

HaplotypeCaller Follow

 gatk.broadinstitute.org/hc/en-us/articles/360040096812-HaplotypeCaller



GATK Team

- March 04, 2020 11:56
- Updated

Call germline SNPs and indels via local re-assembly of haplotypes

Category Short Variant Discovery

Overview

Call germline SNPs and indels via local re-assembly of haplotypes

The HaplotypeCaller is capable of calling SNPs and indels simultaneously via local de-novo assembly of haplotypes in an active region. In other words, whenever the program encounters a region showing signs of variation, it discards the existing mapping information and completely reassembles the reads in that region. This allows the HaplotypeCaller to be more accurate when calling regions that are traditionally difficult to call, for example when they contain different types of variants close to each other. It also makes the HaplotypeCaller much better at calling indels than position-based callers like UnifiedGenotyper.

In the GVCF workflow used for scalable variant calling in DNA sequence data, HaplotypeCaller runs per-sample to generate an intermediate GVCF (not to be used in final analysis), which can then be used in GenotypeGVCFs for joint genotyping of multiple samples in a very efficient way. The GVCF workflow enables rapid incremental processing of samples as they roll off the sequencer, as well as scaling to very large cohort sizes (e.g. the 92K exomes of ExAC).

In addition, HaplotypeCaller is able to handle non-diploid organisms as well as pooled experiment data. Note however that the algorithms used to calculate variant likelihoods is not well suited to extreme allele frequencies (relative to ploidy) so its use is not

recommended for somatic (cancer) variant discovery. For that purpose, use Mutect2 instead.

Finally, HaplotypeCaller is also able to correctly handle the splice junctions that make RNAseq a challenge for most variant callers, on the condition that the input read data has previously been processed according to our recommendations as documented [here](#).

How HaplotypeCaller works

1. Define active regions

The program determines which regions of the genome it needs to operate on (active regions), based on the presence of evidence for variation.

2. Determine haplotypes by assembly of the active region

For each active region, the program builds a De Bruijn-like graph to reassemble the active region and identifies what are the possible haplotypes present in the data. The program then realigns each haplotype against the reference haplotype using the Smith-Waterman algorithm in order to identify potentially variant sites.

3. Determine likelihoods of the haplotypes given the read data

For each active region, the program performs a pairwise alignment of each read against each haplotype using the PairHMM algorithm. This produces a matrix of likelihoods of haplotypes given the read data. These likelihoods are then marginalized to obtain the likelihoods of alleles for each potentially variant site given the read data.

4. Assign sample genotypes

For each potentially variant site, the program applies Bayes' rule, using the likelihoods of alleles given the read data to calculate the likelihoods of each genotype per sample given the read data observed for that sample. The most likely genotype is then assigned to the sample.

Input

Input bam file(s) from which to make variant calls

Output

Either a VCF or GVCF file with raw, unfiltered SNP and indel calls. Regular VCFs must be filtered either by variant recalibration (Best Practice) or hard-filtering before use in downstream analyses. If using the GVCF workflow, the output is a GVCF file that must first be run through GenotypeGVCFs and then filtering before further analysis.

Usage examples

These are example commands that show how to run HaplotypeCaller for typical use cases. Have a look at the [method documentation](#) for the basic GVCF workflow.

Single-sample GVCF calling (outputs intermediate GVCF)

```
gatk --java-options "-Xmx4g" HaplotypeCaller \
  -R Homo_sapiens_assembly38.fasta \
  -I input.bam \
  -O output.g.vcf.gz \
  -ERC GVCF
```

Single-sample GVCF calling with [allele-specific annotations](#)

```
gatk --java-options "-Xmx4g" HaplotypeCaller \
  -R Homo_sapiens_assembly38.fasta \
  -I input.bam \
  -O output.g.vcf.gz \
  -ERC GVCF \
  -G Standard \
  -G AS_Standard
```

Variant calling with [bamout](#) to show realigned reads

```
gatk --java-options "-Xmx4g" HaplotypeCaller \
  -R Homo_sapiens_assembly38.fasta \
  -I input.bam \
  -O output.vcf.gz \
  -bamout bamout.bam
```

Caveats

We have not yet fully tested the interaction between the GVCF-based calling or the multisample calling and the RNAseq-specific functionalities. Use those in combination at your own risk.

Special note on ploidy

This tool is able to handle many non-diploid use cases; the desired ploidy can be specified using the `-ploidy` argument. Note however that very high ploidies (such as are encountered in large pooled experiments) may cause performance challenges including excessive slowness. We are working on resolving these limitations.

Additional Notes

- When working with PCR-free data, be sure to set ``-pcr_indel_model NONE`` (see argument below).
 - When running in ``-ERC GVCF`` or ``-ERC BP_RESOLUTION`` modes, the confidence threshold is automatically set to 0. This cannot be overridden by the command line. The threshold can be set manually to the desired level in the next step of the workflow (GenotypeGVCFs)
 - We recommend using a list of intervals to speed up analysis. See [this document](#) for details.
-

Additional Information

Read filters

These Read Filters are automatically applied to the data by the Engine before processing by HaplotypeCaller.

HaplotypeCaller specific arguments

This table summarizes the command-line arguments that are specific to this tool. For more details on each argument, see the list further down below the table or click on an argument name to jump directly to that entry in the list.

Argument name(s)	Default value	Summary
Required Arguments		
--input <code>-I</code>	<code>[]</code>	BAM/SAM/CRAM file containing reads
--output <code>-O</code>		File to which variants should be written
--reference <code>-R</code>		Reference sequence file
Optional Tool Arguments		

Argument name(s)	Default value	Summary
<u>--alleles</u>		The set of alleles to force-call regardless of evidence
<u>--annotate-with-num-discovered-alleles</u>	false	If provided, we will annotate records with the number of alternate alleles that were discovered (but not necessarily genotyped) at a given site
<u>--annotation-A</u>	[]	One or more specific annotations to add to variant calls
<u>--annotation-group-G</u>	[]	One or more groups of annotations to apply to variant calls
<u>--annotations-to-exclude-AX</u>	[]	One or more specific annotations to exclude from variant calls
<u>--arguments_file</u>	[]	read one or more arguments files and add them to the command line
<u>--assembly-region-out</u>		Output the assembly region to this IGV formatted file
<u>--assembly-region-padding</u>	100	Number of additional bases of context to include around each assembly region
<u>--base-quality-score-threshold</u>	18	Base qualities below this threshold will be reduced to the minimum (6)
<u>--cloud-index-prefetch-buffer-CIPB</u>	-1	Size of the cloud-only prefetch buffer (in MB; 0 to disable). Defaults to cloudPrefetchBuffer if unset.
<u>--cloud-prefetch-buffer-CPB</u>	40	Size of the cloud-only prefetch buffer (in MB; 0 to disable).
<u>--contamination-fraction-to-filter-contamination</u>	0.0	Fraction of contamination in sequencing data (for all samples) to aggressively remove

Argument name(s)	Default value	Summary
<u>--correct-overlapping-quality</u>	false	Undocumented option
<u>--dbsnp</u> <i>-D</i>		dbSNP file
<u>--disable-bam-index-caching</u> <i>-DBIC</i>	false	If true, don't cache bam indexes, this will reduce memory requirements but may harm performance if many intervals are specified. Caching is automatically disabled if there are no intervals specified.
<u>--disable-sequence-dictionary-validation</u>	false	If specified, do not check the sequence dictionaries from our inputs for compatibility. Use at your own risk!
<u>--founder-id</u>	[]	Samples representing the population "founders"
<u>--gcs-max-retries</u> <i>-gcs-retries</i>	20	If the GCS bucket channel errors out, how many times it will attempt to re-initiate the connection
<u>--gcs-project-for-requester-pays</u>	""	Project to bill when accessing "requester pays" buckets. If unset, these buckets cannot be accessed.
<u>--graph-output</u> <i>-graph</i>		Write debug assembly graph information to this file
<u>--help</u> <i>-h</i>	false	display the help message
<u>--heterozygosity</u>	0.001	Heterozygosity value used to compute prior likelihoods for any locus. See the GATKDocs for full details on the meaning of this population genetics concept
<u>--heterozygosity-stdev</u>	0.01	Standard deviation of heterozygosity for SNP and indel calling.

Argument name(s)	Default value	Summary
<u>--indel-heterozygosity</u>	1.25E-4	Heterozygosity for indel calling. See the GATKDocs for heterozygosity for full details on the meaning of this population genetics concept
<u>--interval-merging-rule</u> <i>-imr</i>	ALL	Interval merging rule for abutting intervals
<u>--intervals</u> <i>-L</i>	[]	One or more genomic intervals over which to operate
<u>--max-assembly-region-size</u>	300	Maximum size of an assembly region
<u>--max-reads-per-alignment-start</u>	50	Maximum number of reads to retain per alignment start position. Reads above this threshold will be downsampled. Set to 0 to disable.
<u>--min-assembly-region-size</u>	50	Minimum size of an assembly region
<u>--min-base-quality-score</u> <i>-mbq</i>	10	Minimum base quality required to consider a base for calling
<u>--native-pair-hmm-threads</u>	4	How many threads should a native pairHMM implementation use
<u>--native-pair-hmm-use-double-precision</u>	false	use double precision in the native pairHMM. This is slower but matches the java implementation better
<u>--num-reference-samples-if-no-call</u>	0	Number of hom-ref genotypes to infer at sites not present in a panel
<u>--output-mode</u>	EMIT_VARIANTS_ONLY	Specifies which type of calls we should output
<u>--pedigree</u> <i>-ped</i>		Pedigree file for determining the population "founders"

Argument name(s)	Default value	Summary
<u>--population-callset</u> <i>-population</i>		Callset to use in calculating genotype priors
<u>--sample-name</u> <i>-ALIAS</i>		Name of single sample to use from a multi-sample bam
<u>--sample-ploidy</u> <i>-ploidy</i>	2	Ploidy (number of chromosomes) per sample. For pooled data, set to (Number of samples in each pool * Sample Ploidy).
<u>--sites-only-vcf-output</u>	false	If true, don't emit genotype fields when writing vcf file output.
<u>--standard-min-confidence-threshold-for-calling</u> <i>-stand-call-conf</i>	30.0	The minimum phred-scaled confidence threshold at which variants should be called
<u>--version</u>	false	display the version number for this tool
Optional Common Arguments		
<u>--add-output-sam-program-record</u>	true	If true, adds a PG tag to created SAM/BAM/CRAM files.
<u>--add-output-vcf-command-line</u>	true	If true, adds a command line header line to created VCF files.
<u>--create-output-bam-index</u> <i>-OBI</i>	true	If true, create a BAM/CRAM index when writing a coordinate-sorted BAM/CRAM file.
<u>--create-output-bam-md5</u> <i>-OBM</i>	false	If true, create a MD5 digest for any BAM/SAM/CRAM file created

Argument name(s)	Default value	Summary
<u>--create-output-variant-index</u> <i>-OVI</i>	true	If true, create a VCF index when writing a coordinate-sorted VCF file.
<u>--create-output-variant-md5</u> <i>-OVM</i>	false	If true, create a MD5 digest any VCF file created.
<u>--disable-read-filter</u> <i>-DF</i>	[]	Read filters to be disabled before analysis
<u>--disable-tool-default-read-filters</u>	false	Disable all tool default read filters (WARNING: many tools will not function correctly without their default read filters on)
<u>--exclude-intervals</u> <i>-XL</i>	[]	One or more genomic intervals to exclude from processing
<u>--gatk-config-file</u>		A configuration file to use with the GATK.
<u>--interval-exclusion-padding</u> <i>-ixp</i>	0	Amount of padding (in bp) to add to each interval you are excluding.
<u>--interval-padding</u> <i>-ip</i>	0	Amount of padding (in bp) to add to each interval you are including.
<u>--interval-set-rule</u> <i>-isr</i>	UNION	Set merging approach to use for combining interval inputs
<u>--lenient</u> <i>-LE</i>	false	Lenient processing of VCF files
<u>--QUIET</u>	false	Whether to suppress job-summary info on System.err.
<u>--read-filter</u> <i>-RF</i>	[]	Read filters to be applied before analysis

Argument name(s)	Default value	Summary
<u>--read-index</u>	[]	Indices to use for the read inputs. If specified, an index must be provided for every read input and in the same order as the read inputs. If this argument is not specified, the path to the index for each input will be inferred automatically.
<u>--read-validation-stringency</u> <i>-VS</i>	SILENT	Validation stringency for all SAM/BAM/CRAM/SRA files read by this program. The default stringency value SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded.
<u>--seconds-between-progress-updates</u>	10.0	Output traversal statistics every time this many seconds elapse
<u>--sequence-dictionary</u>		Use the given sequence dictionary as the master/canonical sequence dictionary. Must be a .dict file.
<u>--tmp-dir</u>		Temp directory to use.
<u>--use-jdk-deflater</u> <i>-jdk-deflater</i>	false	Whether to use the JdkDeflater (as opposed to IntelDeflater)
<u>--use-jdk-inflater</u> <i>-jdk-inflater</i>	false	Whether to use the JdkInflater (as opposed to IntelInflater)
<u>--verbosity</u>	INFO	Control verbosity of logging.
Advanced Arguments		
<u>--active-probability-threshold</u>	0.002	Minimum probability for a locus to be considered active.
<u>--adaptive-pruning</u>	false	Use Mutect2's adaptive graph pruning algorithm

Argument name(s)	Default value	Summary
<u>--adaptive-pruning-initial-error-rate</u>	0.001	Initial base error rate estimate for adaptive pruning
<u>--all-site-pls</u>	false	Annotate all sites with PLs
<u>--allele-informative-reads-overlap-margin</u>	2	Likelihood and read-based annotations will only take into consideration reads that overlap the variant or any base no further than this distance expressed in base pairs
<u>--allow-non-unique-kmers-in-ref</u>	false	Allow graphs that have non-unique kmers in the reference
<u>--bam-output</u> <i>-bamout</i>		File to which assembled haplotypes should be written
<u>--bam-writer-type</u>	CALLED_HAPLOTYPES	Which haplotypes should be written to the BAM
<u>--comparison</u> <i>-comp</i>	[]	Comparison VCF file(s)
<u>--contamination-fraction-per-sample-file</u> <i>-contamination-file</i>		Tab-separated File containing fraction of contamination in sequencing data (per sample) to aggressively remove. Format should be "" (Contamination is double) per line; No header.
<u>--debug-assembly</u> <i>-debug</i>	false	Print out verbose debug information about each assembly region
<u>--disable-optimizations</u>	false	Don't skip calculations in ActiveRegions with no variants
<u>--disable-tool-default-annotations</u>	false	Disable all tool default annotations
<u>--do-not-run-physical-phasing</u>	false	Disable physical phasing

Argument name(s)	Default value	Summary
<u>--dont-increase-kmer-sizes-for-cycles</u>	false	Disable iterating over kmer sizes when graph cycles are detected
<u>--dont-use-soft-clipped-bases</u>	false	Do not analyze soft clipped bases in the reads
<u>--emit-ref-confidence-ERC</u>	NONE	Mode for emitting reference confidence scores (For Mutect2, this is a BETA feature)
<u>--enable-all-annotations</u>	false	Use all possible annotations (not for the faint of heart)
<u>--floor-blocks</u>	false	Output the band lower bound for each GQ block regardless of the data it represents
<u>--force-active</u>	false	If provided, all regions will be marked as active
<u>--force-call-filtered-alleles-genotype-filtered-alleles</u>	false	Force-call filtered alleles included in the resource specified by --alleles
<u>--gvcf-gq-bands-GQB</u>	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 70, 80, 90, 99]	Exclusive upper bounds for reference confidence GQ bands (must be in [1, 100] and specified in increasing order)
<u>--indel-size-to-eliminate-in-ref-model</u>	10	The size of an indel to check for in the reference model
<u>--kmer-size</u>	[10, 25]	Kmer size to use in the read threading assembler
<u>--max-alternate-alleles</u>	6	Maximum number of alternate alleles to genotype

Argument name(s)	Default value	Summary
<u>--max-genotype-count</u>	1024	Maximum number of genotypes to consider at any site
<u>--max-mnp-distance</u> <i>-mnp-dist</i>	0	Two or more phased substitutions separated by this distance or less are merged into MNPs.
<u>--max-num-haplotypes-in-population</u>	128	Maximum number of haplotypes to consider for your population
<u>--max-prob-propagation-distance</u>	50	Upper limit on how many bases away probability mass can be moved around when calculating the boundaries between active and inactive assembly regions
<u>--max-unpruned-variants</u>	100	Maximum number of variants in graph the adaptive pruner will allow
<u>--min-dangling-branch-length</u>	4	Minimum length of a dangling branch to attempt recovery
<u>--min-pruning</u>	2	Minimum support to not prune paths in the graph
<u>--num-pruning-samples</u>	1	Number of samples that must pass the minPruning threshold
<u>--pair-hmm-gap-continuation-penalty</u>	10	Flat gap continuation penalty for use in the Pair HMM
<u>--pair-hmm-implementation</u> <i>-pairHMM</i>	FASTEST_AVAILABLE	The PairHMM implementation to use for genotype likelihood calculations
<u>--pcr-indel-model</u>	CONSERVATIVE	The PCR indel model to use
<u>--phred-scaled-global-read-mismatching-rate</u>	45	The global assumed mismapping rate for reads

Argument name(s)	Default value	Summary
<u>--pruning-lod-threshold</u>	2.302585092994046	Ln likelihood ratio threshold for adaptive pruning algorithm
<u>--recover-all-dangling-branches</u>	false	Recover all dangling branches
<u>--showHidden</u>	false	display hidden arguments
<u>--smith-waterman</u>	JAVA	Which Smith-Waterman implementation to use, generally FASTEST_AVAILABLE is the right choice
<u>--use-filtered-reads-for-annotations</u>	false	Use the contamination-filtered read maps for the purposes of annotating variants
Deprecated Arguments		
<u>--recover-dangling-heads</u>	false	This argument is deprecated since version 3.3

Argument details

Arguments in this list are specific to this tool. Keep in mind that other arguments are available that are shared with other tools (e.g. command-line GATK arguments); see Inherited arguments above.

[--active-probability-threshold](#) / NA

Minimum probability for a locus to be considered active.

double 0.002 [[-∞ ∞]]

[--adaptive-pruning](#) / NA

Use Mutect2's adaptive graph pruning algorithm

A single edge multiplicity cutoff for pruning doesn't work in samples with variable depths, for example exomes and RNA. This parameter enables the probabilistic algorithm for pruning the assembly graph that considers the likelihood that each chain in the graph comes from real variation.

boolean false

--adaptive-pruning-initial-error-rate / NA

Initial base error rate estimate for adaptive pruning

Initial base error rate guess for the probabilistic adaptive pruning model. Results are not very sensitive to this parameter because it is only a starting point from which the algorithm discovers the true error rate.

double 0.001 [[-∞ ∞]]

--add-output-sam-program-record / -add-output-sam-program-record

If true, adds a PG tag to created SAM/BAM/CRAM files.

boolean true

--add-output-vcf-command-line / -add-output-vcf-command-line

If true, adds a command line header line to created VCF files.

boolean true

--all-site-pls / NA

Annotate all sites with PLs

Advanced, experimental argument: if SNP likelihood model is specified, and if EMIT_ALL_ACTIVE_SITES output mode is set, when we set this argument then we will also emit PLs at all sites. This will give a measure of reference confidence and a measure of which alt alleles are more plausible (if any). WARNINGS: - This feature will inflate VCF file size considerably. - All SNP ALT alleles will be emitted with corresponding 10 PL values. - An error will be emitted if EMIT_ALL_ACTIVE_SITES is not set, or if anything other than diploid SNP model is used

boolean false

--allele-informative-reads-overlap-margin / NA

Likelihood and read-based annotations will only take into consideration reads that overlap the variant or any base no further than this distance expressed in base pairs

int 2 [[-∞ ∞]]

--alleles / NA

The set of alleles to force-call regardless of evidence

FeatureInput[VariantContext]

--allow-non-unique-kmers-in-ref / NA

Allow graphs that have non-unique kmers in the reference

By default, the program does not allow processing of reference sections that contain non-unique kmers. Disabling this check may cause problems in the assembly graph.

boolean false

--annotate-with-num-discovered-alleles / NA

If provided, we will annotate records with the number of alternate alleles that were discovered (but not necessarily genotyped) at a given site

Depending on the value of the --max_alternate_alleles argument, we may genotype only a fraction of the alleles being sent on for genotyping. Using this argument instructs the genotyper to annotate (in the INFO field) the number of alternate alleles that were originally discovered at the site.

boolean false

--annotation / -A

One or more specific annotations to add to variant calls

Which annotations to include in variant calls in the output. These supplement annotations provided by annotation groups.

List[String] []

--annotation-group / -G

One or more groups of annotations to apply to variant calls

Which groups of annotations to add to the output variant calls. Any requirements that are not met (e.g. failing to provide a pedigree file for a pedigree-based annotation) may cause the run to fail.

List[String] []

--annotations-to-exclude / -AX

One or more specific annotations to exclude from variant calls

Which annotations to exclude from output in the variant calls. Note that this argument has higher priority than the -A or -G arguments, so these annotations will be excluded even if they are explicitly included with the other options.

List[String] []

--arguments_file / NA

read one or more arguments files and add them to the command line

List[File] []

--assembly-region-out / NA

Output the assembly region to this IGV formatted file

If provided, this walker will write out its assembly regions to this file in the IGV formatted TAB-delimited output: <http://www.broadinstitute.org/software/igv/IGV> Intended to make debugging the active region calculations easier

String

--assembly-region-padding / NA

Number of additional bases of context to include around each assembly region

Parameters that control assembly regions

int 100 [[-∞ ∞]]

--bam-output / -bamout

File to which assembled haplotypes should be written

The assembled haplotypes and locally realigned reads will be written as BAM to this file if requested. Really for debugging purposes only. Note that the output here does not include uninformative reads so that not every input read is emitted to the bam. Turning on this mode may result in serious performance cost for the caller. It's really only appropriate to use in specific areas where you want to better understand why the caller is making specific calls. The reads are written out containing an "HC" tag (integer) that encodes which haplotype each read best matches according to the haplotype caller's likelihood calculation. The use of this tag is primarily intended to allow good coloring of reads in IGV. Simply go to "Color

Alignments By > Tag" and enter "HC" to more easily see which reads go with these haplotype. Note that the haplotypes (called or all, depending on mode) are emitted as single reads covering the entire active region, coming from sample "HC" and a special read group called "ArtificialHaplotype". This will increase the pileup depth compared to what would be expected from the reads only, especially in complex regions. Note also that only reads that are actually informative about the haplotypes are emitted. By informative we mean that there's a meaningful difference in the likelihood of the read coming from one haplotype compared to its next best haplotype. If multiple BAMs are passed as input to the tool (as is common for M2), then they will be combined in the bamout output and tagged with the appropriate sample names. The best way to visualize the output of this mode is with IGV. Tell IGV to color the alignments by tag, and give it the "HC" tag, so you can see which reads support each haplotype. Finally, you can tell IGV to group by sample, which will separate the potential haplotypes from the reads. All of this can be seen in [this screenshot](#)

String

--bam-writer-type / NA

Which haplotypes should be written to the BAM

The type of BAM output we want to see. This determines whether HC will write out all of the haplotypes it considered (top 128 max) or just the ones that were selected as alleles and assigned to samples.

The --bam-writer-type argument is an enumerated type (WriterType), which can have one of the following values:

ALL_POSSIBLE_HAPLOTYPES

A mode that's for method developers. Writes out all of the possible haplotypes considered, as well as reads aligned to each

CALLED_HAPLOTYPES

A mode for users. Writes out the reads aligned only to the called haplotypes. Useful to understand why the caller is calling what it is

WriterType CALLED_HAPLOTYPES

--base-quality-score-threshold / NA

Base qualities below this threshold will be reduced to the minimum (6)

Bases with a quality below this threshold will reduced to the minimum usable quality score (6).

byte 18 [[-∞ ∞]]

--cloud-index-prefetch-buffer / -CIPB

Size of the cloud-only prefetch buffer (in MB; 0 to disable). Defaults to cloudPrefetchBuffer if unset.

int -1 [[-∞ ∞]]

--cloud-prefetch-buffer / -CPB

Size of the cloud-only prefetch buffer (in MB; 0 to disable).

int 40 [[-∞ ∞]]

--comparison / -comp

Comparison VCF file(s)

If a call overlaps with a record from the provided comp track, the INFO field will be annotated as such in the output with the track name (e.g. -comp:FOO will have 'FOO' in the INFO field). Records that are filtered in the comp track will be ignored. Note that 'dbSNP' has been special-cased (see the --dbSNP argument).

List[FeatureInput[VariantContext]] []

--contamination-fraction-per-sample-file / -contamination-file

Tab-separated File containing fraction of contamination in sequencing data (per sample) to aggressively remove. Format should be "" (Contamination is double) per line; No header.

This argument specifies a file with two columns "sample" and "contamination" specifying the contamination level for those samples. Samples that do not appear in this file will be processed with CONTAMINATION_FRACTION.

File

--contamination-fraction-to-filter / -contamination

Fraction of contamination in sequencing data (for all samples) to aggressively remove

If this fraction is greater than zero, the caller will aggressively attempt to remove contamination through biased down-sampling of reads. Basically, it will ignore the

contamination fraction of reads for each alternate allele. So if the pileup contains N total bases, then we will try to remove (N * contamination fraction) bases for each alternate allele.

double 0.0 [[-∞ ∞]]

--correct-overlapping-quality / NA

Undocumented option

boolean false

--create-output-bam-index / -OBI

If true, create a BAM/CRAM index when writing a coordinate-sorted BAM/CRAM file.

boolean true

--create-output-bam-md5 / -OBM

If true, create a MD5 digest for any BAM/SAM/CRAM file created

boolean false

--create-output-variant-index / -OVI

If true, create a VCF index when writing a coordinate-sorted VCF file.

boolean true

--create-output-variant-md5 / -OVM

If true, create a MD5 digest any VCF file created.

boolean false

--dbSNP / -D

dbSNP file

A dbSNP VCF file.

FeatureInput[VariantContext]

--debug-assembly / -debug

Print out verbose debug information about each assembly region

boolean false

--disable-bam-index-caching / -DBIC

If true, don't cache bam indexes, this will reduce memory requirements but may harm performance if many intervals are specified. Caching is automatically disabled if there are no intervals specified.

boolean false

--disable-optimizations / NA

Don't skip calculations in ActiveRegions with no variants

If set, certain "early exit" optimizations in HaplotypeCaller, which aim to save compute and time by skipping calculations if an ActiveRegion is determined to contain no variants, will be disabled. This is most likely to be useful if you're using the -bamout argument to examine the placement of reads following reassembly and are interested in seeing the mapping of reads in regions with no variations. Setting the --force-active flag may also be necessary.

boolean false

--disable-read-filter / -DF

Read filters to be disabled before analysis

List[String] []

--disable-sequence-dictionary-validation / -disable-sequence-dictionary-validation

If specified, do not check the sequence dictionaries from our inputs for compatibility. Use at your own risk!

boolean false

--disable-tool-default-annotations / -disable-tool-default-annotations

Disable all tool default annotations

Hook allowing for the user to remove default annotations from the tool

boolean false

--disable-tool-default-read-filters / -disable-tool-default-read-filters

Disable all tool default read filters (WARNING: many tools will not function correctly without their default read filters on)

boolean false

--do-not-run-physical-phasing / NA

Disable physical phasing

As of GATK 3.3, HaplotypeCaller outputs physical (read-based) information (see version 3.3 release notes and documentation for details). This argument disables that behavior.

boolean false

--dont-increase-kmer-sizes-for-cycles / NA

Disable iterating over kmer sizes when graph cycles are detected

When graph cycles are detected, the normal behavior is to increase kmer sizes iteratively until the cycles are resolved. Disabling this behavior may cause the program to give up on assembling the ActiveRegion.

boolean false

--dont-use-soft-clipped-bases / NA

Do not analyze soft clipped bases in the reads

boolean false

--emit-ref-confidence / -ERC

Mode for emitting reference confidence scores (For Mutect2, this is a BETA feature)

The reference confidence mode makes it possible to emit a per-bp or summarized confidence estimate for a site being strictly homozygous-reference. See

<https://software.broadinstitute.org/gatk/documentation/article.php?id=4017> for information about GVCFs. For Mutect2, this is a BETA feature that functions similarly to the HaplotypeCaller reference confidence/GVCF mode.

The `--emit-ref-confidence` argument is an enumerated type (`ReferenceConfidenceMode`), which can have one of the following values:

NONE

Regular calling without emitting reference confidence calls.

BP_RESOLUTION

Reference model emitted site by site.

GVCF

Reference model emitted with condensed non-variant blocks, i.e. the GVCF format.

`ReferenceConfidenceMode NONE`

--enable-all-annotations / NA

Use all possible annotations (not for the faint of heart)

You can use the `-AX` argument in combination with this one to exclude specific annotations. Note that some annotations may not be actually applied if they are not applicable to the data provided or if they are unavailable to the tool (e.g. there are several annotations that are currently not hooked up to HaplotypeCaller). At present no error or warning message will be provided, the annotation will simply be skipped silently. You can check the output VCF header to see which annotations were activated and thus might be applied (although this does not guarantee that the annotation was applied to all records in the VCF, since some annotations have additional requirements, e.g. minimum number of samples or heterozygous sites only -- see the documentation for individual annotations' requirements).

`boolean false`

--exclude-intervals / -XL

One or more genomic intervals to exclude from processing

Use this argument to exclude certain parts of the genome from the analysis (like `-L`, but the opposite). This argument can be specified multiple times. You can use samtools-style intervals either explicitly on the command line (e.g. `-XL 1` or `-XL 1:100-200`) or by loading in a file containing a list of intervals (e.g. `-XL myFile.intervals`).

`List[String] []`

--floor-blocks / NA

Output the band lower bound for each GQ block regardless of the data it represents

Output the band lower bound for each GQ block instead of the min GQ -- for better compression

boolean false

--force-active / NA

If provided, all regions will be marked as active

boolean false

--force-call-filtered-alleles / -genotype-filtered-alleles

Force-call filtered alleles included in the resource specified by --alleles

boolean false

--founder-id / -founder-id

Samples representing the population "founders"

List[String] []

--gatk-config-file / NA

A configuration file to use with the GATK.

String

--gcs-max-retries / -gcs-retries

If the GCS bucket channel errors out, how many times it will attempt to re-initiate the connection

int 20 [[-∞ ∞]]

--gcs-project-for-requester-pays / NA

Project to bill when accessing "requester pays" buckets. If unset, these buckets cannot be accessed.

String ""

--graph-output / -graph

Write debug assembly graph information to this file

This argument is meant for debugging and is not immediately useful for normal analysis use.

String

--gvcf-gq-bands / -GQB

Exclusive upper bounds for reference confidence GQ bands (must be in [1, 100] and specified in increasing order)

When HC is run in reference confidence mode with banding compression enabled (-ERC GVCF), homozygous-reference sites are compressed into bands of similar genotype quality (GQ) that are emitted as a single VCF record. See the FAQ documentation for more details about the GVCF format. This argument allows you to set the GQ bands. HC expects a list of strictly increasing GQ values that will act as exclusive upper bounds for the GQ bands. To pass multiple values, you provide them one by one with the argument, as in ``-GQB 10 -GQB 20 -GQB 30`` and so on (this would set the GQ bands to be ``[0, 10), [10, 20), [20, 30)`` and so on, for example). Note that GQ values are capped at 99 in the GATK, so values must be integers in [1, 100]. If the last value is strictly less than 100, the last GQ band will start at that value (inclusive) and end at 100 (exclusive).

List[Integer] [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 70, 80, 90, 99]

--help / -h

display the help message

boolean false

--heterozygosity / NA

Heterozygosity value used to compute prior likelihoods for any locus. See the GATKDocs for full details on the meaning of this population genetics concept

The expected heterozygosity value used to compute prior probability that a locus is non-reference. The default priors are for provided for humans: $het = 1e-3$ which means that the probability of N samples being hom-ref at a site is: $1 - \sum_i 2N (het / i)$ Note that heterozygosity as used here is the population genetics concept:

http://en.wikipedia.org/wiki/Zygosity#Heterozygosity_in_population_genetics That is, a hets value of 0.01 implies that two randomly chosen chromosomes from the population of organisms would differ from each other (one being A and the other B) at a rate of 1 in 100 bp. Note that this quantity has nothing to do with the likelihood of any given sample having a heterozygous genotype, which in the GATK is purely determined by the probability of the observed data $P(D | AB)$ under the model that there may be a AB het genotype. The posterior probability of this AB genotype would use the het prior, but the GATK only uses this posterior probability in determining the prob. that a site is polymorphic. So changing the het parameters only increases the chance that a site will be called non-reference across all samples, but doesn't actually change the output genotype likelihoods at all, as these aren't posterior probabilities at all. The quantity that changes whether the GATK considers the possibility of a het genotype at all is the ploidy, which determines how many chromosomes each individual in the species carries.

Double 0.001 [[-∞ ∞]]

--heterozygosity-stdev / NA

Standard deviation of heterozygosity for SNP and indel calling.

The standard deviation of the distribution of alt allele fractions. The above heterozygosity parameters give the *mean* of this distribution; this parameter gives its spread.

double 0.01 [[-∞ ∞]]

--indel-heterozygosity / NA

Heterozygosity for indel calling. See the GATKDocs for heterozygosity for full details on the meaning of this population genetics concept

This argument informs the prior probability of having an indel at a site.

double 1.25E-4 [[-∞ ∞]]

--indel-size-to-eliminate-in-ref-model / NA

The size of an indel to check for in the reference model

This parameter determines the maximum size of an indel considered as potentially segregating in the reference model. It is used to eliminate reads from being indel informative at a site, and determines by that mechanism the certainty in the reference base. Conceptually, setting this parameter to X means that each informative read is consistent with any indel of size $< X$ being present at a specific position in the genome, given its alignment to the reference.

int 10 $[-\infty \infty]$

--input / -l

BAM/SAM/CRAM file containing reads

 List[String] []

--interval-exclusion-padding / -ixp

Amount of padding (in bp) to add to each interval you are excluding.

Use this to add padding to the intervals specified using -XL. For example, '-XL 1:100' with a padding value of 20 would turn into '-XL 1:80-120'. This is typically used to add padding around targets when analyzing exomes.

int 0 $[-\infty \infty]$

--interval-merging-rule / -imr

Interval merging rule for abutting intervals

By default, the program merges abutting intervals (i.e. intervals that are directly side-by-side but do not actually overlap) into a single continuous interval. However you can change this behavior if you want them to be treated as separate intervals instead.

The --interval-merging-rule argument is an enumerated type (IntervalMergingRule), which can have one of the following values:

ALL

OVERLAPPING_ONLY

IntervalMergingRule ALL

--interval-padding / -ip

Amount of padding (in bp) to add to each interval you are including.

Use this to add padding to the intervals specified using -L. For example, '-L 1:100' with a padding value of 20 would turn into '-L 1:80-120'. This is typically used to add padding around targets when analyzing exomes.

int 0 [[-∞ ∞]]

--interval-set-rule / -isr

Set merging approach to use for combining interval inputs

By default, the program will take the UNION of all intervals specified using -L and/or -XL. However, you can change this setting for -L, for example if you want to take the INTERSECTION of the sets instead. E.g. to perform the analysis only on chromosome 1 exomes, you could specify -L exomes.intervals -L 1 --interval-set-rule INTERSECTION. However, it is not possible to modify the merging approach for intervals passed using -XL (they will always be merged using UNION). Note that if you specify both -L and -XL, the -XL interval set will be subtracted from the -L interval set.

The --interval-set-rule argument is an enumerated type (IntervalSetRule), which can have one of the following values:

UNION

Take the union of all intervals

INTERSECTION

Take the intersection of intervals (the subset that overlaps all intervals specified)

IntervalSetRule UNION

--intervals / -L

One or more genomic intervals over which to operate

List[String] []

--kmer-size / NA

Kmer size to use in the read threading assembler

Multiple kmer sizes can be specified, using e.g. '--kmer-size 10 --kmer-size 25'.

List[Integer] [10, 25]

--lenient / -LE

Lenient processing of VCF files

boolean false

--max-alternate-alleles / NA

Maximum number of alternate alleles to genotype

If there are more than this number of alternate alleles presented to the genotyper (either through discovery or GENOTYPE_GIVEN_ALLELES), then only this many alleles will be used. Note that genotyping sites with many alternate alleles is both CPU and memory intensive and it scales exponentially based on the number of alternate alleles. Unless there is a good reason to change the default value, we highly recommend that you not play around with this parameter. See also {@link #MAX_GENOTYPE_COUNT}.

int 6 [[-∞ ∞]]

--max-assembly-region-size / NA

Maximum size of an assembly region

int 300 [[-∞ ∞]]

--max-genotype-count / NA

Maximum number of genotypes to consider at any site

If there are more than this number of genotypes at a locus presented to the genotyper, then only this many genotypes will be used. The possible genotypes are simply different ways of partitioning alleles given a specific ploidy assumption. Therefore, we remove genotypes from consideration by removing alternate alleles that are the least well supported. The estimate of allele support is based on the ranking of the candidate haplotypes coming out of the graph building step. Note that the reference allele is always kept. Note that genotyping sites with large genotype counts is both CPU and memory intensive. Unless there is a good reason to change the default value, we highly recommend that you not play around with this parameter. The maximum number of alternative alleles used in the genotyping step will be the lesser of the two: 1. the largest number of alt alleles, given ploidy, that yields a genotype count no higher than {@link #MAX_GENOTYPE_COUNT} 2. the value of {@link #MAX_ALTERNATE_ALLELES} See also {@link #MAX_ALTERNATE_ALLELES}.

int 1024 [[-∞ ∞]]

--max-mnp-distance / -mnp-dist

Two or more phased substitutions separated by this distance or less are merged into MNPs.

Two or more phased substitutions separated by this distance or less are merged into MNPs.

int 0 $[-\infty \infty]$

--max-num-haplotypes-in-population / NA

Maximum number of haplotypes to consider for your population

The assembly graph can be quite complex, and could imply a very large number of possible haplotypes. Each haplotype considered requires N PairHMM evaluations if there are N reads across all samples. In order to control the run of the haplotype caller we only take maxNumHaplotypesInPopulation paths from the graph, in order of their weights, no matter how many paths are possible to generate from the graph. Putting this number too low will result in dropping true variation because paths that include the real variant are not even considered. You can consider increasing this number when calling organisms with high heterozygosity.

int 128 $[-\infty \infty]$

--max-prob-propagation-distance / NA

Upper limit on how many bases away probability mass can be moved around when calculating the boundaries between active and inactive assembly regions

int 50 $[-\infty \infty]$

--max-reads-per-alignment-start / NA

Maximum number of reads to retain per alignment start position. Reads above this threshold will be downsampled. Set to 0 to disable.

Other parameters

int 50 $[-\infty \infty]$

--max-unpruned-variants / NA

Maximum number of variants in graph the adaptive pruner will allow

The maximum number of variants in graph the adaptive pruner will allow

int 100 $[-\infty \infty]$

--min-assembly-region-size / NA

Minimum size of an assembly region

Parameters that control active regions

int 50 [[-∞ ∞]]

--min-base-quality-score / -mbq

Minimum base quality required to consider a base for calling

Bases with a quality below this threshold will not be used for calling.

byte 10 [[-∞ ∞]]

--min-dangling-branch-length / NA

Minimum length of a dangling branch to attempt recovery

When constructing the assembly graph we are often left with "dangling" branches. The assembly engine attempts to rescue these branches by merging them back into the main graph. This argument describes the minimum length of a dangling branch needed for the engine to try to rescue it. A smaller number here will lead to higher sensitivity to real variation but also to a higher number of false positives.

int 4 [[-∞ ∞]]

--min-pruning / NA

Minimum support to not prune paths in the graph

Paths with fewer supporting kmers than the specified threshold will be pruned from the graph. Be aware that this argument can dramatically affect the results of variant calling and should only be used with great caution. Using a prune factor of 1 (or below) will prevent any pruning from the graph, which is generally not ideal; it can make the calling much slower and even less accurate (because it can prevent effective merging of "tails" in the graph). Higher values tend to make the calling much faster, but also lowers the sensitivity of the results (because it ultimately requires higher depth to produce calls).

int 2 [[-∞ ∞]]

--native-pair-hmm-threads / NA

How many threads should a native pairHMM implementation use

int 4 $[-\infty \infty]$

--native-pair-hmm-use-double-precision / NA

use double precision in the native pairHmm. This is slower but matches the java implementation better

boolean false

--num-pruning-samples / NA

Number of samples that must pass the minPruning threshold

If fewer samples than the specified number pass the minPruning threshold for a given path, that path will be eliminated from the graph.

int 1 $[-\infty \infty]$

--num-reference-samples-if-no-call / NA

Number of hom-ref genotypes to infer at sites not present in a panel

When a variant is not seen in any panel, this argument controls whether to infer (and with what effective strength) that only reference alleles were observed at that site. E.g. "If not seen in 1000Genomes, treat it as AC=0, AN=2000".

int 0 $[-\infty \infty]$

--output / -O

File to which variants should be written

A raw, unfiltered, highly sensitive callset in VCF format.

 String

--output-mode / NA

Specifies which type of calls we should output

The --output-mode argument is an enumerated type (OutputMode), which can have one of the following values:

EMIT_VARIANTS_ONLY

produces calls only at variant sites

EMIT_ALL_CONFIDENT_SITES

produces calls at variant sites and confident reference sites

EMIT_ALL_ACTIVE_SITES

Produces calls at any region over the activity threshold regardless of confidence. On occasion, this will output HOM_REF records where no call could be confidently made. This does not necessarily output calls for all sites in a region. This argument is intended only for point mutations (SNPs); it will not produce a comprehensive set of indels.

OutputMode EMIT_VARIANTS_ONLY

--pair-hmm-gap-continuation-penalty / NA

Flat gap continuation penalty for use in the Pair HMM

int 10 $[-\infty \infty]$

--pair-hmm-implementation / -pairHMM

The PairHMM implementation to use for genotype likelihood calculations

The PairHMM implementation to use for genotype likelihood calculations. The various implementations balance a tradeoff of accuracy and runtime.

The --pair-hmm-implementation argument is an enumerated type (Implementation), which can have one of the following values:

EXACT

ORIGINAL

LOGLESS_CACHING

AVX_LOGLESS_CACHING

AVX_LOGLESS_CACHING_OMP

EXPERIMENTAL_FPGA_LOGLESS_CACHING

FASTEST_AVAILABLE

Implementation FASTEST_AVAILABLE

--pcr-indel-model / NA

The PCR indel model to use

When calculating the likelihood of variants, we can try to correct for PCR errors that cause indel artifacts. The correction is based on the reference context, and acts specifically around repetitive sequences that tend to cause PCR errors). The variant likelihoods are penalized in increasing scale as the context around a putative indel is more repetitive (e.g.

long homopolymer). The correction can be disabled by specifying '-pcrModel NONE'; in that case the default base insertion/deletion qualities will be used (or taken from the read if generated through the BaseRecalibrator). **VERY IMPORTANT: when using PCR-free sequencing data we definitely recommend setting this argument to NONE.**

The --pcr-indel-model argument is an enumerated type (PCRErrorModel), which can have one of the following values:

NONE

no specialized PCR error model will be applied; if base insertion/deletion qualities are present they will be used

HOSTILE

a most aggressive model will be applied that sacrifices true positives in order to remove more false positives

AGGRESSIVE

a more aggressive model will be applied that sacrifices true positives in order to remove more false positives

CONSERVATIVE

a less aggressive model will be applied that tries to maintain a high true positive rate at the expense of allowing more false positives

PCRErrorModel CONSERVATIVE

--pedigree / -ped

Pedigree file for determining the population "founders"

File

--phred-scaled-global-read-mismatching-rate / NA

The global assumed mismatching rate for reads

The phredScaledGlobalReadMismatchingRate reflects the average global mismatching rate of all reads, regardless of their mapping quality. This term effects the probability that a read originated from the reference haplotype, regardless of its edit distance from the reference, in that the read could have originated from the reference haplotype but from another location in the genome. Suppose a read has many mismatches from the reference, say like 5, but has a very high mapping quality of 60. Without this parameter, the read would contribute $5 * Q30$ evidence in favor of its 5 mismatch haplotype compared to reference, potentially enough to make a call off that single read for all of these events. With this

parameter set to Q30, though, the maximum evidence against any haplotype that this (and any) read could contribute is Q30. Set this term to any negative number to turn off the global mapping rate.

int 45 [[-∞ ∞]]

--population-callset / -population

Callset to use in calculating genotype priors

Supporting external panel. Allele counts from this panel (taken from AC,AN or MLEAC,AN or raw genotypes) will be used to inform the frequency distribution underlying the genotype priors. These files must be VCF 4.2 spec or later. Note that unlike CalculateGenotypePosteriors, HaplotypeCaller only allows one supporting callset.

FeatureInput[VariantContext]

--pruning-lod-threshold / NA

Ln likelihood ratio threshold for adaptive pruning algorithm

Log-10 likelihood ratio threshold for adaptive pruning algorithm.

double 2.302585092994046 [[-∞ ∞]]

--QUIET / NA

Whether to suppress job-summary info on System.err.

Boolean false

--read-filter / -RF

Read filters to be applied before analysis

List[String] []

--read-index / -read-index

Indices to use for the read inputs. If specified, an index must be provided for every read input and in the same order as the read inputs. If this argument is not specified, the path to the index for each input will be inferred automatically.

List[String] []

--read-validation-stringency / -VS

Validation stringency for all SAM/BAM/CRAM/SRA files read by this program. The default stringency value SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded.

The --read-validation-stringency argument is an enumerated type (ValidationStringency), which can have one of the following values:

STRICT
LENIENT
SILENT

ValidationStringency SILENT

--recover-all-dangling-branches / NA

Recover all dangling branches

By default, the read threading assembler does not recover dangling branches that fork after splitting from the reference. This argument tells the assembly engine to recover all dangling branches.

boolean false

--recover-dangling-heads / NA

This argument is deprecated since version 3.3

As of version 3.3, this argument is no longer needed because dangling end recovery is now the default behavior. See GATK 3.3 release notes for more details.

boolean false

--reference / -R

Reference sequence file

 String

--sample-name / -ALIAS

Name of single sample to use from a multi-sample bam

You can use this argument to specify that HC should process a single sample out of a multisample BAM file. This is especially useful if your samples are all in the same file but you need to run them individually through HC in -ERC GVC mode (which is the recommended usage). Note that the name is case-sensitive.

String

--sample-ploidy / -ploidy

Ploidy (number of chromosomes) per sample. For pooled data, set to (Number of samples in each pool * Sample Ploidy).

Sample ploidy - equivalent to number of chromosomes per pool. In pooled experiments this should be = # of samples in pool * individual sample ploidy

int 2 [[-∞ ∞]]

--seconds-between-progress-updates / -seconds-between-progress-updates

Output traversal statistics every time this many seconds elapse

double 10.0 [[-∞ ∞]]

--sequence-dictionary / -sequence-dictionary

Use the given sequence dictionary as the master/canonical sequence dictionary. Must be a .dict file.

String

--showHidden / -showHidden

display hidden arguments

boolean false

--sites-only-vcf-output / NA

If true, don't emit genotype fields when writing vcf file output.

boolean false

--smith-waterman / NA

Which Smith-Waterman implementation to use, generally FASTEST_AVAILABLE is the right choice

The --smith-waterman argument is an enumerated type (Implementation), which can have one of the following values:

FASTEST_AVAILABLE

use the fastest available Smith-Waterman aligner that runs on your hardware

AVX_ENABLED

use the AVX enabled Smith-Waterman aligner

JAVA

use the pure java implementation of Smith-Waterman, works on all hardware

Implementation JAVA

--standard-min-confidence-threshold-for-calling / -stand-call-conf

The minimum phred-scaled confidence threshold at which variants should be called

The minimum phred-scaled confidence threshold at which variants should be called. Only variant sites with QUAL equal or greater than this threshold will be called. Note that since version 3.7, we no longer differentiate high confidence from low confidence calls at the calling step. The default call confidence threshold is set low intentionally to achieve high sensitivity, which will allow false positive calls as a side effect. Be sure to perform some kind of filtering after calling to reduce the amount of false positives in your final callset. Note that when HaplotypeCaller is used in GVCF mode (using either -ERC GVCF or -ERC BP_RESOLUTION) the call threshold is automatically set to zero. Call confidence thresholding will then be performed in the subsequent GenotypeGVCFs command. Note that the default was changed from 10.0 to 30.0 in version 4.1.0.0 to accompany the switch to use the the new quality score by default.

double 30.0 [[-∞ ∞]]

--tmp-dir / NA

Temp directory to use.

GATKPathSpecifier

--use-filtered-reads-for-annotations / NA

Use the contamination-filtered read maps for the purposes of annotating variants

boolean false

--use-jdk-deflater / -jdk-deflater

Whether to use the JdkDeflater (as opposed to IntelDeflater)

boolean false

--use-jdk-inflater / -jdk-inflater

Whether to use the JdkInflater (as opposed to IntelInflater)

boolean false

--verbosity / -verbosity

Control verbosity of logging.

The --verbosity argument is an enumerated type (LogLevel), which can have one of the following values:

ERROR

WARNING

INFO

DEBUG

LogLevel INFO

--version / NA

display the version number for this tool

boolean false

[Return to top](#)

GATK version 4.1.5.0-SNAPSHOT built at Wed, 4 Mar 2020 13:47:07 -0500.

