohle, Anuj Goenka, Logan A. Walsh, Fang Fang, Emrullah Yilmaz, Carl Campos, Armida W. M. Fabius, Chao Lu, Patrick S. Ward, et al. *IDH1* mutation is sufficient to establish the glioma hypermethylator phenotype. *Nature*, 483(7390):479–483, 2012.

[178] Dusten Unruh, Makda Zewde, Adam Buss, Michael R. Drumm, Anh N. Tran, Denise M. Scholtens, and Craig Horbinski. Methylation and transcription patterns are distinct in IDH mutant gliomas compared to other IDH mutant cancers. *Scientific Reports*, 9:8946, 2019.

[179] Gilbert J Rahme, Nauman M Javed, Kaitlyn L Puorro, Shouhui Xin, Volker Hovestadt, Sarah E Johnstone, and Bradley E Bernstein. Modeling epigenetic lesions that cause gliomas. *Cell*, 186(17):3674–3685, 2023.

[180] Florian Rubelt, Christian E Busse, Syed Ahmad Chan Bukhari, Jean-Philippe Bürckert, Encarnita Mariotti-Ferrandiz, Lindsay G Cowell, Corey T Watson, Nishanth Marthandan, William J Faison, Uri Hershberg, et al. Adaptive immune receptor repertoire community recommendations for sharing immune-repertoire sequencing data. *Nature immunology*, 18(12):1274–1278, 2017.

[181] Ulf Klein and Riccardo Dalla-Favera. Germinal centres: role in b-cell physiology and malignancy. *Nature Reviews Immunology*, 8(1):22–33, 2008.

[182] Cinque Soto, Robin G Bombardi, Andre Branchizio, Nurgun Kose, Pranathi Matta, Alexander M Sevy, Robert S Sinkovits, Pavlo Gilchuk, Jessica A Finn, and James E Crowe Jr. High frequency of shared clonotypes in human b cell receptor repertoires. *Nature*, 566(7744):398–402, 2019.

[183] Bing He, Shuning Liu, Mengxin Xu, Yunqi Hu, Kexin Lv, Yuanyuan Wang, Yong Ma, Yanmei Zhai, Xinyu Yue, Lin Liu, et al. Comparative global b cell receptor repertoire difference induced by sars-cov-2 infection or vaccination via single-cell v (d) j sequencing. *Emerging Microbes & Infections*, 11(1):2007–2020, 2022.

[184] Li Song, David Cohen, Zhangyi Ouyang, Yang Cao, Xihao Hu, and X Shirley Liu. Trust4: immune repertoire reconstruction from bulk and single-cell rna-seq data. *Nature methods*, 18(6):627–630, 2021.

[185] Kenneth B Hoehn and Steven H Kleinstein. B cell phylogenetics in the single cell era. *Trends in Immunology*, 2024.

[186] Kenneth B Hoehn, Gerton Lunter, and Oliver G Pybus. A phylogenetic codon substitution model for antibody lineages. *Genetics*, 206(1):417–427, 2017.

[187] William S DeWitt III, Luka Mesin, Gabriel D Victora, Vladimir N Minin, and Frederick A Matsen IV. Using genotype abundance to improve phylogenetic inference. *Molecular biology and evolution*, 35(5):1253–1265, 2018.

[188] Kristian Davidsen and Frederick A Matsen IV. Benchmarking tree and ancestral sequence inference for b cell receptor sequences. *Frontiers in immunology*, 9:2451, 2018.

[189] Leah L Weber, Derek Reiman, Mrinmoy S Roddur, Yuanyuan L Qi, Mohammed El-Kebir, and Aly A Khan. Tribal: Tree inference of b cell clonal lineages. *bioRxiv*, pages 2023–11, 2023.

[190] J Felsenstein. Phylip: Phylogenetic inference program, version 3.6. *University of Washington, Seattle*, 2005.

[191] Bui Quang Minh, Heiko A Schmidt, Olga Chernomor, Dominik Schrempf, Michael D Woodhams, Arndt Von Haeseler, and Robert Lanfear. Iq-tree 2: new models and efficient methods for phylogenetic inference in the genomic era. *Molecular biology and evolution*, 37(5):1530–1534, 2020.

[192] Les R Foulds and Ronald L Graham. The steiner problem in phylogeny is np-complete. *Advances in Applied mathematics*, 3(1):43–49, 1982.

[193] Salem Malikic, Farid Rashidi Mehrabadi, Simone Ciccolella, Md Khaledur Rahman, Camir Ricketts, Ehsan Haghshenas, Daniel Seidman, Faraz Hach, Iman Hajirasouliha, and S Cenk Sahinalp. Phiscs: a combinatorial approach for subperfect tumor phylogeny reconstruction via integrative use of single-cell and bulk sequencing data. *Genome research*, 29(11):1860–1877, 2019.