Computational Techniques for Analyzing Tumor DNA Data

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Sean R. Landman

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Adviser: Vipin Kumar
Co-Adviser: Michael Steinbach

June, 2016

Acknowledgments

It's been a long road to reach this point, and throughout the course of this journey I've realized how fortunate I am to be surrounded by an abundance of supportive and influential people in my life. I wouldn't be where I am today, and this dissertation wouldn't have been possible, without the guidance, advise, support, encouragement, and friendship of so many people that I would like to thank.

First and foremost, I would like to thank my adviser, Vipin Kumar. Your positivity and enthusiasm for research has been a constant source of encouragement throughout my time in graduate school. You've given me the freedom to explore my own research ideas, pushed me to take on new challenges, and have always given me the support I've needed to succeed.

I would also like to express my gratitude to my co-adviser, Michael Steinbach. Thank you for all of the countless times you've helped me by discussing ideas and working through problems together. Your insights have been so influential in helping me grow as a researcher.

Imad Rahal, my adviser during my time at St. John's University, deserves special recognition for helping to send me along this career path I have chosen. You introduced me to research and to bioinformatics, encouraged me to pursue graduate school, and have been a mentor and friend throughout the years. To my other committee members, Rui Kuang, Scott Dehm, and Kevin Silverstein, thank you all for your advise and support throughout this entire process. I would also like to thank Robert Stadler, Hongfang Liu, and Kavishwar Wagholikar, who provided me with guidance and mentorship during my various internships.

The nature of my research work has made it heavily reliant on interdisciplinary collaborations. As such, none of this would have been possible without a tremendous

network of collaborators and co-authors. In particular, I would like to thank Tae Hyun Hwang, Tim Starr, Jonathan Rawson, Lou Mansky, and Bonnie Westra for being such excellent collaborators over the years. I have also been fortunate to have had fantastic funding sources throughout my PhD program, including the National Institute of Health, the National Science Foundation, and the University of Minnesota.

I am grateful for all of my fellow lab mates, past and present, in the Data Mining for Biomedical Informatics Research Group, including Gang Fang, Gowtham Atluri, Vanja Paunić, Sanjoy Dey, Wen Wang, Pranjul Yadav, Lisiane Pruinelli, Kevin Schiroo, Wonsuk Oh, Andrew Hangsleben, Katie Hauwiller, Jake Johnson, and Alex Hoff. Whether or not we worked on a project together, it was always a pleasure getting to share an academic home and to work alongside such a fun and supportive group of people.

Last but certainly not least, I would like to thank my family: my parents Jean and Kevin, my brother Eric, and my fiancé Sally. Sally, I couldn't have imagined these last four years without you by my side. Thank you for being my go-to source of support and encouragement. Mom and Dad, your love and constant support over all these years has made me the person I am today. I owe this all to you.

Abstract

Cancer has often been described as a disease of the genome, and understanding the underlying genetics of this complex disease opens the door to developing improved treatments and more accurate diagnoses. The abundant availability of next-generation DNA sequencing data in recent years has provided a tremendous opportunity to enhance our understanding of cancer genetics. Despite having this wealth of data available, analyzing tumor DNA data is complicated by issues such as genetic heterogeneity often found in tumor tissue samples, and the diverse and complex genetic landscape that is characteristic of tumors. Advanced computational analysis techniques are required in order to address these challenges and to deal with the enormous size and inherent complexity of tumor DNA data.

The focus of this thesis is to develop novel computational techniques to analyze tumor DNA data and address several ongoing challenges in the area of cancer genomics research. These techniques are organized into three main aims or focuses. The first focus is on developing algorithms to detect patterns of co-occurring mutations associated with tumor formation in insertional mutagenesis data. Such patterns can be used to enhance our understanding of cancer genetics, as well as to identify potential targets for therapy. The second focus is on assembling personal genomic sequences from tumor DNA. Personal genomic sequences can enhance the efficacy of downstream analyses that measure gene expression or regulation, especially for tumor cells. The final focus is on estimating variant frequencies from heterogeneous tumor tissue samples. Accounting for heterogeneous variants is essential when analyzing tumor samples, as they are often the cause of therapy resistance and tumor recurrence in cancer.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

The field of genomics research has entered into an era of incredible advancements and impactful discoveries over the last several decades. Much of this progress traces back to the successful completion of the Human Genome Project, which provided us with a human reference genome that could serve as a catalog of all the genes and nucleotides (i.e. the molecular building blocks of DNA) in a typical person [1, 2]. This ambitious project sought to drastically increase our understanding of heritable diseases and usher in an era of personalized medicine. Countless discoveries have been made in the thirteen years since the completion of the Human Genome Project, and this momentum has translated into multiple other large-scale international genomics research efforts, such as the 1000 Genomes Project [3, 4], the International HapMap Project [5], and the ENCODE (Encyclopedia of DNA Elements) project [6, 7, 8]. However, this new era of genomics research has also generated many puzzling new questions. The further we delve, the more we realize just how complex the landscape of human genetics actually is.

A prime example of this is in the field of cancer research. Cancerous tumors are brought on by mutations in our DNA that cause uncontrollable cell proliferation. New knowledge of the genetics involved in this process has led to breakthroughs in cancer therapy, but has also uncovered many new complexities and unsolved problems that require additional research work to address. For example, it is now understood that most types of cancer are not caused by certain specific mutations in specific genes, but rather by combinations of many rare mutations which require huge studies with many patients in order to fully catalog. The number of somatically-acquired mutations (i.e. mutations not present initially but that are acquired later in life) that can be found in a tumor varies drastically between different cancer types, and can vary drastically even between two people with the same type of cancer [9].

Additionally, we have learned that not only do different types of cancers have different genetic features, but there can be a variety of genetic features even within the same tumor. These groups of different cells within a tumor (called subpopulations or subclones) are each defined by their own set of mutations. This intra-tumor genetic heterogeneity makes it difficult to determine the key mutations that need to be studied for a particular tumor. Furthermore, it is also necessary to understand how these mutations affect gene expression (i.e. the frequency with which genes are "activated" and produce their corresponding proteins) if we want to fully understand the process by which these tumors grow. However, in the case of cancer research, it can be difficult to study gene expression due to the highly-mutated genomes found in tumors.

While these complexities have led to new avenues of research to pursue, new technology has also provided us with an abundance of data to help solve these problems. Next-generation sequencing is a method to collect DNA data that has become immensely popular over the last decade as its cost has plummeted and its efficiency has improved. Using this technology, DNA data is collected by randomly fragmenting the DNA molecules in a cell into billions of small segments for which the order of the nucleotides can be more easily, and more cheaply, determined. These small segments are called reads, and collectively this type of data is known as sequencing data. Because of the randomness of the segmentation and the inherent error rates of the sequencing process, sequencing has to be done at high coverage levels (i.e. collect an average of many random segments from the same regions of the genome) in order to reliably cover all of the regions of interest, resulting in sequencing data that can be hundreds of gigabytes in size for just a single sample. Thus, analyzing this data requires efficient algorithms and advanced computational techniques, which have tended to lag behind the exponential explosion of sequencing data being generated.

1.2 Thesis Contributions

The focus of this thesis is to develop novel computational techniques to analyze tumor DNA data and address several ongoing challenges in the area of cancer genomics research. This falls generally into three main aims, which are (1) mining for co-occurring genetic mutations in tumors, (2) assembling personal genomic sequences, and (3) estimating tumor heterogeneity.

The first area of emphasis is in developing methods to identify coordinating genetic mutations that may drive tumor growth. Insertional mutagenesis experiments are a technique often used to screen for potential oncogenes and other cancer drivers in laboratory mice. Analysis of insertional mutagenesis data sets has mostly focused on identifying single genes or genomic regions that are significantly associated with tumor formation, with a lack of emphasis on efficient and accurate techniques to identify higher order patterns of multiple cooperating mutations that may be relevant. The first part of this thesis focuses on applying a data mining methodology known as association analysis to this problem, with the goal of identifying frequently co-occurring insertion sites in these insertional mutagenesis data sets.

The second area of emphasis is in assembling personal genomic sequences using sequencing data from tumor cells. Accurate personal genomic sequences offer the ability to improve downstream next-generation sequencing analyses, such as RNA-seq and ChIP-seq, by providing a more accurate reference sequence tailored for an individual sample. This is particularly important for the highly-mutated DNA found in tumors. Toward this end, we have developed an open-source software package called SHEAR (Sample Heterogeneity Estimation and Assembly by Reference), which leverages structural variant (SV) prediction algorithms in order to improve assembly efficiency by focusing on assembling the areas of the genome that differ substantially from the reference.

Finally, the last area of emphasis in this thesis focuses on developing algorithms for dealing with heterogeneous sequencing samples from tumors. Tumors often contain a mixture of cellular subpopulations, each with different genetic backgrounds, that make analyzing DNA data from tumor tissue a more difficult process than that from normal tissue. There is currently an unmet need for algorithms that can identify SVs in a heterogeneous sequencing sample as well as estimate the frequency with which they are present in the sample. Another component of our SHEAR framework addresses this problem by utilizing alignment information at SV breakpoints (i.e. the locations in the genome at which DNA is rearranged) in order to estimate variant frequencies for SVs and other mutations.

1.3 Thesis Outline

The organization of the rest of this thesis is as follows. A broad overview of computational analysis techniques for tumor DNA data and the associated challenges are presented in Chapter 2. In Chapter 3 we present our approach for discovering patterns of co-occurring genetic mutations in insertional mutagenesis data sets. Chapter 4 focuses on SHEAR, particularly as it relates to personal genome assembly. Our approach for estimating allele frequencies for structural variants from heterogeneous sequencing samples is presented in Chapter 5. Finally, we conclude in Chapter 6 with a summary discussion and an overview of future work.

Chapter 2

Overview of Tumor DNA Data Analysis and Challenges

2.1 Introduction

In this chapter, we introduce much of the data and concepts that will be discussed throughout this thesis. Specifically, Section 2.2 will describe next-generation sequencing data and the various ways that it is can be analyzed for genomics research. Section 2.3 will then provide an overview of some of the issues and challenges that may arise in cancer genomics research specifically.

2.2 Next-Generation Sequencing Data

2.2.1 Data description

DNA is often visualized or described as a sequence of four different nucleotides, adenine, cytosine, guanine, and thymine, the order of which encodes all of the genetic information for an organism. A DNA molecule is double-stranded with a forward and reverse strand. The two sequences of nucleotides are complementary to each other (adenine binds with thymine, and cytosine binds with guanine), and thus only one strand is necessary to know the content of both strands. These paired nucleotides are also known as base pairs (bp), and the term base pair is used as a unit of measurement for the length of a DNA sequence. DNA can be represented digitally as a string containing the letters A, C, G, and T. The entire human genome is thus organized into a set of strings, one for the forward strand on each chromosome. The process by which the DNA from a particular biological sample is obtained and transformed into a digital representation is known as sequencing.

Next-generation sequencing (NGS) technology, also called second-generation sequencing, was developed with a primary objective to drastically increase the throughput and cost-effectiveness of DNA sequencing as compared with traditional Sanger sequencing [10]. In general, this process works by extracting DNA from cells, amplifying the DNA content to obtain duplicate copies for a stronger signal, and randomly fragmenting the DNA molecules into millions of short segment. Each fragment is then amplified in order to increase signal strength and accuracy, and the sequence of bases that compose each fragment are then determined. The mechanism by which bases are determined depends upon the sequencing technology being used. For example, Illumina sequencing uses bridge polymerase chain reaction (PCR) to clone fragments into clusters, and nucleotide-specific fluorescent labels and a camera are used to determine the sequence base-by-base. For a review of different NGS sequencing technologies, see Shendure and Ji [10].

Because the error rate typically increases for bases farther along in a fragment's sequence, the sequence of determined bases is typically on the order of hundreds of base pairs. Each of these sequences are called reads. However, the same fragment can be "read" from both ends, resulting in two reads (each 100-200 bp in length for much of the data used throughout this thesis) which are called paired-end reads, or read pairs. The approximate size of fragments is typically known, so there is also knowledge about how far apart the read pairs are from each other in the underlying genome that they were sequenced from.

The resulting data often contains millions of pairs of reads, each composed of a string of A, C, G, and T, corresponding to a portion of the forward strand on the genome they were sequenced from. Note also that these reads can be overlapping, in the sense that the same region of the genome can have many reads that were sequenced from it due to the fact that the initial DNA molecules are amplified. DNA strands are "read" in a particular direction, which is denoted as 5' to 3'. Unless otherwise specified, a DNA sequence string should be interpreted as being a 5' to 3' sequence of nucleotides.

2.2.2 Alignment and assembly

One of the typical first steps in NGS data analysis is to determine where each of these read pairs “came from” relative to a reference genome. Efficient and specialized string-matching algorithms, known as alignment algorithms, have been developed for this purpose. Examples of common alignment algorithms for NGS data include BWA [11, 12, 13, 14] Bowtie [15], and GSNAP [16]. Aligners will determine the most likely location in the genome that matches each read pair, and will transform the raw sequencing data into an alignment file. Alignment files contain the same sequence data as the raw read files, along with additional associated information describing the location and quality of the read's alignment. Since sequencing data is inherently error-prone, alignment algorithms allow for the possibility of mismatched bases, skipped bases, and inserted bases when aligning a read against a reference genome. Reads may also be aligned such that only a portion of the read is considered to be a match against a location in the reference genome, with the remainder being discarded. This situation is referred to as soft-clipping.

Once an alignment is generated, it creates a more clear picture of the genetic mutations present in a sample. Areas of the genome in which the aligned reads contain different bases than the reference sequence are indicative of a variant sequence in that location of the genome for that sample. This will be further discussed next in Section 2.2.3. If the genome of the sample is sufficiently different than the reference genome, such as for a highly-mutated genome, or the genome of a different species, then many reads will remain unmapped by the alignment algorithm.

The second main type of analysis that can be done using NGS data is assembly. This is the process by which raw reads are fit together, like pieces of a jigsaw puzzle, to obtain the genomic sequence of the sample. This is a very difficult process due to the high error rate for NGS data, as well as the repetitive genomic regions often present in eukaryotic organisms. However, when sequencing DNA from a new organism, this is often a necessary step in order to achieve accurate alignments for future samples. A brief background of various assembly techniques will be discussed later in Section 4.2.

2.2.3 Variant detection

As mentioned previously, one of the most common types of analysis done on NGS data is detecting variants. This is primarily done by identifying irregularities in the alignment. For example, if many of the reads aligned in a region contain an A nucleotide at a certain position while the reference genome contains a T nucleotide at that position, is is evidence supporting the presence of a single nucleotide polymorphism (SNP) variant. Other common small variants include deletions and insertions of bases. Together these are known as INDELs (Insertions/Deletions) and can be detected in a similar way (e.g. many reads aligned in a region that are missing a short sequence of bases, or that contain a novel non-reference sequence of bases).

A more interesting type of analysis is in detecting structural variants (SVs), which are deletions, insertions, inversions, duplications, or translocations of large segments of DNA. A subcategory of SVs are copy number variants (CNVs), which specifically refer to deletions and duplications. Historically, array comparative genomic hybridization (array CGH) and SNP microarray techniques have been used for SV and CNV discovery because of their low cost and high throughput. However, with the falling costs of next-generation sequencing, NGS-based methods for SV discovery have become increasingly popular. NGS-based SV discovery techniques offer the advantages of more precise breakpoint resolution and a larger spectrum in both the size and type of SVs that are capable of being discovered [17, 18, 19]. Since SVs often affect thousands of bases at a time, they cannot be detected using the same approaches as are used to detect SNPs and INDELs from an alignment. However, a similar logic of looking for clusters of irregularly-aligned reads (albeit different kinds of irregularity) can still be used to identify SVs. If the sequenced genome contains SVs, reads will not align correctly near the breakpoints of these variants, and evidence of this sort of unexpected behavior can be extracted from the alignment. This type of information falls into three main categories: read-pair, split-read, and read-depth information.

Unexpected alignment in terms of read-pair behavior includes paired-end reads that align further apart or closer together than would be expected from the distribution of fragment sizes in the sample, or paired-end reads that align in an incorrect orientation (i.e. the expected directions of strands are flipped). For example, if a pair of reads aligns to a reference genome significantly further than expected from each other, it could be indicative of a deletion in the sequenced genome since it suggests that the sequenced genome is lacking a region between the reads that is otherwise present in the reference sequence. Conversely, read pairs that align close together could be due to an insertion. Discordant orientation of read pairs can also reveal other types of SVs, such as an inversion in the sequenced genome that flips the orientation of one read of a pair that spans the inversion breakpoint. Utilizing read-pair information is the most well-established of the NGS-based SV discovery techniques, and includes programs such as BreakDancer [20], Hydra [21], VariationHunter [22], MoDIL [23], and PEMer [24].

Split-read approaches look at reads that are only partially aligned to the reference, with the remainder being "soft-clipped." This methodology discovers SVs by finding pairs of soft-clip clusters that match each other in the reference, indicating an adjacency of genomic regions that are normally separated in the reference. This situation can be indicative of a deletion or insertion in the sequenced genome. For example, a 100 bp read might align concordantly with a reference genome for its first 60 bp, up until location A in the reference. If the remaining 40 bp segment happens to correspond with a region further down the reference genome starting at location B, then this split read likely indicates the presence of a deletion in the sequenced genome of the region between A and B in the reference genome. These approaches offer the advantage of exact breakpoint determination, but are difficult to achieve in practice because of the small size of most next-generation sequencing reads. Small reads (e.g. 100 bp) are difficult to align with a reference genome of three billion bp while allowing for sequencing errors, SNPs, and small INDELs, and this difficulty only increases when aligning split reads. Split-read techniques for SV discovery include Pindel [25] and CREST [26].

Finally, read-depth approaches for SV discovery look at the total number of reads that align to different regions of the genome. Ideally, the sequenced reads are uniformly taken from the genome, and thus the number of reads that align to a particular location in the reference should be fairly consistent across the genome. If the read-depth is significantly higher or lower in a particular region of the reference genome, it could be a sign of a duplication or a deletion of that region in the sequenced genome, respectively. These approaches can be significantly complicated by naturally repetitive regions of the genome or by heterozygosity in the sequenced sample. The accuracy of determining SVs using this approach is also heavily dependent on having high and consistent read coverage (i.e. average read-depth) throughout the sequence. On the other hand, this means that the power to predict SVs from read-depth information is greater for longer insertions and deletions because it becomes less likely to observe disparate read-depth along long regions by chance. Since read-pair and split-read approaches are limited to discovering SV signals locally near a pair of reads, read-depth approaches can offer complementary discovery of larger SVs. Programs that fall into this category of SV discovery include EWT [27] and CNVnator [28].

Because of the different ways that these approaches analyze alignments, they tend to produce differing SVs calls. For example, a recent review of SV studies for the 1000 Genomes Project [3] found that 80% of the SVs discovered by read-depth approaches were not found by split-read or read-pair approaches [29]. Furthermore, of the 15,893 SVs from these studies, only 303 were detected by all three approaches. This knowledge has lead to the development of new techniques that combine multiple types of information to predict SVs in a more robust and accurate manner, including SVSeq [30], forestSV [31], PRISM [32], DELLY [33], GASV [34], LUMPY [35], SoftSV [36].

2.3 Challenges in Analyzing Tumor DNA

These various types of NGS analyses become complicated by a number of factors when applied to sequencing data originating from tumor tissue cells as opposed to normal tissue DNA.

First, tumors are often characterized by a high level of genetic heterogeneity. The genomic instability that is typically present in tumor cells, coupled with their inherent rapid proliferation, means that new mutations are constantly being introduced as a tumor grows. Some of these mutations will result in a selective advantage for the cell, which may then eventually proliferate into a cellular subpopulation within the tumor containing those mutations. This results in an “evolutionary battle” between different subpopulations of cells (or subclones) within the same tumor. A sequencing sample taken from tumor tissue is likely to obtain DNA from several of these different tumor subclones, as well as DNA from normal tissue cells, all containing different sets of genetic mutations.

This heterogeneous mix of sequencing data can increase the difficulty in making inferences from alignments. For example, consider the example illustrated in Figure 2.1. In this case, the blue cells are normal tissue cells, and the orange and brown cells represent two different subpopulations of tumor cells. Some genetic variants will have been germline mutations (i.e. mutations that are present in every cell of the body). Evidence for these can be found in all aligned reads at the variant's locations, such as in the cases of SNP B and Deletion C in this example. However, other variants may originate only from cells that are less frequent in the sequencing sample, such as those evidenced by reads colored in orange and brown in the figure. Since there is no way to tell which cell a particular read originated from, this can make variant calling more complicated. For example, an SV prediction algorithm may assume that the evidence for Deletion B, which has few supporting reads, is merely a random fluctuation in coverage level or erroneously clipped reads, instead of a true variant present in a small tumor subclone.

The complexity of the genetic mechanisms that drive cancer is an additional challenge in analyzing tumor DNA. Associations between particular mutations and genetic diseases have been identified over the years, and in the cases of some disease even the complete pathogenesis from a genetics perspective has been discovered. Progress in understanding the underlying genetic mechanisms of cancer has been slower due to many oncogenic and cancer-driving mutations being extremely rare, and thus difficult to identify as statistically significant [9]. The uncontrollable cellular proliferation that characterizes cancer requires a multitude of failures on the cellular level in order to occur, including the development of self-sufficiency in growth signals, resistance to antigrowth signals, and an acquired evasion of programmed cell death [37, 38]. In other words, mutations and disruptions to multiple complex cellular systems must co-occur simultaneously for a tumor to develop, resulting in many of these mutations being rare across even large cohorts of patients, and thus difficult to identify. Accounting for this issue may require looking at mutations on the level of genetic pathways and networks of interacting genes, which in and of itself are still poorly understood [39].

Complicating this issue of identifying the causal mutations that drive tumorigenesis is the noise generated by spurious mutations and alterations resulting from the genomic instability characteristic in tumor cells [40]. Tumors contain an abundance of mutations, and a significant challenge is in distinguishing between true genes of interest and the

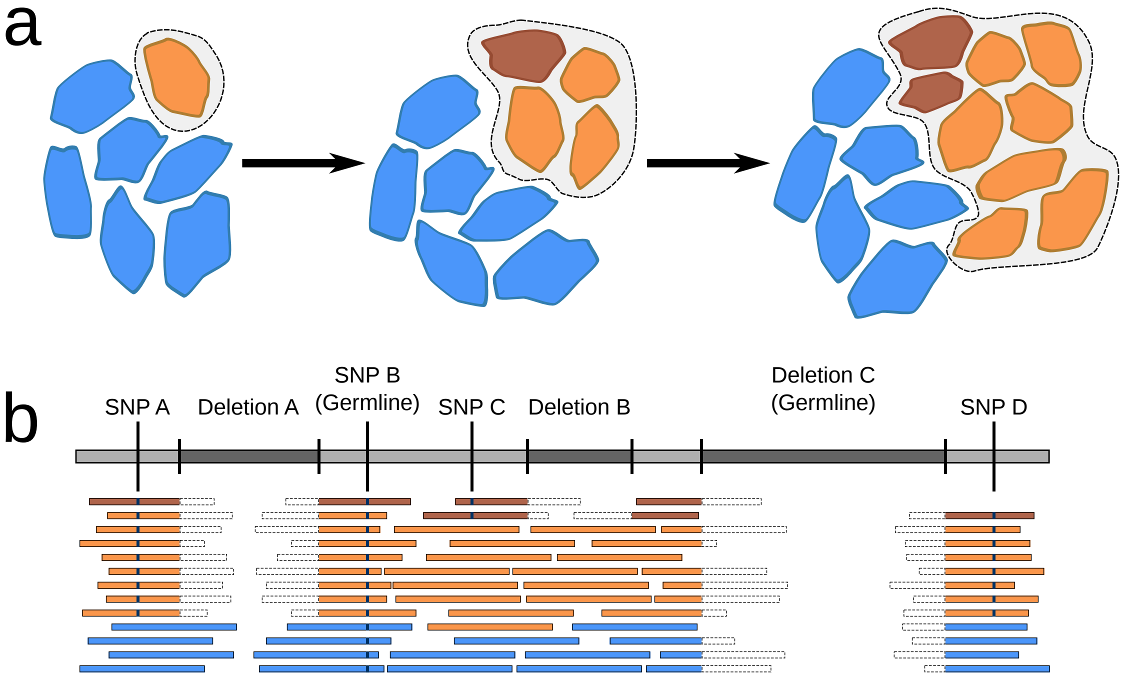


Figure 2.1: Example of alignment patterns found in a heterogeneous tumor sample. **(a)** The progression of tumor development results in several subpopulations of cells with different genetic backgrounds. The tumor cells (orange) originate with a set of genetic mutations in one cell and grow rapidly to outnumber the normal cells (blue). Additional mutations drive the formation of a new subclone (brown) with a distinct genetic background. **(b)** Reads sampled from the normal cells (blue) and cells from the two tumor subclones (orange/brown) are aligned against the reference sequence. Soft-clipped portions are indicated by a dotted line border, and SNPs are marked with small blue ticks on the reads. Evidence for germline-acquired mutations are present in all of the reads, but tumor-specific mutations are present in only a portion of the reads.

numerous insignificant alterations that are acquired a tumors evolves.

Chapter 3

Mining for Co-Occurring Genetic Mutations in Tumors

3.1 Introduction

The initial growth of a tumor (or the expansion of a new subclone within a developing tumor) is triggered by genetic mutations that disrupt normal cellular behavior and lead to uncontrolled cell proliferation. Determining which genetic defects can coordinate to cause tumor growth is an important topic in cancer research because it identifies possible targets for drug therapy as well as risk factor mutations that can predict cancer susceptibility.

One approach to addressing this problem is to conduct experiments in which many genes are disrupted throughout the genome across a number of laboratory mice. This can be done by inserting DNA randomly across the genome, via techniques such as viral insertion [41] or the *Sleeping Beauty* transposon system [42, 43], to disrupt gene regulator or protein-coding regions and thus “break” a set of genes. Alternatively, these insertions may also alter normal genetic behavior by acting as gain-of-function mutations. Some of these sets of insertional mutagens will cause a specific set of genetic defects that will trigger tumor growth, such as activation or increased expression of an oncogene, or inactivation or disruption of a tumor suppressor gene. DNA from those tumors are then sequenced and the locations of the inserted DNA are determined. The result is called insertional mutagenesis data, and it consists of a number of tumors each containing a set of insertion locations. Many of the insertion locations in each tumor will not be relevant for the analysis (e.g. if they occur in non-coding regions or near genes that have no affect on tumorigenesis), but there will be a subset of insertions in each tumor that are likely to be responsible for driving tumor growth.

Lab mice are typically the subjects of these experiments due to a long history of studies related to mouse genetics, existing technologies to manipulate mouse genomes, and most importantly a high level of homology to the human genome [44]. 99% of mouse genes have a homologous gene in the human genome, and for 80% of mouse genes there is a human ortholog (i.e. the best matching human gene correspondingly has its best match to the mouse gene) [45]. Because of this relationship, discoveries found via mouse genomics research is likely to lead to actionable insights toward the understanding of human genetics and disease, including cancer.

A number of techniques can be used to introduce these insertional mutagens into the mouse genome. One approach is to leverage the natural insertion and replication mechanisms of retroviruses [41]. Retroviruses infect cells by integrating themselves into a host cell genome after undergoing reverse transcription (i.e. RNA to DNA). The inserted genetic sequence, called a provirus, can then be translated and transcribed using the normal mechanisms of the host cell, and the proteins produced from the activated proviral genes work to replicate the retrovirus and begin the process anew. When used for the purposes of insertional mutagenesis, the known proviral sequence can be captured and amplified to identify the set of genetic locations affected by the retroviral insertions.

The *Sleeping Beauty* (SB) transposon system is another technique commonly used for insertional mutagenesis [43]. This system was originally derived from evolutionarily-dormant tranposable elements in salmonid fish, which were modified via genetic engineering to be reactivated [42]. The SB transposon system is a “cut and paste” process in which the genetic sequence to be inserted is removed from the transposon and inserted at a random TA dinucleotide (i.e. thymine-adenine) in the recipient genome. Similar to the retroviral insertional mutagenesis approach, the insertions locations are then determined by capturing and amplifying the known genetic sequence of the insertion.

The goal with analyzing these insertional mutagenesis data sets is to identify individual insertion locations that are most frequent, or common sets of insertions that co-occur across multiple tumor samples at significant levels. These insertions are likely to reflect genomic regions that, when mutated or disrupted, coordinate to drive tumor growth. Further validation is required to establish a functional role or direct link between candidate patterns and tumor growth. Thus, these types of analyses can be thought of as “hypothesis-generating” and are used to produce genes or sets of genes that should be further investigated for their potential roles in tumorigenesis. Insertional mutagenesis studies have been frequently used over the years to identify oncogenes, tumor suppressor genes, and other genetic drivers of cancer [46, 47, 48, 49, 50, 51].

Various methods have been developed that identify single common insertion sites [52, 47, 48, 53, 54]. However, tumor growth is often caused by mutations in multiple interacting genes, and thus discovering co-occurring common insertions is more likely to help explain the underlying genetic pathways involved. Furthermore, there may be co-occurring sets of insertions which are significantly enriched in a data set when considered together, but are below a significance threshold when considered on their own. Thus, there exists a need for methods that can identify higher order patterns of commonly co-occurring locations.

One approach uses a Gaussian kernel convolution (GKC) method to find common co-occurring pairs of insertions [55]. In this approach, all possible pairs of insertion locations are mapped to a two-dimensional space, and the density peaks are approximated using a kernel density estimation. The top density peaks represent the most common co-occurring insertions. In theory, this approach can be extended to find co-occurring triplets or higher order patterns. However, this can quickly become computationally infeasible when there is a large search space of potential patterns to explore, and the approach is still computationally intensive for identifying pairs for dense data sets (i.e. many insertions per tumor). Furthermore, there are additional complications that arise when multiple tumors in an insertional mutagenesis data set originate from the same mouse. This situation is not addressed by current approaches. New approaches are needed that are more efficient, allow for discovery of higher order patterns, and account for the possibility of multi-tumor mice in the input data set.

One potential solution to this problem may be to leverage algorithms from the association analysis branch of data mining research. These techniques were developed to efficiently identify frequent sets of items present across a data set of market basket transactions, which is made possible my smart pruning of the large search space as potential patterns are examining and ruled out. Association analysis and the *Apriori* family of algorithms have been successfully developed and applied to numerous and diverse types of data mining problems [56]. With some small modifications, the problem of identifying common sets of co-occurring insertion locations in a set of tumors can be transformed into that of analyzing these market basket transaction data sets. Instead of sets of transactions there are sets of tumors, and instead of a transaction containing a set of items we have a tumor containing a set of insertion locations. Applying *Apriori* or similar frequent item set mining algorithms can offer drastic improvements in efficiency for discovering all higher order patterns of co-occurring insertion locations.

In this chapter we examine a new approach for analyzing insertional mutagenesis data sets to find arbitrarily-sized co-occurring sets of insertion locations in a more efficient manner. This approach is based on an association analysis methodology. In Section 3.2 we describe the data and problem statement in more detail. An overview of related work will be presented in Section 3.3. We describe the details of our proposed approach in Section 3.4. Section 3.5 contains experimental results that demonstrate the efficacy of our approach, including comparisons against the GKC approach, evaluation using simulated insertional mutagenesis data sets, and results from a collection of new data sets generated via SB transposon systems. Finally, we conclude with a discussion of issues and ideas for future work in Section 3.6.

3.2 Data Description and Problem Statement

3.2.1 Data description

An insertional mutagenesis data set, $T = \{t_1, t_2, \ldots, t_N\}$, consists of N tumor data objects. Each tumor can be defined as a set of insertion locations (i.e. $t_i = \{l_{i1}, l_{i2}, \ldots, l_{ik}\}$, where k is the number of insertion locations in t_i), which are each represented as a genomic coordinate with a chromosome and a location within that chromosome. For mice, which are typically the subjects of insertional mutagenesis experiments, including all those described here, there are 22 possible chromosomes (19 autosomal chromosomes ($chr1 - chr19$), the two sex chromosomes ($chrX$ and $chrY$) and the mitochondrial DNA which is often represented as its own chromosome ($chrM$)).

Insertional mutagenesis data sets may consist of tumors that are each obtained from their own distinct mouse. In this case, mice and tumors may be referenced interchangeably. In other insertional mutagenesis data sets however, multiple distinct tumors may be obtained from the same mouse. This creates a many-to-one correspondence between tumors and mice. In other words, each tumor object t_i is associated with a particular mouse, and each mouse can be characterized as a collection of one or more tumor objects. Analyzing these types of insertional mutagenesis data sets with multi-tumor mice can produce spurious results if care is not taken to address the possibility of metastatic tumors and evolutionary relationships between tumors originating from the same mouse. This is discussed further in Section 3.4.4.

The GRCm38 mouse reference genome is $\sim 2,700,000,000$ base pairs in length across all chromosomes, while the total number of insertions across all tumors in one insertional mutagenesis data set typically ranges in order of magnitude from thousands to hundreds of thousands. Thus, most individual insertion locations will be unique across a data set T. In order to identify patterns across the tumors in a data set, insertion locations must first be generalized from specific genomic coordinates to genomic regions or neighborhoods (also called insertion sites).

3.2.2 Problem statement

The goal is to identify all sets of insertion sites that are common among the tumors in T. Individual insertion sites that are common are referred to as common insertion sites (CISs), while sets of insertion sites that are jointly common are referred to as common co-occurring insertions (CCIs). Note that CCIs will also be referred to interchangeably as patterns or item sets throughout this chapter, depending on context. This general problem statement leaves open several questions which we will discuss in our proposed approach and investigate in our experimental results, including:

- How to generalize insertion locations from genomic coordinate to genomic region.
- How to define what qualifies as a "common" set of insertion sites.
- How to evaluate the statistical significance of candidate patterns.

Additionally, the requirement to identify the complete collection of CCIs introduces a computational challenge in managing the search space. An effective solution to this problem must also be able to efficiently identify all patterns and evaluate their statistical significance in a reasonable amount of time.

3.3 Related Work

3.3.1 Poisson distribution-based techniques

Many techniques have been developed to detect common insertion sites (CISs) or common sets of insertion sites. One straightforward approach for detecting single CISs is based on Poisson distribution statistics. Assuming a uniform distribution of insertion locations across the genome, the Poisson distribution can be used to estimate the expected number of insertions in a given window size given the total number of insertions and genome size. Specifically, the probability of observing x insertions in a window of size w (in base pairs) is given by:

$$P(x|w,g,n) = \frac{e^{-\frac{nw}{g}}\left(\frac{nw}{g}\right)^x}{x!}$$

where g is the genome size (in base pairs) and n is the total number of observed insertions across the genome.

This model can be used to assess the probability of an observed number of insertions in a given window occurring under the null distribution (i.e. all insertions locations being equally probable). After using Bonferroni correction to account for multiple hypothesis testing, this results in p-values for candidate CISs. It may be necessary to examine a variety of window sizes, each of which results in a set of CIS results.

One publicly available open-source software package that uses this method is Transposon Annotation Poisson Distribution Association Network Connectivity Environment (TAPDANCE) [54]. TAPDANCE also handles processing and mapping of the raw sequencing data to initially identify the insertion locations themselves. The software also allows for automatic annotating of CISs (such as via a user-provided gene location coordinates file), as well as identifying associations between two CISs or a CIS and a phenotype.

One advantage of using Poisson distribution-based techniques to identify CISs is that it only requires straightforward and fast computations, and is thus able to handle high-throughput data sets in an efficient manner. However, significance is only assessed for single insertion locations and not sets of co-occurring insertion locations. This approach also makes assumptions about a uniform distribution of insertion locations across the genome under a null hypothesis, which is not strictly accurate and may produce spurious results.

3.3.2 Monte Carlo-based techniques

Another approach that has been used to estimate the expected number of insertions in a given window uses a Monte Carlo simulation [47] [48]. These techniques are intended to address the issues with assuming a uniform distribution of insertions under the null hypothesis. In reality, there are numerous regions of the mouse genome which are inaccessible for transposons, or that remain unresolved in the reference genome and thus cannot be mapped to during insertion identification. For the SB transposon system, insertions are also known to occur only at TA dinucleotide locations, which are not necessarily uniformly distributed across the genome.

Using this approach, a number of Monte Carlo simulations are done to randomly place the observed number of insertions in each chromosome at TA dinucleotide sites that are in resolved regions of the reference genome for that chromosome. These sets of simulations produce null distributions that can be used to identify significance thresholds for the window size of given counts of insertion locations. For example, one study [48] used Monte Carlo simulation to determine significance thresholds of six insertions within 130 kbp of each other, five insertions within 65 kbp of each other, or four insertions within 20 kbp of each other. These were then used as rules to identify CISs.

These techniques can produce results that are theoretically more accurate than Poisson distribution-based techniques due to accounting for insertion location biases. However, this is also less efficient and it has been shown that in practice results can still be very similar to Poisson-based techniques [54]. Again, these approaches also focus only on identifying only single common insertion locations instead of sets of co-occurring locations.

3.3.3 Gaussian kernel convolution techniques

One of the leading approaches has been based on Gaussian kernel convolution (GKC). This has been developed for both CIS detection [52] and CCI detection of pairs [55]. The main concept behind GKC-based techniques is to apply a kernel function to create a density surface in a one- or two-dimensional space that represents the insertion density across the genome. Optimization techniques are then used to find the peaks on this density surface, which represent CISs (for the one-dimensional version) or CCIs (for the two-dimensional version). Kernel density estimation has traditionally been used to estimate a probability distribution function of a random variable given a sample [57, 58].

In the two-dimensional version, all unique pairs of insertion locations are first generated. The chromosomes of the genome are concatenated, and the pairs represent points on a two-dimensional domain of size $GenomeWidth \times GenomeWidth$. The density surface is built by applying a Gaussian-like kernel function with maximum value of one to each insertion pair. Specifically, the density at a given point (x, y) is defined as:

$$density(x, y) = \sum_{i=1}^{NumPairs} K(x - x_i)K(y - y_i)$$

where $K(z) = e^{-2z^2/h^2}$, (x_i, y_i) is the ith pair, and h is a kernel width parameter that controls how quickly the value of the kernel function $K(z)$ decreases for points away from the insertion pairs. In the original 2DGKC paper, h values of 1000, 17487, 30579, 93506, 163512, 285930, and 500000 (in base pairs) were used [55].

Once the collection of all unique pairs is generated, an optimization technique is used to locate the nearest peak on the density surface starting at each pair. Insertion pairs are grouped into windows of size 1,500,000 bp, and nearest peaks are calculating using only the insertion pairs in the current window and the surrounding eight windows. This speeds up the optimization and is justified by the negligible impact from insertion pairs outside that range. The collection of all unique peaks generated by this algorithm represent the candidate CCI pairs. The original one-dimensional version of GKC is similar, but only generates a density surface in one dimension (i.e. $density(x) = \sum_{i=1}^{NumInsertions} K(x - x_i)$) and thus does not require first generating all unique insertion pairs.

Finally, permutation simulations are performed to evaluate significance of candidate

CCI pairs. The 2DGKC algorithm defines a concept of cross-scale common co-occurring insertions (csCCI), which are insertion pairs that are significant across a sufficient number of kernel widths (i.e. the h parameter in the above kernel function) [55]. These are then taken to be the final result set from this approach.

The GKC approach was the first to identify significant pairs of insertions rather than significant single insertion sites. This technique can in theory be applied to find higher order CCIs, but the combinatorial explosion of the search space (i.e. generating all unique triples, quads, etc.) makes this computationally infeasible. For particularly dense insertional mutagenesis data sets, GKC also does not scale well even in the two-dimensional version. Another potential problem with this approach is that concatenating chromosomes together to produce the insertion location feature space may produce spurious results from insertion locations that get groped together at ends of two different chromosomes. However, this is not really observed in practice due to the telomeric ends of chromosomes where insertions are not likely to be mapped to.

3.3.4 Other techniques

Unlike other techniques that search for CISs or CCIs across the whole genome, Gene-centric common insertion site analysis (gCIS) [53] is a method that only analyzes genomic regions within RefSeq genes [59], including promoter regions. A p-value is estimated for each RefSeq gene by comparing the observed and expected numbers of insertions in the gene using a Chi-square test. The model for expected number of insertions takes into account the number of tumors and their insertion counts, as well as the number of candidate insertion locations within the gene (i.e. TA dinucleotides for SB-generated data sets). The gCIS algorithm has been shown to identify novel CISs not discovered by Poisson distribution-based techniques or Monte Carlo-based techniques [53]. However, focusing on only analyzing insertions within RefSeq genes and promoter regions may miss novel genes. The gCIS algorithm is also limited in being only able to identify CISs and CCIs.

Another method that has been developed is Poisson Regression Insertion Model (PRIM) [60]. This approach uses Poisson regression to estimate expected insertion counts on a window-by-window basis. The more generalized Poisson model used in PRIM avoids the overly-simplistic assumptions of more basic Poisson statistics-based techniques by accounting for variables such as chromosomal bias for insertion sites, or TA dinucleotides within a window. Furthermore, PRIM also features a pairwise model to estimate expected co-occurrence of candidate window pairs. However, higher order patterns remain a challenge due to the combinatorial explosion of the search space.

3.4 Proposed Approach

In order to address the scalability challenges of existing techniques and locate higher order CCIs, we have developed a new approach that identifies arbitrarily-sized co-occurring insertion locations in a more efficient manner [61, 50, 51]. Our approach handles the computational challenges of identifying higher order patterns by applying association analysis and the *Apriori* principle [62, 63, 64]. Additionally, this approach uses a biologically-meaningful approach for generalizing insertion locations, evaluates statistical significance of candidate CCIs in a novel way, and accounts for the possibility of multiple tumors originating from the same mouse in an insertional mutagenesis data set.

3.4.1 Apriori principle and frequent item set mining

Association analysis is traditionally discussed in relation to market basket transaction data, in which the input data is a binary matrix in which rows are transactions or purchases, columns are items that can be purchased, and matrix entries indicate the set of items purchased in each transaction. The goal is to find item sets that co-occur in a minimum number of transactions [62, 63].

The number of transactions that an item set co-occurs in is known as the *support count* of the item set. All possible combinations of items must be examining to determine whether or not they exceed a predefined support count threshold. The huge search space of possible item set combinations is dealt with using an algorithm known as *Apriori* which restricts the search space as infrequent item sets are found [64, 63]. An illustrated example of this can be seen in Figure 3.1). Since support count is an anti-monotonic property (i.e. any superset of a given item set must have the same or smaller support count as that item set), supersets of an infrequent item set (i.e. support count below the threshold level) can be removed from the search space, as they must be infrequent as well.

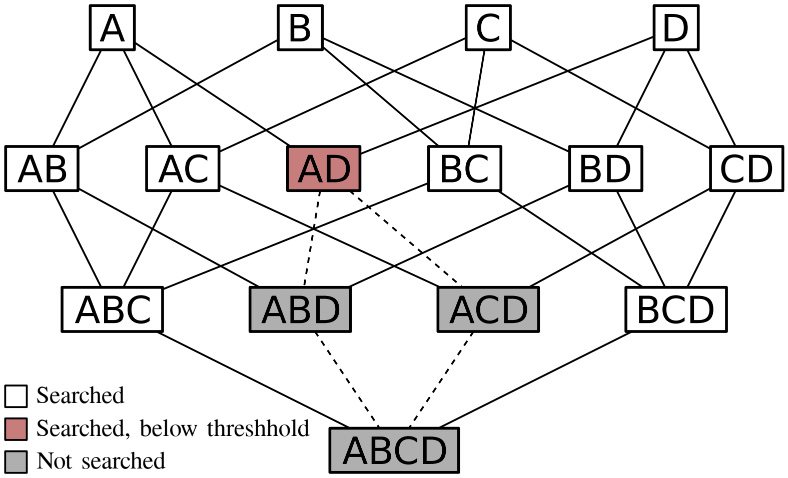


Figure 3.1: Illustration of support-based pruning in the *Apriori* algorithm. Letters represent items, and boxes are possible item sets. When searching for item sets that are above the support count threshold, the search space can be restricted by removing all of an item set's supersets when it is below the threshold. In the illustration above, since item set AD is below the support count threshold, we know that ABD, ACD, and ABCD will be as well, and thus they do not need to be examined or considered.

The *Apriori* algorithm has grown into a family of algorithms centered around this principle that have improved the efficiency over the original technique. For example, the Eclat algorithm [65] uses a depth-first search of the pattern space (as opposed to the breadth-first search of *Apriori*) to improve performance in many cases. Frequent Pattern growth (FP-growth) [66] is another newer frequent item set mining algorithm that uses prefix tree data structures to obtain improved performance, and is the algorithm used in our pipeline (see Section 3.4.3).

Association analysis techniques can be applied to insertional mutagenesis data in order to address the computational challenges of examining all possible insertion site combinations to discover higher order candidate CCIs. Rather than evaluating a set of transactions, each defined by a set of items, we instead are evaluating a set of tumors, each defined by a set of insertion sites. The resulting item sets, which can be single items or sets of multiple items, can then be considered candidate CISs or CCIs.

3.4.2 Generalizing insertion locations into genomic regions

In order to apply association analysis techniques to discover item sets in insertional mutagenesis data, the data must first be transformed into a binary transaction matrix in which rows represent tumors and columns represent insertion sites. Since nearly all of the genomic coordinates of the insertion locations are likely to be unique across a data set, these locations must first be generalized to represent genomic *regions* instead of specific genomic coordinates. This allows for items (i.e. genomic regions) to be present across multiple tumors in the transaction matrix. The process of generalizing insertion locations can have a significant impact on the outcome of the overall analysis. If items remain too specific to individual tumors, very few item sets will be returned. On the other hand, generalization into very large genomic regions will result in single items that grouped together insertion locations that have very different affects from each other, and thus produce spurious results.

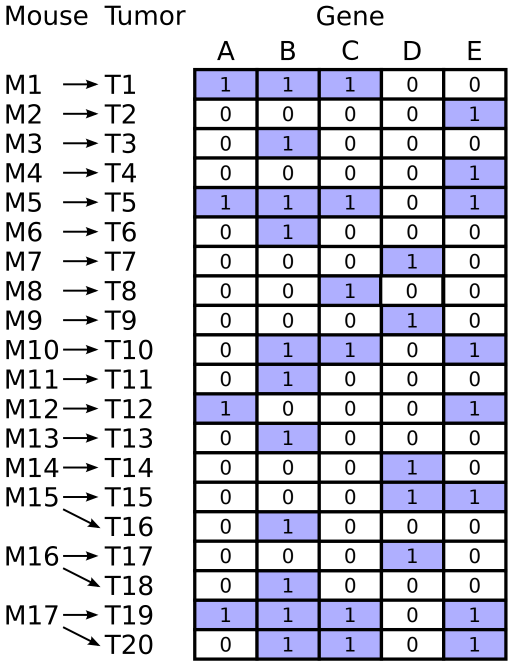
One approach to generalizing insertion locations into genomic regions is to apply the one-dimensional version of the previously described GKC approach [52] (see Section 3.3.3). In this case, each insertion location is used to build a density function defined by:

$$density(x) = \sum_{i=1}^{NumInsertions} K(x - x_i)$$

where $K(z) = e^{-2z^2/h^2}$, x_i is the ith insertion location, and h is a kernel width parameter that controls how quickly the value of the kernel function $K(z)$ decreases for points farther away from the an insertion location. The nearest peak on this density surface for each insertion location is then identified. This is essentially a version of "clustering" insertion locations together via their density.

An alternative approach would be to make each item in the transaction matrix represent a particular gene, and then assign each insertion location to its nearest gene. This offers the advantage of working with biologically-meaningful items. In our algorithm, the nearest gene for each insertion location is determined using the *closest* command from the *bedtools* software package [67]. Gene coordinates are taken from the RefSeq database [59] using the relevant mouse reference genome (i.e. GRCm37 or GRCm38, depending on the reference used for mapping insertion locations).

In our pipeline, we have chosen to use the nearest gene approach for generalizing



Mouse	Tumor	Gene				
		A	B	C	D	E
M1	T1	1	1	1	0	0
M2	T2	0	0	0	0	1
M3	T3	0	1	0	0	0
M4	T4	0	0	0	0	1
M5	T5	1	1	1	0	1
M6	T6	0	1	0	0	0
M7	T7	0	0	0	1	0
M8	T8	0	0	1	0	0
M9	T9	0	0	0	1	0
M10	T10	0	1	1	0	1
M11	T11	0	1	0	0	0
M12	T12	1	0	0	0	1
M13	T13	0	1	0	0	0
M14	T14	0	0	0	1	0
M15	T15	0	0	0	1	1
	T16	0	1	0	0	0
M16	T17	0	0	0	1	0
	T18	0	1	0	0	0
M17	T19	1	1	1	0	1
	T20	0	1	1	0	1

Figure 3.2: An example of an insertional mutagenesis data set after mapping insertions to their nearest genes and transforming the data into a binary transaction matrix. Ones represent presence of a gene in a given tumor. Tumors T1–T14 are each obtained from their own unique mouse (i.e. M1–M14). Mice M15, M16, and M17 each have two tumors.

insertion locations. In some of the experiments described below in which we compare our approach against the 2DGKC approach [55], the GKC-based clustering approach was used instead in order to facilitate fair comparison of results.

3.4.3 Generating candidate CCIs

The first step in our proposed approach is to remove insertions that occur in hotspot or artifact regions identified in previous studies [54, 68]. Each remaining insertion is then mapped to its nearest gene using the previously described process in Section 3.4.2, resulting in a set of tumors each containing a set of genes (rather than raw insertion locations). A binary transaction matrix is then constructed in which rows represent tumors and columns represent genes. Note that some tumors may contain multiple insertions that map to the same gene. A toy example of an insertional mutagenesis data set after mapping insertions to genes is given in Figure 3.2.

With a binary transaction matrix constructed, frequent item set mining can then be applied. We use the FP-growth algorithm [66] for this purpose, specifically the implementation developed by Christian Borgelt [69, 70]. The support count threshold is set to three, meaning all item sets must co-occur in at least three rows (i.e. three tumors) of the transaction matrix in order to be considered frequent. We use $sup(I)$ to indicate the support count of item set I throughout the description of our proposed approach

Rather than identify all frequent item sets, we return only closed frequent item sets [71, 72]. This excludes item sets with the same support count as one of their supersets. For example, in Figure 3.2, item sets $\{A, B, C\}$, $\{A, B\}$, $\{A, C\}$, and $\{B, C\}$ are all frequent (among others). Of those four item sets however, only $\{A, B, C\}$ and $\{B, C\}$ are closed frequent item sets, as none of their supersets have the same support count as they do (three and five, respectively). Item sets $\{A, B\}$ and $\{A, C\}$ are not closed since they have the same support count (three) as their superset $\{A, B, C\}$, and thus not returned. They are not considered interesting because they provide no additional information not already captured by the frequent item set $\{A, B, C\}$. Closed frequent item sets contain the same information as the set of all frequent item sets, but in a more compact form, which allows for lower support count thresholds without running into computational or memory issues during item set mining.

To determine statistical significance, a p-value is calculated for each candidate item set by modeling its support count (i.e. the number of tumors containing all of the items in the set) as the test statistic. The null distribution is modeled as a binomial with the number of trials equal to the number of tumors and the probability parameter equal to the joint probability of the individual insertions in the item set occurring together. That probability is calculated by multiplying the observed frequencies of each item in the item set. Thus, the p-value for an item set $I = \{i_1, i_2, i_3, \ldots, i_m\}$ is defined as:

$$p\text{-}value = \sum_{j=sup(I)}^{N} \binom{N}{j} p(I)^j (1 - p(I))^{N-j} \tag{3.1}$$

where N is the number of tumors in the insertional mutagenesis data set, and $p(I)$ is defined as:

$$p(I) = \prod_{k=1}^{m} \frac{sup(\{i_k\})}{N} \tag{3.2}$$

For example, consider the item set $I = \{A, B, C\}$ in Figure 3.2. The frequency of A in the overall data set is 4/20, the frequency of B is 11/20, and the frequency of C is 6/20. Thus we have:

$$\begin{aligned} p(I) &= \prod_{k=1}^{m} \frac{sup(\{i_k\})}{N} \\ &= \frac{4}{20} \cdot \frac{11}{20} \cdot \frac{6}{20} \\ &= 0.033 \end{aligned}$$

as the expected probability of observing item set I in an individual tumor. Using the above binomial model for the null distribution for support count, which was observed as $sup(I) = 3$, we then have:

$$\begin{aligned} p\text{-}value &= \sum_{j=sup(I)}^{N} \binom{N}{j} p(I)^j (1 - p(I))^{N-j} \\ &= \sum_{j=3}^{20} \binom{20}{j} 0.033^j (1 - 0.033)^{20-j} \\ &= 0.0269 \end{aligned}$$

3.4.4 Accounting for multi-tumor mice

Another aspect that existing approaches do not consider is the possibility of spurious patterns that arise from the presence of multi-tumor mice in the insertional mutagenesis data set. In an ideal data set, each tumor develops independently in multi-tumor mice. However, there is a possibility that some tumors in the data set may be metastatic tumors that developed from the same evolutionary tree as another primary tumor in the same mouse. In such cases, there would be high correlation of insertion patterns in the two tumors.

For example, consider mouse M17 in Figure 3.2, which has two tumors that originated from it (i.e. T19 and T20), each with a similar set of insertion-affected genes. A possible explanation for this scenario is that T20 was the primary tumor and contained insertions in or near genes B, C, and E. Following that, the tumor metastasized to form secondary tumor T19, which later also acquired an additional insertion in or near gene A. If the item set $\{B, C, E\}$ is not actually associated with tumorigenesis, it is then at risk of being reported as a spurious result. Our model would wrongly assume that this set of insertions was acquired in T20 and T19 independently when it was really just acquired in one of them, and thus return an artificially low p-value.

To address this concern, we propose a modification to our model that accounts for this situation when two of more of the supporting tumors for an item set originate from the same mouse. Specifically, for an item set $I = \{i_1, i_2, i_3, \ldots, i_m\}$, we define $sup^*(I)$ to be the number of mice that the item set exists in (in contrast to $sup(I)$ which defines the number of tumors that the item set exists in). We also define $supmod(I) = sup(I) - sup^*(I)$ to be the *support modification*, which is the difference between the two. The p-value is then calculated as if any tumors containing I and coming from the same mouse were actually a single tumor, with corresponding changes to overall tumor counts and support counts. Specifically, we have:

$$p\text{-}value = \sum_{j=sup^*(I)}^{N-supmod(I)} \binom{N - supmod(I)}{j} p(I)^j (1 - p(I))^{N-supmod(I)-j} \tag{3.3}$$

with:

$$p(I) = \prod_{k=1}^{m} \frac{sup^*(\{i_k\})}{N - supmod(I)} \tag{3.4}$$

In the prior example of item set $\{B, C, E\}$ in Figure 3.2, under our original model we would have:

$$\begin{aligned} p(I) &= \prod_{k=1}^{m} \frac{sup(\{i_k\})}{N} \\ &= \frac{11}{20} \cdot \frac{6}{20} \cdot \frac{8}{20} \\ &= 0.066 \end{aligned}$$

and:

$$
\begin{aligned}
p\text{-}value &= \sum_{j=sup(I)}^{N} \binom{N}{J} p(I)^j (1-p(I))^{N-j} \\
&= \sum_{j=4}^{20} \binom{20}{j} 0.066^j (1-0.066)^{20-j} \\
&= 0.0392
\end{aligned}
$$

But under the modified model that merges the two same-mouse tumors and thus effectively considers 19 overall tumors and a modified support count of 3 for the item set, we have:

$$
\begin{aligned}
p(I) &= \prod_{k=1}^{m} \frac{sup^*(\{i_k\})}{N - supmod(I)} \\
&= \frac{10}{19} \cdot \frac{5}{19} \cdot \frac{7}{19} \\
&= 0.051
\end{aligned}
$$

and:

$$
\begin{aligned}
p\text{-}value &= \sum_{j=sup^*(I)}^{N-supmod(I)} \binom{N - supmod(I)}{j} p(I)^j (1-p(I))^{N-supmod(I)-j} \\
&= \sum_{j=3}^{19} \binom{19}{j} 0.051^j (1-0.051)^{19-j} \\
&= 0.0699
\end{aligned}
$$

Additionally, an item set I is removed if $sup^*(I)$ is less than the support count threshold, which is set to three in our pipeline.

Note that it is inadequate to simply consider mice to be the rows of the transaction matrix, merging together insertions from all of a mouse's tumors. Doing so would create false co-occurrences between insertions from distinct tumors. The tumors from the same mouse may share some insertions, but may also have their own set of distinct insertions. Thus, we believe our proposed model is appropriate to correct for this situation on an "item set by item set" basis.

3.4.5 Handling multiple hypothesis testing

One of the challenges in evaluating the statistical significance of CCIs is in handling multiple hypothesis testing. Standard Bonferroni correction is often inadequate due to the large result set size that can be returned. As part of our analysis pipeline, we have developed two approaches for handling multiple hypothesis testing. The first approach is based on generating permutation simulations to calculate the false discovery rate (FDR) and corresponding q-values for each pattern. The second approach applies a filtering criteria to first remove uninteresting patterns and then apply Bonferroni correction on the much smaller result set.

Permutation-based false discovery rate approach

In order to account for multiple hypotheses testing under the permutation-based FDR approach, the significance of each candidate item set is determined by empirically estimating its q-value [73], which is the minimum FDR at which the test may be called significant [74]. This is done by running thousands of simulations in which the tumor that each insertion appears in is randomized while preserving the overall set of insertion locations and the number of insertions in each tumor. The q-value for each candidate item set is calculated as the percent of simulated results that had a p-value better than or equal to the p-value of the candidate item set divided by the percent of real item sets with a p-value better than or equal that of the candidate item set. CCI results can then be sorted by q-values to prioritize by statistical significance, and a q-value cutoff can be used to determine “final” result sets.

An ideal strategy for the permutation simulations should randomize the tumor membership for all items while maintaining the overall frequency for each item across tumors, as well as maintaining the number of insertions in each tumor. In other words, the row and column margin counts for the binary transaction matrix should be conserved. This keeps the characteristics of the distribution consistent while breaking relationships between individual items, thus forming a null distribution of insertions in which real co-occurrence between items disappears.

One randomization approach that accomplishes this is the swap randomization technique [75, 76]. Briefly, swap randomization works by representing the binary transaction matrix as a bipartite graph in which one disjoint set of nodes represents the rows of the transaction matrix, the other set of nodes represents the columns, and edges represent the presence of an item in the corresponding entry of the transaction matrix. Two edges that do not share a node on either side of the bipartite graph are chosen at random, and their connections are "swapped." This is then repeated a sufficient number of times to create a random bipartite graph with the same number of edges and same node degrees, and thus a binary transaction matrix with the same row and column margin counts. The approach developed by Gionis et al. [76] uses a Markov chain in order to guarantee that the new graph is uniformly sampled (i.e. the uniform stationary distribution of the Markov chain is uniform) from the space of all possible bipartite graphs with the same number of edges and the same node degrees.

For some data sets, swap randomization may be too computationally inefficient considering swap randomization runs should ideally occur thousands of times to build an adequate set of permutation simulations. This is discussed further in Section 3.5.3. Alternative approaches include sampling the appropriate number of insertions for each tumor based on the overall frequencies observed for all insertions, as well as permuting the tumor-to-insertion relationship across all tumor-insertion pairs (which we will refer to later in Section 3.5.3 as the permute randomization approach). These techniques are not guaranteed to conserve margin counts of the binary transaction matrix, but are much more computationally efficient than swap randomization.

Self-sufficient item set filtering approach

One way to filter item sets before evaluating statistical significance is to simply raise the support threshold parameter for the closed frequent item set mining algorithm. For particularly dense insertional mutagenesis data sets, this may actually be necessary. In general though, this is undesirable because many of the CCIs that actually directly lead to tumorigenesis are expected to have low support counts. A better filtering criteria would attempt to measure the interestingness of a pattern in some way that accounts for more than just its support count.

Toward this end, we have applied a modified version of the self-sufficient item set concept developed by Webb [77]. Self-sufficient item sets are item sets that meet three criteria for (1) productivity, (2) non-redundancy, and (3) independent productivity. Item sets that do not meet these criteria can be filtered out, as they are either not informative or redundant with a different self-sufficient item set.

An item set is considered to be productive if, over every possible partition of the item set into two subsets, its support count is greater than would be expected from the support counts of the two subsets. In other words, every possible two-subset partition must be positively associated with each other. This is measured via a Fisher's exact test and a significance threshold of 0.05. Specifically, an item set I is productive if:

$$\max_{X \subsetneq I} \big(p_F(X, I \setminus X) \big) \leq 0.05 \tag{3.5}$$

where $p_F(X, I \setminus X)$ is the p-value from a one-tailed Fisher's exact test of positive association between item set X and item set $I \setminus X$ (i.e. the items in I that are not in subset X). This is defined formally as:

$$p_F(X, Y) = \sum_{i=0}^{\omega} \frac{\binom{sup(X)}{sup(X \cup Y)+1} \binom{N - \mathrm{sup}(X)}{sup(Y) - sup(X \cup Y) - 1}}{\binom{N}{sup(Y)}} \tag{3.6}$$

where $\omega = \min(sup(X) - sup(X \cup Y), sup(Y) - sup(X \cup Y))$ and N is the total number of tumors in the insertional mutagenesis data set (or rows in the binary transaction matrix).

The second requirement for self-sufficient item sets is non-redundancy. An item set I is redundant if it contains a proper subset Y, Y contains an item i, and every row in the transaction matrix that contains $Y \setminus i$ also contains i. In other words, i always implies the "rest of" Y. As an example, consider a binary transaction matrix of medical data containing the frequent item set $\{$*Hospitalized*, *DischargedToHome*, *HeartAttack*$\}$. The subset $Y = \{$*Hospitalized*, *DischargedToHome*$\}$ contains an item $i =$ *DischargedToHome* that always implies the presence of item *Hospitalized*. Thus, the item set $\{$*DischargedToHome*, *HeartAttack*$\}$ may potentially be interesting, but the item set $\{$*Hospitalized*, *DischargedToHome*, *HeartAttack*$\}$ is redundant and can be filtered out from the result set. Formally, an item set I is redundant if:

$$\exists i, Y : i \in Y \wedge Y \subsetneq I \wedge \{t : t \in T \wedge i \in t\} \supseteq \{t : t \in T \wedge Y \setminus i \subseteq t\} \tag{3.7}$$

where T is the set of all transaction rows in the transaction matrix (or the set of all tumors in our case). Such item sets are redundant because i is known to always co-occur with $Y \setminus i$, and thus these redundant item sets can be filtered out.

Finally, self-sufficient item sets are also required to be independently productive. This accounts for situations in which one self-sufficient item set is a superset of another. For an item set to be independently productive, it must be productive even in the context of rows in the transaction matrix that do not contain any self-sufficient supersets. This avoids "driver-passenger" situations in which a superset item set is the main item set "driving" the pattern, and its subsets appear to be significant or productive, but are actually spurious patterns and not the main patterns of interest. The formal definition of independent productivity is the same as for productivity, except that Equation (3.5) is only applied to a subset of the transaction matrix. Specifically, when evaluating the independent productivity of item set I, we first remove all rows of the transaction matrix that contain sets of additional items from any self-sufficient supersets of I. For example, if there exist item sets $\{A, B, C, D\}$ and $\{A, B, C, E\}$ that are self-sufficient, then when evaluating the independent productivity of item set $\{A, B, C\}$ we first remove all rows of the transaction matrix that contain either D or E and then apply Equation (3.5).

Self-sufficient item sets have been shown to be an effective technique for filtering down to the most relevant or informative patterns after frequent item set mining [77, 78]. For our pipeline, we use a modified version of this approach that removes the requirement for non-redundancy. The requirement for non-redundant item sets is meant to address strict dependencies in the data set (i.e. one item, by definition, always implies the presence of another item or set of items). This makes sense for certain types of data, such as the prior example of the item $DischargedToHome$ always implying the presence of item $Hospitalized$. In the case of insertional mutagenesis data however, the insertion process happens randomly and insertions in one location do not require the presence of a different insertion site in order to occur. It should be noted that strict dependencies between observed insertion sites may be present in the data set by chance, but these do not reflect real dependencies.

Because of this caveat, we remove the requirement for item sets to be non-redundant. However, independently productivity of item sets is still evaluated in the context of the item set's supersets that are fully self-sufficient (i.e. productive, non-redundant, and also independently productive in the same sense). This approach effectively is less stringent in allowing for item sets to pass filtering by not requirement non-redundancy, but just as stringent for deciding which item sets are part of the context in which independent productivity is evaluated. The motivation for this is discussed later in Section 3.5.2, with evidence from simulated insertional mutagenesis data sets with known ground truth. Unless otherwise specified, we will define the concept of independently productivity in this context for the rest of this chapter.

Finally, after removing all item sets that do not meet the modified self-sufficient item set filtering criteria, we are left with an ideally far smaller set of the most informative item sets. In order to account for multiple hypothesis testing on this much smaller result set, a Bonferroni correction is applied to the item set p-values.

3.5 Experimental Results

3.5.1 Comparison with GKC approach

To compare our approach (referred to as AA for association analysis in this section) with the Gaussian kernel convolution approach (GKC), we ran both algorithms on insertional mutagenesis data from the Retroviral Taged Cancer Gene Database (RTCGD) [79]. This data set was used to evaluate the 2DGKC approach in its original paper [55]. The version of the *RTCGD* data set used in de Ridder et al. [55] contains insertions generated from retroviral tagging. After removing tumors with only one insertion (which can be ignored as the goal is to find higher order patterns), the data set contains 5,143 insertions in 1,031 tumors.

For these experiments we used a support count threshold of two for finding the closed frequent item sets. Even this weak constraint performs a significant pruning of the pattern space. In particular, out of 14,499 possible pairs, only 2,804 pairs (across all eight kernel widths) survived this minimal level of pruning. Of these, the highest support count was eleven. To facilitate ease of comparison between results from the AA and the GKC approaches, we use the GKC-based clustering approach for generalizing insertion locations into insertion sites to prepare input data for the AA approach.

There were 137 triples with a support count of two and one triple with a support count of three. There were only six quads (i.e. CCIs containing four genes), all with a support count of two. A more detailed account of these results, broken down into results

	Support Count												
Kernel width	**Pairs**										**Triples**		**Quads**
(base pairs)	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**2**	**3**	**2**
10,000	77	6	5	1	1	0	0	1	0	0	1	0	0
17,487	107	10	4	4	1	0	0	0	1	0	2	0	0
30,759	137	13	6	4	1	0	0	0	1	0	4	0	0
53,472	179	22	8	4	2	0	0	0	1	0	6	0	1
93,506	250	29	13	4	2	1	1	1	2	0	14	0	1
163,512	363	43	15	5	2	1	1	1	2	0	20	0	1
285,930	538	91	18	10	2	2	1	1	1	1	36	0	1
500,000	648	109	29	11	3	3	1	1	1	1	54	1	2
Total	2299	323	98	43	14	7	4	5	9	2	137	1	6

Table 3.1: Summary of item set results from AA approach for *RTCGD* data set. Each entry is the number of occurrences of item sets with a particular support count and size (i.e. pair, triple, or quad). Each row reflects the results from using a different kernel width to generalize insertion locations into insertion sites and items in the binary transaction matrix.

per kernel width, is shown in Table 3.1. As might be expected, the number of patterns increases as the size of the kernel width increases. We are only interested in the pairs produced by the AA approach when comparing against the GKC approach, as GKC does not identify higher order patterns.

Consistency of resulting patterns

For the same *RTCGD* data set, 86 significant CCIs (pairs) were found using the GKC approach, which evaluates significance by requiring a CCI to be statistically significant across a sufficient number of kernel widths (see [55] for details.) Such pairs are referred to as cross scale common co-occurring insertions (csCCI). Since the locations of similar CCIs are not necessarily the same between the AA and GKC approaches, we use the requirement that two CCIs must have a Euclidean distance less than half their window size (see Section 3.3.3) in order to be considered a match (i.e. in order to claim that both represent the same CCI).

For our first comparison with the csCCIs found by GKC, we compared all the pairs found by the AA approach with the ranked list of 86 csCCIs from the GKC approach. All but one of the csCCIs are found in at least one of the kernel widths in the AA approach. The one csCCI that is not found is the pair with coordinates 801,913,615 and 802,003,315 in terms of absolute base pair locations within the genome. This csCCI could be caused by a single CIS that was considered as a pair in the GKC approach, but not by our AA approach. Overall, the AA approach found all of the significant CCIs that the GKC approach produces, plus additional CCIs not found using the GKC approach. However, there are differences in significance.

Since only examination of the pairs of genes by domain experts can determine if a CCI is biologically meaningful, it is difficult to evaluate either approach in terms of its ability to return biologically relevant results. Instead, we can only rank results in order of statistical significance, in the belief that the more biologically relevant patterns are likely to exist near the top of these lists. Therefore, in addition to knowing that the CCIs from the GKC approach can be found in patterns produced by the AA approach, we also wanted to examine if the "best" CCIs are consistent between the two approaches. We computed the FDR q-values for the CCIs produced by the AA approach in all 8 kernel widths. These FDR q-values are not intended to prove any biological meaning, as that can only be done through *in vitro* or *in vivo* experimental validation. We are using q-values, as one possible measure of statistical significance, to rank our pairs to verify that the top pairs found by the AA approach are consistent with those found by the GKC approach. Of the original 2,804 CCIs, there are 66 that were found to have an FDR q-value of 0.5 or less. We take these 66 CCIs to represent the "best" of the CCIs found by the AA approach. Note that this set includes some duplicate pairs that are found using more than one kernel width.

Table 3.2 shows, for the top ten csCCIs from the GKC approach, how many times that pair was found in the list of 66 significant CCIs produced by the AA approach. The best csCCIs were found to be significant in more kernel widths in the AA approach. The top three csCCIs from the GKC approach were significant in all eight kernel widths in the AA approach, the 4th through 9th ranked csCCIs were significant in four to six kernel widths each, and a select few of the csCCIs ranked between 10 and 23 were significant in one or two kernel widths in the AA approach. For reference, Table 3.3 presents the complete list of the top 66 insertions that were discovered by the AA approach. Note that the better csCCIs also tended to have better average FDR q-values for their matched pairs from the AA approach. This demonstrates that the ten most significant

csCCI Rank	# Matches from AA approach across kernel widths	Average FDR q-value
1	8	0.054
2	8	0.259
3	8	0.324
4	6	0.382
5	4	0.144
6	5	0.367
7	5	0.395
8	4	0.402
9	5	0.395
10	1	0.499

Table 3.2: Number of matched significant pairs from the AA approach for each of the top ten pairs from the GKC approach and their average FDR q-value.

pairs produced by the GKC approach can be found among the top 66 significant pairs (csCCIs) produced by the AA approach. Table 3.3 also shows that the AA approach produces roughly the same ranking in terms of statistical significance for these pairs.

Computational efficiency

As discussed previously, a major disadvantage of the GKC approach is the exponential computational complexity involved in finding higher order patterns. The AA approach is designed to scale well by efficiently pruning the search space of higher order patterns to examine. Figure 3.3 shows the run times for these two approaches for *RTCGD* along with two other more dense insertional mutagenesis data sets, which we refer to as *mp1* and *mp2*. The run times for the two approaches on the *RTCGD* data set are roughly equivalent, although it should be noted that the GKC approach only finds significant pairs. Thus, the AA approach finds CCIs of all sizes in the same amount of time that it takes the GKC approach to find only CCI pairs.

We did not develop algorithms for finding triples or other higher order patterns using the GKC methodology, but as mentioned earlier the combinatorial explosion of possible patterns would result in an exponential increase in run time, since the GKC approach does not have the search space pruning offered by the AA approach. Additionally, the run times using the *mpl* and *mp2* data sets demonstrate that the AA approach scales much better than the GKC approach for denser data sets, while again also locating

Gene(s) 1	Gene(s) 2	Kernel widths found in	Min q-val	Max q-val	Average q-val	csCCI Rank
Sox4	Hhex	1,2,3,4,5,6,7,8	0.011	0.161	0.054	1
Gfi1	Myc	1,2,3,4,5,6,7,8	0.051	0.498	0.259	2
Rras2	Myc	1,2,3,4,5,6,7,8	0.151	0.498	0.324	3
Hoxa9/Hoxa7	Meis1	1,2,3,4,5,6	0.319	0.466	0.382	4
Gfi1	Myb	4,5,6,7,8	0.259	0.498	0.367	6
Notch2	Sox4	1,2,3,4,5	0.319	0.499	0.395	9
Myb	Myc	5,6,7,8	0.087	0.238	0.144	5
Myc	Pim1	2,3,4,5	0.319	0.499	0.395	7
AA81470/Iqce	Sox4	4,5,6,8	0.331	0.466	0.402	8
Rpa1	Sox4	7,8	0.318	0.437	0.377	15
Sox4	Rreb1	1,4	0.407	0.498	0.452	11
Gfi1	Sox4	4,5	0.498	0.499	0.498	13
Gfi1	Ccnd2	4	0.498	0.498	0.498	16
Sox4	Ifnar1	4	0.498	0.498	0.498	17
Notch2	Rreb1	4	0.498	0.498	0.498	23
Ccnd1	Myc	5	0.499	0.499	0.499	10

Table 3.3: Pairs (described by their probable cooperating genes) found by the AA approach with an FDR q-value less than 0.5; kernel widths they are found in; minimum, average, and maximum FDR q-values from those kernel widths; and rank of the csCCI pair they are matched with from the GKC approach.

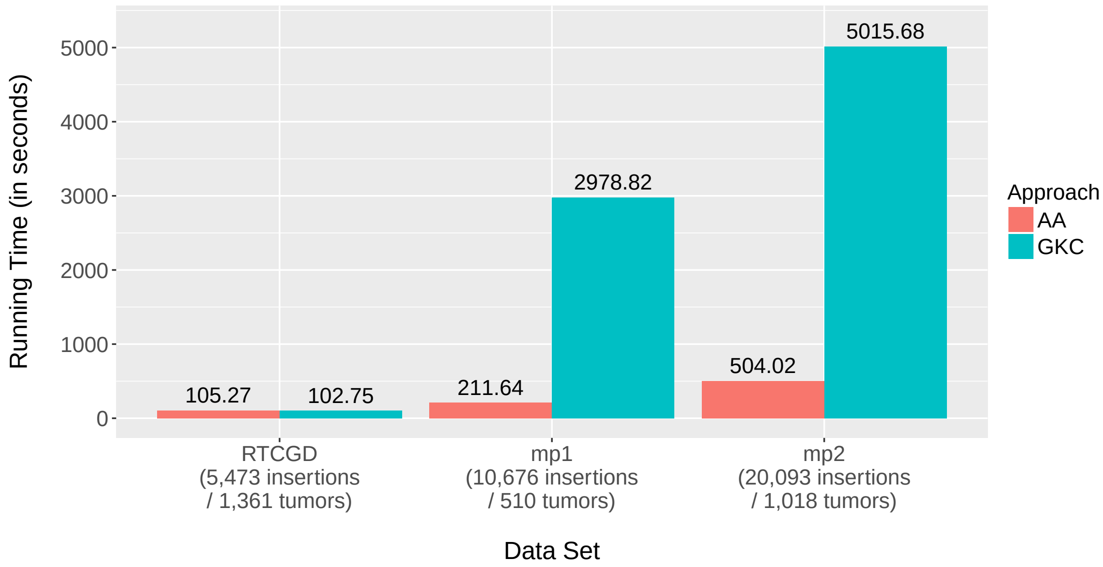


Figure 3.3: Comparison of running times (in seconds) between the AA and GKC approaches for the *RTCGD*, *mp1*, and *mp2* data sets. Experiments are performed using a 2.3GHz AMd Opteron processor. The GKC algorithm used here is an optimized version of that used in de Ridder et al. [55].

higher order patterns instead of just pairs. Note that for the GKC algorithm, we used our own implementation of GKC, which is more efficient than the original one used in de Ridder et al. [55]. Experiments showed that this optimized version performed 16 to 26 times as efficiently, depending on the kernel width.

Our experiments show that there is considerable overlap of the top ten pairs produced by the GKC approach and the top pairs of the AA approach. Furthermore, all of the top 86 pairs found by GKC were also discovered by the AA approach, although not at the same levels of significance (there was one exception explained previously). For the *RTCGF* data set, the AA approach runs in a similar amount of time as the GKC approach while also finding additional higher order patterns. For more dense insertional mutagenesis data sets containing more insertions per tumor, the AA approach is also found to scale better than the GKC approach.

3.5.2 Results from simulated data sets

In order to evaluate our approach on insertional mutagenesis data sets with a known ground truth (i.e. the sets of genes that lead to tumor formation when all faced with concurrent insertions), we created a series of simulated data sets. This was done to evaluate the efficacy of the self-sufficient item set filtering criteria described in Section 3.4.5. Specifically, we are interested in examining how effective the filtering is in terms of reducing final item set calls, as well as in terms of retaining the "true" CCIs after filtering.

Each simulated insertional mutagenesis data set consisted of 100 tumors and 1000 possible genes that can be affected by an insertion. For each data set, we also randomly generated a collection of cancer gene sets that, when all affected by insertions in the same sample, cause that sample to be categorized as a tumor. The number of insertions in each sample was drawn from a Gaussian distribution with a mean of 30 insertions and a standard deviation of 15 insertions. The insertion locations were equally likely to affect any gene window. If for a particular sample there were insertions in every gene of one of the cancer gene sets for the current data set, then that sample would be categorized as a tumor. Samples were repeatedly generated until there were 100 total tumor samples.

Each cancer gene set contained three different genes, and there were no genes that were present in multiple cancer gene sets in the same data set. We simulated four scenarios of simulated data, containing three, five, ten, and twenty cancer gene sets respectively. For each of these scenarios, there were ten iterations of data sets that were generated, and results were averaged across these ten iterations.

We evaluated three different filtering criteria all based upon the concept of self-sufficient item sets. The first approach is to use the full version of self-sufficient item sets, as described in Webb and Vreeken [78]. Because items in insertional mutagenesis data sets should not have strict dependencies (at least in terms of insertion site combinations that are theoretically possible), the requirement for item sets to be non-redundant may not be appropriate for these types of data sets. The second and third approaches remove this requirement from the filtering criteria. However, this leaves a question of how to evaluate independent productivity of item sets. Normally, this is evaluated in the context of all other fully self-sufficient item sets. Our second filtering approach evaluates independent productivity in the context of all other item sets that are productive and independently productive (but can still be redundant). Our third filtering approach still

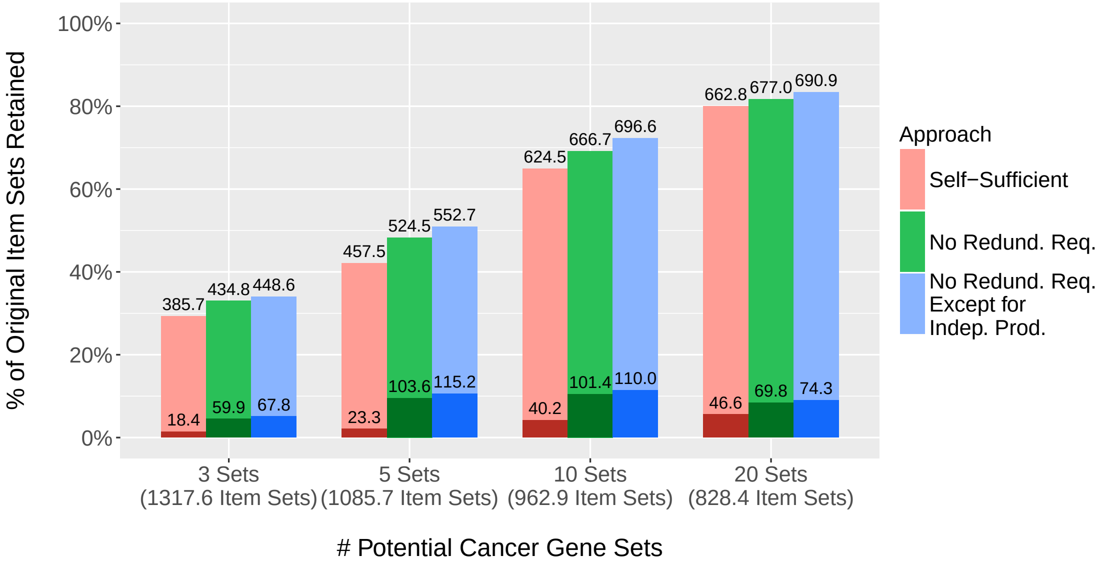


Figure 3.4: Filtering efficiency from simulated insertional mutagenesis data sets. Three filtering approaches are compared on four different groups of data sets. The number of closed frequent item sets from each data set group is displayed along the x axis, and the percentage of item sets remaining after different filtering approaches is plotted (with the actual number remaining displayed on the bars). The lighter colors indicate the item sets after filtering, and the darker colors indicate the item sets that are significant at a 0.05 level after filtering and Bonferroni correction. Results for each data set group are averaged over ten simulation iterations.

evaluates independent productivity in the context of all fully self-sufficient item sets (i.e. item sets that are productive, non-redundant, and independently productive).

Figure 3.4 displays the filtering performance results of these three filtering approaches (colored in order red, green, and blue, respectively) on the four aforementioned simulated data set scenarios. The results for each scenario are the average taken over the ten simulation iterations. In general, as the number of cancer gene sets used to generate the simulated data increases, the number of raw closed frequent item sets return decreases while the percentage of item sets remaining after filtering increases. Within each scenario, the three filtering approaches are reasonably similar in terms of number of item sets filtered. This ranges from $\sim 35\%$ of item sets remaining after filtering for the scenario of three potential cancer gene sets, to $\sim 80\%$ remaining for the twenty potential cancer gene sets scenario.

These results demonstrate that the combination of filtering and significance testing after Bonferroni correction drastically reduces the number of item sets in the final result set. In all cases, less than 15% of all of the original item sets are marked as significant and are returned in the final result set using any version of our self-sufficient item set filtering approach.

We are also interested in examining how many of the cancer gene sets used to generate the simulated data remain in the final result set after filtering and significance testing, since these represent the "true" CCIs that our approach should be identifying. The percentage and number of cancer gene sets marked as significant in all scenarios is shown in Figure 3.5. For the first three scenarios (i.e. three, five, and ten cancer gene sets used to generate the simulated data sets), the third filtering approach (i.e. no redundancy requirement except for independent productivity) had all cancer gene sets in the final result set for every simulation iteration, while the other two approaches often filtered some of these out. In the scenario of twenty cancer gene sets, not all of these were returned in the final result set by the third filtering approach, and performance was more similar between the first and third approaches. Note however that in this case (and for hypothetical scenarios of even more cancer gene sets), not all of the potential gene sets are necessarily present in tumors, thus some may not have a strong signal regardless of filtering.

For the first filtering approach (i.e. self-sufficient item sets), the requirement for non-redundancy often results in true CCIs being filtered out due to a perceived strict dependency amongst genes in the set. In order to be considered non-redundant, each of the genes in the set must occur independently of other genes in the set in at least some tumors, which should not necessarily be a requirement for identifying interesting CCIs. These simulated data sets with known ground truth demonstrate that "real" cancer gene sets can also be redundant item sets.

Both the second and third filtering approaches remove the non-redundancy requirement but evaluate independent productivity in different ways. Surprisingly, the context in which independent productivity is evaluated can have a drastic effect on the percent of "true" CCIs returned. For the second filtering approach (i.e. no non-redundancy requirement), independent productivity is evaluated in the context of item sets that may

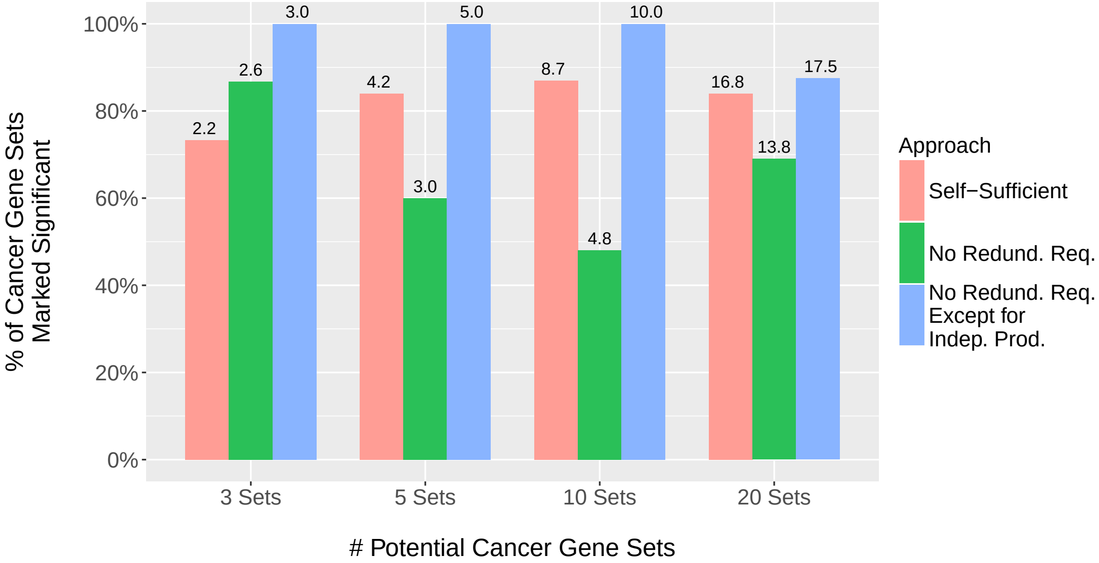


Figure 3.5: Filtering accuracy from simulated insertional mutagenesis data sets. Three filtering approaches are compared on four different groups of data sets. The number of closed frequent item sets from each data set group is displayed along the x axis, and the percentage of “true” cancer gene sets remaining after different filtering approaches is plotted (with the actual number remaining displayed on the bars). Results for each data set group are averaged over ten simulation iterations.

be redundant. This increase the size of the context set, and thus may include extra uninteresting patterns in that context set. “True” CCIs may be filtered out because they are not independently productive in the context of such uninteresting patterns, which may explain why these real patterns of interest are not being returned by the second filtering approach.

In all scenarios, the third filtering approach (i.e. no redundancy requirement except for independent productivity) retained more of the true cancer gene sets than the other two filtering approaches. This, combined with the evidence that the filtering efficiency differences are fairly negligible between approaches (see Figure 3.4), suggests that the third filtering approach is ideal for reducing the size of the result set while also retaining the causal cancer gene sets.

These simulated data sets do not mirror the intricacies and complexity of real-world insertional mutagenesis data sets. More advanced simulation models could be explored in order to further investigate the right filtering approach, but ultimately only experimental wet-lab validation of CCIs from real data sets can be relied on to validate the efficacy of any approach. However, the evidence reported here at least demonstrates that our filtering approach is effective at reducing the number of CCIs in the final result set while likely still retaining the important ones.

3.5.3 Results from *Sleeping Beauty* transposon experiment data sets

We further evaluated our approach on a series of ten insertional mutagenesis data sets. All data sets were developed in mice using the *Sleeping Beauty* transposon systems, but vary in terms of the size (i.e. number of tumors), insertion density, and ratio of tumors to mice. Table 3.4 contains a summary of the characteristics for all ten data sets.

Evaluation using permutation-based FDR approach

After using our approach to generate closed frequent item sets on these data sets and to correct for multi-tumor mice, we first applied our permutation-based FDR strategy for handling multiple hypothesis testing. We use two previously discussed randomization strategies, permute randomization and swap randomization, each over 1000 iterations. The best FDR q-values in each data set’s results are shown in Table 3.5. Data sets *HS*, *TOS*, and *MET* are the only three in which the swap randomization approach could

Data Set	Tumors	Insertions	Insertions / Tumor (mean ± SD)	Mice	Tumors / Mouse (mean ± SD)
CRC [60]	135	16,599	122.96 ± 99.47	47	2.87 ± 2.36
HS [50]	92	1,554	16.89 ± 19.42	36	2.56 ± 1.36
LC [51]	23	22,311	970.04 ± 343.06	13	1.77 ± 1.17
GITP19	51	15,989	313.51 ± 179.84	28	1.82 ± 0.77
GITP53	30	27,479	915.97 ± 372.00	25	1.20 ± 0.41
TOS	23	4,351	189.17 ± 200.10	21	1.10 ± 0.30
NF	267	136,290	510.45 ± 324.09	54	4.94 ± 3.22
PNST	99	32,676	330.06 ± 311.19	62	1.60 ± 0.93
OSQ	122	17,878	146.54 ± 137.11	89	1.37 ± 0.66
MET	127	9,573	75.38 ± 63.66	23	5.52 ± 4.13

Table 3.4: Summary of size and density of *Sleeping Beauty*-generated insertional mutagenesis data sets analyzed.

complete 1000 iterations in under 72 hours. Note that for the *NF* data set, which was the largest data set by far with 136,290 total insertions, the minimum support count threshold was set to twenty instead of three in order to avoid out-of-memory issues.

As noted earlier in Section 3.4.5, the swap randomization approach for permutation is more accurate in terms of conserving row and column margin totals in the binary transaction matrix. However, it is only able to run in a reasonable amount of time for 1000 iterations in the three data sets with the smallest numbers of total insertions. Thus, this permutation strategy does not scale well for larger or more dense insertional mutagenesis data sets.

The permute randomization approach can be run successfully for all ten data sets, but the randomization does not guarantee conservation of the margin totals in the binary transaction matrix, which would mean that the null distribution of insertions generated over the iterations may not be a valid model. There is some evidence that this translates into differences in the resulting patterns. For the *HS* data set, the same ten CCIs top the ranked list of results (sorted by FDR q-value) and the FDR q-values are fairly consistent. Differences occur in the *MET* and *TOS* data set however. Although the top FDR q-values from the two randomization approaches are similar for *MET*, only six CCIs can be found in the top ten results from both, and the top three from the permute randomization approach are outside of the top twenty in the swap randomization approach. For the *TOS* data set, although the top three CCIs are the same in both approaches, there are

Data Set	Best FDR q-value (Permute Randomization)	Best FDR q-value (Swap Randomization)
CRC	0.633	-
HS	<0.001	<0.001
LC	0.639	-
GITP19	0.480	-
GITP53	0.494	-
TOS	0.245	0.561
NF	<0.001	-
PNST	0.635	-
OSQ	0.517	-
MET	0.918	0.967

Table 3.5: Best FDR q-values for *Sleeping Beauty*-generated insertional mutagenesis data sets using the permutation-based FDR based approach and 1000 iterations of both permute randomization and swap randomization. Dashes indicate the randomization iterations did not complete within 72 hours.

major differences in their associated FDR q-values (i.e. 0.245 for permute randomization vs. 0.561 for swap randomization for the top CCI), and there is only an overlap of four between the two sets of top ten CCIs.

Although the permute randomization approach appears to act as a sufficient proxy for swap randomization for the *HS* data set, that does not appear to be the case for larger or more dense data sets. Since the swap randomization approach is computationally infeasible for more dense data sets, the permutation-based FDR approach for handling multiple hypothesis testing may be insufficient for analyzing such large insertional mutagenesis data sets. It should also be noted that regardless of the randomization methodology, the permutation-based FDR approach in general does not account for multi-tumor mice in the null distribution model. This may lead to spurious results for data sets with higher average tumors per mice, such as *MET*.

Evaluation using self-sufficient item set filtering approach

We developed the self-sufficient item set filtering approach for multiple hypothesis testing in order to address the issues with scalability for the permutation-based FDR approach. This was applied to the same ten SB insertional mutagenesis data sets. Specifically, we used the filtering criteria that was shown to be most effective in Section 3.5.2. Item sets

Data Set	Closed Item Sets	$\|I\| \leq 10$		$sup^*(I) \geq 3$		After Filtering	
CRC	3,512	3,512	(100.0%)	3,228	(91.9%)	3,228	(91.9%)
HS	132	126	(95.5%)	82	(62.1%)	82	(62.1%)
LC	63,377	47,423	(74.8%)	62,730	(99.0%)	47,306	(74.6%)
GITP19	16,567	15,292	(92.3%)	15,724	(94.9%)	14,654	(88.5%)
GITP53	190,742	103,379	(54.2%)	190,592	(99.9%)	103,378	(54.2%)
TOS	110	110	(100.0%)	105	(95.5%)	105	(95.5%)
NF	642,916	642,916	(100.0%)	75,206	(11.7%)	75,206	(11.7%)
PNST	202,004	191,277	(94.7%)	200,868	(99.4%)	190,445	(94.3%)
OSQ	5,591	5,589	(<100.0%)	5,487	(98.1%)	5,486	(98.1%)
MET	1,212	1,144	(94.4%)	371	(30.6%)	371	(30.6%)

Table 3.6: Initial counts of closed frequent item set results for *Sleeping Beauty*-generated insertional mutagenesis data sets and counts after initial filtering. Item sets are filtered by size (must be no larger than ten genes) and by modified support count (must be no larger than three, except for *NF* which uses a support count threshold of twenty). The last column contains the number of item sets that pass both filtering criteria.

must be productive as well as independently productive in the context of all item sets that are fully self-sufficient. Because of the requirement to examine every possible subset of item sets when evaluating their productivity, the scalability of a self-sufficient item set filtering approach is limited by large item sets. Thus we apply an initial filtering of closed frequent item sets by removing all item sets that contain more than ten genes.

Table 3.6 shows the number of unfiltered closed frequent item sets, the number of those item sets meeting the size threshold, the number meeting the modified support count threshold after correcting for multi-tumor mice, and the total number of item sets remaining after both of these initial filterings. Again, for the *NF* data set we use a support count threshold of twenty instead of three. The largest initial filtering by percentage is done for the *NF* and *MET* data sets, with 11.7% and 30.6% of the item sets remaining after initial filtering, respectively. For these two data sets this significant filtering is directly attributable to item set support counts that fall under the threshold after modifying them for multi-tumor mice. This is not surprising, as *NF* and *MET* are the two data sets with the most tumors per mouse (4.94 and 5.52 respectively). This demonstrates the importance of accounting for multi-tumor mice due to the large numbers of spurious patterns that can result otherwise.

Data Set	**Filtered Item Sets**	**Productive Item Sets**	**Independently Productive Item Sets**	**Indep. Prod. w/ Corrected p-value $\leq$ 0.05**
CRC	3,228	2,674 (82.8%)	2,376 (73.6%)	405 (12.5%)
HS	82	76 (92.7%)	65 (79.3%)	42 (51.2%)
LC	47,306	781 (1.7%)	698 (1.5%)	100 (0.2%)
GITP19	14,654	128 (0.9%)	104 (0.7%)	32 (0.2%)
GITP53	103,378	0 (0.0%)	0 (0.0%)	0 (0.0%)
TOS	105	85 (81.0%)	85 (81.0%)	30 (28.6%)
NF	75,206	58,833 (78.2%)	31,222 (41.5%)	1,651 (2.2%)
PNST	190,445	161,885 (85.0%)	131,631 (69.1%)	72,535 (31.8%)
OSQ	5,486	4,508 (82.1%)	3,775 (68.8%)	940 (17.1%)
MET	371	257 (69.2%)	245 (66.0%)	21 (5.7%)

Table 3.7: Independent productivity filtering and statistical significance filtering for *Sleeping Beauty*-generated insertional mutagenesis data sets.

After initial filtering by item set size and modified support threshold, productivity and independent productivity is calculated for all remaining item sets. Table 3.7 contains the number of productive item sets, number of independently productive item sets, and the number of item sets with a Bonferroni-corrected p-value less than 0.05 applied after filtering via independent productivity. Filtering via productivity is particularity effective for the *LC* and *GITP53* data sets, and actually removes all item sets from the *GITP19* data set. *LC* and *GITP19* have significantly more insertions per tumor than the other data sets (see Table 3.4), meaning there is more likelihood of having the “real” patterns hidden among more noise. This may be a factor in why the vast majority of item sets from these more dense data sets are non-productive. Filtering via independent productivity is most effective for the *NF* data set, in which the 58,833 productive item sets are reduced to 31,222 that are independently productive.

After applying all filters and using Bonferroni correction to account for multiple hypothesis testing on the remaining independently productive item sets, we are left with a much more manageable result sets for nearly all data sets. Furthermore, this was achievable while still keeping the initial support count threshold low (three in all cases except for *NF*), which means that rare CCIs can still be discovered if they also pass filtering criteria and tests for significance. For the nine data sets that used a support count threshold of three, the final result set did indeed contain CCIs with modified support counts of just three in all cases but for *LC*, where the minimum modified support count among the final result set was five. Notably, there are still 72,535 final CCIs from the *PNST* data set after all filtering steps and significance testing. *PNST* is also the data set that contained the most candidate item sets after initial filtering. Future work will explore additional filtering possibilities for such cases.

Histiocytic sarcoma results

We next examine more closely the results from two of the ten SB insertional mutagenesis data sets, the *HS* and *LC* data sets. The *HS* data set was derived from 92 tumors in 36 mice in which SB transposon insertional mutagenesis was targeted to myeloid cells in order to develop histiocytic sarcoma [50]. Histiocytic sarcoma is a rare form of cancer that develops in the immune system, and its genetic mechanisms are poorly understood.

The top ten CCIs from our analysis (after applying the self-sufficient item set filtering approach for handling multiple hypothesis testing and sorting by Bonferroni-corrected p-values) are shown in Table 3.8. These CCIs all have a modified support count of either three or four (i.e. after correcting for multi-tumor mice), and range in size from five to eight genes. There are fifteen unique genes across these ten CCIs. Notably, only five of these genes (*Bach2*, *Erg*, *Ncoa2*, *Raf1*, and *Serpinf1*) are considered CISs by the TAPDANCE algorithm [54]. Thus, our approach of mining for higher order patterns of co-occurring insertion sites discovers genes that would not be identified when considered for significance on their own.

The majority of these fifteen genes (*Bach2*, *Dctn4*, *Dennd2c*, *Erg*, *Mitf*, *Ncoa2*, *Pcf11*, *Raf1*, and *Serpinf1*), including at least one in each of the top ten CCIs, have also been found to be associated with cancer in the the Candidate Cancer Gene Database (CCGD) [80]. In particular, mutations in *Mitf* and *Raf1* have been shown to be drivers for other blood-related cancers [81, 82]. The CCIs generated by our approach may correspond with unknown genetic interactions between known cancer drivers and other genes that can distinguish histiocytic sarcoma from other cancer types. Further experiments are needed in order to confirm a causal relationship between any of these CCIs and histiocytic sarcoma.

Item Set	Genes	Item Set Size	Modified Support Count	p-value
#1	*Dctn4*, *Dennd2c*, *Erg*, *Ncoa2*, *Pcf11*, *Serpinf1* *Basp1*, *Bach2*	8	3	$<1 \times 10^{-15}$
#2	*Dctn4*, *Dennd2c*, *Erg*, *Ncoa2*, *Pcf11*, *Basp1* *Bach2*	7	3	$<1 \times 10^{-15}$
#3	*Hspb6*, *Dctn4*, *Dennd2c*, *Ncoa2*, *Pcf11*, *Serpinf1*	6	3	$<1 \times 10^{-15}$
#4	*Dctn4*, *Dennd2c*, *Ncoa2*, *Pcf11*, *Serpinf1*	5	4	$<1 \times 10^{-15}$
#5	*Col27a1*, *Dennd2c*, *Kif2c*, *Mitf*, *Pcf11*, *Raf1*, *Serpinf1*, *Basp1*	8	3	$<1 \times 10^{-15}$
#6	*Col27a1*, *Dennd2c*, *Grlf1*, *Kif2c*, *Ncoa2*, *Pcf11*, *Serpinf1*	7	3	7.938×10^{-14}
#7	*Dctn4*, *Dennd2c*, *Grlf1*, *Kif2c*, *Ncoa2*, *Pcf11* *Raf1*, *Serpinf1*	8	3	1.299×10^{-13}
#8	*Dctn4*, *Denndc2*, *Grlf1*, *Kif2c*, *Ncoa2*, *Pcf11*, *Raf1*	7	3	1.660×10^{-13}
#9	*Dennd2c*, *Kif2c*, *Pcf11*, *Raf1*, *Serpinf1*	5	4	5.102×10^{-12}
#10	*Dennd2c*, *Kif2c*, *Pcf11*, *Plcl1*, *Raf1*, *Serpinf1*	6	3	5.131×10^{-12}

Table 3.8: Top ten CCIs (by Bonferroni-corrected p-value) for the *HS* data set.

Lung cancer results

We also examined the top ten results from the *LC* data set (see Table 3.9). This data set was obtained from SB transposon that generated 23 lung cancer tumors in 13 mice [51]. Mice were *Pten*-deficient, which increased the likelihood of tumor development in the lungs. This means that CCIs resulting from this analysis may also interact with *Pten* deficiency in potential mechanisms for tumorigenesis.

The results in Table 3.9 are also generated via the self-sufficient item set filtering approach for handling multiple hypothesis testing. The modified support counts for the top ten CCIs ranges from four to seven, and the item set sizes range from seven to ten. There are a total of 27 unique genes across all ten CCIs.

Similar to the *HS* data set, few of the genes in this result set are significant on an individual level, and the majority of the genes are known to be associated with cancer. Only the genes *Svil* and *Tle4* are marked as CISs by TAPDANCE. CCGD lists 16 of the 27 genes (*Acvr2a*, *Arrdc3*, *Cntn4*, *Dnajb9*, *Grm7*, *Lpp*, *Lrba*, *Lrp1b*, *Meocom*, *Pcdh9*, *Pixna4*, *Ptprm*, *Svil*, *Tbl1x*, *Tle4*, and *Zbtb20*) as having cancer associations. *Svil*, which is part of the gene set for three of the top ten CCIs, is involved in signal regulating for p53, a well-known tumor suppressor protein.

The results from both the *HS* and *LC* data sets are promising in that they contain gene sets with many genes not significant on an individual level (thus demonstrating the benefits of mining for higher order patterns) and with many genes associated with cancer (thus demonstrating the biological relevance of resulting patterns). Further work is needed to validate functional cooperation between genes in the same CCI, as well as relationships with tumorigenesis. Validated findings could eventually be used as detection signals or to develop targeted therapies.

3.6 Discussion and Future Work

In this chapter we have mostly described insertional mutagenesis as a process in which insertions have a potentially detrimental affect on a gene and its expression. However, it may be possible to activate a gene or increase its expression via insertional mutagenesis as well [49]. Patterns produced via our framework should be treated as potentially interesting sets of genes to investigate in general, without assumption as to the mechanism. It

Item Set	Gene	Item Set Size	Modified Support Count	p-value
#1	*4930474G06Rik*, *Epha3*, *Frm7*, *Lrfn5*, *Lrrc4c*, *Meocom*, *Tbl1x*, *Tenm4*, *Tle4*, *Zbtb20*	10	6	3.174×10^{-7}
#2	*4930474G06Rik*, *Cntn4*, *Grm7*, *Lrrc4c*, *Plxna4*, *Tbl1x*, *Tenm4*, *Zbtb20*	8	6	9.187×10^{-6}
#3	*4930474G06Rik*, *Cntn4*, *Cobl*, *Diap2*, *Grm7*, *Pcdh17*, *Pcdh9*, *Svil*, *Tenm4*	9	6	1.037×10^{-5}
#4	*4930474G06Rik*, *Cobl*, *Diap2*, *Dnajb9*, *Lphn3*, *Pcdh9*, *Plxna4*, *Zbtb20*	8	7	1.088×10^{-5}
#5	*4930474G06Rik*, *Grm7*, *Lrfn5*, *Lrp1b*, *Lrrc4c*, *Ptprm*, *Tbl1x*, *Tenm4*, *Tle4*	9	6	1.438×10^{-5}
#6	*4930474G06Rik*, *Grm7*, *Lpp*, *Lrfn5*, *Lrp1b*, *Lrrc4c*, *Meocom*, *Ptprm*, *Tenm4*, *Tle4*	10	5	5.893×10^{-5}
#7	*4930459L07Rik*, *Acvr2a*, *Cntn4*, *Lrp1b*, *Lrrc4c*, *Nudt12*, *Svil*, *Tenm4*	8	4	8.765×10^{-5}
#8	*4930474G06Rik*, *Acvr2a*, *Cobl*, *Grm7*, *Lrp1b*, *Lrrc4c*, *Pcdh17*, *Ptprm*, *Tenm4*	9	6	1.058×10^{-4}
#9	*4930459L07Rik*, *Arrdc3*, *Cntn4*, *Lrba*, *Lrrc4c*, *Svil*, *Tenm4*	7	4	1.107×10^{-4}
#10	*4930474G06Rik*, *Acvr2a*, *Cobl*, *Grm7*, *Nudt12*, *Pcdh17*, *Pcdh9*, *Ptprm*, *Tenm4*	9	6	1.688×10^{-4}

Table 3.9: Top ten CCIs (by Bonferroni-corrected p-value) for the *LC* data set.

may be the case that the knockout of all genes in the set results in tumor formation, but it could also be that some of the genes are in fact being activated or positively expressed by their associated insertions. This also raises the possibility of insertions with different affects mapping to the same item in the transaction matrix (i.e. one insertion location inactivates the gene while a different insertion location activates it), which could lead to spurious patterns. Future work will investigate this problem and potential solutions to it.

Another potential concept to explore is the incorporation of genetic pathway information to prune the search space or identify patterns of interacting genetic pathways. For example, there may be genes X and Y that are critical components of the same genetic pathway. If either is disrupted, the pathway is unable to function properly. The function of this pathway may interact with other genes or pathways (A and B, for instance) in such a way that the disruption of all of them results in tumor formation. In such a case, patterns $\{A, B, X\}$ and $\{A, B, Y\}$ would both result in tumor formation via the same mechanism. Incorporating this genetic pathway information would allow for identification of this type of situation, resulting in potentially novel significant patterns that may not have been significance when evaluated separately. Furthermore, using genetic pathway information to cluster insertion locations by affected pathways rather than affected genes or genomic regions may be an effective approach to pruning the search space and discovering more robust patterns.

Future work will also investigate the effects of mining for different types of item sets, such as hyperclique patterns. Hyperclique patterns add an additional constraint onto item sets, requiring that each has a higher support count than all possible subsets [83, 84]. The purpose of hypercliques is to filter out item sets that are primarily driven by one frequently-occurring item and instead focus on sets of items that are all strongly associated with each other. It's possible that this would be redundant with our approach of filtering by the self-sufficient item set criteria. Nonetheless, hypercliques may present another method to further reduce spurious item set results generated from high density insertional mutagenesis data.

We also intend to incorporate and adapt additional ideas from existing approaches into our pipeline. For example, working with the raw next-generation sequencing data directly (as is done with the TAPDANCE software [54]), rather than already-processed insertional mutagenesis data, would allow for greater flexibility and introduce opportunities to incorporate signal strength of insertion locations (i.e. how much confidence there is in the presence of individual insertions). We also plan to incorporate information about insertional bias into our statistical model, such as chromosomal preferences for insertion locations and TA dinucleotide locations for SB-generated data sets, as is done by Monte Carlo-based techniques [47, 48].

Finally, a key part of future work will be in investigating the significance and broader impact of candidate CCIs from biological, genetics, and medical perspectives. CCIs generated by our approach must be validated *in vitro* or *in vivo* in mice to verify a causal (rather than associative) relationship with tumorigenesis, as well as to better understand the functional impact of sets of co-occurring mutations. Functionally-validated CCI-tumor relationships can be further investigated for potential opportunities in cancer therapies and/or cancer detection tests in humans.

Chapter 4

Assembling Personal Genomic Sequences

4.1 Introduction

Advancing our understanding of the genetics of cancer relies not just on discovering the genetic mutations driving tumorigenesis, but also on studying how these mutations affect gene regulation or gene expression. These types of analyses depend upon examining features found throughout the whole genome, including non-coding regions and other genomic sequences previously thought to be inconsequential (i.e. what was once referred to as "junk DNA"). Gene regulation and gene expression can be measured and analyzed using next-generation sequencing technologies known as ChIP-seq and RNA-seq, respectively. ChIP-seq selectively sequences DNA that is involved in certain protein interactions, with the goal of locating transcription factors or regions of gene regulation. RNA-seq uses next-generation sequencing to reveal a snapshot of the RNA molecules present in a cell, which can be used as a measurement of a gene's expression level. Standard analysis pipelines for both of these technologies include aligning reads to a reference genome.

Usually, the standard human reference genome can be used as the reference genome during RNA-seq and ChIP-seq analyses. When studying tumor cells however, the underlying genomes are often so mutated or rearranged that the standard reference is not a close enough approximation. This process is greatly improved if that reference genome is an accurate reflection of the actual genome being sampled. For instance, ChIP-seq analysis was recently shown to be more effective if aligned using a personal genome as opposed to a reference genome [8]. The 1000 Genomes Project also recently found that using personalized reference genomes resulted in marked differences in measured exon expression levels from RNA-seq analysis, many of which were due to accounting for personal structural variants in particular [85]. Thus, there exists a need for algorithms that assemble tumor genomes using normal DNA sequencing reads, which can then be used as a "personalized" reference for downstream analyses such as ChIP-seq and RNA-seq.

One way to approach the assembly problem is to do alignment-based assembly, in which reads are aligned to an existing reference genome using a short-read aligner such as BWA [11], Bowtie [15], or GSNAP [16]. Small mutations are then determined by consensus amongst the aligned reads and used to generate the new personalized reference genome. Although this is usually efficient, assembly of highly-mutated genomes (such as those found in tumors) must also account for the presence of structural variants, which are too complex to be accounted for by an alignment-based assembly. Another approach is to use *de novo* assembly to create personalized reference genomes, which involves using just the raw read data and their overlap patterns to "reassemble" the underlying genomic sequence. This is somewhat akin to how the original human reference genome from the Human Genome Project was assembled. This class of assembly algorithms is more effective for assembling highly-mutated sequences, but is also extremely inefficient and requires deep sequencing in order to achieve acceptable accuracy.

In recent years, the significant genetic impact of structural variation (i.e. deletions, insertions, duplications, inversions, or translocations of large segments of DNA) has become more understood. These structural variants (SVs) often account for a large portion of the genetic landscape of mutations present in tumors, especially due to the genomic instability that is one of the principle characteristics of cancer [37, 38]. One recent study found that as many as 61% of all actionable mutations from a sample of 2,221 cancer patients were SVs [86]. SVs have also been shown to contribute heavily to the genetic variation between normal individuals [87, 29]. Thus, strategies for assembling personal genomic sequences should be able to account for a wide and diverse range of SVs.

Alignment-based assembly approaches do not account for SVs, and *de novo* assembly can often be computationally inefficient, as well as inadequate for assembling repetitive regions in the genome. A variety of techniques have also been developed in recent years that combine aspects of both approaches [88, 89, 90, 91, 92]. Although these hybrid *de novo* / alignment-based assembly programs can overcome some of the pitfalls of using either approach on their own, there remain additional challenges. In particular, computational efficiency is still unideal due to unnecessary operations in assembling concordant regions, and certain SV types are still difficult to assemble, such as tandem duplications.

In many cases, personal genome assembly is also complicated by sample heterogeneity, such as tumor samples with subpopulations of somatically-acquired variants that are present in only a subset of the sequencing data. Genomic instability and rapid division of cancer cells can lead to a high degree of cellular heterogeneity within tumor tissue. Sequencing data from tumor samples will usually contain mixtures of DNA from diverse tumor cell populations, as well as from normal somatic cells, each with their own set of variants. The assembly programs currently used to create personal genomic sequences often assume sample homogeneity and are plagued by an inability to handle non-universal variants (see Figure 2.1 for an example illustration), but these tools must account for this type of data if they are to be useful for studying cancer genomics.

In order to address these existing challenges in personal genome assembly for tumor genomes, we have developed SHEAR (Sample Heterogeneity Estimation and Assembly by Reference), an open-source and easy-to-use software package that predicts variants (including SVs, SNPs, and small INDELs) and assembles personal genomic sequences using those predictions [93]. SHEAR creates personal genomic sequences by predicting variants, including SVs, and generating the new genomes based on the existing reference after correcting for those variants directly. A key component of this framework is a local realignment strategy for correcting errant alignment near probable SV breakpoints. SHEAR also detects variants in small subpopulations of a sequencing sample and estimates the heterogeneity level for those variants (i.e. the frequency of the variant in the sample), thus making it ideal for applications to tumor samples.

In this chapter we will describe the SHEAR framework in general, including the pipelines for variant detection and personal genome assembly. The variant frequency estimation framework, an additional component of SHEAR, will be presented later in

Chapter 5. The rest of this chapter will be organized as follows. In Section 4.2, we present an overview of related work in the area of personal genome assembly. Section 4.3 describes the SHEAR framework and our proposed approach for personal genome assembly. Experimental results that demonstrate the efficacy of SHEAR compared with a leading approach will be presented in Section 4.4. Finally, we conclude with a discussion of issues and ideas for future work in Section 4.5.

4.2 Related Work

If there is a closely-related reference genome available, alignment-based assembly is a simplistic way to perform assembly of personal genomes. The sequenced reads are aligned to the reference genome using a short-read aligner, and SNPs and small INDELs are determined by consensus amongst the aligned reads. A common framework for SNP/INDEL calling is the GATK Best Practices framework developed by DePristo et al. [94]. One important step in this process is the local realignment of reads containing INDELs to improve the signal and accuracy for INDEL predictions. See also Nielsen et al. [95] for a further review of SNP/INDEL calling. Although this is usually efficient and highly accurate, complete personal genomic sequences must also account for the presence of SVs, which is not possible using alignment-based assembly.

The opposite extreme of approaches is to join together all the sequenced reads, like pieces of a puzzle, by determining how they overlap with one another. This class of assembly algorithms, known as *de novo* assembly, does not require a reference sequence and is useful for assembling regions that are significantly different from the available reference genome, such as novel insertions. However, *de novo* assembly may struggle to properly assemble repetitive regions and can be extremely inefficient at the high coverage levels often required for assembling whole genomes. Examples of global *de novo* assembly algorithms include Velvet [96], SOAPdenovo [97], and ALLPATHS-LG [98].

Algorithms have recently been developed that combine aspects of both alignment-based assembly and *de novo* assembly. Seq-Cons [88] and LOCAS [89] use localized versions of *de novo* assembly in order to assemble reads in separate blocks determined by the area of the reference that they are first aligned to, rather than trying to determine possible overlaps between all of the reads globally, without a preliminary alignment. A similar approach was also shown to be successful in assembling several variant strains of *Arabidopsis thaliana* from a related reference genome [90]. RACA [91] uses a reference genome to arrange the scaffolds that are first produced through *de novo* assembly, but also requires multiple outgroup genomes (i.e. from other closely related species) as input.

In contrast to the above methods, which can be thought of as either "global *de novo* assembly followed by alignment" or "alignment followed by local *de novo* assembly", IMR/DENOM is a reference-guided assembly approach that combines alignment-based assembly and *de novo* assembly in parallel and merges the results [92]. The alignment-based half of the algorithm, IMR, is an iterative procedure that creates an alignment to the original reference sequence using Stampy [99], generates a new reference sequence from consensus variants in the alignment, realigns the paired-end reads to the new reference sequence, and repeats this procedure until convergence. DENOM takes contigs that are assembled *de novo* using SOAPdenovo [97] and aligns them to the reference in order to handle larger SVs, such as novel insertions not present in the reference sequence. The results of these two approaches are then merged to generate a personal genomic sequence.

All of these assembly programs assume sample homogeneity, resulting in unexpected behavior when sequencing samples contain variants only present in subpopulations of the cells, and thus are difficult to apply to heterogeneous sequencing samples from tumor DNA. Certain types of SVs, such as tandem duplications, also remain a challenge. Additionally, these approaches are often inefficient in the context of personal genome assembly due to the large amount of redundant operations performed, such as the multiple alignments of every read in IMR, or the assembly of every read in reference-guided assemblers such as Seq-Cons or LOCAS. A more efficient approach for generating personal genomic sequences might be to leverage the specialized ability of pre-existing SV detection programs to locate individual SVs and address them directly, rather than hoping to discover their signature through *de novo* assembly (which is made more difficult in the presence of heterogeneous sequencing samples). A variety of SV detection algorithms have been developed over the last few years which can serve for this purpose [20, 21, 22, 23, 24, 25, 26, 27, 28, 30, 31, 32, 33, 34, 35, 36]. For further reviews of SV detection algorithms, see Medvedev et al. [17], Alkan et al. [18], and Pirooznia et al. [19].

4.3 Proposed Approach

4.3.1 SHEAR pipeline overview

In this chapter, we propose a new software package called SHEAR (Sample Heterogeneity Estimation and Assembly by Reference) which predicts variants, accounts for heterogeneous variants by estimating their frequencies in the sequencing sample, and generates personal genomic sequences to be used for downstream analysis [93]. Our novel contributions come in the form of two key concepts that integrate and expand several pre-existing programs. The first is a pipeline for correcting soft-clipping errors in the alignment that occur at the breakpoints of candidate SVs. Although these alignment errors are often minor, correcting for them improves the reliability of SVs predictions, as well as the accuracy of our SV frequency estimations. The second component is a novel scheme for estimating the variant frequencies for SV predictions. This is done by comparing the soft-clipped reads to the spanning reads at SV breakpoints. An illustration of SHEAR's overall workflow can be see in Figure 4.1.

SHEAR has two "modes", or operations, that it is used for: SHEAR-SV and SHEAR-Assemble. Variants can be predicted and their respective frequencies estimated using the SHEAR-SV component. SHEAR-Assemble takes a set of variant predictions, along with an existing reference genome, and outputs a new personal reference genome using that information. Both components rely upon a pipeline involving an iterative local realignment process to obtain accurate soft-clipping at SV breakpoints, and thus result in more accurate assembly as well as more accurate variant frequency estimations.

Unlike *de novo* assemblers or the iterative alignment-based assembly done by IMR, SHEAR works by doing a one-time global alignment which is used to predict variants (including SVs), followed by generating a personal genomic sequence based on the existing reference after correcting for those variants. This can be thought of as an alignment-based assembly approach, except with the additional component of leveraging SV predictions in addition to just SNPs and INDELs. After reads are aligned to the reference genome, SV types and locations are estimated using an SV prediction algorithm. Since the original alignment is not aware of any present SVs, alignment near SV breakpoints can be imprecise (see Figure 4.3). Once SVs are predicted, the reads neighboring candidate SV breakpoints are realigned using a local alignment algorithm. New or refined SVs can

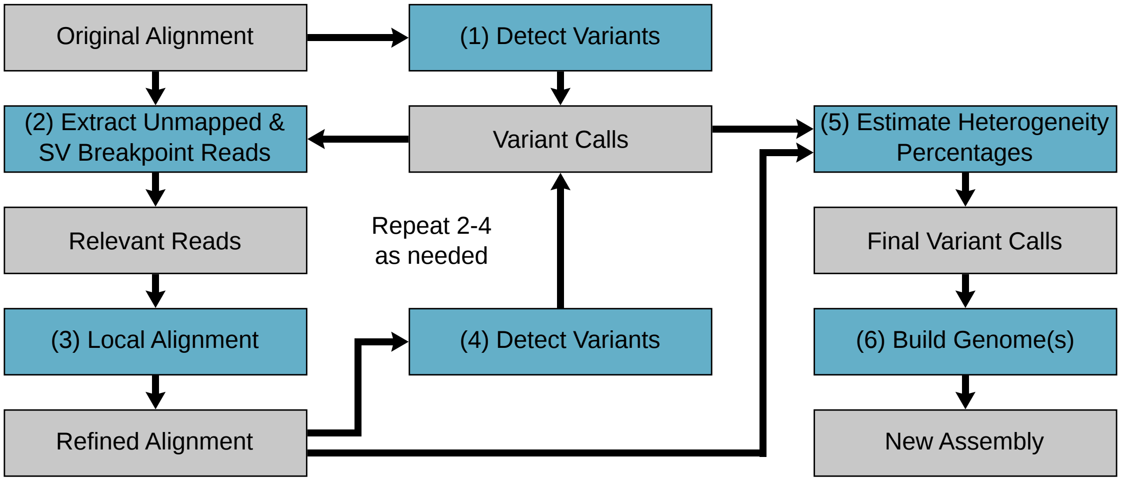


Figure 4.1: SHEAR workflow diagram. **(1)** SHEAR's workflow begins by using an SV prediction algorithm to predict the locations of SVs from the original SAM/BAM alignment file. **(2)** Reads neighboring the breakpoints of the predicted SVs, as well as all unmapped reads, are then extracted from the original alignment. **(3)** These extracted reads are then realigned using a local alignment algorithm to improve the soft-clipping accuracy near the breakpoints. Breakpoint extracted reads are aligned in their original neighborhoods, and unmapped extracted reads are aligned against the whole reference sequence. **(4)** SVs are again predicted from the new alignment, which contains only the realigned reads near the original candidate breakpoints, as well as reads that were initially unmapped but have been realigned using the local alignment algorithm. The new SV predictions will potentially include new SVs and refined breakpoints of previously predicted SVs. Steps 2–4 may be repeated as necessary to pick up new SV events, and will usually only need to be repeated 2–3 times before SV predictions remain constant. **(5)** Using the refined SV breakpoints, the variant frequency (or heterogeneity percentage) of each SV is estimated by comparing the soft-clipped and spanning reads at the breakpoints. This calculation varies depending on the SV type (see Section 5.3). **(6)** Finally, a new personal genomic sequence is created by using the predicted variants to directly modify the original reference sequence.

be predicted following realignment, and this can continue in an iterative process until convergence. Finally, the resulting SVs with their high-precision breakpoints, along with the small SNPs and INDELs determined via traditional alignment-based assembly, are used to directly modify the original reference genome to create a new personal genome. Thus, SHEAR-Assemble is able to accurately assemble complex SV regions while also maintaining efficiency by focusing time-consuming operations only on divergent regions (i.e. SV breakpoints) instead of concordant regions.

SHEAR is written in the Java programming language and is implemented using the Genome Analysis Toolkit (GATK) framework [100] to provide efficient data access and for ease of parallelization. Additional tools are used from SAMtools [101] and Picard [102].

4.3.2 SV prediction

SVs are predicted from the alignment using an SV prediction algorithm, such as CREST [26] or DELLY [33]. Any algorithm can be used for this purpose, with the only requirements being that it predicts breakpoints at base pair resolution so that it can be used for accurate and precise assembly, and that it predicts variants in a heterogeneous sequencing sample. Additionally, SV prediction algorithms that use the split-read approach for their prediction methodology are ideal because prediction accuracy can benefit from SHEAR's iterative local realignment procedure at SV breakpoints.

SHEAR's framework is designed so that new SV prediction programs can be easily incorporated. Thus, as as the state of the art for SV prediction improves, so too will SHEAR's assembly performance. We designed an abstract Java class, `VariantCaller`, in the SHEAR source code that can be used to incorporate SV prediction algorithms. SV prediction algorithms can be added by creating new Java classes that extends the `VariantCaller` abstract class, specifically by implementing the following two method interfaces:

```
public abstract String
    runVariantCaller(String bamFile, String outputPrefix,
                        List<String> options);
public abstract TreeSet<Variant>
    getVariants(String variantCallerOutputFile);
```

The runVariantCaller method calls the SV prediction program itself. The input is a string containing the path to the input BAM alignment file, a string containing the prefix to use for output files, and a list of strings that can provide an arbitrary number of additional arguments to pass to the external program. Additional information, such as the location of the reference genome being used, is available within the VariantCaller class. The return value is a string containing the path to the SV prediction output file (often a VCF file, or else some other program-specific format)

The getVariants method takes as input the path to the file containing SV predictions (i.e. return value of runVariantCaller) and returns a list of Variant objects. This method contains the parsing logic used to convert the specific type of SV prediction output file to a list of Variant objects.

In the latest version of SHEAR (i.e. v1.1.2), we have implemented the CREST [26] and DELLY [33] SV prediction algorithms for available use. CREST uses a split-read methodology for SV prediction, whereas DELLY uses both split-read and read-pair information. CREST was selected for this purpose in the original versions of SHEAR because it handles many different SV types (with good support for tandem duplications in particular), and is designed for somatically-acquired variants as well as germline variants. Native support for DELLY was added to the SHEAR distribution in more recent versions (i.e. v1.0.0+) due to its improved computational efficiency in comparison with CREST, which can be essential when analyzing whole-genome or deep sequencing samples.

4.3.3 Breakpoint microhomology

SHEAR-Assemble also contains logic to account for microhomology at SV breakpoints. Breakpoint microhomology is when an SV has sequences of identical bases at the two breakpoints of the junction. This can be common in SVs, specially when the mechanism of mutation is microhomology-mediated break-induced repair or form stalling and template switching [103].

An example of breakpoint microhomology for a deletion can be seen in Figure 4.2. In this example, there is a deletion on the variant genome of bases 3001–6000 from the reference genome. However, there is a 5 bp microhomology at the deletion breakpoints, highlighted in yellow, where the bases GAATT are present at both bases 3001–3005 and bases 6001-6005. The fusion sequence demonstrates that the size of the deletion is

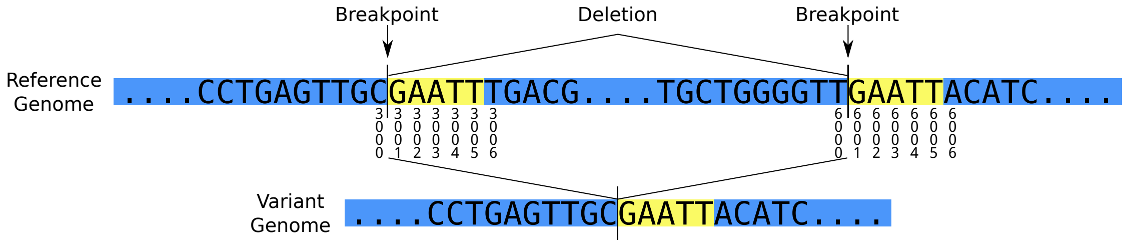


Figure 4.2: Example of breakpoint microhomology is a deletion. The variant genome (below) has a 3000 bp deletion relative to the reference genome (above). Breakpoint microhomology is highlighted in yellow. Genomic coordinates are displayed underneath bases around the breakpoint.

3000 bp, but the breakpoint microhomology makes it ambiguous as to how to report the coordinates of the deletion. For example, it is equivalent to say either that there is a deletion of bases 3001-6000 or that there is a deletion of bases 3006-6005 (or any other coordinate ranges in between).

In our pipeline, we convert each variant call into a "canonical" representation which uses the 5'-most coordinate for the first breakpoint. This makes our results consistent and also ensures that duplicate results are not reported in the final result set. For example, there is no guarantee as to which breakpoint coordinates DELLY will report in a situation such as that depicted in Figure 4.2. By always converting SV calls to their canonical representation, we ensure that the same SV is not predicted a second time with different coordinates in a later iteration of local realignment and SV prediction.

When using CREST for SV prediction, additional steps must be taken to account for breakpoint microhomology. CREST does not consider breakpoint microhomology when reporting SV predictions, and the deletion in Figure 4.2 would actually be reported as a deletion of bases 3006–6000, which is incorrect as it excludes both ends of homologous bases. When processing CREST results, we first detect the amount of breakpoint microhomology and then update breakpoint coordinates accordingly. This microhomology detection is done by checking the number of bases on either side of the breakpoint that the consensus sequence aligns with and comparing it against the length of the consensus sequence itself. Note that the consensus sequence is reported by CREST's output format as part of an SV result entry. If the length of the consensus sequence is shorter than the sum of the aligned bases on either side of the breakpoint, then that means that part of the consensus sequence is aligned twice. These are the homologous bases.

4.3.4 Refining soft-clipping at candidate SV breakpoints

After SV prediction and breakpoint microhomology detection, we are left with an initial alignment on the whole sequence (referred to here as the original alignment) as well as a set of predicted SVs, each of which is characterized by one or two breakpoint locations in the genome. Both the SV prediction accuracy and our estimation of variant frequencies depend on having an accurate alignment in which reads supporting the SV breakpoint are properly soft-clipped at those breakpoints. Unfortunately, this is not always the case after the initial global alignment. If a sufficient global alignment is not found for an aligned read's pair, BWA will perform local alignment in the genomic region near where the read's pair is predicted to occur. This local alignment algorithm will soft-clip any ends that do not align. However, many reads that should be soft-clipped are missed for two reasons. First, reads that are almost fully aligned to the reference sequence (i.e. only a few bases extend past the breakpoint) can be aligned with mismatches in the global alignment portion of the BWA algorithm. In this situation, local alignment is not attempted because the global alignment is sufficient, and thus there is no soft-clipping done. An example of this can be seen in the bottom two aligned reads in Figure 4.3. Second, reads that should be soft-clipped at a breakpoint might remain unmapped completely. This occurs when BWA determines that neither the global alignment nor the local alignment are of sufficient quality. In both cases, the soft-clipped local alignment represents the "true" alignment, but the aligner has no way of knowing the location of the SV breakpoint, and thus will instead force a mismatched global alignment or leave the read unmapped.

These issues could potentially be addressed by modifying BWA's parameters to be more stringent for accepting global alignments and less stringent for soft-clipped local alignments. However, this can introduce additional issues when aligning other concordant reads. Assuming we have at least some reads from variant breakpoints that are aligned with the correct soft-clipping, we can use the predicted SVs to hone in on the exact areas that are likely to experience the above problems and correct them.

To address these problems, we remap these reads using a local algorithm, such as BWA-SW [12] or BWA-MEM [13]. These algorithms are less efficient than the global

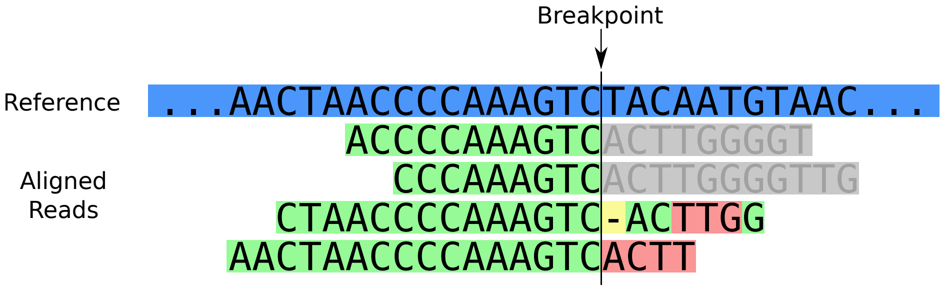


Figure 4.3: Soft-clipped reads vs. improperly aligned reads at a structural variant breakpoint. The reference sequence is indicated in blue. Green bases are alignment matches, pink bases are mismatches, yellow bases are skips, and gray bases are soft-clipped. All four reads are sampled from the variant sequence and span the breakpoint, but only the top two are properly soft-clipped. The bottom two reads are aligned with mismatches and skips because the portion that should be soft-clipped is only a few bases long and the global alignment is used instead. Inaccurate soft-clipping can lead to false negatives for split-read based SV prediction algorithms and also lowers the accuracy of SHEAR's SV variant frequency estimation algorithms.

alignment of the regular BWA algorithm and can only align reads individually, not in pairs, but it does allow for soft-clipping rather than requiring the entire read to be aligned. We account for this inefficiency by remapping only reads that might be affected by the two issues discussed above. Specifically, all unmapped reads, as well as reads that align either spanning or soft-clipped at one of these breakpoints, are extracted from the original alignment. This collection of reads are then locally realigned using default parameters to produce a new alignment. Leaving the concordantly aligned reads alone improves the efficiency of our approach in comparison with *de novo* assemblers or the iterative alignment of IMR. Aligning these reads individually removes the pairing restrictions that may lead to unmapped reads when using default BWA alignment, as discussed above. Local realignment around candidate small INDELs is a common processing step for INDEL prediction [94]. Our approach is effectively doing something analogous for SVs by locally realigning reads at SV breakpoints.

The corrected version of the alignment is then passed back to the SV prediction algorithm to make revised predictions. New SV predictions can be picked up if there were not a significant number of supporting soft-clipped reads in the original alignment. At this point, the unmapped and breakpoint reads can be extracted again from the original alignment to correct the soft-clipping at these new SV locations, using the same procedure. In practice, we observe that this rarely needs to be repeated more than three times before SV predictions remain unchanged. Finally, before estimating variant frequencies for each SV, SHEAR will remove some variants that are slight derivations from other predicted SVs (i.e. only differing by a few base pairs on one breakpoint). If the differing breakpoint has only a few supporting soft-clipped reads in comparison with a "main" predicted breakpoint with many more, it is usually due to sequencing error and thus can be discarded as a false positive.

Similar to the SV prediction module of the SHEAR pipeline, we created an abstract Java class (`Aligner`) that can be extended to easily incorporate new local alignment algorithms into the framework. A new `Aligner` subclass must implement the following method:

```
public abstract void align(String fastqInputFilePath,
                           String samOutputFilePath,
                           List<String> options)
   throws FileNotFoundException, IOException;
```

The input parameters for the `align` method include a string containing the file path to the FASTQ file containing all reads that should be realigned, a string containing the desired file path for the SAM output file, and a list of strings that can provide an arbitrary amount of additional arguments to pass to the external local alignment program. Additional information, such as the location of the reference genome being used, is available within the `Aligner` class.

In the latest version of SHEAR (i.e. v1.1.2), we have implemented the BWA-SW [12] and BWA-MEM [13] local alignment algorithms for available use. Newer alignment algorithms that support soft-clipping, such as GSNAP [16], may offer stronger capabilities for local alignment at SV breakpoints and will be incorporated in future versions of SHEAR.

4.4 Experimental Results

We compare our results with IMR/DENOM [92] on a variety of simulated data sets and one real data set to demonstrate the advantages of our approach, namely, improved computational efficiency, better support for tandem duplications, and the ability to handle personal genome assembly in the presence of heterogeneous sequencing samples. IMR/DENOM alone was chosen for comparison because it was the mos similar approach available. Other reference-guided assembly methods produce contigs or are generally designed for creating new reference sequences rather than “personalizing” existing reference sequences, which is the purpose of SHEAR.

4.4.1 Simulated data

A reference sequence to be used for simulation was taken from a 70 kbp region of chromosome 15 (25,420,001–25,490,000) chosen because of its non-repetitiveness. The length of this sequence is intended to be on the scale of the size of a long gene. Twenty different variant sequences are then created by introducing ten different sets of deletions and ten different sets of tandem duplications at known locations. These represent two SV types easily handled by CREST, which was used as the underlying SV prediction algorithm in this case. Each set of SVs contained three non-overlapping SVs of sizes 150 bp, 1000 bp, and 30 kbp.

Sequencing simulation was then performed on each of the variant sequences by randomly sampling paired-end reads, with read lengths of 75 bp and fragment sizes sampled from a truncated normal distribution with a mean of 250 bp, standard deviation of 20 bp, inclusive lower bound of 175 bp, and inclusive upper bound of 325 bp. For each sequencing simulation, a portion of the paired-end reads were sampled from the original sequence as well as from the variant sequence. This heterogeneity level was varied (20%, 40%, 60%, 80%, 90%, and 100% from the variant sequence), as was the overall average coverage (10×, 20×, 30×, 50×, 100×, 500×, and 1000×). There were no synthetic sequencing errors and base quality was reported as perfect (i.e. Phred quality scores of 40). This was to eliminate the effects of sequencing errors in order to evaluate the two methods solely on their algorithmic approach. These simulated data sets were intended to be easy to handle, in order to control for issues with fragment size distributions, sequencing errors, SNPs, small INDELs, and cross-chromosomal events. Instead, we focus specifically on the issue of how to account for heterogeneous SVs. Using a 20× overall coverage and varying the portion of simulated reads that originated from the variant sequence, our method demonstrates a strong ability to handle heterogeneous SVs (see

	Deletions		Tandem Duplications	
Variant Frequency	**SHEAR**	**IMR/DENOM**	**SHEAR**	**IMN/DENOM**
20%	6 / 30	0 / 30	3 / 30	0 / 30
40%	20 / 30	0 / 30	14 / 30	0 / 30
60%	26 / 30	0 / 30	24 / 30	0 / 30
80%	29 / 30	1 / 30	26 / 30	0 / 30
90%	29 / 30	1 / 30	24 / 30	0 / 30
100%	28 / 30	27 / 30	26 / 30	0 / 30

Table 4.1: Correctly detected SVs for simulated data at 20× coverage under varying levels of heterogeneity All simulations are done on a 70,000 bp portion of chromosome 15 after introducing deletions and tandem duplications of sizes 150 bp, 1000 bp, and 30,000 bp, each over 10 different iterations, for a total of 30 different deletion events, and 30 different tandem duplication events.

Table 4.1) IMR/DENOM is only able to reliably detect deletions in relatively homogeneous sequencing samples. Even at high overall coverage (i.e. 1000×), IMR/DENOM is still unable to pick out heterogeneous variants, suggesting that this is not due to a lack of supporting reads (see Table 4.2). Tandem duplications are never identified using IMR/DENOM with our simulated data.

Table 4.4 demonstrates our method's ability to scale down to lower coverage levels even in the heterogeneous case where only 20% of the reads are sampled from the variant sequence. IMR/DENOM fails to detect any of the SVs present in the sample, while our method scales down well to 50× overall coverage and even picks up a few events from the 30× and 20× coverage data sets. The depth of coverage required to detect SVs depends on the variant frequency of each SV. For example, as seen in Table 4.3, 20× coverage is too low to reliably pick up variants that only comprise 20% of the sequencing sample. However, the same coverage level was enough to detect most of the variants (50/60) simulated at 60% heterogeneity level (see Table 4.4).

Additionally, SHEAR's post-processing of SV predictions via local realignment to correct soft-clipping errors improves the SV predictions in comparison to using CREST alone. Of the 2520 SVs in the simulated data set (i.e. twenty different variant sequences × three SVs per variant sequence × seven different coverage settings × six different heterogeneity percentage settings), 2072 were predicted by both CREST and SHEAR, but SHEAR improved the accuracy of breakpoint prediction for 502 (24.22%) of these,

	Deletions		Tandem Duplications	
Variant Frequency	**SHEAR**	**IMR/DENOM**	**SHEAR**	**IMN/DENOM**
20%	29 / 30	0 / 30	28 / 30	0 / 30
40%	29 / 30	0 / 30	28 / 30	0 / 30
60%	29 / 30	0 / 30	28 / 30	0 / 30
80%	29 / 30	0 / 30	28 / 30	0 / 30
90%	29 / 30	0 / 30	28 / 30	0 / 30
100%	29 / 30	17 / 30	28 / 30	0 / 30

Table 4.2: Correctly detected SVs for simulated data at 1000× coverage under varying levels of heterogeneity All simulations are done on a 70,000 bp portion of chromosome 15 after introducing deletions and tandem duplications of sizes 150 bp, 1000 bp, and 30,000 bp, each over 10 different iterations, for a total of 30 different deletion events, and 30 different tandem duplication events.

	Deletions		Tandem Duplications	
Variant Frequency	**SHEAR**	**IMR/DENOM**	**SHEAR**	**IMN/DENOM**
10×	1 / 30	0 / 30	0 / 30	0 / 30
20×	6 / 30	0 / 30	3 / 30	0 / 30
30×	13 / 30	0 / 30	13 / 30	0 / 30
50×	21 / 30	0 / 30	21 / 30	0 / 30
100×	29 / 30	0 / 30	28 / 30	0 / 30
500×	29 / 30	0 / 30	28 / 30	0 / 30
1000×	29 / 30	0 / 30	28 / 30	0 / 30

Table 4.3: Correctly detected SVs for simulated data at 20% heterogeneity under varying levels of coverage All simulations are done on a 70,000 bp portion of chromosome 15 after introducing deletions and tandem duplications of sizes 150 bp, 1000 bp, and 30,000 bp, each over 10 different iterations, for a total of 30 different deletion events, and 30 different tandem duplication events.

	Deletions		Tandem Duplications	
Variant Frequency	**SHEAR**	**IMR/DENOM**	**SHEAR**	**IMN/DENOM**
10×	7 / 30	0 / 30	7 / 30	0 / 30
20×	26 / 30	0 / 30	24 / 30	0 / 30
30×	29 / 30	0 / 30	26 / 30	0 / 30
50×	29 / 30	0 / 30	27 / 30	0 / 30
100×	29 / 30	0 / 30	28 / 30	0 / 30
500×	29 / 30	0 / 30	28 / 30	0 / 30
1000×	29 / 30	0 / 30	28 / 30	0 / 30

Table 4.4: Correctly detected SVs for simulated data at 60% heterogeneity under varying levels of coverage All simulations are done on a 70,000 bp portion of chromosome 15 after introducing deletions and tandem duplications of sizes 150 bp, 1000 bp, and 30,000 bp, each over 10 different iterations, for a total of 30 different deletion events, and 30 different tandem duplication events.

whereas CREST only improved the breakpoint accuracy for three SVs (see Table 4.5). Additionally, SHEAR's local realignment component tends to increase the number of supporting soft-clipped reads for each predicted SV, with a 7.26% relative increase of supporting reads for each SV prediction.

SHEAR and IMR/DENOM were also evaluated on a deeply-sequenced (i.e. 6000×) tumor cell line for the *AR* gene locus, which will be discussed further in Section 5.4.2. Six heterogeneous deletions were detected in this data set by SHEAR, with four being experimentally validated, while none were assembled by IMR/DENOM due to the heterogeneity of the sequencing sample. This data set also demonstrates how the assembly component of SHEAR scales well for deeply-sequenced data in comparison with IMR/DENOM. The entire SHEAR pipeline (including initial alignment using BWA, SHEAR-SV, and SHEAR-Assemble) completed execution in just under 16 hours using a cluster of eight 2.66 GHz processors. The two components of IMR/DENOM could be run in parallel, with IMR, the more computationally expensive component due to the iterative alignments, taking more than three days on 24 2.66 GHz processors. These results indicate that SHEAR offers an efficiency advantage over IMR even though both operate iteratively, because SHEAR excludes concordantly aligned reads from future iterations. For example, the first execution of CREST in the SHEAR pipeline took seven hours, whereas the two subsequent executions of CREST took less than an hour combined.

SVs predicted by CREST:	2073 / 2520
SVs predicted by SHEAR:	2073 / 2520
SVs predicted by both:	2072 / 2520
More accurate breakpoints with CREST:	3 / 2072
More accurate breakpoints with SHEAR:	502 / 2072
Same breakpoints:	1567 / 2072

Table 4.5: Summary of performance for SHEAR versus standalone CREST on simulated data sets. All simulations are done on a 70,000 bp portion of chromosome 15 after introducing deletions and tandem duplications of sizes 150 bp, 1000 bp, and 30,000 bp, each over 10 different iterations, for a total of 30 different deletion events, and 30 different tandem duplication events. Sequencing data was simulated for each of these synthetic sequences, with heterogeneity percentage varying over six settings (i.e. 20%, 40%, 60%, 80%, 90%, and 100% of reads from the variant sequence) and overall average coverage varying over seven settings (10×, 20×, 30×, 50×, 100×, 500×, 1000×) for a total of 2520 SVs present in the simulated data sets.

4.5 Discussion and Future Work

Although we have shown SHEAR-Assemble to be a more appropriate assembly pipeline in the context of tumor sequencing data due to its computational efficiency and ability to account for heterogeneous variants, its practicality is still somewhat limited. Currently, SHEAR-Assemble requires a manual step of selecting desired variants to include in the assembly, which makes it difficult to apply when trying to assemble multiple genomes from a heterogeneous sequencing sample. Grouping variants together into their respective cellular subpopulations will allow for automatic assembly of multiple personal genomes. This is a nontrivial task and often requires deep sequencing in order to obtain reasonably accurate phasing information (i.e. information that can be used to group together co-occurring variants). Specific ideas for future work that address this need will be discussed later in Section 5.5.

The methods and results discussed in this chapter were limited to small, intra-chromosomal SV types for proof of concept in leveraging SV predictions to generate personal genomic sequence. Future versions of SHEAR will allow for assembly of more complex SV types. For example, SVs fusions often contain additional or altered bases at the breakpoint fusion, called microinsertions and microinversions, which are not addressed by the current SHEAR framework. Microinsertions or microinversions at the fused junction of a deletion have been found to be the most common type of complex SV [104, 105]. More complex SVs may include overlapping patterns (e.g. a large inversion within a duplication) or even more extreme rearrangements of entire chromosomes (i.e. chromoplexy or chromothripsis). These complex SVs create additional challenges in phasing variants to assemble multiple genome outputs, as well as in dealing with resolving conflicts in the assembly process itself. One possible solution to explore would be to use graph theory algorithms to determine the best way to represent such chromosomes in the assembly (i.e. associating co-occurring breakpoints via graph edges).

An important advancement in assembly over the last several years has been the development of single molecule, real-time (SMRT) sequencing. This new generation of sequencing technologies has focused on offering much longer read lengths (i.e. tens of thousands of base pairs) compared with Illumina or other second-generation sequencing technologies. Although error rates are much lower, an important aspect is that sequencing errors tend to be uniformly distributed throughout reads. This unbiased error rate, combined with the long read lengths to resolve repetitive regions, mean that complete *de novo* assembly is theoretically possible in many cases given enough sequencing depth. See Chaisson et al. [106] for a review and discussion comparing SMRT-based *de novo* assembly with other technologies.

SMRT sequencing and its associated technologies and algorithms offer the potential for reference-quality *de novo* assembly. Assembly algorithms specifically designed for SMRT sequencing, such as FALCON [107], have been developed in recent years, and the technology has started to be used to do complete assembly of large genomes in many organisms, including in humans [108, 109]. It appears likely that SMRT sequencing will emerge as an ideal solution to personal genome assembly as well in the future, though improvements in read length and throughput are still necessary before complete human *de novo* assembly becomes routine [106]. Until such improvements to SMRT sequencing are realized, techniques such as SHEAR-Assemble can offer an effective alternative approach that emphasizes efficiency and ease-of-use by leveraging more readily-available sequencing technologies such as Illumina data.

Chapter 5

Estimating Tumor Heterogeneity

5.1 Introduction

Tumors develop through genetic mutations that lead to abnormal cell growth. Tumors also develop through a "survival of the fittest" evolutionary process, with different tumor cells competing with each other for space, oxygen, and other molecular nutrients. Because tumors grow at such an elevated rate, cells divide more frequently, leading to an accumulation of new mutations in different cells. Mutations that give a cell a selective advantage over other tumor cells in the environment can cause that cell to proliferate into a dominant population, known as a clonal expansion. This usually results in one subclone (i.e. a subpopulation of cells with a similar genetic makeup) that dominates the tumor and drives the tumor's growth. However, the ongoing evolutionary battle between the different cellular subpopulations means that there is always a heterogeneous mix of subclones, each with their own genomes, within any given tumor. Tumors in which the internal microenvironments differ significantly across the tumor can also lead to a more evenly distributed heterogeneous mix of subclones, in which different subclones can dominate different parts of the tumor. For example, one portion of the tumor tissue may have more direct access to blood flow, and the subclone that dominates there might be different from other areas of the tumor where the dominant subclone has a selective advantage for dealing with more limited access to blood.

Tumor heterogeneity is an important problem because it can often lead to poor clinical outcome when treating cancer patients. Tumors with a high level of cellular heterogeneity can make it difficult to classify the tumor's phenotype, and thus difficult to prescribe the most effective treatment option. Even for tumors that are relatively homogenous (i.e. one cell subclone that has a very strong dominance in the tumor), effective treatment may successfully target the primary tumor clone, but a small subclone may rise up to take its place if it contains mutations that lead to resistance against the treatment. For example, a recent study analyzing a prostate cancer cell line found that a deletion in the androgen receptor (AR) gene was present in a very small portion of the cell line at first, but grew to be present in the majority of the cells after being cultured in an androgen-deficient environment [110]. This suggests that the deletion is a marker for resistance to androgen depletion therapy (ADT), a common treatment for castration-resistant prostate cancer (CRPCa). This demonstrates the importance of studying all variants, including those present in small portions of a heterogeneous tumor sample. Tumor heterogeneity can also cause clinical problems if metastases are genetically different from the dominant clone in the primary tumor, which again would require a shift in treatment plan.

All of this leads to a currently unmet need for effective methods to characterize heterogeneity in a given tumor. This will play a vital role in our ability to understand cancer, and subsequently our ability to effectively treat it.

Various techniques have been developed that use SNP array data or next-generation sequencing data to estimate tumor purity or tumor ploidy for a sample. These techniques attempt to quantify the proportion of the sample that contains tumor cells (i.e. tumor purity) as well as the average copy number of the DNA content in those cells (i.e. tumor ploidy). Newer tumor purity/ploidy estimation algorithms have also begun to focus on predicting multiple tumor subclones. The output from these algorithms can provide a good "big-picture" view of the tumor heterogeneity in a sample, but lack detailed information, particularity due to ignoring information about copy-neutral events.

Another way to address this problem is to predict and examine variants that are present in only a portion of the sequencing sample. The frequency of each variant in the sequencing sample can then be estimated using information gathered from the alignment. For SNPs and small INDELs, the variant frequency can be estimated using the counts of reads aligned at the locus that contain the variant versus those that do not. This is not immediately possible for SVs because these types of variants often fuse together different parts of the genome. Thus, a direct locus-specific comparison involving the reads that contain the variant versus those that do not is not possible. Some tumor purity/ploidy estimation algorithms effectively produce frequency estimations for copy number variants (CNVs) by estimating the number of copies of a segment in the sequencing sample, but in general a variant frequency estimation scheme that accounts for all types of SVs has not been developed.

The second main functionality that we have developed for SHEAR (previously introduced in Chapter 4) is SHEAR-SV, an algorithm to estimate the frequencies for all types of SVs using information about reads that are soft-clipped at, or that span across, SV breakpoints and fusion junctions. In this chapter, we will introduce our scheme for SV frequency estimation, and present results demonstrating the efficacy of our approach on simulated and real sequencing data from heterogeneous tumor samples. Section 5.2 contains a discussion of related work on the subject of tumor heterogeneity estimation. In Section 5.3, we present the SHEAR-SV model for SV frequency estimation using soft-clipped and breakpoint-spanning read information. Our experimental results from both simulated and real tumor data sets are presented in Section 5.4. Finally, we conclude with a discussion of issues and ideas for future work in Section 5.5.

5.2 Related Work

Estimating tumor heterogeneity is a challenging and complex analysis, and techniques have been developed that address the problem in a number of diverse ways. The ultimate goal with this type of analysis would be to determine the number of different subclones or subpopulations present in a sequencing sample (including potentially normal tissue cells), the variants present in each subpopulation, and the proportion of the sample that each subpopulation composes. No technique has been established that offers a complete solution to tumor heterogeneity estimation, but many existing approaches address different components of the problem.

One component of the problem is estimating the proportion of the sample that a particular variant is present in. For SNPs and small INDELs, estimating the variant frequency is a straightforward calculation that compares the number of reads aligned at the variant site that contain the SNP/INDEL with the number of reads that do not. For

Algorithm	Data Used	Information Used	Multiple Tumor Genomes
ABSOLUTE [112]	SNP Array	Copy Number (SNP Locus) Karyotype Likelihoods	-
ASCAT [113]	SNP Array	Copy Number (SNP Locus) SNP Allele Frequency	-
CNAnorm [114]	NGS	Copy Number (Intervals)	-
AbsCN-seq [115]	NGS	Copy Number (Intervals)	-
THetA [116, 117]	NGS	Copy Number (Intervals)	Yes
Clomial [118]	NGS	SNP Allele Frequency	Yes
EXPANDS [119]	NGS or SNP Array	Copy Number (SNP Locus) SNP Allele Frequency	Yes

Table 5.1: A comparison of several tumor heterogeneity estimation algorithms.

example, the *HaplotypeCaller* program in GATK [100] produces a VCF file [111] with entries for each SNP or INDEL that is detected. For each SNP/INDEL, the VCF entry also contains an AD tag, which is defined as "allelic depths for the ref and alt alleles in the order listed." This is essentially counts of the number of reads containing the variant (or multiple variants if the site has multiple alternative alleles) and the number of reads containing the reference bases. This information can be used to directly compute the variant frequency for each predicted SNP and INDEL. This is a variant-by-variant approach of estimating tumor heterogeneity, but is only applicable for SNPs and INDELs.

Alternatively, there have been a number of techniques that attempt to analyze tumor heterogeneity from a "big picture" perspective. These algorithms estimate tumor ploidy and tumor purity from sequencing data or SNP array data derived from a tumor sample (see Table 5.1 for a summary). Tumor ploidy refers to the average total copy number of the DNA content in the tumor cells. A ploidy of two would be equivalent to the normal genome, whereas a smaller or greater value would correspond with a net loss or gain of DNA content, respectively. Tumor purity refers to the percentage of cells in the sequencing sample that are tumor cells (as opposed to normal cells). By attempting to distinguish between normal and tumor genomes in a sequencing sample (albeit with assuming only one tumor subclone), tumor ploidy/purity estimation algorithms represent an initial effort to solve a portion of the larger tumor heterogeneity problem.

ABSOLUTE [112] and ASCAT [113] are two of the earliest tumor ploidy/purity estimation algorithms. Both of these work on SNP array data, which which contains an allele frequency and copy number information for each SNP. A statistical model is used to jointly estimate ploidy and purity using that information. ABSOLUTE also contains an additional step where ambiguous solutions are resolved by using a database of common cancer karyotypes to find the more likely solution. CNAnorm [114] works on next-generation sequencing data instead of the older SNP array technology, but estimates ploidy first followed by estimating purity, rather than estimating both jointly. AbsCN-seq [115] is another approach that uses next-generation sequencing, but infers ploidy and purity jointly, along with absolute copy numbers. These approaches all assume there is only one tumor subclone.

More recently an algorithm called THetA was developed that allows for multiple tumor genomes in its inference [117, 116]. The genome is first partitioned into different segments by locating segments with constant coverage and separating neighboring segments with coverage differences. This is done using a copy number prediction or DNA segmentation algorithm, such as BIC-seq [120]. This data can be thought of as a read vector in which each entry represents the number of reads that uniquely align to a particular segment. THetA then models this data as a multinomial distribution in order to estimate the probability of sampling from each segment using maximum likelihood. This probability is the product of two factors: (1) a C matrix that represents the actual integer copy number for each segment in each genome, and (2) a μ vector that represents the proportion of each genome in the sequencing sample. The product of these two values, $C\mu$, is a vector that is the probability of sampling from each segment during sequencing. This can also be thought of as the overall proportion that each segment has in the overall mass of DNA being sequenced. Thus, the maximum likelihood estimation of the $C\mu$ parameter can suggest a variety of C and μ combinations. THetA offers an important improvement over ABSOLUTE or ASCAT in that it can predict multiple tumor genomes, but is still limited to only analyzing copy number variants (CNVs) in pre-determined segments, and does not deal with SNPs or copy-neutral structural variants (i.e. inversions or translocations).

Clomial is another tumor heterogeneity estimation algorithm that can infer multiple tumor subclones [118]. Unlike THetA, which uses CNV segments from a tumor and a normal cell sequencing sample as input, Clomial makes its inferences using SNP and small

INDEL information from a normal cell sequencing sample as well as sequencing samples from multiple sections of a tumor (possibly both primary and metastatic). Specifically, loci that are homozygous in the normal sample but exhibit an alternate SNP or INDEL allele in at least one of the tumor samples (in at least 15 reads) are used as input. The number of reads containing the alternate allele at a given locus i in a given tumor section sample j is modeled as a binomial distribution with parameters $R_{i,j}$ (i.e. the total number of aligned reads at locus i in sample j) and $\pi_{i,j}$ (i.e. the probability of observing an alternate allele read for locus i in sample j). The parameter $\pi_{i,j}$ is a function of the clone frequencies (i.e. the distribution of clones in each sample) and the clone genotypes (i.e. the set of alternate allele mutations found in each clone). An optimal solution is found using a customized Expectation-Maximization (EM) algorithm [121] in which the total and alternate allele read counts are observed variables, the clone genotypes are latent variables, and the clone frequencies are the model parameters. The number of clones is not inferred but rather provided as a hyperparameter. This approach allows for accurate tumor heterogeneity estimation using input information at only a small number of loci, but suffers the drawbacks of requiring sequencing samples from multiple tumor sections and being confined to small heterozygous somatic variants while ignoring information about SVs.

Another tumor heterogeneity estimation algorithm with support for predicting multiple tumor subclones is EXPANDS [119]. EXPANDS uses a clustering-based methodology to infer the tumor subclones and their genotypes. The input to EXPANDS is a set of SNPs, each with an allele frequency and local copy number estimate. A probability density is generated for each SNP to describe the likelihood of different cell frequencies (i.e. the percentage of cells containing the variant). For example, a SNP with a 30% allele frequency and an estimated copy number of two would have probability density peaks at around 30% and 60% cell frequencies, corresponding to the possible explanations of a homozygous mutation in 30% of the cells or a heterozygous mutation in 60% of the cells, respectively. The SNPs are then clustered based on their probability density functions to find similar peaks across different SNPs. After some filtering, the remaining clusters represent the predicted subclones, and the SNPs assigned to each one determine the subclone's genotype.

All of these tumor heterogeneity estimation algorithms either focus only on frequency variants for SNPs and small INDELS, or else focus more on estimating the tumor ploidy of subclones instead of their representative variants. Importantly, none of these techniques account for copy-neutral SVs, such as inversions and translocations. These types of variants are important to study as well, since they can result in changes to gene expression by moving genes next to different regulatory elements, or even create fusion genes when breakpoints occur in the middle of two different genes.

5.3 Proposed Approach

SHEAR can help address the tumor heterogeneity problem by assigning estimated frequencies to predicted variants, thus determining heterogeneity on the level of individual mutations. One of the novel components of SHEAR is an algorithm (SHEAR-SV) for estimating these frequencies for SVs (including copy-neutral events such as inversions or translocations) using soft-clipping information.

After the iterative local realignment pipeline is completed, the frequencies for each variant can be estimated by comparing the average number of reads per locus from the reference-like sequence (i.e. reference depth R) with the number of reads per locus from the variant sequence (i.e. variant depth V). The estimated frequency of a variant (or its heterogeneity level) is simply equal to:

$$H = \frac{\hat{V}}{\hat{R} + \hat{V}} \tag{5.1}$$

where $\hat{R}$ and $\hat{V}$ are the maximum likelihood estimates for R and V. In the case of SNPs and small INDELs, this is a trivial estimation, as $\hat{R}$ is merely the number of reads aligned over the locus for the SNP/INDEL that contain the reference bases, and $\hat{V}$ is the number of reads aligned over the locus that contain the SNP/INDEL.

As mentioned previously, this approach is not directly applicable to estimating variant frequencies for SVs, and some heuristics must be used to obtain estimates for $\hat{R}$ and $\hat{V}$ in the case of SVs. These can be estimated from the numbers of reads that span the SV breakpoints and the number of reads that are soft-clipped at these breakpoints. The model to calculate $\hat{R}$ and $\hat{V}$ depends on the type of SV. Below we present our variant frequency estimation schemes for all types of SVs (i.e. deletions, tandem duplications, inversions, translocations, and insertions). In all cases, the goal is to determine the estimated reference depth ($\hat{R}$) and the estimated variant depth ($\hat{V}$), and to estimate the heterogeneity percentage by comparing the two as in Equation (5.1). Each case will be accompanied by a toy example in which the variant subpopulation composes 20% of the sample (i.e. the true value of H) and the alignment behavior is depicted in a corresponding figure. Let the total number of aligned reads that span both SV breakpoints (or the sole SV breakpoint in the case of an insertion) be A, and the total number of relevant soft-clipped reads in each situation be B.

In the case of a deletion, the original reference has two breakpoints that are merged into the same locus in the variant subpopulation (see Figure 5.1). Reads sampled from the variant subpopulation that span this breakpoint will align, with soft-clipping, to either of the corresponding breakpoints in the reference depending on which half of the read the breakpoint is at. Thus, the relevant soft-clipped reads are all of those that are clipped on their 3' ends at the left breakpoint, and all of those that are clipped on their 5' ends at the right breakpoint, which together are are used as the estimate for the variant depth (i.e. $\hat{V} = B$). The estimated reference depth $\hat{R}$ is taken from the average number of spanning reads between the two breakpoints, (i.e. $\frac{1}{2}A$). The heterogeneity percentage, H, is estimated as:

$$H = \frac{\hat{V}}{\hat{R} + \hat{V}} = \frac{B}{(\frac{1}{2}A) + (B)} = \frac{2B}{A + 2B} \tag{5.2}$$

In the example in Figure 5.1, $A = 16$ and $B = 2$, giving an estimated variant depth of 2× and an estimated reference depth of $\frac{1}{2}16 = 8\times$, for a 20% estimated variant frequency.

The case of a tandem duplication is slightly more complicated (see Figure 5.2). Reads from the variant subpopulation that span the new fusion (i.e. the middle of the tandem repeat) will again map to either of the two reference breakpoints depending on which side of the breakpoint each is more aligned with. In this case, the relevant soft-clipped reads (i.e. B) are composed of all the reads that are clipped on their 5' end at the left breakpoint or that are clipped on their 3' end at the right breakpoint. However, reads that align correctly to the outside edges of the duplicated segment could have been sampled from either the reference-like sequence or the variant sequence. In order to get an accurate estimate of the number of reads from the reference-like sequence, we must

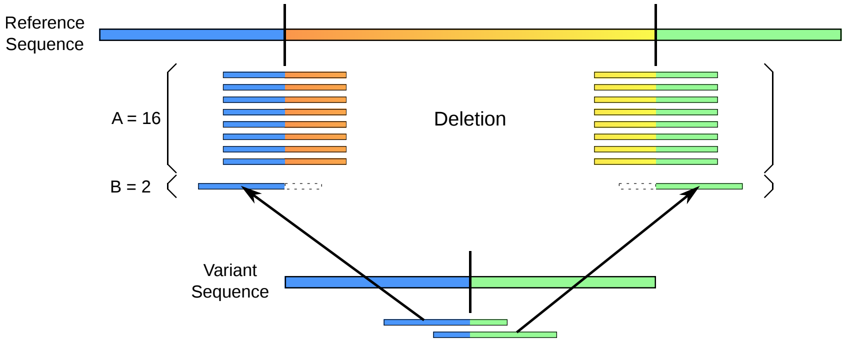


Figure 5.1: Alignment of reads from a heterogeneous deletion. There is a 10× overall depth of coverage, with 20% of the sample coming from the variant sequence containing the deletion (i.e. 2× depth) on the bottom and 80% from the reference-like sequence (i.e. 8× depth) on top. Arrows indicate how reads from the variant sequence will be aligned against the reference genome, with soft-clipping indicated by dotted line borders. A indicates the total number of reads that span either breakpoint, and B indicates the total number of relevant soft-clipped-reads. A and B are used by SHEAR to estimate the variant frequency for a deletion.

account for the fact that A contains reads from both populations. Again we take the total number of relevant soft-clipped reads, B, as the estimated variant depth $\hat{V}$. This is then subtracted from the average number of spanning reads to arrive at an estimated reference depth. The heterogeneity percentage is thus estimated as:

$$H = \frac{\hat{V}}{\hat{R} + \hat{V}} = \frac{B}{(\frac{1}{2}A - B) + (B)} = \frac{2B}{A} \tag{5.3}$$

In the example in Figure 5.2, $A = 20$ and $B = 2$, giving an estimated variant depth of 2× and an estimated reference depth of $\frac{1}{2}20 - 2 = 8\times$, for a 20% estimated variant frequency.

Unlike deletions and tandem duplications which have a difference in the number of breakpoints involved between the reference-like sequence and the variant sequence, inversions have two breakpoints in each sequence (see Figure 5.3). Reads sampled from the variant subpopulation that span these breakpoints will again map to either of the reference breakpoints, depending on their location. Reads that are soft-clipped in either direction at either breakpoint are all relevant in this case. Because there is no copy number change in an inversion, we can directly estimate the heterogeneity by comparing

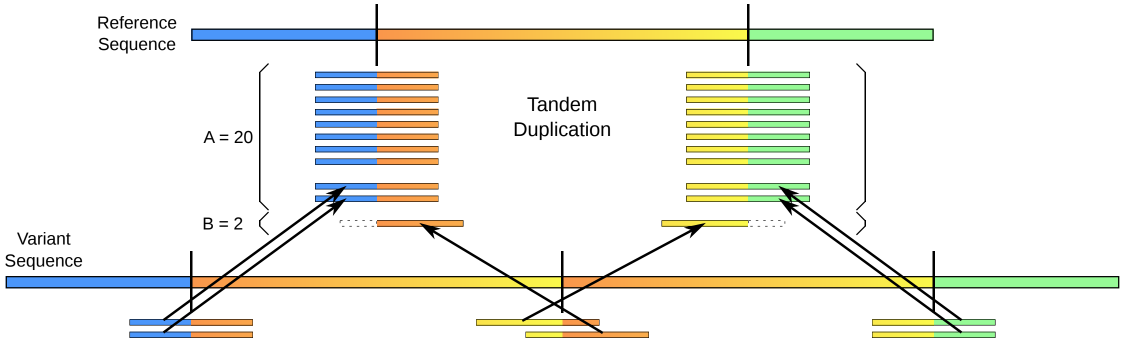


Figure 5.2: Alignment of reads from a heterogeneous tandem duplication. There is a 10× overall depth of coverage, with 20% of the sample coming from the variant sequence containing the tandem duplication (i.e. 2× depth) on the bottom and 80% from the reference-like sequence (i.e. 8× depth) on top. Arrows indicate how reads from the variant sequence will be aligned against the reference genome, with soft-clipping indicated by dotted line borders. A indicates the total number of reads that span either breakpoint, and B indicates the total number of relevant soft-clipped-reads. A and B are used by SHEAR to estimate the variant frequency for a tandem duplication.

the total number of soft-clipped reads with the total number of spanning reads at the two breakpoints of the inversion. Heterogeneity is thus estimated as:

$$H = \frac{\hat{V}}{\hat{R} + \hat{V}} = \frac{\frac{1}{2}B}{(\frac{1}{2}A) + (\frac{1}{2}B)} = \frac{B}{A + B} \tag{5.4}$$

In the example in Figure 5.3, $A = 16$ and $B = 4$, and a direct estimation of variant frequency is again 20%.

In the case of the translocation depicted in Figure 5.4, in which the 3' end of the forward strand on one chromosome is joined to the 5' end of the forward strand on another chromosome, the relevant soft-clipped reads (i.e. B) are those that are clipped on the 3' end of the first breakpoint or clipped on the 5' end of the second breakpoint. Note that this may change depending on the translocation type. For example, in the case of the 3' end of the forward strand on one chromosome being joined to the 5' end of the reverse strand on another chromosome, the relevant soft-clipped reads at the second breakpoint would instead be those that are clipped on their 3' end as well (relative to the forward strand). In all translocation cases though, regardless of how B is determined, the variant frequency model is the same as that of a deletion:

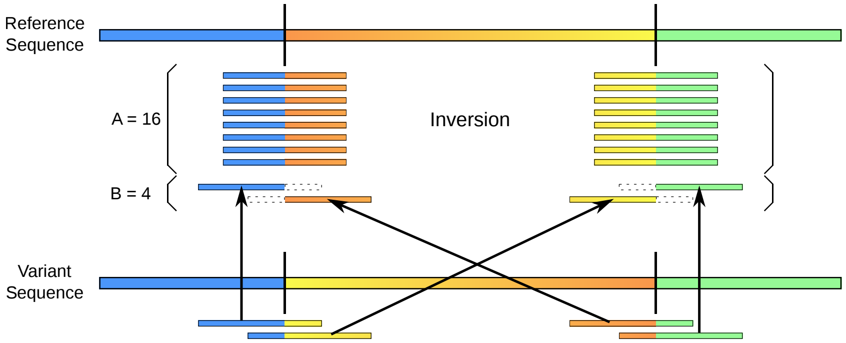


Figure 5.3: Alignment of reads from a heterogeneous inversion. There is a 10× overall depth of coverage, with 20% of the sample coming from the variant sequence containing the inversion (i.e. 2× depth) on the bottom and 80% from the reference-like sequence (i.e. 8× depth) on top. Arrows indicate how reads from the variant sequence will be aligned against the reference genome, with soft-clipping indicated by dotted line borders. A indicates the total number of reads that span either breakpoint, and B indicates the total number of relevant soft-clipped-reads. A and B are used by SHEAR to estimate the variant frequency for a inversion.

$$H = \frac{\hat{V}}{\hat{R} + \hat{V}} = \frac{B}{(\frac{1}{2}A) + (B)} = \frac{2B}{A + 2B} \tag{5.5}$$

In the example in Figure 5.4, $A = 16$ and $B = 2$, giving an estimated variant depth of 2× and an estimated reference depth of $\frac{1}{2}16 = 8\times$, for a 20% estimated variant frequency.

Finally, we have the case of an insertion (see Figure 5.5). There is only one breakpoint in this case, which contains portions of clipped reads from two different loci on the variant sequence. In this case, the total number of reads that span the lone breakpoint is used as the reference-depth estimation $\hat{R}$ rather than taking the average of all spanning reads at two breakpoints. The relevant soft-clipped reads are all taken from the same lone breakpoint, and can be clipped in either direction. Heterogeneity is thus estimated as:

$$H = \frac{\hat{V}}{\hat{R} + \hat{V}} = \frac{\frac{1}{2}B}{A + (\frac{1}{2}B)} = \frac{2B}{A + 2B} \tag{5.6}$$

In the example in Figure 5.5, $A = 8$ and $B = 4$, giving an estimated variant depth of $\frac{1}{2}4 = 2\times$ and an estimated reference depth of 8×, for a 20% estimated variant frequency.

Note in that the above discussion of the models for H we are assuming intra-cellular homozygosity (i.e. all copies of the genetic locus within the cell contain the variant),

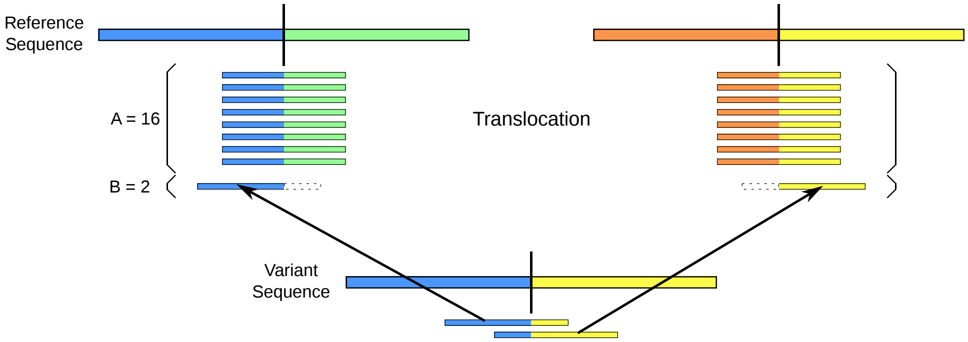


Figure 5.4: Alignment of reads from a heterogeneous translocation. There is a 10× overall depth of coverage, with 20% of the sample coming from the variant sequence containing the translocation (i.e. 2× depth) on the bottom and 80% from the reference-like sequence (i.e. 8× depth) on top, which includes two different chromosomes. Arrows indicate how reads from the variant sequence will be aligned against the reference genome, with soft-clipping indicated by dotted line borders. A indicates the total number of reads that span either breakpoint, and B indicates the total number of relevant soft-clipped-reads. A and B are used by SHEAR to estimate the variant frequency for a translocation.

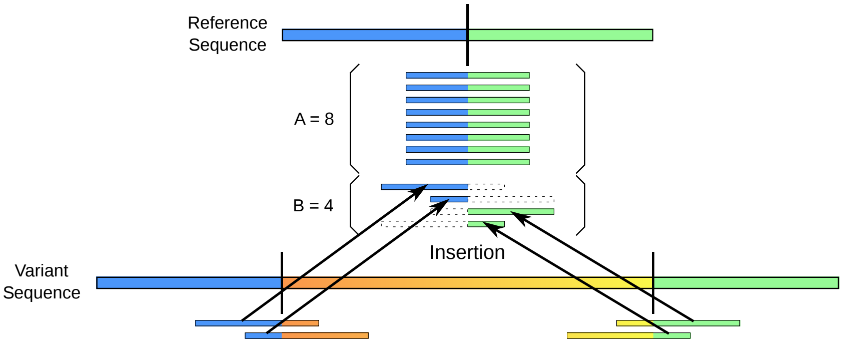


Figure 5.5: Alignment of reads from a heterogeneous insertion. There is a 10× overall depth of coverage, with 20% of the sample coming from the variant sequence containing the insertion (i.e. 2× depth) on the bottom and 80% from the reference-like sequence (i.e. 8× depth) on top. Arrows indicate how reads from the variant sequence will be aligned against the reference genome, with soft-clipping indicated by dotted line borders. A indicates the total number of reads that span the breakpoint, and B indicates the total number of relevant soft-clipped-reads. A and B are used by SHEAR to estimate the variant frequency for a insertion.

but the likelihood of heterozygosity in real-world data sets should be considered when interpreting the meaning of H. Thus, a reported variant frequency of $H = 50\%$ could imply either that 100% of the cells in the sequencing sample are heterozygous for the variant in a diploid case, or that 50% of the cells in the sequencing sample are homozygous for the variant.

5.4 Experimental Results

5.4.1 Simulated data

We estimated SV frequencies for all SVs that were found in the simulated heterogeneous tumor data sets analyzed previously in Section 4.4.1. Table 5.2 demonstrates the consistent accuracy of our variant frequency estimations on SVs that are discovered. As expected, there is more error in estimating the heterogeneity levels of tandem duplications than there is for deletions due to the heuristics used to estimate the number of reads from the reference-like sequence (see Figure 5.2). The average error of variant frequency estimation is higher at lower coverage levels, due to the smaller sample size of reads, and this effect is amplified for the more difficult problem of estimating the variant frequencies of tandem duplications. SHEAR's local realignment component also helps to improve the accuracy of SV frequency estimation by fixing incorrectly soft-clipped reads. In comparison with estimating variant frequencies from the original alignment, performance is improved by an average of 11.26 percentage points across the 2520 SVs in the simulated data set after SHEAR's targeted local realignment.

5.4.2 Tumor cell line data

To evaluate our SV frequency estimation methodology on an experimental real-world data set, we used next-generation sequencing data from previous work that examined the role that variants in the androgen receptor (AR) gene have on castration-resistant prostate cancer (CRPCa) [110]. Paired-end reads were sampled at 6000× coverage from non-repetitive regions of the AR locus in DNA from the CWR-R1 cell line model of CRPCa. Reads were 76 bp in length with a 208 bp median fragment size (62.44 bp standard deviation).

	Deletions			Tandem Duplications		
Depth	**150 bp**	**1000 bp**	**30 kbp**	**150 bp**	**1000 bp**	**30 kbp**
10×	0.00	1.22	1.46	13.98	13.73	12.27
20×	0.08	0.52	1.14	8.91	11.22	12.40
30×	0.17	0.42	1.06	9.64	11.48	11.86
50×	0.10	0.27	1.17	9.83	9.69	9.20
100×	0.08	0.21	1.21	5.56	5.47	6.06
500×	0.09	0.21	1.04	4.17	2.35	4.09
1000×	0.08	0.18	1.06	4.58	2.11	3.40

Table 5.2: Average error of SV frequency estimation. Each entry reports the average absolute error for estimation of variant frequencies for a variety of SVs at different overall coverage levels. For each pairing of SV type and coverage level, ten iterations of simulation were sampled from each of seven different underlying heterogeneity percentages (20%, 40%, 60%, 80%, 90%, and 100%) for a total of 70 simulations per entry in the table. The reported error is the absolute difference between SHEAR's estimation of variant frequency and the true percentage of breakpoint reads originating from the variant sequence. Each entry in the table contains the average error for that scenario, ignoring simulations in which the SV was not predicted. For example, for the first entry in the table (150 bp deletion at 10× depth), the SV is only predicted in 24 out of the 70 simulations due to the low coverage, and thus the average error is from those 24 estimations.

No.	SV Type	Breakpoints		Variant Frequency	Validated
1	Translocation	chr15:49,498,516	chrX:66,829,481	1.42%	-
2	Deletion	chrX:66,812,839	chrX:66,861,669	29.21%	Yes
3	Deletion	chrX:66,813,091	chrX:66,861,564	1.73%	Yes
4	Deletion	chrX:66,830,140	chrX:66,861,904	1.68%	Yes
5	Deletion	chrX:66,874,605	chrX:66,896,916	0.76%	Yes
6	Deletion	chrX:66,941,805	chrX:66,942,669	0.36%	-

Table 5.3: Left and right breakpoint locations on the reference sequence are given for each predicted structural variation, as well as the estimated levels of heterogeneity.

Table 5.3 lists the results found from this data set. SV #1 is a translocation between the *GALK2* gene locus on chromosome 15 and intron 1 of *AR*. SVs #2–4 are deletions within intron 1 of *AR* while SV #5 is located in intron 2. SV #6 is a deletion of the exact locus of intron 6 in *AR* and thus is likely the result of cDNA copies of mRNA present in the sequencing sample, as this sample was prepared in a lab that frequently works with *AR* expression vectors. Thus, the supporting reads for this SV call might have come from cDNA containing exons 6 and 7 spliced together.

SVs #2, #3, #4, and #5 were experimentally validated in this cell line sample using nested polymerase chain reaction (PCR) with deletion-spanning primers to amplify candidate SV regions, and Sanger sequencing to verify the joined sequences. We validated SV #3 in a previous study [110], and the PCR gel enrichments and electropherogram peak traces for SVs #2, #4, and #5 clearly confirm their presence in the sequencing sample (see Figure 5.6). Additionally, SHEAR removed eight CREST predictions that were slight derivations of these four reported SVs and that SHEAR determined to be false positives due to sequencing error. The experimental validation of SHEAR's reported SV breakpoints confirms that the removed CREST predictions were indeed false positives. We were unable to validate SV #1 via nested PCR because both of its breakpoints are located in repeat regions of the genome, making validation more difficult. This result could be spurious, however, the PCR validation of SVs #3, #4, and #5 suggests that SHEAR has the capability to identify true variants present in a very small percentage of the sample.

SVs #1 and #6 were not predicted by running CREST alone, outside of the SHEAR

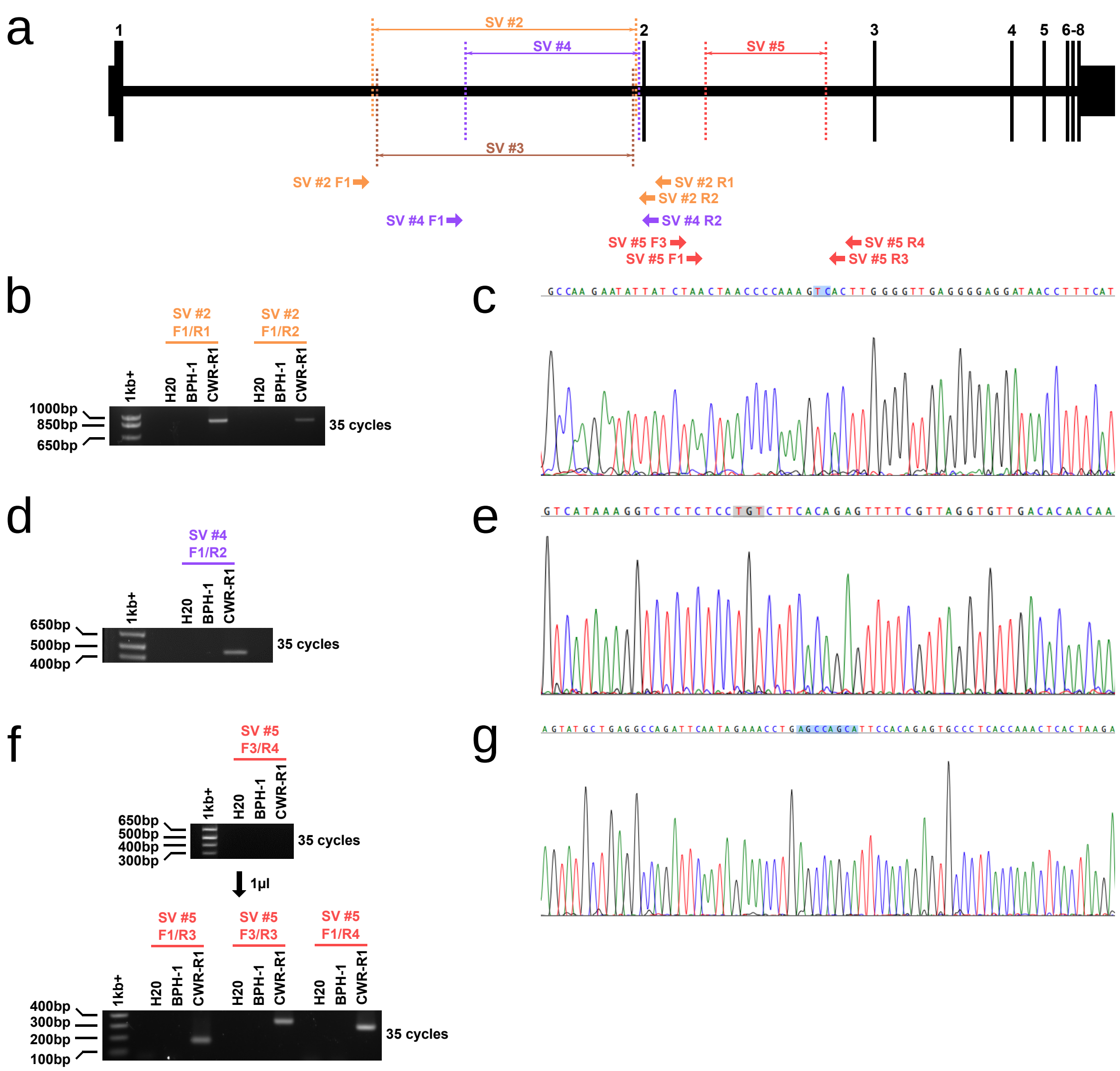


Figure 5.6: Verified SVs from *AR* locus in CWR-R1 cell line. The locations of the four verified deletions within the *AR* gene locus are depicted in **(a)**. The eight exons are marked with vertical bars, with the 5' and 3' UTRs marked with shorter bars at the ends of the locus. The orientation and order of PCR primers used to verify each SV are shown along the bottom. SV #3 was verified in a previous study. Some features and positioning on the figure may not be to scale. Amplified product from PCR validates the presence of **(b)** SV #2, **(d)** SV #4, and **(f)** SV #5. For SV #5, the first round of nested PCR using the outermost primers did not reveal any amplified product, but the second round using the interior primers validates the presence of the variant. Electropherogram peak traces validate the fusion signature for **(c)** SV #2, **(d)** SV #4, and **(g)** SV #5. The highlighted bases represent microhomology of identical bases on both breakpoint boundaries.

framework. It is only by re-running CREST after performing our pipeline to fix soft-clipping errors that there is enough evidence to successfully detect these two SVs. As mentioned, we were unable to validate SV #1 via nested PCR, and SV #6 is believed to be the result of RNA contamination. However, even though SV #6 is not an SV of interest, it still likely represents a true signal in the sample (i.e. RNA contamination) and thus demonstrates how the SHEAR pipeline can improve upon using CREST alone. Additionally, for the other SVs, the SHEAR pipeline improves the confidence of the CREST prediction by increasing the number of soft-clipped reads that are concordant with the breakpoint pairs. For example, SV #2 is supported by 773 soft-clipped reads after running CREST on the default alignment, but has 1114 supporting reads using the SHEAR pipeline.

In our previous work, we also determined that there is a 2030% decrease in copy number in the region of these deletions using multiplex ligation-dependent probe assay (MLPA) [110]. Previously thought to be attributed to a subpopulation with SV #3, this is instead precisely consistent with the heterogeneity level of the deletion of SV #2, as estimated by SHEAR. At the left breakpoint of SV #2, there are 702 reads that are soft-clipped on their 3' end and 3,094 reads that span the breakpoint, while the right breakpoints has 412 reads that are soft-clipped on their 5' end along with 2,306 spanning reads (see Figure 5.7). Using SHEAR's variant frequency estimation scheme for a deletion (see Equation (5.2)), for this SV we would have $A = 3,094 + 2,306 = 5,400$ and $B = 702 + 412 = 1,114$ for an estimated variant frequency of:

$$H = \frac{2B}{A + 2B} = \frac{2 \times 1,114}{5,400 + 2 \times 1,114} = 29.21\%$$

Deletions in this region have been implicated in alternative splicing of the *AR* gene, which can result in resistance to androgen depletion therapy (ADT) in CRPCa patients.

It should be noted that there is a natural bias towards sampling DNA that is more similar to the reference sequence when doing targeted sequencing due to the “baits” used to target specific regions for sequencing. Divergent genomic sequences will be less likely to be sampled, especially when the baits are close to the breakpoints. This bias would be present in this tumor cell line data, meaning that our calculated heterogeneity percentages are likely underestimated by an unknown amount. The agreement between

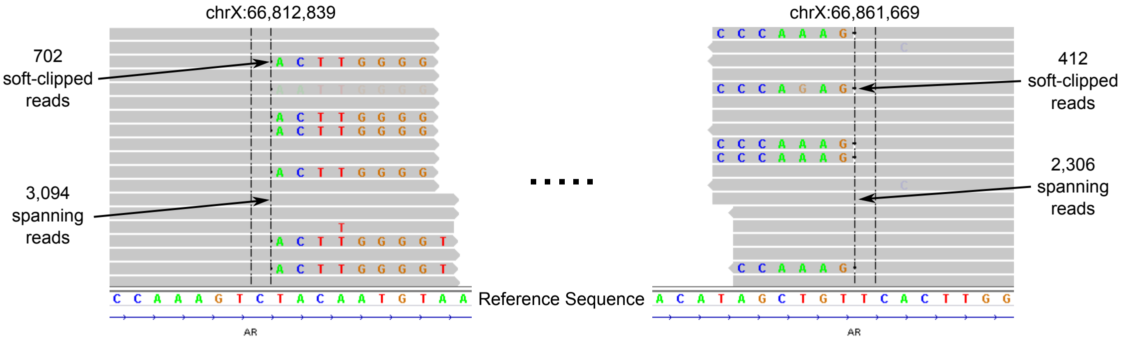


Figure 5.7: Alignment at breakpoints for SV #2 from CWR-R1 cell line. a portion of the alignment of CWR-R1 sequencing data is shown at the two breakpoints for SV #2, located within the *AR* gene, with reference sequence shown along the bottom. At the left breakpoint (chrX:66,812,839) there are 702 reads that are soft-clipped on the 5' side, and 3,094 reads that span the breakpoint. At the right breakpoint (chrX:66,861,669) there are 412 reads that are soft-clipped on the 3' side, and 2,306 reads that span the breakpoint. Using SHEAR's heterogeneity estimation scheme, these numbers predict a variant frequency level of 29.21%. Note that there is a microhomology of 2 bp (TC) that borders both breakpoints, and the fusion of the two sides will only contain one copy of these two nucleotides. Visualization performed using IGV [122, 123].

our computational estimation and the MLPA wet-lab estimation suggests that this underestimation is small, however it is still present nonetheless.

5.4.3 Other tumor data

SHEAR-SV has been applied to numerous other heterogeneous tumor data sets, which has led to a number of interesting findings and insights that have been made possible through our approach. Here we present one example of such a finding:

SHEAR-SV was applied to a series of tumor samples that originated from primary, secondary, and tertiary osteosarcoma tumors, as well as a matched normal tissue sample, all from the same patient. The secondary and tertiary tumors were sequenced after the patient went through a chemotherapy treatment. SHEAR-SV identified a translocation between the 3' untranslated region of the *TP53* gene on chromosome 17 and the first intron on the *KPNA3* gene on chromosome 13. This fusion signature was present in the three tumor samples, but not in the germline sample.

Interestingly, the variant frequency for this translocation drastically increases in the secondary (95.5%) and tertiary (91.8%) tumor samples as compared with the primary tumor sample (13.9%). In other words, this particular variant is present in the original tumor in a smaller subclone, but grows to encompass the vast majority of the post-chemotherapy tumor samples, suggesting that there may be a mechanism for chemotherapy resistance (though more experiments are required to verify any functional impact of this mutation). *TP53* is a well-known tumor suppressor gene that has been found to be associated with many different cancer types, though its unknown what impact a *TP53-KPNA3* fusion may have.

5.5 Discussion and Future Work

There are a number of potential issues with our proposed approach, as well as a variety of extensions that can be explored to address such issues and offer additional functionality through SHEAR.

First, even if the alignment is done perfectly, with no sequencing errors and correct soft-clipping, the estimated heterogeneity level cannot fully account for random fluctuations between the coverage of the reference-like DNA and the variant DNA in the sample. This uncertainty can always be minimized by increasing the depth of the sequencing to reduce the variation in these coverage level fluctuations. Future development for SHEAR will include reporting confidence intervals for heterogeneity estimations to help quantify this uncertainty.

Another issue that can arise is biased SV frequency estimations that results when the input data comes from a capture array. In these cases, specific regions of the genome are extracted, amplified, and sequenced. If a translocation were to be present that joins a genomic region in the capture array with a region outside of the capture array, the assumptions of our variant frequency estimation model do not hold true. Specifically, there would likely not be any reads that span the breakpoint at the non-capture array location and thus the estimated reference depth would be an underestimate. Future versions of SHEAR will investigate ways to address this issue. One possibility is to allow an extra configuration option in SHEAR where users can provide capture array coordinates. If translocations contain breakpoints outside of those coordinates, the variant frequency estimation model would be updated to only look at spanning reads at the capture array breakpoint.

We will also explore the possibility of applying SHEAR to other kinds of heterogeneous sequencing data sets, such as pooled population samples or metagenomic samples that can share a similar reference sequence. We believe that SHEAR's ability to quantify the heterogeneity percentage of predicted SVs makes it an ideal tool to help analyze these types of data sets.

Finally, a large part of future work in this area will be in leveraging the SV frequency estimation model of SHEAR as part of a broader solution to the tumor heterogeneity estimation problem in general. As mentioned previously, an ideal tumor heterogeneity estimation solution would determine the number of different subclones or subpopulations present in a sequencing sample (including potentially normal tissue cells), the variants present in each subpopulation, and the proportion of the sample that each subpopulation composes. This is not yet realistic using existing approaches, but SHEAR-SV can provide an important component (i.e. SV frequency estimations, especially for copy-neutral events) that can combine with SNP/INDEL frequency estimations and tumor purity/ploidy estimations to potentially offer a more complete solution to this problem.

Estimation of tumor heterogeneity, like any statistical inference, can be made more accurate with more observed data. Recent tumor heterogeneity estimation algorithms have combined multiple sources of information to achieve more accurate results, such as ABSOLUTE [112] leveraging karyotype databases to help refine its results from SNP array data, or EXPANDS [119] using both CNV data and SNP allele frequencies in its model. Frequencies of SVs, particularly for copy-neutral events, are a piece of currently under-utilized information that can help improve these estimations by providing more data points to increase accuracy. Furthermore, existing tumor heterogeneity estimation algorithms do not utilize SV breakpoint information, another piece of information that could be used to improve accuracy.

SHEAR's SV frequency estimation approach described in this chapter is a novel scheme that can provide currently under-utilized information (i.e. the breakpoints and frequency of copy-neutral and copy-number-changing SVs as estimated from soft-clipping information). Currently, SHEAR-SV simply outputs the predicted set of variants along with their estimated frequencies, but does not predict the genotypes of tumor subclones. To be of more use, these predicted variants must be grouped into their respective subclones. This is something that existing tumor purity/ploidy estimation algorithms are good at. A natural extension of both SHEAR and the these other tumor heterogeneity estimation algorithms is to combine SHEAR's SV frequencies with the SNP and CNV information normally used by tumor heterogeneity estimation algorithms to create a more accurate prediction model. By grouping together variants into their respective subclones, these extensions can also be used to improve the SHEAR-Assemble pipeline by automating the creation of multiple personal genomic sequences from a heterogeneous sample (see Section 4.5).

One potential avenue to pursue would be to extend the model used by EXPANDS [119] to cluster SVs as well as SNPs. As described previously in Section 5.2, EXPANDS works by clustering the cell frequency probability densities associated with each SNP to find common peaks that identify tumor subclones and their respective variants. The SV frequency estimations done by SHEAR-SV can be used to generate analogous cell frequency probability densities for SVs as well. By clustering SNPs and SVs simultaneously, it may be possible to achieve more accurate tumor heterogeneity estimations.

Another approach may be to combine the outputs of THetA [116, 117] with that of SHEAR-SV to predict more complete subclone genotypes. The output of THetA is a set of subclones, each characterized by its frequency and a copy number in each segment of the genome. Thus, it does not contain SNPs or copy-neutral SVs in predicted subclone genotypes. SHEAR-SV predicts variant frequencies for both of these, and these frequencies can be assigned to predicted subclones by determining which subclone has a frequency and copy number in the variant's segment that most closely matches the predicted variant frequency.

Finally, one last potential way to combine SHEAR-SV with existing methodologies is to use deletion and duplication predictions from SHEAR-SV to help guide DNA segmentation algorithms. Both the EXPANDS and THetA models, as well as other tumor heterogeneity estimation algorithms, use copy number segments as input data, which are generated by DNA segmentation algorithms such as BIC-seq [120]. In the case of small frequency subclones with unique copy number changes, those variants may not be reflected in the estimated segmentation because the gain or loss in read depth can appear to be noise when viewed in the context of the whole heterogeneous sequencing sample. SHEAR-SV can predict CNVs with very small variant frequencies, and those breakpoints can be used to either specify required segments during the DNA segmentation process or post-process the segmentation results. This will result in more accurate copy number segments to be used by the tumor heterogeneity estimation models.

Chapter 6

Conclusion and Discussion

6.1 Discussion and Future Work

Each of the three aims presented in this thesis have included discussions on opportunities for future improvements for each individual technique. However, there are also opportunities for future improvement that combine several of these concepts.

For analyzing insertional mutagenesis data, one of the challenges that occurs is in handling the situation of multi-tumor mice, as discussed previously in Section 3.4.5. These tumors can contain highly-correlated sets of insertion sites if they are evolutionarily-related, which may result in spurious patterns if this situation is not accounted for. Our approach uses a modification of support counts and the overall tumor counts when evaluating the significance of CCIs that are supported by multi-tumor mice. A more thorough approach would be to identify the evolutionary relationships (if any) between tumors in the same mouse, and then to use that information to develop a more robust model for the statistical evaluation of CCIs. The proposed extensions to SHEAR discussed in Section 5.5 for estimating tumor subclones could also be leveraged to evaluate such relationships between tumors. For example, the data from all tumors from the same mouse could be pooled together, and a tumor heterogeneity estimation algorithm could be applied to it. This could identify the insertion sites that are shared by all tumors as well as those that are present in only certain tumors, and thus create a more clear picture of the evolutionary relationship between such tumors.

On a related note, the variant frequency estimation algorithms developed in Chapter 5 could also be applied to account for potential heterogeneity in insertional mutagenesis data, which has not been considered by existing approaches. Just like other tumors, the tumors formed via insertional mutagenesis may also be characterized by genetic heterogeneity. This possibility has clear implications for discovering CCIs in such data sets, and the extent and impact of this situation should be further explored.

It should also be emphasized that the technology used to generate these types of data is evolving rapidly, and the challenges and problems discussed in this thesis may also be addressed by entirely new types of genetic data in the near future. We had previously discussed the impact that SMRT sequencing technology may have on the future of personal genomic assembly in Section 4.5. Future work will explore how new technologies can be integrated with or expand off of the techniques presented here, such as using single-cell sequencing [124] for categorizing tumor heterogeneity, or using CRISPR [125] for validating the functional impact of CCIs identified in insertional mutagenesis data.

6.2 Summary of Contributions

Throughout this dissertation, we have presented a number of novel computational techniques for analyzing tumor DNA data, and demonstrated their efficacy. In Chapter 3, we presented a framework for efficiently identifying co-occurring sets of insertion sites in insertional mutagenesis data. This algorithm improved upon existing techniques by being more efficient, identifying higher order patterns of CCIs, and handling scenarios of multiple tumors originating from the same mouse. The results from these analyses can be used to identify potential genes of interest that may be oncogenic or else act as potential therapy targets for cancer treatment. In Chapter 4, we presented SHEAR, an open-source and easy-to-use software package that can be used for assembling personal genomic sequences. The SHEAR framework improves upon existing approaches by focusing on discordant regions of an alignment to improve efficiency and assemble accurate personal genomic sequences. Such sequences can be used as references for RNA-seq or ChIP-seq analysis of tumor cells, which is important to explore in order to determine the downstream affects of oncogenic mutations on gene expression and gene regulation. Finally, in Chapter 5, we presented another component of the SHEAR software package that is used for estimating variant frequencies for SVs found in heterogeneous tumor samples. Existing tumor heterogeneity estimation algorithms have not focused on SV-specific variant frequencies, especially not for copy-neutral events such as inversions and translocations. SHEAR has been used effectively to identify heterogeneous variants in tumor samples for a variety of cancer types, which has contributed to new insights into the underlying genetic mechanisms of cancer.

References

[1] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001.

[2] International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–945, 2004.

[3] The 1000 Genomes Project Consortium. A map of human genome variation from population-scale sequencing. *Nature*, 467(7319):1061–1073, 2010.

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

[5] The International HapMap 3 Consortium. Integrating common and rate genetic variation in diverse human populations. *Nature*, 467(7311):52–58, 2010.

[6] The ENCODE Project Consortium. The ENCODE (ENCyclopedia Of DNA Elements) project. *Science*, 306(5696):636–640, 2004.

[7] The ENCODE Project Consortium. Identification and analysis of funcitonal elements in 1% of the human genome by the ENCODE pilot project. *Nature*, 447 (7146):799–816, 2007.

[8] The ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74, 2012.

[9] David A Wheeler and Linghua Wang. From human genome to cancer genome: the first decade. *Genome Research*, 23(7):1054–1062, 2013.

[10] Jay Shendure and Hanlee Ji. Next-generation DNA sequencing. *Nature Biotechnology*, 26(10):1135–1145, 2008.

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

[12] Heng Li and Richard Durbin. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics*, 26(5):589–595, 2010.

[13] BWA-MEM. URL `http://bio-bwa.sourceforge.net`.

[14] Heng Li. Exploring single-sample SNP and INDEL calling with whole-genome *de novo* assembly. *Bioinformatics*, 28(14):1838–1844, 2012.

[15] Ben Langmead, Cole Trapnell, Mihai Pop, and Steven L Salzberg. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biology*, 10(3):R25, 2009.

[16] Thomas D Wu and Serban Nacu. Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics*, 26(7):873–881, 2010.

[17] Paul Medvedev, Monica Stanciu, and Michael Brudno. Computational methods for discovering structural variation with next-generation sequencing. *Nature Methods*, 6(Suppl 11):S13–S20, 2009.

[18] Can Alkan, Bradley P Coe, and Evan E Eichler. Genome structural variation discovery and genotyping. *Nature Review Genetics*, 12(5):363–376, 2011.

[19] Mehdi Pirooznia, Fernando S Goes, and Peter P Zandi. Whole-genome CNV analysis: advances in computational approaches. *Frontiers in Genetics*, 6(1):138, 2015.

[20] Ken Chen, John W Wallis, Michael D McLellan, David E Larson, Joelle M Kalicki, Craig S Pohl, Sean D McGrath, Michael C Wendl, Qunyuan Zhang, Devin P Locke, Xiaoqi Shi, Robert S Fulton, Timothy J Ley, Richard K Wilson, Li Ding, and Elaine R Mardis. BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nature Methods*, 6(9):677–681, 2009.

[21] Aaron R Quinlan, Royden A Clark, Svetlana Sokolova, Mitchell L Leibowitz, Yujun Zhang, Matthew E Hurles, Joshua C Mell, and Ira M Hall. Genome-wide mapping and assembly of structural variant breakpoints in the mouse genome. *Genome Research*, 20(5):623–635, 2010.

[22] Fereydoun Hormozdiari, Iman Hajirasouliha, Andrew McPherson, Evan E Eichler, and S Cenk Sahinalp. Simultaneous structural variation discovery among multiple paired-end sequenced genomes. *Genome Research*, 21(12):2203–2212, 2011.

[23] Seunghak Lee, Fereydoun Hormozdiari, Can Alkan, and Michael Brudno. MoDIL: detecting small indels from clone-end sequencing with mixtures of distributions. *Nature Methods*, 6(7):473–474, 2009.

[24] Jan O Korbel, Alexej Abyzov, Xinmeng Jasmine Mu, Nicholas Carriero, Philip Cayting, Zhengdong Zhang, Michael Snyder, and Mark B Gerstein. PEMer: a computational framework with simulation-based error models for inferring genomic structural variants from massive paired-end sequencing data. *Genome Biology*, 10 (2):R23, 2009.

[25] Kai Ye, Marcel H Schulz, Quan Long, Rolf Apweiler, and Zemin Ning. Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics*, 25(21):2865–2871, 2009.

[26] Jianmin Wang, Charles G Mullighan, John Easton, Stefan Roberts, Sue L Heatley, Jing Ma, Michael C Rusch, Ken Chen, Christopher C Harris, Li Ding, Linda Holmfeldt, Debbie Payne-Turner, Xian Fan, Lei Wei, David Zhao, John C Obenauer, Clayton Naeve, Elaine R Mardis, Richard K Wilson, James R Downing, and Jinghui Zhang. CREST maps somatic structural variation in cancer genomes with base-pair resolution. *Nature Methods*, 8(8):652–654, 2011.

[27] Seungtai Yoon, Zhenyu Xuan, Vladimir Makarov, Kenny Ye, and Jonathan Sebat. Sensitive and accurate detection of copy number variants using read depth of coverage. *Genome Research*, 19(9):1586–1592, 2009.

[28] Alexej Abyzov, Alexander E Urban, Michael Snyder, and Mark Gerstein. CNVnator: an approach to discover, genotype, and characterize typical and atypical CNVs from family and population genome sequencing. *Genome Research*, 21(6): 974–984, 2011.

[29] Ryan E Mills, Klaudia Walter, Chip Stewart, Robert E Handsaker, Ken Chen, Can Alkan, Alexej Abyzov, Seungtai Chris Yoon, Kai Ye, R Keira Cheetham, Asif Chinwalla, Donald F Conrad, Yutao Fu, Fabian Grubert, Iman Hajirasouliha, Fereydoun Hormozdiari, Lilia M Iakoucheva, Zamin Iqbal, Shuli Kang, Jeffrey M Kidd, Miriam K Konkel, Joshua Korn, Ekta Khurana, Deniz Kural, Hugo Y K Lam, Jing Leng, Ruiqiang Li, Yingrui Li, Chang-Yun Lin, Ruibang Luo, Xinmeng Jasmine Mu, James Nemesh, Heather E Peckham, Tobias Rausch, Aylwyn Scally, Xinghua Shi, Michael P Stromberg, Adrian M Stütz, Alexander Eckehart Urban, Jerilyn A Walker, Jiantao Wu, Yujun Zhang, Zhengdong D Zhang, Mark A Batzer, Li Ding, Gabor T Marth, Gil McVean, Jonathan Sebat, Michael Snyder, Jun Wang, Kenny Ye, Evan E Eichler, Mark B Gerstein, Matthew E Hurles, Charles Lee, Steven A McCarroll, Jan O Korbel, and 1000 Genomes Project. Mapping copy number variation by population-scale genome sequencing. *Nature*, 470(7332):59–65, 2011.

[30] Jin Zhang and Yufeng Wu. SVseq: an approach for detecting exact breakpoints of deletions with low-coverage sequence data. *Bioinformatics*, 27(23):3228–3234, 2011.

[31] Jacob J Michaelson and Jonathan Sebat. forestSV: structural variant discovery through statistical learning. *Nature Methods*, 9(8):819–821, 2012.

[32] Yue Jiang, Yadong Wang, and Michael Brudno. PRISM: pair-read informed split-read mapping for base-pair level detection of insertion, deletion and structural variants. *Bioinformatics*, 28(20):2576–2583, 2012.

[33] Tobias Rausch, Thomas Zichner, Andreas Schlattl, Adrian M Stütz, Vladimir Benes, and Jan O Korbel. DELLY: structural variant discovery by integrated paired-end and split-read analysis. *Bioinformatics*, 28(18):i333–i339, 2012.

[34] Suzanne S Sindi, Selim Önal, Luke C Peng, Hsin-Ta Wu, and Benjamin J Raphael.

An integrative probabilistic model for identification of structural variation in sequencing data. *Genome Biology*, 13(3):R22, 2012.

[35] Ryan M Layer, Colby Chiang, Aaron R Quinlan, and Ira M Hall. LUMPY: a probabilistic framework for structural variant discovery. *Genome Biology*, 15(6): R84, 2012.

[36] Christoph Bartenhagen and Martin Dugas. Robust and exact structural variation detection with paired-end and soft-clipped alignments: SoftSV compared with eight algorithms. *Briefings in Bioinformatics*, 17(1):51–62, 2016.

[37] Douglas Hanahan and Robert A Weinberg. The hallmarks of cancer. *Cell*, 100(1): 57–70, 2000.

[38] Douglas Hanahan and Robert A Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144(5):646–674, 2011.

[39] Li Ding, Michael C Wendl, Daniel C Koboldt, and Elaine R Mardis. Analysis of next-generation genomic data in cancer: accomplishments and challenges. *Human Molecular Genetics*, 19(R2):R188–R196, 2010.

[40] Matthew Meyerson, Stacey Gabriel, and Gad Getz. Advances in understanding cancer genomes through second-generation sequencing. *Nature Reviews Genetics*, 11(10):685–696, 2010.

[41] A G Uren, J Kool, A Berns, and M van Lohuizen. Retroviral insertional mutagenesis: past, present and future. *Oncogene*, 24(52):7656–7672, 2005.

[42] Zoltán Ivics, Perry B Hackett, Ronald H Plasterk, and Zsuzsanna Izsvák. Molecular reconstruction of *Sleeping Beauty*, a *Tc1*-like transposon from fish, and its transposition in human cells. *Cell*, 91(4):501–510, 1997.

[43] Lans S Collier, Corey M Carlson, Shruthi Ravimohan, Adam J Dupuy, and David A Largaespada. Cancer gene discovery in solid tumours using transposon-based somatic mutagenesis in the mouse. *Nature*, 436(7048):272–276, 2005.

[44] Mark S Boguski. Comparative genomics: the mouse that roared. *Nature*, 420 (6915):515–516, 2002.

[45] Mouse Genome Sequencing Consortium. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420(6915):520–562, 2002.

[46] Corey M Carlson and David A Largaespada. Insertional mutagenesis in mice: new perspectives and tools. *Nature Reviews Genetics*, 6(7):568–580, 2005.

[47] Timothy K Starr, Raha Allae, Kevin A T Silverstein, Rodney A Staggs, Aaron L Sarver, Tracy L Bergemann, Mihir Gupta, M Gerard O'Sullivan, Ilze Matise, Adam J Dupuy, Lara S Collier, Scott Powers, Ann L Oberg, Yan W Asmann, Stephen N Thibodeau, Lino Tessarollo, Neal G Copeland, Nancy A Jenkins, Robert T Cormier, and David A Largaespada. A transposon-based genetic screen in mice identifies genes altered in colorectal cancer. *Science*, 323(5922):1747–1750, 2009.

[48] Vincent W Keng, Augusto Villanueva, Derek Y Chiang, Adam J Dupuy, Barbara J Ryan, Ilze Matise, Kevin A T Silverstein, Aaron Sarver, Timothy K Starr, Keiko Akagi, Lino Tessarollo, Lara S Collier, Scott Powers, Scott W Lowe, Nancy A Jenkins, Neal G Copeland, Josep M Llovet, and David A Largaespada. A conditional transposon-based insertional mutagenesis screen for genes associated with mouse hepatocellular carcinoma. *Nature Biotechnology*, 27(3):264–274, 2009.

[49] Marco Ranzani, Stefano Annunziato, David J Adams, and Eugenio Montini. Cancer gene discovery: exploiting insertional mutagenesis. *Molecular Cancer Research*, 11(10):1141–1158, 2013.

[50] Raha A Been, Michael A Linden, Courtney J Hager, Krista J DeCoursin, Juan E Abrahante, Sean R Landman, Michael Steinbach, Aaron L Sarver, David A Largaespada, and Timothy K Starr. Genetic signature of histiocytic sarcoma revealed by a Sleeping Beauty transposon genetic screen in mice. *PLOS ONE*, 9(5): e97280, 2014.

[51] Casey Dorr, Callie Janik, Madison Weg, Raha A Been, Justin Bader, Ryan Kang, Brandon Ng, Lindsey Foran, Sean R Landman, M Gerard O'Sullivan, Michael Steinbach, Aaron L Sarver, Kevin A T Silverstein, David A Largaespada, and Timothy K Starr. Transposon mutagenesis screen identifies potential lung cancer

drivers and *CUL3* as a tumor suppressor. *Molecular Cancer Research*, 13(8):1238–1247, 2015.

[52] Jeroen de Ridder, Anthony Uren, Jaap Kool, Marcel Reinders, and Lodewyk Wessels. Detecting statistically significant common insertion sites in retroviral insertional mutagenesis screens. *PLoS Computational Biology*, 2(12):e166, 2006.

[53] Benjamin T Brett, Katherine E Berquam-Vrieze, Kishore Nannapaneni, Jian Huang, Todd E Scheetz, and Adam J Dupuy. Novel molecular and computational methods improve the accuracy of insertion site analysis in Sleeping Beauty-induced tumors. *PLOS ONE*, 6(6):e244668, 2011.

[54] Aaron L Sarver, Jesse Erdman, Tim Starr, David A Largaespada, and Kevin A T Silverstein. TAPDANCE: an automated tool to identify and annotate transposon insertion CISs and associations between CISs from next generation sequence data. *BMC Bioinformatics*, 13(1):154, 2012.

[55] Jeroen de Ridder, Jaap Kool, Anthony Uren, Jan Bot, Lodewyk Wessels, and Marcel Reinders. Co-occurrence analysis of insertional mutagenesis data reveals cooperating oncogenes. *Bioinformatics*, 23(13):i133–i141, 2007.

[56] Jiawei Han, Hong Cheng, Dong Xin, and Xifeng Yan. Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery*, 15 (1):55–86, 2007.

[57] Murray Rosenblatt. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837, 1956.

[58] Emanuel Parzen. On estimation of a probability density function and more. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

[59] Kim D Pruitt, Tatiana Tatusova, and Donna R Maglott. NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Research*, 35(Database issue):D61–D65, 2007.

[60] Tracy L Bergemann, Timothy K Starr, Haoyu Yu, Michael Steinbach, Jesse Erdmann, Yun Chen, Robert T Cormier, David A Largaespada, and Kevin A T Silverstein. New methods for finding common insertion sites and co-occurring common

insertion sites in transposon- and virus-based genetic screens. *Nucleic Acids Research*, 40(9):3822–3833, 2012.

[61] Michael Steinbach, Haoyu Yu, Sean Landman, and Vipin Kumar. Identification of co-occurring insertions in cancer genomes using association analysis. *International Journal of Data Mining and Bioinformatics*, 10(1):65–82, 2014.

[62] Rakesh Agrawal, Tomasz Imielinski, and Arun Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on management of data*, pages 207–216, Washington DC, USA, May 1993.

[63] Rakesh Agrawal, Tomasz Imielinski, and Arun Swami. Database mining: a performance perspective. *IEEE Transactions on Knowledge and Data Engineering*, 5 (6):914–925, 1993.

[64] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th international conference on very large data bases*, pages 487–499, Santiago, Chile, September 1994.

[65] Mohammed J Zaki, Srinivasan Parthasarathy, Mitsunori Ogihara, and Wei Li. New algorithms for fast discovery of association rules. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, pages 283–286, Newport Beach, CA, USA, August 1997.

[66] Jiawei Han, Jian Pei, and Yiwen Yin. Mining frequent patterns without candidate generation. In *Proceedings of the 1993 ACM SIGMOD international conference on management of data*, pages 1–12, Dallas, TX, USA, May 2000.

[67] Aaron R Quinlan and Ira M Hall. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–842, 2010.

[68] Jesse D Riordan, Luke J Drury, Ryan P Smith, Benjamin T Brett, Laura M Rogers, Todd E Scheetz, and Adam J Dupuy. Sequencing methods and datasets to improve functional interpretation of *sleeping beauty* mutagenesis screens. *BMC Bioinformatics*, 15(1):1150, 2014.

[69] Christian Borgelt. An implementation of the FP-growth algorithm. In *Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations*, pages 1–5, Chicago, IL, USA, August 2005.

[70] Christian Borgelt. Frequent item set mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(6):437–456, 2012.

[71] Mohammed J Zaki and Mitsunori Ogihara. Theoretical foundations of association rules. In *Proceedings of the 3rd ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, pages 71–78, Seattle, WA, USA, June 1998.

[72] Nicolas Pasquier, Yves Bastide, Rafik Taouil, and Lotfi Lakhal. Discovering frequent closed itemsets for association rules. In *Proceedings of the 17th International Conference on Database Theory*, pages 398–416, Jerusalem, Israel, January 1999.

[73] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, and Jill P Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *PNAS*, 102(43):15545–15550, 2005.

[74] John D Storey. A direct approach to false discovery rates. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):479–498, 2002.

[75] George W Cobb and Yung-Pin Chen. An application of Markov chain Monte Carlo to community ecology. *The American Mathematical Monthly*, 110(4):265–288, 2003.

[76] Aristides Gionis, Heikki Mannila, Taneli Mielikäinen, and Panayiotis Tsaparas. Assessing data mining results via swap randomization. *ACM Transactions on Knowledge Discovery from Data*, 1(3):14, 2007.

[77] Geoffrey I Webb. Self-sufficient itemsets: an approach to screening potentially interesting associations between items. *ACM Transactions on Knowledge Discovery from Data*, 4(1):3, 2010.

[78] Geoffrey I Webb and Jilles Vreeken. Efficient discovery of the most interesting associations. *ACM Transactions on Knowledge Discovery from Data*, 8(3):15, 2014.

[79] Keiko Akagi, Takeshi Suzuki, Robert M Stephens, Nancy A Jenkins, and Neal G Copeland. RTCGD: retroviral tagged cancer gene database. *Nucleic Acids Research*, 32(Database issue):D423–D527, 2004.

[80] Kenneth L Abbott, Erik T Nyre, Juan Abrahante, Yen-Yi Ho, Rachel Isaksson Vogel, and Timothy K Starr. The Candidate Cancer Gene Database: a database of cancer driver genes from forward genetic screens in mice. *Nucleic Acids Research*, 43(Database issue):D844–D848, 2015.

[81] Armin Zebisch, Philipp B Staber, Ali Delavar, Claudia Bodner, Karin Hiden, Katja Fischereder, Manickam Janakiraman, Werner Linkesch, Holger W Auner, Werner Emberger, Christian Windpassingerand Michael G Schimek, Gerald Hoefler, Jakob Troppmair, and Heinz Sill. Two transforming *C-RAF* germ-line mutations identified in patients with therapy-related acute myeloid leukemia. *Cancer Research*, 66 (7):3401–3408, 2006.

[82] Rizwan Haq and David E Fisher. Biology and clinical relevance of the micropthalmia family of transcription factors in human cancer. *Journal of Clinical Oncology*, 29(25):3474–3482, 2011.

[83] Hui Xiong, Pang-Ning Tan, and Vipin Kumar. Mining strong affinity association patterns in data sets with skewed support distribution. In *Proceedings of the 3rd IEEE International Conference on Data Mining*, pages 387–394, Melbourne, FL, USA, November 2003.

[84] Hui Xiong, Pang-Ning Tan, and Vipin Kumar. Hyperclique pattern discovery. *Data Mining and Knowledge Discovery*, 13(2):219–242, 2006.

[85] Peter H Sudmant, Tobias Rausch, Eugene J Gardner, Robert E Handsaker, Alexej Abyzov, John Huddleston, Yan Zhang, Kai Ye, Goo Jun, Markus Hsi-Yang Fritz, Miriam K Konkel, Ankit Malhotra, Adrian M Stütz, Xinghua Shi, Francesco Paolo Casale, Jieming Chen, Fereydoun Hormozdiari, Gargi Dayama, Ken Chen, Maika

Malig, Mark J P Chaisson, Klaudia Walter, Sascha Meiers, Seva Kashin, Erik Garrison, Adam Auton, Hugo Y K Lam, Xinmeng Jasmine Mu, Can Alkan, Danny Antaki, Taejeong Bae, Eliza Cerveira, Peter Chines, Zechen Chong, Laura Clarke, Elif Dal, Li Ding, Sarah Emery, Xian Fan, Madhusudan Gujral, Fatma Kahveci, Jeffrey M Kidd, Yu Kong, Eric-Wubbo Lameijer, Shane McCarthy, Paul Flicek, Richard A Gibbs, Gabor Marth, Christopher E Mason, Androniki Menelaou, Donna M Muzny, Bradley J Nelson, Amina Noor, Nicholas F Parrish, Matthew Pendleton, Andrew Quitadamo, Benjamin Raeder, Eric E Schadt, Mallory Romanovitch, Andreas Schlattl, Robert Sebra, Andrey A Shabalin, Andreas Untergasser, Jerilyn A Walker, Min Wang, Fuli Yu, Chengsheng Zhang, Jing Zhang, Xiangqun Zheng-Bradley, Wanding Zhou, Thomas Zichner, Jonathan Sebat, Mark A Batzer, Steven A McCarroll, The 1000 Genomes Project Consortium, Ryan E Mills, Mark B Gerstein, Ali Bashir, Oliver Stegle, Scott E Devine, Charles Lee, Evan E Eichler, and Jan O Korbel. An integrated map of structural variation in 2,504 human genomes. *Nature*, 526(7571):75–81, 2015.

[86] Garrett M Frampton, Alex Fichtenholtz, Geoff A Otto, Kai Wang, Sean R Downing, Jie He, Michael Schnall-Levin, Jared White, Eric M Sanford, Peter An, James Sun, Frank Juhn, Kristina Brennan, Kiel Iwanik, Ashley Maillet, Jamie Buell, Emily White, Mandy Zhao, Sohail Balasubramanian, Selmira Terzic, Tina Richards, Vera Banning, Lazaro Garcia, Kristen Mahoney, Zac Zwirko, Amy Donahue, Himisha Beltran, Juan Miguel Mosquera, Mark A Rubin, Snjezana Dogan, Cyrus V Hedvat, Michael F Berger, Lajos Pusztai, Matthias Lechner, Chris Boshoff, Mirna Jarosz, Christine Vietz, Alex Parker, Vincent A Miller, Jeffrey S Ross, John Curran, Maureen T Cronin, Philip J Stephens, Doron Lipson, and Roman Yelensky. Development and validation of a clinical cancer genomic profiling test based on massively parallel DNA sequencing. *Nature Biotechnology*, 31(11): 1023–1031, 2013.

[87] Jeffrey M Kidd, Gregory M Cooper, William F Donahue, Hillary S Hayden, Nick Sampas, Tina Graves, Nancy Hansen, Brian Teague, Can Alkan, Francesca Antonacci, Eric Haugen, Troy Zerr, N Alice Yamada, Peter Tsang, Tera L Newman, Eray Tüzün, Ze Cheng, Heather M Ebling, Nadeem Tusneem, Robert David,

Will Gillett, Karen A Phelps, Molly Weaver, David Saranga, Adrianne Brand, Wei Tao, Erik Gustafson, Kevin McKernan, Lin Chen, Maika Malig, Joshua D Smith, Joshua M Korn, Steven A McCarroll, David A Altshuler, Daniel A Peiffer, Michael Dorschner, John Stamatoyannopoulos, David Schwartz, Deborah A Nickerson, James C Mullikin, Richard K Wilson, Laurakay Bruhn, Maynard V Olson, Rajinder Kaul, Douglas R Smith, and Evan E Eichler. Mapping and sequencing of structural variation from eight human genomes. *Nature*, 453(7191):56–64, 2008.

[88] Tobias Rausch, Sergey Koren, Gennady Denisov, David Weese, Anne-Katrin Emde, Andreas Döring, and Knut Reinert. A consistency-based consensus algorithm for *de novo* and reference-guided sequence assembly of short reads. *Bioinformatics*, 25(9):1118–1124, 2009.

[89] Juliane D Klein, Stephan Ossowski, Korbinian Schneeberger, Detlef Weigel, and Daniel H Huson. LOCAS — a low coverage assembly tool for resequencing projects. *PLoS ONE*, 6(8):e23455, 2011.

[90] Korbinian Schneeberger, Stephan Ossowski, Felix Ott, Juliane D Klein, Xi Wang, Christa Lanz, Lisa M Smith, Jun Cao, Joffrey Fitz, Norman Warthmann, Stefan R Henz, Daniel H Huson, and Detlef Weigel. Reference-guided assembly of four diverse *Arabidopsis thaliana* genomes. *PNAS*, 108(25):10249–10254, 2011.

[91] Jaebum Kim, Denis M Larkin, Qingle Cai, Asan, Yongfen Zhang, Ri-Li Ge, Loretta Auvil, Boris Capitanu, Guojie Zhang, Harris A Lewin, and Jian Ma. Reference-assisted chromosome assembly. *PNAS*, 110(5):1785–1790, 2013.

[92] Xiangchao Gan, Oliver Stegle, Jonas Behr, Joshua G Steffen, Philipp Drewe, Katie L Hildebrand, Rune Lyngsoe, Sebastian J Schultheiss, Edward J Osborne, Vipin T Sreedharan, André Kahles, Regina Bohnert, Géraldine Jean, Paul Derwent, Paul Kersey, Eric J Belfield, Nicholas P Harberd, Eric Kemen, Christopher Toomajian, Paula X Kover, Richard M Clark, Gunnar Rätsch, and Richard Mott. Multiple reference genomes and transcriptomes for *Arabidopsis thaliana*. *Nature*, 477(7365):419–423, 2011.

[93] Sean R Landman, Tae Hyun Hwang, Kevin A T Silverstein, Yingming Li, Scott M

Dehm, Michael Steinbach, and Vipin Kumar. SHEAR: sample heterogeneity estimation and assembly by reference. *BMC Genomics*, 15(1):84, 2014.

[94] Mark A DePristo, Eric Banks, Ryan Poplin, Kiran V Garimella, Jared R Maguire, Christopher Hartl, Anthony A Philippakis, Guillermo del Angel, Manuel A Rivas, Matt Hanna, Aaron McKenna, Tim J Fennell, Andrew M Kernytsky, Andrey Y Sivachenko, Kristian Cibulskis, Stacey B Gabriel, David Altshuler, and Mark J Daly. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, 43(5):491–498, 2011.

[95] Rasmus Nielsen, Joshua S Paul, Anders Albrechtsen, and Yun S Song. Genotype and SNP calling from next-generation sequencing data. *Nature Reviews Genetics*, 12(6):443–451, 2011.

[96] Daniel R Zerbino and Ewan Birney. Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome Research*, 18(5):821–829, 2008.

[97] Ruiqiang Li, Hongmei Zhu, Jue Ruan, Wubin Qian, Xiaodong Fang, Zhongbin Shi, Yingrui Li, Shengting Li, Gao Shan, Karsten Kristiansen, Songgang Li, Huanming Yang, Jian Wang, and Jun Wang. De novo assembly of human genomes with massively parallel short read sequencing. *Genome Research*, 20(2):265–272, 2010.

[98] Sante Gnerre, Iain MacCallum, Dariusz Przybylski, Filipe J Ribeiro, Joshua N Burton, Bruce J Walker, Ted Sharpe, Giles Hall, Terrance P Shea, Sean Sykes, Aaron M Berlin, Daniel Aird, Maura Costello, Riza Daza, Louise Williams, Robert Nicol, Andreas Gnirke, Chad Nusbaum, Eric S Lander, and David B Jaffe. High-quality draft assemblies of mammalian genomes from massively parallel sequence data. *PNAS*, 108(4):1513–1518, 2011.

[99] Gerton Lunter and Martin Goodson. Stampy: a statistical algorithm for sensitive and fast mapping of Illumina sequence reads. *Genome Research*, 21(6):936–939, 2011.

[100] 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, 2010.

[101] Heng Li, Bob Handsaker, Alex 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.

[102] Picard. URL `http://picard.sourceforge.net`.

[103] Lixing Yang, Lovelace J Luquette, Nils Gehlenborg, Ruibin Xi, Psalm S Haseley, Chih-Heng Hsieh, Chengsheng Zhang, Xiaojia Ren, Alexei Protopopov, Lynda Chin, Raju Kucherlapati, Charles Lee, and Peter J Park. Diverse mechanisms of somatic structural variations in human cancer genomes. *Cell*, 153(4):919–929, 2013.

[104] Donald F Conrad, Christine Bird, Ben Blackburne, Sarah Lindsay, Lira Mamanova, Charles Lee, Daniel J Turner, and Matthew E Hurles. Mutation spectrum revealed by breakpoint sequencing of human germline CNVs. *Nature Genetics*, 42(5):385–391, 2010.

[105] Jeffrey M Kidd, Tina Graves, Tera L Newman, Robert Fulton, Hillary S Hayden, Maika Malig, Joelle Kallicki, Rajinder Kaul, Richard K Wilson, and Evan E Eichler. A human genome structural variation sequencing resource reveals insights into mutational mechanisms. *Cell*, 143(5):837–847, 2010.

[106] Mark J P Chaisson, Richard K Wilson, and Evan E Eichler. Genetic variation and the *de novo* assembly of human genomes. *Nature Reviews Genetics*, 16(11): 627–640, 2015.

[107] FALCON: experimental PacBio diploid assembler. URL `http://github.com/PacificBiosciences/FALCON`.

[108] Mark J P Chaisson, John Huddleston, Megan Y Dennis, Peter H Sudmant, Maika Malig, Fereydoun Hormozdiari, Francesca Antonacci, Urvashi Surti, Richard

Sandstrom, Matthew Boitano, Jane M Landolin, John A Stamatoyannopoulos, Michael W Hunkapiller, Jonas Korlach, and Evan E Eichler. Resolving the complexity of the human genome using single-molecule sequencing. *Nature*, 517(7536): 608–611, 2015.

[109] Konstantin Berlin, Sergey Koren, Chen-Shan Chin, James P Drake, Jane M Landolin, and Adam M Phillippy. Assembling large genomes with single-molecule sequencing and locality-sensitive hashing. *Nature Biotechnology*, 33(6):623–630, 2015.

[110] Yingming Li, Tae Hyun Hwang, LeAnn Oseth, Adam Hauge, Robert L Vessella, Stephen C Schmechel, Betsy Hirsch, Kenneth B Beckman, Kevin A Silverstein, and Scott M Dehm. AR intragenic deletions linked to androgen receptor splice variant expression and activity in models of prostate cancer progression. *Oncogene*, 31 (45):4759–4767, 2012.

[111] Petr Danecek, Adam Auton, Goncalo Abecasis, Cornelius A Albers, Eric Banks, Mark A DePristo, Robert E Handsaker, Gerton Lunter, Gabor T Marth, Stephen T Sherry, Gilean McVean, Richard Durbin, and 1000 Genomes Project Analysis Group. The variant call format and VCFtools. *Bioinformatics*, 27(15):2156–2158, 2011.

[112] Scott L Carter, Kristian Cibulskis, Elena Helman, Aaron McKenna, Hui Shen, Travis Zack, Peter W Laird, Robert C Onofrio, Wendy Winckler, Barbara A Weir, Rameen Beroukhim, David Pellman, Douglas A Levine, Eric S Lander, Matthew Meyerson, and Gad Getz. Absolute quantification of somatic DNA alterations in human cancer. *Nature Biotechnology*, 30(5):413–421, 2012.

[113] Peter Van Loo, Silje H Nordgard, Ole Christian Lingjærde, Hege G Russnes, Inga H Rye, Wei Sun, Victor J Weigman, Peter Marynen, Anders Zetterberg, Bjørn Naume, Charles M Perou, Anne-Lise Børresen-Dale, and Vessela N Kristensen. Allele-specific copy number analysis of tumors. *PNAS*, 107(39):16910–16915, 2010.

[114] Arief Gusnanto, Henry M Wood, Yudi Pawitan, Pamela Rabbitts, and Stefano Berri. Correcting for cancer genome size and tumour cell content enables better

estimation of copy number alterations from next-generation sequence data. *Bioinformatics*, 28(1):40–47, 2012.

[115] Lei Bao, Minya Pu, and Karen Messer. AbsCN-seq: a statistical method to estimate tumor purity, ploidy and absolute copy numbers from next-generation sequencing data. *Bioinformatics*, 30(8):1056–1063, 2014.

[116] Layla Oesper, Ahmad Mahmoody, and Benjamin J Raphael. THetA: inferring intra-tumor heterogeneity from high-throughput DNA sequencing data. *Genome Biology*, 14(7):R80, 2013.

[117] Layla Oesper, Gryte Satas, and Benjamin J Raphael. Quantifying tumor heterogeneity in whole-genome and whole-exome sequencing data. *Bioinformatics*, 30 (24):3532–3540, 2014.

[118] Habil Zare, Junfeng Wang, Alex Hu, Kris Weber, Josh Smith, Debbie Nickerson, ChaoZhong Song, Daniela Witten, C Anthony Blau, and William Stafford Noble. Inferring clonal composition from multiple sections of a breast cancer. *PLoS Computational Biology*, 10(7):e1003703, 2014.

[119] Noemi Andor, Julie V Harness, Sabine Müller, Hans W Mewes, and Claudia Petritsch. EXPANDS: expanding ploidy and allele frequency on nested subpopulations. *Bioinformatics*, 30(1):50–60, 2014.

[120] Ruibin Xi, Angela G Hadjipanayis, Lovelace J Luquette, Tae-Min Kim, Eunjung Lee, Jianhua Zhang, Mark D Johnson, Donna M Muzny, David A Wheeler, Richard A Gibbs, Raju Kucherlapati, and Peter J Park. Copy number variation detection in whole-genome sequencing data using the Bayesian information criterion. *PNAS*, 108(46):E1128–E1136, 2011.

[121] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the *EM* algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–38, 1977.

[122] 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.

[123] Helga Thorvaldsdóttir, James T Robinson, and Jill P Mesirov. Integrative Genomics Viewer IGV: high-performance genomics data visualization and exploration. *Briefings in Bioinformatics*, 14(2):178–192, 2012.

[124] Charles Gawad, Winston Koh, and Stephen R Quake. Single-cell genome sequencing: current state of the science. *Nature Reviews Genetics*, 17(3):175–188, 2016.

[125] Jennifer A Doudna and Emmanuelle Charpentier. The new frontier of genome engineering with CRISPR-Cas9. *Science*, 346(6213):1258096, 2014.

Appendix A

Acronyms

Table A.1: Acronyms

Acronym	**Meaning**
2DGKC	Two-Dimensional Gaussian Kernel Convolution
AA	Association Analysis
ADT	Androgen Depletion Therapy
AR	Androgen Receptor
bp	Base pairs
CCGD	Candidate Cancer Gene Database
CCI	Common Co-occurring Insertions
CGH	Comparative Genomic Hybridization
ChIP	Chromatin Immunoprecipitation
CIS	Common Insertion Site
CNV	Copy Number Variant
CRPCa	Castration-Resistant Prostate Cancer
csCCI	cross scale Common Co-occurring Insertions
DNA	Deoxyribonucleic Acid
EM	Expectation-Maximization
FDR	False Discovery Rate
INDEL	Insertion/Deletion

Continued on next page...

Table A.1: Acronyms — continued from previous page

Acronym	**Meaning**
GATK	Genome Analysis Toolkit
GKC	Gaussian Kernel Convolution
GRCm	Genome Reference Consortium mouse build
HS	Histiocytic Sarcoma
LC	Lung Cancer
MLPA	Multiplex Ligation-dependent Probe Amplification
NGS	Next-Generation Sequencing
PCR	Polymerase Chain Reaction
RNA	Ribonucleic Acid
RTCGD	Retroviral Tagged Cancer Gene Database
SHEAR	Sample Heterogeneity Estimation and Assembly by Reference
SMRT	Single Molecule, Real-Time
SNP	Single Nucleotide Polymorphism
SB	*Sleeping Beauty*
SV	Structural Variant