Benchmarking somatic variant callers:

MuTect1 and GATK Mutect2 on TCGA WES

Thesis submitted to

Havard T.H. Chan School of Public Health

In partial fulfilment of the requirement for the degree of

M.S. in Computational Biology and Quantitative Genetics

By

Qing Zhang

Thesis Advisor

Dr. Gad Getz

Content

List of Figures

List of tables

Abstract

Genomic alteration is a major driving force of tumorigenesis, and single nucleotide variations (SNVs) are an important type of somatic event. Sensitive and specific detection of somatic events has been the foundation of many downstream discoveries. Nevertheless, many benchmark studies have found that the call sets produced from different somatic variant detection algorithms show a high level of discrepancy.

In this work, I benchmark the calls produced by two somatic mutation detection algorithms, MuTect (M1) and GATK Mutect2 (M2). I compared the two call sets generated for 18 cohorts from The Cancer Genome Atlas (TCGA) project and ran the driver gene discovery tool, MutSig, to identify driver genes. The results show that, although concordance analysis shows a reasonably good match across the cohorts, the set of driver genes discovered based on the two callers are often very different. I also demonstrated that the MutSig algorithm can help identify recurrent artifactual mutation calls that cause the differences in the lists of driver genes.

Keyword

Somatic variant calling, MuTect, whole exome sequencing, TCGA, benchmark

Acknowledgments

First I would like to thank all members from Getz lab for all the valuable insights! In particular, I want to thank Julian Hess and Chip Stewart for their input to this work - I could never forget Julian's thorough answer to my questions with several references included, and Chip's always welcoming office that witnessed many "small questions" transformed into quite decent discussions. I would also like to thank Aaron Graubert for implementing and supporting tools that enable efficient running on the cloud. These are great tools that can save many hours. Thanks Mendy for amending the text to enhance readability. Thanks Ziao Lin and Jialin Ma for supporting me as friends!

I would also like to say thank you to the GATK team: David Benjamin who has been responsive for my M2 questions, and Soo Hee Lee who has provided very informative tutorials and feedback on the GATK forum.

Finally, I would like to thank Gaddy for his dedication. Gaddy has always been a great inspiration for me, from the sharp questions from lab meetings to time management and leading a team. It has been a great honor to join Getz lab and do exciting research with great minds.

1. Introduction

Somatic mutation calling is one of the cornerstones of cancer genome analysis. Many downstream discoveries, including tumor subtyping, inference of phylogenies, and driver gene discovery are largely based on accurate detection of somatic alterations.

Next-generation sequencing (NGS) is by far the most promising technology for *de novo* mutation detection. Since most of the known driver events are associated with coding effects, whole-exome sequencing (WES) that captures the exonic regions of known protein-coding genes, has become a very cost-effective way to identify those disease-related mutations (Ng et al. 2009).

There have been many algorithms for *de novo* detection of somatic single-nucleotide variations (SNVs) from tumor-normal sequencing, and many benchmark studies were carried out to compare the performance of such mutation detection methods. However, most benchmark studies that investigate the concordance of call sets, either from simulated data or real data, do not look further into 1) how the mutations affect downstream analyses and 2) how the discrepancies associate with specific differences among the algorithms.

In this work, I attempt to benchmark two methods for single nucleotide variation (SNV) detection: MuTect1 (M1) (Cibulskis et al. 2013) and GATK Mutect2 (M2) (Benjamin et al. 2019). I want to understand the overlap and differences between the methods and investigate how the discordant calls result in differences in driver gene discovery by MutSig. Based on the

different set of significant genes identified by MutSig after either M1 or M2 calling, I hope to identify recurrent artifactual patterns and relate them back to differences in the algorithms.

2. Background

2.1 Challenges in somatic variant detection

One would expect to find all (clonal) somatic variations if given sufficient sequencing depth; however, in real-world scenarios, this is not a trivial task because sequencing depth is usually limited. Moreover, detected variants often include false positive events caused by artifacts and unaccounted-for sequencing errors. A good somatic mutation caller needs to be sensitive to call variants at different variant allele fractions (VAFs), as many true somatic variants' VAF can vary based on sampling admixture of tumor and normal cells (purity), copy number alterations (aneuploidy), or subclonality in the cancer. Apart from sequencing error, false variants can also originate from contamination of DNA from samples of other individuals (Cibulskis et al. 2011). Other artifactual mutations can be caused by DNA damage produced by mutagens during sample preparation (e.g. tissue fixation process in formalin-fixed tissues (Munchel et al. 2015), or oxidative damage during shearing (Costello et al. 2013)) or from alignment artifacts (Narzisi et al. 2018; Li 2014). These artifacts are largely non-random and contribute to a large portion of false positives. There are different algorithms to detect somatic SNVs from tumor-normal pairs, but the sets of calls they produce can be discrepant among them; hence, it is important to properly benchmark their performance.

2.2 Previous work on benchmarking somatic SNV callers

There are three methods to generate synthetic data, with ground-truth variants, for benchmarking studies: (i) synthetic reads, (ii) reference standards, and (iii) real tumor data (C. Xu 2018).

(1) “Synthetic reads” are used in read simulators like SeqMaker (C. Xu 2018; Shifu Chen et al. 2016) and can be used to generate reads with pre-configured error-models; Bamsurgeon produces a hybrid dataset featuring real reads and simulated variants at varying VAFs, and was used in several DREAM challenges (Ewing et al. 2015). However, the noise in real data is usually more complex than simulated data, which makes these methods sub-optimal.

(2) “Reference standards” aims to provide exceptional quality of data wherein the ground truth is known. The Genome in a Bottle Consortium (GIAB) published the NA12878 cell line (Zook et al. 2016) whose sequence has been verified from multiple techniques and callers. This cell line was subsequently used in multiple studies for performance assessment (Cibulskis et al. 2013; Li et al. 2018; Li 2014; H. Xu et al. 2014). There are new methods that incorporate long-read sequencing as ground-truth (Li et al. 2018). Nevertheless, reference standard data is still limited given its cost, and it may not cover many of the artifacts that arise from analyzing real data, as observed across cancer and sequencing platforms.

(3) “Real data” is used as a last resort for benchmarking studies. Usually such methods will use large datasets and orthogonal technologies for cross-validation, such as whole-genome sequencing or RNA-Seq (Ellrott et al. 2018). There are also several drawbacks to this

benchmarking approach: (1) using real data is computationally expensive; (2) in tools that require manual fine-tuning of their parameters for optimal performance, real data is not very amenable to fine-tuning. Despite these drawbacks, however, using real data that helps to capture the wider spectrum tumor heterogeneity will best characterize the performance of the caller.

2.3 Somatic calling algorithm: M1/M2 logic

Somatic mutation calling is usually composed of three steps: (i) a prefiltering step that is used to filter out low quality reads, keeping "good" reads that satisfy some heuristic criteria; (ii) a somatic detection statistical modeling step that aims to calculate the likelihood ratio (or log odds) of existence vs. non-existence of an non-reference allele (i.e., an allele with a somatic mutation); and finally (iii) a post-filtering step that is based on our prior knowledge of different artifact modes.

Prefiltering

In the prefiltering step, high quality reads are retained. The high quality reads are usually defined based on heuristic cutoffs of 1) mapping quality; 2) base quality; and 3) the quality of bases at the ends of each read (also known as clipping); moreover, a high quality read usually incorporates PCR deduplication, which removes reads originating from one single DNA fragment. A detailed comparison of M1/M2 for tumor sample prefiltering can be found in **Table 1.** In M1, more stringent pre-filters are applied to tumor samples compared to normal, while M2 uses identical criteria for both samples.

Category		M1	M2
Mapping quality	Minimum mapping quality	> 0	> 0 and remove secondary alignments
	Mate rescue	Disabled	Disabled. Remove read pairs that aligns to different contigs
	Clipping	Remove reads if >30% bases that are soft-clipped	Soft-clipped bases are included, but can be reverted by "--dont-use-soft-clipped-bases true" [Hard-clip] Remove bases from end of reads until base quality > 10
Base quality	minimum base quality	> 5	> 10
Overlapping read pair		If bases agree in overlapping reads keep the one with better quality, otherwise both reads are discarded	Will adjust model downstream
Remove reads from PCR duplication		Yes	Yes
Other filters		Sum of quality scores of mismatches <= 100	Various formatting / consistency check

Table 1: comparison of prefilters for tumor sample used in M1 and M2

Reassembly and realignment in M2

A read mapper like BWA will choose the optimal alignment for reads as if they are independent, which may not be the optimal multi-read alignment for reads on the same region in the

reference (Li 2014; Chowdhury and Garai 2017). Failure to construct the optimal realignment may result in multiple adjacent mismatches. An alternative approach practiced by several recent callers (Li 2014; Garrison and Marth 2012; Narzisi et al. 2018; Benjamin et al. 2019) is to first perform local reassembly and then realignment. Given this step is computationally expensive, M2 identifies "active regions" and tries to identify sets of haplotypes using a De Bruijn graph. After pruning the paths with less than 3 supporting reads, each read is aligned to the set of potential haplotypes using a Pair HMM algorithm, which results in a read $\times$ haplotype matrix, and then is marginalized to a read $\times$ allele matrix that serves as the input to the somatic detection statistical model.

Somatic detection statistical model

The next step of somatic calling is applying a somatic detection statistical model that aims to differentiate sequencing errors from real non-reference bases. Both M1 and M2 calculate the likelihood ratio between two models; one in which a mutant allele exists with a variant allele fraction of f, and the second model in which there is no non-variant allele, i.e., all non-reference bases represent noise, which is equivalent to using the first model with $f = 0$. Both callers first calculate the likelihood of observing a base assuming the following:

1. All substitutions are equally likely. In M1, this means that the probability that a read comes from the allele α that is not called is $e_i/3$ where e_i is the probability that the sequenced base is incorrect (each base *i* is associated with a Phred-like score $q_i = -10\log_{10}(\text{prob. of it being wrong}))$. In contrast, M2 uses a random variable f_a to

denote whether a read comes from the physical allele α. M2 assumes a flat Dirichlet prior for f_a.

2. The variant allele fraction (VAF) is independent from the exact mutant base. In M1, this independence is translated to $P(m, f) = P(m)P(f)$, and $P(f) = 1$ is assumed (i.e., flat prior on *f*). In M2, this assumption is used to justify the use of variational Bayes and the mean field approximation.

3. Both callers assume dependence between mates within a read pair, but independence between read pairs.

Both models first calculate the likelihood for observing a base given a physical allele α from a read / read pair, then aggregate such likelihood from all reads/read pairs to get the evidence.

In M1, the model is for a single base and uses the sequencing error to calculate the probability of observing a particular base given the allele. For a read *i*, the probability to observe the base b_i is different for the following three cases:

$$f(b_i|e_i, r, m, f) = \begin{cases} fe_i/3 + (1-f)(1-e_i), & \text{if } b_i = r \\ f(1-e_i) + (1-f)e_i/3, & \text{if } b_i = m \\ e_i/3, & \text{otherwise} \end{cases}$$

Where b_i is the observed base at position i, e_i is the probability of error of the sequenced base, r is the reference base, m is the alternative mutant base whose true underlying variant allele fraction is f.

Since M1 assumes reads are independent, the likelihood of having a mutant base m with allele fraction f can be calculated by multiplying the read-level likelihoods together.

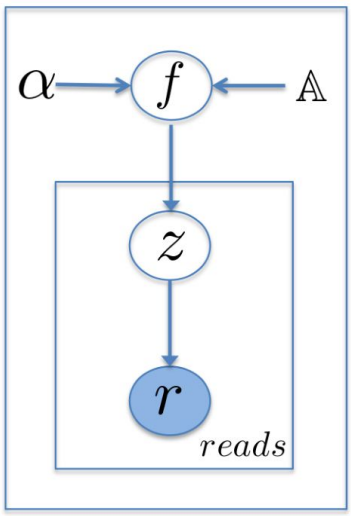
$$L(M_f^m) = P(\{b_i\} \mid \{e_i\}, r, m, f) = \Pi_{i=1}^{d} P(b_i | e_i, r, m, f)$$

The log likelihood ratio between the model with a mutant allele with a variant allele fraction *f* and a model with no mutant allele is formulated as

$$LOD_T(m, f) = \log_{10}\left(\frac{L(M_f^m)P(m,f)}{L(M_0)(1-P(m,f))}\right) \geq \log_{10}\sigma_T$$

and if it is greater than a threshold, the mutation passes to the post-filtering step.

M2 introduced two latent random variables: f, the probability that a read comes from allele α; and z_{ra} the one-hot indicator variable that represents whether fragment *r* comes from allele α. The likelihood model has become multi-allelic – $P(\mathbb{R}|\mathbb{A})$, where A is defined to be all possible sets of physical alleles that contain allele α, and R is the fragment subsets.



$$f \sim \text{Dirichlet}(\alpha)$$
$$z|f \sim \text{Categorical}(f)$$
$$p(r|z_{ra}) = l_{ra}$$
$$\ell_{ra} \equiv P(\text{read } r|\text{allele } a)$$

from PairHMM

$$\log \frac{p(\mathbb{R}|\mathbb{A}_{alt})}{p(\mathbb{R}|\mathbb{A}_{ref})} > \delta = 3.0$$

then emit variant

Figure 2-1: the latent likelihood model used in M2 from GATK training workshop("gatk2019 - gatk2019 - Training - CSC Company Site" n.d.)

Therefore, the full model likelihood in M2 has become

$$P(\mathbb{R}, z, f|\mathbb{A}) = P(f)P(z|f)P(\mathbb{R}|z, \mathbb{A}) = \mathrm{Dir}(f|\alpha)\prod_a \prod_r (f_a \ell_{ra})^{z_{ra}}$$

Variational Bayes was used to get $P(\mathbb{R}|\mathbb{A})$ by marginalizing the two latent variables, using a uniform prior on f (i.e. *P*(*f*)=1). The tumor log odds (LOD) emitted by M2 for an alt allele is the log evidence ratio of an allele set containing all alleles versus an allele set excluding that specific alt allele.

Post-filtering

Post-filtering aims to remove non-random artificial variants emitted from somatic detection models. There are two types of filters: hard filters and probabilistic filters. Hard filters remove sites when certain conditions are met, and probabilistic filters aim to rank variants by their probability of being an error, $P(\mathrm{error})$, which can be used later to determine the cutoff that yields the best performing F-score.

Most of the postfilters in M1 are hard filters. Although filtering is based on some numerical metric, e.g., odds ratio of strand bias, M1 does not attempt to rank variants based on such values and uses predefined cutoffs for them instead. M2 classifies false positives into three categories: non-somatic, artifact error, and sequencing error. The within-category errors are

assumed to be correlated, while the cross-category errors are modelled independently. The final probability of error of a variant is assumed to be:

$$P(\text{error}) = 1 - (1 - P(\text{max artifact prob}))(1 - P(\text{max non-somatic})(1 - P(\text{sequencing error})))$$

Many of the probabilistic filters in M2 are based on an allele fraction clustering model, which assumes that variants can be clustered based on allele fractions to characterize each subclone. It uses a modified version of a Chinese Restaurant Process (Aldous 1985) to assign variants into different clusters with different probabilities. While the somatic detection model emits the tumor LOD assuming a flat prior on *f*, the tumor LOD can be modified to get the likelihood ratio assuming a non-flat prior. In the post-filtering step, M2 clusters the allele fractions and iteratively updates each variant's allele fraction and calculates an allele fraction–adjusted likelihood ratio and the membership of each variant in each cluster until convergence.

Artifact modes	M1	M2
Neighboring indels	"proximal-gap": reject a variant if there are $\geq$ 3 reads with insertions in an 11 base-pair window	**No**
Poor mapping: false positives caused by sequence similarity in the genome	i) "poor-mapping" rejects a variant if >50% reads have map quality < 0 ii) Removes reads with confident mismatches (sum of base quality of mismatches > 100) in prefiltering	i) "clustered-events" filter false positives on realigned regions with more than 3 variants ii) "mapping-quality": filter variants on reads with median mapping quality less than 30
Misalignments	"clustered-position" filter rejects variants clustered at a consistent distance from the start or end of read alignment	"read-position" filters by median median absolute deviation of the position of the variant from end of the supporting reads > 1, **without checking if it shows any "clustering".**

Fragment length	**No**	Filter variants if the difference of alt and ref reads' median fragment length > 2
Triallelic site	Filter variants if the normal sample is heterozygous with alleles A and B, while MuTect is considering an alternative allele C.	Filter variants with two alt bases (note that the somatic calling model is potentially capable of multiallelic modeling, but the filtering step regards those variants as false positives)
Strand bias	"strand-bias" filter stratifies the somatic detection statistic (tumor LOD) by strand, and rejects if strand-specific LOD is < 2.0 in a direction where the sensitivity to have passed that threshold is >90%	The probability of an artifact is learned from a beta-binomial likelihood, **adjusting for the limited number of reads at the tail of an exon.**
Orientation bias	Handled by a separate OxoG filter	The artifact probability is learned from a beta-binomial model
Base quality	Will only call variant with quality > 5 (used in prefilter)	Will only call variant with quality > 10 (used in prefilter) **AND median base quality of alt reads > 20 as defined in "base_quality" filter**
Panel-of-normal	There are two PoNs in M1. i) A blacklist PoN removes calls with two occurrences from tumor-only calling of normal samples. ii) A token PoN that quantifies the noise in normal samples and filters variants that are likely noise.	There is only one PoN used in M2, but this PoN operates in two ways: i) All sites at the PoN VCF are skipped in the initial triage step before proceeding to the assembly and realignment steps (to increase speed), therefore only the sites in the reassembled region are kept. ii) In the post-filtering step, these remaining variants are removed by the "panel-of-normal" filter.
Normal artifacts: a matched normal can be used for distinguishing **technical artifacts for somatic mutations** (e.g., mapping error due	"observed-in-control" filter: A candidate is rejected if, in the control data, there are (i) ≥ 2 observations of the alternate allele or they represent ≥ 3% of the reads; and (ii) their sum of quality scores is > 20.	Identical somatic detection model is applied to the normal to get a normal LOD ("normal artifact log odds"). This represents the likelihood of the reads given that an artifact appears in the normal. It is then multiplied by a prior probability for having an artifact in

to rare SNP in centromere)		the normal artifacts which, after normalization, yields the posterior error probability
Germline filter: a matched normal can be used to detect **germline variant**	Variant classification model calculates a normal LOD that represents the likelihood ratio between the reference model (alt AF = 0) and a variant being germline heterozygous (alt AF = 0.5). The prior probability, P(germline), is determined by whether the site is included in dbSNP.	M2 tries to determine three possibilities: variant exists in tumor and normal is germline het; variant exists in normal, and normal is germline hom alt; variant exists in tumor but not in normal. The normalized sum of the first two possibilities reflects the germline error probability used in the "germline filter". The populational allele fraction is known for sites in gnomAD, or approximated with a predefined beta distribution (Beta(0.01,10)).
Contamination filter	The contamination level (*f-cont*) that is estimated by ContEst(Cibulskis et al. 2011) replaces the reference model M_0 with a variant model $M_{f\text{-}cont}$	calculates the likelihood of *a* alternative reads out of *d* total reads given the contamination estimate. The likelihood is then multiplied by the prior from the AF clustering model to get the posterior probability of error due to contamination.
Bad haplotype	**No**	Since M2 can phase, "bad haplotype" probability is equal to the greatest probability of a technical artifact of any in-phase variant call within a certain distance (default distance is 100 bases).

Table 2: Comparison of post-filtering mechanisms in M1/M2. Yellow blocks are hard filters that directly remove a variant (equivalently set P(error)=1), and blue blocks are probabilistic filters that give a P(error) between 0-1 for ranking.

2.4 Driver gene discovery using MutSig

Figure 2-2 MutSig p values and q-value

Large-scale sequencing of cancers seeks to identify mutated genes that confer a selective advantage to cancer cells (Tokheim et al. 2016). MutSig2CV (Lawrence et al. 2013) is one of the algorithms developed to identify signals of positive selection and distinguish them from passenger events given inferred background mutational processes. MutSig2CV uses the following three statistical tests:

i) Abundance (CV): This part classifies whether a gene is highly mutated relative to a background mutation rate that varies across genes, patients, and sequence context. Several gene-level covariates are used to model the gene-level background mutation rate.

ii) Clustering (CL): Mutation hotspots are often present in true cancer genes. This part of the model bins mutations at the site-level (i.e., base-base). Greater p-values (reflecting lower significance) are assigned to genes with variants distributed uniformly throughout the gene vs. cases in which there are consistent hotspots.

iii) Conservation (FN): Genetic sites that are conserved over evolution will be more functionally important, and thus the genes that are enriched with variants on those sites are assigned with higher significance.

MutSig will output three p-values for the three aspects – pCV, pCL and pFN – and aggregate the three p-values (via Fisher combination) to a single p- and q- value. The Benjamini-Hochberg (Benjamini and Hochberg 1995) q- value is used as a cutoff for determining the list of significant candidate driver genes, representing a predetermined acceptable level of expected false discovery rate (typically, a q-value cutoff of 0.1 is used). In this work, I run MutSig2CV to test the impact of the different mutation calls of M1 and M2 on the list of candidate driver genes. Since MutSig2CV (and in particular MutSigCL) is sensitive to recurrent events (which may be artifacts), I will test the effect on pCL.

2.5 Mutational signatures

Characterizing underlying mutational processes with mutation signature profiles helps to understand cancer initiation and progression. The non-negative matrix factorization (NMF) algorithm has been widely used in deciphering mutational signatures in cancer somatic mutations, stratified by 96 base substitutions in tri-nucleotide sequence contexts (Lawrence et al. 2013; Alexandrov et al. 2020). This classification is based on the six substitution subtypes: C>A, C>G, C>T, T>A, T>C, and T>G (all substitutions are referred to by the pyrimidine of the mutated Watson–Crick base pair). Furthermore, each of the substitutions is examined by incorporating information about the bases immediately 5’ and 3’ to each mutated base, generating 96 possible mutation types (6 types of substitution x 4 types of 5’ base x 4 types of

3’ base). In this study, I compared the 96 base substitution counts in the form of the lego plot (an example can be found at Figure 2-3) to assess the presence of recurrent artifactual mutations.

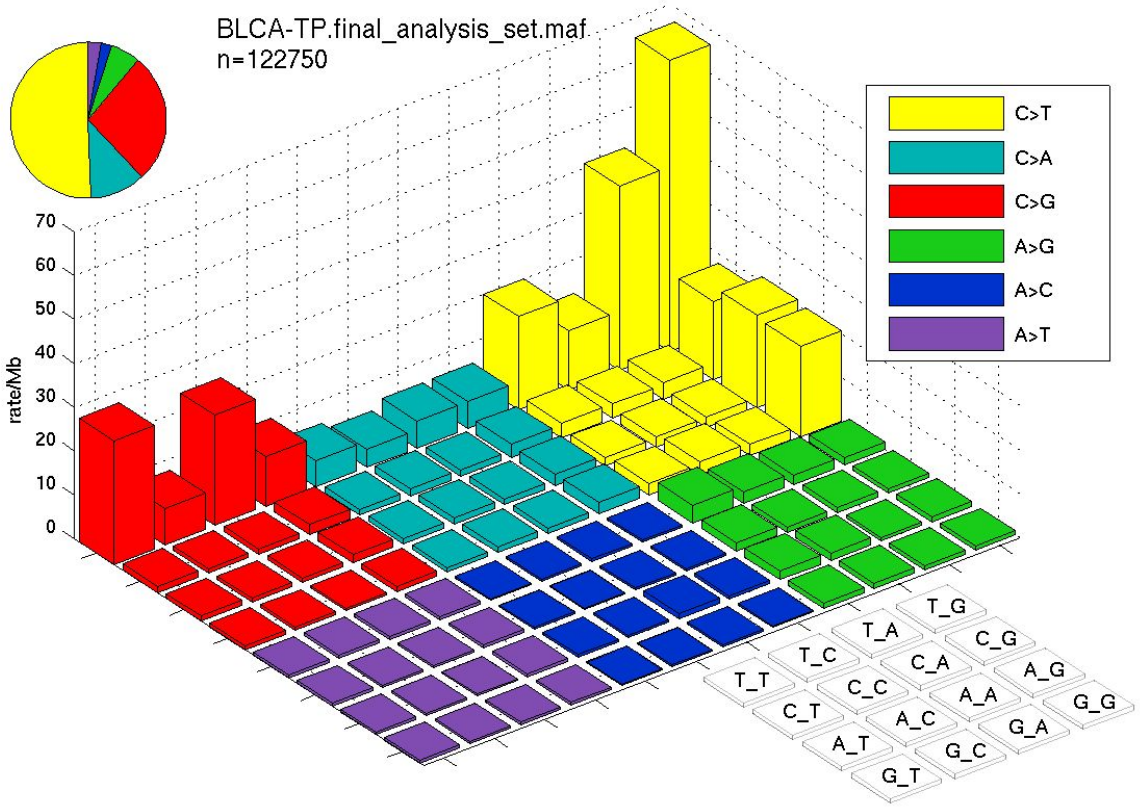


Figure 2-3: An example of a lego plot, generated in Firehose, showing the mutation rate in different contexts (“Mutation Analysis (MutSig 2CV v3.1) - Bladder Urothelial Carcinoma (Primary Solid Tumor)” n.d.)

3. Methods

I used 18 TCGA cohorts in this study (Supplementary Table 2). I ran M1/M2 independently and filtered the calls to a shared set of samples and shared genomic intervals in order to remove differences introduced in the process. Next, I looked at the concordance between the call sets from two callers and ran MutSig on each cohort to identify potential driver genes. As described above, MutSig is not only one of the important downstream discovery tools but also informative for identifying recurrent artifacts generated by the callers. The recurrent calls on genes that are only significant by one caller were manually reviewed in IGV to check for systemic artifacts.

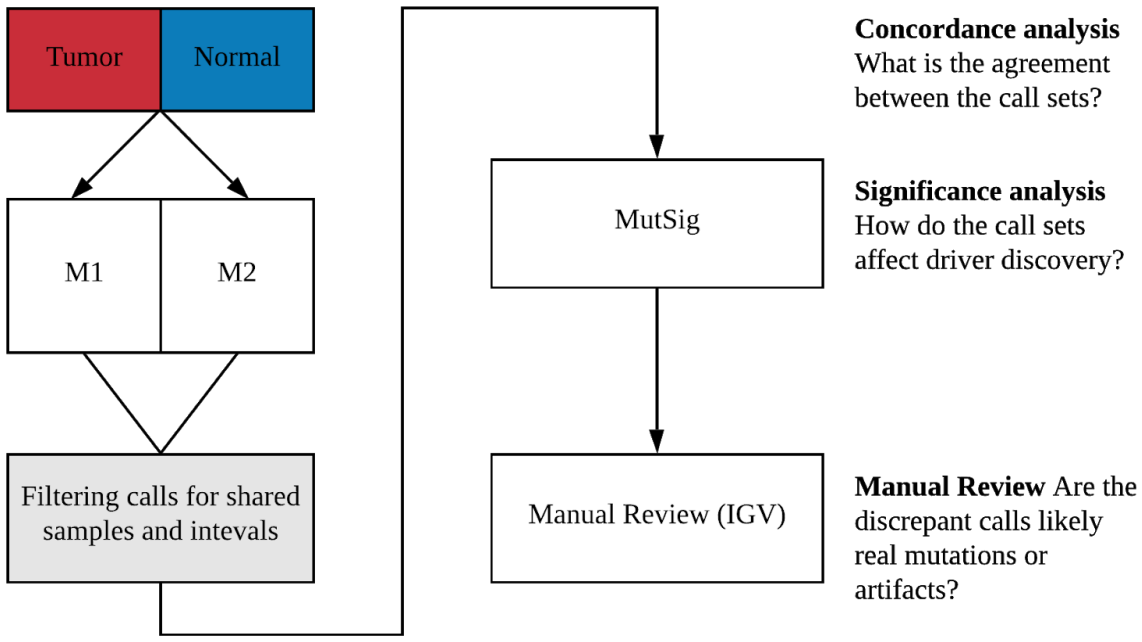


Figure 3: workflow

3.1 Somatic calling pipeline

M1 workflow: M1 call set produced for MC3 project (Ellrott et al. 2018) was used. The call set was filtered by the OxoG filter and then by the token PoN.

M2 workflow: The M2 calling pipeline is adapted from GATK best practice somatic & indel calling pipeline. The original workflow is in WDL, which is developed at the Broad Institute as a workflow definition language that is usually executed in Terra / Firecloud by the internal Cromwell services. As an ongoing effort to resolve Firecloud dependency in the lab, this work was done with Snakemake, and was used in some alpha tests with canine and wolF. The final version of the pipeline can be found at https://github.com/getzlab/smk-m2. It is based on GATK 4.1.4.0 (gatk-workflows) with some minor adaptations:

1) "–independent-mate" flag was used in M2 to escape failures with malformed .bam files. This affects the somatic variant calling step where M2 calculates P(fragment| allele) for overlapping read pairs.

2) The FilterAlignmentArtifact task was not included because it was marked as experimental when the pipeline was drafted, and the reference files were not available at the time. When used, this step will increase the specificity of the M2 calls at the reassembled regions.

3) The reference files were obtained from the GATK best practice Google bucket. The major difference comes from the interval list and PoN. GATK's interval list is a bait list, which is more specific to the exome region but might be overly tailored to the Broad's sequencing protocol. The M1 run used a more generic target list. The comparison of the two interval lists can be

found in Supplementary Figure 1. The PoN provided from GATK is an aggregation of many sequencing technologies instead of being only WES-specific, which might decrease the specificity of the PoN in filtering artifacts observed in normal samples.

3.2 Concordance analysis

In this step, I looked at the Venn diagrams of the call sets produced by the two callers and investigated the difference in variant-level features (e.g., their sequence context, the mutation effect, and the VAF)

3.3 Significance analysis

MutSig2CV was used to characterize significant genes with a significant rate and pattern of mutations. In this step, I looked into the genes that are only determined to be significant from one call set ("M1-only genes" and "M2-only genes") and checked which of the three p-values contributed to the discrepancies.

3.4 Manual review

To better understand the major artifact modes, I examined the raw data using IGV (Robinson et al. 2011). A list of known cancer genes (KCGs) was aggregated from both (i) the Cancer Gene Census (Cosmic 2020) and (ii) additional genes with clinical evidence, which should provide a reasonable prior for representing "true driver genes." This list was then used to stratify differentially significant genes, with the expectation that a missing KCG is likely a false negative,

and a novel non-KCG is likely a false positive. Recurrent variants (called in multiple patients at the same locus) in each strata are reviewed.

4. Results

4.1 Concordance analysis

First, I investigated how much the two call sets overlapped (Figure 4-1). Bladder urothelial carcinoma (BLCA), uterine corpus endometrial carcinoma (UCEC), and skin cutaneous melanoma (SKCM) shared the highest fraction of overlapping calls (~87%), while only ~45% calls overlapped in sarcoma (SARC) and thymoma (THYM). At least 42% of the M1-only calls were removed by at least one of the post-filters. Since the GATK PoN was used in the M2 calling, I also checked how much the PoN contributed to removing variants that were only called in M1. However, less than 2% of the M1-only calls were associated with PoN sites among all cohorts, meaning that the blacklist PoN used in M2 cannot explain the majority of the M1-only calls.

Next, I looked at the post-filters responsible for the M1-only calls. Figure 4-2 shows the distribution of filters for each cohort — here, I noticed that SKCM and UCEC share similar patterns with a predominance of the “clustered_events” and “map_quality” filters. For most of the remaining cohorts, “orientation_bias”, “weak_evidence”, and “strand_bias” are common filters, but these are less cohort-specific.

I also checked the number of calls per patient (Figure 4-3) and per gene (Figure 4-4). The numbers are largely consistent, with the exception of a few cases. As for counts per patient (Figure 4-3), there are several outlier patients in SARC and liver hepatocellular carcinoma (LIHC) who have many more M2 calls than M1 calls, on the order of 5-20x. To understand whether this higher number of calls is due to certain artifacts, I looked at the Lego plot of the patients (Figure 4-3-1). In the SARC tumor type specifically, I checked the four patients (TCGA-X6-A7WB-01A-11D-A351-09, TCGA-DX-A6YS-01A-12D-A351-09, TCGA-X6-A7WA-01A-12D-A351-09, TCGA-DX-A6YV-01A-12D-A351-09) with less than 100 M1 calls but have more than 2000 M2 calls. The Lego plot of SARC outliers shows that these calls are enriched in G(C>A)* mutations, and they are usually supported with less than 5 alt reads in the tumor sample. In LIHC, I checked the four patients (TCGA-FV-A23B-01A-11D-A16V-10, TCGA-G3-A25S-01A-11D-A16V-10, TCGA-G3-A25U-01A-11D-A16V-10, TCGA-G3-A25W-01A-11D-A16V-10) with less than 100 M1 calls but > 500 M2 calls. The calls are enriched in T/C(C>A)G mutations. However, there are other differences observed in the Lego plot that cannot be explained by these LIHC outliers, suggesting other types of undiscovered artifacts.

As for counts per gene (Figure 4-4), 27 variants in *PIM1*, a known driver gene in DLBC (Brault et al. 2012; Kuo et al. 2016), were removed by the M2 “clustered_events” filter in the lymphoid neoplasm diffuse large B-cell lymphoma (DLBC) cohort. In the THYM cohort, 72/80 calls by M1 at chr7:74146970 in the gene *GTF2I* were filtered out by the M2 poor-mapping quality filter. However, this variant has been verified with orthogonal technologies (Radovich et al. 2018;

Petrini et al. 2014) and has been identified as the most significant driver gene in THYM (Radovich et al. 2018).

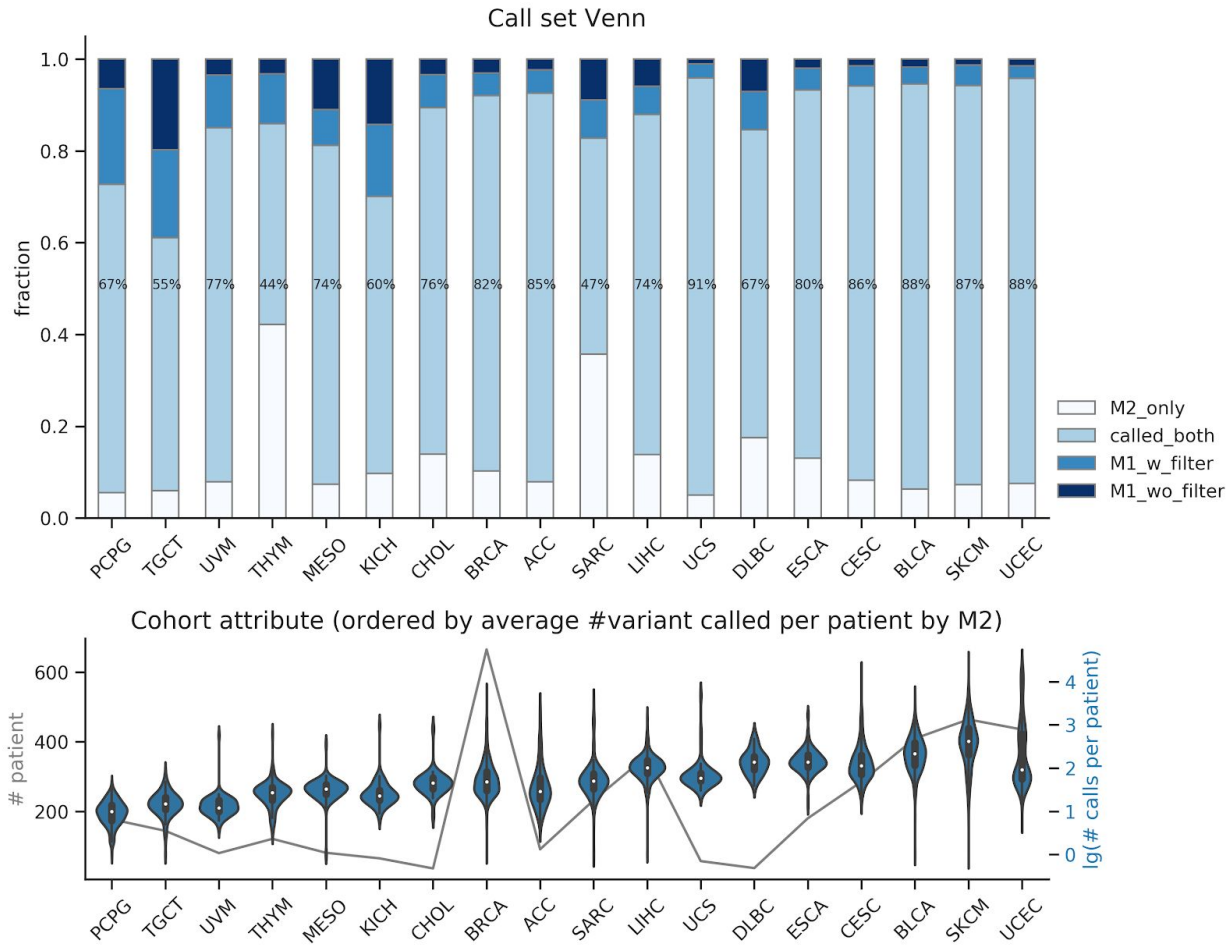


Figure 4-1: Concordance analysis of the two call sets (M1 and M2). The upper plot shows the compact Venn diagram where "M2-only" refers to the variants called in M2 but not in M1; "called both" refers to the variants called by both callers. "M1_w_filter" are the variants that only appear in M1 call sets with known post-filtering reasons for removal from M2, which means these variants have passed the M2 somatic detection step. "M1_wo_filter" are the M1-only calls that did not pass the M2 somatic detection model. The x-axes are cohorts ranked by the average number of SNVs detected per sample. The lower plot shows the number of patients and the distribution of the mutation burden per patient called by M2 (in $\log_{10}$ scale).

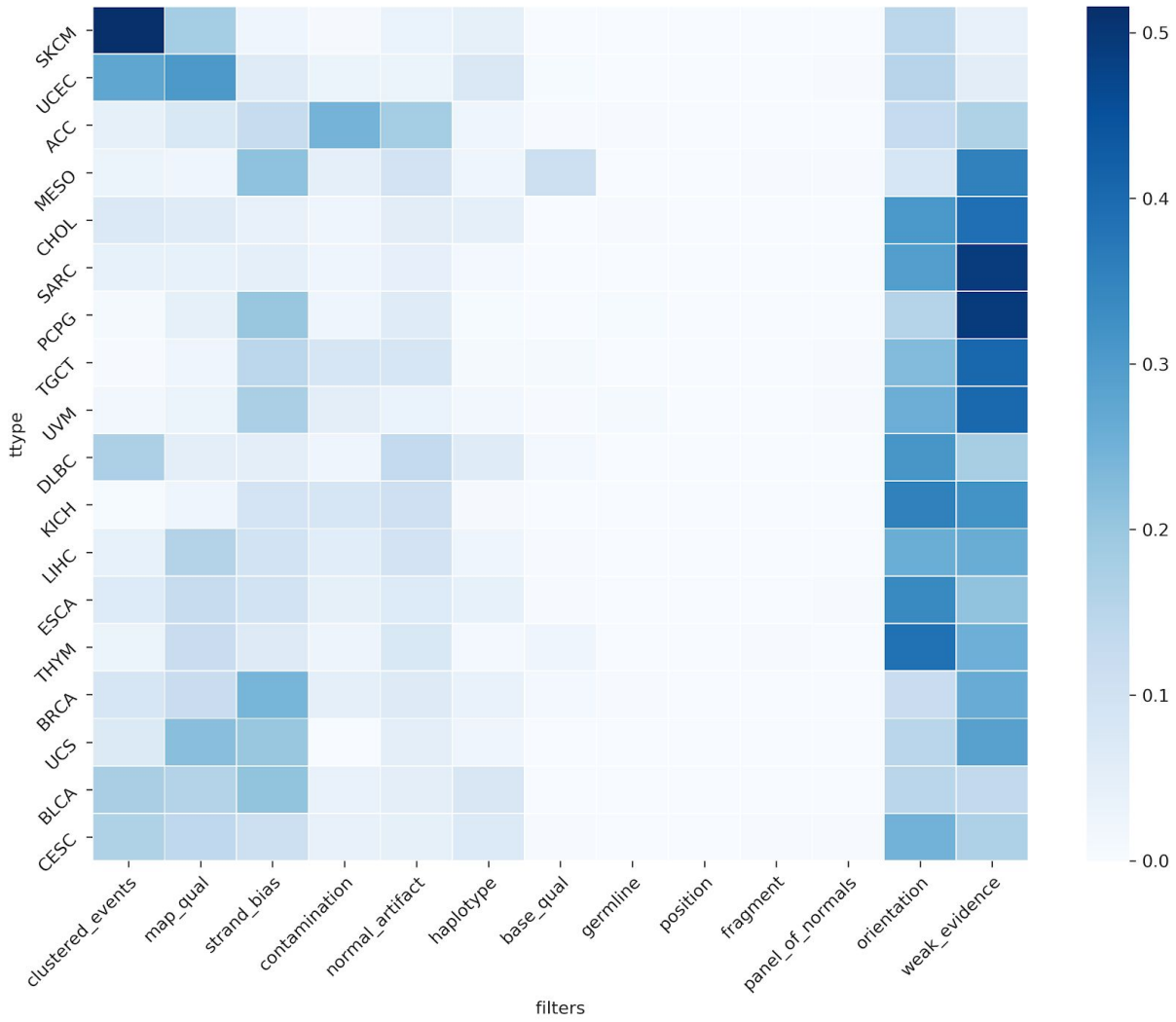


Figure 4-2: Filtering reasons for M1-only calls. Numbers in the bi-clustered matrix are the occurrences normalized by cohort.

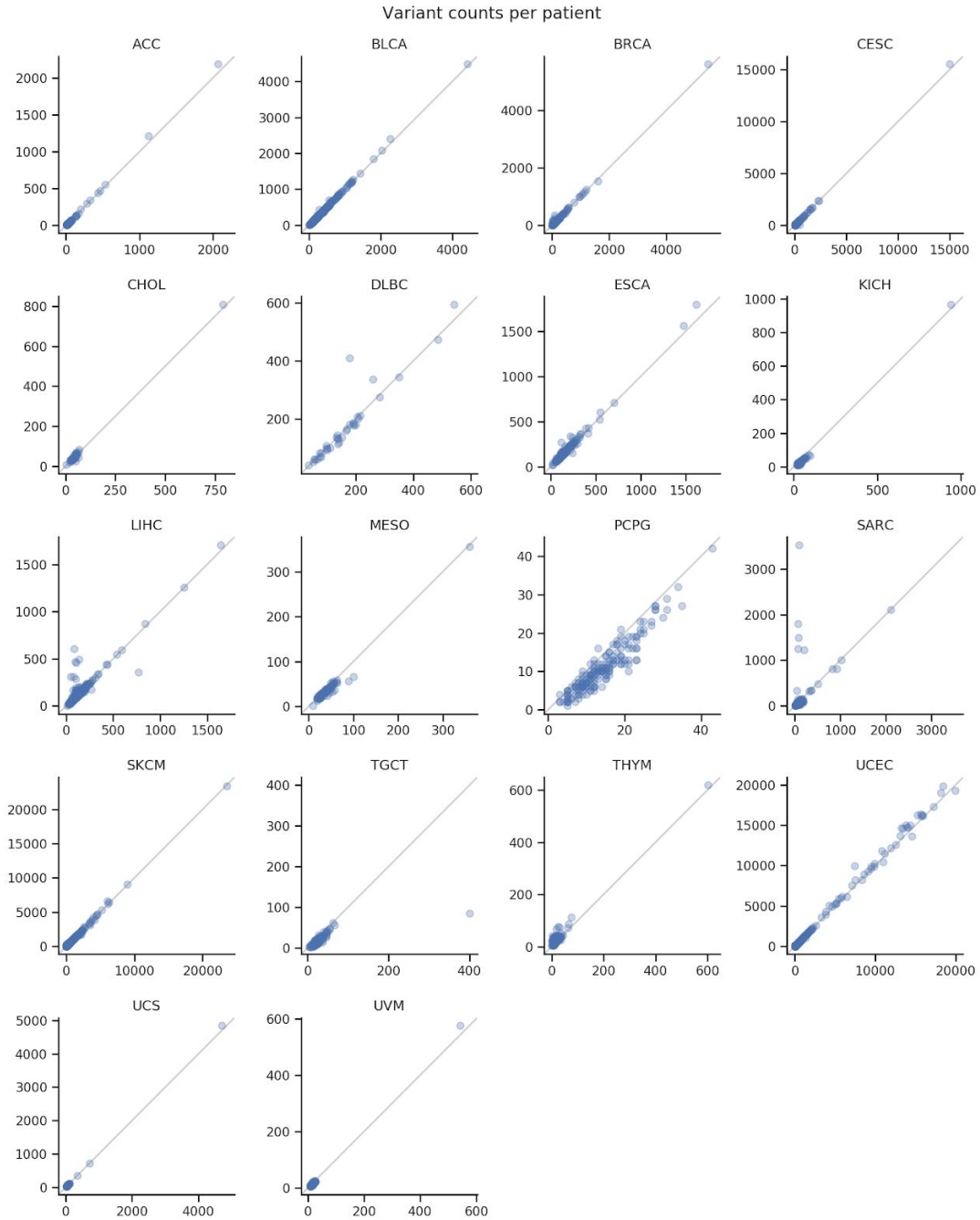


Figure 4-3: Number of calls per patient. Overall, M1 and M2 largely agree. However, there are some outliers in SARC and LIHC, where M2 calls are 5-20x higher than M1 calls.

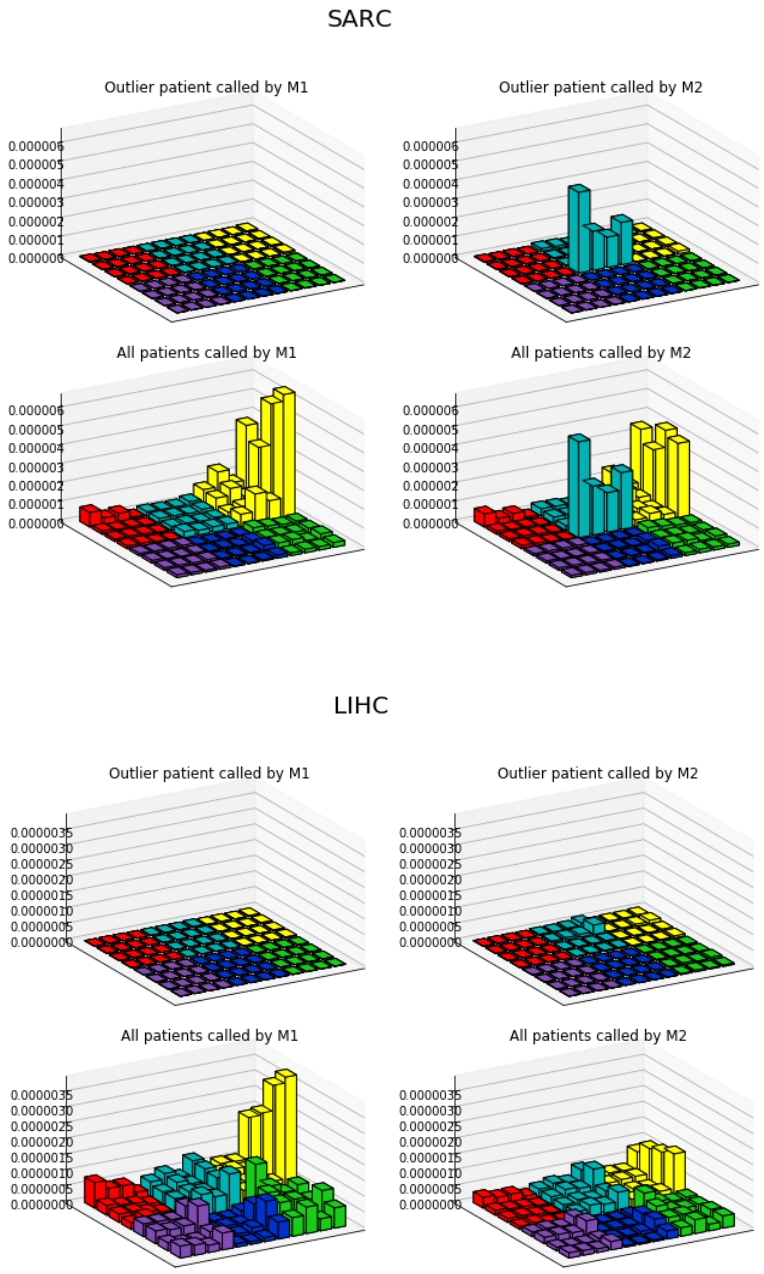


Figure 4-3-1: SARC and LIHC mutation context for outlier patients. For each subplot, the upper-left is the Lego plot for outlier patients called by M1, the lower-left is the Lego plot for all patients called by M2. Similar plots for M2 are on the right. The y-axis of each figure is the number of mutations normalized by the genomic territory for each category.

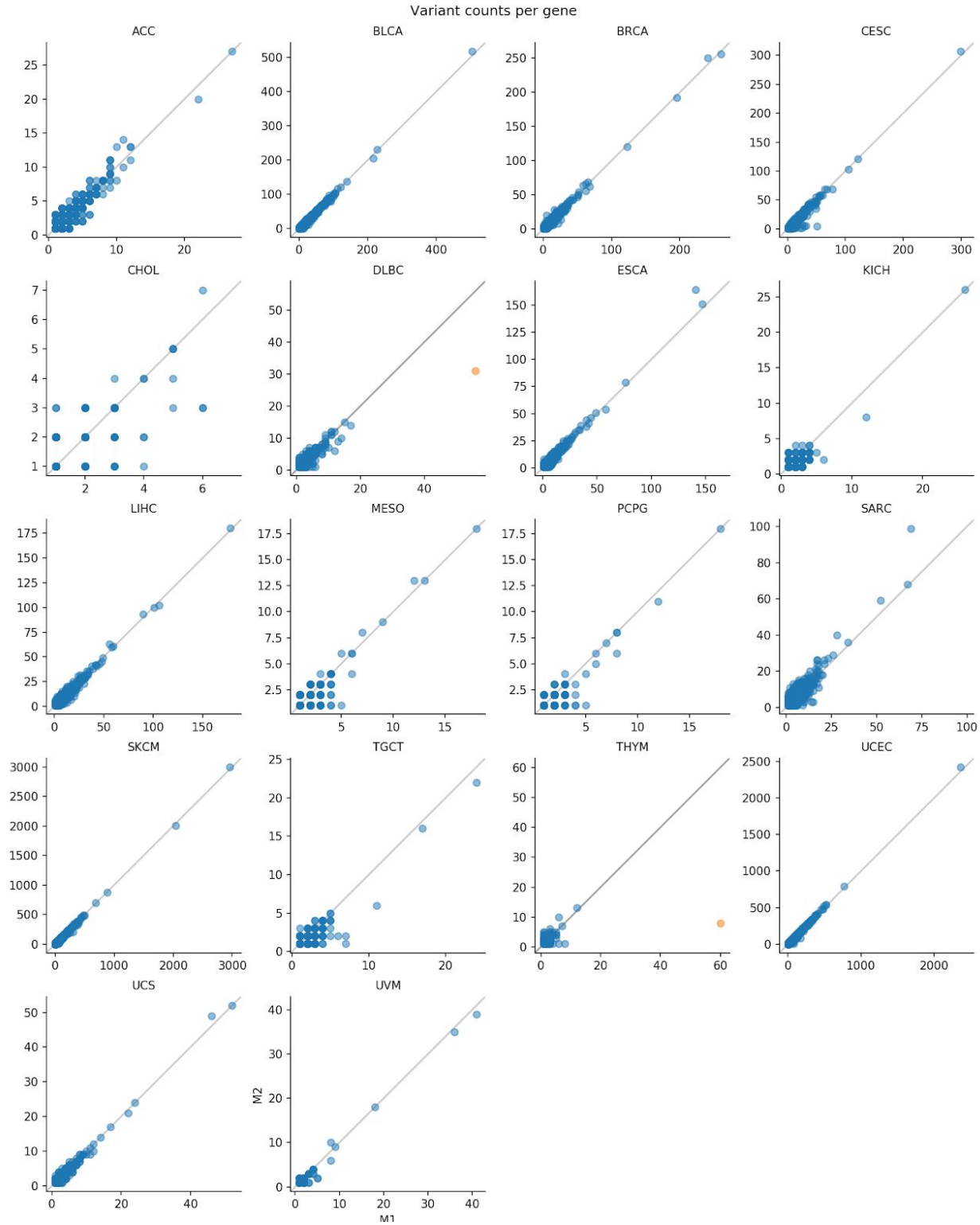


Figure 4-4: Number of mutations per gene by M1 and M2. Overall, M1 and M2 agree with each other. However, there are some outlier genes colored in red -- *PIM1* in DLBC and *GTF2I* in THYM.

M2-only calls, compared to the calls identified by both, are more enriched with mutations with low alternate allele fraction and low number of alternate supporting reads (Figure 4-5). On average, a 3% increase in concordance for the majority of cohorts can be expected, keeping variants with ≧4 supporting reads (Figure 4-6).

M1 avoids calling variants near potential insertion/deletion sites (indels) due to the "proximal gap" filter, so I also checked how this would affect the discrepancies between the two call sets. Figure 4-7 shows that M1 can hardly call at positions that have an indel closer than 10 bases away from the candidate mutation. However, being in the neighborhood of an indel does not explain much of the M1/M2 discrepancy since only ~2% of the M2 calls have a neighboring indel within 20 bp.

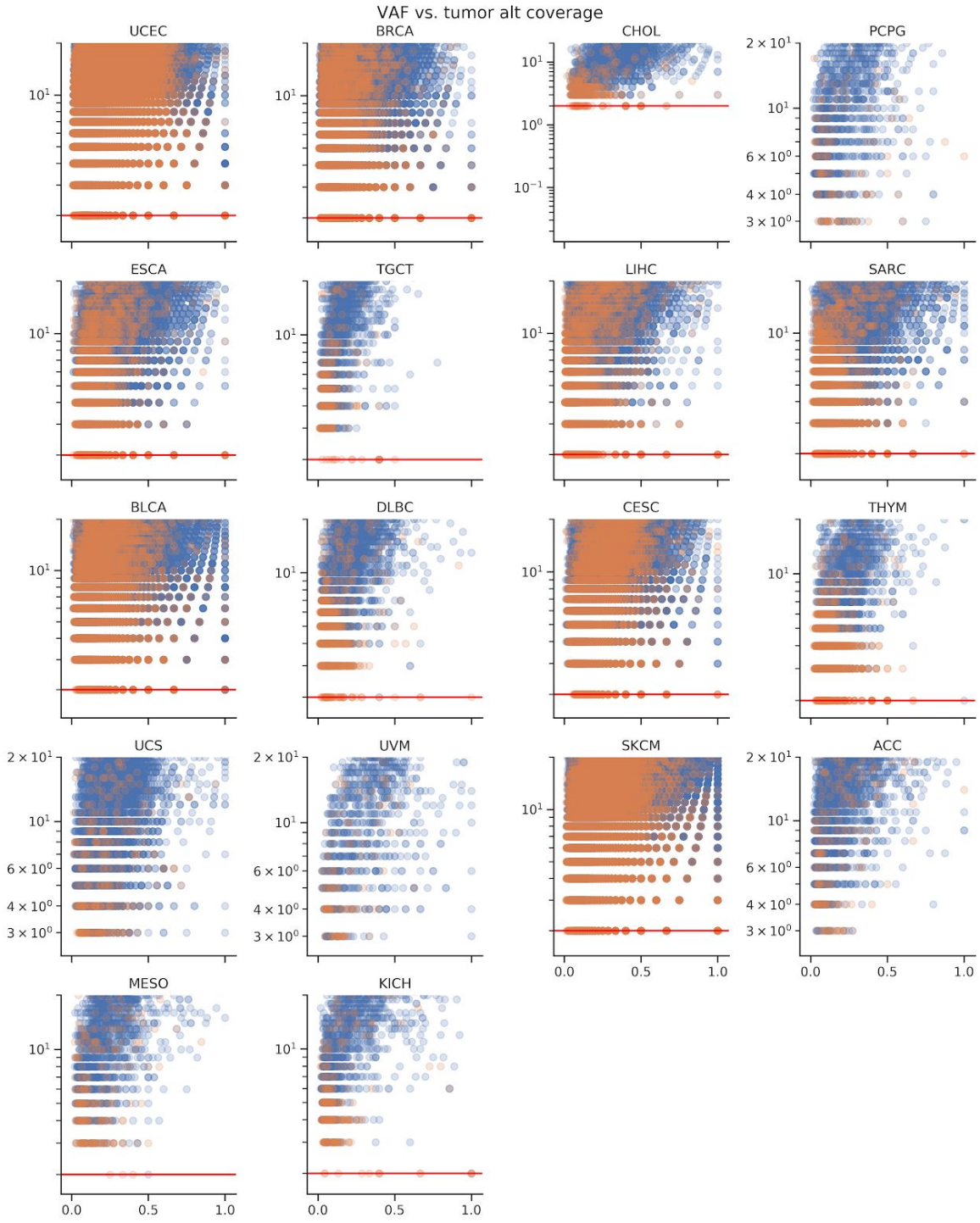


Figure 4-5: M2-only calls are enriched with low alternate allele fraction mutations and ones with a low number of alternate reads in the tumor. The orange dots are variants only called by M2; the blue dots are variants only called by both callers. The red horizontal lines denote the two alternate-supporting reads.

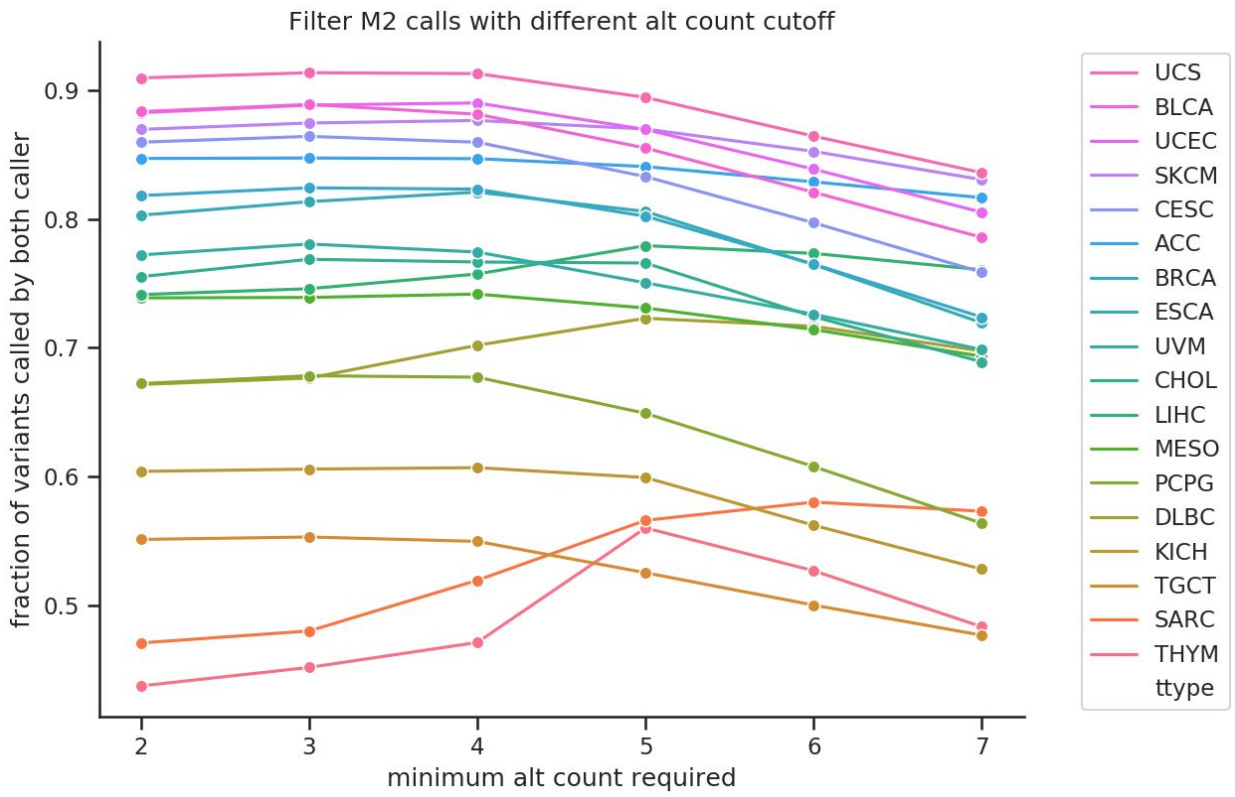


Figure 4-6: Sensitivity analysis of the fraction of concordant calls between M1 and M2, as a function of the minimum number of alternative reads.

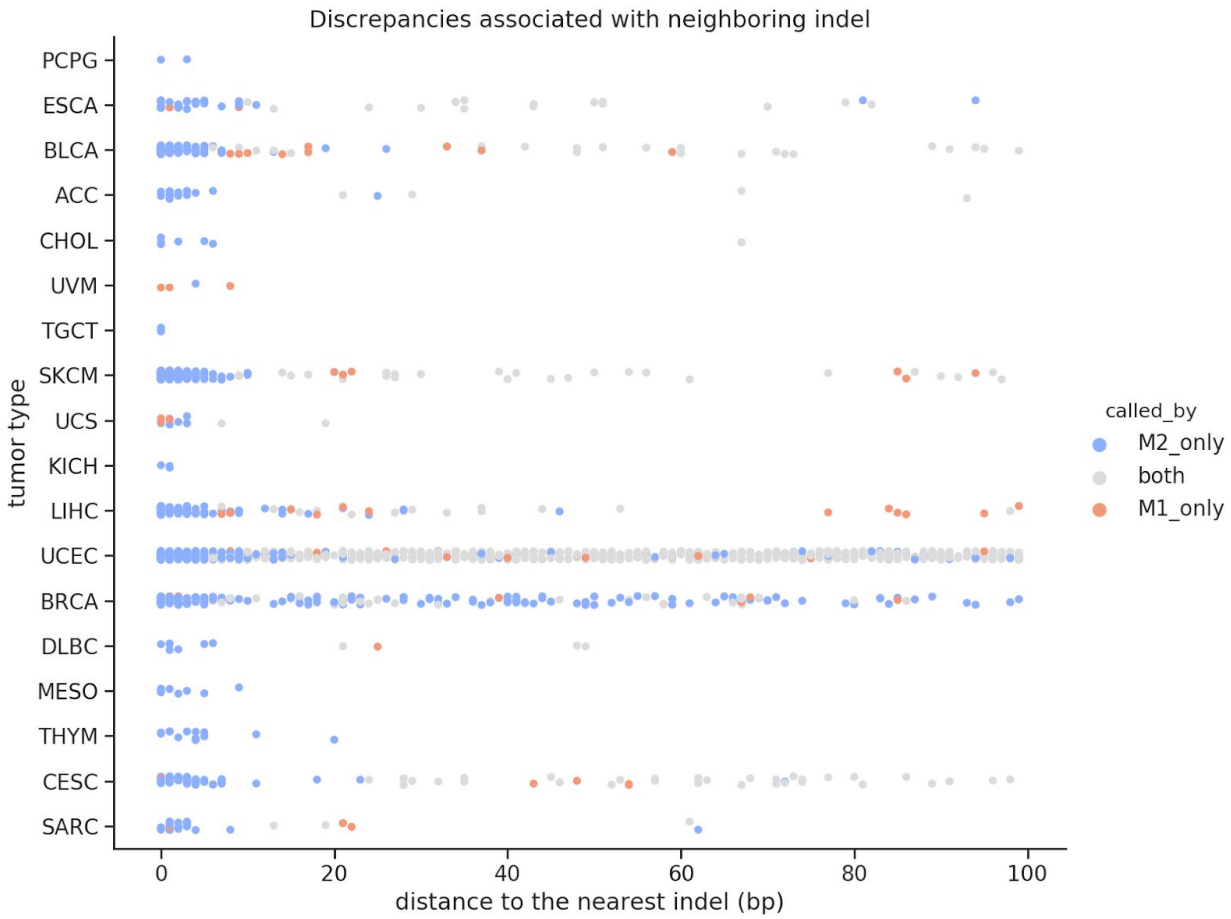


Figure 4-7: M1 filters out variants near regions with gaps (indels). Due to this filter, M1 can hardly call SNV near indels; however, less than 2% of calls have a neighboring indel within a 20bp distance.

4.2 MutSig shows diverging sets of significant genes

MutSig constructs a background model for each gene and assesses whether there is a significant excess of nonsilent mutations, or whether there is an enrichment of functional or clustered mutations. Figure 4-8 displays the qvalue from the two call sets for each cohort. The cohorts can be classified into three subgroups: (i) Cholangiocarcinoma (CHOL),

pheochromocytoma and paraganglioma (PCPG), uterine carcinosarcoma (UCS), and kidney chromophobe (KICH), where the significant gene sets and the majority of rankings are preserved between the two call sets; (ii) Testicular germ cell tumors (TGCT), THYM, and mesothelioma (MESO) in which more significant genes were found using the M1 call set; and (iii) LIHC, breast invasive carcinoma (BRCA), cervical squamous cell carcinoma and endocervical adenocarcinoma (CESC), and UCEC in which many more significant genes were found using the M2 call set.

Since MutSig aggregates the significance from three aspects (described above) , I also investigated which specific p-value is responsible for the overall change in significance. I looked specifically into cohorts in group 3, where M2 gives many more significant genes. The QQ plot shown in Figure 4-9-1 shows more inflation in pCL, which represents clustering of variants across multiple patients. This result led us to focus our manual review efforts on recurrent variants, since they are responsible for pCL inflation (Figure 4-9-2).

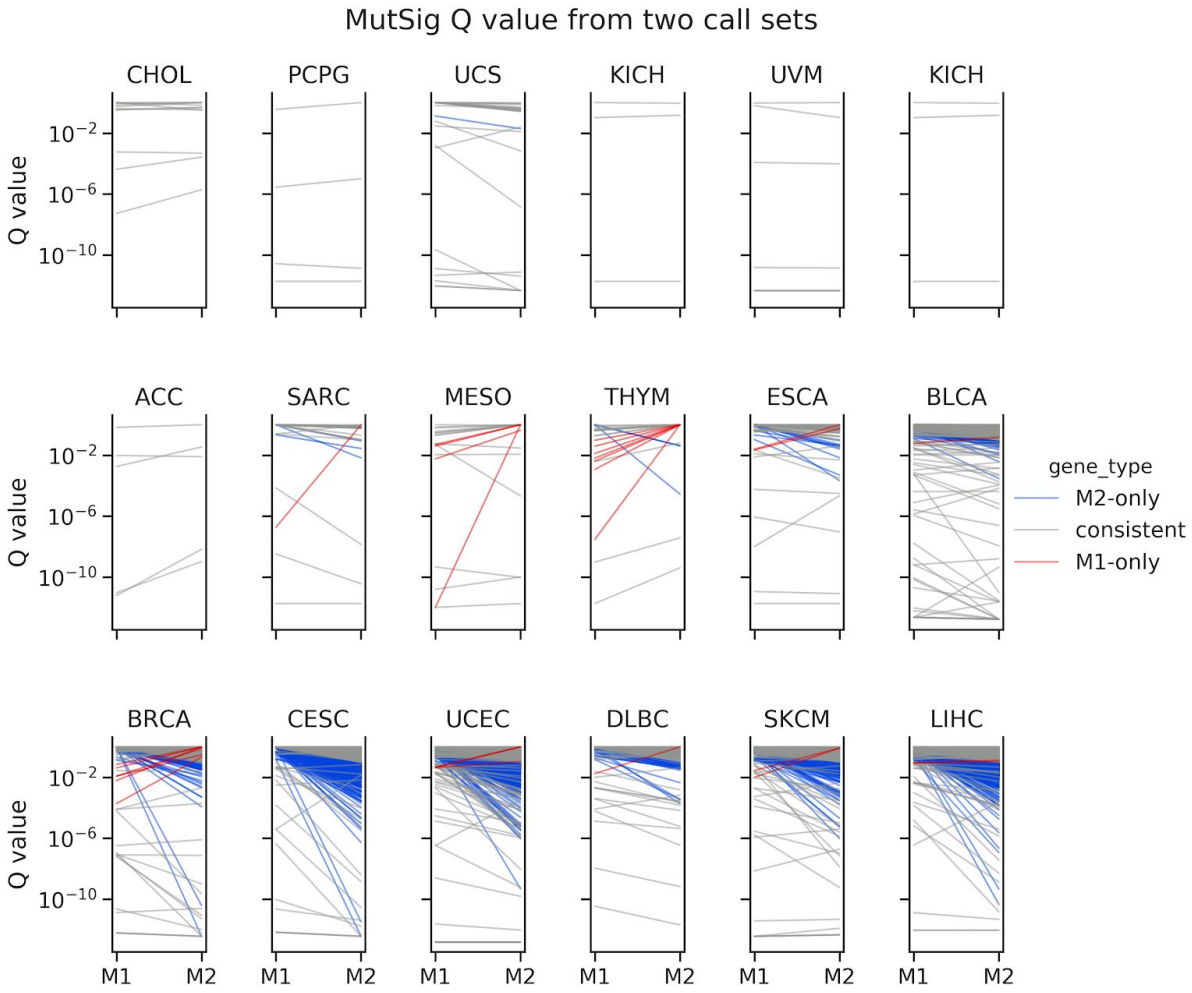


Figure 4-8: Gene significance by cohort. Each line represents a gene with two Q values from each of the call sets (M1 and M2). A red line denotes a gene significant when using the calls found by M1 but not M2 ("M1-only"); a blue line denotes a gene significant in M2 but not in M1 ("M2-only"). CHOL, PCPG, UCS, KICH, and UVM have a consistent set of significant genes; TGCT, THYM, and MESO have more significant genes found using the M1 call set; LIHC, BRCA, CESC, and UCEC show many more significant genes when using the M2 call set.

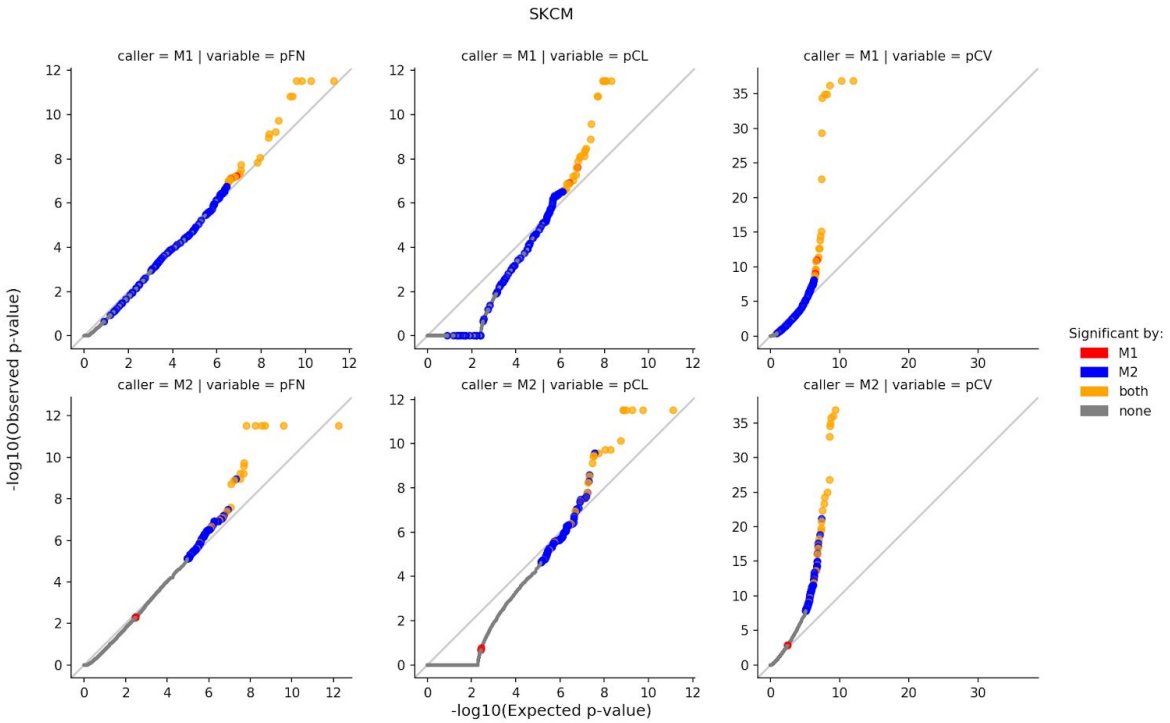


Figure 4-9-1: QQ plot for the three p-values in MutSig for the SKCM cohort. The most severe inflation is in pCL.

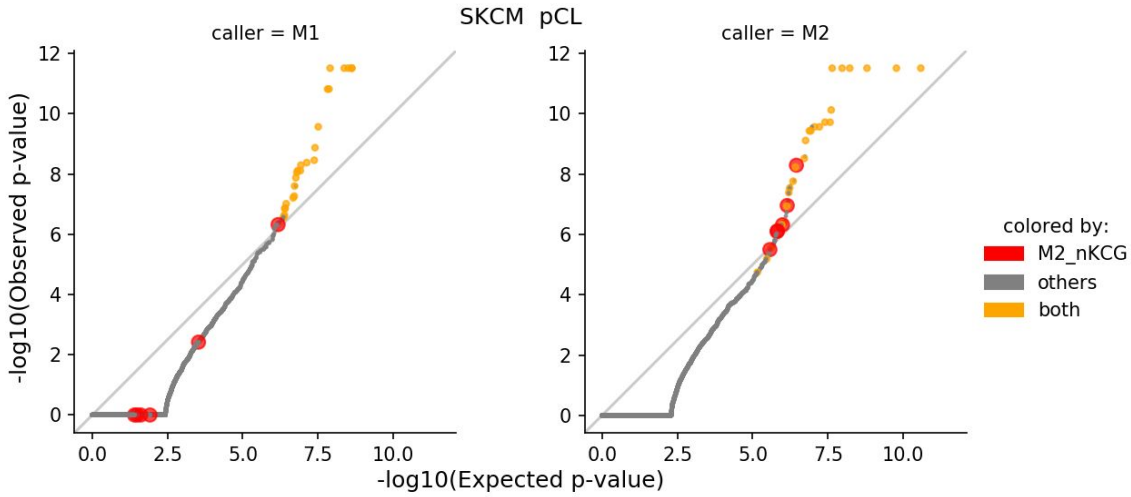


Figure 4-9-2: Recurrent calls from non-KCG are responsible for the inflation of pCL (in SKCM).

4.3 Recurrent events significant by only one caller shed light on systematic artifacts

In this section, I manually review some recurrent variants called only by one caller, stratified by whether a gene is a KCG or not. Non-KCGs are more likely to be false positives, and KCGs missed by one caller could represent false negatives. Figure 4-10 shows the list of the recurrent variants called only by M1, ranked by frequency; Figure 4-11 similarly shows the list for M2.

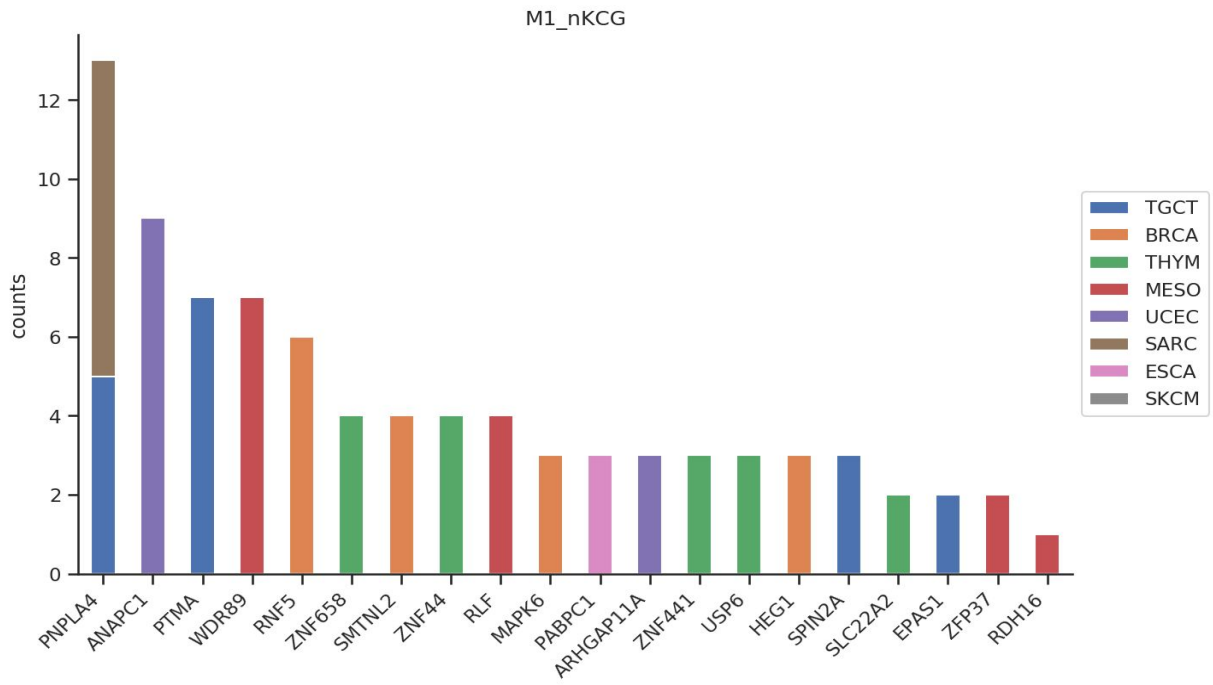


Figure 4-10: The occurrences of most recurrent variants called by M1 in differentially significant genes that are not currently known to be cancer-related (non-KCGs).

For cohorts in group 1, the limited number of patients (Figure 4-1) limits the statistical power to find driver genes; therefore, as expected, only a limited number of significant genes was found

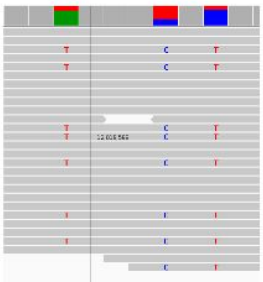
in this cohort — only the really prominent signals can be distinguished from the background mutation rate (i.e., the noise).

For cohorts in group 2, the M1-only significant genes usually represent genes with known mapping issues. Except for *FOXA1* and *CTCF*, most of these genes do not overlap with our list of known cancer genes, and they are all cohort-specific (only deemed to be significant in a single cohort), suggesting that most of them could be (mapping) artifacts.

To test this hypothesis, I reviewed some recurrent variants that are called by M1 but not by M2. *CTCF* and *FOXA1* are two M1-only genes from the BRCA cohort. *CTCF* is affected by the strand bias filter in a region with very few reads supporting the mutation in the tumor (<5), and the "clustered_events" filter behaves as expected in three missense mutations that were filtered out by M1 but were called by M2. The "clustered_events" filter is based on the number of events in an aligned *region;* the presence of 3 or more clustered events suggests that the read is mismapped. However, a closer observation revealed that the three variants are in fact on three independent haplotypes within this region. Therefore, in this case, I think that the "clustered_events" filter was too strict and likely filtered out true variants.

FOXA1, chr14 38061249/261/312

MAP2K4, chr17 12016571

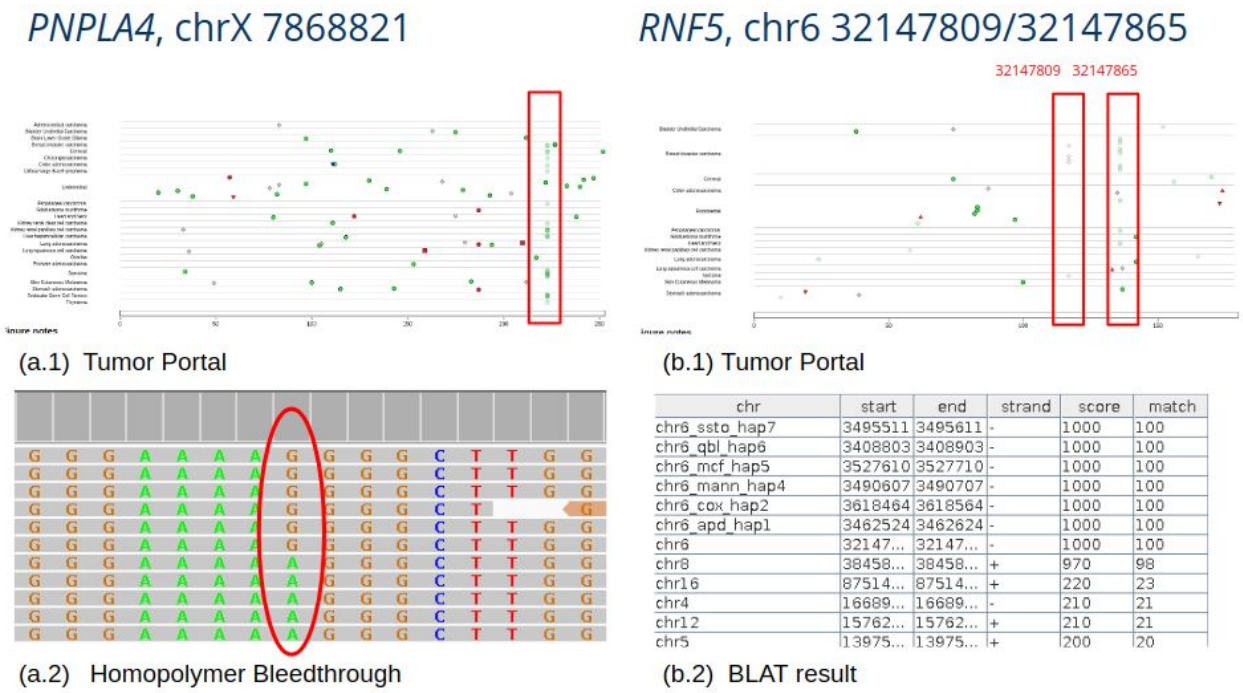


chr	start	end	strand	score	match
chr17	12016559	12016659	+	1000	100
chrX	72745196	72745296	-	880	94
chr17	39329140	39329167	+	220	23

chr	start	end	strand	score	match
chr17	12016535	12016635	-	940	97
chrX	72745220	72745308	+	740	81
chr1	192506...	192506...	+	320	34
chr20	57410245	57410268	+	210	22

Figure 4-10-1: Two examples of the M2 "clustered_events" filter. (a) in *FOXA1*, the three variants occurred in different haplotypes, suggesting that the filter is likely overly aggressive. (b) An example of this filter working properly, wherein it removes reads that were incorrectly mapped, as shown in the BLAT results (right).

Mutations in the *PNPLA4* X:7868821 were called 5 times in TGCT and 8 times in SARC by M1. However, reads in this region have some mapping or sequencing issues (see Figure 4-10-2 a.1), as evident by the homopolymer run that can cause a "bleedthrough" sequencing artifact (Figure 4-10-2 a.2). The two artifacts in *RNF5* are removed from M2 by the "mapping quality" filter. This gene is a zinc-finger protein with several pseudogenes, located on a genomic region with many germline variations (which may also reflect alignment artifacts). It aligns to several MHC haplotypes and regions on other contigs (Figure 4-10-2 b.2). Similar genomic polymorphisms are found at HLA-B.



chr	start	end	strand	score	match
chr6_ssto_hap7	3495511	3495611	-	1000	100
chr6_qbl_hap6	3408803	3408903	-	1000	100
chr6_mcf_hap5	3527610	3527710	-	1000	100
chr6_mann_hap4	3490607	3490707	-	1000	100
chr6_cox_hap2	3618464	3618564	-	1000	100
chr6_apd_hap1	3462524	3462624	-	1000	100
chr6	32147...	32147...	-	1000	100
chr8	38458...	38458...	+	970	98
chr16	87514...	87514...	+	220	23
chr4	16689...	16689...	-	210	21
chr12	15762...	15762...	+	210	21
chr5	13975...	13975...	+	200	20

Figure 4-10-2: Recurrent mapping artifact caught by M2's postfilter but are not removed by M1.

The recurrent artifacts in *HEG1* and *SMTNL2* are associated with problematic base qualities (Figure 4-10-3). In M1, the base qualities are used in the prefiltering step and the somatic detection model. M2 has an additional post-filter, called "base quality", that filters variants by the median base quality of all supporting bases. This filter helps remove artifacts called by the somatic detection model by ensuring that the bases supporting the variant do not have a systematically lower quality compared to other bases. In both genes, a "smear" of alternate bases across reads with a wide range of base qualities is observed. The M2 "base quality" filter removes these variants. Notably, both of these artifacts are not related to mismapping or sequencing bleedthrough.

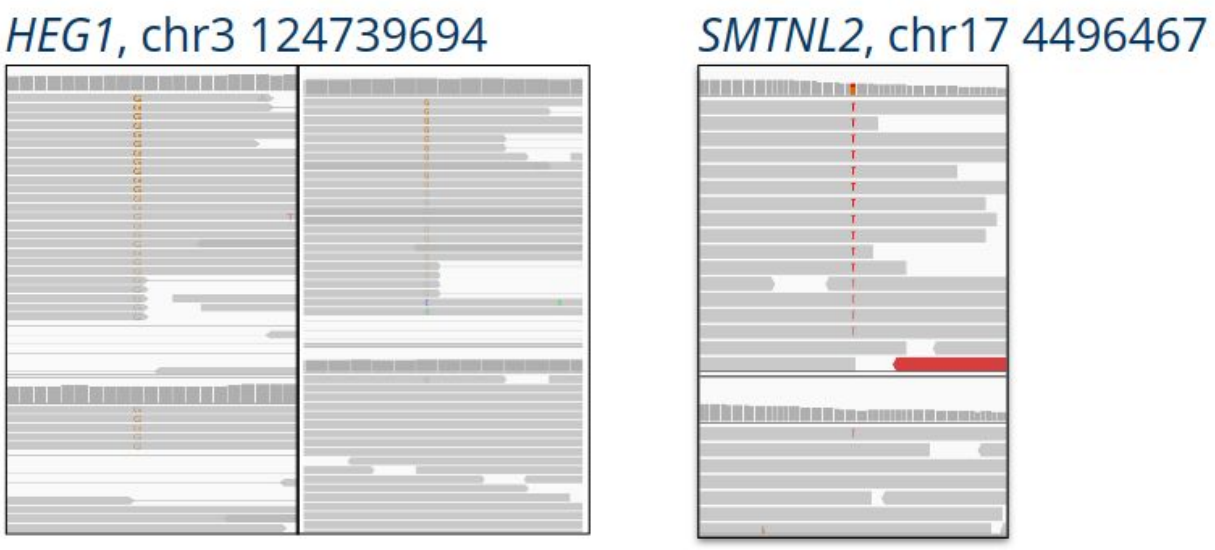


Figure 4-10-3: The "base quality" filter in M2 removes these sequencing artifacts, which were not removed by M1.

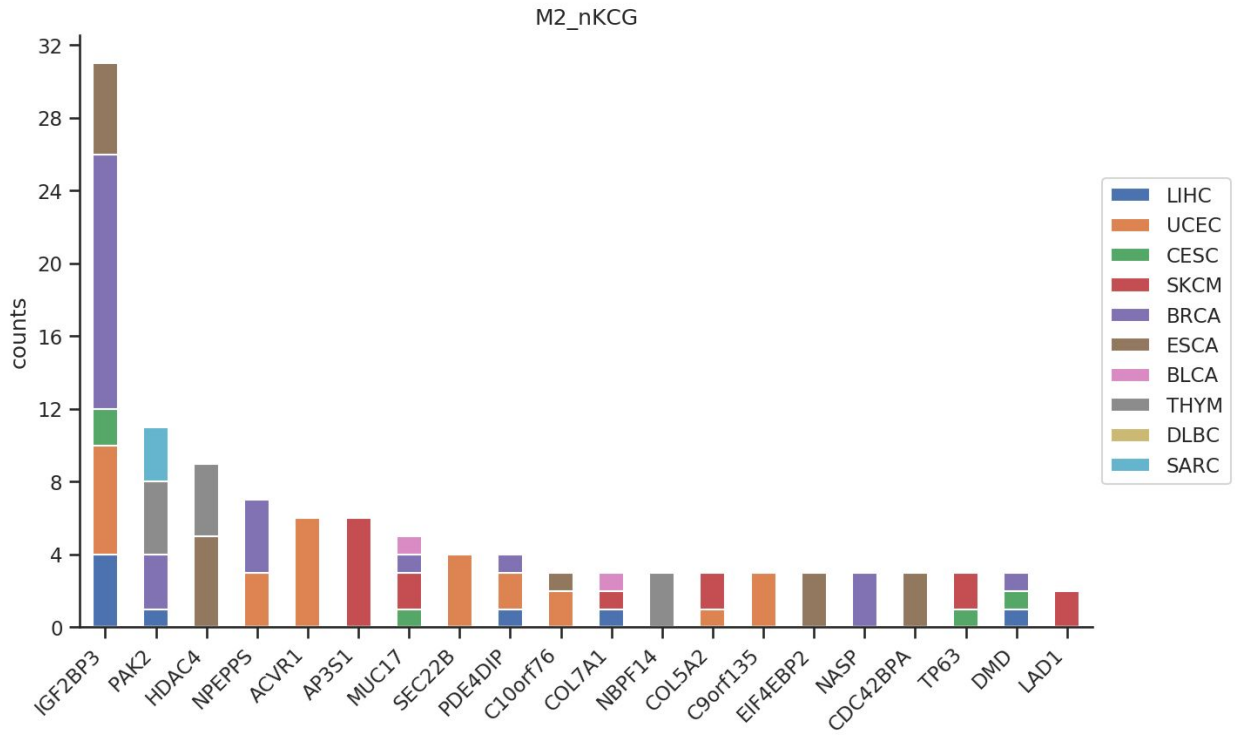
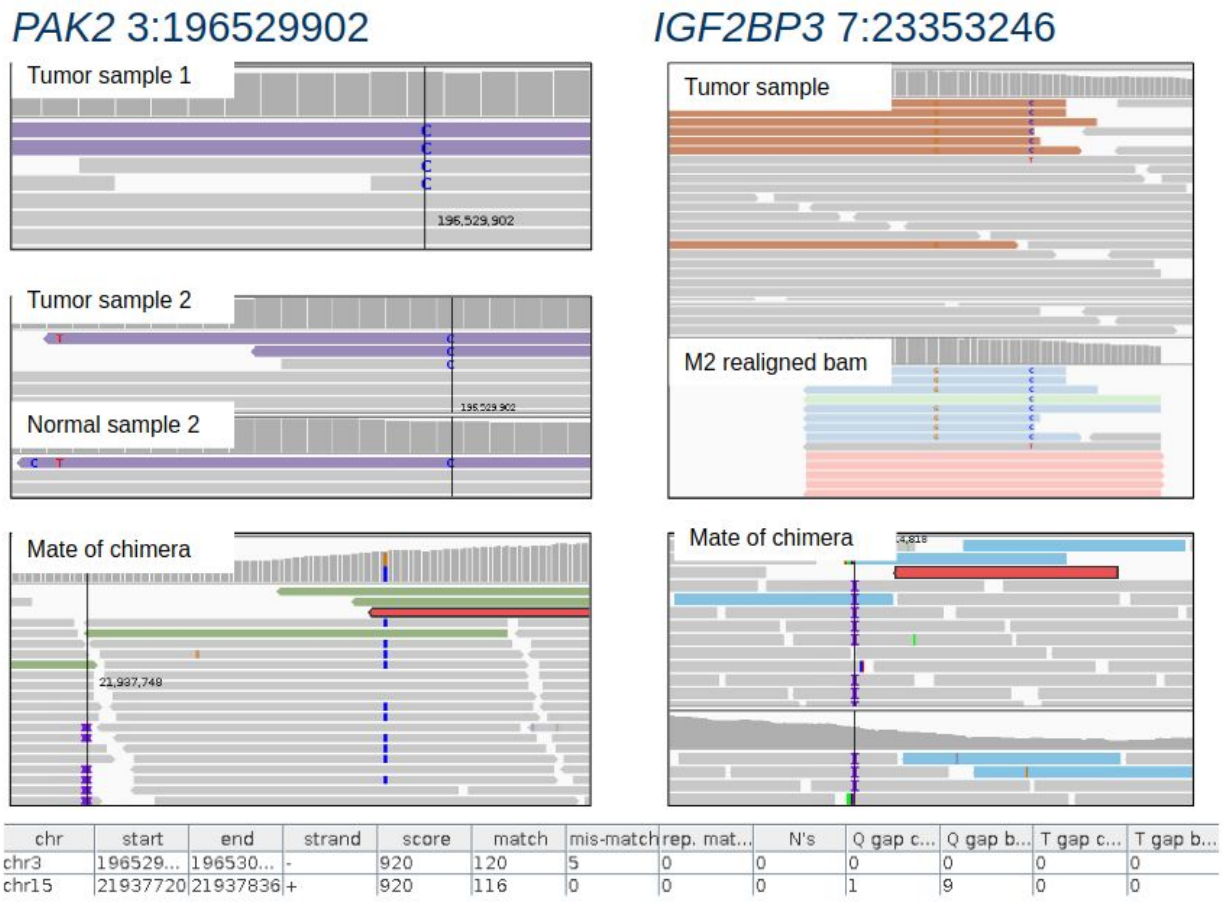


Figure 4-11: The occurrences of most recurrent variants called by M2 in differentially significant genes that are not known to be cancer-related (non-KCGs).

I also checked the recurrent variants in non-KCGs significant in the M2 call set (Figure 4-11) . Manual review of the variants in *IGF2BP3* and *PAK2* reveals some issues with M2's realignment (Figure 4-11-1). In both cases, the variants are found on chimeric reads bearing other variants, and there are several alternative mappings for their mate on the same chromosome with more mismatches or gaps. For an unexplained reason, M2 does not discard those chimeras (as expected in the prefiltering step), which results in these chimeras being assembled and called.



chr	start	end	strand	score	match	mis-match	rep. mat...	N's	Q gap c...	Q gap b...	T gap c...	T gap b...
chr3	196529...	196530...	-	920	120	5	0	0	0	0	0	0
chr15	21937720	21937836	+	920	116	0	0	0	1	9	0	0

Figure 4-11-1: M2 did not filter out the chimeric reads. After realignment, M2 calls these chimeric reads as new haplotypes that represent somatic events.

Variants called only by M2 in *NPEPPS* are the result of joint calling of SNPs and indels with M2 (Figure 4-11-2). The variant is located downstream of a low-complexity region where the mapping quality is generally 0. The variants that were called are downstream of alignment gap events or indels.

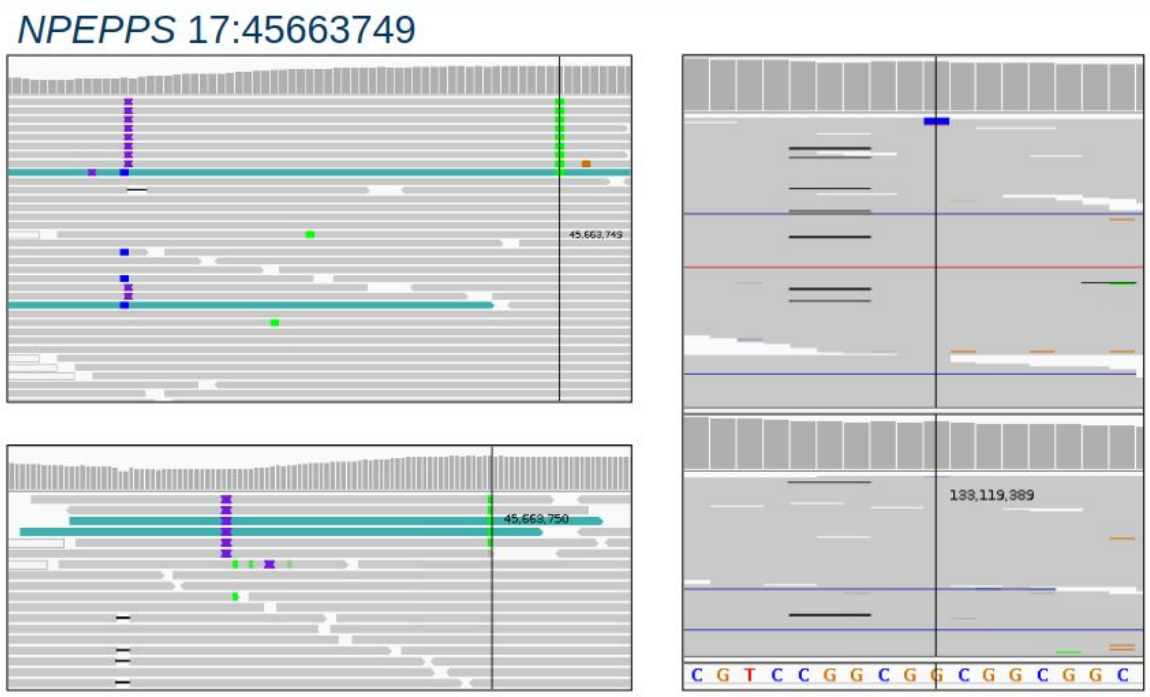


Figure 4-11-2: Reads with mutations potentially affected by the gaps/indels in a low-complexity region (i.e., repeats of CGG).

5. Discussion

5.1 Call sets that appear to be overall concordant might yield significant differences in driver gene discovery

Many previous benchmark studies that compared mutation detection methods focused on concordant analysis, reporting the fractions of calls (out of the union of all call sets) that were detected by a certain number (or specific subsets) of callers. However, our results reveal that even if two call sets share 88% of the variants (as for M1 and M2 in the UCEC cohort), the MutSig results can dramatically differ. This could suggest that the difference from the calls

identified in one caller have systematic artifacts in particular positions, and hence influence the MutSigCL (pCL). Alternatively, it could also suggest that the difference in mutations have slightly altered the MutSig background model, causing a gene that was not significant in one set of calls to become significant in another. The latter option is less likely to yield very different lists of significant genes between M1 and M2 since the lists share many of the variants.

5.2 MutSig's statistical modeling can help identify systematic artifacts in mutation detection methods that affect the same site or gene

In this study, I demonstrated that MutSig can be used as a prioritization tool for systematic artifacts. The highly inflated pCL value indicates that the major contribution to the difference in significance is associated with clustering of variants across patients, which pointed us to various mapping and post-filter artifacts.

5.3 M2 post-filters remove mapping and base quality artifacts, but might be overly aggressive when filtering "clustered events"

The M2 post-filters are responsible for ~50% of the variants that were called only by M1 (i.e., they were detected by M2 but then filtered out). The "mapping quality" and "base quality" filters help remove the *RNF5* / *PNPLA4* and *HEG1* / *SMTNL* artifacts, respectively; however, the

"clustered_events" filter may have been too aggressive since it removed variants based on the number of calls in a region, rather than examining the read-level or haplotype-level variants. The use of "strand-bias" and "orientation-bias" filters might also have been too aggressive since they removed mutations with a limited number of reads with the alternate base, where the filters might not be sufficiently powered to distinguish whether there is a strand-bias or not.

5.4 M2's reassembly and realignment step can produce artifactual haplotypes that might lead to false positives

In order to analyze the M2 realignment step, one needs to run M2 with the "--debug" flag (i.e., debug mode) or use the "-bamout" option. Li (Li 2014) mentioned that most of the discordant germline calls come from low-complexity regions determined by DUST (Li 2014; Morgulis et al. 2006), but many callers that perform realignment fail in very obvious cases. Several reports (Li 2014; Morgulis et al. 2006; Tian et al. 2016) also noted that local alignment has little impact on SNP calling compared to indel calling, and I also have shown here in the present work that only 2% of the calls have a neighboring indel event within 20bp. This could indicate that the inclusion of realignment in many recent mutation detection methods does not necessarily reduce false positives compared to an alternative strategy where I simply filter those calls with proximal gaps, as is done in M1.

5.5 A joint caller would be superior

To further improve somatic calling, several design choices will need to be made. First, the potential confounding factor of indels and gaps will have to be dealt with. The Lancet tool—a recent effort that assembled a colored De Brujin Graph for both tumor and normal samples (Narzisi et al. 2018) — is supposed to be more specific than M2. Moreover, the recent advancements in personalized graph genome make it a more appropriate method than the uniform reference method to compare against (Groza et al. 2020). Second, other aspects of characterizing the cancer genome will have to be considered. M2 uses a model that clusters allele fractions in order to adjust the prior probability of the allele fractions in the somatic detection model. However, the current modeling of subclones in M2 does not take into account the effect of copy number variations and different mutational multiplicities, or even the distribution of allele fractions associated with neutral expansion. Historically, separate methods were used to model clonal and subclonal structure of cancer samples, but the idea of incorporating these tools into the mutation calling methods is worthwhile and can produce improved results.

6 Conclusion

6.1 Contribution

This study compares call sets produced by two mutation detection methods, M1 and M2, in two aspects:

i) At the variant level, I demonstrate that one can expect at least 50% calls to be identified by both M1 and M2. The exact number may vary by cohort.

ii) At the significant gene level, significant genes identified by the two call sets show high levels of divergence. This suggests that some systematic features in the mutations called in the genes are uniquely significant in each of the mutations sets.

I manually reviewed calls on genes that were only deemed to be significant by one call set, and then stratified those genes by whether or not they are known cancer genes (KCGs) or not (non-KCGs). Recurrent calls on genes significant only using the M1 call set are usually removed by the M2 post-filters (e.g., “clustered events”, “base quality”, and “map quality”). I assume that “clustered events” may have been too aggressive, but “base_quality” did remove sequencing artifacts, and “map_quality” helped remove mapping artifacts independent from the Panel-of-Normals.

For calls on genes that are only significant in M2, most are on chimeric or rare haplotypes that triggered M2’s reassembly and realignment. However, I do not currently know definitively

whether those haplotypes are real events or mapping artifacts. It is also possible that those calls have neighboring gaps wherein the alignment needs to balance the tradeoff between gaps and mismatches. Unfortunately, I did not find a simple way to collect the realignment statistics from M2 to assess the contribution of the realignment step in calling SNVs.

To conclude, the majority of the discrepant calls are located in low-complexity regions (LCRs) where the uniqueness is limited by sequence context. While M2 seems to handle general mapping artifacts quite well, increases in the level of mapping issues can trigger the M2's realignment step and produce haplotypes that support artifactual alternative alleles, hence increasing false positive mutations.

6.2 Future Directions

Accurate mutation calling is crucial to many downstream analysis, e.g. tumor subtyping, signature analysis and phylogenetic analysis. In the future, it would be of interest to compare how different call sets produced by each caller affect those too, especially for the phylogenetic analysis where subclones are sometimes identified by only a handful of mutations.

I believe that the effect of the M2 realignment on SNV calling should be further evaluated, and the fraction of false positive SNVs that could be affected by neighboring indels should be estimated. Although the number of calls affected is believed to be limited, they often appear to be clustered and may disrupt significance models.

In addition, the MutSig significance model needs to be updated to use a base-level mutatibility estimate. Although the work that I presented here did not focus on benchmarking MutSig, the

QQ plot of p-values shows some deviation from the null, especially for pCL. Given our new knowledge of passenger hotspots (Hess et al. 2019), the list of driver genes could have been more robust to sporadic discrepancies in the call sets.

Supplementary Information

Supplementary Tables

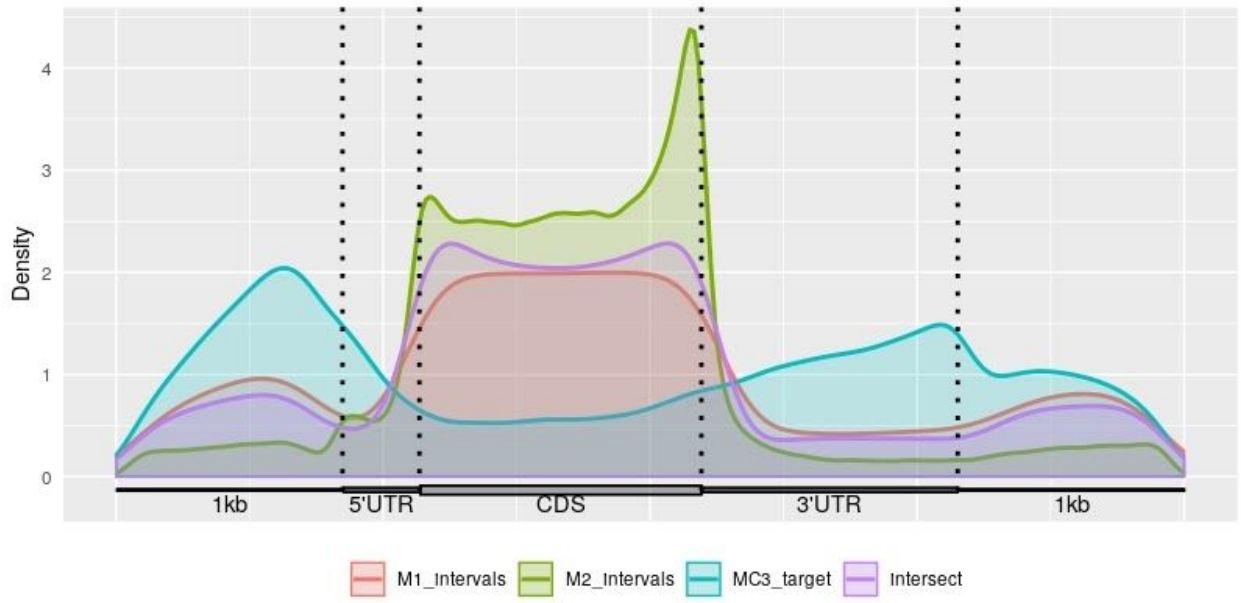
Supplementary Table 1. Abbreviations

NGS	Next-generation sSequencing
TCGA	The Cancer Genome Atlas
WES	Whole exome sequencing
PCR	Polymerase Chain Reaction
GATK	Genome Analysis Toolkit
VAF	Variant allele fraction
IGV	Integrated Genomics Viewer
LOD	Log odds
NMF	Non-negative Matrix Factorization
LCR	Low Complexity Region
KCG	Known Cancer Genes
SNP	Single nucleotide polymorphism
INDEL	Insertion / deletion

Supplementary Table 2. Basic information of 18 cohorts

Study Abbreviation	Study Name	Number of patients
ACC	Adrenocortical carcinoma	91
BLCA	Bladder Urothelial Carcinoma	409
BRCA	Breast invasive carcinoma	665
CESC	Cervical squamous cell carcinoma and endocervical adenocarcinoma	282
CHOL	Cholangiocarcinoma	36
DLBC	Lymphoid Neoplasm Diffuse Large B-cell Lymphoma	37
ESCA	Esophageal carcinoma	180
KICH	Kidney Chromophobe	65
LIHC	Liver hepatocellular carcinoma	360
MESO	Mesothelioma	81
PCPG	Pheochromocytoma and Paraganglioma	177
SARC	Sarcoma	232
SKCM	Skin Cutaneous Melanoma	463
TGCT	Testicular Germ Cell Tumors	144
THYM	Thymoma	121
UCEC	Uterine Corpus Endometrial Carcinoma	435
UCS	Uterine Carcinosarcoma	57
UVM	Uveal Melanoma	80

Supplementary Figures



Supplementary Figure 1. Comparison of three intervals. M1 and M2 Intervals are the two interval lists used by the two callers. The MC3 target list was used in the MC3 project to integrate the results of different callers. "Intersect" denotes the intersection of all three interval lists. I only included variants in this "Intersect" when evaluating the concordance between the two methods and the different MutSig results.

Significant gene list

[significant genes only from the M1 call set]: ANAPC1, AP2B1, ARHGAP11A, BMPER, CRLF3, CTCF, EPAS1, FOXA1, HEG1, HHLA2, HLA-B, KALRN, MAPK6, NHEDC1, PABPC1, PI4K2A, PNPLA4, PTMA, RDH16,RLF,RNF5,SKIL,SLC22A2,SMTNL2,SPIN2A, TP53, USP6, WDR89, ZFP37, ZNF44, ZNF441, ZNF658

[KCGs significant only from the M2 call set]:
ABCB5,ACTB,ACTC1,ACVR1B,ACVR2A,AKT1,ARHGAP26,ARID1A,ARID2,ARID5B,ASXL1,ATM,ATR,ATRX,B2M,BAP1,BMPR1A,BRCA1,C15orf23,CARD11,CASP8,CBFB,CBLC,CD70,CDH1,CDK12,CDKN1A,CDKN1B,CHD4,CIC,COL1A1,COL2A1,CREBBP,CTCF,CTNNB1,DDX3X,DICER1,DNMT3A,DSG3,EGFR,EP300,EPHA2,ERBB3,ERCC4,EZH2,FANCD2,FAS,FAT1,FBXW7,FGFR2,FLT3,GNA11,GNAS,GPC3,HDAC2,HIST1H1C,HIST1H3B,HNF1A,HUWE1,ID3,ING1,INPPL1,IRF2,JAK1,JAK2,KDM6A,KDR,KEAP1,KIT,LZTR1,MAP2K4,MAP3K13,MAPK1,MAX,MED12,MLL2,MLL3,MLL4,MSH6,MTOR,MVK,MYCN,NCOR1,NF2,NRAS,NSD1,P2RY8,PBRM1,PHF6,PIK3R1,PIK3R3,POU2F2,PTEN,RASA1,RB1,RECQL4,RHOB,RNF43,RPS6KA3,SDHAF2,SETD2,SF3B1,SFRS2,SMAD4,SMC1A,SMC3,STAT3,STK11,SUFU,TAF1,TAF1L,TBL1XR1,TG,TP53,TRAF3,TSC1,U2AF1,UBR5,WT1,ZFP36L1

[non-KCGs significant only from the M2 call set]:
AARSD1,ABCA13,ABCB1,ABCC2,ABCC3,ABCC8,ABCG2,ABCG5,ABHD10,ACACA,ACAP2,ACAT2,ACBD7,ACCN4,ACPP,
ACPT,ACSBG1,ACSF2,ACSF3,ACSL5,ACSM2B,ACSM4,ACTG1,ACTN3,ACVR1,ADAM12,ADAM7,ADAM9,ADAMTS2,A
DAMTS3,ADAMTS7,ADAMTSL5,ADARB2,ADAT1,ADCY10,ADNP,AFAP1,AFF2,AFF3,AGPAT9,AGPS,AGXT2,AHCTF1,A
HCYL1,AIM1,AK7,AKAP7,ALAD,ALDH6A1,ALG8,ALKBH6,ALMS1,ALPK2,ALS2,AMBRA1,AMD1,AMPD1,ANGPT1,ANK
3,ANKHD1,ANKRD12,ANKRD13D,ANLN,ANO6,ANO7,ANO9,ANXA3,ANXA5,AOX1,AP2B1,AP3S1,APBB1IP,APCS,APO
BEC3G,AQP11,AQR,ARAP3,ARFGEF2,ARHGAP28,ARHGAP4,ARHGEF11,ARHGEF2,ARHGEF6,ARID4B,ARL8B,ARL9,A
RMC8,ARPC1A,ARRB2,ASAM,ASAP2,ASB3,ASF1B,ASNS,ASTL,ATAD2,ATAD5,ATG2B,ATG5,ATP10D,ATP1A2,ATP1A3
,ATP2B1,ATP2C2,ATP4A,ATP6V0A1,ATP6V0A2,ATP6V1H,ATP8A1,ATP8B2,ATXN2L,B4GALNT1,BACH1,BAX,BAZ2B,B
BS2,BCAP29,BCL10,BCL11A,BCL3,BCLAF1,BDP1,BEST1,BFAR,BGN,BMP2K,BMX,BRAP,BRD1,BRD7,BRE,BRSK2,BSN,
BTAF1,BTG4,BTK,C10orf12,C10orf28,C10orf76,C10orf78,C11orf2,C11orf61,C12orf43,C12orf72,C13orf18,C13orf3
9,C14orf145,C14orf2,C14orf50,C16orf46,C16orf71,C17orf57,C17orf78,C17orf89,C19orf51,C19orf63,C1QTNF9,C1
orf106,C1orf146,C1orf174,C1orf93,C20orf107,C20orf152,C20orf185,C20orf194,C20orf26,C20orf27,C20orf71,C2C
D3,C3,C3orf23,C3orf63,C3orf64,C3orf70,C4orf34,C4orf43,C5orf34,C6,C6orf15,C6orf165,C6orf167,C7orf57,C7orf5
8,C8orf76,C9orf102,C9orf128,C9orf135,C9orf93,CA8,CA9,CABIN1,CABP5,CACNA1E,CACNA1F,CACNA1S,CACNA2D
1,CADM2,CALCOCO1,CALCRL,CAMTA1,CAPN11,CAPN9,CAPZA1,CARD6,CASC3,CASD1,CATSPER1,CATSPER3,CBX1,
CCAR1,CCDC125,CCDC13,CCDC137,CCDC141,CCDC146,CCDC158,CCDC33,CCDC53,CCDC58,CCDC76,CCDC77,CCDC
80,CCR6,CCT2,CCT3,CD109,CD163L1,CD247,CD38,CD40LG,CD53,CD72,CD80,CD81,CDA,CDC123,CDC25A,CDC25C,
CDC42BPA,CDC42EP1,CDC7,CDCA7,CDH17,CDHR2,CDK5RAP2,CDK8,CDK9,CEACAM3,CEACAM4,CEACAM8,CEP192
,CEP350,CEP72,CERK,CGB7,CGN,CHD3,CHD8,CHL1,CHMP6,CHPF,CHRM1,CHRNA1,CHRNB1,CHUK,CILP2,CISH,CKM
T2,CLDN1,CLDN2,CLIC2,CLIP4,CLK2,CLK4,CLSTN2,CLTC,CLTCL1,CNOT2,CNTLN,COBLL1,COG2,COL11A1,COL11A2,C
OL12A1,COL16A1,COL19A1,COL1A2,COL22A1,COL24A1,COL4A1,COL4A2,COL4A4,COL4A5,COL4A6,COL5A1,COL5
A2,COL5A3,COL6A3,COL6A6,COL7A1,COL9A1,COL9A2,COMMD6,COMMD7,COMP,COPA,CP,CPN1,CPS1,CPSF1,CP
T1A,CPT1C,CR1L,CR2,CRABP2,CREB3L3,CRHR2,CRTC2,CSDE1,CSE1L,CSF2RB,CSN3,CTLA4,CTNNBIP1,CTNND1,CTN
ND2,CTSH,CTSL1,CUBN,CUL4B,CXCL1,CXCR4,CXorf56,CXorf58,CYP11B2,CYP3A4,CYP3A5,CYP4B1,CYP8B1,CYTSB,D
AAM1,DAB2IP,DAO,DBN1,DCAF12,DCAF6,DCAF8,DCBLD1,DCHS1,DDR2,DDX23,DDX4,DDX42,DDX47,DDX50,DEFB
113,DENND2C,DENND4A,DENND5B,DEPDC5,DGAT2L6,DHRS12,DHRS4L1,DHX35,DHX37,DHX9,DIAPH1,DIDO1,DM
D,DMKN,DNAH1,DNAH17,DNAH2,DNAI2,DNAJA2,DNAJB14,DNAJB2,DNAJC13,DNAJC15,DNAJC6,DNER,DOCK10,D
OCK11,DOCK2,DOCK8,DPP4,DPP8,DPY19L3,DSC3,DTWD2,DTX3,DUSP22,DYM,DYNC1I1,DYRK1A,DYRK3,DYSF,DZIP
1,DZIP3,E2F6,EBF1,EBLN2,ECE2,ECHDC2,EDN1,EEA1,EEPD1,EFEMP1,EFEMP2,EGFL7,EGFLAM,EIF2AK1,EIF2B3,EIF2
S3,EIF3E,EIF4A2,EIF4E1B,EIF4EBP2,EIF4G1,EIF4G3,EIF5A,ELL2,ELMO2,EML1,EMR1,EMR3,ENGASE,ENKUR,ENTHD1
,EPAS1,EPB41L3,EPB41L4B,EPB42,EPC1,EPHA3,EPHX4,EPX,ERAP2,ERBB2IP,ESRP1,EVC2,EVPL,EXD1,EXOC3L2,F10,F
5,FAF1,FAH,FAM120B,FAM151A,FAM179B,FAM189A2,FAM3A,FAM48A,FAM49A,FAM65C,FAM70B,FAM81B,FAM
83G,FAM86C,FAM92B,FANCM,FARSB,FASTKD2,FAT2,FBN1,FBN2,FBN3,FBXL4,FBXO18,FBXO28,FCN1,FCRL1,FCRL6
,FCRLA,FETUB,FGA,FGD6,FGF14,FGFBP1,FGL2,FIG4,FILIP1,FKBP4,FKBP9,FLG,FLT4,FMO1,FMO4,FMR1,FN1,FOXR1,
FOXRED2,FRA10AC1,FRAS1,FRZB,FSTL1,FZD6,FZR1,GABBR1,GABRA1,GALNTL5,GANAB,GATAD2A,GBP2,GCG,GFPT
1,GFPT2,GH2,GIGYF1,GIMAP4,GIN1,GIP,GIPC2,GK,GLDC,GLG1,GLI2,GLT25D1,GLT25D2,GLT8D2,GLUD2,GMCL1,G
MEB1,GMPS,GNAI3,GNB5,GNL3L,GOLGA5,GOLGB1,GON4L,GOSR2,GOT1,GPA33,GPATCH3,GPATCH8,GPD2,GPKO
W,GPR137B,GPR174,GPR50,GPR55,GPR98,GPRASP2,GPSM2,GPX1,GRAMD1C,GREB1,GRHL2,GRIA1,GRIN1,GRIP2,
GRM8,GSDMC,GTDC1,GYPA,GZMA,H2AFY2,H3F3C,HABP4,HADHB,HAP1,HAUS1,HDAC3,HDAC4,HDAC8,HEATR1,H
ECTD1,HECW2,HELZ,HERC6,HEXIM1,HHIPL2,HIBCH,HIF3A,HIST1H1B,HIST1H2BC,HIST1H2BD,HK3,HLA-C,HLA-DOB,
HLA-E,HLTF,HMCN1,HMG20A,HNF4G,HNRNPC,HNRNPR,HORMAD1,HOXB9,HPS5,HSD17B4,HSP90AB1,HTATSF1,H
TR3C,HTR7,HTT,IARS,IDS,IFNA5,IFT46,IFT80,IGDCC3,IGF2BP3,IGSF3,IKZF1,IL12A,IL12RB1,IL13RA2,IL17B,IL1F8,IL1R
1,IL1RL2,IL21R,IL27RA,IL28A,IL31RA,IL34,IL5RA,IMPACT,INADL,ING5,INO80,INTS7,INTS8,IPO7,IPP,IQCH,IRAK3,IRE
B2,ITCH,ITGAD,ITGAM,ITGB2,ITGB5,ITIH5,ITLN1,ITPKC,ITPR3,ITSN1,IWS1,JARID2,JMJD1C,JPH2,JUB,KANK1,KAT2A,
KBTBD12,KCNA6,KCNA7,KCNAB1,KCNE2,KCNK9,KCNMB3,KCNN3,KCNQ2,KCTD18,KCTD19,KCTD20,KCTD21,KDM2
A,KDM3A,KIAA0090,KIAA0406,KIAA0427,KIAA0430,KIAA0753,KIAA0754,KIAA0913,KIAA1033,KIAA1267,KIAA1324
L,KIAA1377,KIAA1407,KIAA1432,KIAA2018,KIAA2026,KIF14,KIF1B,KIF3B,KIF4A,KIF7,KLHL2,KLHL20,KLHL6,KLK1,KL
KB1,KLRC4,KLRK1,KNDC1,KNTC1,KPNA6,KPNB1,KRT1,KRT15,KRT222,KRT28,KRT3,KRT35,KRT5,KRTAP10-11,KRTAP
10-12,KRTAP10-7,KRTAP10-8,KRTAP3-3,KTN1,KYNU,L1CAM,LAD1,LAMB2,LAMB3,LAMC3,LAPTM4A,LARP4,LASS2,
LATS1,LBR,LCE1B,LCK,LCMT2,LDHC,LEAP2,LEO1,LEPR,LGALS12,LGALS8,LGR6,LILRA3,LILRB5,LIN37,LIN9,LIPH,LLPH,
LMAN2,LMAN2L,LMBR1,LMF2,LNX2,LOC55908,LOC81691,LONP2,LONRF2,LRP1B,LRP2,LRP4,LRP6,LRPPRC,LRRC16
B,LRRC27,LRRC39,LRRC50,LRRC57,LTBP2,LTK,LTV1,LY6G6F,LY75,LYVE1,MACC1,MAGIX,MAGOHB,MAGT1,MAML3,

MAN2A2,MAN2B1,MAP3K5,MAP4K3,MAP7D2,MAP7D3,MAPKAP1,MARK1,MAT1A,MBNL1,MCART2,MCM10,MCM7,MCOLN2,MCRS1,MCTP2,MDN1,MED1,MED14,MED23,MED27,MED6,MEGF8,MELK,METTL3,MFAP1,MFAP5,MFF,MFSD3,MGA,MGRN1,MICAL1,MIIP,MLEC,MLH3,MLLT10,MLLT4,MLN,MLST8,MLXIPL,MMEL1,MNDA,MOBKL2C,MOCOS,MOCS1,MORC2,MORC4,MPO,MRE11A,MRPL47,MST1R,MTHFR,MTIF2,MTM1,MTMR1,MTX3,MUC17,MUC4,MUC5B,MVP,MYH15,MYH7,MYH7B,MYH9,MYO1B,MYO3B,MYO7B,MYO9A,MYOC,MYOF,MYST2,N6AMT1,NAA15,NAA30,NARG2,NASP,NAV1,NAV2,NBEA,NBPF1,NBPF14,NBR1,NCAPD3,NCBP1,NCK1,NCOA6,NCOA7,NCR3,NDC80,NDEL1,NDUFS1,NDUFS2,NECAP1,NEDD4L,NEK2,NFATC2,NFATC4,NFIC,NFKBID,NFXL1,NID2,NIPAL2,NKAIN3,NLGN1,NLRC5,NMT2,NOD2,NOS3,NOV,NOX1,NPBWR2,NPEPPS,NPR2,NPTX1,NR1D1,NR2C1,NR2C2AP,NR2E1,NRAP,NRBP1,NRIP2,NT5DC1,NUBPL,NUP133,NUP160,NUP210L,NUP93,NXF1,NYNRIN,OAZ3,OBFC2A,OCRL,ODF3,ODF4,OFD1,OGDH,OGN,OGT,OLFML2B,OPALIN,OPN4,OR10A3,OR10G7,OR2B2,OR2Y1,OR4D2,OSBPL1A,OSBPL9,OSGEPL1,OTUD5,PA2G4,PACSIN3,PAFAH1B1,PAK2,PAPD4,PAPOLA,PAPOLB,PAPOLG,PARD3,PARP14,PASK,PAX3,PBLD,PCBP1,PCDHGB6,PCSK7,PDCD7,PDE10A,PDE1C,PDE4B,PDE4DIP,PDE6A,PDE6B,PDE6C,PDE9A,PDK3,PDPR,PDS5B,PDZK1,PER1,PET112L,PEX5L,PGBD1,PHEX,PHLDB2,PIAS1,PIGK,PIH1D2,PIK3C2B,PION,PIWIL4,PKD1L1,PKD2L2,PKHD1,PKHD1L1,PKLR,PKM2,PKNOX2,PLA2G2E,PLA2G2F,PLA2G4A,PLA2R1,PLB1,PLCB1,PLCB4,PLCG2,PLCH1,PLD1,PLEK,PLEK2,PLEKHA6,PLEKHB2,PLEKHO2,PLK4,PLRG1,PLXNB2,POF1B,POFUT2,POLG2,POLN,POLQ,POLR2B,PON1,POU5F2,POU6F1,PPA2,PPAN,PPARA,PPDPF,PPFIBP1,PPIG,PPM1H,PPP1R12A,PPP1R12B,PPP1R16B,PPP2CA,PPP2CB,PPP2R3C,PPP2R5A,PRAME,PRCP,PREX1,PRIM1,PRKAA1,PRKCB,PRKD2,PRKDC,PRPF19,PRSS50,PRTG,PSD,PSIP1,PSMA4,PSMC6,PSMD12,PSME4,PTGIS,PTK2,PTK7,PTPN12,PTPN14,PTPN22,PTPN6,PTPRC,PTPRF,PTRF,PTRH1,PTTG2,PUM1,PYGM,PYGO2,PYHIN1,PZP,QTRT1,RAB18,RAB24,RAB3A,RABGGTB,RAD21,RAE1,RAI14,RALB,RALGAPB,RAPGEF2,RAPGEF4,RARG,RASAL2,RASGRP4,RAVER2,RB1CC1,RBBP8,RBL2,RBM26,RBM27,RBM39,RCN2,RCSD1,RECQL5,REEP2,REEP5,REN,REST,RFC1,RFC2,RFC3,RFC4,RFTN2,RFWD2,RG9MTD3,RGL2,RGNEF,RGS14,RGS2,RHOT1,RICTOR,RIOK1,RLTPR,RNF169,RNF216,RNF31,RNF6,ROBO2,ROCK2,RPA1,RPAP1,RPAP2,RPGR,RPL5,RRP12,RRP1B,RS1,RSL1D1,RSRC1,RTEL1,RUNX1T1,RXFP1,RYR1,S100A7A,SAGE1,SAMD4A,SAMD7,SAMM50,SAP130,SAPS3,SASH1,SASH3,SBF1,SBF2,SBNO1,SCAPER,SCN3A,SCNM1,SCRIB,SCYL2,SDCCAG1,SDHA,SDK1,SEC22B,SEC23IP,SEL1L3,SEMA3C,SEMA3E,SEMA4A,SEMA4B,SEMA6C,SEMA6D,SENP7,SEPP1,SEPT12,SERPINB11,SERPINB3,SERTAD4,SETD5,SFRS13B,SFRS15,SFRS18,SFT2D2,SGIP1,SGOL2,SGTB,SH3BGRL,SH3PXD2A,SH3TC2,SH3YL1,SHC1,SHD,SIAH1,SIGLEC10,SIGLEC7,SIPA1L1,SIRPD,SLAMF1,SLC10A4,SLC10A7,SLC11A2,SLC12A4,SLC12A6,SLC16A10,SLC16A9,SLC17A3,SLC19A3,SLC1A6,SLC20A1,SLC22A2,SLC22A8,SLC25A12,SLC26A11,SLC26A7,SLC26A8,SLC2A3,SLC30A1,SLC35F2,SLC35F3,SLC39A10,SLC39A8,SLC40A1,SLC44A1,SLC4A11,SLC4A7,SLC6A19,SLC6A4,SLC6A9,SLC8A1,SLC9A10,SLC9A11,SLC9A7,SLCO2A1,SLFN11,SLFN13,SMAD3,SMARCA1,SMARCA5,SMARCAD1,SMC6,SMEK1,SMG7,SNAI1,SNCAIP,SNX10,SNX21,SOCS2,SOCS6,SOLH,SOS1,SOX2,SOX5,SPAG17,SPARC,SPATS2L,SPEG,SPINK9,SPNS1,SPON1,SPRR4,SPTA1,SPTAN1,SPTBN1,SRBD1,SRC,SRPK2,SRRM2,SSH3,ST8SIA3,ST8SIA4,STAB1,STAB2,STAC2,STAG1,STAG3,STAM,STARD6,STK24,STK3,STK32B,STK4,STOML1,STRADA,STX16,STXBP1,STYX,SUGT1,SULF1,SULT1A1,SUMO1,SUPT6H,SUSD1,SUSD2,SVEP1,SVIL,SYCP1,SYF2,SYNE1,SYNE2,SYNJ1,SYT5,SYTL2,T,TAB3,TAF9,TAF9B,TANC1,TAP1,TAPBP,TAPT1,TAS2R39,TAS2R9,TBC1D2B,TBC1D4,TCEA1,TCEA3,TCEB2,TCF20,TCF7,TCN1,TCOF1,TDGF1,TDRD1,TDRD10,TEP1,TEX10,TEX2,TGFB1,TGFBR1,THBS3,THEMIS,THOC2,TIE1,TIGD5,TLR9,TLX3,TMCO2,TMCO4,TMEM106A,TMEM132E,TMEM161A,TMEM194A,TMEM30A,TMEM51,TMEM54,TMEM62,TMEM8A,TMPRSS7,TNFRSF4,TNFSF12,TNIK,TNKS2,TNNI3K,TNNT3,TNPO1,TNPO3,TOE1,TOMM34,TOMM70A,TOP1,TOPORS,TP53BP1,TP63,TPH1,TPM2,TPO,TPP2,TPST1,TPTE2,TRA2B,TRAIP,TREX2,TRIM26,TRIM37,TRIM41,TRIM55,TRIM63,TRIM7,TRIOBP,TRIP11,TRMT2A,TRMT2B,TRPA1,TRPC4AP,TRPM8,TRUB2,TSGA13,TSPAN2,TSPYL5,TSSC4,TTC13,TTC3,TTC37,TTC7B,TTK,TTLL1,TTLL13,TTLL5,TTLL9,TTN,TUBA8,TUBGCP6,TXNDC16,TXNDC3,TXNRD1,UBAP1,UBE2K,UBE2M,UBE2N,UBE3B,UBE4A,UBE4B,UBR3,UBR4,UBXN2A,UBXN4,UBXN6,UFD1L,UGT1A1,UGT1A10,UGT1A5,UHRF1BP1,ULK2,UMODL1,UNC13B,UNC45B,UPF2,USH1C,USP1,USP19,USP24,USP28,USP30,USP34,USP48,VCAN,VDAC1,VGLL1,VIT,VPS13A,WAPAL,WDR33,WDR47,WDR60,WDR88,WDTC1,WIF1,WNK3,WNT7B,WNT8B,WTAP,XDH,XG,XPNPEP2,XPO5,YTHDC2,YTHDF3,YWHAZ,YY1AP1,ZBTB2,ZBTB32,ZBTB45,ZBTB7C,ZC3H12A,ZC3H12B,ZC3H18,ZC3H7B,ZCCHC2,ZCCHC6,ZDHHC22,ZFHX4,ZFPM2,ZFX,ZFYVE16,ZGPAT,ZIM3,ZMYM2,ZMYND8,ZNF16,ZNF202,ZNF236,ZNF250,ZNF251,ZNF300,ZNF41,ZNF451,ZNF518A,ZNF521,ZNF594,ZNF595,ZNF627,ZNF711,ZNF750,ZNF780A,ZNF782,ZNF91,ZSCAN5A

Code availability

M2 pipeline in snakemake: https://github.com/getzlab/smk-m2

Tool for manual review: https://github.com/getzlab/igv_remote

References

Aldous, David J. 1985. “Exchangeability and Related Topics.” In *École d’Été de Probabilités de Saint-Flour XIII — 1983*, edited by David J. Aldous, Illdar A. Ibragimov, Jean Jacod, and P. L. Hennequin, 1117:1–198. Lecture Notes in Mathematics. Berlin, Heidelberg: Springer Berlin Heidelberg.

Alexandrov, Ludmil B., Jaegil Kim, Nicholas J. Haradhvala, Mi Ni Huang, Alvin Wei Tian Ng, Yang Wu, Arnoud Boot, et al. 2020. “The Repertoire of Mutational Signatures in Human Cancer.” *Nature* 578 (7793): 94–101.

Benjamin, David, Takuto Sato, Kristian Cibulskis, Gad Getz, Chip Stewart, and Lee Lichtenstein. 2019. “Calling Somatic SNVs and Indels with Mutect2.” *Bioinformatics*. bioRxiv.

Benjamini, Yoav, and Yosef Hochberg. 1995. “Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing.” *Journal of the Royal Statistical Society: Series B (Methodological)*. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.

Chowdhury, Biswanath, and Gautam Garai. 2017. “A Review on Multiple Sequence Alignment from the Perspective of Genetic Algorithm.” *Genomics* 109 (5-6): 419–31.

Cibulskis, Kristian, Michael S. Lawrence, Scott L. Carter, Andrey Sivachenko, David Jaffe, Carrie Sougnez, Stacey Gabriel, Matthew Meyerson, Eric S. Lander, and Gad Getz. 2013. “Sensitive Detection of Somatic Point Mutations in Impure and Heterogeneous Cancer Samples.” *Nature Biotechnology* 31 (3): 213–19.

Cibulskis, Kristian, Aaron McKenna, Tim Fennell, Eric Banks, Mark DePristo, and Gad Getz. 2011. “ContEst: Estimating Cross-Contamination of Human Samples in next-Generation Sequencing Data.” *Bioinformatics* 27 (18): 2601–2.

Cosmic. 2020. “COSMIC,” April. https://cancer.sanger.ac.uk/census.

Costello, Maura, Trevor J. Pugh, Timothy J. Fennell, Chip Stewart, Lee Lichtenstein, James C. Meldrim, Jennifer L. Fostel, et al. 2013. “Discovery and Characterization of Artifactual Mutations in Deep Coverage Targeted Capture Sequencing Data due to Oxidative DNA Damage during Sample Preparation.” *Nucleic Acids Research* 41 (6): e67.

Ellrott, Kyle, Matthew H. Bailey, Gordon Saksena, Kyle R. Covington, Cyriac Kandoth, Chip Stewart, Julian Hess, et al. 2018. “Scalable Open Science Approach for Mutation Calling of Tumor Exomes Using Multiple Genomic Pipelines.” *Cell Systems* 6 (3): 271–81.e7.

Ewing, Adam D., Kathleen E. Houlahan, Yin Hu, Kyle Ellrott, Cristian Caloian, Takafumi N. Yamaguchi, J. Christopher Bare, et al. 2015. “Combining Tumor Genome Simulation with Crowdsourcing to

Benchmark Somatic Single-Nucleotide-Variant Detection." *Nature Methods* 12 (7): 623–30.
Garrison, Erik, and Gabor Marth. 2012. "Haplotype-Based Variant Detection from Short-Read Sequencing." http://arxiv.org/abs/1207.3907.
"gatk2019 - gatk2019 - Training - CSC Company Site." n.d. Accessed June 3, 2020. https://www.csc.fi/en/web/training/-/gatk2019.
Groza, Cristian, Tony Kwan, Nicole Soranzo, Tomi Pastinen, and Guillaume Bourque. 2020. "Personalized and Graph Genomes Reveal Missing Signal in Epigenomic Data." *Genome Biology* 21 (1): 1754.
Hess, Julian M., Andre Bernards, Jaegil Kim, Mendy Miller, Amaro Taylor-Weiner, Nicholas J. Haradhvala, Michael S. Lawrence, and Gad Getz. 2019. "Passenger Hotspot Mutations in Cancer." *Cancer Cell* 36 (3): 288–301.e14.
Lawrence, Michael S., Petar Stojanov, Paz Polak, Gregory V. Kryukov, Kristian Cibulskis, Andrey Sivachenko, Scott L. Carter, et al. 2013. "Mutational Heterogeneity in Cancer and the Search for New Cancer-Associated Genes." *Nature* 499 (7457): 214–18.
Li, Heng. 2014. "Toward Better Understanding of Artifacts in Variant Calling from High-Coverage Samples." *Bioinformatics* 30 (20): 2843–51.
Li, Heng, Jonathan M. Bloom, Yossi Farjoun, Mark Fleharty, Laura Gauthier, Benjamin Neale, and Daniel MacArthur. 2018. "A Synthetic-Diploid Benchmark for Accurate Variant-Calling Evaluation." *Nature Methods* 15 (8): 595–97.
Morgulis, Aleksandr, E. Michael Gertz, Alejandro A. Schäffer, and Richa Agarwala. 2006. "A Fast and Symmetric DUST Implementation to Mask Low-Complexity DNA Sequences." *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology* 13 (5): 1028–40.
Munchel, Sarah, Yen Hoang, Yue Zhao, Joseph Cottrell, Brandy Klotzle, Andrew K. Godwin, Devin Koestler, et al. 2015. "Targeted or Whole Genome Sequencing of Formalin Fixed Tissue Samples: Potential Applications in Cancer Genomics." *Oncotarget* 6 (28): 25943–61.
"Mutation Analysis (MutSig 2CV v3.1) - Bladder Urothelial Carcinoma (Primary Solid Tumor)." n.d. Accessed June 2, 2020. http://gdac.broadinstitute.org/runs/analyses__2016_01_28/reports/cancer/BLCA-TP/MutSigNozzleReport2CV/nozzle.html.
Narzisi, Giuseppe, André Corvelo, Kanika Arora, Ewa A. Bergmann, Minita Shah, Rajeeva Musunuri, Anne-Katrin Emde, Nicolas Robine, Vladimir Vacic, and Michael C. Zody. 2018. "Genome-Wide Somatic Variant Calling Using Localized Colored de Bruijn Graphs." *Communications Biology* 1 (March): 20.
Robinson, James T., Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, and Jill P. Mesirov. 2011. "Integrative Genomics Viewer." *Nature Biotechnology* 29 (1): 24–26.
Shifu Chen, Yue Han, Lanting Guo, Jingjing Hu, and Jia Gu. 2016. "SeqMaker: A next Generation Sequencing Simulator with Variations, Sequencing Errors and Amplification Bias Integrated." In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 835–40. IEEE.
Tian, Shulan, Huihuang Yan, Michael Kalmbach, and Susan L. Slager. 2016. "Impact of Post-Alignment Processing in Variant Discovery from Whole Exome Data." *BMC Bioinformatics* 17 (1): 403.
Tokheim, Collin J., Nickolas Papadopoulos, Kenneth W. Kinzler, Bert Vogelstein, and Rachel Karchin. 2016. "Evaluating the Evaluation of Cancer Driver Genes." *Proceedings of the National Academy of Sciences of the United States of America* 113 (50): 14330–35.
Xu, Chang. 2018. "A Review of Somatic Single Nucleotide Variant Calling Algorithms for next-Generation Sequencing Data." *Computational and Structural Biotechnology Journal* 16 (February): 15–24.
Xu, Huilei, John DiCarlo, Ravi Vijaya Satya, Quan Peng, and Yexun Wang. 2014. "Comparison of Somatic Mutation Calling Methods in Amplicon and Whole Exome Sequence Data." *BMC Genomics* 15 (March): 244.

Zook, Justin M., David Catoe, Jennifer McDaniel, Lindsay Vang, Noah Spies, Arend Sidow, Ziming Weng, et al. 2016. “Extensive Sequencing of Seven Human Genomes to Characterize Benchmark Reference Materials.” *Scientific Data*.