Notes on Mutect2

David Benjamin,* Takuto Sato, Lee Lichtenstein, and Megan Shand
Broad Institute, 415 Main Street, Cambridge, MA 02142
(Dated: February 16, 2019)

I. MUTECT2

Here we describe the command-line program `Mutect2` itself, which takes us from aligned reads to unfiltered, annotated variant calls. Code block 3 shows how to invoke `Mutect2` using the gatk launch script.

Listing 1: Mutect2 command

```
gatk Mutect2 -R reference.fasta \
   -L intervals.interval_list \
   -I tumor1.bam \
   # Mutect2 may input more tumor samples from the same individual
   [-I tumor2.bam -I tumor3.bam . . .]  \
   # Mutect2 may input matched normals from the same individual
   [-I normal1.bam -I normal2.bam . . .]  \
   # For most purposes Mutect2 should be supplied with gnomad
   [-germline-resource af-only-gnomad.vcf] \
    # Mutect2 may input a panel of normals to help identify technical artifacts
   [-pon panel_of_normals.vcf . . .]  \
   -O unfiltered.vcf
```

A. Additional modes

The command above encompasses tumor-only, tumor-normal, and multiple-tumor somatic variant calling.

1. *Mitochondria mode*

For mitochondrial calling one should add the `--mitochondria-mode` flag to the `Mutect2` command line. This switches several defaults from values appropriate to somatic variant calling to values that reflect the greater density of mitochondrial mutations. It also activates annotations relevant to alignment artifacts involving nuclear paralogs to mitochondrial DNA.

2. *Force calling*

One can force `Mutect2` to assemble and genotype all the variants in force-calls.vcf by adding `--genotyping-mode GENOTYPE_GIVEN_ALLELES -alleles force-calls.vcf` to the command line. This injects all alleles in force-calls.vcf into the assembly graph, deactivates pruning of subgraphs contained them, and forces `Mutect2` to emit them in the output vcf regardless of their evidence. In order to include even filtered alleles in force-calls.vcf, use the `--genotype-filtered-alleles` flag. Force-called alleles are emitted *in addition* to any alleles that `Mutect2` would otherwise discover. This so-called GGA mode is useful for studying known driver mutations, for monitoring tumors after chemotherapy, and for validating calls from `Mutect2` or other tools against some orthogonal sequencing data.

*Electronic address: `davidben@broadinstitute.org`

B. Finding Active Regions

`Mutect2` triages sites based on their pileup at a single base locus. If there is sufficient evidence of variation `Mutect2` proceeds with local reassembly and realignment. As in the downstream parts of `Mutect2` we seek a likelihood ratio between the existence and non-existence of an alt allele. Instead of obtaining read likelihoods via Pair-HMM, we assign each base a likelihood. For substitutions we can simply use the base quality. For indels we assign a heuristic effective quality that increases with length. Supposing we have an effective quality for each element in the read pileup we can now estimate the likelihoods of no variation and of a true alt allele with allele fraction f. Let $\mathcal{R}$ and $\mathcal{A}$ denote the sets of ref and alt reads. The likelihood of no variation is the likelihood that every alt read was in error. Letting ϵ_i be the error probability of pileup element i we have:

$$L(\text{no variation}) = \prod_{i \in \mathcal{R}} (1 - \epsilon_i) \prod_{j \in \mathcal{A}} \epsilon_j \approx \prod_{j \in \mathcal{A}} \epsilon_j \tag{1}$$

$$L(f) = \prod_{i \in \mathcal{R}} \left[(1-f)(1-\epsilon_i) + f\epsilon_i\right] \prod_{j \in \mathcal{A}} \left[f(1-\epsilon_j) + (1-f)\epsilon_j\right] \approx (1-f)^{N_{\text{ref}}} \prod_{j \in \mathcal{A}} \left[f(1-\epsilon_j) + (1-f)\epsilon_j\right], \tag{2}$$

where the approximations amount to giving ref reads infinite quality, which speeds the computation, and we let $N_{\text{ref}} = |\mathcal{R}|$. This is equivalent to the following model in which we give the nth alt read a latent indicator z_j which equals 1 when the read is an error:

$$P(\text{reads}, f, \mathbf{z}) = (1-f)^{N_{\text{ref}}} \prod_{n=1}^{N_{\text{alt}}} \left[(1-f)\epsilon_n\right]^{z_n} \left[f(1-\epsilon_n)\right]^{1-z_n} \tag{3}$$

We approximate the model evidence $L(f) = \sum_{\mathbf{z}} \int df P(\text{reads}, f, \mathbf{z})$ via a mean field variational Bayes approximation in which we factorize the full data likelihood as $P(\text{reads}, f, \mathbf{z}) \approx q(f)q(\mathbf{z}) = q(f) \prod_n q(z_n)$[1]. For simplicity and speed, we will not iteratively compute $q(f)$. Rather, we use the fact that z_n is almost always 0 to see, by inspection, that

$$q(f) \approx \text{Beta}(f|\alpha, \beta), \quad \alpha = N_{\text{alt}} + 1, \beta = N_{\text{ref}} + 1. \tag{4}$$

Here the "+1"s come from the pseudocounts of a flat prior on f. Then, following the usual recipe of averaging the log likelihood with respect to f and re-exponentiating, we find

$$q(z_n) = \text{Bernoulli}(z_n|\gamma_n), \quad \gamma_n = \frac{\rho\epsilon_n}{\rho\epsilon_n + \tau(1-\epsilon_n)}, \tag{5}$$

where $\ln \rho \equiv E_{q(f)}\left[\ln(1-f)\right] = \psi(\beta) - \psi(\alpha+\beta)$ and $\ln \tau \equiv E_{q(f)}\left[\ln f\right] = \psi(\alpha) - \psi(\alpha+\beta)$. Then, Equation 10.3 of Bishop gives us the variational lower bound on $L(f)$:

$$L(f) \approx E_q\left[\ln P(\text{reads}, f, \mathbf{z})\right] + \text{entropy}[q(f)] + \sum_n \text{entropy}[q(z_n)] \tag{6}$$

$$= H(\alpha, \beta) + N_{\text{ref}} \ln \rho + \sum_n \left[\gamma_n \ln(\rho\epsilon_n) + (1-\gamma_n)\ln(\tau(1-\epsilon_n)) + H(\gamma_n)\right], \tag{7}$$

where $H(\alpha, \beta)$ and $H(\gamma)$ are Beta and Bernoulli entropies. We summarize these steps in the following algorithm:

1: Record the base qualities, hence the error probabilities ϵ_n of each alt read.
2: $\alpha = N_{\text{alt}} + 1$, $\beta = N_{\text{ref}} + 1$
3: $\rho = \exp(\psi(\beta) - \psi(\alpha+\beta))$, $\tau = \exp(\psi(\alpha) - \psi(\alpha+\beta))$.
4: $\gamma_n = \rho\epsilon_n / \left[\rho\epsilon_n + \tau(1-\epsilon_n)\right]$
5: $L(f) \approx H(\alpha, \beta) + N_{\text{ref}} \ln \rho + \sum_n \left[\gamma_n \ln(\rho\epsilon_n) + (1-\gamma_n)\ln(\tau(1-\epsilon_n)) + H(\gamma_n)\right]$

To get the log odds we subtract the log likelihood, $\sum_n \ln \epsilon_n$, from $L(f)$.

[1] The latter step is an induced factorization – once f and $\mathbf{z}$ are decoupled, then the different z_n become independent as well.

C. Local Assembly, Pair-HMM, and Realignment

These topics, which are common to `Mutect2` and `HaplotypeCaller`, are discussed in docs/local_assembly.pdf, docs/pair_hmm.pdf, and docs/variants_from_haplotypes.pdf in the gatk git repository. As a black box, whenever the evidence in the previous section suffices to trigger local assembly and realignment, we end up at each candidate variant site with one read-vs-allele log likelihood matrix ℓ for each sample, where ℓ_{ra} is the log probability of sequencing read r given its base qualities and given that read r is derived from a molecule exhibiting allele a. [2]

D. Somatic Likelihoods Model

We have a set of potential somatic alleles and read-allele likelihoods $\ell_{ra} \equiv P(\text{read } r|\text{allele } a)$. We don't know which alleles are real somatic alleles and so we must compute, for each subset $\mathbb{A}$ of alleles, the likelihood that the reads come from $\mathbb{A}$. A simple model for this likelihood is as follows: each read r is associated with a latent indicator vector $\mathbf{z}_r$ with one-hot encoding $z_{ra} = 1$ iff read r came from allele $a \in \mathbb{A}$. The conditional probabilities of reads given alleles is ℓ_{ra}. There is a latent vector $\mathbf{f}$ of allele fractions such that f_a is the allele fraction of allele a, that is, the prior probability that any given read comes from allele a. Giving $\mathbf{f}$ a Dirichlet prior, we have a full-model likelihood

$$P(\mathbb{R}, \mathbf{z}, \mathbf{f}|\mathbb{A}) = P(\mathbf{f})P(\mathbf{z}|\mathbf{f})P(\mathbb{R}|\mathbf{z}, \mathbb{A}) = \text{Dir}(\mathbf{f}|\boldsymbol{\alpha}) \prod_a \prod_r (f_a \ell_{ra})^{z_{ra}} . \tag{8}$$

We want to marginalize the latent variables to obtain the evidence $P(\mathbb{R}|\mathbb{A})$, which we make tractable via a mean-field approximation $P(\mathbb{R}, \mathbf{z}, \mathbf{f}|\mathbb{A}) \approx q(\mathbf{z})q(\mathbf{f})$, which is exact in two limits. First, if there are many reads, each allele is associated with many reads and therefore the Law of Large Numbers causes $\mathbf{f}$ and $\mathbf{z}$ to become uncorrelated. Second, if the allele assignments of reads are obvious $\mathbf{z}_r$ is effectively determinate, hence uncorrelated with $\mathbf{f}$. In the variational Bayesian mean-field formalism we have

$$q(\mathbf{f}) \propto E_{q(\mathbf{z})} \left[P(\mathbb{R}, \mathbf{z}, \mathbf{f}|\mathbb{A})\right] \propto \text{Dir}(\mathbf{f}|\boldsymbol{\alpha} + \sum_r \bar{\mathbf{z}}_r) \equiv \text{Dir}(\mathbf{f}|\boldsymbol{\beta}), \quad \boldsymbol{\beta} = \boldsymbol{\alpha} + \sum_r \bar{\mathbf{z}}_r \tag{9}$$

$$q(\mathbf{z}_r) \propto E_{q(\mathbf{f})} \left[P(\mathbb{R}, \mathbf{z}, \mathbf{f}|\mathbb{A})\right] \propto \prod_a \left(\tilde{f}_a \ell_{ra}\right)^{z_{ra}} , \tag{10}$$

where, with ψ denoting the digamma function, the moments

$$\bar{z}_{ra} = E_{q(\mathbf{z})} \left[z_{ra}\right] = \frac{\tilde{f}_a \ell_{ra}}{\sum_{a'} \tilde{f}_{a'} \ell_{ra'}} \tag{11}$$

$$\ln \tilde{f}_a = E_{q(\mathbf{f})}[\ln f_a] = \psi(\beta_a) - \psi(\sum_{a'} \beta_{a'}) \tag{12}$$

are easily obtained from the categorical distribution $q(\mathbf{z})$ and the Dirichlet distribution $q(\mathbf{f})$[3] We initialize $\bar{z}_{ra} = 1$ if a is the most likely allele for read r, 0 otherwise and iterate Equations 9 and 10 until convergence. Having obtained the mean fields of $q(\mathbf{z})$ and $q(\mathbf{f})$, we use the variational approximation (Bishop's Eq 10.3) to the model evidence:

$$\ln P(\mathbb{R}|\mathbb{A}) \approx E_q \left[\ln P(\mathbb{R}, \mathbf{z}, \mathbf{f}|\mathbb{A})\right] - E_q \left[\ln q(\mathbf{z})\right] - E_q \left[\ln q(\mathbf{f})\right] . \tag{13}$$

The terms in Eq 13 all involve the standard moments mentioned above, so after a bit of algebraic cancellation we obtain

$$\ln P(\mathbb{R}|\mathbb{A}) \approx g(\boldsymbol{\alpha}) - g(\boldsymbol{\beta}) + \sum_{ra} \bar{z}_{ra} \left(\ln \ell_{ra} - \ln \bar{z}_{ra}\right) , \tag{14}$$

where we define g to be the Dirichlet distribution log normalization:

$$\ln \Gamma(\sum_a \omega_a) - \sum_a \ln \Gamma(\omega_a). \tag{15}$$

---

[2] Technically, pair-HMM produces a read-vs-*haplotype* likelihood matrix, which is then "marginalized" to produce a set of read-vs-allele likelihood matrices. In the future, `Mutect2` may operate directly on this read-vs-haplotype matrix in order to exploit the biological fact that there are only a few haplotypes in any region.

[3] Note that we didn't *impose* this in any way. It simply falls out of the mean field equations.

We now have the model evidence for allele subset $\mathbb{A}$. The `TLOD` emitted by `Mutect2` for an alt allele is the log evidence ratio of an allele set containing all alleles versus an allele set excluding that allele. That is, it is the log odds that an allele exists. When multiple tumor samples are given, `Mutect2` computes a single `TLOD` by combining all tumor reads.

II. FILTERING

The command line tool `FilterMutectCalls` inputs the unfiltered output of `Mutect2` and emits another vcf containing the same variants, annotated with the filters that they fail, if any.

Listing 2: Mutect2 command

```
gatk FilterMutectCalls -V unfiltered.vcf \
   # FilterMutectCalls may input segmentation for one or more tumor samples from Calculat
   [--tumor-segmentation segments1.table] \
   [--tumor-segmentation segments2.table] \
    # FilterMutectCalls may input contamination estimates for one or more tumor samples f
   [--contamination-table contamination1.table] \
   [--contamination-table contamination2.table] \
   -O filtered.vcf
```

The optional inputs from the GATK 4 tool `CalculateContamination` are described below.

A. Filtering Architecture

`FilterMutectCalls` contains a set of filters, each of which computes an error probability for each candidate variant. The filters are divided into three categories: technical artifacts, non-somatic, and sequencing error. Roughly, we assume that different types of errors within a category are correlated, while different categories are independent. For example, whether sequencing errors cause several bases to be misread is independent of whether the DNA being read came from a contaminating sample and of whether an error during library preparation caused a base error prior to sequencing. To obtain an overall error probability `FilterMutectCalls` computes the maximum within categories and an independent product between categories. That is:

$$P(\text{error}) = 1 - (1 - \max \text{ artifact error prob})(1 - \max \text{ non} - \text{somatic prob})(1 - \text{sequencing error prob}). \quad (16)$$

`FilterMutectCalls` goes over an unfiltered vcf in three passes, two to learn any unknown parameters of the filters' models and to set a threshold on $P(\text{error})$, and one to apply the learned filters. This section describes methods for determining the threshold on error probability. One option is to simply set a fixed value p on the error probability, below which a call is considered a real somatic variant. This non-default behavior can be set via `--threshold-strategy CONSTANT --initial-threshold <double>`.

1. *F Score Thresholding*

`FilterMutectCalls` optimizes the F-score – the harmonic mean of recall and precision – as its default thresholding strategy. The `--f-score-beta <double>` command line argument can be set to change the relative weight of recall to precision. In order to optimize the F-score, `FilterMutectCalls` sorts all variants by the probability that they are errors, from least to greatest and calculated the F-score for thresholds in which the first n variants pass, starting from $n = 0$ and ending at $n = N$, the total number of candidates. This is a cheap $O(N)$ computation because initially the expected number of true positive and false positive calls are both zero. When the threshold increases to admit a variant with error probability p, the expected number of true positive calls increases by $1-p$ and the expected number of false positive calls increases by p. The total expected number of real variants is $\sum_n (1 - p_n)$. These quantities suffice to calculate recall and precision for every threshold.

2. *False Discovery Rate Thresholding*

`FilterMutectCalls` can also choose to maximize sensitivity subject to a maximum allowable false discovery rate using the `--threshold-strategy FALSE_DISCOVERY_RATE --false-discovery-rate <double>`. For this calculation

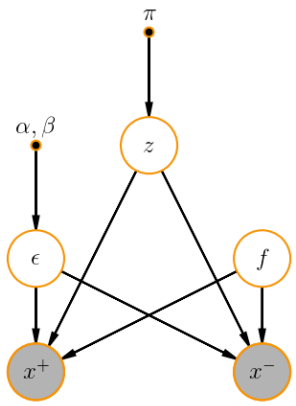

FIG. 1: The probabilistic graphical model for the strand artifact model

`FilterMutectCalls` also sorts the error probabilities p_n from least to greatest. If we allow the first M variants to pass the expected false discovery rate is

$$\frac{1}{M}\sum_{n=1}^{M} p_n \tag{17}$$

This is a non-decreasing function of M because it is the integral from 0 to M of the monotonic function p, hence its derivative with respect to M is p_M, which is monotonic. Thus it is easy to choose the highest M such that the maximum false discovery rate is not exceeded.

B. Hard Filters

Several filters are hard filters that assign an error probability of 1 whenever some annotation exceeds a threshold. Here we summarize all the hard filters of `FilterMutectCalls`, the command line parameters that set their thresholds, and an explanation of the thresholded quantity.

Filter	Threshold	Explanation
`clustered_events`	`max-events-in-region`	mutations sharing an assembly region
`duplicate_evidence`	`unique-alt-read-count`	unique insert start/end pairs of alt reads
`multiallelic`	`max-alt-alleles-count`	passing alt alleles at a site
`base_quality`	`min-median-base-quality`	median base quality of alt reads
`mapping_quality`	`min-median-mapping-quality`	median mapping quality of alt reads
`fragment_length`	`max-median-fragment-length-difference`	difference of alt and ref reads' median fragment lengths
`read_position`	`min-median-read-position`	median distance of alt mutations from end of read
`panel_of_normals`	`panel-of-normals`	presence in panel of normals

C. Strand Artifact Model

The strand artifact filter detects sequencing artifacts in which the evidence for the alt allele consists entirely of forward strand reads alone or reverse strand reads alone. We must detect this while taking into account the fact that at some loci, such as near the end of an exome target, *all* reads are biased towards one direction, and therefore a bias towards a particular strand among alt reads is no cause for alarm.

Let $z \in \{z_+, z_-, z_o\}$ be a latent random variable with 1-hot encoding that represents the artifact state of a suspected variant i.e. $z_+ = 1$ for a forward strand artifact, $z_- = 1$ for a reverse artifact, and $z_o = 1$ otherwise. At each locus, let $a_\pm$ be the number of forward (+) or reverse (-) strand alt reads and let $n_\pm$ be the total depth for each strand. By modeling $a+_\pm$ relative to $n_\pm$ we account for inherent strand bias due to, for example, reads falling at the end of an exome target and do not confuse it for an artifact. Let f be the allele fraction of true variation in case $z_o = 1$. Let ϵ be the strand bias error rate and let θ be the non-strand-biased error rate. We will ignore the case in which significant strand bias coincides with real variation, first because this is exceedingly rare and ignoring it has a negligible effect on the parameters of our model, and secondly because such variants should be considered true positives.

The conditional distributions of our model are binomial

$$a_+|\epsilon, \theta, f, z_+ = 1 \sim \text{Bin}(a_+|n_+, \epsilon) \tag{18}$$
$$a_+|\epsilon, \theta, f, z_- = 1 \sim \text{Bin}(a_+|n_+, \theta) \tag{19}$$
$$a_+|\epsilon, \theta, f, z_o = 1 \sim \text{Bin}(a_+|n_+, f), \tag{20}$$

and similarly for a_-. Putting beta priors on ϵ, θ, and f, with parameters $(\alpha_\epsilon, \beta_\epsilon)$, $(\alpha_\theta, \beta_\theta)$, and (α_f, β_f) and marginalizing latent parameters we obtain likelihoods

$$P(a_+, a_-|z_\pm = 1) = \text{BetaBinom}(a_\pm|n_\pm, \alpha_\epsilon, \beta_\epsilon)\text{BetaBinom}(a_\mp|n_\mp, \alpha_\theta, \beta_\theta) \tag{21}$$
$$P(a_+, a_-|z_o = 1) = \int_0^1 \text{Beta}(f|\alpha_f, \beta_f)\text{Binom}(a_+|n_+, f)\text{Binom}(a_-|n_-, f) \tag{22}$$
$$= \frac{\binom{n_+}{a_+}\binom{n_-}{a_-}}{\binom{n_+ + n_-}{a_+ + a_-}}\text{BetaBinom}(a_+ + a_-|n_+ + n_-, \alpha_f, \beta_f) \tag{23}$$

Finally, we let $\pi/2$ be the prior probability of a forward or reverse strand artifact. From the above equations it is straightforward to calculate the posterior probability of z and to learn π iteratively via the EM algorithm. It is somewhat more complicated to learn $(\alpha_\epsilon, \beta_\epsilon)$ and $(\alpha_\theta, \beta_\theta)$, so we treat these as fixed hyperparameters. We use a flat prior $\alpha_f = \beta_f = 1$ for the true allele fraction.

D. Germline Filter

Suppose we have detected an allele such that its (somatic) likelihood in the tumor is ℓ_t and its (diploid) likelihood in the normal is ℓ_n[4]. By convention, both of these are relative to a likelihood of 1 for the allele *not* to be found. If we have no matched normal, $\ell_n = 1$. Suppose we also have the population allele frequency f of this allele. Then the prior probabilities for the normal to be heterozygous and homozygous alt for the allele are $2f(1-f)$ and f^2 and the prior probability for the normal genotype not to contain the allele is $(1-f)^2$. Finally, suppose that the prior for this allele to arise as a somatic variant is π.

We can determine the posterior probability that the variant exists in the normal genotype by calculating the unnormalized probabilities of four possibilities:

1. The variant exists in the tumor and the normal as a germline het. This has unnormalized probability $2f(1-f)\ell_n\ell_t(1-\pi)$.
2. The variant exists in the tumor and the normal as a germline hom alt. This has unnormalized probability $f^2\ell_n\ell_t(1-\pi)$.
3. The variant exists in the tumor but not the normal. This has unnormalized probability $(1-f)^2\ell_t\pi$.

We exclude possibilities in which the variant does not exist in the tumor sample because we really want the conditional probability that the variant is germline given that it would otherwise be called.

Normalizing, we obtain the following posterior probability that an allele is a germline variant:

$$P(\text{germline}) = \frac{(1)+(2)}{(1)+(2)+(3)} = \frac{\left(2f(1-f)+f^2\right)\ell_n\ell_t(1-\pi)}{\left(2f(1-f)+f^2\right)\ell_n\ell_t(1-\pi) + \ell_t(1-f)^2\pi}. \tag{24}$$

[4] This is the total likelihood for het and hom alt in the normal.

The above equation, in which the factors of ℓ_t could cancel if we wished, is not quite right. The tumor likelihood ℓ_t is the probability of the tumor data given that the allele exists in the tumor *as a somatic variant*. If the allele is in the tumor as a germline het we must modify ℓ_t to account for the fact that the allele fraction is determined by the ploidy – it must be either f_g or $1 - f_g$with equal probability, where f_g is the minor allele fraction of germline hets. It would be awkward to recalculate the tumor likelihood with the allele frequency constrained to these two values[5], but we can estimate a correction factor as follows: assuming that the posterior on the allele fraction in the somatic likelihoods model is fairly tight, the likelihood of a alt reads out of n total reads is $\binom{n}{a}(1-f_t)^{n-a}f^a$, where f_t is the tumor alt allele fraction. That is, our sophisticated model that marginalizes over f_t reduces to something more naive. If the variant is a germline event, the likelihood becomes $\frac{1}{2}\binom{n}{a}\left[(1-f_g)^{n-a}f_g^a + f_g^{n-a}(1-f_g)^a\right]$. Thus, in case (1) we have $\ell_t \rightarrow \chi\ell_t$, where

$$\chi = \frac{1}{2}\frac{(1-f_g)^{n-a}f_g^a + f_g^{n-a}(1-f_g)^a}{(1-f_t)^{n-a}f_t^a}. \tag{25}$$

For germline hom alts, both the tumor and normal allele fractions will be similarly large, so to decent approximation we don't have to modify ℓ_t. Of course, this only applies if the allele fraction is large. Rather than try to model the count of ref reads within a germline hom alt site, we simply set a threshold of allele fraction 0.9, so that in case (2) $\ell_t \rightarrow \mathrm{I}[f_t > 0.9]\ell_t$. and the corrected germline probability is

$$P(\text{germline}) = \frac{(1)+(2)}{(1)+(2)+(3)} = \frac{\left(2f(1-f)\chi + \mathrm{I}[f_t > 0.9]f^2\right)\ell_n(1-\pi)}{\left(2f(1-f)\chi + \mathrm{I}[f_t > 0.9]f^2\right)\ell_n(1-\pi) + (1-f)^2\pi}. \tag{26}$$

To filter, we set a threshold on this posterior probability.

So far we have assumed that the population allele frequency f is known, which is the case if it is found in our germline resource, such as gnomAD. If f is not known we must make a reasonable guess as follows. Suppose the prior distribution on f is $\mathrm{Beta}(\alpha, \beta)$. The mean $\alpha/(\alpha+\beta)$ of this prior is the average human heterozygosity $\theta \approx 10^{-3}$, so we have $\beta \approx \alpha/\theta$. We need one more constraint to determine α and β, and since we are concerned with imputing f when f is small we use a condition based on rare variants. Specifically, the number of variant alleles n at some site in a germline resource with $N/2$ samples, hence N chromosomes, is given by $f \sim \mathrm{Beta}(\alpha, \beta), n \sim \mathrm{Binom}(N, f)$. That is, $n \sim \mathrm{BetaBinom}(\alpha, \beta, N)$. The probability of a site being non-variant in every sample is then $P(n = 0) = \mathrm{BetaBinom}(0|\alpha, \beta, N)$, which we equate to the empirical proportion of non-variant sites in our resource, about 7/8 for exonic sites in gnomAD. Solving, we obtain approximately $\alpha = 0.01, \beta = 10$ for gnomAD. Now, given that some allele found by `Mutect2` is not in the resource, the posterior on f is $\mathrm{Beta}(\alpha, \beta + N)$, the mean of which is, since $\beta << N$, about α/N. By default, `Mutect2` uses this value.

E. Contamination Filter

To calculate the cross-sample calculation of a tumor sample, one runs the GATK tools `GetPileupSummaries` and `CalculateContamination` as follows:

Listing 3: Mutect2 command

```
gatk GetPileupSummaries -I tumor.bam \
   -V common-biallelic.vcf \
   -L common-biallelic.vcf \
   -O tumor.pileups

# if a normal is present, it is helpful to obtain its pileup sumaries
gatk GetPileupSummaries -I normal.bam \
   -V common-biallelic.vcf \
   -L common-biallelic.vcf \
   -O tumor.pileups

gatk CalculateContamination -I tumor.pileups \
   # the normal pileups are useful but optional
```

[5] The model could easily accommodate this change, but the likelihoods are long gone from memory once the germline computation occurs.

```
[-matched normal.pileups] \
-O contamination.table \
# it is highly recommended to produce segments for FilterMutectCalls
[-tumor-segmentation segments.table]
```

Suppose our tumor bam has contamination fraction α and that at some site we have a alt reads out of d total reads. Suppose further that the alt allele has population allele frequency f. We will compute a simple estimate of the posterior probability that these alt reads came from a contaminating sample and not from a true somatic variant. Let π be the prior probability of somatic variation as above. Our crude model for the alt count distribution of somatic variation is a uniform distribution. That is, we assume that any value of a from 0 to d is equally likely. Then the likelihood of the data given a true somatic variant is

$$P(a|\text{somatic}) = \frac{1}{d+1}. \tag{27}$$

We consider two models of contamination. If there are multiple contaminants we approximate each contaminant read as independent. Then the probability of any given read being an alt contaminant read is αf, so we have

$$P(a|\text{many contaminant}) = \text{Binom}(a|d, \alpha f). \tag{28}$$

If there is a single contaminating sample it is heterozygous with probability $2f(1-f)$ and homozygous for the alt with probability f^2, in which cases fractions $\alpha/2$ and α of all reads to be alt contaminants. The contaminant is homozygous for the ref with probability $(1-f)^2$, which yields no alt reads. Thus

$$P(a|\text{one contaminant}) = 2f(1-f)\text{Binom}(a|d, \alpha/2) + f^2\text{Binom}(a|d, \alpha) + (1-f)^2\text{I}[a=0]. \tag{29}$$

We take the likelihood $P(a|\text{contamination})$ to be the maximum of these, which admittedly is not quite rigorous. Usually one will be overwhelmingly larger than the other, however, so it's a decent approximation. Our posterior probability of contamination is then

$$P(\text{contamination}|a) = \frac{P(a, \text{contamination})}{P(a, \text{contamination}) + P(a, \text{somatic})} = \frac{(1-\pi)P(a|\text{contamination})}{(1-\pi)P(a|\text{contamination}) + \pi P(a|\text{somatic})} \tag{30}$$

We filter by setting a threshold on this posterior probability.

III. READ ORIENTATION ARTIFACT FILTER

The read orientation artifact, also known as the orientation bias artifact, arises due to a chemical change in the nucleotide during library prep that results in, for example, G base-paring with A. This kind of artifact has a clear signature (e.g. C to A SNP that occurs predominantly for the middle C in the DNA sequence CCG), and it's single-stranded in nature. Downstream, this artifact manifests as low allele fraction SNPs whose evidence for the alt allele consists almost entirely F1R2 reads or F2R1 reads. A read pair is F1R2 (forward 1st, reverse 2nd) if the sequence of bases in Read 1 maps to the forward strand of the reference (F1), and the sequence of Read 2 to the reverse strand of the reference (R2). F2R1 is defined similarly.

Without loss of generality, suppose that the reference context at locus i is ACT. Let $\mathbf{z}_i$ denote the genotype at locus i with the one-hot encoding $z_{ik} = 1$ iff the genotype of locus i is k, where the possible genotypes are

$$z_i \in \{\text{F1R2}_A, \text{F1R2}_G, \text{F1R2}_T, \text{F2R1}_A, \text{F2R1}_G, \text{F2R1}_T, \text{Hom Ref}, \text{Germline Het}, \text{Somatic Het}, \text{Hom Var}\}$$

$z_i = \text{F1R2}_\text{A}$ denotes that at locus i we have an artifact in which the evidence for alt allele A consists entirely of reads in the F1R2 orientation. The remaining artifact states are defined analogously. Let π denote the prior probabilities of the $\mathbf{z}_i$ under the reference context ACT. Then we have

$$P(\mathbf{z}_i) = \prod_k \pi_k^{z_{ik}} \tag{31}$$

The number of alt reads at a locus depends on the genotype z_i. Let n_i and m_i denote the total depth and alt depth at locus i, respectively. The conditional distribution of m_i is

$$P(m_i|z_{ik} = 1) = \text{BetaBinomial}(m_i|n_i, \alpha_k, \beta_k) \tag{32}$$

where α_k and β_k are fixed hyperparameters for genotype z_k. When the site's genotype indicates in m_i alt reads we expect a heavily skewed distribution of F1R2 reads. This is captured in the conditional distribution of F1R2 alt reads. Let c_i denote the number of F1R2 reads among the m_i alt reads at locus i. Then we have

$$P(c_i|m_i, z_{ik} = 1) = \text{BetaBinomial}(c_i|m_i, \alpha'_k, \beta'_k) \tag{33}$$

We learn the prior artifact probabilities π based on the observed values of n_i, m_i, c_i for each of N loci using the EM algorithm. In the E-step, we compute the posterior probabilities of $\mathbf{z}_i$ for $i = 1...N$. The joint probabilities of $\mathbf{z}$ factorizes over i, thus the posteriors over $\mathbf{z}$ are independent across loci.

$$P(z_{ik} = 1|m_i, c_i) \propto P(z_{ik} = 1, m_i, c_i) = \pi_k \text{BetaBinomial}(m_i|n_i, \alpha_k, \beta_k)\text{BetaBinomial}(c_i|m_i, \alpha'_k, \beta'_k) \tag{34}$$

In the M-step we maximize the expectation of the log complete-data likelihood with respect to π. The log complete data likelihood is given as

$$\ln P(\mathbf{z}, \mathbf{m}, \mathbf{c}) = \sum_i \sum_k z_{ik} \big(\ln \pi_k + \ln \text{BetaBinomial}(m_i|n_i, \alpha_k, \beta_k) + \ln \text{BetaBinomial}(c_i|m_i, \alpha'_k, \beta'_k) \big) \tag{35}$$

Maximizing the log likelihood under the constraint $\sum_k \pi_k = 1$ gives us

$$\pi_k = \frac{N_k}{N} \tag{36}$$

where $N_k = \sum_i P(z_{ik}|m_i, c_i)$ is the effective count of loci with genotype k. We alternate E-step and M-step until convergence. We then use the learned prior genotype probabilities to compute the posterior artifact probabilities of variants in a vcf. The filtering threshold is set such that the false discovery rate doesn't exceed a specified value, as described below.

IV. RELATED GATK TOOLS

The Broad somatics SNVs and indels pipeline involves several GATK tools besides `Mutect2` and `FilterMutectCalls`. We describe them here.

A. Calculating Contamination

Below, we present the GATK's fast, simple, and accurate method for calculating the contamination of a sample. This methods does not require a matched normal, makes no assumptions about the number of contaminating samples, and remains accurate even when the sample has a lot of copy number variation.

The inputs to our tool are a bam file and a vcf of common variants – for example ExAC, gnomAD, or 1000 Genomes – with their allele frequencies. The basic idea, which comes from ContEst[6] by Kristian Cibulskis and others in the Broad Institute Cancer Genome Analysis group, is simply to count ref reads at hom alt sites and subtract the number of ref reads expected from sequencing error to obtain the number of ref reads contaminating these hom alt sites. Finally, we use the allele frequencies to account for the fact that some contaminating reads have the alt allele. The only subtlety is in distinguishing hom alt sites from loss of heterozygosity events, which we describe below.

Suppose we have a set $\mathbb{H}$ of SNPs at which our sample is homozygous for the alternate allele. Let N_{ref} be the total number of ref reads at these sites. We can decompose N_{ref} as follows:

$$N_{\text{ref}} = N_{\text{ref}}^{\text{error}} + N_{\text{ref}}^{\text{contamination}}, \tag{37}$$

where $N_{\text{ref}}^{\text{error}}$ and $N_{\text{ref}}^{\text{contamination}}$ are as the number of ref reads due to error and contamination, respectively. We can obtain N_{ref} by counting reads, and we estimate $N_{\text{ref}}^{\text{error}}$ as follows. Suppose, WLOG, that the ref allele is A and the alt is C. Then, assuming that all substitution errors are equally likely, $N_{\text{ref}}^{\text{error}}$ is approximately half the number of Gs and Ts. This is, of course, not a perfect assumption for any one site, but on average over all the sites in $\mathbb{H}$ it is very good.

---

[6] ContEst: estimating cross-contamination of human samples in next-generation sequencing data, *Bioinformatics* **27**, 2601 (2011)

Next we take the expectation of both sides of Equation 37 to obtain

$$\langle N_{\text{ref}} - N_{\text{ref}}^{\text{error}} \rangle = \left\langle \sum_{s \in \mathbb{H}} \text{number of contaminant ref reads at } s \right\rangle \tag{38}$$

$$= \sum_{s \in \mathbb{H}} \langle \text{number of contaminant ref reads at } s \rangle \tag{39}$$

$$= \sum_{s \in \mathbb{H}} \langle \text{number of contaminant reads at } s \times \text{ref fraction of contaminant reads at } s \rangle \tag{40}$$

$$= \sum_{s \in \mathbb{H}} \langle \text{number of contaminant reads at } s \rangle \times \langle \text{ref fraction of contaminant reads at } s \rangle \tag{41}$$

where we have used the linearity of the expectation and the independence of the total number of contaminant reads with the fraction of contaminant reads that are ref. The expectation of the total number of contaminant reads is the depth d_s at site s times the contamination, which we denote by χ. The expected fraction of contaminant reads that are ref is one minus the alt allele frequency f_s. Crucially, this fact is independent of how many contaminating samples there are. Thus we have

$$\langle N_{\text{ref}} - N_{\text{ref}}^{\text{error}} \rangle = \chi \sum_{s \in \mathbb{H}} d_s (1 - f_s) \tag{42}$$

and obtain the estimate

$$\hat{\chi} \approx \frac{N_{\text{ref}} - N_{\text{ref}}^{\text{error}}}{\sum_{s \in \mathbb{H}} d_s (1 - f_s)} \tag{43}$$

Let us now roughly estimate the error bars on this result. The main source of randomness is the stochasticity in the number of contaminating ref reads. Although the nature of this randomness depends on the number of contaminants, the most variable case, hence an upper bound, is that of a single haploid contaminant, since at each site the only possibilities are the extremes of all contaminant reads being ref or all being alt. In this case, the contribution to the numerator of Eq. 43 from site s is the random variable $X_s Z_s$, where $X_s \sim \text{Binom}(\text{d}_\text{s}, \chi)$ is the number of contaminant reads at s and Z_s is a binary indicator for whether the contaminant reads are ref, with $P(Z_s = 1) = 1 - f_s$. X and Z are independent, so we can work out the variance of XZ as:

$$\text{var}(XZ) = E[X^2 Z^2] - E[XZ]^2 \tag{44}$$

$$= (1 - f_s) E[X^2] - (1 - f_s)^2 E[X]^2 \tag{45}$$

$$= (1 - f_s) \left(\text{var}(X) + E[X]^2 \right) - (1 - f_s)^2 E[X]^2 \tag{46}$$

$$= (1 - f_s) d_s \chi (1 - \chi) + f_s (1 - f_s) d_s^2 \chi^2 \tag{47}$$

And therefore the standard error on $\hat{\chi}$ comes out to the square root of the sum of these per-site variances, divided by the denominator of Eq. 43, that is,

$$\text{std}(\hat{\chi}) = \frac{\sqrt{\sum_s \left[(1 - f_s) d_s \hat{\chi} (1 - \hat{\chi}) + f_s (1 - f_s) d_s^2 \hat{\chi}^2 \right]}}{\sum_s d_s (1 - f_s)} \tag{48}$$

It remains to describe how we determine which sites are hom alt. The fundamental challenge here is that in cancer samples loss of heterozygosity may cause het sites to look like hom alt sites. Our strategy is to partition the genome into allelic copy-number segments, then infer the minor allele fraction of those segments. We segment the genome just as in GATK CNV, using a kernel segmenter with a Gaussian kernel computed on the alt fraction. A nonlinear kernel is important because each segment is multimodal, with peaks for hom ref, alt minor het, alt major het, and hom alt.

We then perform maximum likelihood estimation MLE on a model with learned parameters μ, the local minor allele fraction for each segment, χ, the contamination, and a constant base error rate parameter ϵ determined by counting reads that are neither the ref nor primary alt base at biallelic SNPs as described above. The model likelihood is

$$P(\{a\} | \{f\}, \chi, \{d\}) = \prod_{\text{segments}} \prod_{n \text{ sites}} \sum_{s \text{ genotype } g} P(g | f_s) \text{Binom}(a_s | d_s, (1 - \chi) \phi(g, \mu_n, \epsilon) + \chi f_s) \tag{49}$$

where a_s and d_s are the alt and total read counts at site s, allelic CNV genotypes g run over hom ref, alt minor, alt major, and hom alt with priors $P(\text{hom ref}) = (1 - f_s)^2$, $P(\text{alt minor}) = P(\text{alt major}) = f_s(1 - f_s)$, and $P(\text{homalt} = f^2$.

$\phi(g, \mu, \epsilon)$ is the alt allele fraction of the uncontaminated sample: $\phi(\text{hom ref}) = \epsilon$, $\phi(\text{alt minor}) = \mu$, $\phi(\text{alt major}) = 1-\mu$, $\phi(\text{hom alt}) = 1-\epsilon$. The binomial is the weighted average of the uncontaminated sample and sample reads drawn independently from allele frequency f. This is inconsistent with a single diploid contaminant sample, or indeed with any finite number of contaminants, which is why we do not the the MLE estimate in the final output of the tool. The model also assumes that the uncontaminated and contaminating samples have the same overall depth distribution at each site, which is inconsistent with any differences in copy-number. We perform the MLE by brute force, alternately maximizing with respect to χ with μ fixed and vice versa. In order to make the solution more robust, we exclude segments with low μ from the maximization over χ by taking the highest possible threshold (up to 0.5, of course) for μ that retains at least 1/4 of all sites.

Once we have learned the parameters of this model, we can easily infer the posterior probabilities of hom alt genotypes. We take every site with a posterior probability greater than 0.5. In order to make the result more reliable against CNVs, we again impose a threshold on segment minor allele fraction and apply the above formula only to hom alt sites in these segments. This time, however, we choose the highest possible threshold such that the estimated relative error is less than 0.2.

Finally, we note that the same calculation can be reversed by using alt reads in hom ref sites as the signal and replacing f by $1 - f$ everywhere above. We use the estimate from hom refs as a backup when the hom alt estimate has too great an error, as can occur in the case of targeted panels with few sites. We do not use this as our primary estimate because it is much more affected by uncertainty in the population allele frequencies and is thus susceptible to systematic bias.

B. Proposed tumor in normal estimation tool

Note: the following notes are just a proposal for which no GATK tool yet exists. A popular tool is DeTiN[7] by Amaro Taylor-Weiner and others at the Broad Institute Cancer Genome Analysis group.

Similar to the spirit of CalculateContamination, the fraction of tumor reads in the normal bam is a single number with a large amount of evidence and is probably well-estimated by simple descriptive statistics rather than a full-fledged probabilistic model. It shouldn't be much more complicated than finding somatic variants and comparing their signal in the normal sample to that in the tumor.

We propose the following steps to obtain our input of confident somatic SNVs:

- Run `Mutect2` with the `--genotype-germline-sites` argument to obtain a preliminary list of somatic SNVs, including those that look like germline variants due to tumor-in-normal contamination. For the sake of speed, we could implement a pileup-based mode in which we skip reassembly and equate read likelihoods with base qualities. This would allow us to obtain variant annotations using the existing architecture of `Mutect2` and therefore to filter calls with no new code.

- Run `FilterMutectCalls`. With default settings this eliminates the great majority of sequencing artifacts. To eliminate even more we could increase the log odds threshold slightly, essentially requiring a slightly larger alt allele count. Normally we don't do this because it sacrifices some sensitivity, but for our purposes here a 10 or 20 percent loss of sensitivity is perfectly acceptable as long as we are left with enough SNVs for our estimate. It would also make sense to be especially stringent with variants that have any significant population allele frequency in gnomAD, as well as possible mapping errors. Thus we might also run FilterAlignmentArtifacts with strict parameters.

- Reconsider all variants that are filtered only by the germline and/or normal artifact filters. Those that have enough read counts in the normal that we conclude they are germline variants, as opposed to tumor in normal contamination, should remain filtered.

The above steps are all very reliable, so at this point we can assume we have a collection of confident biallelic somatic SNVs that are hom ref in the germline. Similar to CalculateContamination, we can now estimate the number of alt reads in the normal at these sites:

$$\text{alt in normal} \approx \sum_{\text{sites}} (\text{depth in normal}) \times (\text{alt fraction in tumor}) \times (\text{tumor in normal fraction}) \tag{50}$$

---

[7] DeTiN: overcoming tumor-in-normal contamination, *Nature Methods* **15**, 531 (2018)

Hence we estimate

$$\text{tumor in normal fraction} \approx \frac{\text{total number of alt reads in normal at somatic SNV sites}}{\sum_{\text{somatic SNV sites}}(\text{depth in normal}) \times (\text{alt fraction in tumor})} \tag{51}$$

We could also iterate this process as in CalculateContamination: first get an initial guess of the tumor-in-normal fraction, then use the initial guess to improve our "un-filtering" in the third step above.