Winter 12-15-2017

Knowledge Driven Approaches and Machine Learning Improve the Identification of Clinically Relevant Somatic Mutations in Cancer Genomics

Benjamin John Ainscough
Washington University in St. Louis

Recommended Citation

Ainscough, Benjamin John, "Knowledge Driven Approaches and Machine Learning Improve the Identification of Clinically Relevant Somatic Mutations in Cancer Genomics" (2017). *Arts & Sciences Electronic Theses and Dissertations*. 1190.
https://openscholarship.wustl.edu/art_sci_etds/1190

WASHINGTON UNIVERSITY IN ST. LOUIS
Division of Biology and Biomedical Sciences
Human and Statistical Genetics

Dissertation Examination Committee:
Obi L. Griffith, Chair
Elaine R. Mardis, Co-Chair
Ron Bose
Todd E. Druley
Timothy J. Ley
Christopher A. Maher

Knowledge Driven Approaches and Machine Learning Improve the Identification of Clinically Relevant Somatic Mutations in Cancer Genomics

by
Benjamin John Ainscough

A dissertation presented to
The Graduate School
of Washington University in
partial fulfillment of the
requirements for the degree
of Doctor of Philosophy

December 2017
St. Louis, Missouri

Table of Contents

List of Figures

List of Tables

Acknowledgments

Words cannot adequately express my feelings of gratitude to the many people who have supported me in my life leading up to and throughout my Ph.D.

I cannot thank Obi and Elaine enough for their mentorship and exposing me to the cutting edge of sequencing technology and precision medicine. With your help and guidance, I have grown immensely as an individual and a scientist and have accomplished far more than I thought I could when I started a Ph.D. To Elaine specifically, your guidance, insight, and support, have helped me to develop the scientific instincts, grit, and independence necessary to have the confidence to blaze new trails on my own. I appreciate the time you have sacrificed on my behalf and I consider it a great honor to be mentored by you and am grateful that you took a chance on me as a graduate student. To Obi, who has been in the trenches guiding me day to day, I am extremely grateful for your patience, thoughtfulness, and selflessness to listen to my ideas, work though problems together, and give me advice in every aspect of my career. Your encouragement and support have allowed me to discover and pursue new passions, like machine learning, which will be the focus of the next phase of my career. Your direct guidance and indirect example have had a profound impact on who I have become as a scientist and as an individual. I would be remiss if I failed to mention Malachi, who has acted as de facto mentor, and gave me countless hours of mentorship even though he was not officially a mentor or thesis committee member. Josh Swamidass was also so selfless is sharing his time advising me on machine learning development. I am grateful to him for helping me to see science in a new way and teaching me to never stop reaching for the stars. I have been fortunate to have so many great mentors through my Ph.D. and I am so grateful to each of you and for what you have helped me to become.

I would also like to thank my thesis committee, Tim, Todd, Chris, and Ron, who are each so bright, thoughtful, and talented. Your insights and advice over the years has been extremely helpful and enabled me to revise and apply my work in new ways. To Tim, who served as Chair, I am so grateful to you for the support and encouragement that pushed me through to the end.

To members of the Griffith Lab and McDonnell Genome Institute I am so grateful for your comradery, willingness to help one another, and friendship. Thank you for working with me, teaching me, giving me advice, and most of all the many hours that you patiently listened to me. Special thanks to Avi Ramu, Lee Trani, Zach Skidmore, Jasreet Hundal, Alex Wagner, Erica Barnell, Nick Spies, Aye Wollman, KK, Katie Campbel, Cody Ramirez, Adam Coffman, Josh McMichael, Jason Kunisaki, Arpad Danos, Bob Fulton, Jason Walker, Dave Larson, and Chris Miller. There are many more that I can’t name. Thank you to all of you, it has been a joy.

I am also grateful to DBBS and the HSG program for the support, programs, and training to succeed as a Ph.D. student. I would like to thank Siteman Cancer Center for generous fellowship support.

To my family, thank you for cheering me on throughout the whole Ph.D process. To my parents, words cannot express my gratitude for all that you have done for me. Thank you for your love, patience, and teaching me to stop and ask why. Your love for learning has been a great example in my life and allowed me to push my own boundaries. To my brother and sister, thank you for your awesome examples and love, I am lucky to have you. Thank you to my father and mother-in-law, it has been great to have a second set of parents that listen, encourage, and love me. To my brothers and sister-in-law, it has been fun to watch you grow. Thank you for your friendship and love. To all my in-laws, thank you for accepting me as one of your own, with all my flaws.

To Estee, thank you for loving me just for being your dad and giving me so much joy and happiness. Your energy and excitement are contagious and I am excited to watch you grow up. I am so grateful that you have been so good for your mom and patient with me as I have spent long hours away from you as I have finished my Ph.D.

Last, but by no mean the least, thank you to my dear wife Alexis. There is no one else that I would rather share the journey of life with. Thank you for supporting me and cheering me on at my best and my worst. Your sacrifices on my behalf have enabled me to complete my Ph.D. and there is no way that I could have done it without your support, advice, encouragement and love. Thank you for being such a caring wife and mother, I am excited to welcome 'baby brother' into our family. I strive to make you my top priority and make it my goal in life to support, honor, and protect you. I am so lucky to have you in my life and I cannot thank you enough.

Benjamin J. Ainscough

Washington University in St. Louis

December 2017

Dedicated to Alexis.

ABSTRACT OF THE DISSERTATION

Knowledge Driven Approaches and Machine Learning Improve the Identification of Clinically Relevant Somatic Mutations in Cancer Genomics

by

Benjamin John Ainscough

Doctor of Philosophy in Biology and Biomedical Sciences

Human and Statistical Genetics

Washington University in St. Louis, 2017

Professor Obi L. Griffith, Chair

Professor Elaine R. Mardis, Co-Chair

For cancer genomics to fully expand its utility from research discovery to clinical adoption, somatic variant detection pipelines must be optimized and standardized to ensure identification of clinically relevant mutations and to reduce laborious and error-prone post-processing steps. To address the need for improved catalogues of clinically and biologically important somatic mutations, we developed DoCM, a Database of Curated Mutations in Cancer (http://docm.info), as described in Chapter 2. DoCM is an open source, openly licensed resource to enable the cancer research community to aggregate, store and track biologically and clinically important cancer variants. DoCM is currently comprised of 1,364 variants in 132 genes across 122 cancer subtypes, based on the curation of 876 publications. To demonstrate the utility of this resource, the mutations in DoCM were used to identify variants of established significance in cancer that were missed by standard variant discovery pipelines (Chapter 3). Sequencing data from 1,833 cases across four TCGA projects were reanalyzed and 1,228 putative variants that were missed

in the original TCGA reports were identified. Validation sequencing data were produced from 93 of these cases to confirm the putative variant we detected with DoCM. Here, we demonstrated that at least one functionally important variant in DoCM was recovered in 41% of cases studied. A major bottleneck in the DoCM analysis in Chapter 3 was the filtering and manual review of somatic variants. Several steps in this post-processing phase of somatic variant calling have already been automated. However, false positive filtering and manual review of variant candidates remains as a major challenge, especially in high-throughput discovery projects or in clinical cancer diagnostics. In Chapter 4, an approach that systematized and standardized the post-processing of somatic variant calls using machine learning algorithms, trained on 41,000 manually reviewed variants from 20 cancer genome projects, is outlined. The approach accurately reproduced the manual review process on hold out test samples, and accurately predicted which variants would be confirmed by orthogonal validation sequencing data. When compared to traditional manual review, this approach increased identification of clinically actionable variants by 6.2%. These chapters outline studies that result in substantial improvements in the identification and interpretation of somatic variants, the use of which can standardize and streamline cancer genomics, enabling its use at high throughput as well as clinically.

Chapter 1: Introduction and Background

1.1 Introduction

Cancer is the second leading cause of death in the United States and is a devastating disease caused by genetic alterations that lead to uncontrolled cell growth and proliferation in various tissues and organs of the body. Although improvements in cancer detection and treatment have improved outcomes for many cancer types, this disease remains a significant killer. In their seminal paper, Hanahan and Weinberg described six traits or “hallmarks”, that make cancer cells unique from normal cells, including: uncontrolled growth (“self-sufficiency in growth signals”), lack of response to inhibitory signals (“insensitivity to antigrowth signals”), evasion of cell death (“evading apoptosis”), uncontrolled cell division (“limitless replicative potential”), stimulated development of vasculature (“sustained angiogenesis”), and spread to other tissues (“tissue invasion and metastasis”)[1]. In 2011, Hanahan and Weinberg updated their paper to include 4 additional hallmarks: (1) abnormal metabolic pathways, (2) immune system evasion, (3) genome instability, and (4) inflammation[2]. These hallmarks develop from alterations in the genome or epigenome that, in normal cells, carry the instructions for normal cell biology. These alterations can occur in the germline or in differentiated cells (somatic), and act to change normal cell biology to exhibit one or more hallmarks, leading to cancer onset and progression. One primary reason for updating the hallmarks was that new types of genomic and epigenomic alterations, along with many that were already known, were revealed by large-scale discovery genomics in cancers, facilitated by massively parallel sequencing (MPS) and computational approaches to discover different types of alterations.

Relevant types of alterations include: single nucleotide variants (SNVs), insertions and deletions (INDELs), and structural variants (SVs). SNVs are the most easily identified type of genomic variant and are single nucleotide substitutions (point mutations) that may or may not change the amino acid sequence when they occur in a protein coding gene. INDELs are small (1-50bp) insertions of novel sequence or deleted nucleotides and typically have deleterious effects on amino acid sequences when they occur in genes and affect the translational reading frame. SVs are a broad class of alterations, including inversions, translocations and copy number alterations (CNAs). CNAs are deleted or amplified regions of the genome wherein a segment of a chromosome is either deleted (one or both copies) or amplified (more than two copies). CNAs sometimes, but not always, change the expression level of the proteins encoded in the segment that is copy number altered. Epigenetic alterations, like changes in DNA methylation or chromatin packaging, are also important in cancer onset and progression, because they can result in changes in gene expression levels. For a more detailed description of how different alterations are detected in sequencing data see section 1.2 and of their importance in cancer genomics see section 1.3. The work described in this dissertation has primarily focused on improving the detection of SNVs and INDELs.

A critical challenge in the research and clinical management of cancer is accurately identifying the genetic alterations that cause the disease and have clinical value, such as therapeutic response or resistance, or are of prognostic or diagnostic significance. Somatic alterations are unique to the tumor and are not present in normal tissues. In practice, somatic alterations are typically identified by obtaining tissue samples of the tumor and a patient-matched, non-malignant tissue normal, isolating DNA from each and sequencing genomic regions of interest for each tissue isolate. Here, the regions of interest are defined by the study and can include a panel of known cancer genes, all coding gene sequences (the “exome”) or the entire genome. Current methods to explore cancer genomics utilize MPS instruments and a variety of preparatory methods that are specific to the type of study being performed. MPS instruments typically produce short read lengths of DNA sequence from *in situ* amplified library fragments. The resulting read data from tumor and normal libraries are aligned separately to the human genome reference sequence and the read coverage depth is evaluated. Statistical algorithms compare genomic alterations identified from the tumor data alignments to those identified from the normal data alignments to ‘call’ putative somatic alterations. The use of MPS has revolutionized the field of cancer genomics by facilitating the sequencing of entire cancer and normal genomes as opposed to the use of Sanger sequencing of PCR-targeted genes or loci.

However, there are challenges to the use of MPS methods in cancer research and diagnostics that lead to difficulties in properly interpreting MPS data from cancer samples. Some difficulties include inadequate amounts of tumor tissue available for MPS studies, inaccuracies in the identification of somatic alterations, and the time investment required to carefully interpret the results, especially in the setting of clinical diagnostics. In this dissertation work, these limitations were addressed by improving somatic analysis pipelines to (1) better identify literature-supported clinically and/or biologically important mutations and (2) become more standardized and automated through the removal of manual analyses. As a result, this work will improve the accuracy and efficiency of MPS data in clinical cancer diagnostics.

1.2 Overview of Massively Parallel Sequencing

The introduction of dideoxynucleotide chain termination DNA sequencing by Dr. Frederick Sanger in 1977, coupled with the concepts of molecular cloning from the 1980s, enabled development of the scientific discipline of genomics. These concepts permitted the decoding of several model organism genomes and culminated with sequencing the Human Reference Genome in 2004[3]. Shortly after the completion of the Human Reference Genome, the emergence of several MPS technologies brought about another revolution that eventually led to the use of MPS in clinical cancer diagnostics[4,5]. MPS methods dramatically reduced sequencing costs in comparison to Sanger sequencing by performing sequencing reactions in a ‘massively parallel’ way which, for most technologies, compromised sequencing read lengths in exchange for greater throughput[6]. While several MPS sequencing technologies are available in the market, Illumina’s technology has obtained dominant market share due to its superior accuracy, sequencing throughput, low cost, and versatility of applications (i.e. whole genome, exome, RNA sequencing, etc.)[7].

While MPS methods have made the production of DNA sequencing data cheaper and faster, they have made the computational analysis of sequence data more difficult and expensive. This is due to the greater quantities of data produced, to the complexities of data analysis and to the need to align short-read sequences as a first step to analysis. Correct read alignment of short sequence reads is compute-intensive, especially considering the size and repetitive nature of the human genome. While analysis strategies have, for the most part, kept pace with MPS technologies, cancer variant identification are still in need of method optimization[8].

1.2.1 Illumina Sequencing Technology and Sources of Sequencing Error

Overview of the Illumina sequencing technology

Illumina sequencing technology comprises three main steps: 1) DNA fragmentation, 2) library construction, and 3) sequencing (on the Illumina sequencing instrument). For example, to sequence the entire human genome, genomic DNA is isolated and amplified (if necessary). Then, the high molecular weight DNA is fragmented physically (via sonication or shearing) and size selected by agarose gel or magnetic bead sizing. Next, in the library construction step, end repair and A-tailing are followed by the ligation of platform-specific adapters to the size selected DNA fragments. The resulting whole genome sequencing (WGS) library can then be quantified (by qPCR or a fluorimeter) and diluted to Illumina's specifications to ensure the proper library concentration is loaded onto the sequencing instrument, for optimized data yield per instrument run[9].

Optionally, since only 1.5% of the human genome encodes proteins (the "exome), it is often more cost effective from a sequence data generation and analysis standpoint to sequence the exome (all coding genes) or a panel of selected genes or regions. This selection of specific regions from a whole genome library is accomplished by "hybridization capture" using synthetic probes that select for the genomic DNA targets based on shared sequence similarity. These probes are either DNA or RNA and contain derivatized biotin molecules so the resulting hybrid molecules composed of probe and complementary DNA fragment can be isolated from solution by complexing with streptavidin-coated magnetic beads. Applying a magnetic field to the hybrid capture magnetic bead mixture permits selective pull-down of the regions of interest. The resulting library fragments are amplified by a second PCR step and quantitated/diluted prior to Illumina sequencing.

Illumina MPS occurs in 3 steps: A) cluster generation, B) sequencing by synthesis (SBS), C) paired end sequencing (optional). Cluster generation occurs on the surface of the Illumina flow cell which has a lawn of covalently attached adapters across its surface, each of which is complementary to one of the Illumina specific adapters used in library construction. Due to complementarity, the library fragments hybridize to the flow cell surface and are then amplified on the flow cell surface using an isothermal bridge amplification process that is timed to produce a sufficient fragment density for each cluster. This density of fragments, in turn, permits the instrument’s optics to detect the subsequent SBS reactions. One end of the fragment is released by a chemical cleavage step that releases one end of each of the fragments in each cluster, and permits the detached strands to be washed away under denaturing conditions. Following this step, a sequencing primer is introduced and anneals to the single strands in each cluster, permitting SBS. Sequencing by synthesis is initiated with the addition of DNA polymerase and a mixture of custom nucleotides, each with unique fluorescent labels attached to identify the corresponding nucleotide (A, C, G, or T) and with a 3’-OH blocking group that prevents chain extension once it is incorporated into the synthesized strand. Following the nucleotide incorporation step, the clusters are read by determining the emission wavelength of each cluster after scanning the flow cell surfaces by a laser. Once the instrument’s optics scan both upper and lower flow cell surfaces to detect which nucleotide was added to each cluster, the fluorescent label is cleaved from the nucleotides added in that step, and the 3’-OH blocking group is also cleaved, permitting the polymerase to continue chain elongation by incorporating the subsequent nucleotide in the next synthesis step. This process is repeated for 100 to 300 nucleotides depending on the number of cycles specified (# of cycles=read length). With sequencing complete on one strand of each library fragment, paired end sequencing can be performed by first performing a second, shorter bridge amplification cycle, followed by cleavage of the second adapter from the flow cell strand, which releases the opposite end from the flow cell surface and makes it available for priming by a unique sequencing primer. The sequencing by synthesis then is repeated, as above. Paired end reads have an analytical advantage because read pairs can be matched computationally and then aligned to the human reference genome. Since the approximate distance between the read pairs is known due to the DNA size selection step prior to library construction, read pairs that do not map at the anticipated distance, or map with high quality onto different chromosomes can be interpreted as identifying a structural variant[10].

Sources of Sequencing Error

Sequencing error is defined as nucleotides misread by the sequencing instrument, an event that occurs on a per-base basis between 0.01% and 1%[11]. These errors are a result of a variety of factors including unavoidable inefficiencies in sequencing chemistry, technical errors due to the camera system, and interference from neighboring clusters or stray nucleotides. Chemical errors are a result of the reality that chemical reactions used to produce the reagents are never 100% efficient and these lead to accumulated noise as the step-wise sequencing reactions progress. Some examples of this type of error include “dephasing” errors which occur, for example, when a nucleotide without a proper 3’ -OH blocking group is incorporated, and permits a second nucleotide to be incorporated due to the absence of the blocking group. Similarly, if the 3’-OH blocking group is not properly cleaved, elongation by the polymerase is terminated for that cycle, and the affected fragment(s) will be out of sync with others in the cluster, contributing background noise to the correctly extended and detected fragments in the cluster. These errors are called ‘dephasing’ errors because they result in affected library fragments in the cluster to be out of phase with others. Examples of interference errors, or noise that interferes with the detection of true signals include: failure of enzymatic cleavage of the fluorescent label after detection, incorporation of multiple fluorescent labels in the same incorporation cycle, or stray fluorescent labeled nucleotides near a cluster (not removed during the wash cycle following incorporation). These errors, while small and inconsequential individually (as they typically affect one of many reads in a cluster), occur with sufficiently common frequency that eventually, at higher numbers of sequencing cycles, it is impossible to ascertain the signal of the in phase read fragments in a cluster from the compromised read fragments. As such, Illumina reads tend to be more error prone toward the end of the sequence read. Accumulated noise rates are what ultimately determines the maximum read length of an Illumina sequencing instrument. Downstream computational approaches model this error rate in their identification of sources of false positivity.

1.2.2 Computational Analysis of Massively Parallel Sequencing Data

Following production of short read sequencing data, the single- or paired-end reads must be aligned to the human reference genome as the first step in variant detection. Following alignment, duplicate reads, which have same start and end positions as aligned to the reference genome, are typically removed and variants are called.

Mapping Illumina Sequencing Reads to the Reference Genome

Aligning a ~100 base pair sequencing read to the 3 billion base pair human genome is a challenging computational problem. Substantial development in short read alignment software has provided researchers with many viable solutions[12-14]. Alignment algorithms are typically able to uniquely map only 70-80% of reads produced to the reference genome due to a high rate of homologous or repetitive sequences in the reference genome[15]. The number of uniquely mapped reads has increased as Illumina read lengths have increased. While there are dozens of alignment algorithms available, they are implemented with one of two main architectures: hash based designs (like Novoalign), or Burrows-Wheeler transform (BWT) aligners (like BWA[16], BWA-mem[17], or bowtie2[18]). Hash based aligners use indexes (hashes), which are short 'seeds' of the reference genome, to identify where portions of the read occur in the genome, and thereafter these alignments are presented to a local alignment algorithm, like Smith-Waterman[19] or Needleman Wunsch[20], where the maximum number of contiguous seeds from the reference match the read. The local alignment algorithm then produces a high quality alignment (see **Figure 1.1a**)[14]. Hash based aligners tend to be more accurate, but are more computationally intensive and memory inefficient than BWT aligners. BWT aligners concatenate the entire reference genome into one string, and use a BWT algorithm to store the reference genome in approximately 2GB of RAM. This approach permits the entire sequence read to be aligned to the reference genome in a computationally efficient way via a Burrows-Wheeler search followed by local alignment via Smith-Watterman[19]. BWT aligners produce an alignment that is extremely comparable to the alignment accuracy of hash based approaches, yet with substantially improved computational performance (see **Figure 1.1b**)[14].

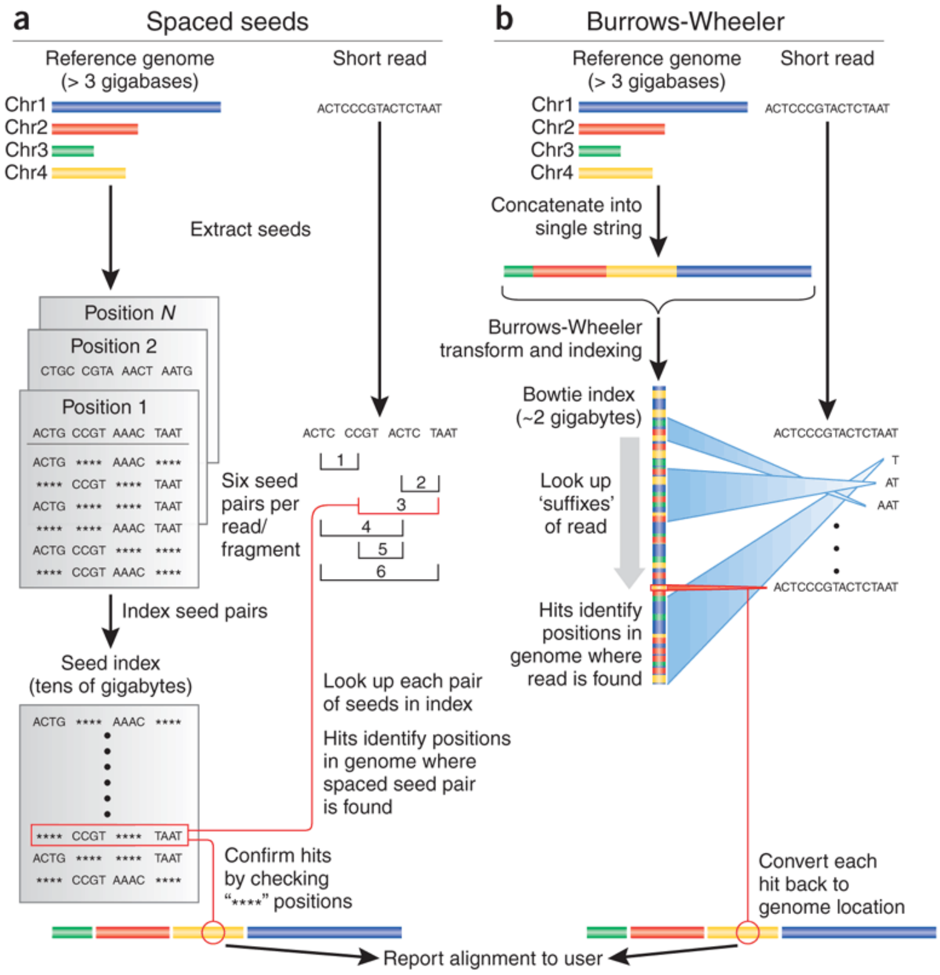


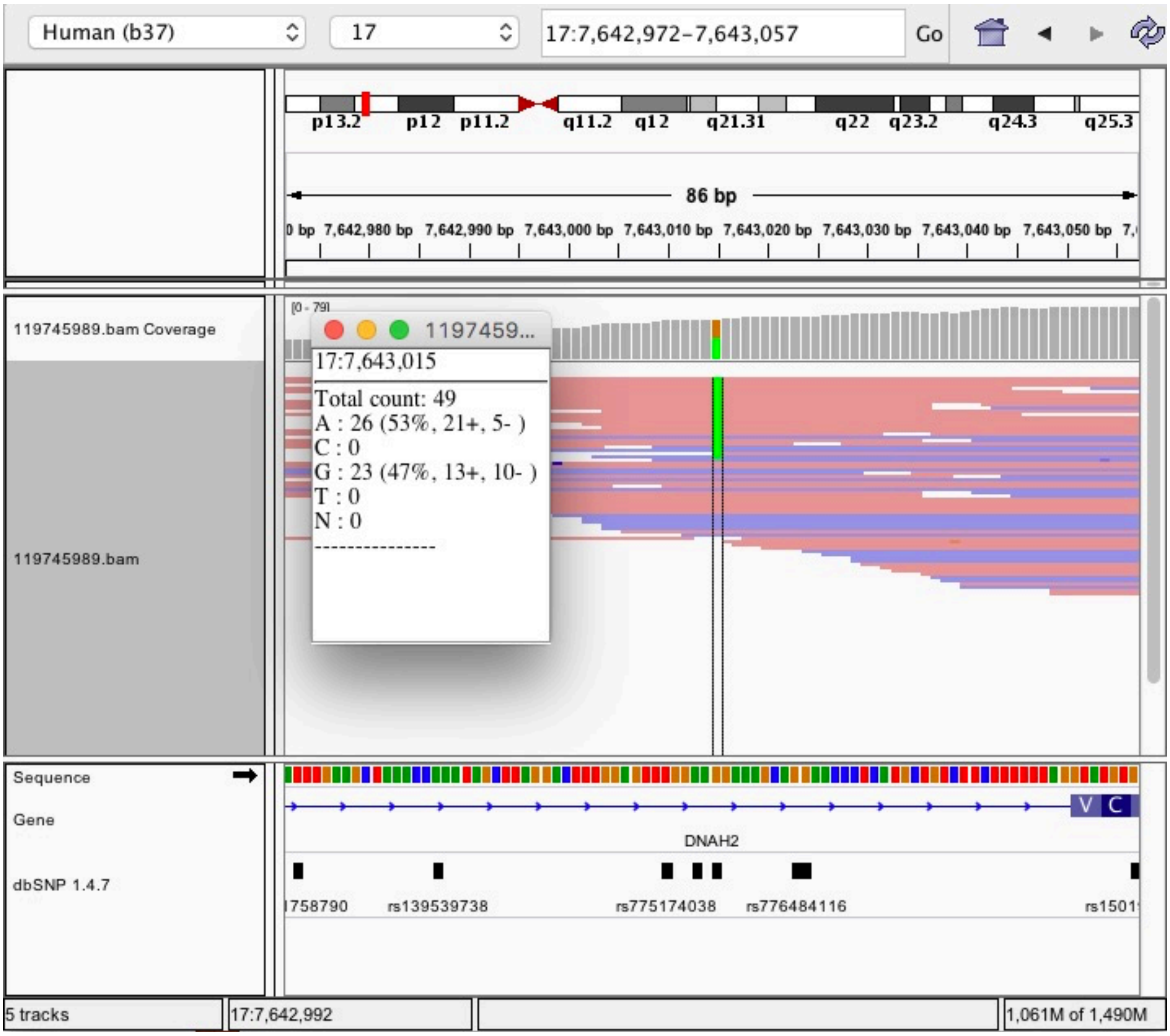
Figure 1.1 Comparison of hash based and BWT alignment algorithms. *Figure originally published in Trapnell et. al.*[14] *"(a) Algorithms based on spaced-seed indexing, such as Maq, index the reads as follows: each position in the reference is cut into equal-sized pieces, called 'seeds' and these seeds are paired and stored in a lookup table. Each read is also cut up according to this scheme, and pairs of seeds are used as keys to look up matching positions in the reference. Because seed indices can be very large, some algorithms (including Maq) index the reads in batches and treat substrings of the reference as*

queries. (b) Algorithms based on the Burrows-Wheeler transform, such as Bowtie, store a memory-efficient representation of the reference genome. Reads are aligned character by character from right to left against the transformed string. With each new character, the algorithm updates an interval (indicated by blue 'beams') in the transformed string. When all characters in the read have been processed, alignments are represented by any positions within the interval. Burrows-Wheeler–based algorithms can run substantially faster than spaced seed approaches, primarily owing to the memory efficiency of the Burrows-Wheeler search. Chr., chromosome." [14]

Germline Variant Calling Approaches

DNA sequencing variation is typically called with respect to the reference genome. Specifically, deviations from the reference sequence are identified and reported as variation. This is accomplished through different types of variant calling algorithms that use a statistical model to identify deviations from reference, taking into consideration the known error profile of sequencing and read alignment. In its simplest form, a variant calling algorithm identifies positions with one or more reads supporting a non-reference base. It then performs a statistical test to determine whether the number of reads supporting the putative variant could be expected by chance given the total number of reads, the error rate of the instrument, and other factors. In general, high-confidence variants will have a higher proportion of variant reads compared to the total coverage at the locus. This proportion (variant supporting reads over the total read coverage) is often referred to as variant allele fraction (or frequency VAF) and is one of several important metrics used by variant calling algorithms, filtering steps, and in the process of manual review (see **Figure 1.2 for an illustration of the VAF concept).** In the germline context, these variants should occur at 50% (heterozygous) or 100% (homozygous) VAF. In the somatic context, variants may occur at a variety of VAF percentages, as determined by numerous factors including their prevalence in the tumor cell population, and overall, by the percentage of tumor cells present in the piece of tumor used for DNA isolation prior to sequencing. There are variant callers that are specifically designed for a single type of genetic alteration (SNVs, INDELs, SVs) and others that are designed to call multiple types of variation. The following section outlines representative examples of callers for each variant type. (for more information on how somatic alterations are identified in tumor samples see section 1.4)

Many different SNV callers are available. Some widely used germline SNV callers include GATK[21], VarScan[22], FreeBayes[23], Platypus[24], and SAMTools[25] (some of these callers also are able to call somatic variants with matched tumor-normal aligned sequence data inputs; see subsection 1.4.2). Many SNV callers also identify INDELs including FreeBayes[23], VarScan[22], and Platypus[24], however, there are INDEL-specific callers that perform more advanced techniques like local re-alignment around INDELs to improve the accuracy of identification (GATK[21], SAMTools[25]). This type of specialized treatment is needed because INDELs are very difficult to uniquely identify and report. This is true because gapped alignments are generally more challenging to produce and because the same event can be represented in multiple ways by read alignments, especially as these events increase in length or occur around repetitive sequences.



17:7,643,015

Total count: 49
A : 26 (53%, 21+, 5-)
C : 0
G : 23 (47%, 13+, 10-)
T : 0
N : 0

Figure 1.2 Illustration of VAF in MPS sequencing data. *This screenshot of the integrative genomics viewer (IGV)*[26,27] *gives a visual representation of MPS data. The histogram near the top of the screen illustrates the sequencing coverage at each position. The reads are visualized in the middle of the screen with the red and blue reads signifying forward and reverse strands, respectively. The green in the reads illustrate those that contain a variant (A) base. At the variant position, the coverage histogram bars are colored for the variant (green='A') and reference (orange='G'). The pop up window shows the overall coverage, the count of each base and the VAF in parentheses (shown as a percentage). VAF is calculated by dividing the count of variant bases (26) by the total coverage (49).*

Since SVs are much more variable in their presentation (event type includes insertions, deletions, CNVs, inversions, and translocations, all over 50bp), there are a variety of techniques to identify them, as Alkan et. al. outline in their review[28]. These computational techniques include read-pair methods (VariationHunter[29-31], BreakDancer[32], SPANNER[33,34] etc.) that assess the span and read orientation of read pairs and perform a clustering step on discordant pairs to identify SVs. This approach is particularly powerful in detecting insertions, deletions, inversions and translocations, however, small insert sizes and reads that span the breakpoints are necessary for fine mapping of the SVs. Additionally, it is very difficult to detect CNVs using read pair relationships alone because they are primarily focused on the relationship between paired-end read alignment and not read coverage. Read-depth methods are much better suited to call CNVs (examples include Breakdancer[32], cn.MOPS[35], and Tigra-SV[36]) which model coverage to a Poisson distribution to identify regions that diverge from the distribution. These methods generally have poor breakpoint resolution and CNVs are generally hard to call in repetitive regions or from any capture-based sequencing strategy like exome sequencing because of the noise introduced by variable/biased hybrid probe capture efficiency[37]. Split-read methods (like Pindel[38]) attempt to identify the breakpoints of SVs by splitting a read and aligning each piece to the genome. While these approaches can finely map SV breakpoints, short read lengths often make it difficult to align the splits uniquely to the genome.

1.3 Cancer Genomics Overview

As mentioned in section 1.1, the prevailing strategy for identifying somatic mutations in cancer is to obtain matched tumor and normal tissue samples from the same individual, perform MPS-based data production, and computationally determine which mutations are unique to the tumor. This was the prevailing strategy used in the TCGA[39] and ICGC[40,41] projects, two examples of large-scale cancer genomics discovery projects that sought to survey the mutational landscape of human cancers. Somatic mutations can also be inferred when only tumor samples are available by filtering out presumed germline variation using large scale population databases like 1000 Genomes[42], ExAC[43], or gnomAD[43], although this approach is largely avoided in genomic discovery projects. Following identification of somatic mutations, functional impact is predicted via *in silico*, *in vitro*, or *in vivo* methods, each of which has highly variable cost and throughputs. Examples of functional prediction include (in order from least costly/time intensive to most): use of automated algorithms (SIFT[44,45], PolyPhen-2[46], MutationAssessor[47], CHASM[48], ParsSNP[49]), annotation of whether variants are present in other tumor sequencing datasets (COSMIC[50], TCGA, ICGC), and literature review of variants identified to annotate discovered etiology. More definitive answers can be pursued by experimental functional strategies using cell lines, xenografts, or mouse models. Computational procedures are often used to filter and focus downstream analyses to prioritize interpretation and experimentation. *In vitro* and *in vivo* functional evaluation is generally necessary for full confidence of a mutation's impact on cancer.

1.3.1 The Genomic Landscape of Human Cancer

Upon completion of the human genome project, there was keen interest in the scientific community to use this resource and establish sequencing methods to help uncover the mutational landscape of cancer. The general hypothesis was that the most frequently mutated genes and amino acid residues (or non-coding positions) were likely to be important in the onset and progression of the disease. The sequencing of the first complete cancer genome was reported by Ley et. al. in 2008[51]. In 2009 the U.S. National Cancer Institute launched the Cancer Genome Atlas (TCGA) project followed soon-after by the creation of the International Cancer Genome Consortium (ICGC). To date these consortia have sequenced tumors in 21 cancer types from over 20,000 individuals identifying about 63.5 million somatic mutations (http://dcc.icgc.org)[41,52]. Meta-analyses across many of these cancer types has revealed significant heterogeneity of mutation rates with hematological and pediatric tumors being the least mutated and tumors from tissue types that are highly exposed to mutagens, such as tobacco smoke or ultra violet radiation in lung cancers and melanomas, having the highest mutational burden (**Figure 1.3**)[53].

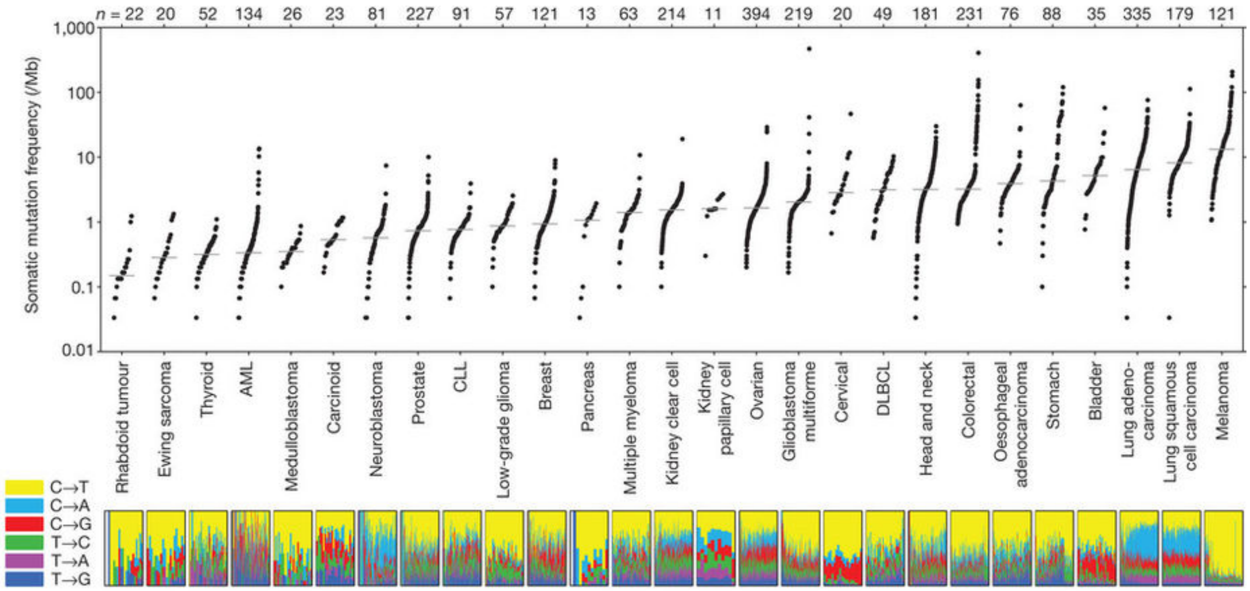


Figure 1.3 Mutational heterogeneity of cancer. *Figure originally published in Lawrence et. al.*[53]*. "Each dot corresponds to a tumor–normal pair, with vertical position indicating the total frequency of somatic mutations in the exome. Tumor types are ordered by their median somatic mutation frequency, with the lowest frequencies (left) found in hematological and pediatric tumors, and the highest (right) in tumors induced by carcinogens such as tobacco smoke and ultraviolet light. Mutation frequencies vary more than 1,000-fold between lowest and highest across different cancers and also within several tumor types. The bottom panel shows the relative proportions of the six different possible base-pair substitutions, as indicated in the legend on the left."* [53]

1.3.2 Challenges and Error Sources Caused by Cancer Tissues in Genomic Studies

Since cancers are localized in specific tissues, they are more challenging to obtain or to sample for genomic sequencing studies, in comparison to germline genetic diseases; where blood, hair, or saliva samples can be used. Tumor biopsies (or surgical resections) are typically obtained for a pathologist to evaluate if cells are cancerous by examining their anatomy under a microscope, often in conjunction with immunohistochemistry. These procedures typically require most of a limited tissue sample. Additionally, in pathology, it is very common for tumor samples to be formalin-fixed and paraffin-embedded (FFPE) to preserve the cellular anatomy. Unfortunately, this technique can lead to degradation of tumor DNA due to crosslinking of the backbone which introduces errors that must be accounted for in subsequent analyses[54]. Flash freezing of a portion of the cancer tissue after biopsy or resection in addition to traditional FFPE preservation can be specified by only when sufficient material is available.

Assuming adequate tumor DNA is obtained for sequencing, cancer biology poses several additional limitations that impede the accurate detection of somatic mutations in a tumor. One example is tumor cellularity; it is very rare for a tumor to be composed of 100% cancerous cells. Rather, normal cells are commonly interspersed with the cancer cells in a tissue sample such that nucleic acid isolation of the tissue will yield corresponding percentages of tumor and normal cell genomes. This proportionality can significantly limit the ability to detect somatic variants when there are more normal cells than tumor, in the worst case making it difficult to separate variants from the MPS error rate[55,56]. Additionally, cancer cells are heterogeneous and constantly evolving, meaning that all cells contain mutations in their genomes that were present in the founder population of cancer cells, yet may have added new mutations that are not present in all other tumor cells . This heterogeneity is manifest in the presence of subclonal tumor cell populations, each with unique mutations and potentially novel drivers[57,58]. Because of these and other factors, there is a great need for methodological developments that improve the sensitivity and accuracy of somatic variant discovery.

1.4 Somatic Variant Identification and Sources of Error

Accurately identifying the somatic SNVs and INDELs from matched tumor/normal sequencing data is an immense challenge because of the technical limitations of MPS technologies and the biological and sampling challenges that lead to reduced variant signal in tumor tissue samples as discussed above. Ideally, robust user-friendly tools would accept aligned bam files, for tumor and normal samples, and output a high-quality list of somatic variants. However, such a tool must also perform well on a wide variety of cancer types, each with distinct mutational burdens, varying degrees of tumor cellularity, and clonal architecture. Additionally, the ideal tool must adapt well to the research budget, i.e. perform as well on lower coverage levels (<50x) as it does on ultra-deep sequencing(<500x) or on various sequencing assay types including WGS, exome, and custom capture sequencing. These 'features' are desired to overcome the limitations of the MPS workflow and accurately detect variants. In practice, we see a variety of somatic variant callers used in the literature with no convergence to any specific caller because each has distinct strengths and weaknesses. Many studies have compared the performance of various variant callers and the only consensus is that there is very imperfect to poor concordance between callers[59,60], each exhibits variable performance at different coverage limits[61], and performance varies based on the aligner used[62]. A primary concern of researchers and clinicians analyzing and interpreting tumor sequencing data is that they do not miss important functional variants. Thus, they are often as concerned about false negatives as they are about false positives. A common analysis strategy is therefore to run multiple variant callers, combine their results via intersections or unions and filter out the false positives[61,63]. In the following section, representative variant calling error profiles observed in MPS data, somatic variant calling algorithms, false positive filtering strategies, and validation strategies will be introduced and explained.

1.4.1 Sources of Error in Variant Calling

Errors that lead to downstream false positive (and false negative) somatic variant calls can be caused by a variety of factors at each phase of the sequencing workflow, including sample preparation (e.g. low DNA yield, DNA degradation, sample swaps), library preparation (e.g.

PCR substitution errors, barcode errors), and sequencing and imaging errors (e.g. cluster crosstalk, cluster dephasing)[64]. In addition to processing-related errors, there are several sources of variant calling errors due to: low variant read data support, low quality variant calling due to noise, mapping errors to low complexity regions, and others.

Low Variant Read Data Support

Confidence that a putative variant signal is real is the result of observing sufficient numbers of variant- supporting reads, given coverage at the locus, that means it is highly unlikely that the observed variant is due to random noise, improper read alignment, and/or analysis-related errors. As mentioned above, this situation occurs when the tumor cell percentage of the isolated sample is low, or if the variant is present in a minor subclonal population of cells in the tumor mass. Both situations can be addressed by sequencing tumors at higher depths, and by the development of variant callers designed to detect variants at lower VAFs[65,66]. However, as coverage depth increases and variant signal decreases, per-base error rates of Illumina sequencing remain the same, which can diminish the power to confidently call such low VAF variants at extraordinarily high depths of coverage. As such, accurately separating variant signal from random sequencing error may require using complicated variant filtering combined with manual review to make sensitive calls with accuracy. Intricate sequencing strategies, like error corrected sequencing, exist that directly address this limitation by barcoding unique DNA fragments prior to library construction and sequencing[55,56]. The development of computational approaches to improve the detection of low VAF variants have also been beneficial as they can be incorporated more easily into existing cancer genomics analytical workflows (see Chapter 3 & 4).

Somatic Variant Interference

A variety of noise sources in the sequencing process can impact the detection of true variant signals including sequencing errors and sample contamination. While sequencing errors occur at a low rate, errors can accumulate at very high depth coverage and thereby are detected as true variants based on depth of variant-containing reads, as discussed above. Additionally, due to accumulated sources of noise in Illumina's technology, sequencing errors occur more commonly at the ends of sequencing reads. Similarly, there are reads that have a higher mismatch rate, due to low complexity repeats, and if many such reads pile up at a locus, a somatic variant could be called. Alternatively, sequencing errors can occur because of enzymatic substitution errors during PCR amplification steps, particularly in samples with low DNA input, which require more PCR amplification.

Another problem with low input samples is caused by PCR 'jackpotting', wherein smaller fragments are preferentially amplified and the overall rate of duplication is significantly elevated. Absent correction, jackpotting can result in erroneous variant calls but computational de-duplication, found in tools like SAMTools[25] or SAMBLASTER[67], can help alleviate PCR jackpotting by removing all but one read of the duplicated read set as defined by shared start and stop points in the genome alignment. However, reads with the same start and end location, are still observed occasionally in manual review of variant calls, even after computational de-duplication has been performed.

Sample contamination is another source of somatic variant caller interference. Contamination can occur in a variety of ways, but tends to predominate in preparatory steps, especially when performed manually instead of by robotic preparation. Here, a trace amount of contaminating DNA in a library can provide reads that invite variant calls in somatic pipelines. Additionally, the biology of hematological malignancies, as discussed in subsection 1.3.1, opens up the possibility for tumor contamination of the normal sample (circulating tumor cells found in the capillaries of the skin sample that often is used as normal tissue). This type of contamination also should be taken into account in somatic variant detection as this is a significant source of false negative calls.

Multiplexing of samples into sequencing lanes also is a source of sample contamination that can interfere with somatic variant calling. For example, as Illumina introduces higher throughput instruments, it has become common to add DNA barcode identifiers (indices) onto the adaptors used in library construction, and then to pool multiple samples into a single sequencing lane on the flow cell. Once the sequencing is completed, samples with the same barcodes can be grouped computationally and aligned to the reference genome. However, Illumina instruments that use the new ExAmp chemistry, found on the HiSeq X ten/3000/4000 and Novaseq platforms, suffer from index switching on up to 5-10% of sequencing reads on multiplexed samples[68]. This process occurs because of low levels of free primers that in conjunction with ExAmp reagents, results in spurious extension of the library with reads that have the wrong index, and is especially problematic when single or dual combinatorial indexing schemes are used[69]. Index switching at the level of even 1% per lane per flow cell with multiple samples can introduce interfering reads from common SNPs of unrelated individuals. Downstream, these errors manifest exactly like a low-level sample contamination issue originating from preparatory steps as described above, and can lead to putative somatic variants (low VAF) that are false-positives. Although such contamination can be identified computationally by performing extra filtering against databases of known SNPs like ExAC or gNOMAD[43], this filtering may miss rare variants. Alternatively, adding unique dual same-same indexes can completely correct for sample barcode switching[69].

Low Complexity Regions and Mapping Errors

As mentioned earlier, the repetitive nature of the human genome causes specific problems in proper mapping of short reads to the reference genome that impact somatic variant calling, specifically in low complexity and homologous regions. This is principally due to the fact that many such regions are larger than either the sequence read or the insert size of the library fragment. Approximately 48% of the human reference is composed of repeats (when considering all types) that are longer than a typical Illumina read (100bp)[15]. Li et al found that the error rate of raw calls of SNVs and INDELs is as high as 1 in 10-15kb, with most errors resulting from low complexity regions and the incomplete reference genome[70]. Similarly, highly homologous regions of the genome exist for many pseudogenes, gene families, or segmental duplication regions. As a result, a number of sequence reads fail to align or align ambiguously to multiple positions in the genome. General purpose alignment algorithms will randomly place reads that map equally to multiple loci, resulting in a putative variant arising from several mis-mapped reads at the wrong locus.

Other Sources of Variant Calling Errors

Some additional examples of signatures that have been associated with false positive variant calls are strand bias and short inserts. Strand bias describes a signature where the forward strands of the sequencing read produce a different result than the reverse strands, or vice-versa[71]. Short inserts occur when a DNA fragment with an insert size less that 2 times the length of a sequencing read is sequenced. This results in the forward and the reverse strand overlapping one another, and causes problems when there are discrepancies between the two read directions, by artificially inflating variant counts and VAFs due to false positive variant calls.

While this subsection outlines numerous variant calling error profiles, it is not comprehensive because new technologies and protocols introduce new sources of error. Although great care is taken in post-variant calling analysis to remove as many false positives as possible, without sacrificing false negatives, these sources of error persist.

1.4.2 Current Practices in Somatic Variant Calling

Somatic variant callers use at least two primary statistical methods to calculate a probabilistic estimate of a variant being somatic; 1) Bayesian classification (e.g. Strelka[65], MuTect[66], and SomaticSniper[72]) and 2) simple categorical statistics such as Fisher's Exact Test (e.g. Varscan2[73] or Shimmer[74]). This subsection will present the approaches by which these callers make their classification, their strengths and weaknesses, and opportunities to improve upon their methods.

All Bayesian variant callers are similar in their use of Bayesian mathematics to compute the probability that a candidate variant is somatic. This is accomplished by encoding known genomic features (such as the expected somatic mutation rate of a cancer, the error rate of sequencing instruments, or the expected VAF of germline variants) into probabilistic priors. These are then interpreted in conjunction with sequencing data features obtained from an aligned tumor and normal, such as the count of variant/reference reads (and other important metrics like base qualities) at each variant position. After candidate variant site probabilities are calculated, heuristic false positive filters are often used to increase sensitivity and specificity (**Table 1.1**).

In practice, different callers accomplished this approach differently. For example, MuTect uses a Bayesian Classifier to identify putative variants in tumor and normal samples independently, performs false positive filtering, and then utilizes another Bayesian Classifier to calculate which calls are somatic, germline, or 'variants' (insufficient evidence in the normal sample to call the variant somatic). By contrast, SomaticSniper and Strelka use a Bayesian Classifier followed by false positive filtering. Additionally, SomaticSniper assumes that variants will present as heterozygous (~50% VAF) or homozygous (~100% VAF), which is rare for somatic variants in tumor samples because of clonal diversity and sample purity. The strict assumptions of homo- and heterozygous VAF substantially limit Somatic Snipers ability to call low VAF somatic variants. Conversely, Strelka and MuTect make the simplifying assumption that the tumor VAF observed is equal to the tumor cell fraction of the sample (in other words, the sample has perfect representation of tumor cellularity and clonality and the VAF observed is equal to the proportion of cancer cells with those variants in the sample). While this is an over-simplification, both methods have more statistical power to detect lower VAF somatic variants because they correct for the heterogeneity or sample purity that does not generalize well to traditional heterozygous or homozygous models. All of these algorithms utilize prior knowledge about mutational load in cancer to prevent calling more somatic mutations than one would expect by chance. However, this can be problematic because of the vast differences in mutational load in different cancer types. While these callers each have parameters that can be modified to appropriately adjust for mutational burden, in practice these priors are not commonly updated in a disease specific manner, and often are not known in advance.

Fisher's Exact Test (FET) somatic variant callers use FET to statistically model if the observed counts of variant and reference reads are significantly different between the tumor and normal sample. In practice this is implemented quite differently in Varscan2 and Shimmer. Varscan2 uses heuristics to identify candidate variant sites, such as minimum coverage of 3x, PHRED base quality >20, and VAF >= 8%. Once candidate sites are identified in the tumor and normal sample independently, sites that occur in the tumor sample but not in the normal are fed into an FET for somatic evaluation. Sites that occur in the tumor and the normal sample are evaluated via FET designed for germline evaluation. The somatic FET compares the counts for reference and variant reads from the tumor and the normal samples and identifies tumor variants that are statistically significant by virtue of having more variant read support than does the normal sample. Those sites that are not statistically significant are evaluated by the germline FET for variant status. The germline FET compares the count of reference and variant reads from the normal sample to the null hypothesis of the expected error profile of the sequencing instrument for a given coverage level, i.e. with 1000x coverage and a 0.001 error rate, one would expect 1 read to be a sequencing error and 999 to be reference sequence. In Varscan2 this entire process is followed by false positive filtering. Shimmer is a much simpler workflow, performing an FET on any candidate variant that has more variant read support than an arbitrary threshold. The FET tests a null hypothesis that variant reads are randomly distributed between the tumor and normal sample. It also differs from Varscan2 through the utilization of multiple test correction to ensure that false positives are not called due to the substantial number of tests performed in somatic variant detection.

Both Bayesian classifier and FET based algorithms rely heavily on filtering to remove false positives and improve sensitivity and specificity (MuTect, Strelka, Somatic Sniper, and Varscan2). However, these filters are often identified empirically and set as heuristics. Different thresholds can wildly influence the performance of each caller. Additionally, there is no consensus, between the different methods, on which filters should be used. Of the 4 callers that utilize false positive filtering, 16 features were listed in the publications for these somatic callers. Of these, no features were used by all 4 callers, only 3 features were used by 3 callers, 5 features were used by two callers, and 8 features were used by only one caller (**Table 1.1**). As a result, there is significant room for improvement and standardization in the development of high quality, automated false positive filtering and prediction for somatic variants (see Chapter 4).

False Positive Filter	*Description*	*Callers*
Mismatches in variant reads	A high number of mismatches in a variant read can be indicative of a mapping error	Strelka, Somatic sniper, Varscan2
Variant repetitive regions	MPS technologies perform more poorly in low complexity regions, including repetitive and homopolymer runs, of the genome	Strelka, MuTect, Varscan2
Poor mapping quality	Mapping quality scores for variant reads are low	MuTect, Somatic sniper, Varscan2
Low base quality in variant read or read pileup	Low base qualities across the variant reads, signifying a sequencing error of the read, or across all the reads in the locus.	Strelka, Somatic sniper
Strand bias	All variant supporting reads are oriented in the same direction	MuTect, Varscan2
Average variant read position	Variants positioned near the end of the read are more likely to be artefactual	MuTect, Varscan2
Proximal gap to variant	INDEL in vicinity of variant read. These variants are likely alignment errors and are a result of the INDEL not an independent substitution.	MuTect, Somatic sniper
Minimum variant support	Number of variant reads above a defined threshold	Somatic sniper, Varscan2
High normal coverage	Unusually high coverage in the normal sample	Strelka
Short insert	Variant called in both overlapping paired-end variant reads	Strelka
Observed in normal	Variant support in the normal sample	MuTect
Tri-allelic site	Multiple variants in the same site	MuTect
Presence in panel of normals	Panel of germline mutations observed in normal samples.	MuTect
Minimum VAF	VAF above a defined threshold.	Varscan2
Distance to 3' end	Variant distance to 3' end of read fragment	Varscan2
Distance to clipped reads	Distance to soft-clipped reads	Varscan2

Table 1.1 False positive filtering strategies used in somatic variant callers. *Many somatic variant callers use heuristic false positive filters to increase accuracy. These values can be modified by the user at runtime to tune performance.*

1.4.3 Post-processing of Somatic Variant Calls

Because the different somatic callers have substantially different performance in different situations (e.g. MuTect is better at detecting low VAF mutations) it is common practice for somatic calling pipelines to run multiple callers and union (concatenate the calls from each caller) together to get a highly sensitive candidate somatic variant list. This list will typically then undergo extensive false positive filtering, manual review via integrated genomic viewer(IGV)[75], and validation sequencing[61]. By following this workflow, the entire variant calling pipeline undergoes project-, and sample-specific post-processing where the relevant disease biology and sample details are carefully considered when setting thresholds for various heuristics. Often due to budget constraints, extremely stringent filtering is used to winnow candidate variants down to a manageable number for the costly and time consuming manual review and validation steps.

Manual review typically occurs on the filtered candidate call list via visualization of the raw MPS sequence alignments through a genomic visualization tool like IGV[26]. Human reviewers can readily recognize patterns that are missed by variant calling and filtering. Manual reviewers look for error patterns like those described in **Table 1.1**, that were missed in the filtering of the somatic caller. Such errors occur when certain types of filtering are omitted (note the lack of overlap between filters in **Table 1.1**), when multiple of these error features occur in the same site, or when a variant does not surpass the arbitrary threshold for a given feature, but is nevertheless plainly obvious to the human eye. For standard procedures on how manual review is performed on somatic variants as well as illustrative examples of errors, see **Appendix 1**.

Validation on an orthogonal sequencing technology that (ideally) does not exhibit the same biases as that used for discovery sequencing, is ideal in somatic variant confirmation. The most relied on technology for validation sequencing remains Sanger sequencing, however, this is also the most costly and time consuming. Other MPS technologies, like IonTorrent, are also commonly used in somatic variant validation as they exhibit different error signatures than the Illumina platform. Remaining on the Illumina platform, but performing targeted custom capture sequencing to much greater depth is also a cost effective and trusted source for validation sequencing. Other methods to consider include digital droplet PCR, which has excellent sensitivity to detect variants. Validation sequencing is often limited by the availability of the tissue to perform the experiment and the cost.

1.5 The Interpretation Bottleneck in Cancer Genomics

The MPS workflow in cancer genomics has been streamlined and optimized from DNA isolation to somatic variant calling such that throughput is enormous. Despite advances in these areas, post processing and clinical or biological interpretations of somatic variant calls are becoming increasingly laborious, resulting in massive bottlenecks that impede the widespread adoption of precision medicine (see **Figure 1.4**)[76].

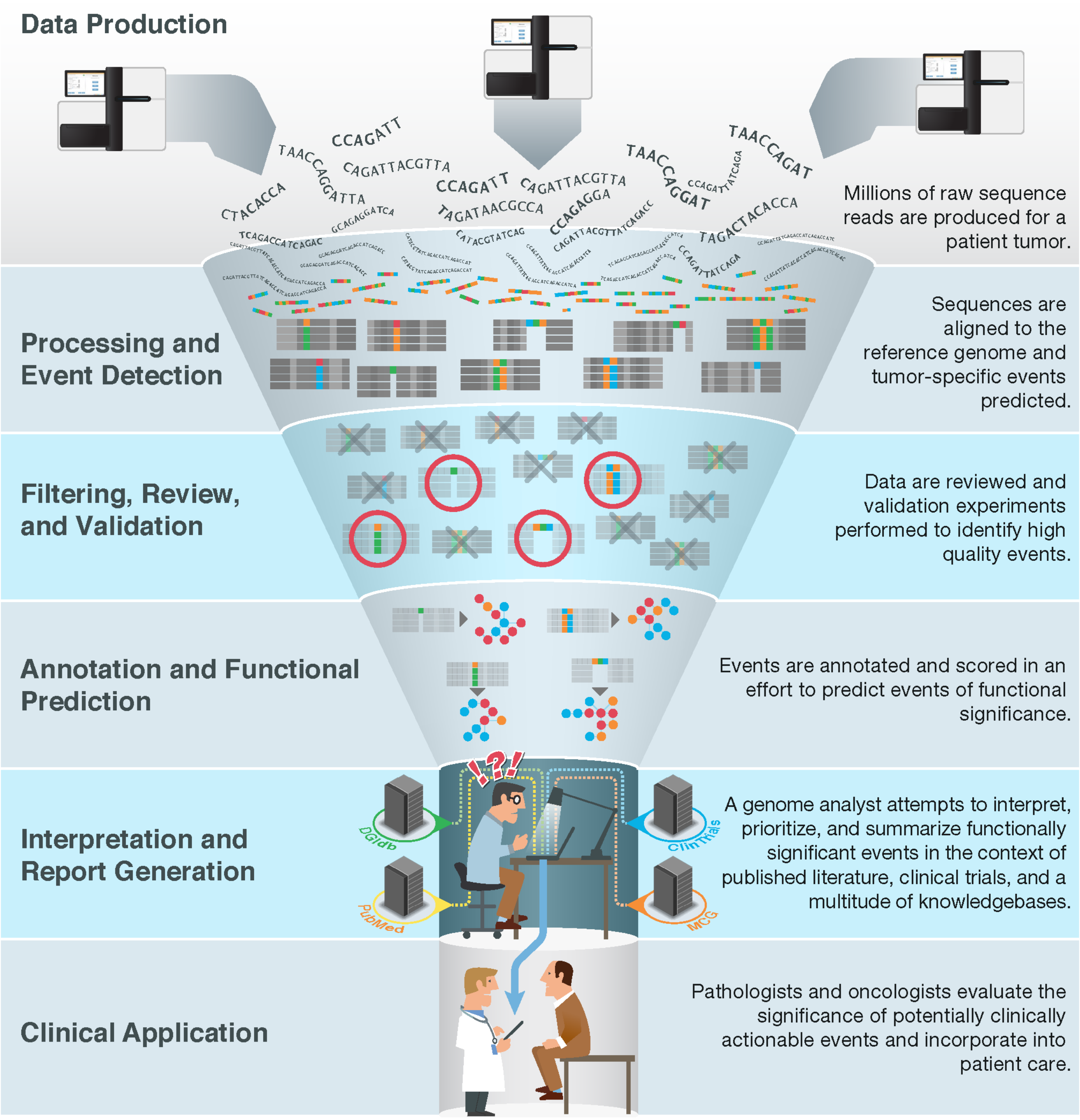

Data Production
CTACACCA
TAACCAGGATTA
CCAGATT
CAGATTACGTTA
CCAGATT
TAGATAACGCCA
CAGATTACGTTA
CCAGAGGA
TAACCAGGAT
TAACCAGAT
TAGACTACACCA
Millions of raw sequence reads are produced for a patient tumor.
Processing and Event Detection
Sequences are aligned to the reference genome and tumor-specific events predicted.
Filtering, Review, and Validation
Data are reviewed and validation experiments performed to identify high quality events.
Annotation and Functional Prediction
Events are annotated and scored in an effort to predict events of functional significance.
Interpretation and Report Generation
!?!
DGIdb
PubMed
ClinTrials
MCG
A genome analyst attempts to interpret, prioritize, and summarize functionally significant events in the context of published literature, clinical trials, and a multitude of knowledgebases.
Clinical Application
Pathologists and oncologists evaluate the significance of potentially clinically actionable events and incorporate into patient care.

Figure 1.4 The interpretation bottleneck of precision medicine. *Figure originally published be Good et. al. 2014*[76] *"A typical cancer genomics workflow, from sequence to report, is illustrated. The upstream, relatively automated steps (shown by their light color here) involve the production of millions of short sequence reads from a tumor sample; alignment to the reference genome and application of event detection algorithms; filtering, manual review and validation to identify high-quality events; and annotation of events and application of functional prediction algorithms. These steps culminate in the production of dozens to thousands of potential tumor-driving events that must be interpreted by a skilled analyst and synthesized in a report. Each event must be researched in the context of current literature (PubMed), drug-gene interaction databases (DGIdb), relevant clinical trials (ClinTrials) and known clinical actionability from sources such as My Cancer Genome (MCG)… [This] attempt to infer clinical actionability represents the most severe bottleneck of the process. The analyst must find their way*

through the dark by extensive manual curation before handing off a report for clinical evaluation and application by medical professionals." [76]

Analysts faced with large sets of variant calls are over-burdened with the need to perform validation or manual review of variant calls and then provide interpretations for those variant observations in the context of clinical and biological knowledge. The current standard procedure is to annotate variants that have been observed in cancer genomic studies and perform literature review to explain a variant's impact on cancer. Resources like TCGA[39], and ICGC[40,41], or databases like COSMIC[50] are very comprehensive and allow for easy annotation against the know mutational landscape of human cancers. However, resources that ascertain the biological or clinical importance of a variant in cancer are much more scarce. Examples include the cancer gene focused database PHIAL[54], or databases that outline clinically important variants and their impact like MyCancerGenome[77], or the Gene Drug Knowledge Database[78]. While these databases allow users to quickly browse their contents for identified variants, there are no programmatic application programming interface (APIs) available to allow this information to annotate variant calls automatically, resulting in a large manual analysis burden. Additionally, these resources only capture a small part of what is known in the literature. As a result, to find biological or clinical information about any given somatic variant, time-consuming literature review often is necessary. When thousands of somatic variants are called in a genomic study, it is not tractable to perform literature review on every single variant for studied functional consequences. Significant developments are necessary to develop resources that streamline the interpretation of somatic variants and improve their detection (see Chapter 2 & 3).

1.6 Machine Learning Applications to Variant Identification, Filtering, and Assessment

As discussed in section 1.4, false positive filtering is used extensively in somatic variant calling pipelines to produce high quality somatic variant lists. Most often these filters are set as heuristics with cut off thresholds defined by analysis of experimental data. This approach is limited by the experimental data used to discover thresholds. If the reference set for delineating variant calling errors was defined with a significantly different cancer type representation, sample heterogeneity, or analysis/sequencing strategy used, it may be necessary to manually alter filters for a new study. This increases analysis cost and complexity by requiring implementation of increasingly sophisticated hierarchical filters (e.g. if feature W is above X, then filter feature Y by Z). Traditional programming defines rules and procedures that are then used to process data (an example is the traditional filtering strategy used in somatic calling pipelines where specific thresholds need to be set for every variable to define heuristic filters). However, divergent from a traditional programing paradigm, machine learning algorithms allow a computer to automatically calculate procedures based on labeled training data. Specific to the filtering strategies described above, a machine learning algorithm utilizes knowledge of which variants are valid or artefactual (labels or classes) to learn structure in input variables (features, like VAF and base quality metrics) that will allow the algorithm to make future classification decisions on unseen data. These types of machine learning methods are often referred to as classification algorithms.

Machine learning algorithms have many advantages. They can learn extremely complicated nonlinear or hierarchical data structures and allow these structures to be applied to previously unseen data points. They can be used to automate time consuming tasks with a great deal of precision, saving a substantial amount of human analysis time. For example, the manual review of somatic variant calls required approximately 1 hour of analysis time for 70 variants reviewed. A machine learning algorithm could potentially reduce this analysis time to milliseconds (see Chapter 4).

While machine learning technology is incredibly promising, it is not without its pitfalls. The saying, 'garbage in, garbage out' aptly describes machine learning model performance on noisy, incomplete or biased training datasets. Great care must be taken in assembling, cleaning, and normalizing datasets in preparation for 'training' a machine learning algorithm. Additionally, it is important to provide a machine learning algorithm with appropriate and relevant features (variables) in the proper format for accurate decision making. For example, it should not be assumed that if a machine learning algorithm is given normal read counts, variant read counts, and sequencing depth that the algorithm will calculate VAF and utilize it in its decision making. The process of identifying and designing useful features is thus a critical step in the development of accurate machine learning algorithms for classification. Once training data are adequately assembled, experimental design steps, such as selection of cross validation strategy and independent (held out) test data, should be taken to ensure that the machine learning algorithm is not over-trained or overfit. This section will introduce standard practices for cleaning training data, engineering features for genomic datasets, evaluating model performance, avoiding overfitting, and using three common machine learning techniques (logistic regression, random forest, and deep learning).

1.6.1 Training Data Assembly, Cleaning, and Feature Engineering in Cancer Genomics

Assembly of high quality training data is often the most time intensive task in a machine learning project[79]. This is because data that are used for training of machine learning models are rarely collected and stored with machine learning in mind. Data typically are stored in various 'silos' and need to be aggregated and standardized for the purposes of machine learning. In cancer genomics, data are typically organized by case or project. A case is comprised of a single patient's DNA sequencing data collected at various times (e.g., at diagnosis, relapse, or after a treatment regimen) or tissue locations (i.e. primary tumor, normal, metastasis, etc.); a project is a collection of cases aggregated to answer a particular hypothesis. Independent cases and project data may exhibit a wide variety of details in terms of how mutations are discovered, formatted, and stored.

Bioinformatics as a discipline commonly deals with a large number of file formats and other caveats that must be considered when attempting to aggregate data across projects. For example, a common first task is ensuring that all genomic coordinates are in the same genome build. This typically needs to be done at the file level, such that every alignment file (BAM), variant call file (VCF), and/or manual review file (BED) file is standardized into the same context, reference genome build and variant coordinate system (i.e., zero-based or one-based). Once all individual data are standardized they can be aggregated. It is not uncommon for different projects to use the sequencing data from the same cases (e.g., if two projects focused on the same tumor type analyzed by the same group include samples sequenced redundantly in a prior project). However, if this is the case, care must be taken to eliminate duplicated data that would otherwise confound the machine learning algorithm (and over-estimate performance) downstream.

Sparsity, or missing data, can also be a problem for machine learning algorithms. In cancer genomics data aggregation, this can be a result of different tissue samples being sequenced between studies (tumor/normal, tumor/normal/relapse, tumor/normal/metastasis, tumor only, etc.), or mutational status (it is unlikely that many different samples will share a high number of mutations at the same loci). These missing data can be accounted for by making simplifying assumptions like treating tumor-normal pairs from different individuals and different tissues in the same individual as independent samples, imputing missing genotypes from population data, or aggregating metrics irrespective of genomic location to compare all variants against each other.

An essential step in machine learning data aggregation is carefully examining the process seeking to be automated. In the case of somatic variant post-processing, this would require extracting and manipulating raw data into numerical values that are most likely to correlate with class labels. This process is referred to as feature engineering and requires a thorough understanding of the problem domain space. Once all data across projects are aggregated and normalized, raw sequencing information can be processed for machine learning. Because bam files are so large, and only a small percentage of the sequence information relates to individual somatic variants, it may be more efficient to extract only variant-specific metrics from the bam file for training. This can be done using a number of tools including SAMtools[25] and bam-readcount (https://github.com/genome/bam-readcount). Once all relevant data are extracted they can be aggregated with other features such as cancer type. These features can then be manipulated to increase their value to the downstream classifier. For example, the tools listed above only provide raw counts for variant and reference bases at a specified position; it is necessary to use the variant read count and locus coverage to calculate the VAF, a highly valuable and useful metric commonly used by manual reviewers in decision making. Other examples of feature engineering include calculating the difference in VAF between the tumor and the normal sample. Some machine learning methods are more capable of performing feature engineering as a part of their methodology than others (deep learning approaches can perform very well with little feature engineering beforehand), however, performance can be substantially improved by engineering the feature to focus machine learning algorithms on metrics thought (or known) to be important in classification. While feature engineering is an important step to achieve better performance, it is also an opportunity for human bias to be introduced into the algorithm and so great thought and care should be taken in regards to what features are included in the model and why they might be important.

1.6.2 Model Performance Assessment and Overfitting

Machine learning methods make predictions by creating a mathematical representation (or equation commonly referred to as a model) representing the problem at hand. The different methods used (e.g., logistic regression, random forest, deep learning, etc) define the mathematical structure of the model, although initially the parameters of that model are often randomly selected. This model, with random parameters, is then assessed for performance using a predefined error function that can use a variety of metrics for evaluation: examples include accuracy, F_1 score, or receiver operator characteristic (ROC) area under the curve (AUC). Once baseline performance has been assessed for random parameters, these parameters can be iteratively changed until the method converges on a minimum error. This approach is known as gradient descent. Each machine learning algorithm performs this process slightly differently but the same general architecture of slowly converging on an optimal solution is followed.

The metrics used to assess model performance can be used in training to converge on the ideal solution, given the data, and/or to evaluate and compare how different machine learning algorithms perform. The metric used to assess importance can therefore have a significant impact on model performance because it is the mathematical constraint used to measure model success and drive convergence. As such, each metric should be thoroughly understood. Accuracy is the most straightforward measure of model performance and can be defined as the number of correct predictions made by the classifier compared to the total number of examples evaluated. The F_1 score is the harmonic average of precision and recall, or in other words, it considers not only the false positive rate, as does accuracy, but also the false negative rate. This is due to the accuracy paradox, where accuracy can increase even though an algorithm's predictive value does not change (or even decreases)[80]. The F_1 score accounts for this limitation. ROC AUC builds upon each of these metrics by evaluating how the precision and recall behaves at all decision boundaries. As the threshold is changed for what is considered a true positive, the classifier's false positive rate and true positive rate are assessed. In contrast to accuracy and F_1 score, which only evaluates model performance at one decision boundary. ROC AUC gives a comprehensive representation of model performance across all decision boundaries.

Preventing overfitting

Overfitting occurs when an algorithm too closely learns the specific structure of a training dataset, or "memorizes" the training dataset, and does not generalize well to other datasets (see **Figure 1.5**[81]). There are two commonly used techniques to protect against overfitting: (1) cross-validation in training; and (2) testing model performance against independent test datasets.

In a typical cross-validation scheme, a training sample is partitioned into *k* equally sized subsamples. These samples are used as a testing set for a model that was trained on each *k*-1 superset. In this way, each sample is used in a testing set only once, and evaluation of each trained model only occurs on samples not used to train that model, minimizing the opportunity for overfitting. Stratified sampling, ensuring that equal distributions of labels are maintained between the whole dataset and the subsamples, is important to achieve consistent performance on datasets with an unbalanced distribution of class labels. While test sets are commonly used in cross-validation schemes, additional confidence in model performance is obtained when results replicate to completely independent datasets. Ideally, these would be additional datasets specifically collected to test the model. However, in practice it is often too costly to collect a dataset solely for testing. Therefore, the data set is sometimes randomly segmented before training and a testing dataset is held out and never utilized in training or cross-validation phases.

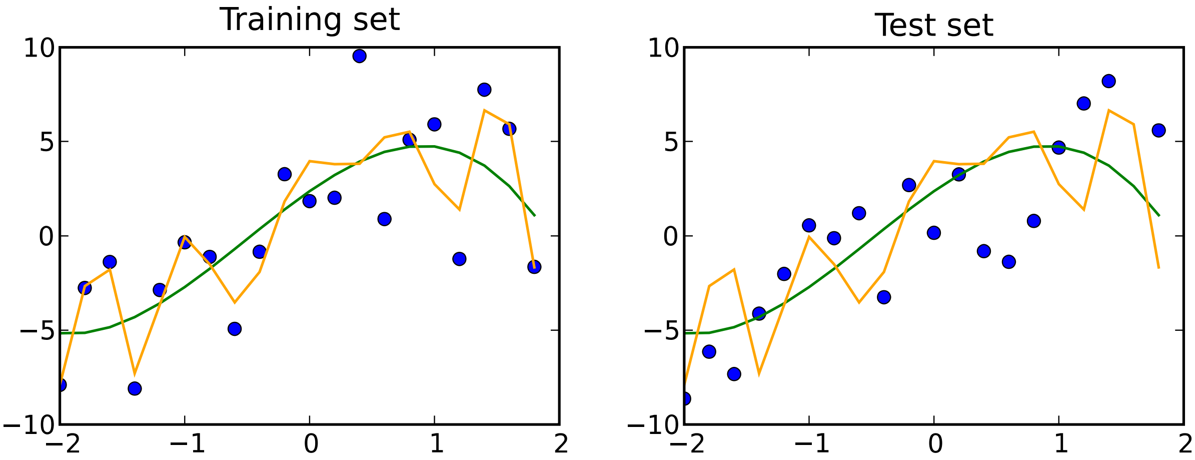


Figure 1.5 Visualization of over-fit and well-fit models on training and test datasets. *The left plot illustrates example data for a training dataset and the right plot for the test dataset. The orange line represents a model that is overfit to the training data and is more poorly fit to the test data. The green line represents the population structure that the accurattley capture the population trend seen in both traning and test sets. By Skbkekas (Own work) [CC BY 3.0 (http://creativecommons.org/licenses/by/3.0)], via Wikimedia Commons*[81].

1.6.3 Supervised Learning to Perform Classification

Logistic Regression

Logistic regression classifiers in their simplest form estimate numerical features to a binary classification (true or false) by fitting experimental data to a sigmoid function[82]. This is illustrated in **Figure 1.6**, a hypothetical example where a sigmoid function is fitted to data that show how hours of studying relate to passing an exam[83]. The sigmoid function estimates the probability of passing an exam given the hours that an individual has studied.

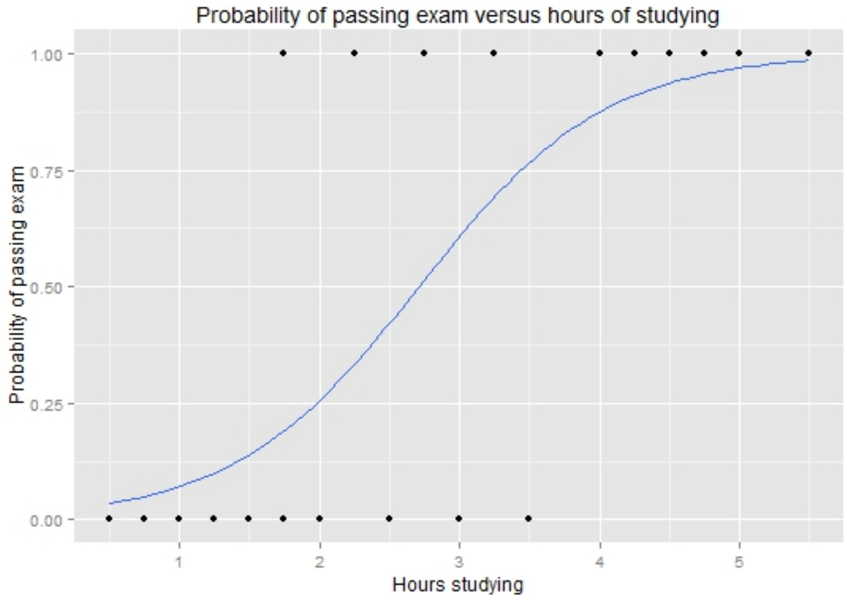


Figure 1.6 Illustration of a sigmoid function fitted to a logistic regression classification estimating the probability that an individual will pass an exam given hours spent studying. *The blue line illustrates the sigmoid function fit to this data. The points at 0 or 1 indicate whether an individual passed or failed an exam after the specified number of hours studying. By Michaelg2015 (Own work) [CC BY-SA 4.0 (https://creativecommons.org/licenses/by-sa/4.0)], via Wikimedia Commons*[83]

Logistic regression classifiers can be extended beyond binary classification tasks via a process called one-versus-all classification. This is a classification scheme where each class is evaluated against all others, or a classifier is produced for each unique class. For example, if we had a dataset with classes A, B, and C, one-versus-all classification would produce 3 binary classifiers. The first would treat the A class as truth (1) and B and C classes as false (0), then, the B class would be true and A/C would be false, etc. Logistic regression can also be applied to multivariate datasets, data with many features, as opposed to only one feature as shown in **Figure 1.6**. For each feature a weight parameter will be learned by the model that, in conjunction with all the other parameters, allow the estimator to produce accurate predictions given a dataset.

Logistic regression models are useful because they are efficient to train and are easily interpretable. The coefficients learned by the model for each feature are the odds ratio, which can be used to identify the impact of each feature in classification[84]. Specifically, the further an individual coefficient is from zero the more important it is in helping the classifier discern between classes. Logistic regression models are able to fit simple non-linear features, however, as training data gets more complex (i.e. the more features that are non-linear) it becomes more difficult for a logistic regression model to fit the data.

Random Forest

A random forest is an ensemble approach that uses decision trees to learn the structure of data and random bootstrap aggregating to correct for overfitting [85-88]. A decision tree is an algorithm that uses thresholds for every feature to make classifications with a high degree of specificity (see **Figure 1.7**[89]).

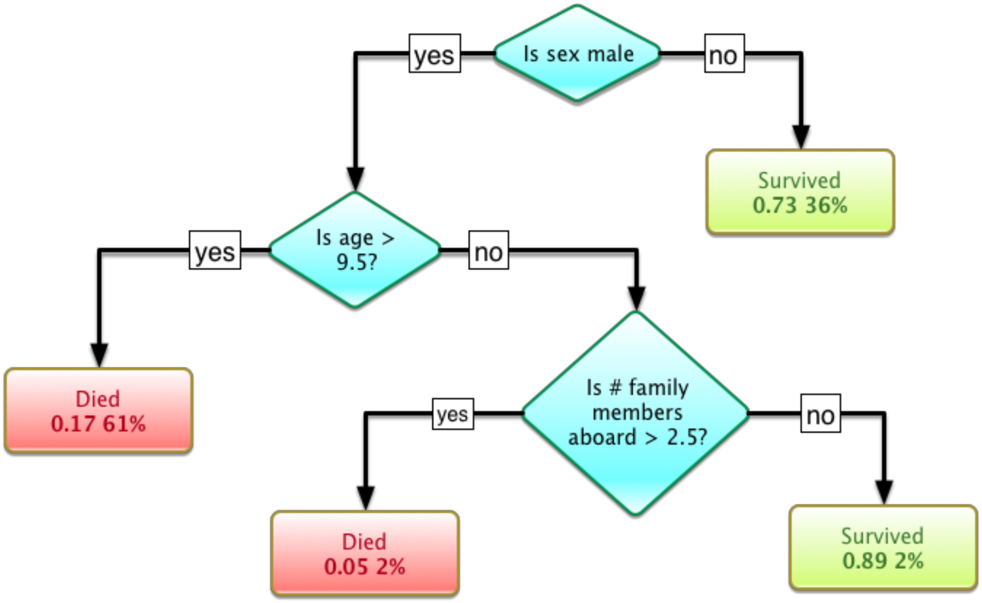


Figure 1.7 Example decision tree of survival on the Titanic. *Each of the features (blue) lead to more specific feature filtering or survival status (green or red).*[89]

Decision trees can be extremely precise in their learning of data and quick to train, however they are prone to overfitting especially on deeper trees[90]. Random forest corrects this overfitting problem by taking the averages of trends learned across many trees. This process sacrifices the innate intuitiveness of decision trees, but substantially improves the performance on unseen data.

Random bootstrap aggregating has 3 steps. 1) taking random samples of the training data with replacement (bootstrapping) to produce many trees (a forest). 2) Randomly selecting a subset of features that will be used to compute the decision boundary for each branch. 3) Assessing performance of the forest by feeding labeled examples through each tree in the forest and letting each tree vote for the classification status of the sample. This process is more opaque than how decision trees make decisions but is still able to calculate feature importance by tallying the 'votes' of how often a feature is used in classification. Random forests are powerful because they can learn mixed numerical and categorical features, can learn complicated non-linear relationships, and are straightforward to work with. A drawbacks of random forests is that they take the same amount of time to make predictions as they do to train (input variables must traverse the entire forest regardless), so they can have comparatively slower prediction times than other algorithms.

Deep Learning

In recent years, the term deep learning has become synonymous with neural networks, a subspecialty of machine learning that has existed for decades[91]. Neural networks have gone in and out of favor many times in the decades since their invention in 1957 and have gone by many names: artificial neural networks, multilayer perceptron, feed forward neural network, deep learning etc. While there are semantic differences referred to with each term in the field, colloquially many of these terms are used interchangeably to describe neural computing.

A neural network is a mathematical representation of how we understand the brain to process information. The brain's neurons take nerve stimuli, from our body (skin, eyes, ears, etc.), as inputs to a complex network of neurons. These neurons process these inputs biochemically and send results to other neurons in an extremely complicated network. While artificial neural networks are mathematical abstractions of this biological reality, they are not nearly as complicated in their network architecture as a brain. In their simplest form, computational neural networks (NN) are comprised of an input layer, a hidden layer, and an output layer (**Figure 1.8**).

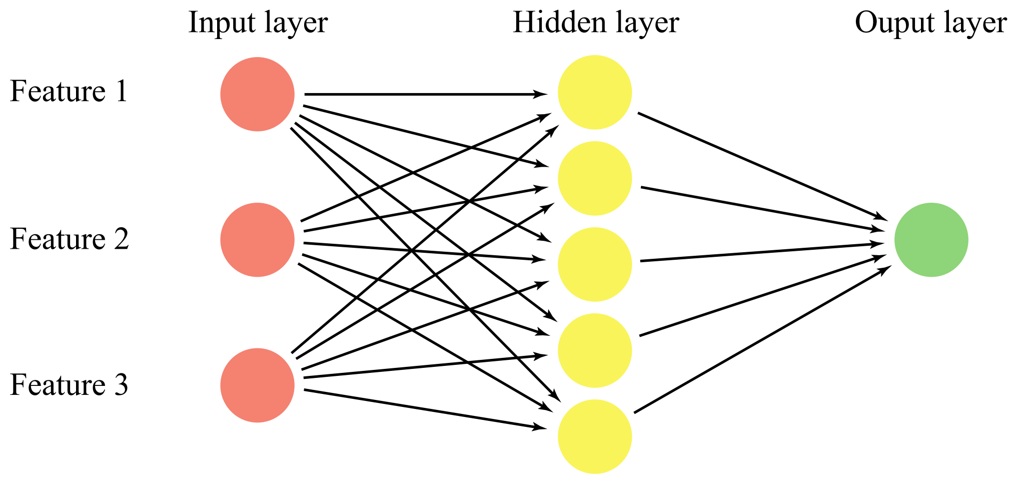


Figure 1.8 Simple illustration of a feed forward neural network.

Each feature feeds into every hidden layer along with a randomly initialized weight variable. These values are fed through the hidden layer into the output layer. Hence the name feed forward neural network. What happens at each node is similar mathematically to the logistic regression algorithm introduced above. The feature values are fed into an activation function, like a sigmoid function, and they output a value between 0 and 1 that is passed into the output layer or another hidden layer. A variety of activation functions are common in neural networks including sigmoid, hyperbolic tangent (tanh), and rectified linear unit (relu). Similar to logistic regression, these networks are trained using some flavor of gradient descent to converge on a minimum error. The main difference in NN is that weights need to be updated for every node of every layer as opposed to updating one function in logistic regression. For this reason, training neural networks can be extremely costly. Recent improvements in computing speed and the discovery of techniques like backpropagation have led to a resurgence of NN in recent years because they are now utilized on enormous data sizes. Backpropagation, or backward propagation of errors, is a mathematical technique that allows the errors to be fed back through the network, similar to how features are fed into the network, and to modify the weights by gradient descent at each and every node with extreme efficiency[92]. These advances have also made training deeper networks, NN with more hidden layers, and other networks with complex architectures (convolutional NN and recurrent NN) more common (these deeper and complex architectures are commonly called deep learning). This has ushered in a renaissance of the field as it has allowed computer intelligence to exceed human performance on tasks like image recognition or even extremely complex strategy games such as GO[93].

While deep learning has led to astounding advances, it has some important weaknesses. First, it typically takes an extremely large training dataset size for deep learning to outperform approaches like random forests. This, coupled with the fact that deep learning is extremely computationally expensive to train can be a deterrent to using this method. Perhaps deep learning's biggest weakness is that it acts like a black box. In other words, it is extremely difficult to determine how and why a deep learning model made the decisions that it did. Despite these drawbacks, the ability to learn complicated structure makes deep learning an attractive method to implement for many machine learning tasks. (see Chapter 4)

1.7 Addressing the Somatic Variant Identification and Interpretation Bottleneck

As described in section 1.5, while substantial progress has been made in the utilization of MPS data for cancer genomics, substantial bottlenecks are still present that contribute to the difficulty of making cancer genomic information useful clinically (see **Figure 1.4**). Open source, openly licensed and computationally friendly resources are needed to widen the somatic variant detection bottleneck[76]. In Chapter 2, DoCM a database of curated mutation in cancers is described as a solution to enable crowd sourced curation of variants from the literature that are clinically and biologically important to cancer. In Chapter 3, the utility of DoCM is demonstrated by rescuing clinically and biologically important somatic variants from four TCGA project datasets through targeted manual review and validation sequencing. In Chapter 4, a machine learning classifier, trained on 41,000 somatic variants from 20 cancer genomics projects, is presented that automates manual review and false positive filtering. This work will widen the somatic variant identification bottleneck and streamline and standardize analysis

Chapter 2: DoCM: a Database of Curated Mutations in Cancer

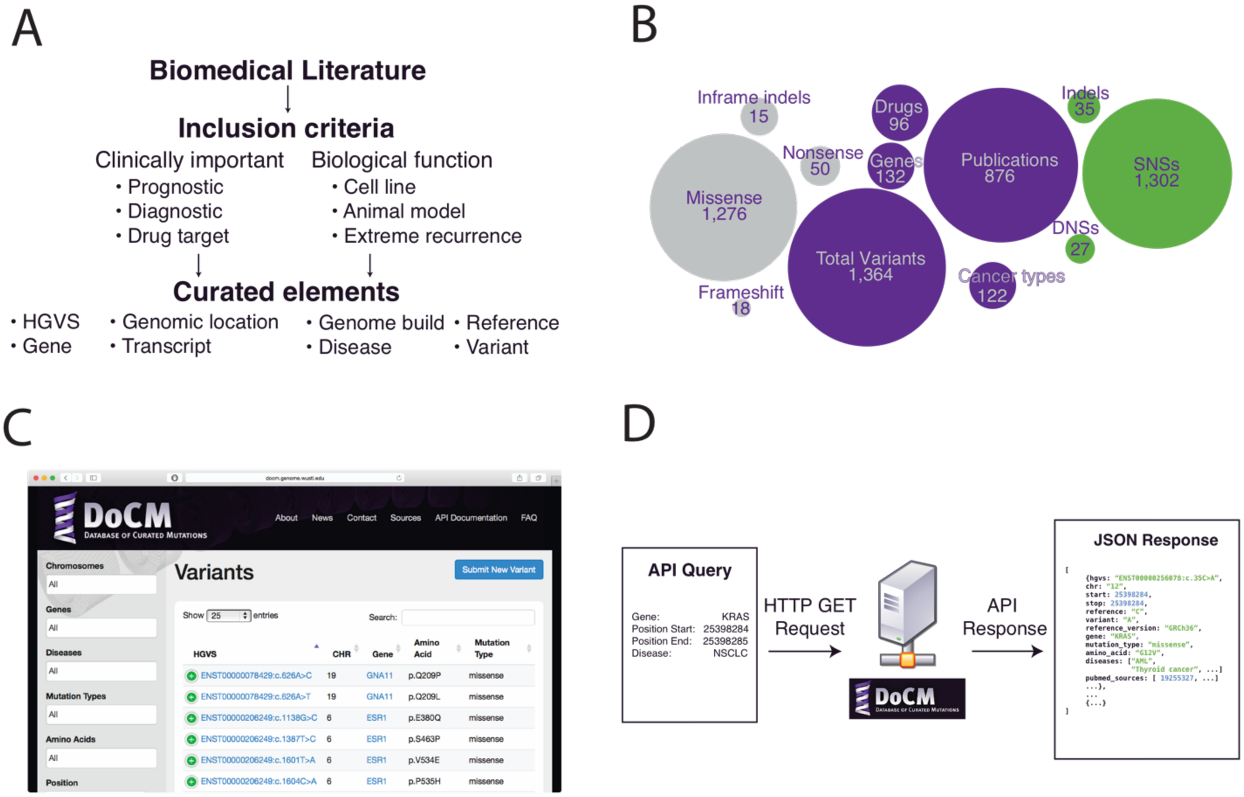
Ainscough, B. J. et al. DoCM: a database of curated mutations in cancer. Nat Methods 13, 806-807, doi:10.1038/nmeth.4000 (2016)

2.1 Introduction

Large-scale cancer genomics discovery projects such as The Cancer Genome Atlas (TCGA) and the International Cancer Genome Consortium (ICGC) have systematically characterized the molecular lesions in human cancer genomes, thereby laying the foundation for precision cancer medicine. However, a curated set of somatic variants with established relevance to cancer biology is essential for clinical annotation and for use in computational data analysis. A variety of somatic cancer variant databases exist that help identify important variants, including gene-level[54], variant-level[41,50], and clinically-focused variant interpretation databases[77,78,94]. These resources have greatly increased our understanding of the landscape of clinically and biologically relevant cancer variants, and when used in aggregate provide an understanding of the relevance of specific variants. We have created a database of curated mutations in cancer (DoCM, http://docm.info), an open-source, openly licensed resource to enable the cancer research community to aggregate, store, and track biologically important cancer variants with provenance supported by the literature.

2.2 Results

DoCM is a curated repository that facilitates the aggregation of gene and variant information for variants with prognostic, diagnostic, predictive, or functional roles from these resources as well as from individually curated publications (**Figure 2.1** and **Table 2.1**).



Chromosomes: All
Genes: All
Diseases: All
Mutation Types: All
Amino Acids: All

Show 25 entries

HGVS	CHR	Gene	Amino Acid	Mutation Type
ENST00000078429:c.626A>C	19	GNA11	p.Q209P	missense
ENST00000078429:c.626A>T	19	GNA11	p.Q209L	missense
ENST00000206249:c.1138G>C	6	ESR1	p.E380Q	missense
ENST00000206249:c.1387T>C	6	ESR1	p.S463P	missense
ENST00000206249:c.1601T>A	6	ESR1	p.V534E	missense
ENST00000206249:c.1604C>A	6	ESR1	p.P535H	missense

API Query

Gene:	KRAS
Position Start:	25398284
Position End:	25398285
Disease:	NSCLC

JSON Response

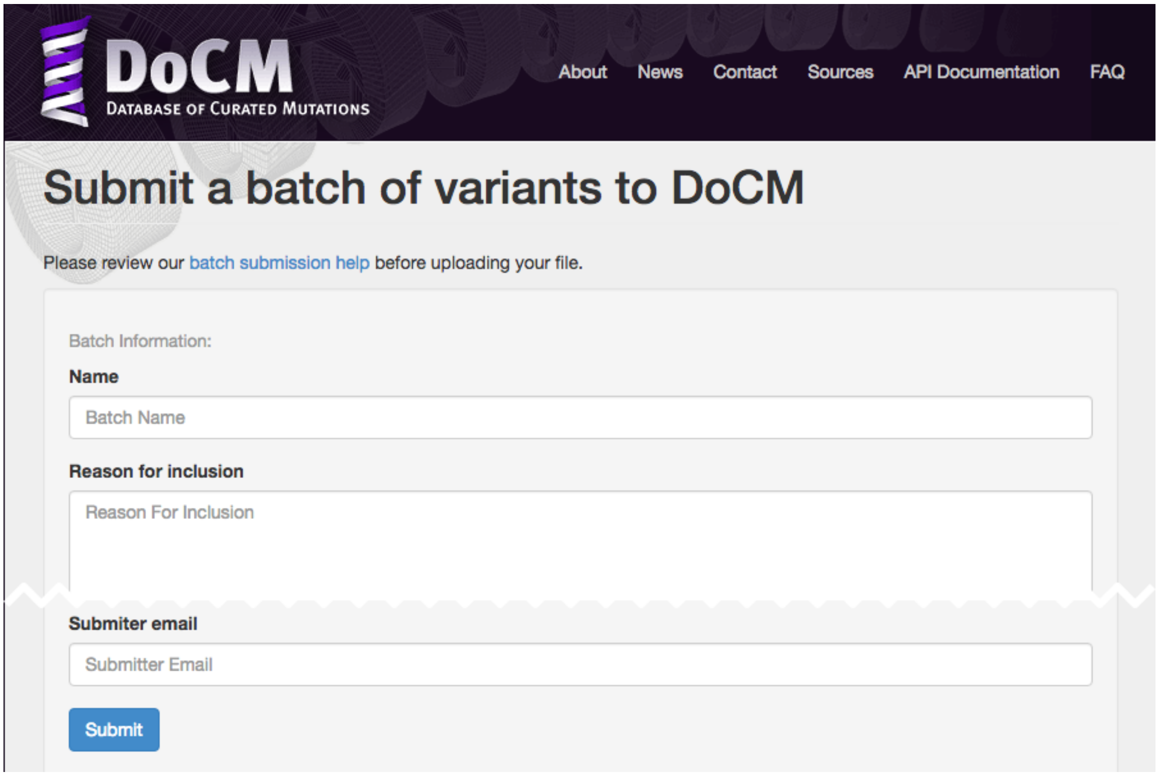
```
[
    {hgvs: "ENST00000256078:c.35C>A",
    chr: "12",
    start: 25398284,
    stop: 25398284,
    reference: "C",
    variant: "A",
    reference_version: "GRCh36",
    gene: "KRAS",
    mutation_type: "missense",
    amino_acid: "G12V",
    diseases: ["AML",
               "Thyroid cancer", ...]
    pubmed_sources: [ 19255327, ...]
    ...},
    ...
    {...}
]
```

Figure 2.1 Overview of DoCM resource*. (A) Outline of criteria to curate a variant. Variants are evaluated for inclusion and then curated elements are identified. (B) Summary of current DoCM contents. DoCM contains SNSs and indels across many cancer subtypes with easy identification of the journal article that outlines the variant's relevance. (C) Screenshot of the DoCM web application available at http://docm.info. (D) Illustration of the API. An HTTP GET request for a variety of parameters including gene, chromosome, position etc. and returns a JSON response with the PubMed ids, diseases and other useful information. The API is thoroughly documented at http://docm.genome.wustl.edu/api.*

DoCM Feature	**Type**	**Quantity**
Variant types	SNSs	1,302
	INDELs	35
	DNSs	27
Variant effect	Missense	1,276
	Stop	45
	Frameshift	18
	Inframe	15
	Start lost	5
	Synonymous	4
	Protein	1
Drugs		96
Cancer subtypes		122
Genes (transcripts)		132 (184)
Publications		876

Table 2.1   Summary of DoCM resource contents (v3.2)

DoCM currently houses 1,364 variants in 132 genes across 122 cancer subtypes, based on the curation of 876 publications (**Figure 2.1B** and **Table 2.1**). The variant types currently in DoCM include SNSs (95.5% of entries), indels (2.6% of entries), and dinucleotide substitutions (DNSs; 2% of entries). To date, the DoCM web application has approximately 24,000 page views and approximately 8,500 unique users. In aggregate, users have spent hundreds of hours interacting with the website and include individuals from every continent with the highest traffic coming from the USA. DoCM's open-access model has allowed the resource to be useful to the community, prior to publication, being used in original research[95], incorporated into other resources[96,97], and highlighted in a review[98].



Submit a batch of variants to DoCM

Please review our batch submission help before uploading your file.

Batch Information:

Name

Batch Name

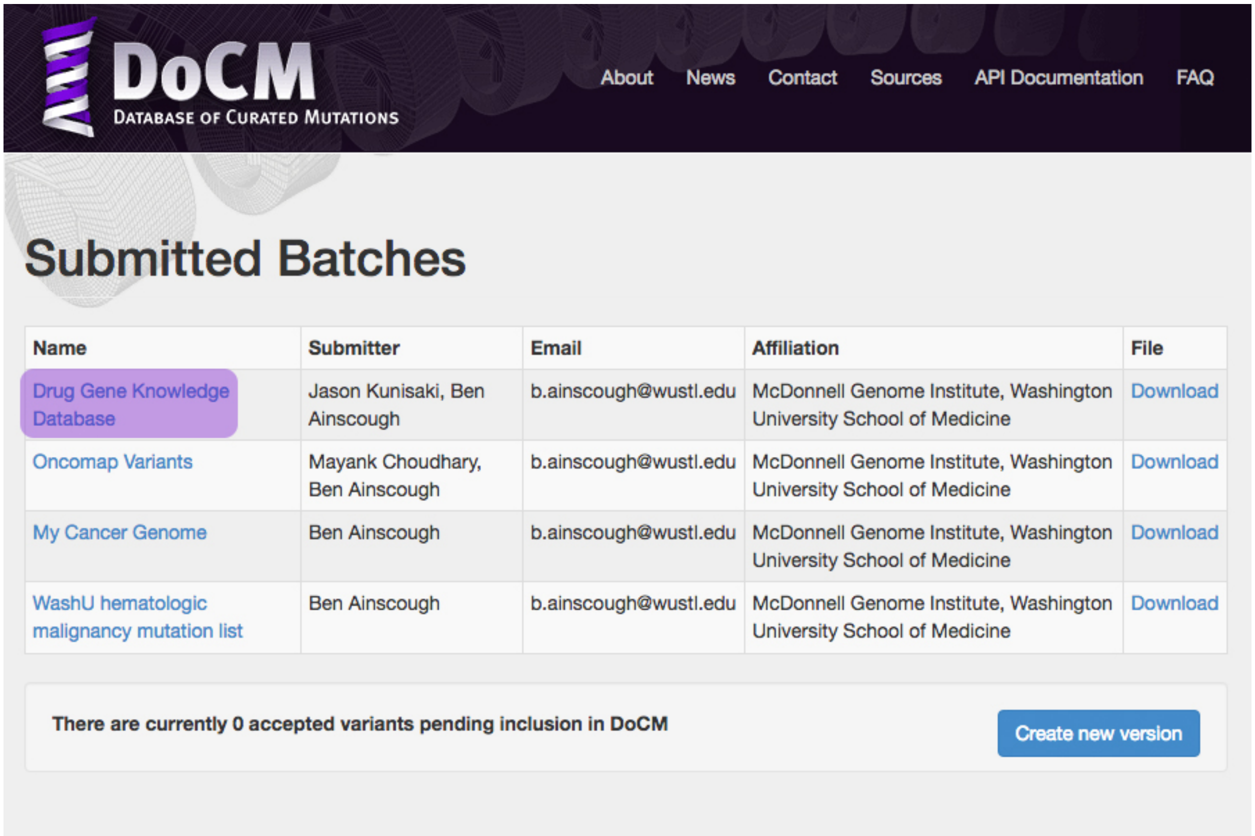
Reason for inclusion

Reason For Inclusion

Submiter email

Submitter Email

Figure 2.2 Screenshot of DoCM batch submission form. *In the batch submission form, users can enter all the parameters necessary for inclusion into DoCM, including the name of the batch, the rationale statement outlining the reason for including the variants and curation details, any relevant urls, tags to be applied to the whole batch, the TSV file with variants and submitter information. Following submission the user will be given a link to review the batch and any messages from moderators.*

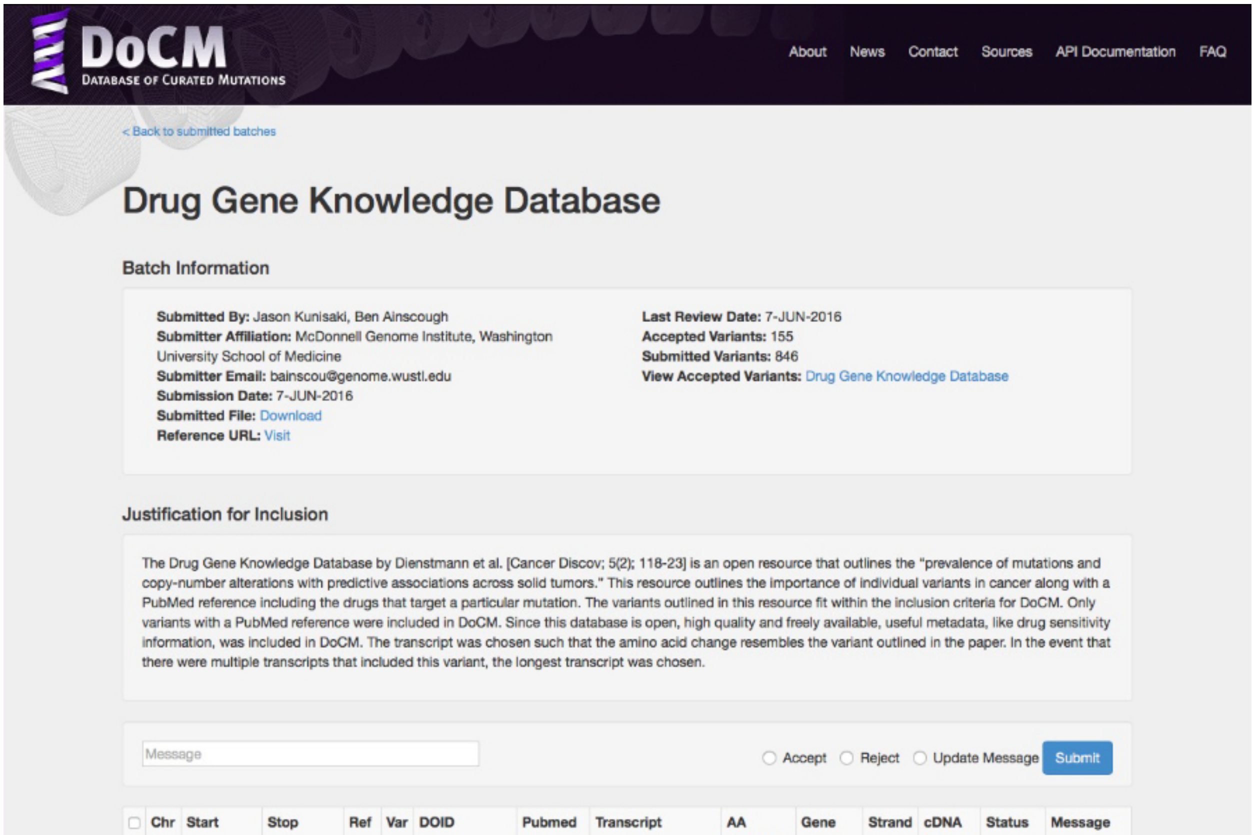


Submitted Batches

Name	Submitter	Email	Affiliation	File
Drug Gene Knowledge Database	Jason Kunisaki, Ben Ainscough	b.ainscough@wustl.edu	McDonnell Genome Institute, Washington University School of Medicine	Download
Oncomap Variants	Mayank Choudhary, Ben Ainscough	b.ainscough@wustl.edu	McDonnell Genome Institute, Washington University School of Medicine	Download
My Cancer Genome	Ben Ainscough	b.ainscough@wustl.edu	McDonnell Genome Institute, Washington University School of Medicine	Download
WashU hematologic malignancy mutation list	Ben Ainscough	b.ainscough@wustl.edu	McDonnell Genome Institute, Washington University School of Medicine	Download

There are currently 0 accepted variants pending inclusion in DoCM

Figure 2.3 Screenshot of moderators view of the submitted batches queue*. Once a batch has been submitted, it can be reviewed in the password protected moderator queue. A listing of current DoCM moderators can be viewed at http://docm.genome.wustl.edu/about. Moderators can select a batch, such as the Drug Gene Knowledge Database highlighted in purple above, to review the batch. Once multiple batches have been accepted a moderator can create a new DoCM version using the blue button at the bottom-right of the screen. The "Drug Gene Knowledge Database" link is highlighted in purple as it is the subject of* ***Figure 2.4****.*



Drug Gene Knowledge Database

Batch Information

Submitted By: Jason Kunisaki, Ben Ainscough
Submitter Affiliation: McDonnell Genome Institute, Washington University School of Medicine
Submitter Email: bainscou@genome.wustl.edu
Submission Date: 7-JUN-2016
Submitted File: Download
Reference URL: Visit

Last Review Date: 7-JUN-2016
Accepted Variants: 155
Submitted Variants: 846
View Accepted Variants: Drug Gene Knowledge Database

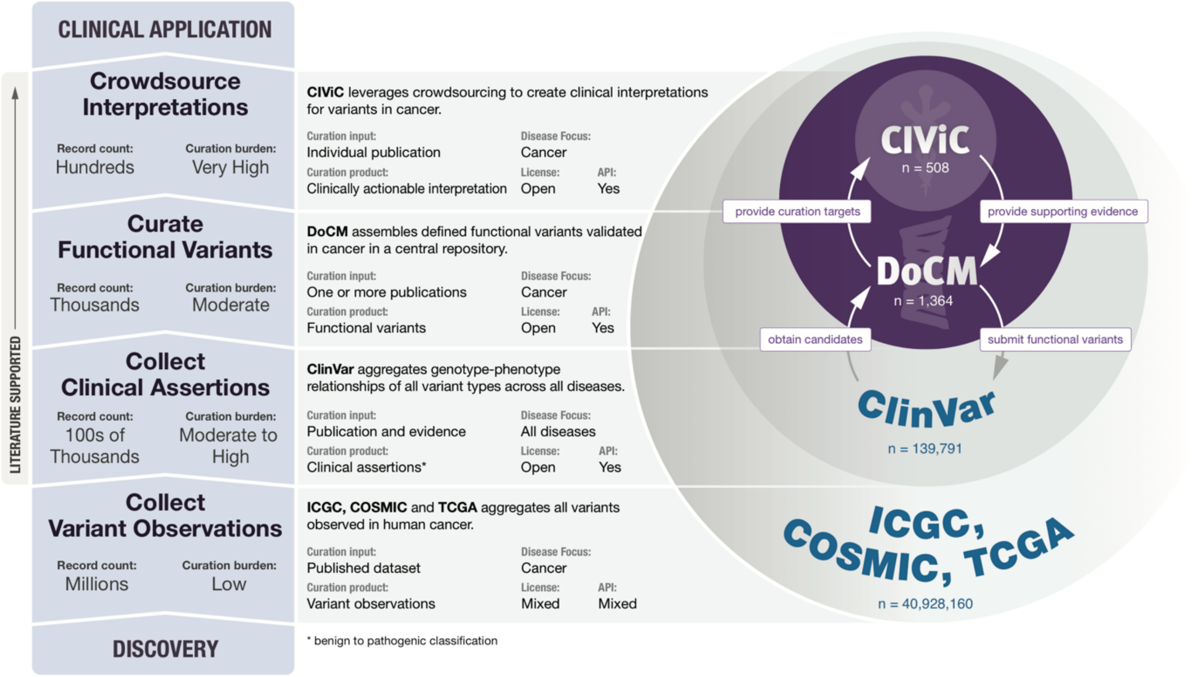
Justification for Inclusion

The Drug Gene Knowledge Database by Dienstmann et al. [Cancer Discov; 5(2); 118-23] is an open resource that outlines the "prevalence of mutations and copy-number alterations with predictive associations across solid tumors." This resource outlines the importance of individual variants in cancer along with a PubMed reference including the drugs that target a particular mutation. The variants outlined in this resource fit within the inclusion criteria for DoCM. Only variants with a PubMed reference were included in DoCM. Since this database is open, high quality and freely available, useful metadata, like drug sensitivity information, was included in DoCM. The transcript was chosen such that the amino acid change resembles the variant outlined in the paper. In the event that there were multiple transcripts that included this variant, the longest transcript was chosen.

□	Chr	Start	Stop	Ref	Var	DOID	Pubmed	Transcript	AA	Gene	Strand	cDNA	Status	Message

Figure 2.4 Screenshot of moderator review page*. A moderator can review all information submitted with a batch and evaluate whether it fits the scope and quality requirements of DoCM. Individual variants can be accepted or rejected and the moderator can leave a message to the submitter.*

DoCM's scope and its batch submission process (**Figures 2.2-2.4**) place it at a critical intersection between the two major tradeoffs of curated resources: comprehensiveness of variants and curation burden (**Figure 2.5**). The automated batch submission and the review system allow DoCM curations to scale easily.



	Record count:	Curation burden:		Curation input:	Disease Focus:	Curation product:	License:	API:
Crowdsource Interpretations	Hundreds	Very High	**CIViC** leverages crowdsourcing to create clinical interpretations for variants in cancer.	Individual publication	Cancer	Clinically actionable interpretation	Open	Yes
Curate Functional Variants	Thousands	Moderate	**DoCM** assembles defined functional variants validated in cancer in a central repository.	One or more publications	Cancer	Functional variants	Open	Yes
Collect Clinical Assertions	100s of Thousands	Moderate to High	**ClinVar** aggregates genotype-phenotype relationships of all variant types across all diseases.	Publication and evidence	All diseases	Clinical assertions*	Open	Yes
Collect Variant Observations	Millions	Low	**ICGC, COSMIC** and **TCGA** aggregates all variants observed in human cancer.	Published dataset	Cancer	Variant observations	Mixed	Mixed

* benign to pathogenic classification

Figure 2.5 ***Putting DoCM in the context of other resources***. *DoCM in the landscape of selected databases that collect and curate variants.*

Curation of the literature to produce a high-quality set of pathogenic somatic variants is not trivial, on account of the large number of papers and laborious curation process (**Figure 2.6**). Hence, we designed DoCM as an open resource that can coordinate contributions from research and clinical practitioners. Once important variants are identified, curation efforts are required to format, standardize, and structure the variants for inclusion in DoCM (see section 2.3 and **Figure 2.7**). A set of such curated variants can be contributed to DoCM by batch submission at http://docm.info/variant_submission, whereupon it is reviewed and evaluated by DoCM editors for possible inclusion. DoCM is licensed under the creative commons attribution license (CC BY 4.0), allowing academic and industry researchers unencumbered access to the content.

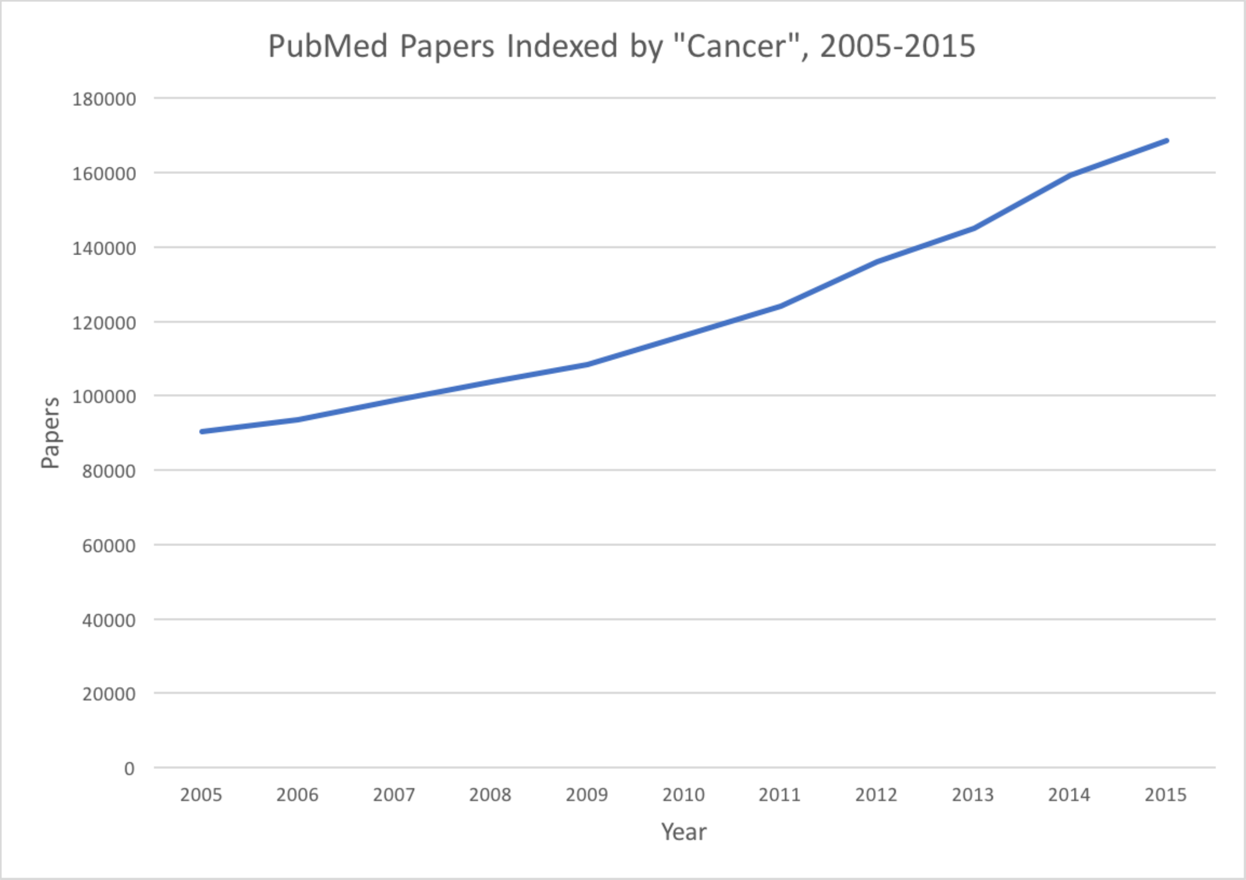


***Figure 2.6 Number of papers in PubMed indexed by "Cancer" per year**. Searching PubMed with the search term "Cancer" yields the number of papers relating to cancer per year. This serves as an upper-bound limit of the number of papers that need to be curated to accurately summarize important cancer variants. There is a need for public resources that reduce the duplication of curation effort.*

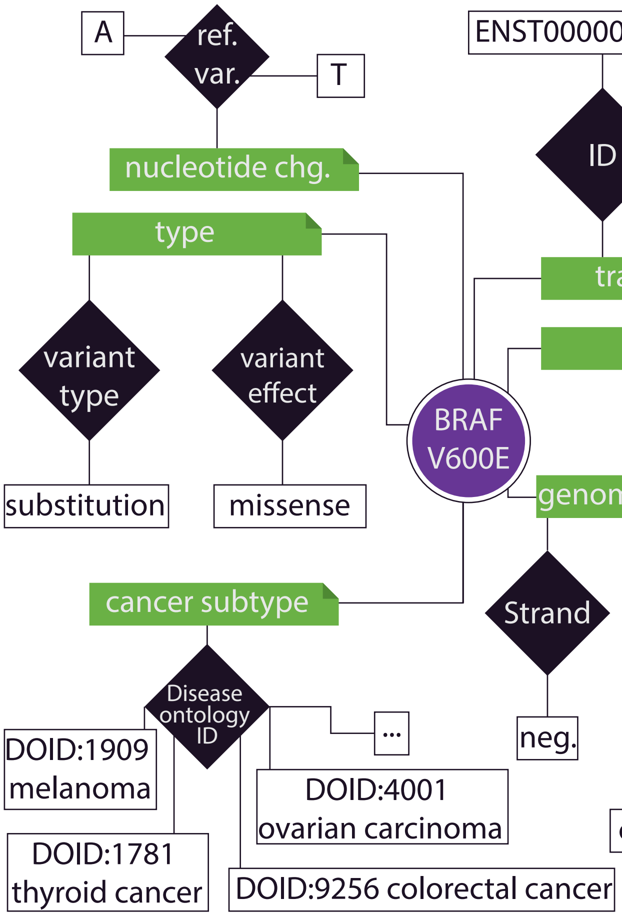


Figure 2.7 Overview of variant curation for every entry into DoCM*. An anecdotal example of the curation involved for the variant BRAF V600E is shown. Typically, the literature only lists the gene and amino acid change (purple in the figure), requiring extensive curation to uniquely identify the variant. Correct genomic coordinates on a consistent genome build need to be identified, with accompanying nucleotide and strand information. Occasionally there are multiple nucleotide changes that are synonymous with a particular amino acid change. A representative transcript that correctly models the variant described in the literature also needs to be specified. Cancer subtypes are specified using the disease ontology nomenclature. Green boxes note the class of information that needs to be captured in DoCM, black boxes show the subtype of each class, and white boxes denote the value.*

2.3 Methods

2.3.1 Somatic variant curation

Curation of the literature to produce a high quality set of pathogenic somatic variants is not trivial. Once identified, these variants require significant curation efforts to format and standardize the variants in a structured way for storage and retrieval in a relational database. For example, publications often only specify the amino acid change and gene name to describe the variant. Hence, for each variant a curator must reconstruct the genomic location, genome version, relevant transcript, and nucleic acid variant from the manuscript. Even when some of this information is specified, it is often from an outdated genome reference or transcript build/version (**Figures 2.1A and 2.6**).

To be included in DoCM, variants needed to be supported by peer-reviewed literature and/or by expert opinion indicating their relevance to cancer or a cancer subtype. Relevant publications were identified by PubMed searches and inspired by other somatic variant databases, such as CIViC (www.civicdb.org)[99], My Cancer Genome[77], OncoMap[94], and the Gene Drug Knowledge Database[78]. We reviewed publications referenced in these resources to determine the variant's suitability for inclusion in DoCM. Where licensing permissions allowed, the underlying metadata from the resource was included in the DoCM knowledgebase; otherwise a link to the source resource was included for reference.

The criteria for inclusion in DoCM are as follows. Variants, single nucleotide substitutions (SNSs) and insertions and deletions (indels), must have published evidence of clinical relevance, such as prognostic or diagnostic information and/or response data for targeted therapies. For example, a BRAF V600E variant might be considered a clinically important variant, as melanomas with this variant are sensitive to the drug vemurafenib[100]. Additionally, variants whose etiology in cancer has been established by functional experimentation, in either cell lines or model organisms, were included. Lastly, variation that has been observed in large scale sequencing efforts as being significantly associated with a particular cancer type were included in the resource (in this category significant recurrence and expert opinion was preferred).

Variants can be grouped into batches by commonalities like disease or mutation type if curated directly from the literature. Batches can also be created based on a publically available listing of variants that is in scope for the DoCM resource, like My Cancer Genome or the Drug Gene Knowledge Database. Batches can be submitted, following the instructions (http://docm.genome.wustl.edu/batch_submission_help), on the batch submission page (**Figure 2.2**)(http://docm.genome.wustl.edu/batches/new). Curators should annotate their curation process and explain the reasoning for including a batch into DoCM in the batch rationale statement on the submission form. This statement provides transparency to the curation process and allows DoCM users to better understand why variants were included.

Variants of sufficient biological or clinical importance were included in DoCM and annotated with the correct genomic position, nucleic acid change, associated cancer subtype, transcript, and any relevant tags. The genomic location of any variant was obtained from the publication, and converted to reference genome build GRCh37. However, publications rarely specify all relevant information needed to easily identify a particular variant. Often only the gene and amino acid change were listed (e.g., BRAF V600E); in such cases the genomic location must be manually determined by referencing resources like the UCSC[101] or IGV[26] genome browsers and appropriate gene annotation tracks such as Ensembl. If there were multiple nucleic acid variants that could result in the same amino acid change, then they were both included in the database. Additionally, a representative transcript that is compatible with the amino acid variant was also obtained if not specified in the publication.

2.3.2 Batch Submission and review process

Following submission of a batch, the DoCM web app automatically annotates the variants using VEP[102], validates the publication's pubmed ids using PubMed, and validates the disease ontology ids using the disease ontology API[103]. After annotation and validation, the variants are reviewed by the moderators listed on DoCM's about page (http://docm.genome.wustl.edu/about). DoCM moderators ensure that the submitted batch contains no errors in annotation and validation, that the batch is within the scope of the resource, and that the variants appear to be referenced in the listed literature. The moderator will start a dialogue with the submitter via email to correct any errors/discrepancies and then accept or reject the variants in the batch. These variants are then staged for inclusion in DoCM and upon submission of multiple batches the moderator can create a new version of the database (**Figures 2.3-2.4**).

2.3.3 DoCM web application implementation

The DoCM web application was built using Ruby on Rails (>= 1.9.3) with a PostgreSQL (>= 9) database backend. The application is open source (code available at https://github.com/genome/docm) and openly licensed (MIT open source). The application is organized using a model-view-controller (MVC) architecture. DoCM variants can be browsed and searched using the web interface, which has both quick search and advanced filtering functions. Additionally, the web application features a RESTful application programming interface (API), that allows for easy integration into other computational systems. Additionally, the API features a versioning system that allows for users to get consistent responses even as the resource is updated and new sources are included. The web application also features the option for direct download of the curated variants in TSV or VCF formats.

2.4 Authors and Contributions

Benjamin J. Ainscough[1,2], Malachi Griffith[1,2,3,*], Adam C Coffman[1], Alex H. Wagner[1], Jason Kunisaki[1], Mayank NK Choudhary[3], Joshua F. McMichael[1], Robert S. Fulton[1,2,3], Richard K. Wilson[1,2,3,4], Obi L. Griffith[1,2,4,*], Elaine R. Mardis[1,2,3,4]

[1]McDonnell Genome Institute, Washington University School of Medicine, St. Louis, Missouri, United States of America
[2]Siteman Cancer Center, Washington University School of Medicine, St. Louis, Missouri, United States of America
[3]Department of Genetics, Washington University School of Medicine, St. Louis, Missouri, United States of America
[4]Department of Medicine, Washington University School of Medicine, St. Louis, Missouri, United States of America
[*]Corresponding authors: obigriffith@wustl.edu, mgriffit@wustl.edu

B.J.A. wrote the manuscript, was responsible for supervising all curation of the literature, initial design of the web interface, testing, creating the knowledge-based variant calling strategy, analysis, initial design of validation sequencing experiment, and figure creation. M.N.K.C, M.G., O.L.G, E.R.M, and A.H.W contributed text and revised the manuscript. A.C.C. designed and implemented the web interface, database, and API. B.J.A., A.C.C., M.G., A.H.W, and J.F.M. made contributions to the code. J.F.M. was the lead user experience web developer. M.G., O.L.G., E.R.M., and A.H.W. provided beta testing feedback. M.N.K.C., J.K., and A.H.W. curated publications to include mutations in DoCM. R.S.F., M.G., O.L.G., and E.R.M. designed and supervised validation sequencing. M.G., O.L.G., and E.R.M. supervised analysis. O.L.G, M.G., E.R.M, and R.K.W provided funding.

2.5 Acknowledgements

The authors gratefully acknowledge Lee Trani, Jennifer Hodges, and Aye Wollam who helped with manual review of variant calls. Tim Ley, Ron Bose, Ramaswamy Govindan, and Siddhartha Devarakonda provided valuable input in the curation of DoCM. James Eldred helped oversee the development of the website. David Larson provided valuable input for the analysis performed. MG was supported by the National Human Genome Research Institute (NIH NHGRI K99HG007940). OLG was supported by the National Cancer Institute (NIH NCI K22CA188163). This work was supported by a grant to Richard K. Wilson from the National Human Genome Research Institute (NIH NHGRI U54HG003079).

Chapter 3: DoCM Rescues Clinically Impactful Cancer Mutations

Ainscough, B. J. et al. DoCM: a database of curated mutations in cancer. Nat Methods 13, 806-807, doi:10.1038/nmeth.4000 (2016)

3.1 Abstract

While DoCM has utility for annotating pathogenic variants with relevant publication support, it can also contribute to the analysis of sequencing data. For example, it is the nature of analytical pipelines for somatic variant detection that bona fide somatic mutations may be missed due to various causes–including overly stringent filtering, alignment challenges, low tumor purity, tumor heterogeneity, tumor contamination of normal, lack of data coverage, and other issues. This is particularly problematic when a variant with a known biological function or clinical action, such as those populating DoCM, is not detected. While missing a variant can adversely impact both research and clinical sequencing, it is particularly pertinent in the clinical assay of tumor DNA, where a false negative may represent a missed opportunity for more optimal disease management. To illustrate the utility of DoCM, we performed a focused knowledge-based variant discovery study to identify pathogenic variants missed in 1,833 cases across four TCGA projects. Validation sequencing data from 93 of these cases showed that at least one functionally important variant in DoCM was recovered in 41% of cases.

3.2 Results

3.2.1 DoCM recovers missed somatic variants

While DoCM has utility for annotating pathogenic variants with relevant publication support, it can also contribute to the analysis of sequencing data. For example, it is the nature of analytical pipelines for somatic variant detection that bona fide somatic variants may be missed due to various causes–including overly stringent filtering, alignment challenges, low tumor purity, tumor heterogeneity, tumor contamination of normal, lack of data coverage, and other issues[61]. This is particularly problematic when a variant with a known biological function or clinical action, such as those populating DoCM, is not detected. While missing a variant can adversely impact both research and clinical sequencing, it is particularly pertinent in the clinical setting, where a false negative may represent a missed opportunity for more optimal disease management. Herein we demonstrate the use of DoCM, a knowledgebase of clinically and/or biologically relevant cancer variants, to recover missed somatic variants in cancer sequencing data.

As a proof of principle for DoCM's utility, we designed a method to recover variants missed through traditional analysis strategies. This method utilizes DoCM to perform knowledge-driven identification of minimally supported variants followed by manual review and validation sequencing (MSRV). Here, MSRV identified pathogenic loci with two or more supporting sequence reads, followed by manual review for errors (**Figure 3.1A**), and subsequent validation sequencing (see section 4.3 methods). We applied the DoCM-MSRV variant recovery method by analyzing all DoCM sites (data freeze; 488 variants) in the sequencing data from four TCGA projects (**Figure 3.1**): AML), BRCA, OVCA, and UCEC. DoCM identified 10,174 minimally supported variants that were manually reviewed by an expert genome analyst. Of these, 1,833 variants passed manual review (i.e. showed sufficient evidence to warrant validation sequencing). A subset of these putative somatic variants, 1,228 of 1,833 (66.99%), was not reported by the respective TCGA variant calling pipeline (**Figure 3.1B, Table 3.1**).

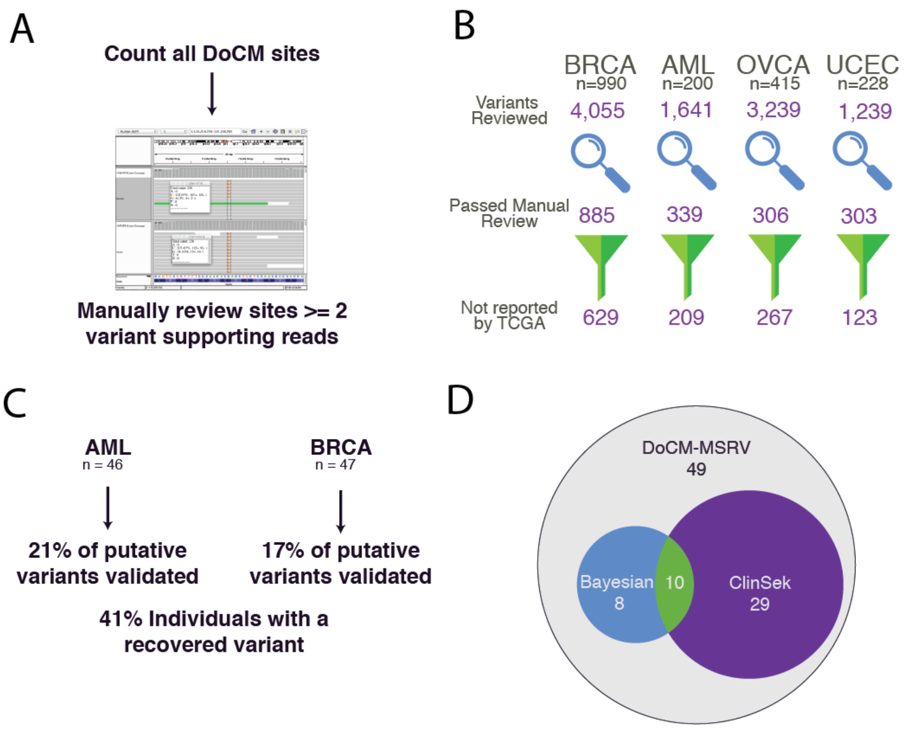


Figure 3.1 Overview of analysis and validation sequencing of four TCGA projects. *(A) Outline of the manual review strategy. DoCM sites with two or more reads of support are evaluated for obvious errors. (B) Summary of the variants that passed manual review and were not identified in the original TCGA analyses. (C) Summary of the variants that were validated in the 93 validation samples. (D) Comparison of DoCM-MSRV to ClinSek and the Bayesian classifier.*

	AML	BRCA	OVCA	UCEC
Number of individuals	200	990	415	228
DoCM sites counted across all samples (488 unique DoCM sites)	95,406	483,120	202,520	140,544
DoCM variant sites with >= 2 reads of support	1,641	4,055	3,239	1,239
DoCM variants called somatic via manual review	339	885	306	303
DoCM variants called somatic after manual review AND identified in TCGA	130	256	40	180
DoCM variants called somatic after manual review NOT identified in TCGA	209	629	267	123

Table 3.1 Summary of variants identified in TCGA data through manual review.

We next selected 48 cases (tumor-normal pairs) from the AML and BRCA TCGA projects, each of which containing one or more of the 1,833 putative variants, and performed validation sequencing using a targeted hybrid capture approach on both tumor and normal samples (see section 4.3 methods). We generated new libraries and used differentially barcoded indices on either end of the sequencing libraries to limit the likelihood that observed variants were due to sequencing artifacts via crosstalk from a single end barcode. One BRCA case and two AML cases failed library construction and QC checks, leaving 93 total cases in our validation cohort. After sequencing, we achieved a mean coverage of 173-fold at DoCM sites, and 95.5% of DoCM sites had at least 50-fold coverage (**Figure 3.2**).

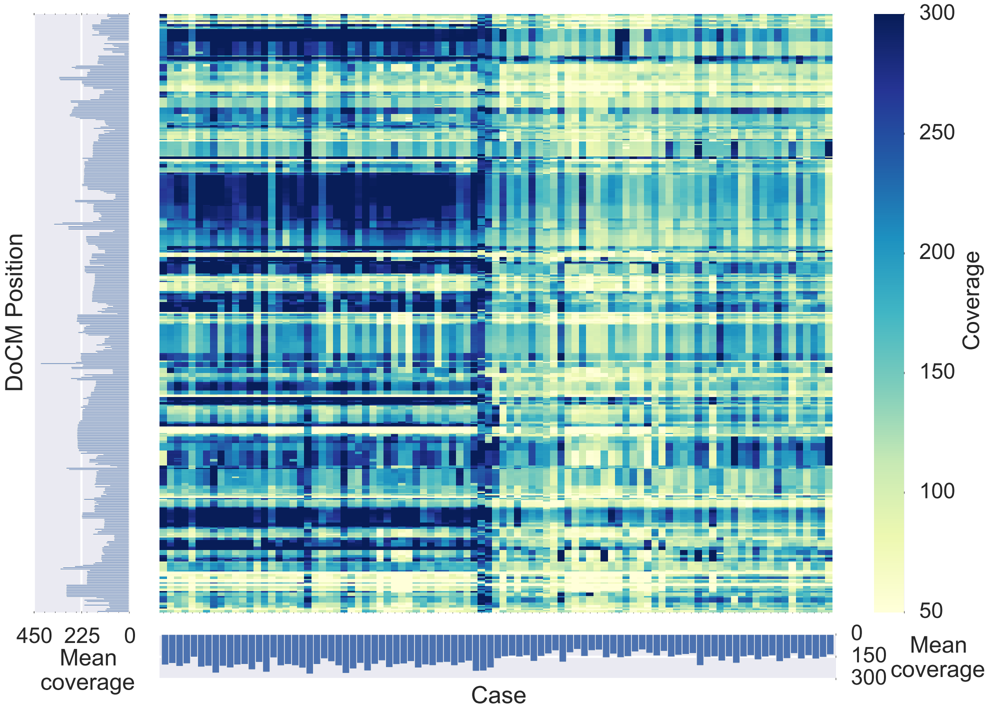


Figure 3.2 Coverage of the custom capture validation sequencing. *Heatmap illustrating the coverage obtained at all target sites in validation sequencing. Bar graphs on the x and y-axes illustrate the mean coverage at each case/position.*

For those cases with validation sequencing data, 253 putative somatic variants were missed in the original TCGA analysis and passed manual review. Of these, 19% (49) also passed the validation threshold (5 or more reads supporting the variant) (**Table 3.2**). Here, DoCM-MSRV was able to detect variants below 10% variant allele fraction (VAF), many of which are approaching the noise level of the sequencing instrument. Assuming a 1% error rate for a sequencing instrument, one would expect at least one read to contain a sequencing artifact at a given position with a depth of 100x. Therefore, our decision to review any variant with 2 or more supporting data reads approaches the detection limits of the sequencing instrument. As a result, we expected and observed a large false positive rate among DoCM-MSRV-identified putative variants prior to validation sequencing (~20% of putative variants validated in subsequent resequencing).

	AML	*BRCA*
Number of individuals where validation sequencing was performed	46	47
Putative variants called somatic via manual review	155	143
Putative somatic variants identified in TCGA	27	18
Putative somatic variants not identified in TCGA	128	125
Validated variants (>=5 reads, not called in TCGA, somatic in manual review)	27	23
Unvalidated variants (<5 reads, not called in TCGA, somatic in manual review)	97	104
Number of individuals with validated variant rescued	22/46	16/47

Table 3.2   Summary of validation sequencing results.

The presence of these validated, rescued variants observed elsewhere in TCGA coupled to the knowledge that all such variants are important in cancer gives high confidence in the validity of these variants. DoCM-MSRV was able to rescue variants across a broad range of VAFs (1.4% - 91%)(**Figure 3.3**) and identified at least one biologically important variant in 38 of 93 (41%) cases (**Figure 3.1C, Table 3.2**). If we extrapolate the rate of DoCM-MSRV detected variants to all cases in all four TCGA projects, we estimate that approximately 238 variants would be rescued. Similarly, we estimate that we would identify approximately 715 (out of 1743) cases where we would recover at least one or more clinically relevant variants that were missed in the TCGA pipeline (**Table 3.3**).

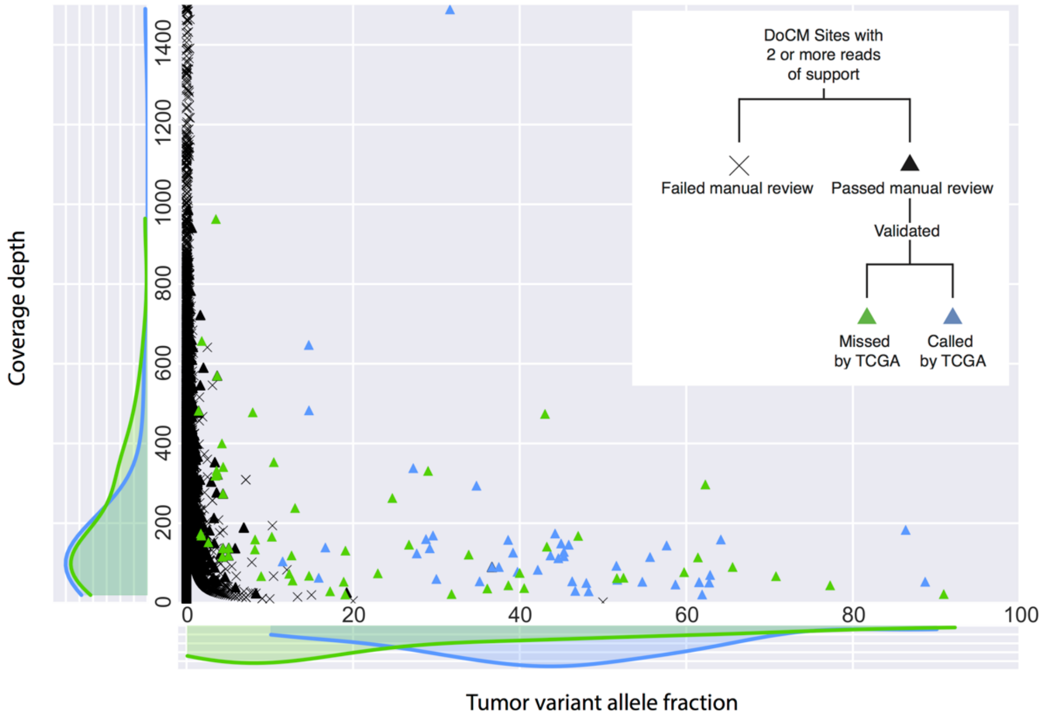


Figure 3.3 Overview of validation sequencing results*. Variant allele fraction plot illustrating the types of variants identified through manual review that validated. Variants called in the original TCGA study are highlighted in blue and those missed are in green. Note that TCGA was unable to call variants below ~10% VAF while the MSRV approach was able to recover many such variants. Density plots on the x and y-axes show the distribution of tumor VAF and coverage depth for validated variants respectively.*

Extrapolated estimation	AML	BRCA	OVCA	UCEC
Estimated number of variants that would validate	40	122	52	24
Estimated number of samples with rescued variants	84	398	169	94

Table 3.3 Estimated number of variants detected and samples with a recovered variant, extrapolated using the rate of recovery, in the validation data

3.2.2 Variant calling of validation sequencing data and comparison to MSRV

We performed orthogonal variant calling on the validation sequencing data using ClinSek[104] to compare how DoCM-MSRV performed against a knowledge-driven statistical approach. ClinSek identified 29 total somatic variants from the AML (13 variants) and BRCA (16 variants) validation cases, all of which were identified via the DoCM-MSRV approach (**Figure 3.1C**). The majority of variants that ClinSek missed (14/20) were AML cases with evidence of tumor contamination of the normal. In a second comparison, we evaluated all sites using a Bayesian classifier with a binomial log likelihood ratio filter (LLR > 10) to identify somatic sites (see section 4.3 methods). The filter identified 18/49 variants found through DoCM-MSRV; no additional variants were identified (**Figure 3.1C**). Of the 31 variants missed by the Bayesian classifier, 21 were likely not identified due to low VAFs (mean tumor VAF: 18%, st. dev.: 27%) and the remaining 10 variants (all from AML samples) showed evidence of tumor contamination of the normal upon manual review. In summary, 10 variants were called by all three methods and 29 were called by at least two methods (**Figure 3.1D**).

3.2.3 Cost benefit analysis of DoCM-MSRV approach

Performing the manual review portion of the DoCM-MSRV approach is not very costly. Using our cutoffs for this experiment (488 variants per sample counted, those with 2 or more reads of support were manually reviewed) there were 5.5 variants on average needing review. That cost us an average of $2.37 per case across all 1,833 cases evaluated. Given that these putative variants have established importance in the literature, this is a small price to pay to ensure that important variants are not missed. Validation sequencing is much more costly; sequencing 96 cases cost on average $322.01 per case. It is worth noting that sequencing costs could be reduced in a production assay, as we designed and purchased hybrid capture probes for this experiment that can be reused on many samples afterwards. Additionally, the targeted capture sequencing methodology could be greatly optimized for cost if performed at scale and it would not be necessary to assess all 488 sites as we did, merely the putative somatic sites could be assayed. While the DoCM-MSRV method proved to have the best sensitivity, computational approaches like ClinSek and the bayesian classifier had excellent specificity and could identify a majority of the important variants at little cost.

3.2.4 Clinically actionable variants rescued by DoCM-MSRV

Using this knowledge-driven approach we were able to rescue missed variants with likely clinical impact, even in the relatively small sample set for which we performed validation sequencing (n=93). For example, FLT3 variants (D835E/N/Y and N676K) were identified in three AML cases (TCGA-AB-2835, TCGA-AB-2919, and TCGA-AB-2922). These variants are potentially sensitive to targeted therapies like sunitinib[105,106]. We also identified two AML cases with KIT N822K variants for which cell line studies indicate sensitivity to dasatinib[107]. We identified 13 BRCA cases that harbored variants in PIK3CA (E542K, E545K, and H1047R) or PTEN (R130Q) suggesting sensitivity to PI3-kinase pathway inhibitors[108,109]. Of the variants we validated, the mean VAF was 25.5% with a standard deviation of 23%, indicating that most validated sites were at low VAF (**Figure 3.3**) and are likely the result of sub-clonality or low tumor purity. Hence, DoCM can provide a vitally important capability to detect challenging variants that are in some cases potential harbingers of acquired resistance.

The identification of variants of biological import is particularly germane to clinical sequencing workflows, which are rapidly being established in clinical cancer care, since a false negative result would miss an important clinical indicator. In the research setting, DoCM has proven valuable in identifying previously undiscovered variants of potential clinical importance that were completely missed by large-scale discovery efforts such as TCGA. As our understanding of the functional landscape of cancer variants grows, and as that information is curated in DoCM, this knowledge-based variant detection strategy will become increasingly valuable.

The DoCM-MSRV method demonstrated that clinically relevant somatic variants are often missed. Hence, including the DoCM-MSRV approach as a failsafe in clinical analysis pipelines can ensure the likelihood of false negatives in a clinical assay is very low. In a comparison to other methods, DoCM-MSRV rescued the largest number of known pathogenic variants due to its sensitivity to variants at low VAF or with contaminating tumor signatures in the “normal” comparator specimen. Other highly sensitive variant calling methods have been developed since these earlier TCGA analyses were performed[65,66]. However, without a knowledge-based filter, these algorithms have substantially higher false negative rates of variant detection when compared to DoCM-MSRV.

3.3 Methods

3.3.1 Recovery of unidentified DoCM variation in the TCGA data using manual review and validation sequencing (MSRV)

Important somatic variants are sometimes missed by variant calling pipelines. To illustrate the utility of DoCM we applied the knowledge-base to our cancer genomics pipeline [110]. We used data from four TCGA projects: Acute Myeloid Leukemia (AML), Breast Cancer (BRCA), Ovarian Carcinoma (OVCA), and Uterine Corpus Endometrial Carcinoma (UCEC)[111-114]. All publically available samples were used. These projects were chosen because of the availability of their data and our access to a subset of the samples allowing us to perform validation. The TCGA exomes are available via the database of genotypes and phenotypes (dbGaP) study accession number phs000178.v9.p8.

We developed an approach for manual review and validation sequencing that first involved obtaining variant and reference base supporting read counts for all variants in DoCM using bam-readcount (https://github.com/genome/bam-readcount). Any variant with more than 2 reads of support was sent to an expert for manual review, a process that we typically perform on variants called by somatic variant callers to eliminate false positives. After manual review, we performed a validation sequencing experiment using targeted capture probes for all DoCM sites in 46 AML and 47 BRCA individuals. Samples with the highest number of putative somatic variant calls were chosen for validation sequencing. For any given individual, this filtering strategy lead to 5-20 variants to review.

3.3.2 Manual Review

Manual review involves inspecting raw sequence data (alignments) to correctly identify real somatic variants and likely sequencing/mapping artifacts using a genome viewer such as IGV[27]. A reviewer examined the bam files for tumor and matched normal samples concurrently and interrogated the data for a variety of causes of error. These errors can be due to sequencing or alignment errors, such as those found at highly repetitive or GC-rich regions of the genome. A reviewer will also look for inflated support for a somatic variant. For example, if a variant with 10 reads supporting a variant in the tumor sample is being evaluated, but 7 of those reads contain multiple discrepancies at other positions on the read (indicating misaligned reads), only three believable reads support this somatic variant at the locus and it is much less likely to be a real variant. In blood cancers in particular, there is often tumor cell contamination of the matched normal sample (e.g., skin) that leads to higher than usual levels of tumor variant signal in the normal data. This can result in false negatives as many somatic variant callers have a strong assumption that a variant will not be present in the normal above the rate expected by random sequencing errors. Also, the clonal heterogeneity of cancer can cause variants from a smaller sub-clone to not have sufficient signal to be identified by a somatic variant caller, however a reviewer can identify a trend of low variant allele fraction (VAF) variants with consistent levels throughout the sample suggesting that they are real. Low VAF can also be caused by low tumor cellularity.

In this study, we established a baseline for missed important variations, ensuring the utmost sensitivity for clinically impactful variant detection. As such, we gave our reviewers special instructions to pass any variants that they believed were worth attempting to validate via custom capture sequencing.

3.3.3 Manual review guidelines

Manual Review Tab Delimited Reports:

Format:

Chromosome	**Start**	**Stop**	**Reference**	**Variant**	**Called Code**	**Description Code**
1	50000	50000	A	C	A	LV,SI

- Called/Description Codes should all be uppercase
- Only 1 Called Code allowed
- Multiple Descriptions Codes allowed and should be comma separated with no spaces in between

Called Code:

Code	**Description**
A	Ambiguous, variant could or could not be real
F	Fail, variant failed manual review
G	Germline, variant is a real germline variant
S	Somatic, variant is a real somatic variant

Description Codes:

Code	Description
AI	Adjacent Indel, Variant likely due to misalignment of an adjacent indel
HDR	High Discrepancy Region, Region contains many reads with multiple mismatches
MM	Multiple Mismatches, Read contains multiple mismatches from reference
MV	Multiple Variants, More than 1 non-reference variant at the same base location
MN	MonoNucleotide run, Region contains pattern of repeat ex. AAAAAA
DN	DiNucleotide run, Region contains pattern of repeat ex. AGAGAG
TR	Tandem Repeat, Region contains pattern of repeat ex. ACGACGACG
LCT	Low Coverage in Tumor, Region contains low coverage in tumor
LCN	Low Coverage in Normal, Region contains low coverage in normal
NCN	No Coverage in Normal, Region contains no coverage in normal
TN	Tumor in Normal, Variant support in normal (common in blood cancers)
LVF	Low Variant Allele Frequency, Variant has a low VAF
LM	Low Mapping quality, Reads are poorly mapped
SI	Short Insert, Reads supporting variant result from short inserts
SIO	Short Insert Only, Reads supporting variant contain only short inserts
SSE	Same Start/END, Reads supporting variant have same start or end points
D	Directional reads, Majority of reads are in the same direction
E	End of reads, Variant only supported by the end of reads
AO	Ambiguous Other, Provide an explanation not otherwise specified here

Glossary:

Short Insert - Fragment sequenced twice because the paired end reads overlap (i.e. the insert size is negative)

3.3.4 Validation sequencing

We designed a custom capture reagent for a data freeze of 488 loci of DoCM, comprised mainly from the first four batches (“My Cancer Genome”, “Drug Gene Knowledge Database”, “Literature”, “WashU hematologic malignancy mutation list”), using probes from Integrated DNA Technologies (IDT, Coralville, IA). Probes that spanned these loci that were already available in house were used and additional probes were designed around the remaining sites. We attempted to validate 48 AML and 48 BRCA samples that had the most putative somatic variants for which tumor and matched normal tissues were available. We created dual-indexed libraries that were sequenced on HiSeq2500 (AML) and HiSeq2000 (BRCA) platforms with paired 2x100 bp reads as previously described[61,110].

3.3.5 Analysis of validation data

Custom capture reads were aligned to build 37 of the NCBI human reference genome (GRCh37) using BWA (version 0.5.9) and duplicates were removed using picard (version 1.46). DoCM sites were counted using bam-readcount (version 0.7.4) to assess coverage and perform downstream analysis. Putative variants with at least 70x coverage and 5 or more reads of support were classified as validated in the knowledge-driven approach. ClinSek version 0.1 was run on the matched tumor and normal bam files for each validation sample using the spileup option with default parameters and the DoCM data freeze as the site list[104]. The Bayesian classifier was run using the Genome Modeling System (GMS)[110] infrastructure at the McDonnell Genome Institute using the ‘gmt validation identify-outliers command’ with a tumor purity value of 66% and a normal contamination rate of 3%. Tumor purity and normal contamination values were estimated using the mean rate observed in samples identified via manual review from the TCGA exome data.

3.3.6 Bayesian Classifier Methodology

The following is an explanation of the methodology of the bayesian classifier used in the manuscript. The program is integrated into the Genome Modeling System[110] and can be ran using the 'gmt validation identify-outliers' command as described below.

gmt validation identify-outliers

SNV/Indel calls were filtered through a Bayesian classifier to select only those sites that were classified as somatic. Briefly, this classifier works by considering the likelihood of the data being generated by 7 different binomial models. Each model's log-likelihood function takes the form:

$$\log\mathcal{L}(N_r, N_v, T_r, T_v) = N_r\log(1-\theta_N) + N_v\log\theta_N + T_r\log(1-\theta_T) + T_v\log\theta_T$$

where N_r is the number of reads supporting the reference allele in the normal sample, N_v is the number of reads supporting the variant allele in the normal sample, T_r is the number of reads supporting the reference allele in the tumor sample, T_v is the number of reads supporting the variant allele in the tumor sample.

Each of the different models defines θ_N and θ_T as below:

$$(\theta_N, \theta_T) = \begin{cases} \epsilon, \epsilon & \text{for the Reference model} \\ 0.5, 0.5 & \text{for the Germline heterozygote model} \\ 1-\epsilon, 1-\epsilon & \text{for the Germline homozygote model} \\ \epsilon + \dfrac{f_t c_n}{p_t}, f_t & \text{for the Somatic model} \\ 0.5, 1-\epsilon & \text{for the LOH variant model} \\ 0.5, \epsilon & \text{for the LOH reference model} \\ f_s, f_s & \text{for the Noise model} \end{cases}$$

where ϵ is a constant error rate of 1%, f_t is the observed variant allele frequency in the tumor, f_s is the observed variant allele frequency at the site (across both samples), c_n is the contamination rate of the normal sample in the tumor, and p_t is the purity of the tumor sample.

Sites pass the filter if the Somatic model has the maximum likelihood amongst all models and the log-likelihood ratio of the Somatic model to the next most likely model is greater than 3.

Code available at:

https://github.com/genome/genome/blob/master/lib/perl/Genome/Model/Tools/Validation/IdentifyOutliers.pm

3.3.7 Code availability

Code for the DoCM web application are available at https://github.com/genome/docm.

3.4 Authors and Contributions

Benjamin J. Ainscough[1,2], Malachi Griffith[1,2,3,*], Adam C Coffman[1], Alex H. Wagner[1], Jason Kunisaki[1], Mayank NK Choudhary[3], Joshua F. McMichael[1], Robert S. Fulton[1,2,3], Richard K. Wilson[1,2,3,4], Obi L. Griffith[1,2,4,*], Elaine R. Mardis[1,2,3,4]

[1]McDonnell Genome Institute, Washington University School of Medicine, St. Louis, Missouri, United States of America
[2]Siteman Cancer Center, Washington University School of Medicine, St. Louis, Missouri, United States of America
[3]Department of Genetics, Washington University School of Medicine, St. Louis, Missouri, United States of America
[4]Department of Medicine, Washington University School of Medicine, St. Louis, Missouri, United States of America
[*]Corresponding authors: obigriffith@wustl.edu, mgriffit@wustl.edu

B.J.A. wrote the manuscript, was responsible for supervising all curation of the literature, initial design of the web interface, testing, creating the knowledge-based variant calling strategy, analysis, initial design of validation sequencing experiment, and figure creation. M.N.K.C, M.G., O.L.G, E.R.M, and A.H.W contributed text and revised the manuscript. A.C.C. designed and implemented the web interface, database, and API. B.J.A., A.C.C., M.G., A.H.W, and J.F.M. made contributions to the code. J.F.M. was the lead user experience web developer. M.G., O.L.G., E.R.M., and A.H.W. provided beta testing feedback. M.N.K.C., J.K., and A.H.W. curated publications to include mutations in DoCM. R.S.F., M.G., O.L.G., and E.R.M. designed and supervised validation sequencing. M.G., O.L.G., and E.R.M. supervised analysis. O.L.G, M.G., E.R.M, and R.K.W provided funding.

3.5 Acknowledgements

The authors gratefully acknowledge Lee Trani, Jennifer Hodges, and Aye Wollam who helped with manual review of variant calls. Tim Ley, Ron Bose, Ramaswamy Govindan, and Siddhartha Devarakonda provided valuable input in the curation of DoCM. James Eldred helped oversee the development of the website. David Larson provided valuable input for the analysis performed. MG was supported by the National Human Genome Research Institute (NIH NHGRI K99HG007940). OLG was supported by the National Cancer Institute (NIH NCI K22CA188163). This work was supported by a grant to Richard K. Wilson from the National Human Genome Research Institute (NIH NHGRI U54HG003079).

Chapter 4: Somatic Variant Refinement: Recovering Cancer Variants Lost to Manual Review

Ainscough, B. J. et al. in preparation (2017)

4.1 Abstract

Cancer genomic analysis requires accurate identification of somatic variants in sequencing data. While most steps in identifying somatic variants have been automated, manual review, or 'somatic variant refinement', is still required to remove false positives. However, this process of manual variant refinement is time-consuming, costly, poorly standardized, and non-reproducible. It remains, nonetheless, indispensable for accurate analysis of cancer data, especially as cancer genomics is brought into the clinic, where it is increasingly used to guide therapy. Here, we systematized and standardized somatic variant refinement using machine learning models, and training data from 41,000 manually reviewed variant calls from 21 different studies representing 9 cancer types and 440 cases. The final model accurately reproduced the manual variant refinement process, and accurately predicted which variants would be confirmed by orthogonal validation sequencing data. Highlighting its significance, this approach identified several clinically actionable variants missed by manual variant refinement, increasing the number of actionable variants identified by 6.1%. These results indicate that systematization of the variant refinement process can substantially increase the efficiency, accuracy, and reproducibility of cancer genomic analysis.

4.2 Introduction

Somatic variant callers are commonly used to identify somatic variants from aligned sequence reads in cancer genomics studies and in clinical cancer assays[61]. These callers attempt to statistically model sample purity, sequencing errors, zygosity, ploidy, and other factors. Post-processing (variant filtering and manual review) of somatic single nucleotide variants (SNVs) is a process we term 'somatic variant refinement' and is an important next step and is distinct from variant calling because it eliminates false positives from a candidate somatic variant list by setting thresholds for a variety of metrics including read coverage depth, variant allele fraction detected, base quality metrics, and others. The manual review of somatic variants is a process that requires an individual to directly examine aligned reads using a read alignment viewer such as IGV[26] to identify common error patterns that are consistently missed by state-of-the-art somatic variant callers[75,115].

Somatic variant refinement often improves calls by taking into account information neglected or unavailable to standard variant callers. Reviewers look for several patterns that can increase or reduce confidence in a variant call, for example, (1) all supporting reads are oriented in the same read direction; (2) independent fragment support is lower due to overlapping reads from short DNA fragments; (3) alignment errors related to homopolymer stretches, short repeats, or other low-complexity sequences; (4) supporting reads include multiple errors, indicating possible misalignment; (5) variants consistent with tumor contamination of normal; (6) all support occurs at the end of the sequencing reads, where overall error rates are higher; and others(**Figure 4.1**).

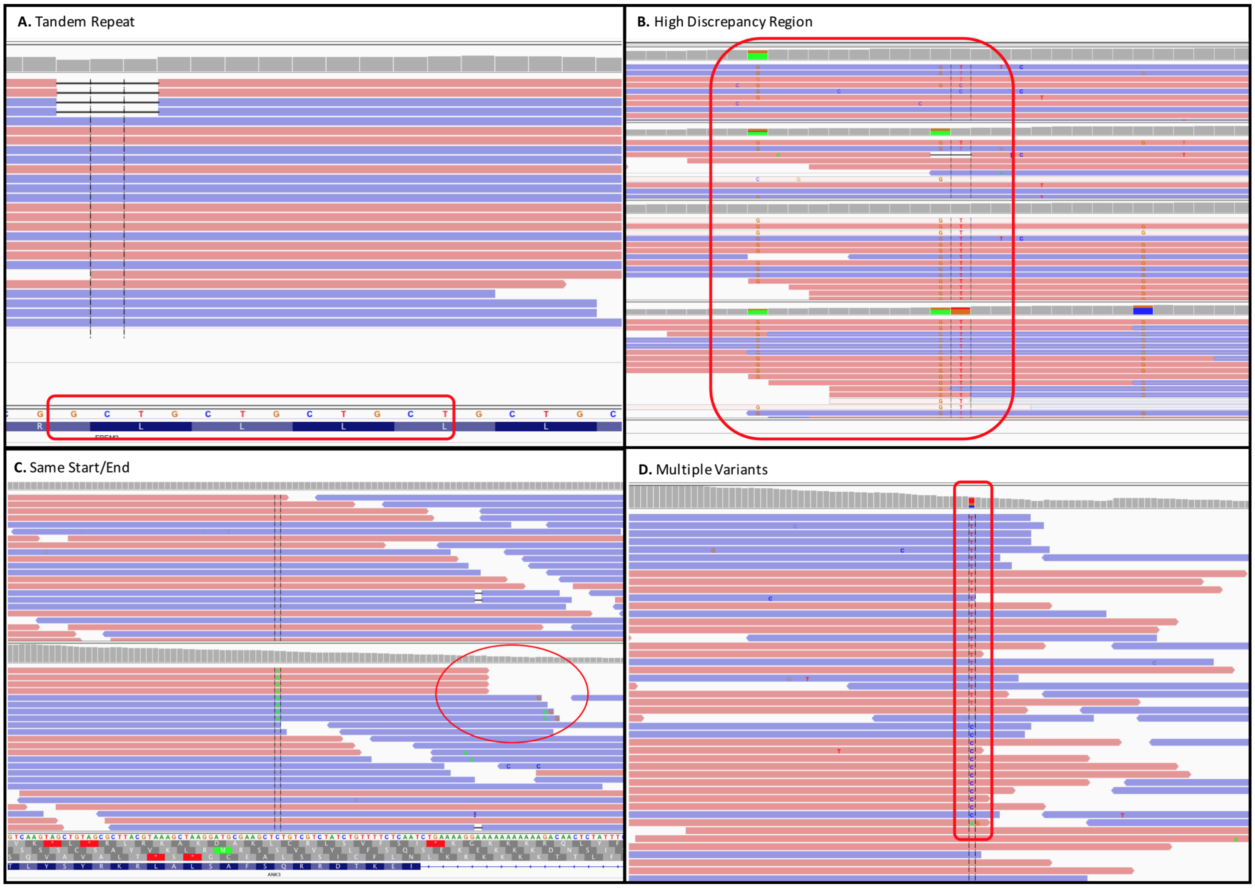


Figure 4.1 ***IGV Snapshots of common false positives that allow manual reviewers to filter out artefactual somatic calls.*** *(A) Example of Tandem Repeat (B) Example of a High Discrepancy Region (C) Example of Same Start/End (D) Example of Multiple Variants at the same loci*

If the number of problematic variant reads at a locus is high, a reviewer may eliminate the variant identified by a somatic variant caller. Manual reviewers often balance several observations to disqualify reads and make their final variant call.

Somatic variant refinement has a large impact on the final variant calls. A previous study showed that up to 44% of calls made by variant callers are failed during the variant refinement process[116]. Although post-processing of somatic SNVs can dramatically improve the accuracy of somatic variant calling, filtering and manual variant refinement strategies are often unstated or only briefly mentioned. Typically, methods for sequencing studies will state, “Translocation calls were manually inspected in IGV“[117], or “All indels were manually reviewed in Integrative Genomics Viewer”[118]. Some methods do not even use manual somatic refinement and merely require, “coverage of at least 50 reads in both tumor and normal samples, >20% of reads supporting the variant in tumor samples and <5% of reads supporting the variant in normal samples”[119]. These are just a small sampling of a very prevalent history of under-reporting the details of variant refinement processes from our institute and others [120]. As a result, standardization of somatic variant calling or comparison between studies becomes nearly impossible. Discrepancies in manual review operating procedures could result in major inter- and intra-lab variability and error. To address the issue of reproducibility, our standard operating procedure for somatic variant refinement can be found in **Appendix 1**. Additionally, while post-processing variant calling is extremely important, it is very time-consuming and therefore, expensive. Experienced reviewers can evaluate approximately 70-100 variants per hour and these numbers are likely lower in clinical cancer NGS assays where consequence of mislabels are greater. In one study completed by our lab, 10,112 variants were called by somatic variant callers, 1,066 were filtered out using sequencing metrics, and 9,046 variants required direct manual review. This manual review would have taken approximately 90-130 hours to complete by highly trained staff[116].

Previous studies have used machine learning algorithms to call somatic variants[121,122], however these studies have had small training data sizes (fewer than 3,000 variants) or were conducted with data from a limited number of cancer types. These studies have attempted to automate somatic variant calling. However, the high quality, multi-factor filtering currently provided by the manual refinement of putative somatic variant calls has not yet been automated[123]. In this study, we demonstrate that machine learning can be used to systematize and standardize somatic variant refinement, alleviating the manual review bottleneck.

In this study, we present a model that automates refinement of variants called from sequencing data. We show that the use of a model, as presented here, could substantially reduce a major bottleneck in cancer genomic analysis while improving reproducibility and transparency in genomic studies and in the clinical setting. This model is built on a training dataset of 41,000 variants from 21 studies with 440 cases, including 9 cancer types. All cases included paired tumor and normal samples that had been evaluated by manual variant refinement, and required an estimated 585 hours of effort. For each variant, we assembled 71 features to train the model including cancer type, tumor and normal read depth, tumor and normal variant allele fraction, base quality, mapping quality, etc. (**Appendix 2**). This dataset is an order of magnitude larger than previous studies that utilized machine learning to detect somatic variants and has representation across both solid and hematological malignancies, which have distinct genomic signatures that often complicate somatic variant calling (**Table 4.1**). Additionally, this dataset includes cancer types covering a broad range of average mutation burden, from leukemia on the low end to lung cancer and melanoma on the high end. This broad representation should allow the machine learning algorithms to generalize well across different cancer subtypes.

	Training Set (n=27,470)	**Testing Set** (n=13,530)	**Total** (n=41,000)
Malignancy (410 cases)			
Leukemia (243 cases)	5,815	2,877	8,692
Lymphoma (23 cases)	1,263	628	1,891
Breast (135 cases)	8,986	4,320	13,306
Small Cell Lung (18 cases)	9,177	4,601	13,778
Glioblastoma (17 cases)	844	412	1,256
Melanoma (1 cases)	185	100	285
Colorectal (1 case)	842	419	1,261
Gastrointestinal Stromal (1 case)	70	31	101
Malignant Peripheral Nerve Sheath (1 case)	288	142	430
Sequencing Methods			
Capture Sequencing	9,479	4,755	14,234
Exome Sequencing	9,367	4,677	14,044
Whole Genome Sequencing	8,624	4,098	12,722
Variant Calls			
Somatic	12,266	6,115	18,381
Ambiguous	7,189	3,454	10,643
Fail	5,909	2,945	8,854
Germline	2,106	1,016	3,122

Table 4.1 Overview of the cancer sequence data included in the training and testing set with regard to malignancy, sequencing approach, and manual review variant calls.

Here, we present a deep learning classifier that automates manual review of aligned and variant called sequencing data. We show that the use of a classifier as presented here could substantially reduce a major bottleneck in cancer genomic analysis while improving reproducibility and transparency in genomic studies and in the clinical setting.

4.3 Methods

4.3.1 Data assembly and standardization

We assembled manual review data from 21 different recent cancer genomic studies at the McDonnell Genome Institute, including 11 genomic discovery cohorts, 1 clinical trial, and 9 case studies (**Table 4.2**)[61,124-130]. In total, 440 samples were evaluated, with 266 samples derived from hematologic malignancies and 174 samples derived from solid tumors. Samples were only included in the training dataset if paired tumor-normal sequencing data and manual review calls were available. Since some samples were used in multiple studies, we eliminated sample duplicates by removing all sample pairs with more than 70% co-occurrence of genomic mutations. Sequencing data was produced from whole genome, exome, or custom capture sequencing and all data were aligned to reference genome hg19/GRCh37 using bwa or bwa-mem[17,131](**Table 4.1**).

Manual review for all projects was performed as outlined in **Appendix 1**. Reviewers manually refine variants using 4 distinct classes: "somatic" (S) - a variant that has sufficient support in the tumor in the absence of obvious sequencing artifacts; "ambiguous" (A) - a variant with insufficient sequencing support to definitively classify the variant; "germline" (G) - a variant that has substantial support in the normal sample beyond what might be considered attributable to tumor contamination of the normal; and "fail" (F) - a variant with low variant read support and/or reads that indicate sequencing artifacts, yet has acceptable variant coverage. As reviewers call variants "ambiguous", "germline", or "fail", they often provide additional notes or tags describing the reason for each call; these tags classify common reasons for rejection. For example, apparent deletions can be caused by misalignment of reads to tandem repeat regions in the reference sequence (**Figure 4.1A**), high discrepancy regions can be caused by failure to properly align to the correct homolog (**Figure 4.1B**), reads with the same start and end can be caused by PCR amplification artifacts that were not removed by de-duplication software (**Figure 4.1C**), or multiple variants can be supported at the same loci (e.g. at a site with a reference allele of A reads supporting a G and a T allele are present)(**Figure 4.1D**). "Germline" and "failed" calls enumerate two distinct types of failure, however, since germline and failed calls do not invoke any downstream analysis procedures, they were merged into one class called “failed”. Therefore, the machine learning classifier will make calls for “somatic” (true positive), “ambiguous” (uncertain if somatic or failed) and “fail” classes. All manual review results were standardized to a one based coordinate system using a Python tool developed for this purpose (https://github.com/griffithlab/convert_zero_one_based). Relevant metrics were extracted from the bam files using bam-readcount (https://github.com/genome/bam-readcount). Bam file metrics were merged with cancer type and reviewer information. All continuous features were normalized to fall between 0 and 1 using Scikit-learn’s MinMaxScaler[132]. All categorical variables were one-hot boolean indexed to split any feature with n categories into a n column boolean array. Following processing, the training dataset totaled 71 features (**Appendix 2**).

Project Name	Cases (n=441)	Malignancy	Publication status	Pubmed link	DBGAP Accession #
GTB11	98	MDS	In preparation		
Benign Breast Cancer	70	Breast	Submitted		
Traztuzamab	48	Breast	Published	https://www.ncbi.nlm.nih.gov/pubmed/28453704	phs001291.v1.p1
Relapse AML	47	AML	Published	https://www.ncbi.nlm.nih.gov/pubmed/26305651	phs000159.v6.p4.
AML Decitabine	39	AML	Published	https://www.ncbi.nlm.nih.gov/pubmed/27959731	phs000159
Folicular Lymphoma	22	Lymphoma	Published	https://www.ncbi.nlm.nih.gov/pubmed/28064239	phs001229.v1
AML Quads	21	AML	Published	https://www.ncbi.nlm.nih.gov/pubmed/26305651	phs000159.v6.p4.
MDS DECITABINE	20	MDS	Published	https://www.ncbi.nlm.nih.gov/pubmed/27740633	phs000159
SCLC	18	SCLC	In preparation		phs0001049
BKM120	17	Breast	Published	https://www.ncbi.nlm.nih.gov/pubmed/26563128	none in publication
AML Post Transplant Relapse	16	AML	Submitted		
GBM	16	Glioblastoma	In preparation		
GST1	1	Gastrointestinal Stromal Tumor	Case study (no publication)		
GTB19	1	Large Granular Lymphocytic Leukemia	Case study (no publication)		
GTB2	1	Melanoma	Case study (no publication)		
ALL1	1	ALL	Published	https://www.ncbi.nlm.nih.gov/pubmed/27181063	phs001066.v1.p1
AML31	1	AML	Published	https://www.ncbi.nlm.nih.gov/pubmed/26645048	phs000159
CRC1	1	Colorectal	Case study (no publication)		
DLBCL	1	Lymphoma	Case study (no publication)		
LGG1	1	Glioblastoma	Case study (no publication)		
MPNST	1	Malignant peripheral nerve sheath tumor	Case study (no publication)		

Table 4.2 Data Availability of Sequencing Results for all Cases Used in the Machine Learning Classifier Development.

4.3.2 Model Development and Analysis

Logistic regression, random forest, and deep learning were tested as alternate models of somatic variant refinement. A logistic regression classifier was implemented using the keras library (https://github.com/fchollet/keras). Scikit-learn was used to implement the random forest classifier[132]. The random forest was trained using the parameters n_estimators=1000 and trees max_features=8. The deep learning model was implemented using the keras library as a feed-forward network with the input layer equaling the number of features, four hidden layers with 20-node hidden layers, and an output layer equaling the three outputs (‘somatic’, ‘ambiguous’, ‘fail’). The input and hidden layers used a hyperbolic tangent (tanh) activation function, the output layer used a softmax activation function. Categorical cross-entropy was used as a loss function and the Adam optimizer was used over 700 epochs with a batch size of 2,000. L2 regularization was used with a weight of 0.001.

To compare model performance, one-versus-all receiver operator characteristic (ROC) curves were generated and area under the curve metrics (AUC) quantified using scikit-learn[132]. A random subset of two thirds of the 41,000 variant dataset was used as a training set, and the remaining third was withheld as a validation set. Scores for examples in the training set were computed using ten fold cross-validation, so performance on the training set could be used to estimate generalization accuracy[132].

Feature importance for the deep learning model was calculated by training a model on the cross-validation dataset, independently shuffling each of the 71 features, and determining the change in average AUC, by comparing baseline and shuffled performance. The random forest feature importance metric was obtained from scikit-learn’s built in feature_importances_ parameter on a trained random forest model.

4.3.3 Validation of model performance by independent sequencing data.

Orthogonal validation of variant calls can be obtained from a technical replicate of sequencing data obtained independently of the data used by the manual reviewers. To assess the machine learning model’s performance, we evaluated variant calls for a single acute myeloid leukemia case (AML31) that had orthogonal validation sequencing. Ultra-deep whole genome sequencing data (average coverage = 312x) was previously produced for AML31 to evaluate seven different variant callers. Orthogonal custom capture validation sequencing (average coverage = 1000X) was used to validate mutations (n=192,241) identified by any of the seven variant callers61. Variants identified as somatic by orthogonal sequencing (the “Platinum SNV List”) were considered true positives (n = 1,343). Variants that were identified by only one of the seven callers but not validated by ultra deep-sequencing were considered true negatives (n = 190,898). Features were obtained from whole genome bam files for every site that was called by at least one of seven variant callers in the original study and had been selected for targeted re-sequencing (n = 192,241). The random forest and deep learning models (trained using the 41,000 call training dataset) were used to predict calls for each of the sites in the AML31 dataset and receiver operator curve (ROC) figures were used to illustrate model performance.

4.3.4 Independent Test Set

To assess model robustness against batch effects, an independent test dataset was assembled from 4 additional small cell lung cancer (SCLC) paired tumor-normal cases (2,686 variants). This independent test dataset had been sequenced on different instruments (HiSeq 2500 vs HiSeq 4000), utilized different false positive filtering thresholds, and was manually reviewed by different individuals. To test model performance on these data, we trained a deep learning and random forest model on the entire training dataset of 41,000 calls and made predictions for the 2,686 calls of the independent test samples. We assessed the model performance by creating ROC curves and reliability diagrams as outlined above.

4.3.5 Annotations of Clinical Relevance

All variants identified as somatic by either manual somatic refinement or by the classifier (n=21,100) were evaluated for clinical significance. False positives were defined as variants identified as somatic by the manual review pipeline but labeled as ambiguous or fail by the classifier. False negatives were defined as variants identified as ambiguous or fail during manual review but identified as somatic by the classifier. Variants were annotated using two clinical annotation databases (The Database of Curated Mutations (DoCM)[115] and the Clinical Interpretations of Variants in Cancer (CIViC) database[99]). Using the DoCM database, we evaluated exact overlap (chromosome, start, stop, reference base, variant base) between DoCM entries and misclassifications (i.e. false positives and false negatives). To evaluate overlap with the CIViC database, coordinates were queried from the CIViC interface using the public API (http://griffithlab.org/civic-api-docs/). Given that not not all variants within CIViC can be analyzed using whole genome or whole exome sequencing, we used the Sequence Ontology IDs to filter out variants that cannot be analyzed using DNA-sequencing, such as 'increased expression' or 'methylation', etc. (**Table 4.3**). We queried coordinates from the CIViC interface to determine overlap between CIViC annotations and misclassifications, such as therapeutic sensitivity, therapeutic resistance, prognosis, diagnosis, or predisposing, were determined using CIViC evidence statements.

SOID	Description
SO:0001587	stop_gained
SO:0001623	5_prime_UTR_variant
SO:0001578	stop_lost
SO:0001575	splice_donor_variant
SO:0000694	SNP
SO:0001822	inframe_deletion
SO:0001650	inframe_variant
SO:0001821	inframe_insertion
SO:0001583	missense_variant
SO:0001624	3_prime_UTR_variant
SO:0001819	synonymous_variant
SO:0002092	5_prime_UTR_exon_variant
SO:0001909	frameshift_elongation
SO:0001629	splice_site_variant
SO:0001592	minus_1_frameshift_variant
SO:0001992	nonsynonymous_variant
SO:0001566	regulatory_region_variant
SO:0002015	3_prime_UTR_truncation
SO:0002054	loss_of_function_variant
SO:0002053	gain_of_function_variant
SO:0001627	intron_variant
SO:0001564	gene_variant
SO:0001594	plus_1_frameshift_variant
SO:0001017	silent_mutation
SO:0000667	insertion
SO:0001826	disruptive_inframe_deletion
SO:0002052	dominant_negative_variant
SO:0002012	start_lost
SO:0001429	DNA_binding_site
SO:0001820	inframe_indel

Table 4.3 Sequence Ontology IDs used to filter variants within the CIViC database that can be analyzed on DNA-sequencing platforms.

4.3.6 Code and data availability

All analysis, preprocessing code, readcount training data, manual review calls and trained deep learning and random forest models are available on github at https://github.com/griffithlab/manual_review_classifier. Breast cancer cell line (HCC1395) variants and manual review calls are included in the github repo to serve as test data[110]. All plots were produced using the MatPlotlib library[133]. Data availability for all the projects included in this study are outlined in **Table 4.2**.

4.4 Results

4.4.1 Data assembly and standardization

The 41,000 variants used to train the model were derived from 440 individual tumors spread across 9 cancer types. The sequencing approach used to assess tumor genetics was evenly split between capture sequencing (14,234 variants), exome sequencing (14,004 variants), and whole genome sequencing (12,722 variants). Among all manually reviewed variant calls, 18,381 were confirmed as somatic, 10,643 were assessed as ambiguous, 8,854 as failed, and 3,122 as germline. The training data has representation of both hematopoietic (10,583 variants) and solid tumors (30,417 variants), which often have distinct characteristics during manual variant refinement (**Table 4.1)**.

4.4.2 Machine Learning Model Development and Analysis

We developed three models (logistic regression, random forest, and deep learning) using the 41,000 call data set. To test against overfitting, we randomly selected one third of the dataset as a holdout test set and used the remaining two thirds as a training set in a 10-fold cross-validation strategy for model development. In the 10-fold cross-validation, all three models (logistic regression, random forest, and deep learning) achieved better than random performance on the classification of somatic variants. The logistic regression model demonstrated the worst performance, indicating that a linear separator is insufficient for adequate classification (average AUC=0.89). Specifically, it showed limited ability to classify ambiguous calls (AUC=0.79), which is the most poorly defined of all the variant classes. However, both random forest and deep learning models performed very well across all classes attaining an average AUC of 0.98

and 0.96 respectively (**Figure 4.2A**). The random forest and deep learning models perform similarly across all three classes. The holdout test set also performed similarly (average AUC = 0.97) to the prior 10-fold cross validation (**Figure 4.3A**).

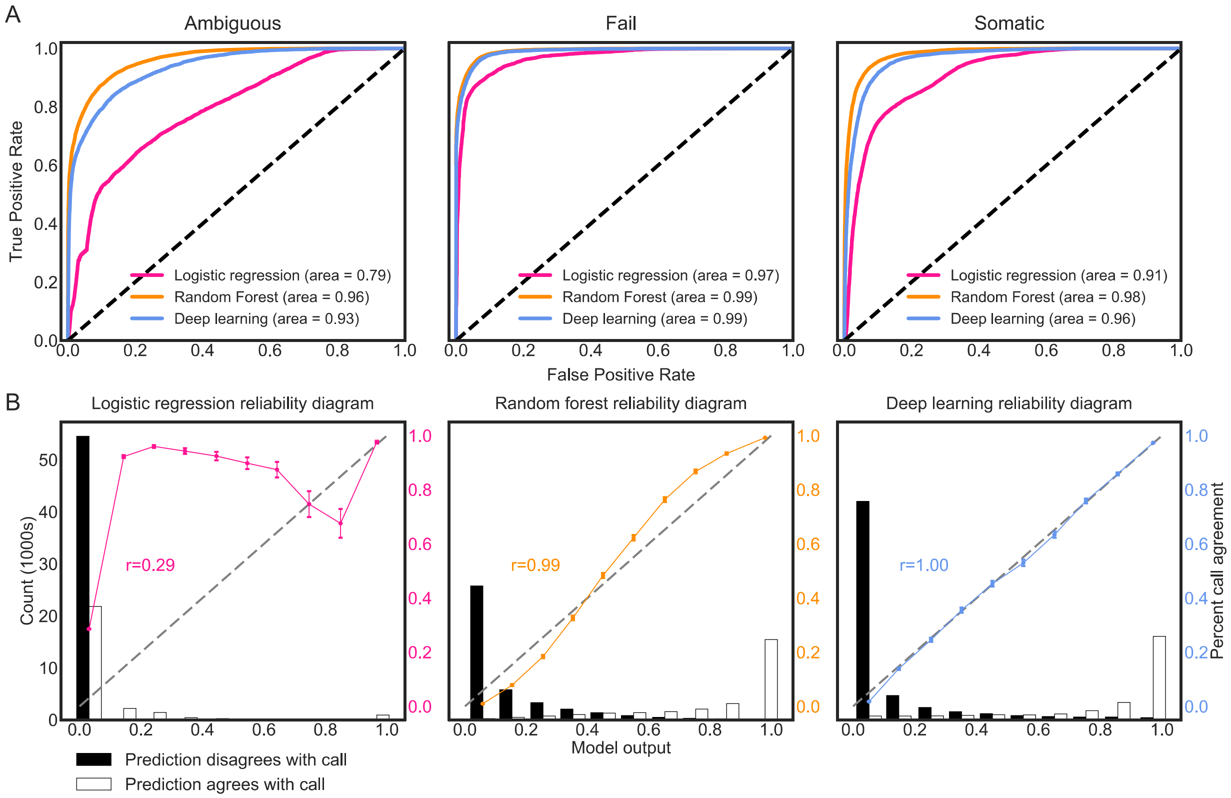


Figure 4.2 Deep learning and random forest method achieve excellent classification performance on manual review calls*. (A) Comparison of performance of various machine learning models via receiver operating characteristic (ROC) curve for the three classification classes in the cross-validation data. The diagonal line indicates baseline performance. The deep learning and random forest classifiers perform approximately identically. (B) The deep learning and random forest classifier outputs are well scaled to a probability (between 0 and 1). The bar graphs plot the distribution of model output that agrees with the manual review call versus output that disagrees with the call in 10 equally distributed bins of model output. The diagonal line indicates a perfectly scaled probabilistic prediction. The colored points display the ratio of predictions that agree with the call to the total number of predictions for a given bin. Binomial proportion confidence intervals were calculated for each bin. Pearson's correlation coefficient comparing colored points to the diagonal line was calculated to assess the output of the respective model.*

Reliability diagrams were used to determine if model outputs could be interpreted as a probability of a variant call. Model output, which was a continuous value, was plotted for 10 equally distributed bins that were separated by whether the model's output matched or did not match the manual variant refinement call. For each bin, we calculated the ratio between the number of sites where the model agrees with the call and the total number of sites in the bin. It is expected that if the model output estimates a well-scaled probability, then the calculated ratio will be correlated to an identity line (x=y). Pearson’s correlation coefficient was used to test for a well-scaled probability using the scipy.stats.pearsonr function[134]. Comparing the reliability diagrams for each model indicated that the random forest model and the deep learning model produced outputs that are most closely scaled to a probability. The random forest model and the deep learning model obtained a pearson’s correlation coefficient (r) of 0.99 and 1.00, respectively (**Figure 4.2B**). The logistic regression model output was most divergent from a well-scaled probability with r=0.29. When reliability diagrams are plotted independently for each class (somatic, ambiguous, and fail), all classes produce well-scaled output (**Figure 4.4).**

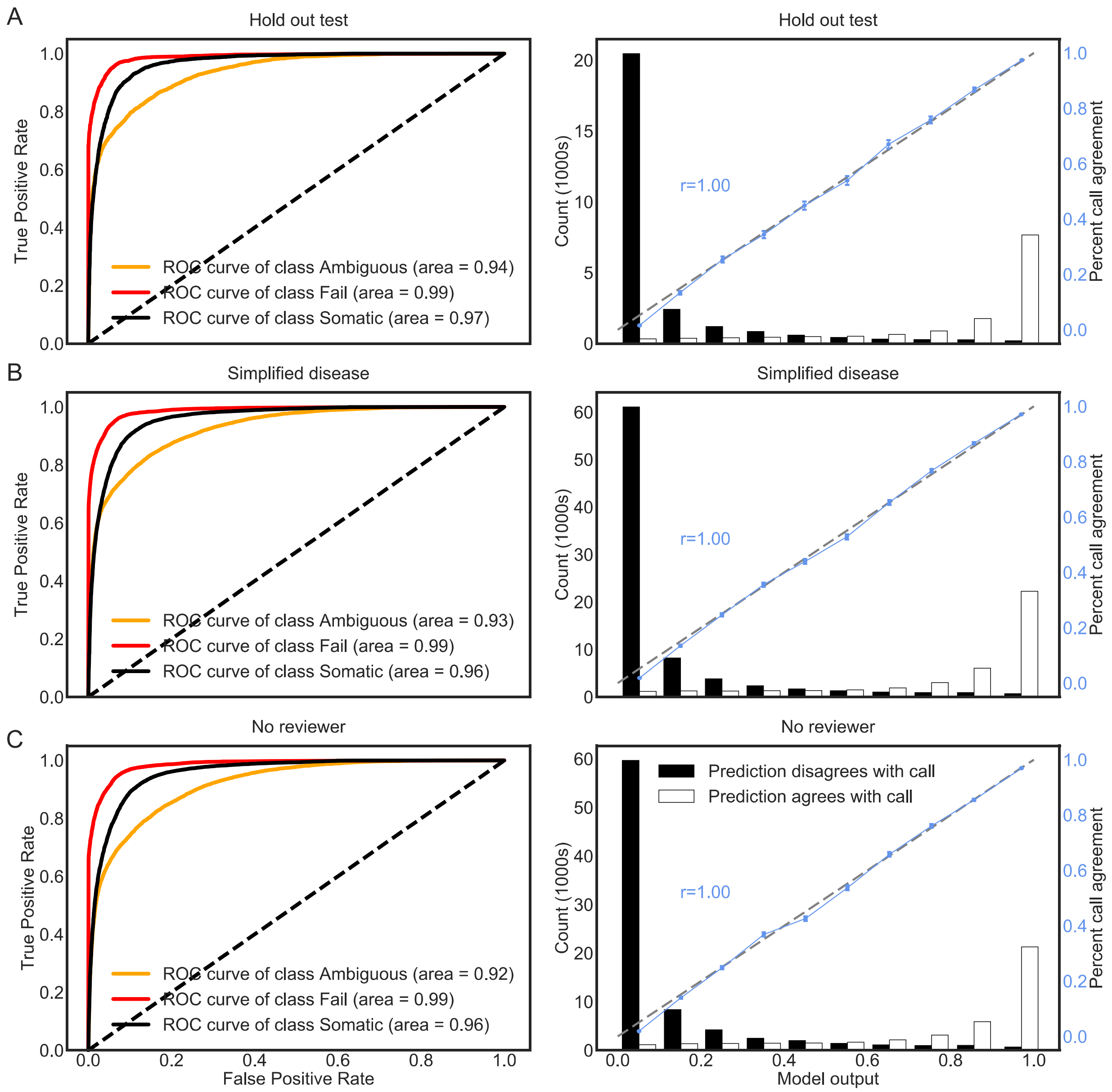


Figure 4.3 Deep learning manual review classifier performs well on hold out test data, and on cross validation results with a simplified disease feature and the reviewer removed. *ROC curve and reliability diagram performance of the deep learning classifier with the cancer type feature distilled down to solid tumor status. Performance of deep learning model on a ⅓ holdout test set of the original training dataset.*

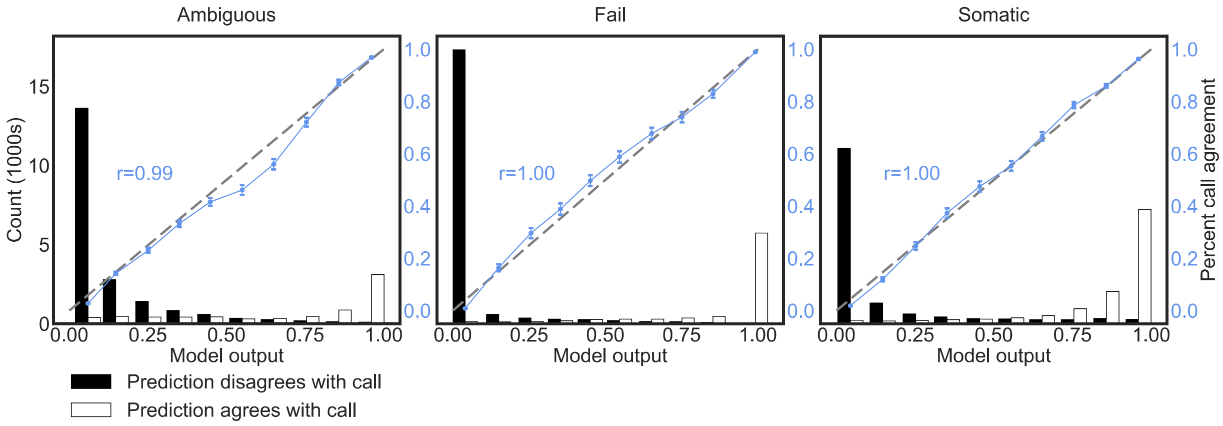


Figure 4.4 ***Deep learning model outputs are well scaled across all predicted classes.*** *The correlation between the model output and the manual review call to a well scaled probability was assessed for all three different classes of calls: ambiguous, fail, and somatic. The probabilities for each class were binned into 10 classes ranging from 0.00-1.00. For each bin, the total number of manual review calls that agree and disagree with the individual class were plotted. The ratio of agreement to disagreement was plotted for each bin and compared to the identity line (x=y) using the Pearson's correlation coefficient (r).*

4.4.3 Feature Importance

Feature importance analysis was used to determine which features each model was relying upon to make predictions. The deep learning model ranked feature importance using the average change in the AUC after randomly shuffling individual features, and the random forest model using the built-in feature importance metric. To assess how manual reviewers rank the importance of different features, seven manual reviewers at our institute were asked to rank their top 15 (of 71) features, that were most important in their manual review decision-making process. The average importance of features to the reviewers was used as a comparison to the model feature rankings. All three importances were rank normalized for comparison. Comparison shows that the models rely on many features that expert manual reviewers also use to make classification decisions (**Figure 4.5**). The random forest feature importance was moderately correlated to the deep learning and manual reviewer feature importance (Pearson's r = 0.47 and 0.48 respectively). The deep learning importance was only weakly correlated with manual reviewer survey results (pearson's r=0.11). Of note, both the random forest model and the deep learning model ranked reviewer higher than reviewers themselves ranked this feature. Similarly, cancer type was ranked in their top 5 most important features of each model, however, this was not ranked heavily by manual reviewers.

We hypothesized that most of the value of the cancer type feature is mediated by differences between liquid and solid tumors. This is because, the normal sample of hematological cancers is more likely to be contaminated by cancer cells than in solid tumors and requires special consideration. This contamination ultimately increases the risk that a somatic variant will be mis-called. To test this hypothesis, we collapsed the cancer type features to a single solid/liquid boolean. Comparison of model performance with parsed tumor types to model performance with simplified tumor type (solid/liquid) showed similar performance (**Figure 4.3B**). This supports our hypothesis that the distinction between liquid and solid tumors is important for calling variants accurately and individual disease type is not as important beyond this key difference.

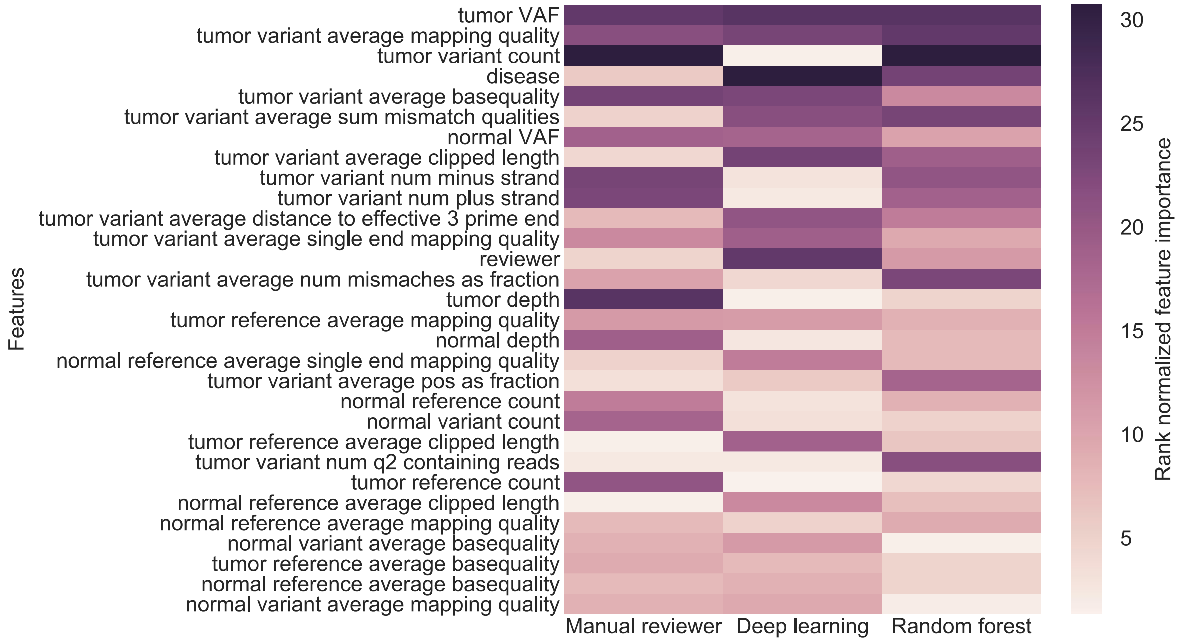


Figure 4.5 Deep learning and random forest models use similar features to manual reviewers in making manual review classification decisions. *Most features ranked as important by RF and deep learning models agree well with features that human manual reviewers rank highly for classification decisions. Human manual reviewer feature importance was determined by asking 7 individuals who have contributed to manual review data to rank feature importance. Single feature impact for the deep learning model was obtained by training a model on the entire training dataset then shuffling each feature individually and calculating the mean ROC AUC for all three classes. The change in mean ROC AUC for all classes was sorted and plotted. Random forest feature importance was obtained via scikit-learn's feature importance parameter. The random forest feature importance is moderately correlated to the deep learning and manual reviewer feature importance (pearson's r= 0.47 and 0.48 respectively). The deep learning importance was weakly correlated with manual reviewer survey results (pearson's r=0.11). All feature importance metrics were ranked normalized. The top 30 most important features (average rank), of the three metrics, are shown.*

4.4.4 Inter-reviewer Variability

Reviewer identity was also an important feature, among the top 30 features, which indicates reviewer-specific patterns in manual review. Consequently, we studied the inter-reviewer variability. To quantify the variability between manual reviewers we had three independent reviewers call a random subset of 176 sites from the training dataset. This resulted in 3 independent review calls for each of the 176 variants. The calls were randomly chosen from the training cohort and represent various malignancies. Reviewers achieved fair agreement with a Fleiss' Kappa statistic of 0.37, which is a statistic that lies between -1 and 1 where a Kappa statistic below 0 indicates poor agreement and above 0 indicates good agreement. When evaluating all calls in the inter-reviewer variability analysis, 77.3% of all calls showed reasonable agreement (i.e. all three reviewers agree on the call or two reviewers agree on the call and the third reviewer labels the call as ‘ambiguous’) (**Figure 4.6A**). Model performance was correlated with reviewer agreement such that when all three reviewers call a variant as ‘somatic’, the model produces a high ‘somatic’ probability (output above 0.8). Conversely, when all reviewers agree that a call is ‘fail’ the model produces a low somatic probability (below 0.2). As expected, in times of inter-reviewer disagreement, the model produces a wider distribution of ‘somatic’ probabilities (**Figure 4.6B-C**). Together, these results indicate that there is as much as 22.7% disagreement among reviewers, especially on ambiguous calls.

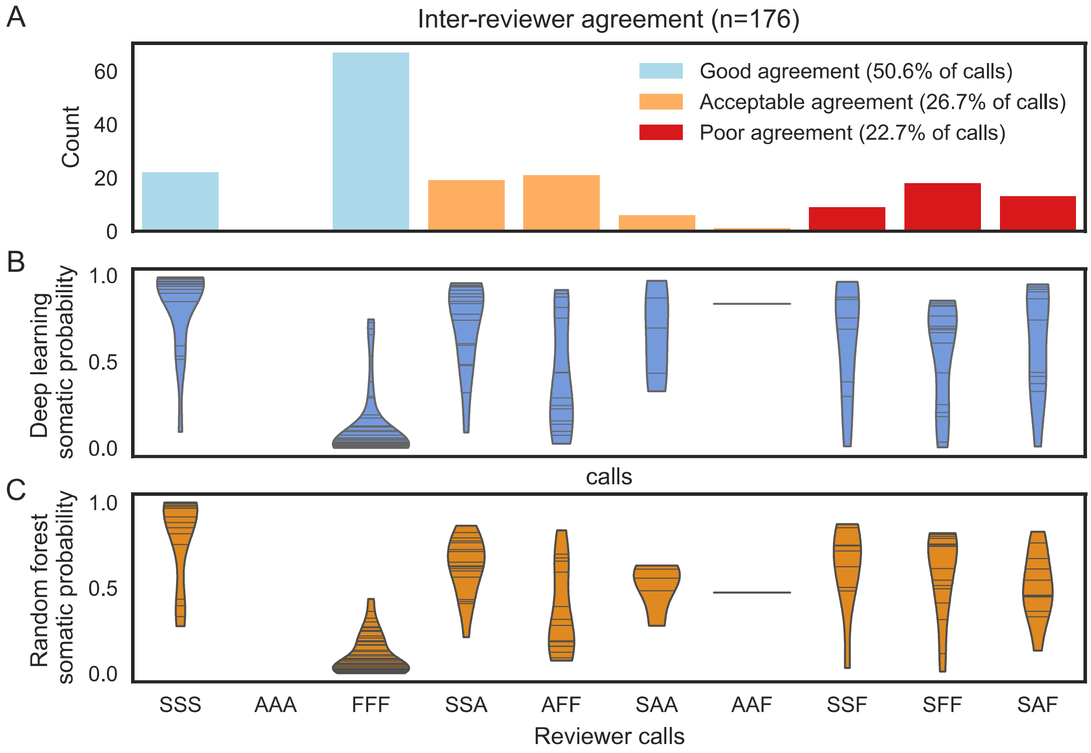


Figure 4.6 Manual reviewers exhibit varied inter-reviewer agreement; however, model confidence closely parallels reviewer confidence (i.e. when all reviewers agree the model give high probabilities, when reviewers disagree the model assigns lower probabilities). *(A) The binned agreement of 3 reviewers on 176 variants. The x-axis outlines all possible permutations of agreement among three reviewers. The y-axis outlines the frequency of each group of agreement. 'S' denotes a somatic call, 'A' denotes ambiguous, and 'F' denotes a fail call. 'SSS' is the case where all three reviewers call the same variant somatic and the other permutations follow a similar pattern (e.g. 'SAF'= somatic, ambiguous, fail). It is considered 1) good agreement when all three reviewers agree, 2) acceptable agreement when reviewers only disagree between ambiguous and somatic or ambiguous and germline calls, 3) and poor agreement when one reviewer gives a somatic call another fail on the same variant. Problematic sites are where one reviewer calls a variant somatic while another calls a variant fail. (B) Violin plots of deep learning somatic probability, the horizontal lines indicate the occurrence of a probability and the width indicates the distribution of probabilities. (C) Violin plots of random forest somatic probability.*

Variant calls that do not depend on reviewer identity are most desirable for reproducibility, and to reduce the impact of idiosyncratic criteria. Previous analysis used 71 features, whereby one feature was the individual who performed manual review. New models were developed after removing the reviewer feature from the training data to assess performance in situations when the reviewer is unknown. The deep learning model with all 71 features resulted in an average AUC of 0.960 compared to an average AUC of 0.956 with the reviewer feature removed. This experiment illustrates expected performance on de novo data that does not include a reviewer feature (**Figure 4.4C**).

4.4.5 Validation of model performance by independent sequencing data.

All the variant sites in our training and independent test datasets were obtained through matched tumor-normal genomic sequencing, variant calling via statistical somatic callers, various filtering strategies, and finally manual variant refinement. To understand how our model performs on unfiltered "raw" variant calls, we ran our model on 192,241 putative somatic variants called by one of seven variant callers in a rich sequencing dataset described by Griffith et al.[61]. In addition to the ultra-deep whole genome sequencing available in this study for discovery, orthogonal custom capture validation sequencing was performed on all 192,241 sites. Sites validated by the custom capture data were considered positives and those that failed considered negatives. Comparing somatic model predictions from the random forest model and the deep learning model to validation sequencing results achieved a ROC AUC of 0.96 and 0.95, respectively (**Figure 4.7**).

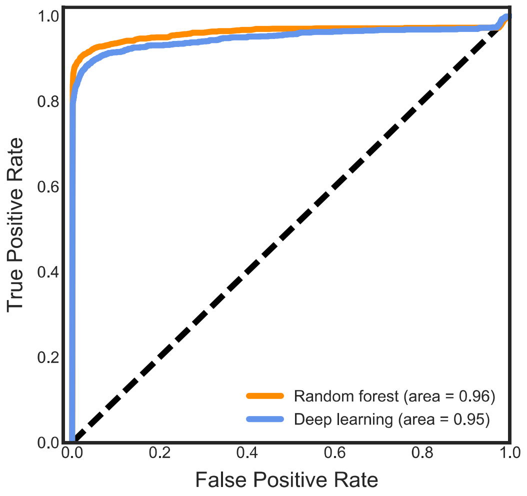


Figure 4.7 Deep learning and random forest classifiers accurately predict orthogonal validation sequencing results*. Across 192,241 putative somatic variants identified by one of 7 variant callers in a single AML case with 300x whole genome sequencing data the random forest classifier accurately validation obtained from 1000x orthogonal validation sequencing data.*

4.4.6 Independent Manual Review Data

To test model performance on independent data, we obtained a sample set that was sequenced independently, underwent a different filtering strategy, and was reviewed by new manual reviewers. This dataset had 2,686 manual review calls and genomic data from 4 paired tumor-normal small cell lung cancer (SCLC) cases. This dataset had a different distribution of call classes (94.1% somatic, 5.4% fail, and 0.5% ambiguous) when compared to the training cohort (44.8% somatic, 29.2% fail, and 26% ambiguous). The independent test dataset had an average AUC of 0.70 for random forest and 0.81 for deep learning (**Figure 4.8A**). We tested ways to overcome batch effects between SCLC tumors in the training set and SCLC tumors in the independent test set. To overcome the apparent batch effect, we randomly selected manual review calls from the independent test set in 5% increments from 0% to 75% to include in training the model. We then tested the performance of the newly trained model on the remaining manual review calls from the independent test set. For the deep learning model and the random forest model, model performance was restored to levels observed in cross-validation after inclusion of approximately 250 manual review calls (**Figure 4.8B**).

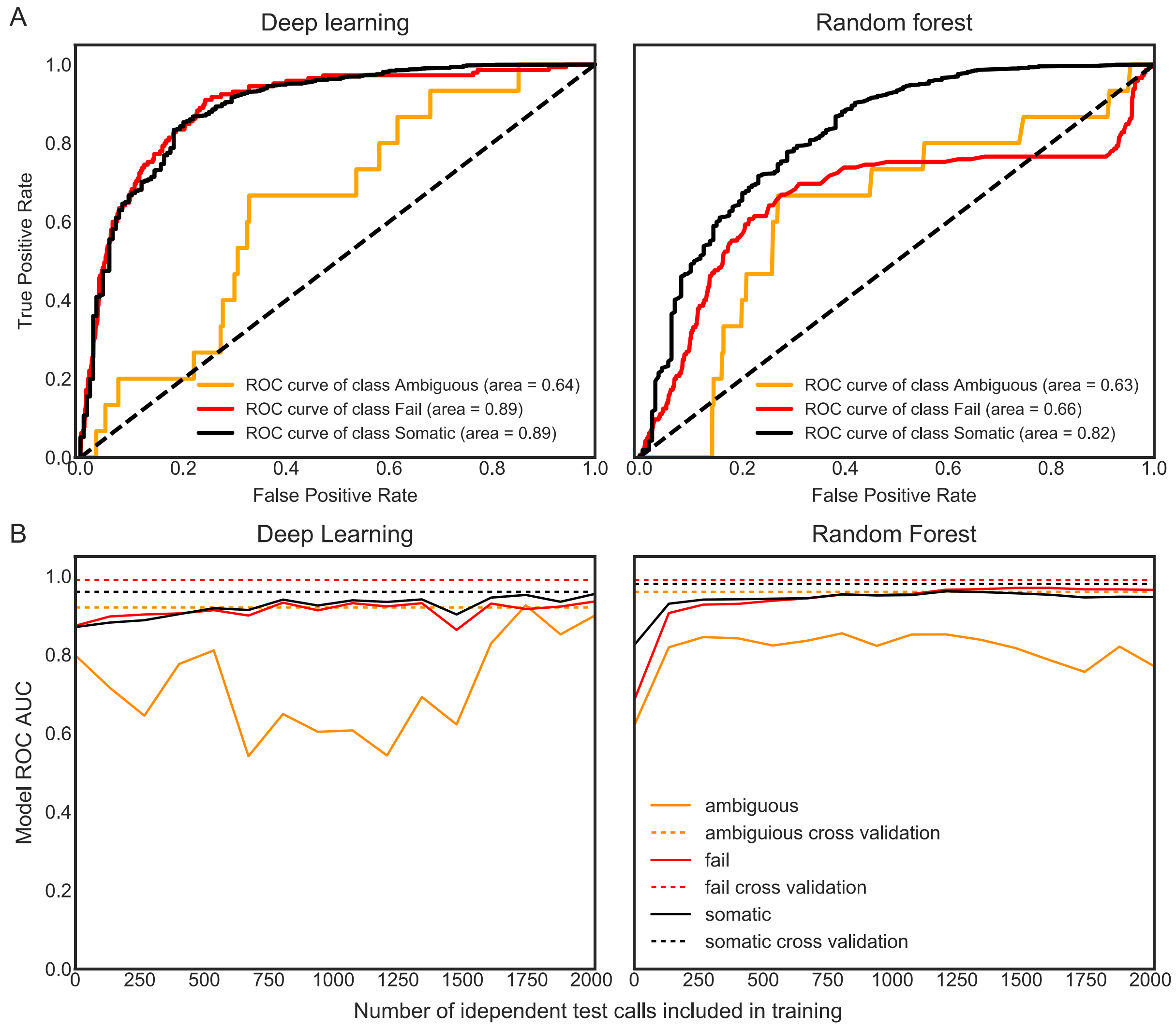
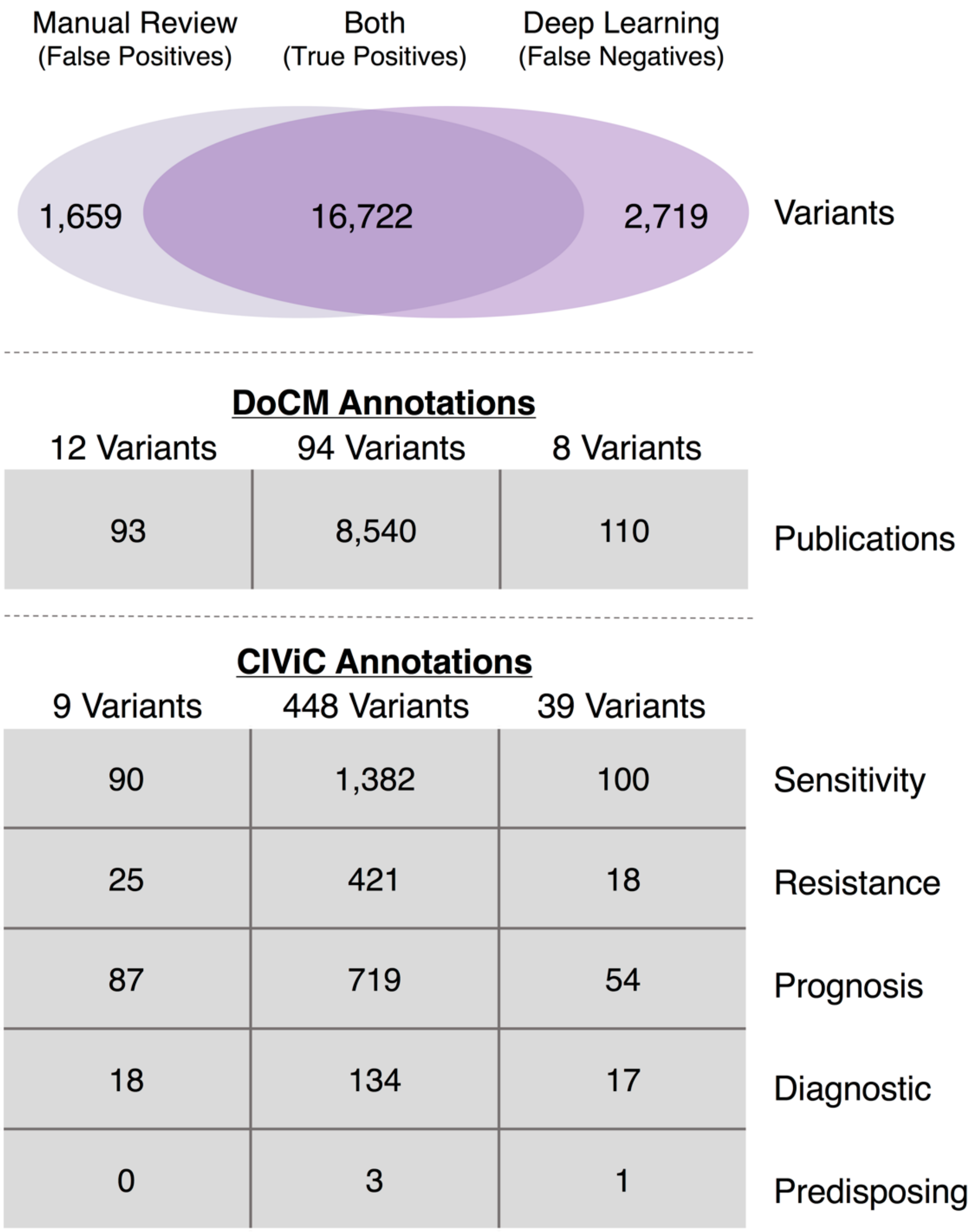


***Figure 4.8 The deep learning classifier performs better than the random forest on validation sequencing data**. (A) ROC curves outlining deep learning and random forest model performance on the independent test dataset consisting of 4 SCLC cases with 2,686 variants. (B) Independent test set batch effect correction. Independent test set data was partitioned in random stratified increments of 5% from 0-75% of the total independent test dataset size and used to train a new model. The x-axis outlines the number of test variants included in training. The y-axis plots the resulting model's ROC AUC. The ambiguous class shows significant stochasticity due to low representation in the test dataset (n=15).*

4.4.7 Recovery of Clinically Actionable Variants

The deep learning model was used to assess if machine learning algorithms for variant analysis could improve detection of true clinically actionable mutations mislabeled by traditional variant post-processing strategies. Of the 21,100 variants identified as somatic by either the deep learning model or by manual variant refinement, there were 16,722 variants that were called as somatic by both methods, 1,659 variants called as somatic only by manual variant refinement (i.e. false positives), and 2,719 variants that were called as somatic only by the deep learning model (i.e. false negatives) (**Figure 4.9**).

Mislabeled variants were evaluated for biological importance using the Database of Curated Mutations (DoCM). DoCM is a database that outlines literature supported biologically important and clinically relevant variants. Of the 2,719 false negatives identified by the deep learning model, 8 variants had an exact match with an annotation in DoCM. There were 110 total publications associated with these variants with an average of 13-14 publications per variant. The average model output for the somatic class of all 8 variants identified in DoCM was 0.877. Of the 1,659 false positives identified by the deep learning model, 12 variants had an exact match with an annotation in DoCM. There were 93 publications associated with the variants. The average model output for the somatic class of all 12 false positives identified in DoCM was 0.256 (**Figure 4.9** and **Table 4.4**). The classifier could properly call several clinically actionable variants, as determined by DoCM, that had been improperly labeled by manual review.



DoCM Annotations

12 Variants	94 Variants	8 Variants	
93	8,540	110	Publications

CIViC Annotations

9 Variants	448 Variants	39 Variants	
90	1,382	100	Sensitivity
25	421	18	Resistance
87	719	54	Prognosis
18	134	17	Diagnostic
0	3	1	Predisposing

***Figure 4.9 Manual review misclassification rectified by the deep learning model**. The venn diagram illustrates variants identified as somatic by only the manual review model (false positives), only the deep learning model (false negatives), and by both pipelines (true positives). For these three groups, we showed the number of variants that have direct overlap with both DoCM Annotations and CIViC annotations. For DoCM annotations, we listed the total number of publications associated with with all variants within each group. For CIViC annotations, we listed the total number of evidence items associated with all variants within each group. These evidence items are parsed by those that convey variant sensitivity to a drug, variant resistance to a drug, variant that confers better or worse prognosis, variant that confers disease diagnosis, and variant shows predisposing evidence for disease. Re-review of the 53 false positive somatic variants with CIViC annotations showed that 44 were erroneously called somatic by original manual reviewers. Similarly, re-review of the 40 false negative somatic variants with CIViC annotations showed that 39 variant were erroneously failed original manual review.*

A

Deep Learning Classifier										
Chr	Start	Stop	Ref	Var	Gene	Disease	Probability	MR Call	Classifier	Publications
1	115258747	115258747	C	G	NRAS	AML	0.89069724	f	s	19
2	209113112	209113112	C	T	IDH1	AML	0.83565396	f	s	23
1	115258747	115258747	C	T	NRAS	AML	0.93176877	a	s	26
11	119148891	119148891	T	C	CBL	AML	0.93619579	a	s	1
3	178936091	178936091	G	A	PIK3CA	breast	0.93707067	a	s	23
3	178936103	178936103	G	A	PIK3CA	breast	0.94812727	a	s	12
2	25457243	25457243	G	A	DNMT3A	AML	0.86303806	a	s	7
2	209113113	209113113	G	A	IDH1	AML	0.67307019	a	s	12

B

Random Forest Classifier										
Chr	Start	Stop	Ref	Var	Gene	Disease	Probability	MR Call	Classifier	Publications
3	178916930	178916930	G	T	PIK3CA	SCLC	0.544	a	s	1
1	115258747	115258747	C	G	NRAS	AML	0.833	f	s	19
2	209113112	209113112	C	T	IDH1	AML	0.936	f	s	12
1	115258747	115258747	C	T	NRAS	AML	0.778	a	s	26
11	119148891	119148891	T	C	CBL	AML	0.857	a	s	1
3	178936091	178936091	G	A	PIK3CA	breast	0.92	a	s	23
3	178936103	178936103	G	A	PIK3CA	breast	0.927	a	s	12
2	25457243	25457243	G	A	DNMT3A	AML	0.612	a	s	7
2	209113113	209113113	G	A	IDH1	AML	0.745	a	s	10

Table 4.4 There was a high level of exact overlap between false negatives recovered by the machine learning classifiers and DoCM annotations. *For each table, the 'Probability' is the output of the classifier, the 'MR call' is the call made by the human manual reviewer, the 'Classifier' is the call made by the machine learning model, and the 'Publications' is the total number of publication hits within the DoCM database. (A) This table shows false negative calls derived from the Deep Learning classifier that had exact overlap (chr, start, stop, ref, var) with DoCM annotations. (B) This table shows false negative calls derived from the Random Forest classifier that had exact overlap (chr, start, stop, ref, var) with DoCM annotations.*

Mislabeled variants were also evaluated for clinical relevance using the Clinical Interpretations of Variants in Cancer database (CIViC)[99]. CIViC is a freely accessible database that promotes community-driven expert curation of clinically relevant variants in cancer. Each annotation within CIViC is based on evidence summaries that detail therapeutic, prognostic, predisposing, or diagnostic implications in cancer. Based on filtering using the sequencing ontology IDs (**see section 4.3**) to remove annotations that could not be analyzed using DNA sequencing, there were 425 clinically relevant CIViC annotations. Of the false negatives identified by the deep learning model, 40 variants were actionable with 100 evidence items related to therapeutic sensitivity, 18 evidence items related to therapeutic resistance, 54 evidence items detailing prognostic information, 17 indicating diagnostic information, and 1 evidence item that supported predisposition for cancer. Of the 1,659 false positives that were reclassified as 'fail' by the deep learning model, there were 53 clinically relevant variants that had overlap with CIViC. Of these false positives identified by the deep learning model, there were 90 evidence items related to therapeutic sensitivity, 25 evidence items related to therapeutic resistance, 87 evidence items detailing prognostic information, and 18 illustrating diagnostic information (**Appendix 3**). There are, therefore, several clinically actionable variants from the CIViC database that were mislabeled by manual review and rectified by the model.

Retrospective review of these mislabeled variants in Integrative Genomic Viewer (IGV)[26] confirmed confidence in model predictions. When re-reviewing the 53 false positives, 83% (44/53) were determined to be miscalls by the original manual reviewer. Four examples of manual review miscalls are shown in **Figure 4.10**. Of the 9 variants whereby the original manual review was deemed correct and the model was incorrect, 6 were small insertions, indicating a potential issue with the model's ability to properly label small insertions as somatic. When re-reviewing the 40 false negatives, (39/40) 97.5% appeared to be high quality somatic cells erroneously called ambiguous or fail by the original reviewer. In one example, 2 clinically relevant PIK3CA mutations were missed due to the manual reviewer assuming that two adjacent variants on the same strand were considered multiple mismatches (**Figure 4.11**). In another example a TP53 mutation was missed in an acute myeloid leukemia case due to the manual reviewer's lack of recognition that hematologic cancers typically have tumor contamination in normal tissue (**Figure 4.11**). According to the re-review of false positive and false negative calls, the model is more sensitive and specific than the data on which it was trained.

The manual variant refinement workflow identified 92.1% of all eligible clinically actionable mutations. In contrast, the deep learning model identified 98.2% of all possible clinically relevant variants. Therefore, the model was able to recover 6.1% more variants than a traditional manual approach. Similarly, 8.9% of variants (n=44) were false positives via manual variant refinement whereas 0.2% (n=1) false positives were reported using the model. Therefore, we predict that model use would reduce total number of misreported variants to physicians by as much as 8.7%. This result suggests that automated models of variant refinement might ensure more accurate and reproducible analysis of cancer genome data, alleviating a key bottleneck in analysis.

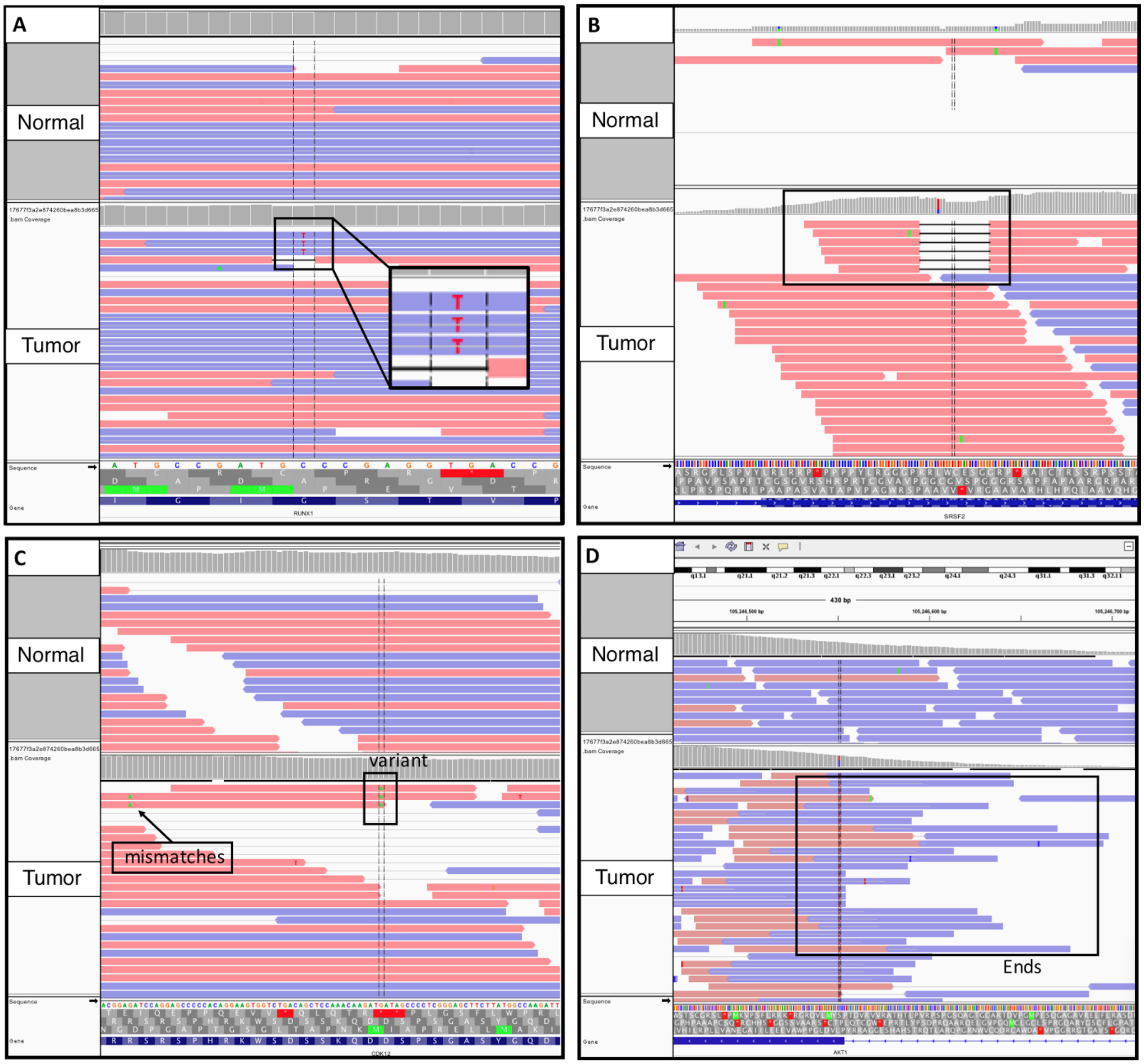
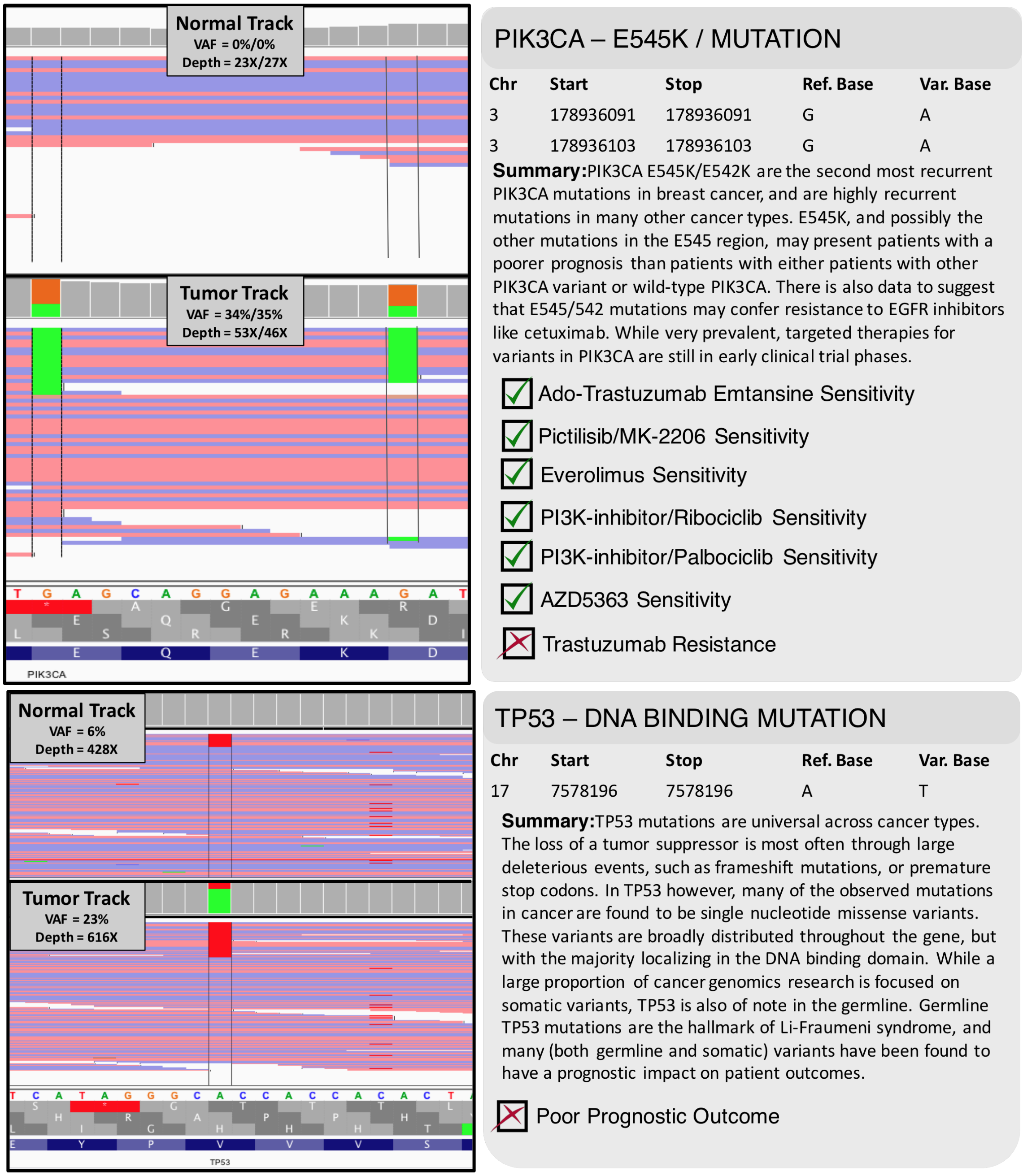


Figure 4.10 IGV snapshots of clinically relevant false positives that were mislabeled during manual review as somatic but re-identified as failed using the Deep Learning classifier*. A. Failure due to short inserts and directional artifacts. B. Failure due to directional artifacts C. Failure due to adjacent mismatches and directional artifacts. D. Failure due to ends artifact.*



PIK3CA – E545K / MUTATION

Chr	Start	Stop	Ref. Base	Var. Base
3	178936091	178936091	G	A
3	178936103	178936103	G	A

Summary:PIK3CA E545K/E542K are the second most recurrent PIK3CA mutations in breast cancer, and are highly recurrent mutations in many other cancer types. E545K, and possibly the other mutations in the E545 region, may present patients with a poorer prognosis than patients with either patients with other PIK3CA variant or wild-type PIK3CA. There is also data to suggest that E545/542 mutations may confer resistance to EGFR inhibitors like cetuximab. While very prevalent, targeted therapies for variants in PIK3CA are still in early clinical trial phases.

☑ Ado-Trastuzumab Emtansine Sensitivity

☑ Pictilisib/MK-2206 Sensitivity

☑ Everolimus Sensitivity

☑ PI3K-inhibitor/Ribociclib Sensitivity

☑ PI3K-inhibitor/Palbociclib Sensitivity

☑ AZD5363 Sensitivity

☒ Trastuzumab Resistance

TP53 – DNA BINDING MUTATION

Chr	Start	Stop	Ref. Base	Var. Base
17	7578196	7578196	A	T

Summary:TP53 mutations are universal across cancer types. The loss of a tumor suppressor is most often through large deleterious events, such as frameshift mutations, or premature stop codons. In TP53 however, many of the observed mutations in cancer are found to be single nucleotide missense variants. These variants are broadly distributed throughout the gene, but with the majority localizing in the DNA binding domain. While a large proportion of cancer genomics research is focused on somatic variants, TP53 is also of note in the germline. Germline TP53 mutations are the hallmark of Li-Fraumeni syndrome, and many (both germline and somatic) variants have been found to have a prognostic impact on patient outcomes.

☒ Poor Prognostic Outcome

Figure 4.11 Manual visualization for two clinically relevant variants recovered by classifiers using IGV showed evidence of mislabeling by manual reviewer. *Within IGV snapshots, the normal tracks and the tumor tracks show aligned reads that were obtained from normal tissue and the tumor tissue, respectively. Variant loci are bracketed by horizontal grey bars. Variant summaries obtained from CIViC show gene name, variant type, variant coordinates, clinical summary, and relevant clinical action items. (A) Reviewer conservatively labeled both PIK3CA variants as ambiguous due to multiple mismatches in reads, however, both variants appear to be somatic. (B) Evidence of tumor variant reads in normal track can be attributed to tumor derived from patient with AML.*

4.5 Discussion

The random forest and deep learning models achieved excellent classification performance with comparable performance across all variant refinement classes (somatic, ambiguous, and fail), whereas the logistic regression model achieved poorer performance, particularly with the ambiguous class. High performance of model predictions on cross validation data and hold-out test set confirms the ability of an automated strategy to dramatically reduce the need for manual variant refinement. Our ability to virtually eliminate manual review was further demonstrated by removal of the manual reviewer as a feature for training the model. Attaining high performance with and without reviewer information provides the ability to extrapolate the trained model onto new data with an unknown reviewer (**Figure 4.3C**). Through our inter-reviewer analysis, we elucidated that inter-reviewer agreement can be as low as 50% (**Figure 4.6A**), therefore, the use of a machine learning algorithm for automated somatic variant refinement should also produce a more reproducible and consistent result.

Suggesting a path to further improvements in cancer genomics pipelines, high accuracy on orthogonal validation data suggests that the machine learning models might even reduce the need for validation sequencing. Both the deep learning and random forest models showed high accuracy in an independent test sample for classifying the validated Platinum Level SNVs outlined by Griffith et al.[61]. Given a high level of sequencing data (300X coverage), we are confident that the machine learning model classifies variants with a high accuracy even without performing any manual review to train the model. Future investigation could explore down-sampling reads to show if decreased variant signal alters model performance. While the performance of the deep learning model on the independent test data (SCLC samples) was less than the internal cross validation, the random forest model performed noticeably poorer. This decrease in performance could be due to batch effects, reviewer differences, or model overfitting. However, introduction of as few as 250 calls from the independent test set in training is a viable option to recover high performance (**Figure 4.8B**). We recommend manually reviewing a small subset of variants called via statistical variant callers (e.g. 200-1000) to evaluate model performance on new datasets. This will elucidate if the existing model has decreased performance due to potential batch effects. If necessary, model performance can be restored by including a small subset of manually reviewed variants into the training data and retraining the model.

These results together show that a machine learning model can be effectively used to automate variant refinement. Automation of variant refinement allows for standardization and systematization of identifying putative somatic variants. This decreases the human variability associated with any manual process and increases the reproducibility of variant calling. Additionally, automation of variant refinement eliminates a labor bottleneck, and its associated costs, allowing any number of somatic variant calls to be evaluated in a negligible amount of time. Finally, since the model offers probabilistic output, an economic framework can be used to set thresholds for confirmatory follow up testing allowing investigators to optimize experimental design to improve accuracy within budgetary constraints [135].

To illustrate the extent of this advance, compared to a standard cancer genomics analysis workflow, for a hypothetical whole genome breast cancer study with 100 tumor/normal paired cases, there would be approximately 2,300,000 variants identified via somatic mutation callers. Following a similar filtering strategy to a recent study in our group, these variants could then be filtered to a set of approximately 25,000 variants needing manual variant refinement after applying standard false positive filters[129]. This equates to about 360 person hours of manual variant refinement, if a typical reviewer processes 70 variants per hour. A machine learning based automated approach would essentially eliminate this manual labor burden. To ensure that the model's training data is optimized for the hypothetical breast cancer data and corrected for any associated batch effects (**Figure 4.8B**), 1000 randomly selected variants could be manually reviewed, resulting in ~14 person hours of manual review. In this hypothetical example the manual review burden would be reduced from 360 hours to ~14, a major improvement 25 times more efficient.

It is apparent that automated models perform as well or better than manual review for somatic variant refinement. Employing the model, relative to manual review, allowed for a 5.0% increase in variants reliably detected and 6.1% increase in identification of known clinically relevant variants. Using the automated approach can thus recover therapeutic opportunities, elucidate potentially resistant therapeutics, predict prognosis and stratify relapse risk. For example, **Figure 4.11** showed a missed TP53 mutation in a patient with acute myeloid leukemia. Based on evidence outlined in CIViC, this TP53 mutation suggests poor prognosis, which could be important information when making treatment decisions for patients with AML. Given that variant interpretation is one of the main barriers preventing genomic analysis for clinical workflows[76], the ability to reduce manual effort while improving the accuracy in detecting clinically relevant variant calls could facilitate integration of genomics data into clinics.

This model does have some limitations. Since manual variant refinement calls are far from the gold standard of somatic variant identification, the training data likely contains a substantial amount of noise due to inter-reviewer variation, as shown in our analysis. In an ideal scenario, highly accurate and orthogonal validation sequencing would be performed to determine somatic variant status. Unfortunately, validation sequencing has a large monetary and tissue material expense, limiting our ability to use this type of data in the training set. Lastly, while the training data was produced using a varied array of capture strategies, libraries, and Illumina sequencing instruments, the model will likely require evaluation and some amount of retraining for non-Illumina sequencing instruments and divergent somatic variant analysis pipelines. It is also possible that the model has learned various other institutional batch-effects from our sequencing and analysis workflows. However, our results suggest that retraining with a small amount of supplemental calls from an independent dataset may be sufficient to overcome these effects.

Finally, while this study demonstrates impressive performance on somatic variant classification there is room for improvement with the use of additional genomic and sequencing features such as proximal sequence complexity (e.g., presence of repeat regions), functional prediction (e.g., conservation based variant impact scores), and other indicators associated with false positives. Additionally, this problem is an ideal candidate for implementing active learning[136], that could continuously update and improve the model on low performing cases. The assembly of a large gold standard somatic variant dataset that has validation sequencing results and pan cancer representation could yield a substantial improvement in somatic variant refinement performance over state-of-the-art filtering and manual review approaches.

4.6 Authors and Contributions

Benjamin J. Ainscough[1,2], Erica K. Barnell[1], Malachi Griffith[1,2,3,4], Thomas E. Rohan, Ramaswamy Govindan[2,3], Elaine R. Mardis[5,6,] S. Joshua Swamidass[7]*, Obi L. Griffith[1,2,3,4]*

[1] McDonnell Genome Institute, Washington University School of Medicine, St. Louis, MO, USA.
[2] Siteman Cancer Center, Washington University School of Medicine, St. Louis, MO, USA.
[3] Division of Oncology, Department of Medicine, Washington University School of Medicine, St. Louis, MO, USA.
[4] Department of Genetics, Washington University School of Medicine, St. Louis, MO, USA.
[5] Institute for Genomic Medicine, The Research Institute at Nationwide Children's Hospital, Columbus, USA.

[6] Department of Pediatrics, The Ohio State University College of Medicine, Columbus, USA.
[7] Department of Pathology and Immunology, Washington University School of Medicine, St. Louis, Missouri, USA.
*Corresponding author

BJA wrote the manuscript, assembled and cleaned training data, performed model training and analysis, and wrote all code. EKB performed manual review, analysis, and wrote the manuscript. RG and TER shared genomic data that was used in training the classifier. MG, SJS, ERM, and OLG supervised the project and revised the paper.

4.7 Acknowledgements

We would like to thank Allegra Petti, Gue Su Chang, Tiandao Li, Chris Miller, Lee Trani, Robert Lesurf, Zach Skidmore, Kilannin Krysiak, Avi Ramu, Felicia Gomez and Alex Wagner for assisting in data assembly. We also acknowledge Lee Trani for performing manual review on the HCC1395 cell line and Lee Trani and Alex Wagner for valuable discussion on the project. We gratefully acknowledge Lukas Wartman, John DiPersio, and Tim Ley for sharing genomic data that made this project possible. The data on benign breast disease were generated through grants to TER from NIH/NCI (R01CA142942) and the Breast Cancer Research Foundation. Select sample data was funded by the Genomics of AML PPG (T.Ley, PI, P01 CA101937). BJA was supported by the Siteman Cancer Center. SJS is funded by the National Library of Medicine (NIH NLM R01LM012222 and NIH NLM R01LM012482). MG is funded by the National Human Genome Research Institute (NIH NHGRI R00HG007940). OLG is funded by the National Cancer Institute (NIH NCI K22CA188163 and NIH NCI U01CA209936).

Chapter 5: Conclusions and Future Directions

The research presented herein will result in a significant improvement in quality and reproducibility to somatic variant identification. Openly licensed, open access curated somatic variant knowledgebases, like DoCM described in Chapter 2, aim to enable automated utilization of the cancer genomics literature in computational workflows. The knowledge-driven variant discovery strategy, described in Chapter 3, permits sensitive detection of known, clinically relevant and functionally important cancer variants, which has broad utility both in research and clinical sequencing analyses. Finally, by automating the post-processing of somatic variant calls through machine learning (Chapter 4), the analysis time needed in MPS-based cancer studies will be reduced dramatically, while enabling more reproducible and data-driven variant identification. These improvements also lift the technical limitation on the number of somatic calls that can be evaluated, in contrast to the time constraints imposed by manual analysis.

5.1 DoCM: a Database of Curated Variants in Cancer

In Chapter 3, DoCM, a Database of Curated Mutations in Cancer (http://docm.info), is described. DoCM is an open source, openly licensed resource that enables the cancer research community to aggregate, store and track biologically important cancer variants. DoCM is currently comprised of 1,364 variants in 132 genes across 122 cancer subtypes, based on the curation of 876 publications. Due to the massive growth in the cancer literature (see **Figure 3.6**) it is a laborious task to accurately catalogue the mutations whose importance in cancer has been defined. Crowdsourced, open resources can help address this enormous problem.

While DoCM is a substantial improvement on earlier resources in terms of its size and utility in computational pipelines, it is far from a comprehensive listing of all the important mutations discovered in cancer. In fact, many similar cancer variant resources that have been released following DoCM, like CIViC[124], have minimal variant overlap, suggesting that there is a need for many more individuals to focus on the curation problem and to develop resources that aggregate curation results.

5.2 Knowledge Based Variant Discovery of Functional Variants

The use of DoCM to focus knowledge-based discovery of biologically or functionally important cancer mutations resulted in the recovery of at least one functionally important variant in 41% of TCGA cases examined (Chapter 3). These results demonstrated that 1,228 variants from four TCGA studies that were missed by the original sequencing analysis pipeline could be recovered through DoCM-guided discovery. Approaches such as this can rescue true variants that may be important to identify in clinical assays of cancer DNA that aim to identify pathogenic mutations with corresponding targeted therapies.

The MSRV approach, outlined in Chapter 4, could reliably recover missed somatic variants through validation sequencing-based confirmation. However, the process was limited by a high manual review burden and validation sequencing cost. Additionally, the heuristics used to identify putative variants and validate the findings leave room for skepticism regarding the consistency and bias of the approach. This study motivated the development of the machine learning classifier, described in Chapter 4, to address these limitations by substantially decreasing the manual review burden, and providing a measure of statistical confidence to inform subsequent decision-making.

5.3 Automation of Manual Review through a Machine Learning Classifier

The results in Chapter 4 illustrate that a machine learning-based classifier can automate manual review with excellent performance. Specifically, the deep learning approach we developed classifies somatic calls with a ROC AUC of 0.98 in cross-validation performance, 0.89 on an independent sequencing dataset, and 0.95 on an independent dataset with orthogonal sequencing validation data. It is particularly promising that this method can accurately predict orthogonal validation sequencing results, since these often represent a necessary, time-consuming and expensive step in cancer variant discovery studies. With further assessment, confidence in somatic variant calls evaluated with this approach could obviate orthogonal validation. Even if validation sequencing remains a necessity, an economic framework can be used in conjunction with the probabilistic output of the classifier to help investigators to tune somatic variant discovery to be most efficient for their budgets[135]. Additionally, the machine learning classifier provides a standardized, consistent, and reproducible analysis strategy that will facilitate improved inter- and intra-pipeline comparison, facilitating quantifiable somatic variant identification performance for the entire variant identification pipeline between runs in the same lab and between individual labs. This could help reduce the inter-pipeline variability seen across many genomic datasets[62].

While the machine learning classifier is extremely promising, there are several caveats and limitations that should be considered when using it. Since the classifier was trained on data from 20 different projects at the same institute (sequenced with similar laboratory procedures, sequencing instruments, and a standardized analytical pipeline) it remains to be demonstrated that the classifier will deliver consistent results on data produced under different circumstances. Additionally, while the training data set is large and represents various cancer types, it is far from a comprehensive representation of all cancer types. Lastly, while the classifier's performance compared to validation sequencing results was highly comparable, it should be noted that this sample had substantially more coverage than is typical in most studies in the peer-reviewed literature (300-fold WGS discovery sequencing and 1000-fold custom capture validation sequencing compared to typical studies that achieve targets of 60-fold and 200-fold coverage in WGS discover and custom capture validation sequencing, respectively).

There are several future analyses that can improve the classifier. Additional evaluation of model performance on sequencing data produced at various sequencing centers, at various coverage levels, and with a variety of cancer types and analysis pipelines should be performed to test the model's robustness to global differences from single-center data. In the short term, the model could be improved by and would be well suited to an active learning implementation, where the machine learning algorithm sends low confidence calls to a human reviewer, and then is iteratively trained based on the results, to further improve model performance. Such a scheme could train the classifier to actively adapt to changes in sequencing protocols and analysis.

In the longer term, the success of this approach on the automated post-processing of somatic variant calls, coupled with the application of machine learning algorithms to call germline variants, could yield a machine learning algorithm well suited to somatic variant calling that might replace, rather than simply supplement, existing callers[137]. This development would require a large training dataset that has good representation from a variety of cancer types, sequencing pipelines, and analysis workflows, and with high quality validation sequencing results, to serve as training data. While collecting such a dataset would require investment of time and resources, it is likely based on our results that it would outperform existing options. Given the extensive number of cancer genomes already sequenced by TCGA and ICGC, the production of a such a training dataset could be accomplished by performing extensive validation sequencing on previously identified somatic mutations if the initial sample DNAs were still available.

5.4 Conclusion

The research presented in this dissertation describes methods that substantially reduce the time and cost to identify, analyze, and prioritize somatic cancer mutations while simultaneously increasing somatic variant calling accuracy and the number of clinically relevant mutations identified. DoCM reduces interpretation time by providing an open resource to catalogue the results from publications that describe the clinical or functional impact of known somatic mutations in cancer. This reduces the literature review that is duplicated every time a somatic variant is identified and evaluated by different studies. The machine learning-based classifier is not limited by human time to review variants, as such orders of magnitude more somatic variant calls can be classified, while simultaneously reducing required manual review by an order of magnitude. This classifier also improves on prior methods by providing a concrete probability that represents the confidence of a variant call. Additionally, the DoCM MSRV approach recovers at least one clinically or biologically important somatic mutation in as many as 41% of samples evaluated (see subsection 3.3.5), while the manual review classifier improves the identification of clinically important variants because substantially more variants are classified. The results of this dissertation suggest that there is substantial room for development to improve detection of somatic cancer mutations as genomics is translated into clinical use. As shown in **Figure 4.7**, it is possible for machine learning methods to accurately predict which somatic variant calls will be validated by orthogonal validation sequencing. Given this result, it is plausible that future model development and evaluation could lead to the elimination of confirmatory validation sequencing of somatic variants. The results of this dissertation increase the potential to apply cancer genomic data in clinical practice by widening major bottlenecks in analysis of somatic variants, while simultaneously increasing the accuracy and the number of clinically important variants identified.

References

1 Hanahan, D. & Weinberg, R. A. The hallmarks of cancer. *Cell* **100**, 57-70 (2000).

2 Hanahan, D. & Weinberg, R. A. Hallmarks of cancer: the next generation. *Cell* **144**, 646-674, doi:10.1016/j.cell.2011.02.013 (2011).

3 International Human Genome Sequencing, C. Finishing the euchromatic sequence of the human genome. *Nature* **431**, 931-945, doi:10.1038/nature03001 (2004).

4 Mardis, E. R. A decade's perspective on DNA sequencing technology. *Nature* **470**, 198-203, doi:10.1038/nature09796 (2011).

5 Mardis, E. R. The translation of cancer genomics: time for a revolution in clinical cancer care. *Genome Med* **6**, 22, doi:10.1186/gm539 (2014).

6 Sboner, A., Mu, X. J., Greenbaum, D., Auerbach, R. K. & Gerstein, M. B. The real cost of sequencing: higher than you think! *Genome Biol* **12**, 125, doi:10.1186/gb-2011-12-8-125 (2011).

7 Mohamed, S. & Syed, B. A. Commercial prospects for genomic sequencing technologies. *Nat Rev Drug Discov* **12**, 341-342, doi:10.1038/nrd4006 (2013).

8 Ding, L., Wendl, M. C., Koboldt, D. C. & Mardis, E. R. Analysis of next-generation genomic data in cancer: accomplishments and challenges. *Hum Mol Genet* **19**, R188-196, doi:10.1093/hmg/ddq391 (2010).

9 Robin, J. D., Ludlow, A. T., LaRanger, R., Wright, W. E. & Shay, J. W. Comparison of DNA Quantification Methods for Next Generation Sequencing. *Sci Rep* **6**, 24067, doi:10.1038/srep24067 (2016).

10 Mardis, E. R. Next-generation sequencing platforms. *Annu Rev Anal Chem (Palo Alto Calif)* **6**, 287-303, doi:10.1146/annurev-anchem-062012-092628 (2013).

11 Wang, X. V., Blades, N., Ding, J., Sultana, R. & Parmigiani, G. Estimation of sequencing error rates in short reads. *BMC Bioinformatics* **13**, 185, doi:10.1186/1471-2105-13-185 (2012).

12 Li, H. & Homer, N. A survey of sequence alignment algorithms for next-generation sequencing. *Brief Bioinform* **11**, 473-483, doi:10.1093/bib/bbq015 (2010).

13 Fonseca, N. A., Rung, J., Brazma, A. & Marioni, J. C. Tools for mapping high-throughput sequencing data. *Bioinformatics* **28**, 3169-3177, doi:10.1093/bioinformatics/bts605 (2012).

14 Trapnell, C. & Salzberg, S. L. How to map billions of short reads onto genomes. *Nat Biotechnol* **27**, 455-457, doi:10.1038/nbt0509-455 (2009).

15 Treangen, T. J. & Salzberg, S. L. Repetitive DNA and next-generation sequencing: computational challenges and solutions. *Nat Rev Genet* **13**, 36-46, doi:10.1038/nrg3117 (2011).

16 Li, H. & Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25**, 1754-1760, doi:10.1093/bioinformatics/btp324 (2009).

17 Li, H. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv preprint arXiv:1303.3997* (2013).

18 Langmead, B. & Salzberg, S. L. Fast gapped-read alignment with Bowtie 2. *Nat Methods* **9**, 357-359, doi:10.1038/nmeth.1923 (2012).

19 Smith, T. F. & Waterman, M. S. Identification of common molecular subsequences. *J Mol Biol* **147**, 195-197 (1981).

20 Needleman, S. B. & Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol* **48**, 443-453 (1970).

21 McKenna, A. *et al.* The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res* **20**, 1297-1303, doi:10.1101/gr.107524.110 (2010).

22 Koboldt, D. C. *et al.* VarScan: variant detection in massively parallel sequencing of individual and pooled samples. *Bioinformatics* **25**, 2283-2285, doi:10.1093/bioinformatics/btp373 (2009).

23 Garrison, E. & Marth, G. Haplotype-based variant detection from short-read sequencing. *arXiv preprint arXiv:1207.3907* (2012).

24 Rimmer, A. *et al.* Integrating mapping-, assembly- and haplotype-based approaches for calling variants in clinical sequencing applications. *Nat Genet* **46**, 912-918, doi:10.1038/ng.3036 (2014).

25 Li, H. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics* **27**, 2987-2993, doi:10.1093/bioinformatics/btr509 (2011).

26 Robinson, J. T. *et al.* Integrative genomics viewer. *Nat Biotechnol* **29**, 24-26, doi:10.1038/nbt.1754 (2011).

27 Thorvaldsdottir, H., Robinson, J. T. & Mesirov, J. P. Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration. *Brief Bioinform* **14**, 178-192, doi:10.1093/bib/bbs017 (2013).

28 Alkan, C., Coe, B. P. & Eichler, E. E. Genome structural variation discovery and genotyping. *Nat Rev Genet* **12**, 363-376, doi:10.1038/nrg2958 (2011).

29 Hormozdiari, F., Alkan, C., Eichler, E. E. & Sahinalp, S. C. Combinatorial algorithms for structural variation detection in high-throughput sequenced genomes. *Genome Res* **19**, 1270-1278, doi:10.1101/gr.088633.108 (2009).

30 Hormozdiari, F. *et al.* Next-generation VariationHunter: combinatorial algorithms for transposon insertion discovery. *Bioinformatics* **26**, i350-357, doi:10.1093/bioinformatics/btq216 (2010).

31 Hormozdiari, F., Hajirasouliha, I., McPherson, A., Eichler, E. E. & Sahinalp, S. C. Simultaneous structural variation discovery among multiple paired-end sequenced genomes. *Genome Res* **21**, 2203-2212, doi:10.1101/gr.120501.111 (2011).

32 Chen, K. *et al.* BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nat Methods* **6**, 677-681, doi:10.1038/nmeth.1363 (2009).

33 Mills, R. E. *et al.* Mapping copy number variation by population-scale genome sequencing. *Nature* **470**, 59-65, doi:10.1038/nature09708 (2011).

34 Redon, R. *et al.* Global variation in copy number in the human genome. *Nature* **444**, 444-454, doi:10.1038/nature05329 (2006).

35 Klambauer, G. *et al.* cn.MOPS: mixture of Poissons for discovering copy number variations in next-generation sequencing data with a low false discovery rate. *Nucleic Acids Res* **40**, e69, doi:10.1093/nar/gks003 (2012).

36 Chen, K. *et al.* TIGRA: a targeted iterative graph routing assembler for breakpoint assembly. *Genome Res* **24**, 310-317, doi:10.1101/gr.162883.113 (2014).

37 Belkadi, A. *et al.* Whole-genome sequencing is more powerful than whole-exome sequencing for detecting exome variants. *Proc Natl Acad Sci U S A* **112**, 5473-5478, doi:10.1073/pnas.1418631112 (2015).

38 Ye, K., Schulz, M. H., Long, Q., Apweiler, R. & Ning, Z. Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics* **25**, 2865-2871, doi:10.1093/bioinformatics/btp394 (2009).

39 Cancer Genome Atlas Research, N. *et al.* The Cancer Genome Atlas Pan-Cancer analysis project. *Nat Genet* **45**, 1113-1120, doi:10.1038/ng.2764 (2013).

40 International Cancer Genome, C. *et al.* International network of cancer genome projects. *Nature* **464**, 993-998, doi:10.1038/nature08987 (2010).

41 Zhang, J. *et al.* International Cancer Genome Consortium Data Portal--a one-stop shop for cancer genomics data. *Database (Oxford)* **2011**, bar026, doi:10.1093/database/bar026 (2011).

42 Genomes Project, C. *et al.* A global reference for human genetic variation. *Nature* **526**, 68-74, doi:10.1038/nature15393 (2015).

43 Lek, M. *et al.* Analysis of protein-coding genetic variation in 60,706 humans. *Nature* **536**, 285-291, doi:10.1038/nature19057 (2016).

44 Kumar, P., Henikoff, S. & Ng, P. C. Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm. *Nat Protoc* **4**, 1073-1081, doi:10.1038/nprot.2009.86 (2009).

45 Ng, P. C. & Henikoff, S. Predicting the effects of amino acid substitutions on protein function. *Annu Rev Genomics Hum Genet* **7**, 61-80, doi:10.1146/annurev.genom.7.080505.115630 (2006).

46 Adzhubei, I. A. *et al.* A method and server for predicting damaging missense mutations. *Nat Methods* **7**, 248-249, doi:10.1038/nmeth0410-248 (2010).

47 Reva, B., Antipin, Y. & Sander, C. Predicting the functional impact of protein mutations: application to cancer genomics. *Nucleic Acids Res* **39**, e118, doi:10.1093/nar/gkr407 (2011).

48 Carter, H. *et al.* Cancer-specific high-throughput annotation of somatic mutations: computational prediction of driver missense mutations. *Cancer Res* **69**, 6660-6667, doi:10.1158/0008-5472.CAN-09-1133 (2009).

49 Kumar, R. D., Swamidass, S. J. & Bose, R. Unsupervised detection of cancer driver mutations with parsimony-guided learning. *Nat Genet* **48**, 1288-1294, doi:10.1038/ng.3658 (2016).

50 Forbes, S. A. *et al.* COSMIC: somatic cancer genetics at high-resolution. *Nucleic Acids Res* **45**, D777-D783, doi:10.1093/nar/gkw1121 (2017).

51 Ley, T. J. *et al.* DNA sequencing of a cytogenetically normal acute myeloid leukaemia genome. *Nature* **456**, 66-72, doi:10.1038/nature07485 (2008).

52 Garraway, L. A. & Lander, E. S. Lessons from the cancer genome. *Cell* **153**, 17-37, doi:10.1016/j.cell.2013.03.002 (2013).

53 Lawrence, M. S. *et al.* Mutational heterogeneity in cancer and the search for new cancer-associated genes. *Nature* **499**, 214-218, doi:10.1038/nature12213 (2013).

54 Van Allen, E. M. *et al.* Whole-exome sequencing and clinical interpretation of formalin-fixed, paraffin-embedded tumor samples to guide precision cancer medicine. *Nat Med* **20**, 682-688, doi:10.1038/nm.3559 (2014).

55 Schmitt, M. W. *et al.* Detection of ultra-rare mutations by next-generation sequencing. *Proc Natl Acad Sci U S A* **109**, 14508-14513, doi:10.1073/pnas.1208715109 (2012).

56 Kinde, I., Wu, J., Papadopoulos, N., Kinzler, K. W. & Vogelstein, B. Detection and quantification of rare mutations with massively parallel sequencing. *Proc Natl Acad Sci U S A* **108**, 9530-9535, doi:10.1073/pnas.1105422108 (2011).

57 Hiley, C., de Bruin, E. C., McGranahan, N. & Swanton, C. Deciphering intratumor heterogeneity and temporal acquisition of driver events to refine precision medicine. *Genome Biol* **15**, 453, doi:10.1186/s13059-014-0453-8 (2014).

58 Welch, J. S. *et al.* The origin and evolution of mutations in acute myeloid leukemia. *Cell* **150**, 264-278, doi:10.1016/j.cell.2012.06.023 (2012).

59 Kroigard, A. B., Thomassen, M., Laenkholm, A. V., Kruse, T. A. & Larsen, M. J. Evaluation of Nine Somatic Variant Callers for Detection of Somatic Mutations in Exome and Targeted Deep Sequencing Data. *PLoS One* **11**, e0151664, doi:10.1371/journal.pone.0151664 (2016).

60 Xu, H., DiCarlo, J., Satya, R. V., Peng, Q. & Wang, Y. Comparison of somatic mutation calling methods in amplicon and whole exome sequence data. *BMC Genomics* **15**, 244, doi:10.1186/1471-2164-15-244 (2014).

61 Griffith, M. *et al.* Optimizing cancer genome sequencing and analysis. *Cell Syst* **1**, 210-223, doi:10.1016/j.cels.2015.08.015 (2015).

62 Alioto, T. S. *et al.* A comprehensive assessment of somatic mutation detection in cancer using whole-genome sequencing. *Nat Commun* **6**, 10001, doi:10.1038/ncomms10001 (2015).

63 Callari, M. *et al.* Intersect-then-combine approach: improving the performance of somatic variant calling in whole exome sequencing data using multiple aligners and callers. *Genome Med* **9**, 35, doi:10.1186/s13073-017-0425-1 (2017).

64 Robasky, K., Lewis, N. E. & Church, G. M. The role of replicates for error mitigation in next-generation sequencing. *Nat Rev Genet* **15**, 56-62, doi:10.1038/nrg3655 (2014).

65 Saunders, C. T. *et al.* Strelka: accurate somatic small-variant calling from sequenced tumor-normal sample pairs. *Bioinformatics* **28**, 1811-1817, doi:10.1093/bioinformatics/bts271 (2012).

66 Cibulskis, K. *et al.* Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nat Biotechnol* **31**, 213-219, doi:10.1038/nbt.2514 (2013).

67 Faust, G. G. & Hall, I. M. SAMBLASTER: fast duplicate marking and structural variant read extraction. *Bioinformatics* **30**, 2503-2505, doi:10.1093/bioinformatics/btu314 (2014).

68 Sinha, R. *et al.* Index switching causes “spreading-of-signal” among multiplexed samples in Illumina HiSeq 4000 DNA sequencing. *bioRxiv*, 125724 (2017).

69 Costello, M. *et al.* Characterization and remediation of sample index swaps by non-redundant dual indexing on massively parallel sequencing platforms. *bioRxiv*, 200790 (2017).

70 Li, H. Toward better understanding of artifacts in variant calling from high-coverage samples. *Bioinformatics* **30**, 2843-2851, doi:10.1093/bioinformatics/btu356 (2014).

71 Guo, Y. *et al.* The effect of strand bias in Illumina short-read sequencing data. *BMC Genomics* **13**, 666, doi:10.1186/1471-2164-13-666 (2012).

72 Larson, D. E. *et al.* SomaticSniper: identification of somatic point mutations in whole genome sequencing data. *Bioinformatics* **28**, 311-317, doi:10.1093/bioinformatics/btr665 (2012).

73 Koboldt, D. C. *et al.* VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res* **22**, 568-576, doi:10.1101/gr.129684.111 (2012).

74 Hansen, N. F., Gartner, J. J., Mei, L., Samuels, Y. & Mullikin, J. C. Shimmer: detection of genetic alterations in tumors using next-generation sequence data. *Bioinformatics* **29**, 1498-1503, doi:10.1093/bioinformatics/btt183 (2013).

75 Robinson, J. T., Thorvaldsdottir, H., Wenger, A. M., Zehir, A. & Mesirov, J. P. Variant Review with the Integrative Genomics Viewer. *Cancer Res* **77**, e31-e34, doi:10.1158/0008-5472.CAN-17-0337 (2017).

76 Good, B. M., Ainscough, B. J., McMichael, J. F., Su, A. I. & Griffith, O. L. Organizing knowledge to enable personalization of medicine in cancer. *Genome Biol* **15**, 438, doi:10.1186/s13059-014-0438-7 (2014).

77 Yeh, P. *et al.* DNA-Mutation Inventory to Refine and Enhance Cancer Treatment (DIRECT): a catalog of clinically relevant cancer mutations to enable genome-directed anticancer therapy. *Clin Cancer Res* **19**, 1894-1901, doi:10.1158/1078-0432.CCR-12-1894 (2013).

78 Dienstmann, R. *et al.* Standardized decision support in next generation sequencing reports of somatic cancer variants. *Mol Oncol* **8**, 859-873, doi:10.1016/j.molonc.2014.03.021 (2014).

79 Witten, I. H., Frank, E., Hall, M. A. & Pal, C. J. *Data Mining: Practical machine learning tools and techniques*. (Morgan Kaufmann, 2016).

80 Valverde-Albacete, F. J. & Pelaez-Moreno, C. 100% classification accuracy considered harmful: the normalized information transfer factor explains the accuracy paradox. *PLoS One* **9**, e84217, doi:10.1371/journal.pone.0084217 (2014).

81 Skbkekas. *Traintest.svg By Skbkekas (Own work) [CC BY 3.0 (*http://creativecommons.org/licenses/by/3.0*)], via Wikimedia Commons*, <https://commons.wikimedia.org/wiki/File%3ATraintest.svg> (2009).

82 Freedman, D. *Statistical models : theory and practice*. 128 (Cambridge University Press, 2009).

83 Michaelg2015. *By Michaelg2015 (Own work) [CC BY-SA 4.0 (*https://creativecommons.org/licenses/by-sa/4.0*)], via Wikimedia Commons*, <https://commons.wikimedia.org/wiki/File%3AExam_pass_logistic_curve.jpeg> (2015).

84 Szumilas, M. Explaining odds ratios. *J Can Acad Child Adolesc Psychiatry* **19**, 227-229 (2010).

85 Ho, T. K. in *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on.* 278-282 (IEEE).

86 Ho, T. K. The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence* **20**, 832-844 (1998).

87 Breiman, L. Random forests. *Machine learning* **45**, 5-32 (2001).

88 *Random forest. Accessed 11/17/17*, <https://en.wikipedia.org/wiki/Random_forest> (2017).

89 Dlary. *By Dlary [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC BY-SA 4.0 (https://creativecommons.org/licenses/by-sa/4.0)], via Wikimedia Commons*, <https://commons.wikimedia.org/wiki/File%3ATitanic_Survival_Decison_Tree_SVG.png> (2017).

90 Hastie, T., Tibshirani, R. & Friedman, J. H. *The elements of statistical learning : data mining, inference, and prediction*. 587-588 (Springer, 2001).

91 Rosenblatt, F. *The perceptron, a perceiving and recognizing automaton Project Para*. (Cornell Aeronautical Laboratory, 1957).

92 Hinton, G. E. Learning multiple layers of representation. *Trends in cognitive sciences* **11**, 428-434 (2007).

93 LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436-444, doi:10.1038/nature14539 (2015).

94 MacConaill, L. E. *et al.* Prospective enterprise-level molecular genotyping of a cohort of cancer patients. *J Mol Diagn* **16**, 660-672, doi:10.1016/j.jmoldx.2014.06.004 (2014).

95 McGranahan, N. *et al.* Clonal status of actionable driver events and the timing of mutational processes in cancer evolution. *Sci Transl Med* **7**, 283ra254, doi:10.1126/scitranslmed.aaa1408 (2015).

96 Leiserson, M. D. *et al.* MAGI: visualization and collaborative annotation of genomic aberrations. *Nat Methods* **12**, 483-484, doi:10.1038/nmeth.3412 (2015).

97 Wagner, A. H. *et al.* DGIdb 2.0: mining clinically relevant drug-gene interactions. *Nucleic Acids Res* **44**, D1036-1044, doi:10.1093/nar/gkv1165 (2016).

98 Krogan, N. J., Lippman, S., Agard, D. A., Ashworth, A. & Ideker, T. The cancer cell map initiative: defining the hallmark networks of cancer. *Mol Cell* **58**, 690-698, doi:10.1016/j.molcel.2015.05.008 (2015).

99 Griffith, M. *et al.* CIViC is a community knowledgebase for expert crowdsourcing the clinical interpretation of variants in cancer. *Nat Genet* **49**, 170-174, doi:10.1038/ng.3774 (2017).

100 Morris, V. & Kopetz, S. BRAF inhibitors in clinical oncology. *F1000Prime Rep* **5**, 11, doi:10.12703/P5-11 (2013).

101 Kent, W. J. *et al.* The human genome browser at UCSC. *Genome Res* **12**, 996-1006, doi:10.1101/gr.229102. Article published online before print in May 2002 (2002).

102 McLaren, W. *et al.* Deriving the consequences of genomic variants with the Ensembl API and SNP Effect Predictor. *Bioinformatics* **26**, 2069-2070, doi:10.1093/bioinformatics/btq330 (2010).

103 Schriml, L. M. & Mitraka, E. The Disease Ontology: fostering interoperability between biological and clinical human disease-related data. *Mamm Genome* **26**, 584-589, doi:10.1007/s00335-015-9576-9 (2015).

104 Zhou, W. *et al.* ClinSeK: a targeted variant characterization framework for clinical sequencing. *Genome Med* **7**, 34, doi:10.1186/s13073-015-0155-1 (2015).

105 Kancha, R. K., Grundler, R., Peschel, C. & Duyster, J. Sensitivity toward sorafenib and sunitinib varies between different activating and drug-resistant FLT3-ITD mutations. *Exp Hematol* **35**, 1522-1526, doi:10.1016/j.exphem.2007.07.008 (2007).

106 Kindler, T., Lipka, D. B. & Fischer, T. FLT3 as a therapeutic target in AML: still challenging after all these years. *Blood* **116**, 5089-5102, doi:10.1182/blood-2010-04-261867 (2010).

107 Mpakou, V. E. *et al.* Dasatinib inhibits proliferation and induces apoptosis in the KASUMI-1 cell line bearing the t(8;21)(q22;q22) and the N822K c-kit mutation. *Leuk Res* **37**, 175-182, doi:10.1016/j.leukres.2012.10.011 (2013).

108 Courtney, K. D., Corcoran, R. B. & Engelman, J. A. The PI3K pathway as drug target in human cancer. *J Clin Oncol* **28**, 1075-1083, doi:10.1200/JCO.2009.25.3641 (2010).

109 Janku, F. *et al.* PI3K/AKT/mTOR inhibitors in patients with breast and gynecologic malignancies harboring PIK3CA mutations. *J Clin Oncol* **30**, 777-782, doi:10.1200/JCO.2011.36.1196 (2012).

110 Griffith, M. *et al.* Genome Modeling System: A Knowledge Management Platform for Genomics. *PLoS Comput Biol* **11**, e1004274, doi:10.1371/journal.pcbi.1004274 (2015).

111 Cancer Genome Atlas Research, N. Genomic and epigenomic landscapes of adult de novo acute myeloid leukemia. *N Engl J Med* **368**, 2059-2074, doi:10.1056/NEJMoa1301689 (2013).

112 Cancer Genome Atlas, N. Comprehensive molecular portraits of human breast tumours. *Nature* **490**, 61-70, doi:10.1038/nature11412 (2012).

113 Cancer Genome Atlas Research, N. Integrated genomic analyses of ovarian carcinoma. *Nature* **474**, 609-615, doi:10.1038/nature10166 (2011).

114 Cancer Genome Atlas Research, N. *et al.* Integrated genomic characterization of endometrial carcinoma. *Nature* **497**, 67-73, doi:10.1038/nature12113 (2013).

115 Ainscough, B. J. *et al.* DoCM: a database of curated mutations in cancer. *Nat Methods* **13**, 806-807, doi:10.1038/nmeth.4000 (2016).

116 Griffith, O. L. *et al.* Truncating Prolactin Receptor Mutations Promote Tumor Growth in Murine Estrogen Receptor-Alpha Mammary Carcinomas. *Cell Rep* **17**, 249-260, doi:10.1016/j.celrep.2016.08.076 (2016).

117 Rasche, L. *et al.* Spatial genomic heterogeneity in multiple myeloma revealed by multi-region sequencing. *Nat Commun* **8**, 268, doi:10.1038/s41467-017-00296-y (2017).

118 Ott, P. A. *et al.* An immunogenic personal neoantigen vaccine for patients with melanoma. *Nature* **547**, 217-221, doi:10.1038/nature22991 (2017).

119 Giannakis, M. *et al.* RNF43 is frequently mutated in colorectal and endometrial cancers. *Nat Genet* **46**, 1264-1266, doi:10.1038/ng.3127 (2014).

120 Cancer Genome Atlas Research, N. *et al.* Comprehensive, Integrative Genomic Analysis of Diffuse Lower-Grade Gliomas. *N Engl J Med* **372**, 2481-2498, doi:10.1056/NEJMoa1402121 (2015).

121 Ding, J. *et al.* Feature-based classifiers for somatic mutation detection in tumour-normal paired sequencing data. *Bioinformatics* **28**, 167-175, doi:10.1093/bioinformatics/btr629 (2012).

122 Spinella, J. F. *et al.* SNooPer: a machine learning-based method for somatic variant identification from low-pass next-generation sequencing. *BMC Genomics* **17**, 912, doi:10.1186/s12864-016-3281-2 (2016).

123 Strom, S. P. Current practices and guidelines for clinical next-generation sequencing oncology testing. *Cancer Biol Med* **13**, 3-11, doi:10.28092/j.issn.2095-3941.2016.0004 (2016).

124 Griffith, M. *et al.* Comprehensive genomic analysis reveals FLT3 activation and a therapeutic strategy for a patient with relapsed adult B-lymphoblastic leukemia. *Exp Hematol* **44**, 603-613, doi:10.1016/j.exphem.2016.04.011 (2016).

125 Ma, C. X. *et al.* A Phase I Trial of BKM120 (Buparlisib) in Combination with Fulvestrant in Postmenopausal Women with Estrogen Receptor-Positive Metastatic Breast Cancer. *Clin Cancer Res* **22**, 1583-1591, doi:10.1158/1078-0432.CCR-15-1745 (2016).

126 Klco, J. M. *et al.* Association Between Mutation Clearance After Induction Therapy and Outcomes in Acute Myeloid Leukemia. *JAMA* **314**, 811-822, doi:10.1001/jama.2015.9643 (2015).

127 Welch, J. S. *et al.* TP53 and Decitabine in Acute Myeloid Leukemia and Myelodysplastic Syndromes. *N Engl J Med* **375**, 2023-2036, doi:10.1056/NEJMoa1605949 (2016).

128 Krysiak, K. *et al.* Recurrent somatic mutations affecting B-cell receptor signaling pathway genes in follicular lymphoma. *Blood* **129**, 473-483, doi:10.1182/blood-2016-07-729954 (2017).

129 Lesurf, R. *et al.* Genomic characterization of HER2-positive breast cancer and response to neoadjuvant trastuzumab and chemotherapy-results from the ACOSOG Z1041 (Alliance) trial. *Ann Oncol* **28**, 1070-1077, doi:10.1093/annonc/mdx048 (2017).

130 Uy, G. L. *et al.* Dynamic changes in the clonal structure of MDS and AML in response to epigenetic therapy. *Leukemia* **31**, 872-881, doi:10.1038/leu.2016.282 (2017).

131 Li, H. & Durbin, R. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics* **26**, 589-595, doi:10.1093/bioinformatics/btp698 (2010).

132 Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **12**, 2825-2830 (2011).

133 Hunter, J. D. Matplotlib: A 2D graphics environment. *Comput Sci Eng* **9**, 90-95, doi:Doi 10.1109/Mcse.2007.55 (2007).

134 Oliphant, T. E. Python for scientific computing. *Comput Sci Eng* **9**, 10-20, doi:Doi 10.1109/Mcse.2007.58 (2007).

135 Swamidass, S. J., Bittker, J. A., Bodycombe, N. E., Ryder, S. P. & Clemons, P. A. An economic framework to prioritize confirmatory tests after a high-throughput screen. *J Biomol Screen* **15**, 680-686, doi:10.1177/1087057110372803 (2010).

136 Settles, B. Active learning literature survey. *University of Wisconsin, Madison* **52**, 11 (2010).

137 Poplin, R. *et al.* Creating a universal SNP and small indel variant caller with deep neural networks. *bioRxiv*, 092890 (2016).

Appendix 1: Manual Review Standard Operating Procedures

Created by:

Erica K. Barnell
Katie M. Campbell
Kilannin Krysiak
Benjamin J. Ainscough
Obi L. Griffith
Malachi Griffith

* This document is actively maintained for the most recent version of the SOP please visit https://docs.google.com/document/d/e/2PACX-1vRWPqcGIRe8YK_kn6ZFj79zdFxs2imrfSktlgYIjUQU8z41wASIoWtKCsyNdpmgYuf85CDXpQvp7LEP/pub

Preface

Purpose

The purpose of this guide is to introduce the manual review process to standardize the process. It describes how to download and use the Integrative Genomic Viewer (IGV), how to download and use the IGV Navigator, and how to call variants.

What is Manual Review

When tumor DNA is sequenced and aligned to the reference genome, the next step to analyze its somatic profile involves calling single nucleotide variants (SNVs) and short insertions and deletions (indels) that are present in the tumor. This is done by comparing its genome to a matched normal DNA sequence (usually blood for solid tumors and skin or buccal swab for liquid tumors) to identify differences specific to the tumor. SNVs and indels are detected by ‘variant callers,’ or software tools that scan the tumor and normal genomes to identify somatic variants. These are then filtered through pipelines and annotated to generate a confident list of SNVs and indels for the analyst to review.

There are limitations in variant calling due to the challenges and inaccuracies that occur during sample preparation, sequencing, and alignment, as well as fundamental genomic differences across individuals. To identify these errors, the analyst pursues a process of manual review. This process involves manually looking at each individual SNV and indel called and filtered in our pipeline and labeling it confidently as a real somatic variant (true positive) or a false positive for reasons related to sequencing and alignment errors. Given that there is some ambiguity in the process there is also the ability to call variants as a possible somatic variant.

Why Is Manual Review Important?

Variants called during manual review are used to finalize the somatic call set. Correctly calling true somatic variants is important for downstream analysis of the tumor. Somatic variants can be used to understand tumor dynamics, change treatment protocol, or compare to other tumor types.

Introduction

What is Integrative Genomics Viewer

The Integrative Genomics Viewer (IGV) is a high-performance visualization tool for analysis of large genomics datasets. It supports array-based as well as massively parellel sequence data with genomic annotations. It is used to conduct manual review of variants identified by somatic variant callers. The IGV has been supported by funding from the National Cancer Institute, the National Institutes of Health, the Informatics Technology for Cancer Research of the NCI, and the Starr Cancer Consortium. To cite IGV in publications please use the following citations:

James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, Jill P. Mesirov. Integrative Genomics Viewer. Nature Biotechnology 29, 24–26 (2011). [PUBMED]

Helga Thorvaldsdóttir, James T. Robinson, Jill P. Mesirov. Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration. Briefings in Bioinformatics 14, 178-192 (2013). [PUBMED]

How to Download IGV

The IGV Desktop Application can be accessed at http://software.broadinstitute.org/software/igv/

The IGV Downloads Page can be accessed at http://software.broadinstitute.org/software/igv/download

How to Use IGV

The IGV User Guide can be accessed at http://software.broadinstitute.org/software/igv/UserGuide

Helpful IGV Definitions:

Track = IGV displays data in horizontal rows called tracks. Typically, each track represents one sample or experiment.

Read Strand = Tracks are composed of read strands. Each read represents one molecule that was sequences and analyzed. Reads are mapped to the reference genome and differences in the read versus the reference genome are shown.

What is IGV Navigator

IGV Navigator (IGV Nav) is a tool developed at the Griffith Lab to assist in analyzing variants during manual review. Its input is a text file with variant coordinates and its output provides an annotation of these variant coordinates. This annotation includes the call (i.e. somatic, germline, ambiguous, or fail), tags to provide additional information if a variant is called ambiguous or fail, and a notes section for free text.

How to Download IGV Nav

IGV Nav can be downloaded from the Griffith Lab GitHub Repo (https://github.com/griffithlab/igvnav). Directions for download can be viewed

When using IGVNav for Mac, the program can be downloaded (IGVNav.zip), unzip and added to Applications/ folder.

How to Use IGV Nav

Input file

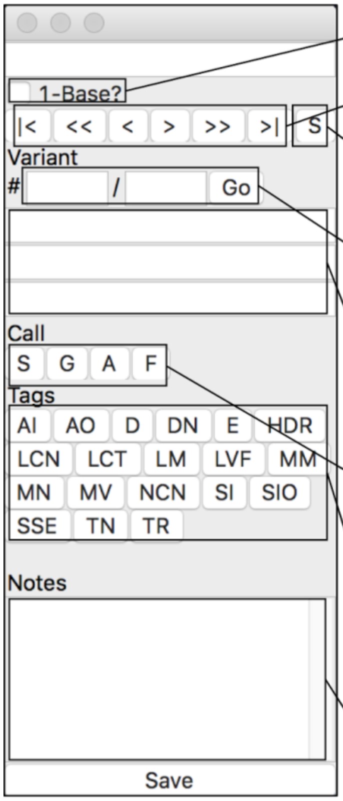
The input file for IGVNav is a five column bed file and accepts both 0- and 1-based genomic coordinates. The five columns correspond to chromosome, start coordinate, stop coordinate, reference allele, and called variant allele for each SNV and indel. The application requires the input file to contain a header line. This line will be replaced with a heading by the application upon opening.

Warning: If you do not provide a header, it will silently overwrite your first variant.

Navigation by IGVNav

IGV MUST be open before opening IGVnav. It operates by commanding IGV to move to the variant's corresponding genomic position in the IGV session.

IGVNav Interface



Check this box if you have 1-based input.

This bar permits you to navigate variants in your variant file.

This button sorts the variant by base to list mismatches first.

The first slots displays the current variant number and second slot displays the total number of variant in the variant file. The Go button will move to the variant displayed in the first slot.

This field will display coordinate information for the current variant listed in the field above

These buttons will record the manual review call (Somatic, Germline, Ambiguous, Fail)

These buttons will record the tags associated with each variant. Multiple tags can be applied to each variant. Typically, tags are recommended for all variants described as either ambiguous or fail.

The notes section allows for a written description for future analysis.

Step-by-Step: Setting up and IGV_2.3.79 Session

Step 1: Open an IGV Session

Step 2: Load Tracks (BAM files)

If you have a file accessible via URL: File > Load from URL... > input URL
If you have a locally accessible file: File > Load from File… > input file

Step 3: Load Additional Tracks

If you have a tumor only session, we recommend loading Common SNPs:
File > Load from Server… > Annotations > All Snps 1.4.2

Step 4: Color Tracks by Read Strand

Right click on each sample track loaded > Color Alignments by > read strands

Step 5: View Reads as Pairs

Right click on each sample track loaded > Make sure "view is pairs" is checked

Step 6: Open IGV Nav Session

Step 7: Load Variant File

Open file associated with IGV session. Variant file must be a tab separated file with the following columns: chr, start, stop, ref, var, call, tags, notes

Types of Analysis

Tumor Sample Only

If only tumor (DNA) is available, the normal track will not be loaded within the IGV session. True somatic variants must be determined by evaluating only the tumor DNA.

Note: It may be helpful to load a population SNP track within your IGV session to reduce the number of SNPs common in the human population (e.g. 1000 Genomes, ExAC). Somatic mutations are generally associated with lower frequencies in the human population.

Tumor Sample + Normal Sample

When tumor DNA and normal DNA are available they will both be loaded within the IGV session. This increases the ability to label true somatic variants during manual review due to the comparison with the normal track(s).

Note: Liquid tumor types might have tumor contamination in the normal tracks. Refer to Tumor Normal (TN) variant for information on how to call these variants.

Tumor Sample + Normal Sample + Other (RNAseq, Relapse, Metastasis)

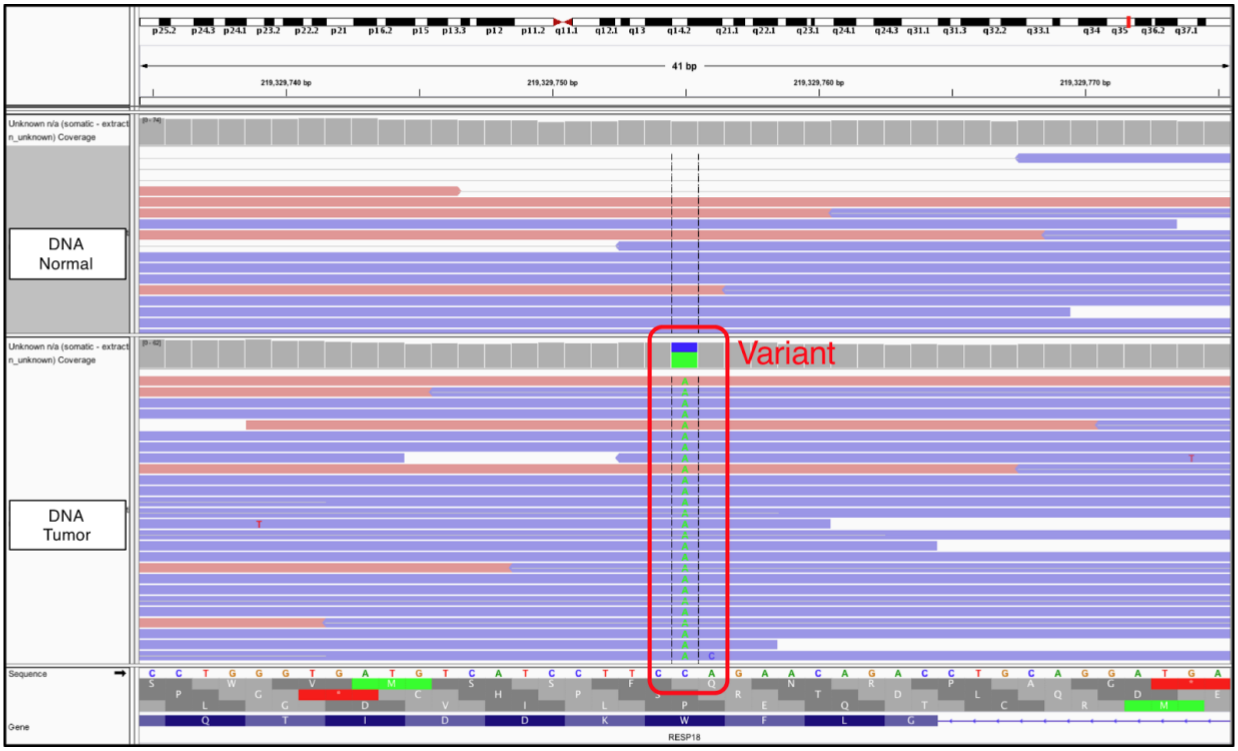
When tumor DNA, normal DNA and other DNA or RNA are available, they will all be loaded within the IGV session. Any support from other tracks increases the likelihood that the variant called is a somatic variant. Support from RNA sequencing data can be especially convincing and should be used to confirm somatic calls in expressed genes.

Note: Multiple tracks can be loaded into the same IGV session, however, increasing the total number of tracks will increase the time required for the IGV session to load. Downsizing the number of visualized reads, or taking a sampling of reads, can assist in decreasing loading time. This can be done using the IGV preferences pane. Warning: This should be considered when evaluating low variant allele frequency variants as it may cause visual artifacts such as variant support in reads in a single direction when in reality the variant is supported by reads in both directions.

Examples of Calls

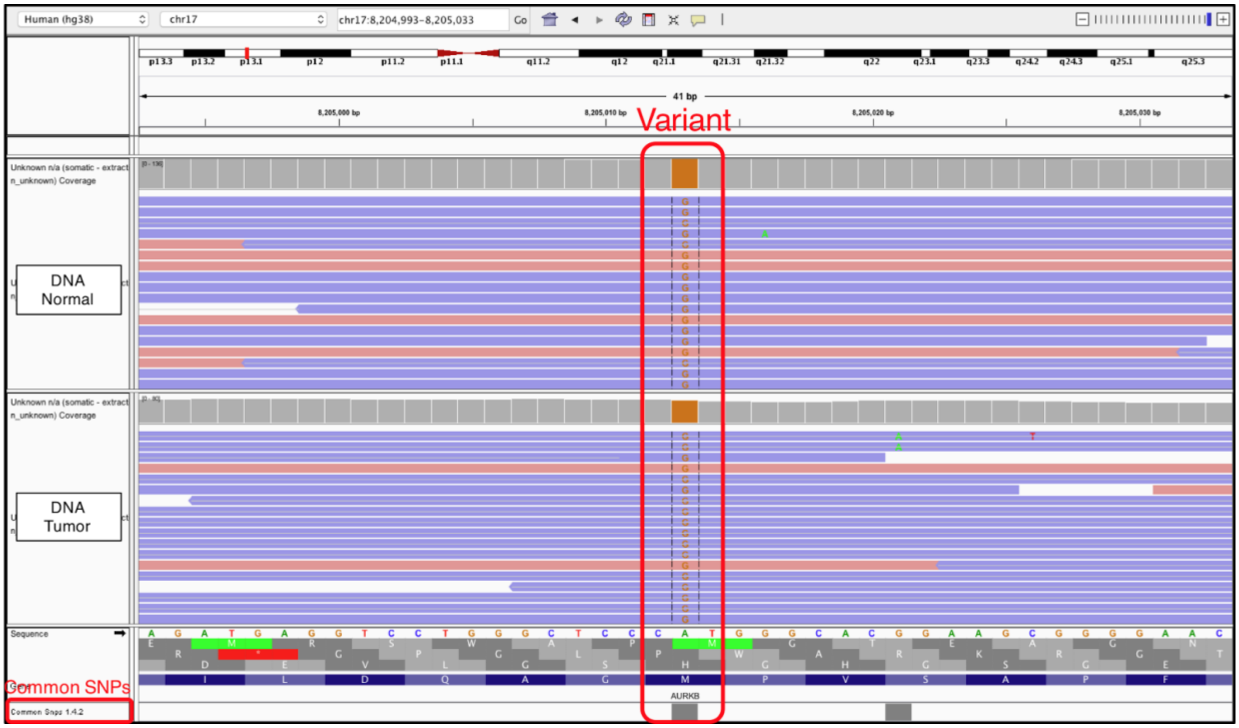
Somatic Calls

The variant is a real somatic variant. No reads supporting the variant (green, “A”) are visible in the normal sample.


p25.2 p24.3 p24.1 p23.2 p22.2 p21 p16.2 p15 p13.3 p12 p11.2 q11.1 q12.1 q13 q14.2 q21.1 q22.1 q23.1 q24.1 q24.3 q31.1 q31.3 q32.2 q33.1 q34 q35 q36.2 q37.1
41 bp
219,329,740 bp
219,329,750 bp
219,329,760 bp
219,329,770 bp
Unknown n/a (somatic - extract n_unknown) Coverage
[0 - 74]
DNA Normal
Unknown n/a (somatic - extract n_unknown) Coverage
[0 - 62]
Variant
DNA Tumor
Sequence
C C T G G G T G A T G T C A T C C T T C C A G A A C A G A C C T G C A G G A T G A
Gene
RESP18

Germline Calls

The variant is also present in the matched normal sample, indicating that the variant is germline. The variant was not somatic or specific to the tumor.



Helpful Hints:

* Typically this variant is present with a Variant Allele Frequency (VAF) near 50% or 100%, indicating that the germline polymorphism is either heterozygous (i.e. located on one allele) or homozygous (i.e. located on both alleles) in both the tumor and the normal.

* Bulk tumors are "contaminated" with normal cells. For this reason, 100% VAF in a non-purified tumor sample at a site with good depth should be suspicious and likely a homozygous germline polymorphism.

* If using GRCH38 you can view Common SNPs in the human genome, click:

"File" > "Load from Server" > "Annotations" > "Common Snps 1.4.2"

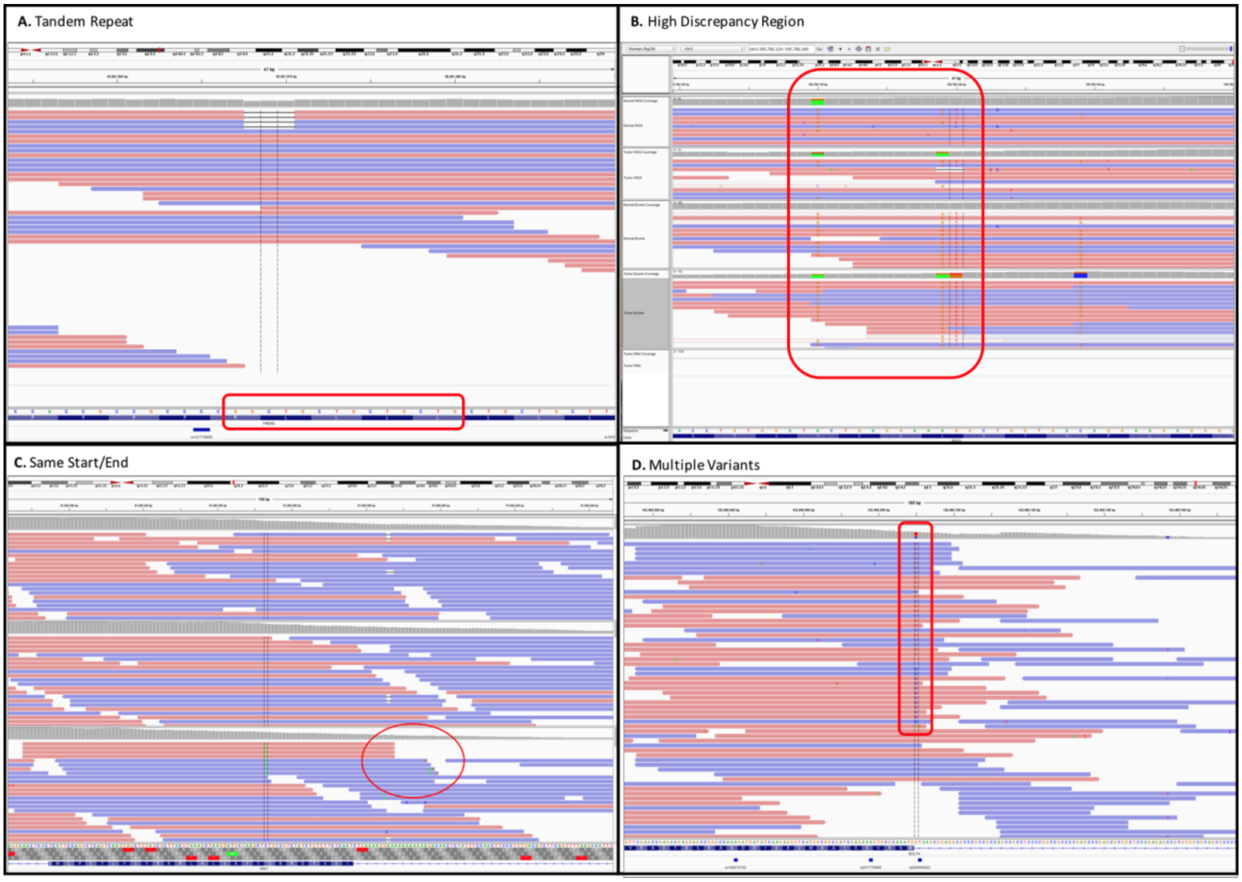
This will provide a "Common Snps 1.4.2" track that can be used to elucidate germline SNPs. If a variant in the tumor is also present in the Common SNPs track then it is most likely germline.

* If using GRCH37 you can view dbsnps 1.4.7 in the human genome, click:
"File" > "Load from Server" > "Annotations" > "dbsnps 1.4.2"

This will provide a "dbsnps 1.4.2" track that can be used to elucidate germline SNPs. If a variant in the tumor is also present in the Common SNPs track then it is most likely germline.

Failed Calls

The variant does not look real. Refer to the ‘Tags’ section to see reasons for labeling a variant as a false positive.

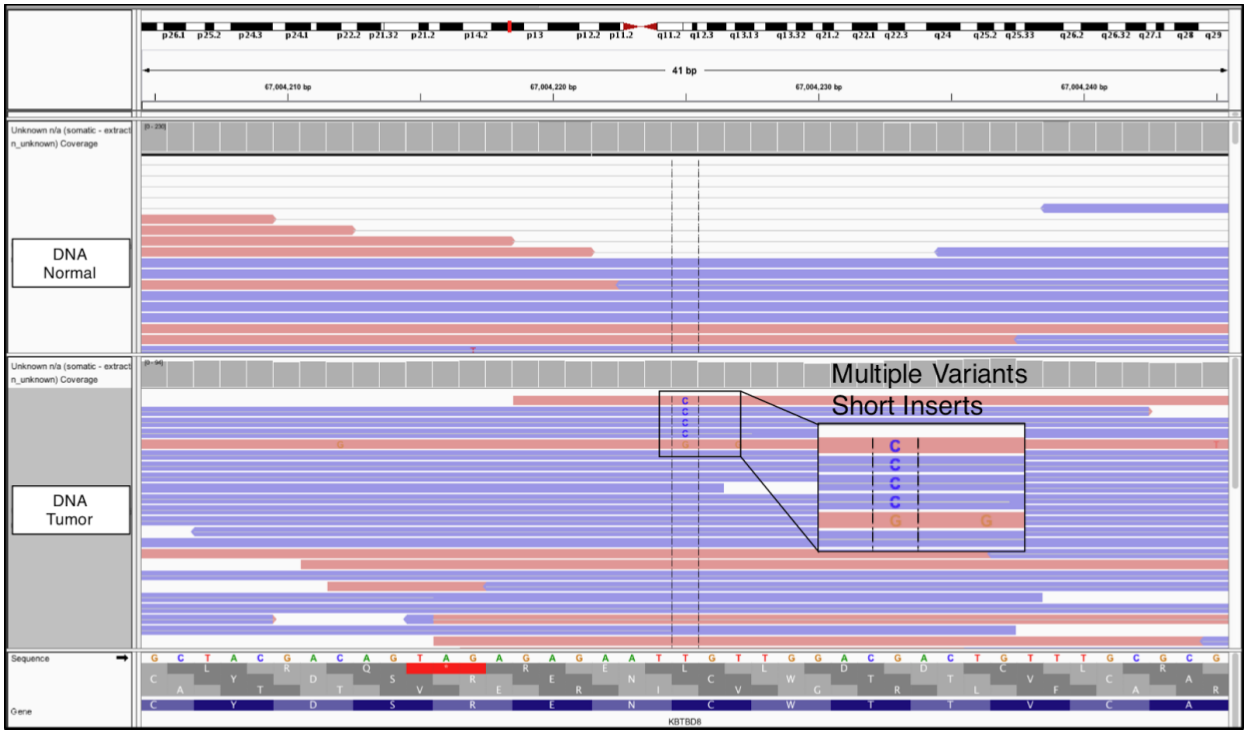


Helpful Hints:
Using Tags and Notes can help you in why variants were given failed calls.

Make sure you set up your IGV sessions to be able to easily pick up certain types of variants like “Directional” or “Short Inserts Only”. This requires coloring by read strand and viewing as pairs.

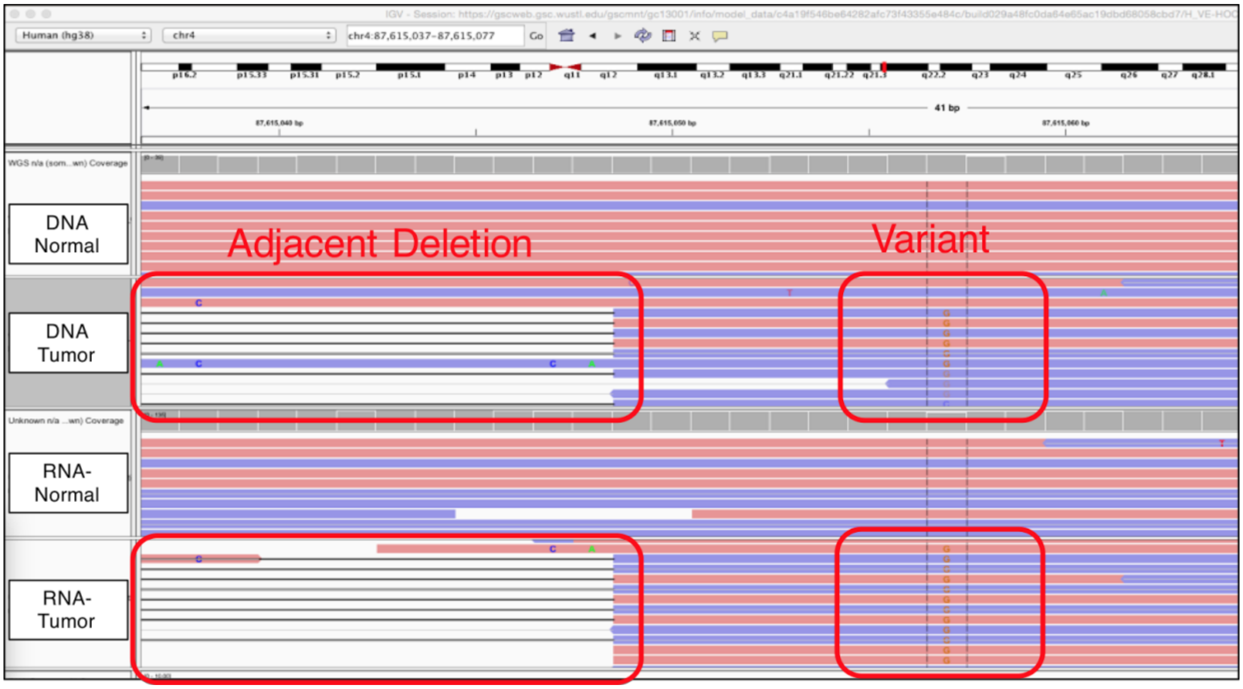
Ambiguous Calls

The variant could be a real somatic variant, but the reviewer is not confident due to features of the variant called and corresponding reads. Refer to ‘Tags’ to see reasons to define ambiguity. In this case, the variant has support from one normal read and 3 short insert read pairs (denoted by the white line in the middle of the read that shows that there are two overlapping reads from the same read pair at this position). The DNA fragment was too small for sequencing from each end to provide non-overlapping information about the sequence and commonly observed in DNA derived from archived (FFPE) material. The variant count for this position will be 7 (1 normal read and 6 short insert reads); however these reads were derived from a maximum of 4 independent DNA fragments. In this way short inserts can inflate variant counts and throw off variant callers.



Examples of Tags

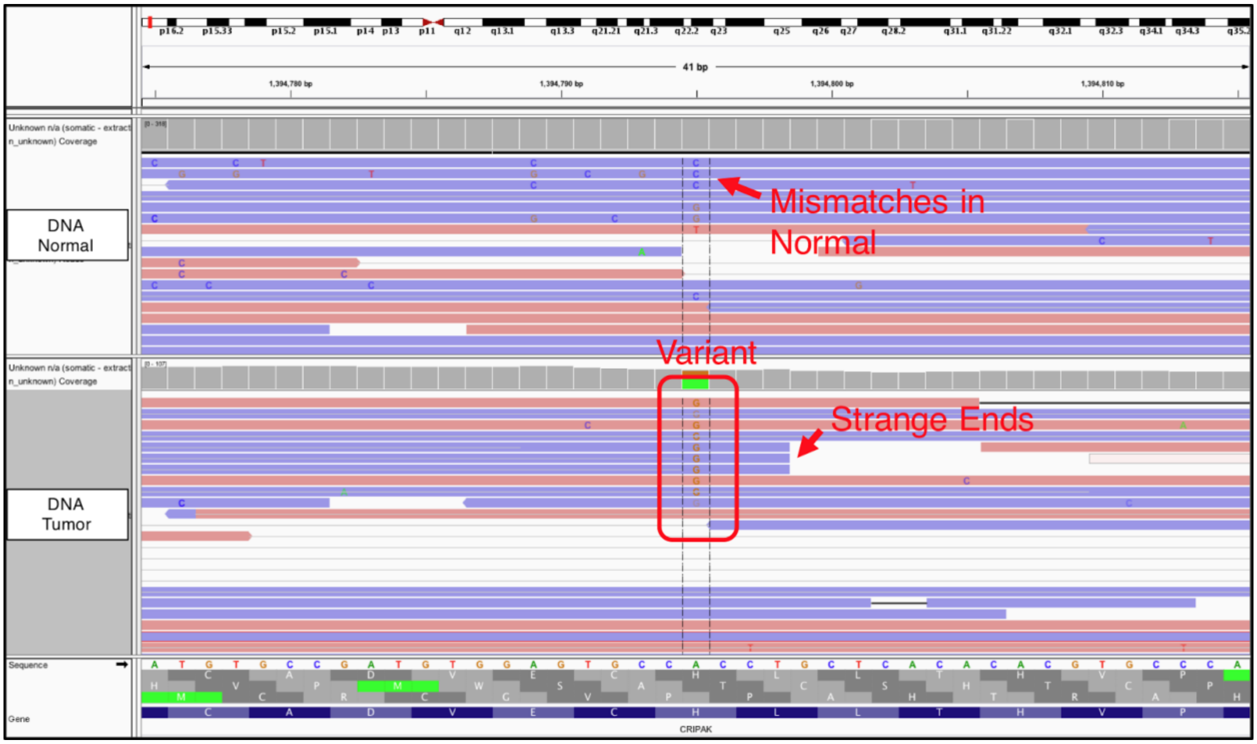
Adjacent Indel (AI)
An adjacent indel (insertion or deletion) is an artifact that is induced by a nearby true somatic variant. Typically, there is an insertion or a deletion near the artifact that induces a failure to align the reads properly to the reference genome.



Helpful Hints:
To adequately catch this artifact, it is necessary to zoom out on the IGV session to ensure that you visualize the adjacent insertion or deletion.

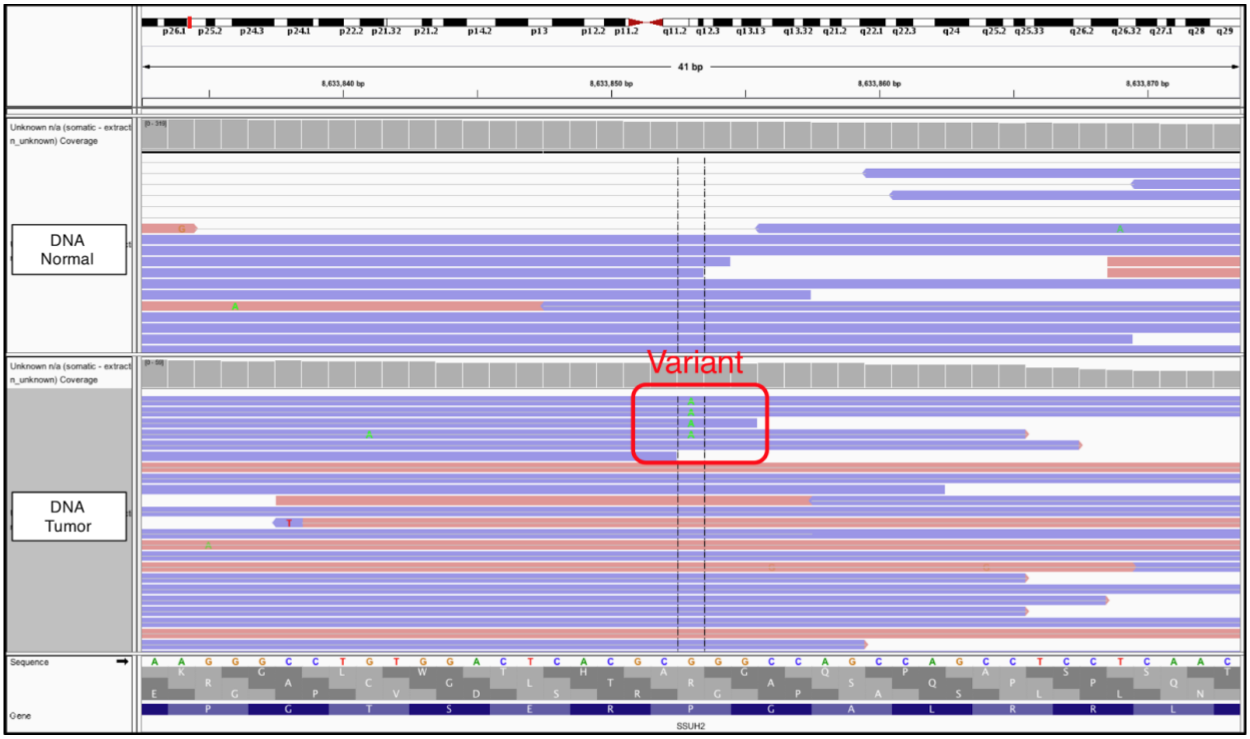
Ambiguous Other (AO)

Ambiguous other is used to define a variant surrounded by inconclusive genomic features that can’t be explained by the other tags available. For example, genomic regions with increased A/T or G/C content that are not contained within tandem or dinucleotide repeats. This could also be described as a low complexity region. If the Ambiguous Other tag is used, it is highly recommended to include a short description in the Notes section.



Directional (D)

A directional artifact is when the variant being evaluated can only be found on reads that are sequenced in either the forward or the reverse direction. Typically, this is caused by strand bias during sequencing. To properly visualize the directional artifacts, you must make sure the IGV tracks are colored by read strand.

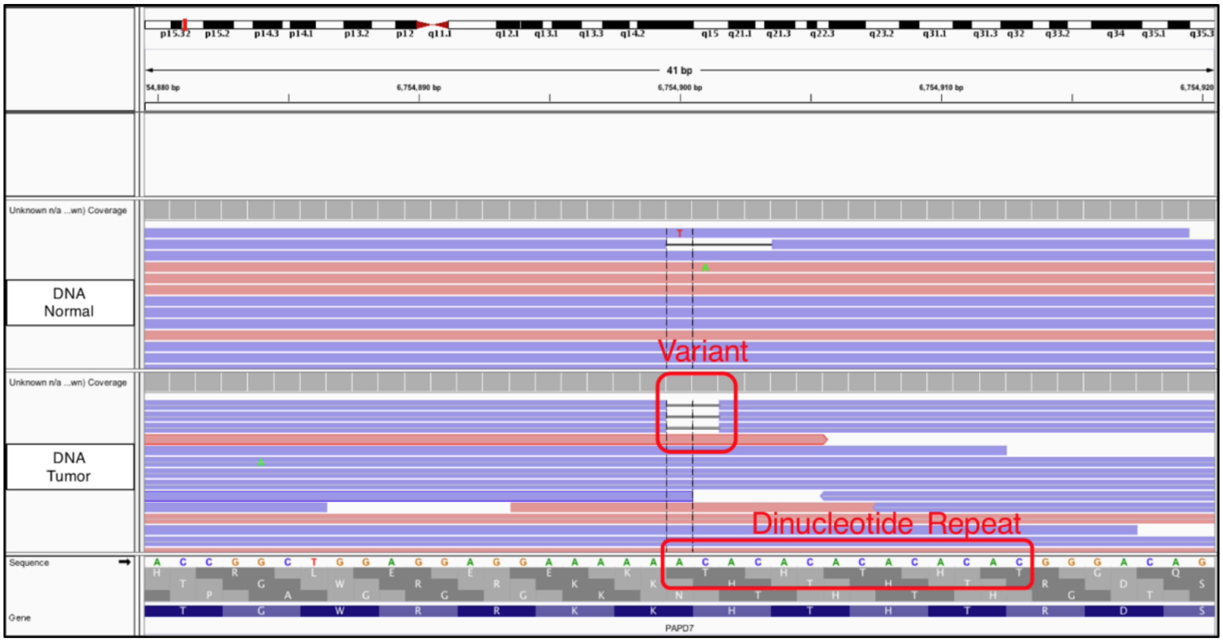

p26.1 p25.2 p24.3 p24.1 p22.2 p21.32 p21.2 p14.2 p13 p12.2 p11.2 q11.2 q12.3 q13.13 q13.32 q21.2 q22.1 q22.3 q24 q25.2 q25.33 q26.2 q26.32 q27.1 q28 q29
41 bp
8,633,840 bp
8,633,850 bp
8,633,860 bp
8,633,870 bp
Unknown n/a (somatic - extract n_unknown) Coverage
DNA Normal
Unknown n/a (somatic - extract n_unknown) Coverage
Variant
DNA Tumor
Sequence
Gene
SSUH2

Helpful Hints:

To adequately catch this artifact, it is necessary to color the alignments by readstrand:
Right click on the track you want to color > click *"Color alignments by"* > click *"read strand"*

Dinucleotide Repeats (DN)

The dinucleotide repeat artifact refers to when the reference sequence contains a region alternating between two nucleotides (e.g. ATATAT…). The variant being evaluated may be a SNV or indel directly adjacent or within this region and called due to misalignment or ends of reads. Repeat regions are areas where some sequencers, particularly those dependent on the polymerase enzyme, are prone to making mistakes. However, it is important to note that these are areas of normal human variation and real *de novo* mutations due to errors produced by polymerase during DNA replication. Other factors such as the size of the repeat, appearance in the normal and indels of varying length should be considered during evaluation.



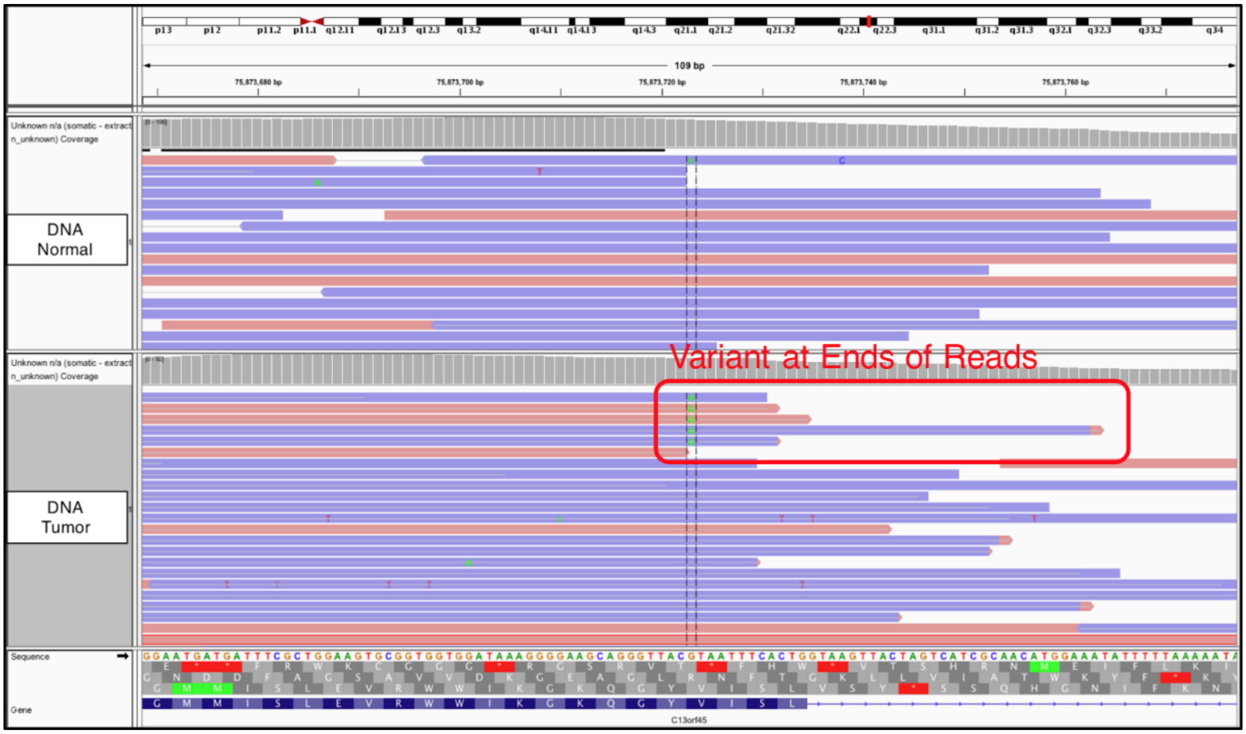
Helpful Hints:

Typically, these variants are small deletions or small insertions and they are usually visualized in the both the tumor tracks and the normal tracks.

Although the variant being evaluated may be a 2bp deletion, deletions of different sizes or even insertions are often observed with artifacts.

Ends (E)

The variant called is only present close to the end (within 10 base pairs) of the variant-supporting reads. There are no or few reads supporting the variant that contain the variant with high concordance with the surrounding reference sequence.

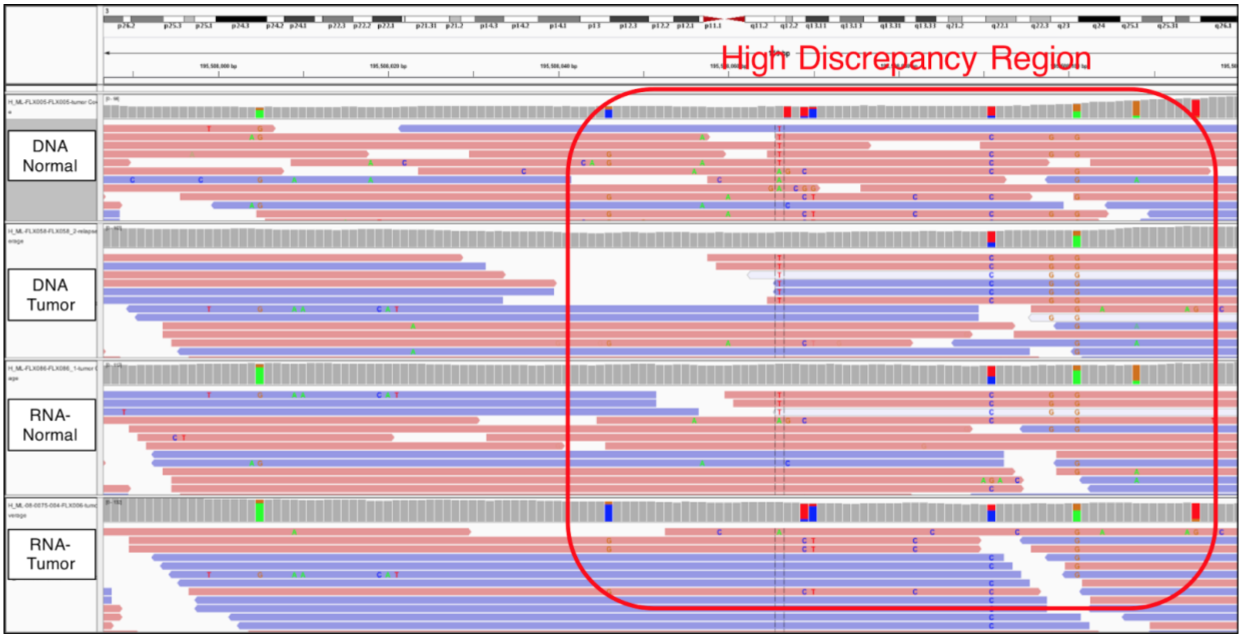


Helpful Hints:

To adequately catch this artifact, it is necessary to zoom out on the IGV session to ensure that you visualize the ends of the reads.

High Discrepancy Region (HDR)

When a variant is present in reads that also have many mismatches, and the mismatches across the whole track or in multiple tracks are the same, we use the High Discrepancy Region tag. HDR occurs when there are homologs across the genome and mis-mapping of the reads to homologs causes an apparent variant when they merely represent differences between the homologs and alignment artifacts.



Helpful Hints:

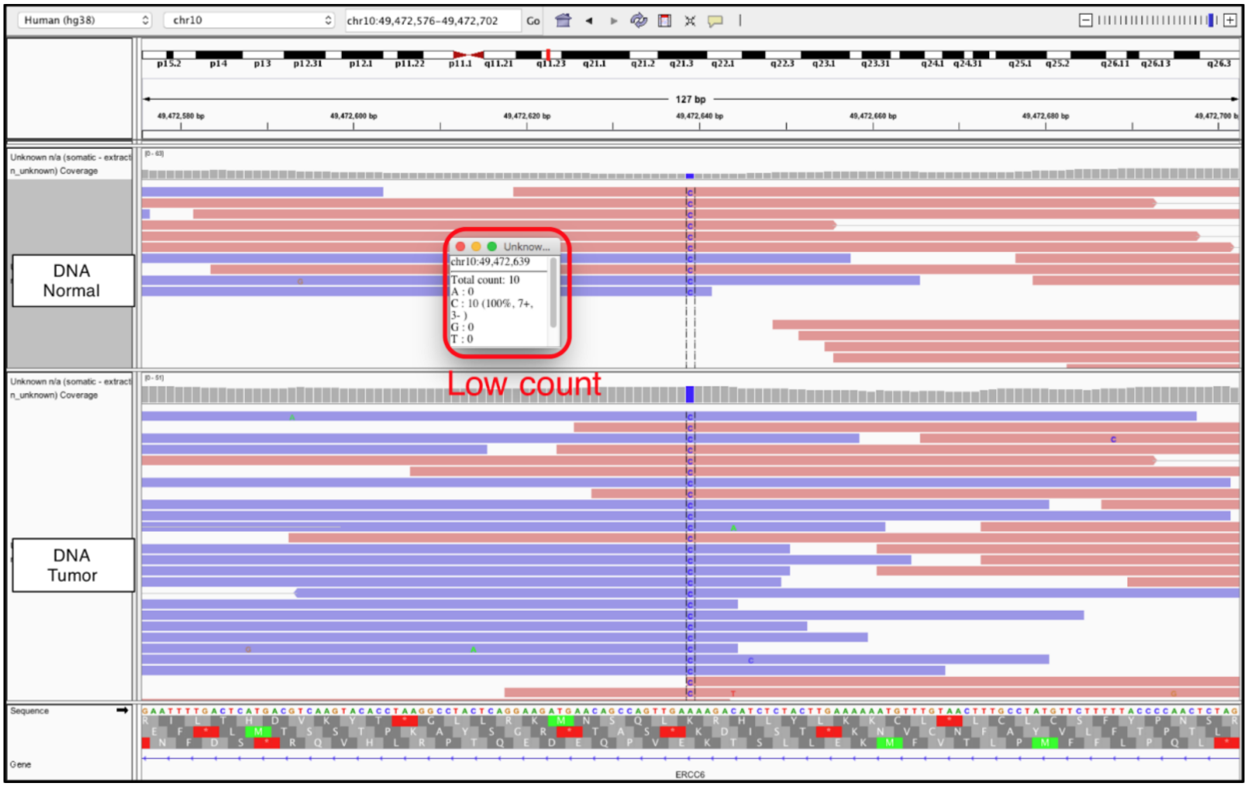
This is distinguished from Multiple Mismatches (MM) by the similarities of the mismatches across multiple tracks. In this example, all tracks contain the exact same mismatches at the same loci in the genome.

If there are multiple variants in a row that only 10-20 bases apart in the same gene then you should zoom out and make sure that you are not within an HDR.

Be sure that this is not due to adjacent SNPs. Clusters of common SNPs can happen and be real. It is unlikely that a truly somatic variant would be observed on both alleles of a heterozygous SNP; therefore, reads supporting a variant should also support only 1 allele of the heterozygous SNP (be in linkage with one allele). This is another instance when having a track identifying common polymorphisms can be helpful.

Low Count Normal (LCN)

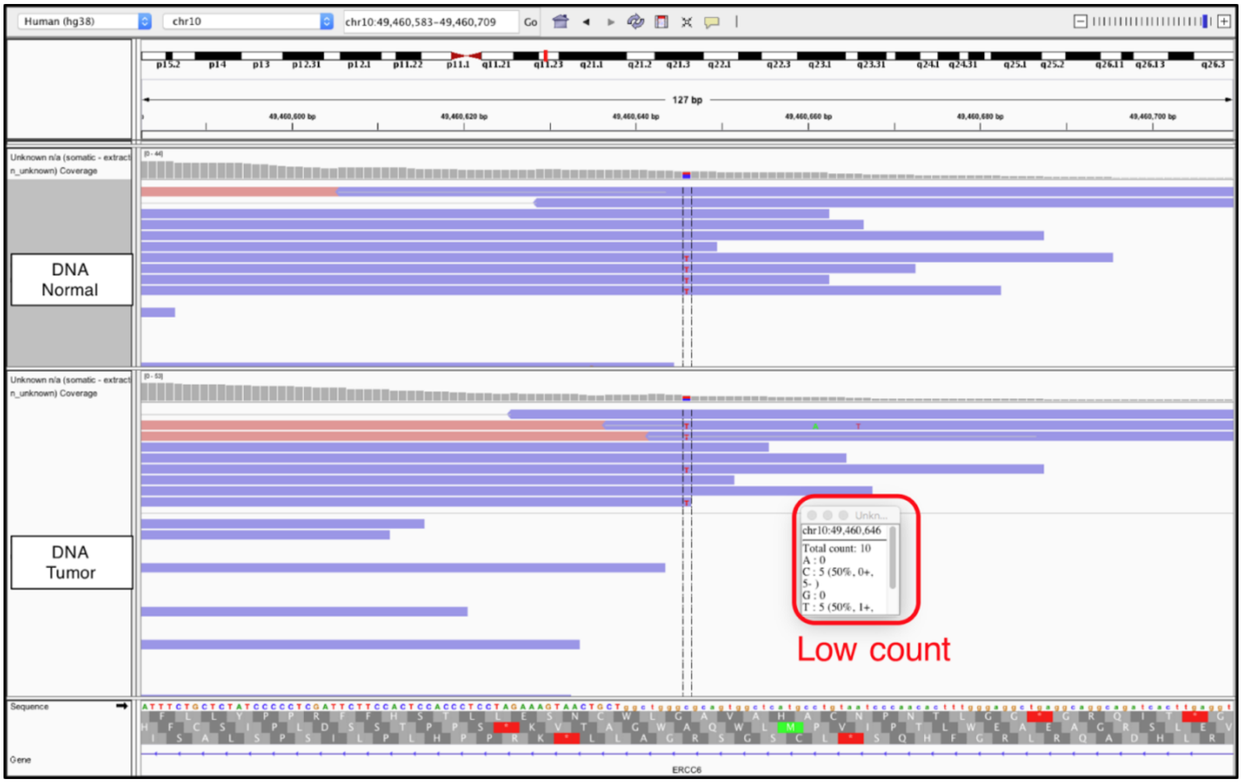
There is insufficient coverage in the normal sample to effectively compare the tumor sample. This can be assessed by clicking on the loci in the coverage track to reveal the total number of reads. Typically, we require at least 20X coverage in both tumor and normal to be sure that a manual review call is a true somatic variant.


Human (hg38)
chr10
chr10:49,472,576-49,472,702
Go
127 bp
Unknown n/a (somatic - extract n_unknown) Coverage
DNA Normal
Unknow...
Low count
Unknown n/a (somatic - extract n_unknown) Coverage
DNA Tumor
Sequence
Gene
ERCC6

chr10:49,472,639
Total count: 10
A : 0
C : 10 (100%, 7+, 3-)
G : 0
T : 0

Low Count Tumor (LCT)

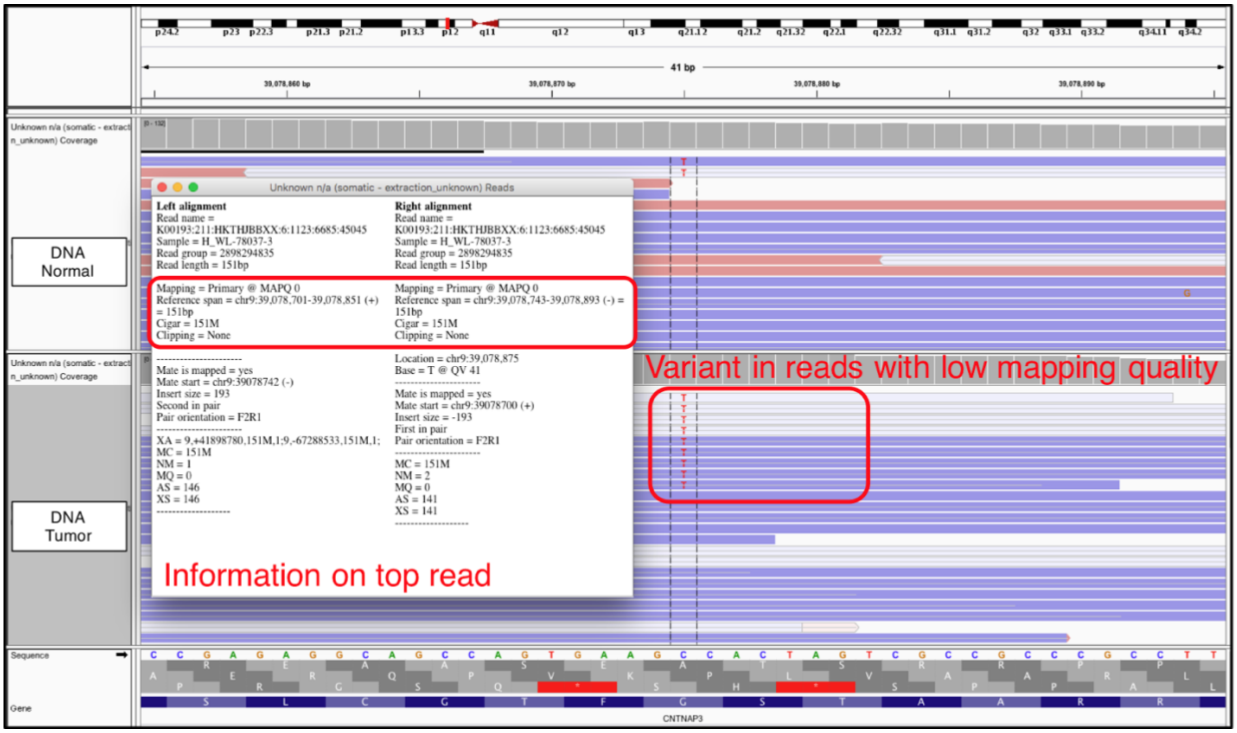
Coverage in the tumor is lower than the average. This threshold is experiment-specific. Low coverage at the site containing the variant will inflate the frequency associated with the variant called.


Human (hg38)
chr10
chr10:49,460,583-49,460,709
Go
127 bp
Unknown n/a (somatic - extract n_unknown) Coverage
DNA Normal
Unknown n/a (somatic - extract n_unknown) Coverage
DNA Tumor
Unkn...
Low count
Sequence
Gene
ERCC6

```
chr10:49,460,646
Total count: 10
A : 0
C : 5 (50%, 0+,
5- )
G : 0
T : 5 (50%, 1+,
```

Low Mapping (LM)

The mapping quality of a read, when the reads are "colored by strand," is indicated by the opacity of the read. Lighter reads have lower mapping quality, while darker reads have higher mapping quality. This value can also be obtained by clicking on a read for its information. By default, reads with a mapping quality of 0 are transparent, indicating they map to multiple regions in the genome and cannot be used to accurately call a somatic variant at this locus. Reads are colored if they have a mapping quality of >0. This threshold can be changed in View > Preferences > Alignments > Mapping Quality Threshold.



Unknown n/a (somatic - extraction_unknown) Reads

Left alignment
Read name =
K00193:211:HKTHJBBXX:6:1123:6685:45045
Sample = H_WL-78037-3
Read group = 2898294835
Read length = 151bp

Mapping = Primary @ MAPQ 0
Reference span = chr9:39,078,701-39,078,851 (+) = 151bp
Cigar = 151M
Clipping = None

Mate is mapped = yes
Mate start = chr9:39078742 (-)
Insert size = 193
Second in pair
Pair orientation = F2R1

XA = 9,+41898780,151M,1;9,-67288533,151M,1;
MC = 151M
NM = 1
MQ = 0
AS = 146
XS = 146

Right alignment
Read name =
K00193:211:HKTHJBBXX:6:1123:6685:45045
Sample = H_WL-78037-3
Read group = 2898294835
Read length = 151bp

Mapping = Primary @ MAPQ 0
Reference span = chr9:39,078,743-39,078,893 (-) = 151bp
Cigar = 151M
Clipping = None

Location = chr9:39,078,875
Base = T @ QV 41

Mate is mapped = yes
Mate start = chr9:39078700 (+)
Insert size = -193
First in pair
Pair orientation = F2R1

MC = 151M
NM = 2
MQ = 0
AS = 141
XS = 141

Information on top read

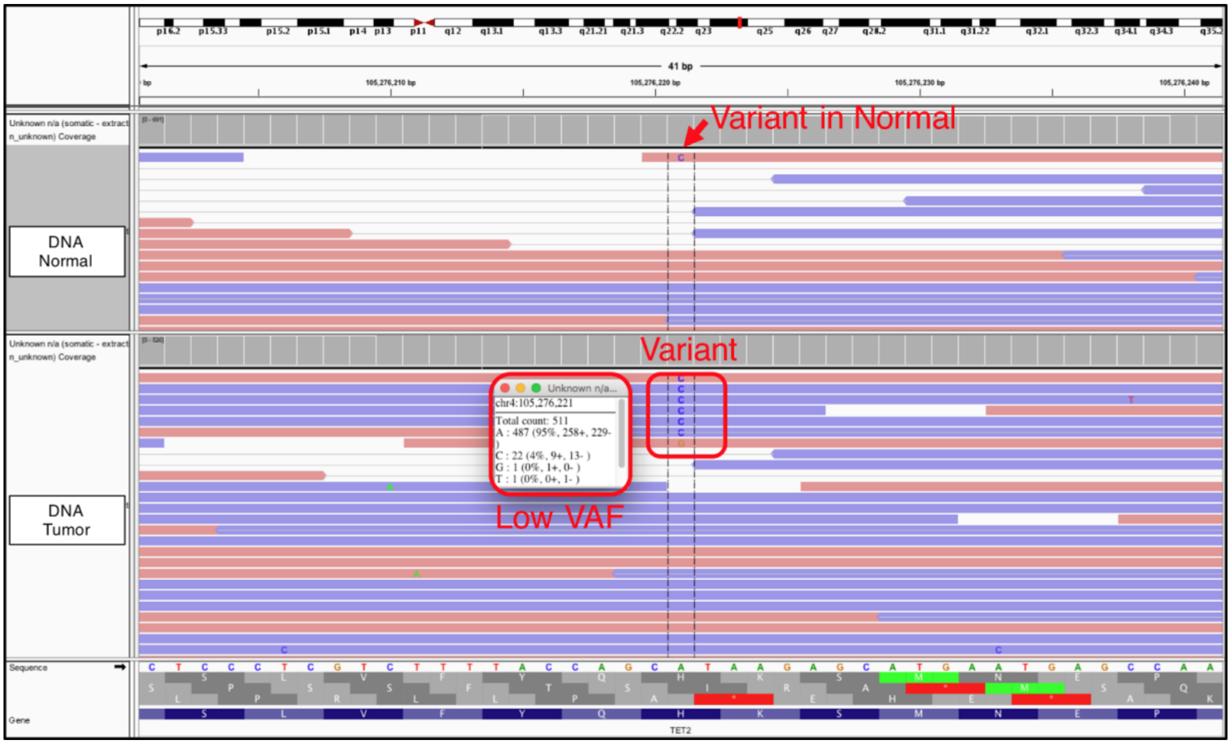
Variant in reads with low mapping quality

Helpful Hints:

Regions with numerous reads with a mapping quality of 0 are often associated with genes with other homologs in the genome and therefore HDRs (see this call type above) and result in low mapping quality reads in both the normal and tumor.

Low Variant Frequency (LVF)

The Low Variant Frequency tag is used when there are some reads of support for the variant, but the variant allele frequency (VAF) is relatively low. To quickly assess the VAF, you can click on the coverage track to pull up the total read counts, the number of reads for each base and the support for different directions. Usually, the LVF must be used in conjunction with other tags to fail the variant.



chr4:105,276,221

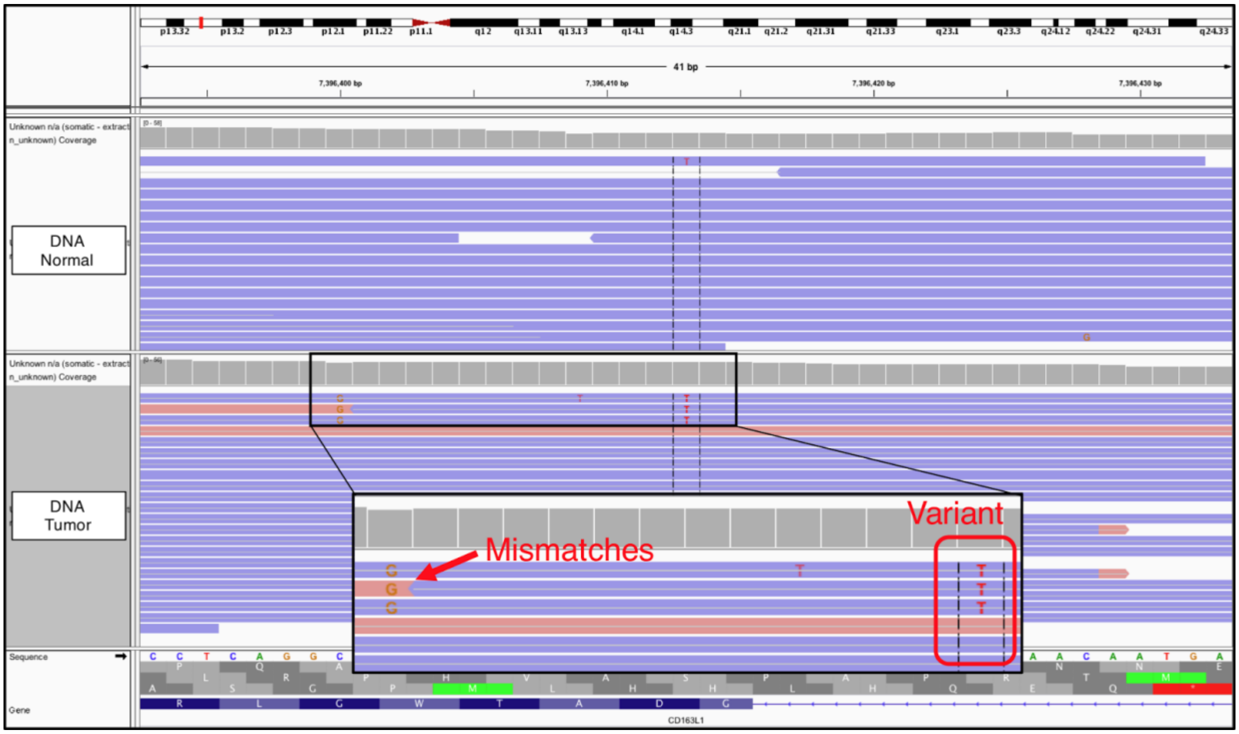
Total count: 511
A : 487 (95%, 258+, 229-
)
C : 22 (4%, 9+, 13-)
G : 1 (0%, 1+, 0-)
T : 1 (0%, 0+, 1-)

Helpful Hints:

The coverage track will be colored according to base when a variant is present at a 15% VAF, by default. This cutoff can be changed by altering this threshold in View > Preferences > Alignments > Coverage allele-fraction threshold. This can be particularly helpful with high depth samples.

Multiple Mismatches (MM)

The multiple mismatches tag is used when the reads that contain the variant have other mismatched base pairs, indicating a less trustworthy read. This is like the HDR tag; however, it can include mismatches that are not exactly the same across the reads.

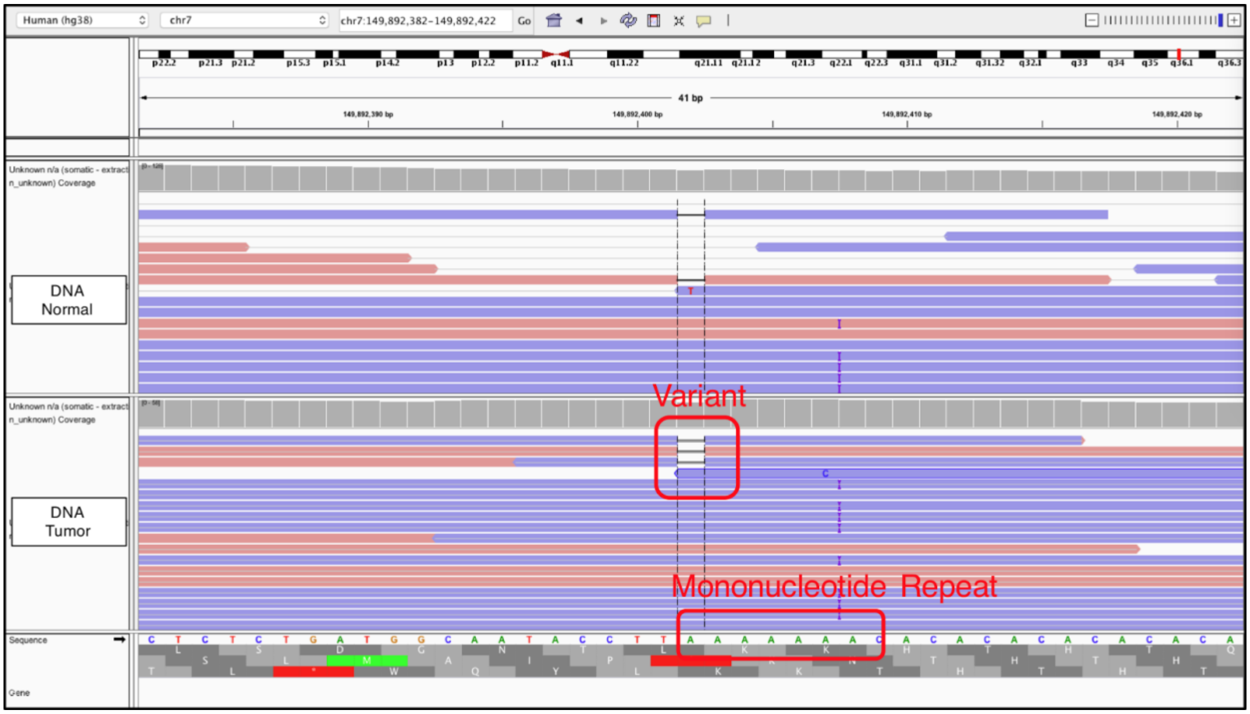


Helpful Hints:

The mismatch base color becomes more transparent as the base quality gets lower, so if the adjacent mismatch is dark in color, then you have reduced confidence in that read being properly sequenced and/or aligned to the reference genome.

Mononucleotide (MN)

The mononucleotide repeat artifact refers to when the reference sequence contains a region of a single nucleotide (e.g. AAAAAAA…). The variant being evaluated may be a SNV or indel directly adjacent or within this region and called due to misalignment or ends of reads. Repeat regions are areas where some sequencers, particularly those dependent on the polymerase enzyme, are prone to making mistakes. However, it is important to note that these are areas of normal human variation and real *de novo* mutations due to errors produced by polymerase during DNA replication. Like the Dinucleotide (DN) tag, other factors such as the size of the repeat, appearance in the normal and indels of varying length should be considered during evaluation.


Human (hg38)
chr7
chr7:149,892,382-149,892,422
Go
p22.2 p21.3 p21.2 p15.3 p15.1 p14.2 p13 p12.2 p11.2 q11.1 q11.22 q21.11 q21.12 q21.3 q22.1 q22.3 q31.1 q31.2 q31.32 q32.1 q33 q34 q35 q36.1 q36.3
41 bp
149,892,390 bp
149,892,400 bp
149,892,410 bp
149,892,420 bp
Unknown n/a (somatic - extract n_unknown) Coverage
DNA Normal
Unknown n/a (somatic - extract n_unknown) Coverage
DNA Tumor
Variant
Mononucleotide Repeat
Sequence
Gene

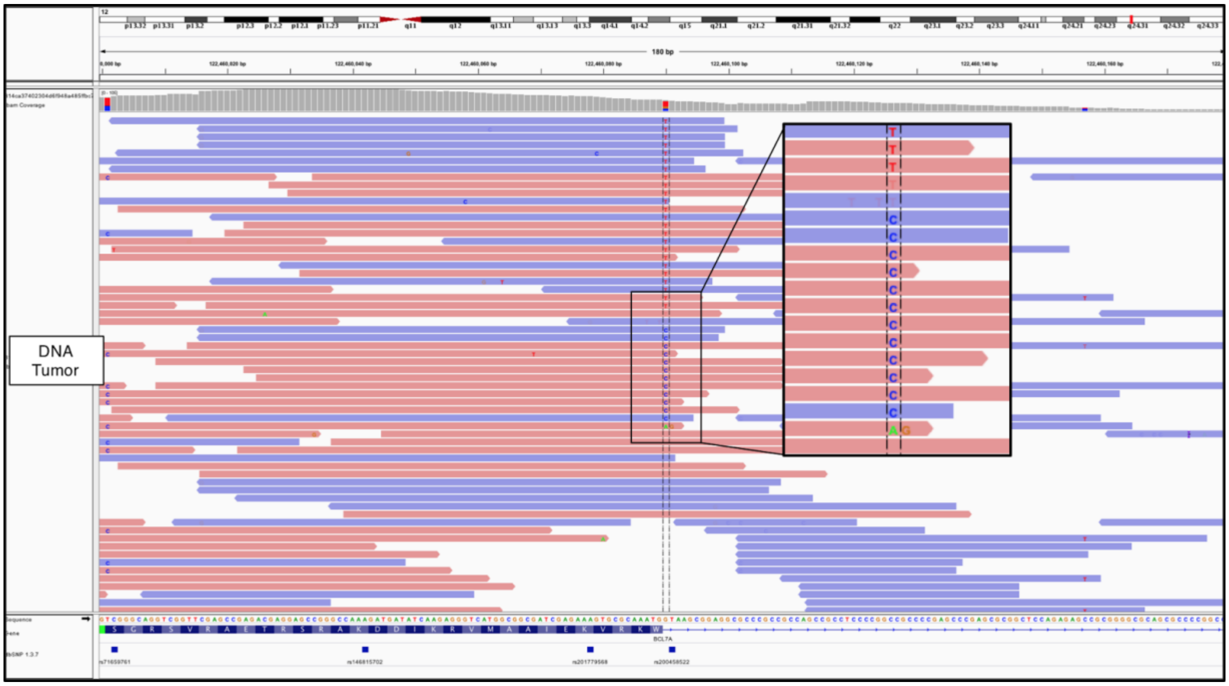
Helpful Hints:

Typically, these variants are small deletions or small insertions and they are usually visualized in the both the tumor tracks and the normal tracks.

Although the variant being evaluated may be a 1bp deletion, deletions of different sizes or even insertions are often observed with artifacts.

Multiple Variants (MV)

The Multiple Variants artifact is used If the variant called has reads supporting multiple different variants at the same loci. In the example shown, there were calls for all four different nucleotides at the same loci, making it an unreliable call.



Helpful Hints:

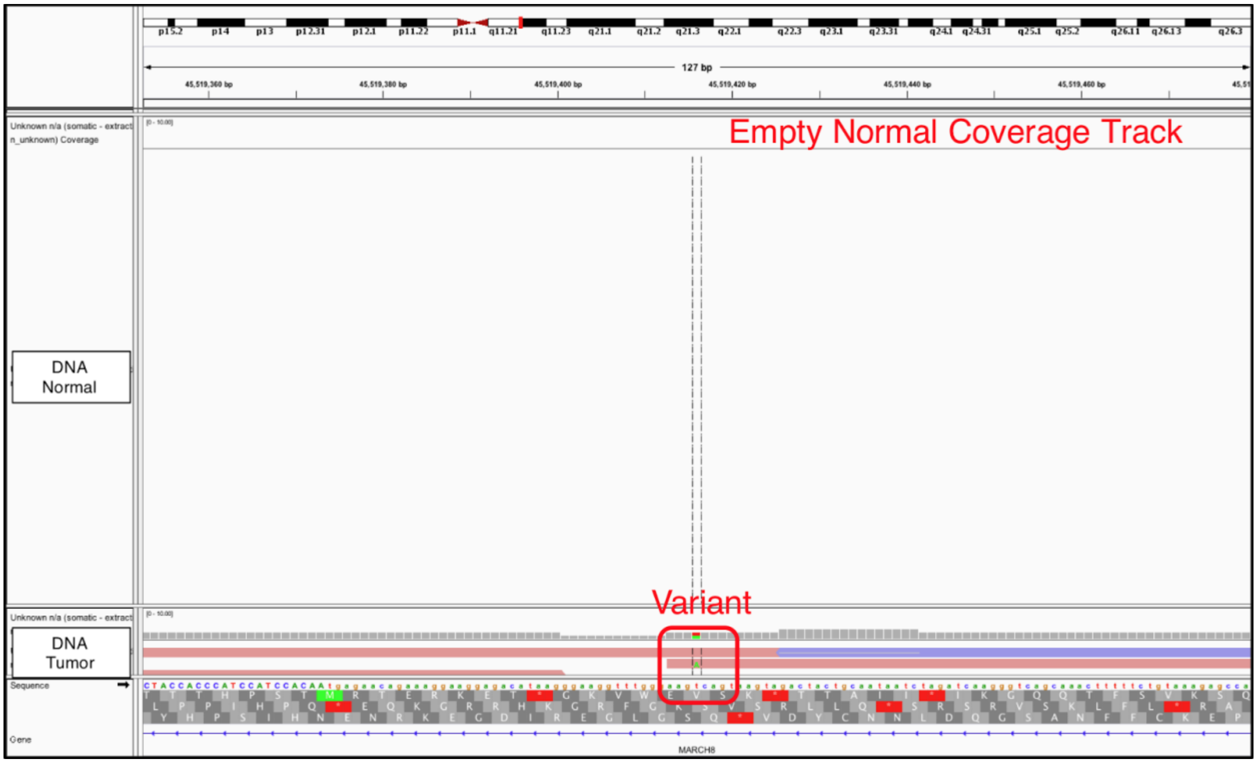
Make sure you scroll all the way to the bottom of the track to visualize all the reads. It is not enough just to rely on the coverage coloring as there might be multiple variants that have a VAF too small to be represented in the coverage bar.

Clicking the coverage track will give you the relative abundance of each base at that site.

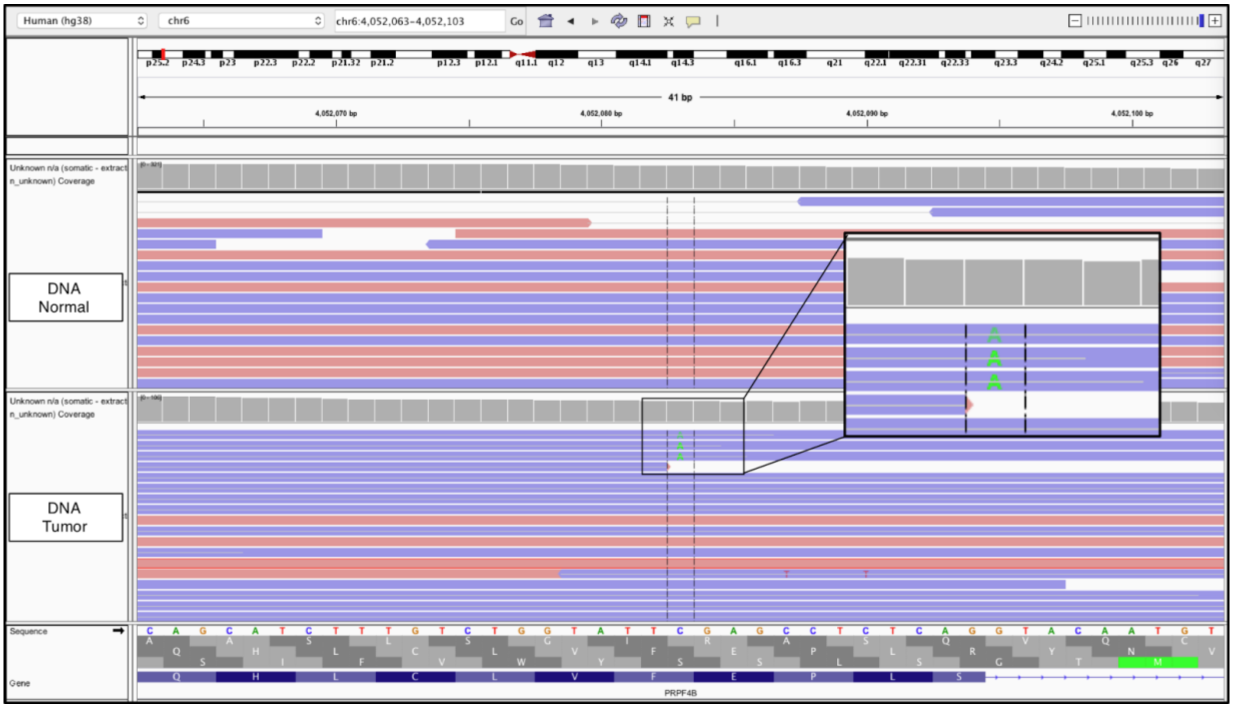
For very deep data, multiple variants due to random error will start to accumulate. The relative abundance of each base should be considered in cases with deep coverage.

No Coverage in Normal (NCN)

There is no coverage in the normal sample to effectively compare the tumor sample. This can occur when you do not have a normal track for comparison or you do not have any reads in the normal track that help assess if the variant is true. Typically, we require at least 20X coverage to be sure that a call is truly somatic.



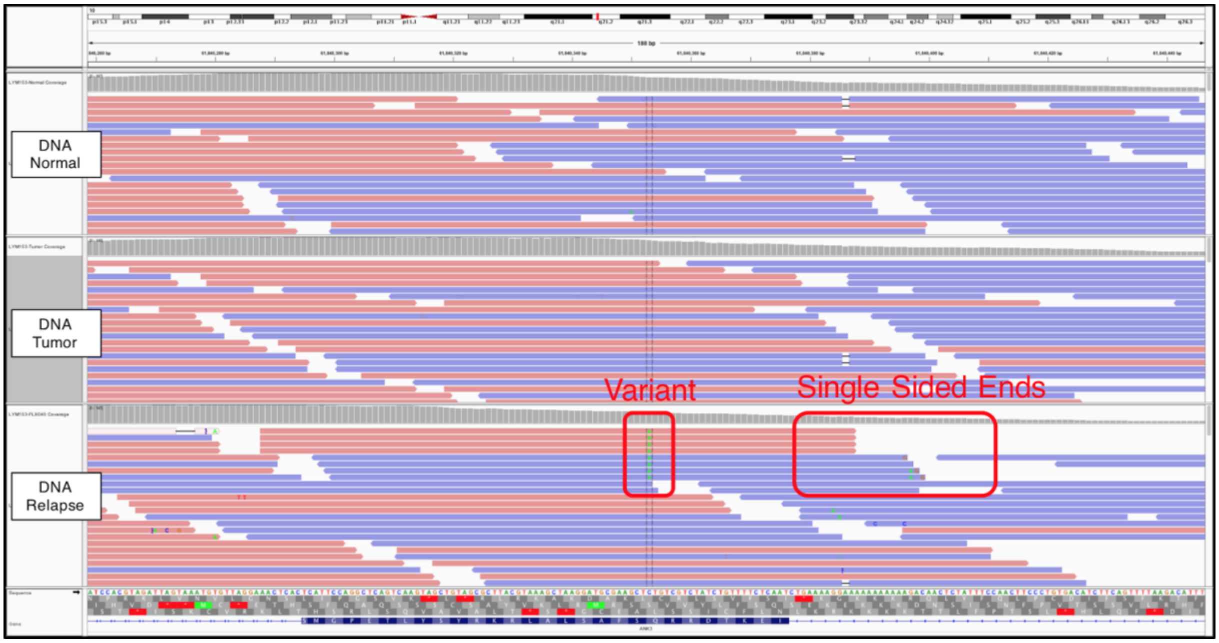
Short Inserts (SI) and Short Inserts Only (SIO)
Short inserts refer to instances when the DNA fragment is small enough that sequencing from each end of the molecule results in overlapping reads. Variants supported by reads produced from these short fragments result in the appearance of 2 reads supporting this variant when in reality they represent a single molecule of DNA, inflating the VAF of the variant. Support for the called variant is present in both reads of a read pair and other reads (SI) or support for the called variant is ONLY present in paired reads with a short insert size (SIO). That is, the variant appears in the overlapping region of the two read fragments, indicated by the line through the middle of the reads. These are prevalent in data derived from archival material (FFPE) or other source material with short DNA fragments (cell-free DNA). Artifacts are generally present at lower frequencies, and are present in two or three read pairss, or (four to six readfragments in total).


Human (hg38)
chr6
chr6:4,052,063-4,052,103
Go
p25.2 p24.3 p23 p22.3 p22.2 p21.32 p21.2 p12.3 p12.1 q11.1 q12 q13 q14.1 q14.3 q16.1 q16.3 q21 q22.1 q22.31 q22.33 q23.3 q24.2 q25.1 q25.3 q26 q27
41 bp
4,052,070 bp
4,052,080 bp
4,052,090 bp
4,052,100 bp
Unknown n/a (somatic - extraction_unknown) Coverage
DNA Normal
Unknown n/a (somatic - extraction_unknown) Coverage
DNA Tumor
Sequence
Gene
PRPF4B

Helpful Hints:
To visualize short insert variants you must view the tracks as pairs. Short inserts will be condensed and a grey line indicates the areas where these reads overlap.

Same StartEnd (SSE)

The Same Sitart/End artifact occurs when the variant is only contained by reads that start and stop at the same loci. This is typically attributed to a variant called in multiple reads created from the same molecule during the amplification process in the genome sequencing protocol, but erroneously not removed during read de-duplication.

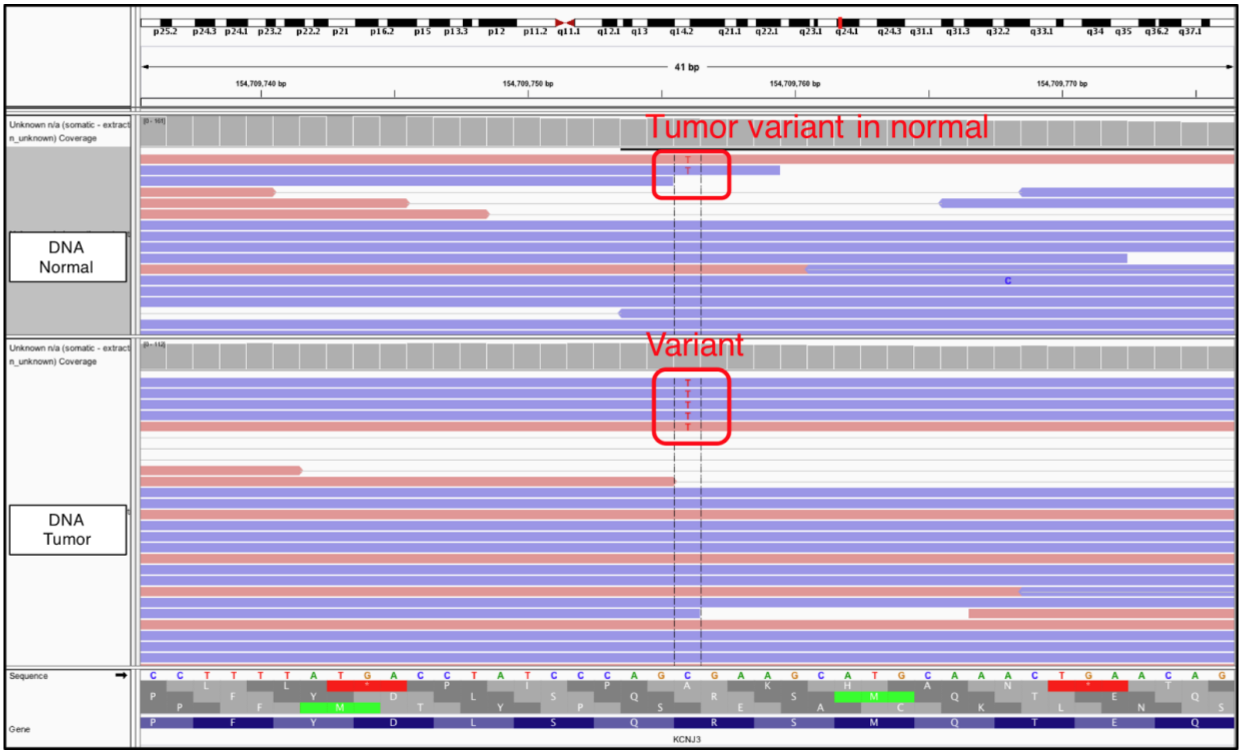


Helpful Hints:

Make sure you sort by the variant and zoom out to show the entire length of the reads. This will allow you to visualize if the read ends line up.

Tumor in Normal (TN)

Tumor in normal can occur if the variant has reads of support in the normal tracks. This is a common occurrence in blood tumors (leukemia and lymphoma) as well as tumors that are highly metastatic. Although this might not be a reason for failing the variant call, it can be used in cases of ambiguity to denote reasons for potential failure. Variants created by sequencing or alignment artifacts will often occur in both the tumor and the normal sample.



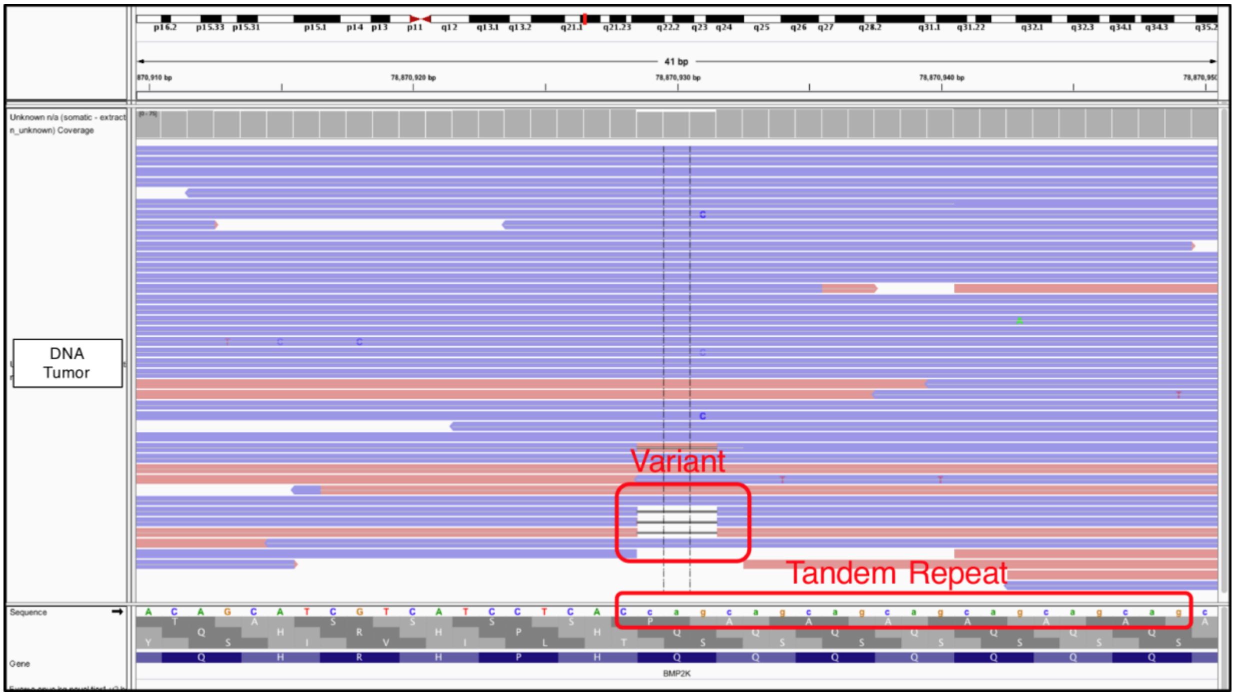
Helpful Hints:

This does not occur in all hematopoietic tumors but is likely when tumor cells are circulating in the blood stream such as acute myeloid leukemias with high blast counts.

Evaluating other normal samples from your cohort can help differentiate sequencing and pipeline artifacts from normal contamination.

Tandem Repeat (TR)

The tandem repeat artifact refers to when the reference sequence contains a region alternating between three nucleotides (e.g. CAGCAGCAG…). The variant being evaluated may be a SNV or indel directly adjacent or within this region and called due to misalignment or ends of reads. Repeat regions are areas where some sequencers, particularly those dependent on the polymerase enzyme, are prone to making mistakes. However, it is important to note that these are areas of normal human variation and real *de novo* mutations due to errors produced by polymerase during DNA replication. Like the Mononucleotide (MN) and Dinucleotide (DN) tags, other factors such as the size of the repeat, appearance in the normal and indels of varying length should be considered during evaluation.



Helpful Hints:

Typically, these variants are small deletions or small insertions and they are usually visualized in the both the tumor tracks and the normal tracks.

Although the variant being evaluated may be a 3bp deletion, deletions of different sizes or even insertions are often observed with artifacts.

Appendix 2: Features Used in Manual Review Classifier

Feature	*Description*
disease_AML	Disease label = AML
disease_AML	Disease label = AML
disease_GST	Disease label = GIST
disease_MPNST	Disease label = Malignant Peripheral Nerve Sheath Tumors
disease_SCLC	Disease label = Breast Cancer
disease_breast	Disease label = Colorectal Cancer
disease_colorectal	Disease label = Glioblastoma
disease_glioblastoma	Disease label = Lymphoma
disease_lymphoma	Disease label = AML
disease_melanoma	Disease label = Melanoma
normal_VAF	Variant allele frequence of normal reads
normal_depth	Depth of normal reads
normal_other_bases_count	Count of non ref/var reads in normal
normal_ref_avg_basequality	Average basequality of reference reads in normal
normal_ref_avg_clipped_length	Average clipped length of reference reads in normal
normal_ref_avg_distance_to_effective_3p_end	Average distance to 3' end of reference reads in normal
normal_ref_avg_distance_to_q2_start_in_q2_reads	Average distance to a run where Illumina's Read Segment Quality Control Indicator=2. False positives more commonly appear near these regions.
normal_ref_avg_mapping_quality	Average mapping quality for reference reads in normal
normal_ref_avg_num_mismaches_as_fraction	Average number of mismatches on these reads per base
normal_ref_avg_pos_as_fraction	Average position on the read as a fraction. This value is normalized to the center of the read
normal_ref_avg_se_mapping_quality	Mean single ended mapping quality
normal_ref_avg_sum_mismatch_qualities	Average sum of the base qualities of mismatches in the reads
normal_ref_count	Counts of reference basepairs in normal
normal_ref_num_minus_strand	Average minus strand reference reads in normal
normal_ref_num_plus_strand	Average plus strands reference reads in normal
normal_ref_num_q2_containing_reads	Number of reads containing a run where Illumina's Read Segment Quality Control Indicator=2. False positives more commonly appear near these regions.
normal_var_avg_basequality	Average basequality of variant reads in normal
normal_var_avg_clipped_length	Average clipped length of var reads in normal
normal_var_avg_distance_to_effective_3p_end	Average distance to 3' end of var reads in normal
normal_var_avg_distance_to_q2_start_in_q2_reads	Average distance of position (as fraction of unclipped read length) to the start of the q2 run
normal_var_avg_mapping_quality	Average mapping quality for var reads in normal
normal_var_avg_num_mismaches_as_fraction	Average number of mismatches on these reads per base

normal_var_avg_pos_as_fraction	Average position on the read as a fraction. This value is normalized to the center of the read
normal_var_avg_se_mapping_quality	Mean single ended mapping quality
normal_var_avg_sum_mismatch_qualities	Average sum of the base qualities of mismatches in the reads
normal_var_count	Average count for var reads in normal
normal_var_num_minus_strand	Average minus strand var reads in normal
normal_var_num_plus_strand	Average plus strand var reads in normal
normal_var_num_q2_containing_reads	Number of reads containing a run where Illumina's Read Segment Quality Control Indicator=2. False positives more commonly appear near these regions.
reviewer_1	Call from reviewer 1
reviewer_2	Call from reviewer 2
reviewer_3	Call from reviewer 3
reviewer_4	Call from reviewer 4
tumor_VAF	Variant allele frequence of tumor reads
tumor_depth	Depth of tumor reads
tumor_other_bases_count	Average count of non ref/var bases in tumor
tumor_ref_avg_basequality	Average basequality of reference reads in tumor
tumor_ref_avg_clipped_length	Average clipped length of ref reads in tumor
tumor_ref_avg_distance_to_effective_3p_end	Average distance to effective 3' end in ref tumor
tumor_ref_avg_distance_to_q2_start_in_q2_reads	Average distance to a run where Illumina's Read Segment Quality Control Indicator=2. False positives more commonly appear near these regions.
tumor_ref_avg_mapping_quality	Average mapping quality of reference reads in tumor
tumor_ref_avg_num_mismaches_as_fraction	Average number of mismatches on these reads per base
tumor_ref_avg_pos_as_fraction	Average position on the read as a fraction. This value is normalized to the center of the read
tumor_ref_avg_se_mapping_quality	Mean single ended mapping quality
tumor_ref_avg_sum_mismatch_qualities	Average sum of the base qualities of mismatches in the reads
tumor_ref_count	Counts of reference basepairs for tumor sample
tumor_ref_num_minus_strand	Number of negative read strands in tumor ref calls
tumor_ref_num_plus_strand	Number of positive read strands in tumor ref calls
tumor_ref_num_q2_containing_reads	Number of reads containing a run where Illumina's Read Segment Quality Control Indicator=2. False positives more commonly appear near these regions.
tumor_var_avg_basequality	Average basequality of variant reads in tumor
tumor_var_avg_clipped_length	Average clipped length of var reads in tumor
tumor_var_avg_distance_to_effective_3p_end	Average distance to 3' end of var reads in tumor
tumor_var_avg_distance_to_q2_start_in_q2_reads	Average distance to a run where Illumina's Read Segment Quality Control Indicator=2. False positives more commonly appear near these regions.
tumor_var_avg_mapping_quality	Average mapping quality of a tumor variant reads
tumor_var_avg_num_mismaches_as_fraction	Average number of mismatches on these reads per base
tumor_var_avg_pos_as_fraction	Average position on the read as a fraction. This value is normalized to the center of the read

tumor_var_avg_se_mapping_quality	Mean single ended mapping quality
tumor_var_avg_sum_mismatch_qualities	Average sum of the base qualities of mismatches in the reads
tumor_var_count	Counts of variant basepairs for tumor sample
tumor_var_num_minus_strand	Number of negative read strands in tumor var calls
tumor_var_num_plus_strand	Number of positive read strands in tumor var calls
tumor_var_num_q2_containing_reads	Number of reads containing a run where Illumina's Read Segment Quality Control Indicator=2. False positives more commonly appear near these regions.
disease_GST	Disease label = GIST

Appendix 3: High overlap between false negatives recovered by the machine learning classifiers and CIViC annotations.

For each table, the ‘Confidence’ is the well-scaled probability of the call being labeled as somatic, the ‘MR Call’ is the call made by the human manual reviewer, the ‘model Call’ is the call made by the deep learning model, and ‘CIViC Score’ represents the quality of evidence within CIViC based on number of evidence statements and trust rating. **A**. This table shows false negative calls that had overlap with CIViC annotations. **B**. This table shows false positive calls that had overlap with CIViC annotations.

A

False Negative Overlap with CIViC

Chr	Start	Stop	Ref	Var	Gene	Disease	Confidence	MR Call	Modell Call	Evidence Items	CIViC Score
1	115258747	115258747	C	G	NRAS	AML	0.890697241	f	s	17	174
2	209113112	209113112	C	T	IDH1	AML	0.835653961	f	s	11	16
1	115258747	115258747	C	T	NRAS	AML	0.931768775	a	s	17	174
19	33793237	33793238	-	CCGC	CEBPA	AML	0.466763496	a	s	8	110
17	7578196	7578196	A	T	TP53	AML	0.498811215	f	s	2	28
17	74732959	74732959	G	T	SRSF2	AML	0.582512021	a	s	1	20
12	49424359	49424359	C	A	KMT2D	breast	0.49920553	a	s	1	12
12	49432313	49432313	C	T	KMT2D	breast	0.885561228	a	s	1	12
13	32972788	32972788	C	T	BRCA2	breast	0.611487806	a	s	18	228
21	36228727	36228727	G	A	RUNX1	breast	0.872433066	a	s	6	88
12	49431112	49431112	G	A	KMT2D	breast	0.700233936	a	s	1	12
12	49437569	49437569	T	C	KMT2D	breast	0.571465731	a	s	1	12
12	49440556	49440556	C	T	KMT2D	breast	0.523925006	a	s	1	12
13	49037971	49037971	G	A	RB1	breast	0.991140544	a	s	3	20
17	37872148	37872148	A	G	ERBB2	breast	0.65927732	a	s	2	25
17	41258453	41258453	A	G	BRCA1	breast	0.760988951	a	s	11	148
12	49445051	49445051	G	A	KMT2D	breast	0.483478546	a	s	1	12

12	49426772	49426777	GCTGCT	-	KMT2D	breast	0.649460375	a	s	1	12
12	49431959	49431959	C	T	KMT2D	breast	0.733782172	a	s	1	12
17	29685594	29685594	G	A	NF1	breast	0.597693563	a	s	8	56
17	74732942	74732942	T	C	SRSF2	AML	0.512601137	a	s	1	20
4	106190898	106190898	C	A	TET2	AML	0.947028816	a	s	6	96
4	55598158	55598158	C	A	KIT	AML	0.91864562	a	s	3	24
9	9425502	9425502	G	T	PTPRD	SCLC	0.675933957	a	s	1	6
19	1220655	1220655	G	-	STK11	SCLC	0.501361549	a	s	6	44
9	139418369	139418369	C	G	NOTCH1	SCLC	0.924308896	a	s	1	4
3	178936091	178936091	G	A	PIK3CA	breast	0.937070668	a	s	37	195
3	178936103	178936103	G	A	PIK3CA	breast	0.94812727	a	s	19	146
7	124538437	124538437	A	-	POT1	breast	0.708491325	f	s	1	12
22	42523636	42523636	C	A	CYP2D6	breast	0.510425448	a	s	1	16
12	49431045	49431045	G	A	KMT2D	breast	0.883784831	a	s	1	12
22	42524327	42524327	A	G	CYP2D6	breast	0.391751111	a	s	1	16
22	42523636	42523636	C	A	CYP2D6	breast	0.605027139	a	s	1	16
22	38379767	38379767	C	G	SOX10	breast	0.789068222	a	s	1	8
2	25457243	25457243	G	A	DNMT3A	AML	0.863038063	a	s	28	411
4	55561907	55561907	C	G	KIT	AML	0.748645484	f	s	3	24
4	55564663	55564663	T	G	KIT	AML	0.883287668	a	s	3	24
4	55564702	55564702	C	T	KIT	AML	0.90420109	a	s	3	24
2	209113113	209113113	G	A	IDH1	AML	0.673070192	a	s	17	68
22	42522550	42522550	G	A	CYP2D6	glioblastoma	0.644029975	a	s	1	16

B

False Positive Overlap with CIViC											
Chr	Start	Stop	Ref	Var	Gene	Disease	Confidence	MR Call	Classifier Call	Evidence Items	CIViC Score
13	28609758	28609758	C	G	FLT3	AML	0.008960945	s	f	4	43
2	209113113	209113113	G	A	IDH1	AML	0.258962572	s	f	17	68
4	55589770	55589771	-	AGGTGGG	KIT	AML	0.386413515	s	f	3	24
9	5073770	5073770	G	T	JAK2	AML	0.270821482	s	a	5	70
19	33792395	33792396	-	CCACGTTGCGCTGCTTGG	CEBPA	AML	0.309725046	s	a	7	104
21	36206887	36206887	T	A	RUNX1	AML	0.137544215	s	a	6	88
17	7577602	7577602	A	G	TP53	AML	0.109305017	s	a	2	28
2	25457243	25457243	G	T	DNMT3A	AML	0.403970242	s	a	28	411
17	29665755	29665756	-	A	NF1	AML	0.000784582	s	f	8	56
9	98220365	98220365	C	A	PTCH1	AML	0.093966253	s	a	1	16
17	7578190	7578190	T	C	TP53	AML	0.253813058	s	a	2	28
4	106156072	106156072	C	T	TET2	AML	0.309087753	s	a	6	96
17	74733070	74733070	G	A	SRSF2	AML	0.139067724	s	a	1	20
12	49433619	49433619	C	T	KMT2D	breast	0.468647331	s	a	1	12
17	29654591	29654591	C	T	NF1	breast	0.148995921	s	a	8	56
17	41223066	41223066	G	A	BRCA1	breast	0.241903991	s	a	11	148
17	41245274	41245274	C	A	BRCA1	breast	0.273376912	s	a	11	148
21	36171722	36171722	G	A	RUNX1	breast	0.27454862	s	a	6	88
5	67522556	67522556	G	A	PIK3R1	breast	0.48558706	s	a	1	4
12	49432165	49432165	C	T	KMT2D	breast	0.412108928	s	a	1	12
21	36164787	36164787	C	T	RUNX1	breast	0.432980925	s	a	6	88
12	49421834	49421834	G	A	KMT2D	breast	0.40430221	s	a	1	12
21	36164642	36164642	G	A	RUNX1	breast	0.478969276	s	a	6	88

21	36252919	36252920	-	TAGCATTTCTCAGCTC	RUNX1	AML	0.304293782	s	f	6	88
17	74732936	74732959	GGCGGCTGTGGTGTGAGTCCGGGG	-	SRSF2	AML	0.13832365	s	f	1	20
3	178927980	178927981	-	GTC	PIK3CA	breast	0.370807111	s	f	19	146
17	37619027	37619027	G	A	CDK12	breast	0.037594032	s	a	1	8
17	7577538	7577538	C	T	TP53	breast	0.485756129	s	a	5	60
19	33792250	33792255	GCAGTT	-	CEBPA	AML	0.082897209	s	f	7	104
21	36164687	36164688	-	TC	RUNX1	AML	0.330570877	s	f	6	88
15	90631838	90631838	C	T	IDH2	AML	0.012970433	s	f	3	28
17	74732936	74732959	GGCGGCTGTGGTGTGAGTCCGGGG	-	SRSF2	AML	0.107938401	s	f	1	20
17	7577581	7577581	A	T	TP53	AML	0.195514232	s	a	2	28
19	33793247	33793257	GCGTGCGGGGG	-	CEBPA	AML	0.107198857	s	f	8	110
12	49422933	49422933	C	T	KMT2D	breast	0.14578785	s	a	1	12
4	153244185	153244185	G	A	FBXW7	breast	0.132516399	s	a	2	10
4	153249384	153249384	C	T	FBXW7	breast	0.186400592	s	a	4	26
1	11184571	11184571	G	T	MTOR	breast	0.13387543	s	a	1	6
1	11184573	11184573	G	A	MTOR	breast	0.118334189	s	a	2	10
12	56478851	56478851	C	T	ERBB3	breast	0.320064127	s	a	1	12
14	105246551	105246551	C	T	AKT1	breast	0.327136993	s	a	5	35
12	49444545	49444545	G	A	KMT2D	breast	0.315532327	s	a	1	12
17	29533260	29533260	C	T	NF1	breast	0.089598835	s	a	8	56
17	37879588	37879588	A	G	ERBB2	breast	0.059859622	s	a	2	25

4	55594177	55594177	C	T	KIT	breast	0.311021984	s	a	3	24
17	41262680	41262680	G	A	BRCA1	breast	0.130686581	s	a	11	148
2	212537889	212537889	C	T	ERBB4	breast	0.022910668	s	a	1	8
5	67569746	67569746	G	A	PIK3R1	breast	0.143235296	s	a	1	4
5	67576437	67576437	C	T	PIK3R1	breast	0.321904689	s	a	1	4
12	49418629	49418629	C	T	KMT2D	breast	0.214990601	s	a	1	12
12	49433883	49433883	G	A	KMT2D	breast	0.061000999	s	a	1	12
6	41903845	41903845	C	G	CCND3	AML	0.224974856	s	a	1	6
13	48941711	48941711	A	T	RB1	SCLC	0.425798684	s	a	3	20