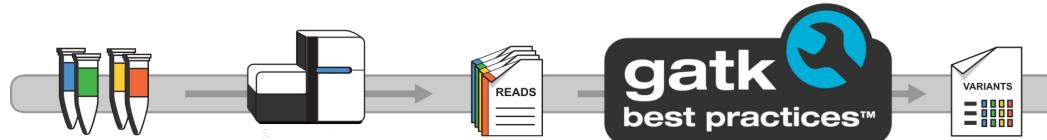


Introduction to Variant Discovery

Basic concepts, variant types
and respective workflows



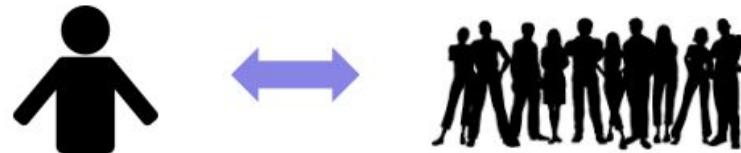
Human genomic variation



3 billions sites in the human genome

Human shares 99.5% DNA with any other human

Variant sites are commonly shared among human and most of these are biallelic



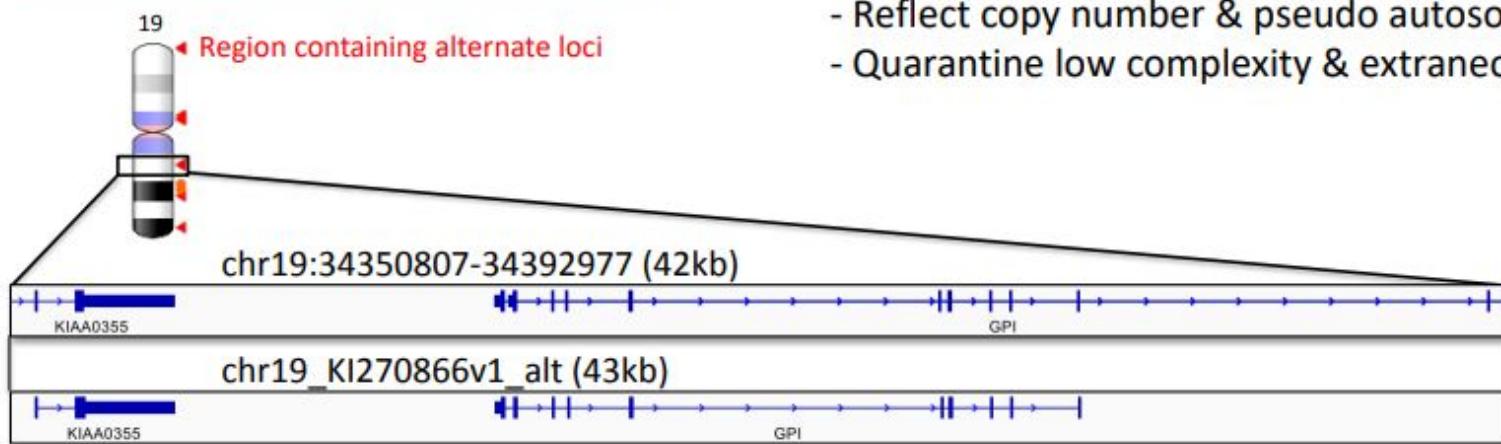
Human genome reference build GRCh38

NCBI Build 34/hg16 (2003)



Iterative refinement

GRCh38 (2013)



Improvements to the reference aim to:

- Better account for population diversity
- Enable detection of complex variants
- Reflect copy number & pseudo autosomal regions
- Quarantine low complexity & extraneous sequences

Jan. 2022 (T2T CHM13v2.0/hs1) (hs1)

Summary	GRCh38p13	CHM13v1.1	±%
Assembled bases (Gbp)	2.92	3.05	+4.5%
Unplaced bases (Mbp)	11.42	0	-100.0%
Gap bases (Mbp)	120.31	0	-100.0%
# Contigs	949	24	-97.5%
Ctg NG50 (Mbp)	56.41	154.26	+173.5%
# Issues	230	46	-80.0%
Issues (Mbp)	230.43	8.18	-96.5%
Gene Annotation			
# Genes	60,090	63,494	+5.7%
protein coding	19,890	19,969	+0.4%
# Exclusive genes	263	3,604	
protein coding	63	140	
# Transcripts	228,597	233,615	+2.2%
protein coding	84,277	86,245	+2.3%
# Exclusive transcripts	1,708	6,693	
protein coding	829	2,780	
Segmental duplications (SDs)			
% SDs	5.00%	6.61%	
SD bases (Mbp)	151.71	201.93	+33.1%
# SDs	24097	41528	+72.3%
RepeatMasker			
% Repeats	50.03%	53.94%	
Repeat bases (Mbp)	1,516.37	1,647.81	+8.7%
LINE	626.33	631.64	+0.8%
SINE	386.48	390.27	+1.0%
LTR	267.52	269.91	+0.9%
Satellite	76.51	150.42	+96.6%
DNA	108.53	109.35	+0.8%
Simple repeat	36.5	77.69	+112.9%
Low complexity	6.16	6.44	+4.6%
Retroposon	4.51	4.65	+3.3%
rRNA	0.21	1.71	+730.4%

What is variant calling ?

Identification of probable variants in an alignment.

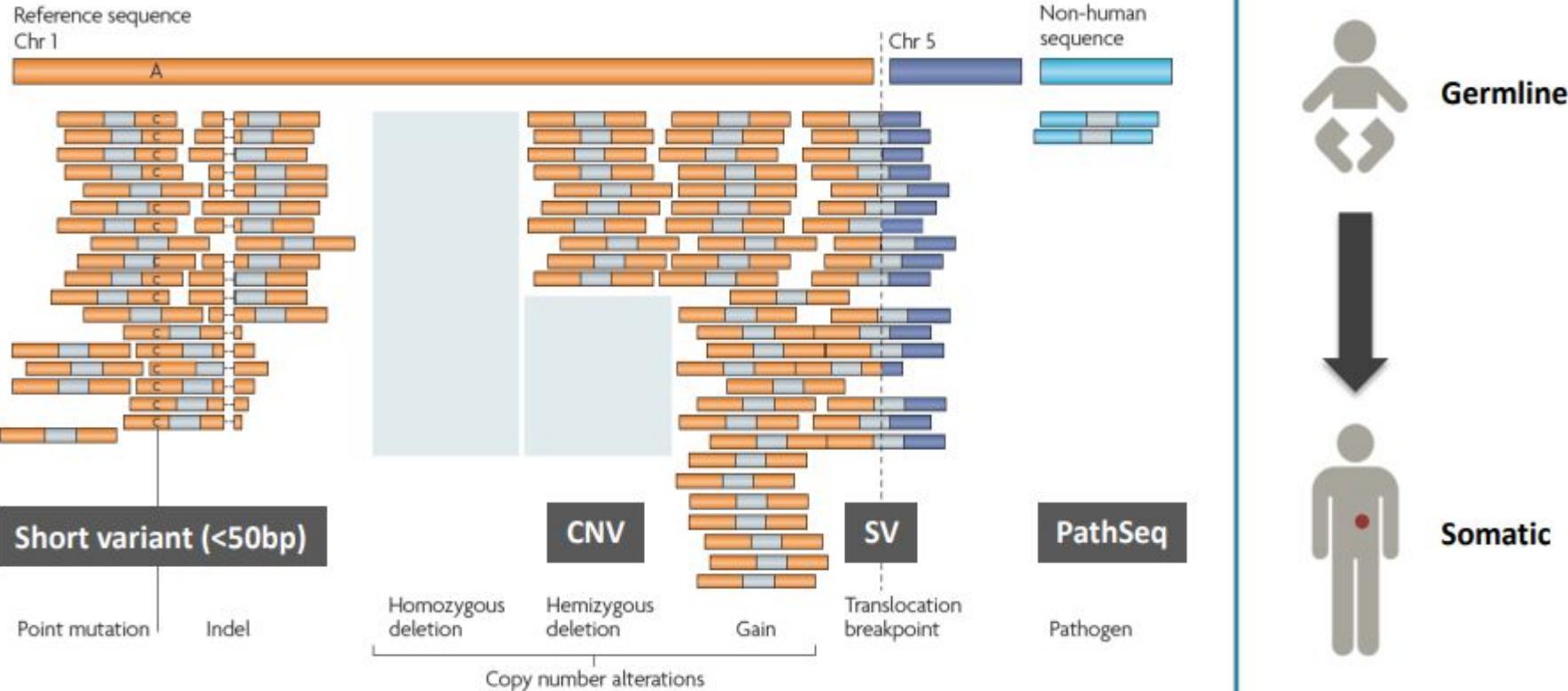
Four main types of variants:

1. Single nucleotide polymorphisms(SNPs) / Short indels
2. Copy number variations
3. Structural variants
4. Microsatellite Instability

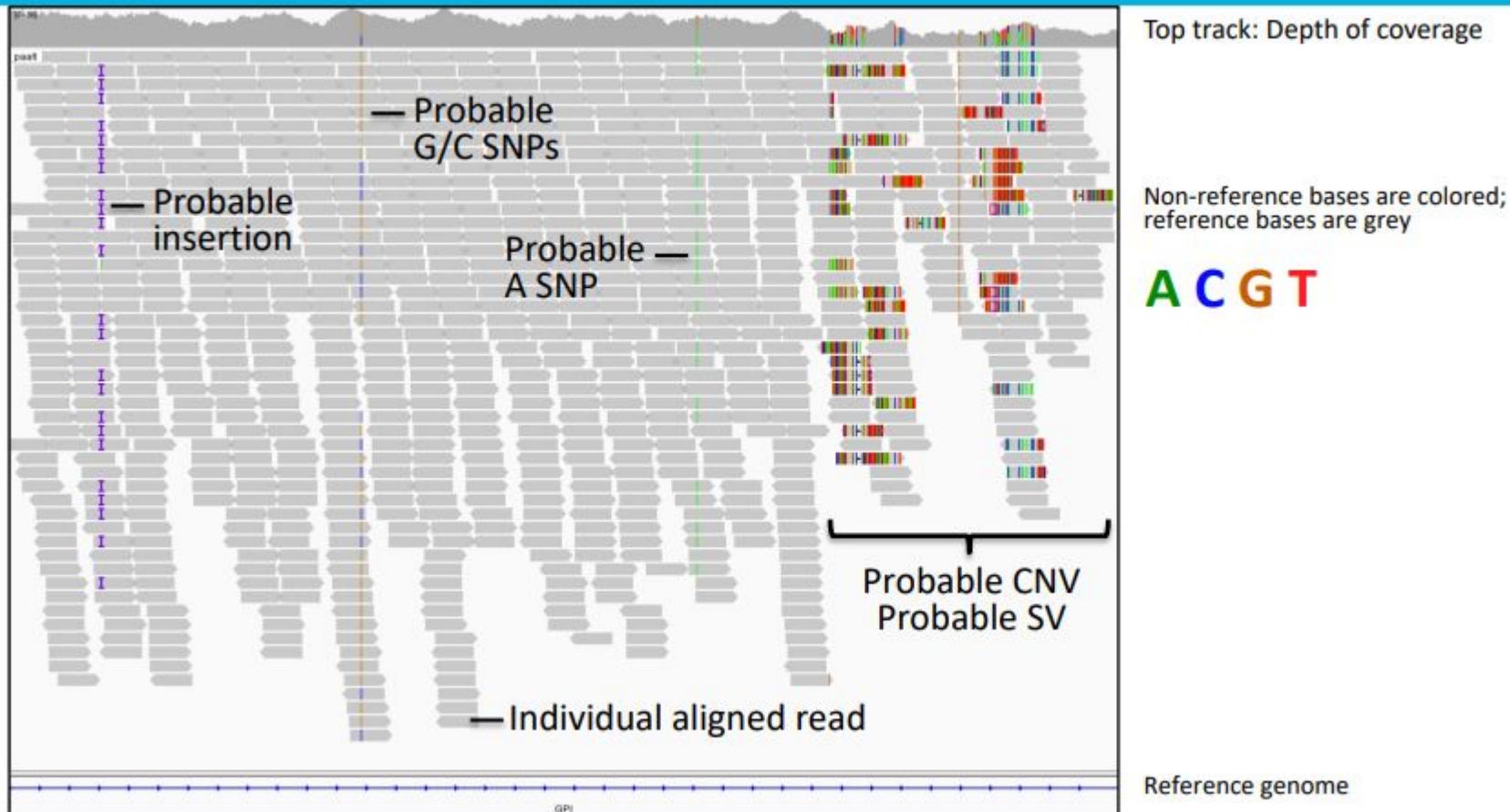
2 main types of variant classifications:

1. Germline variants
2. Somatic variants

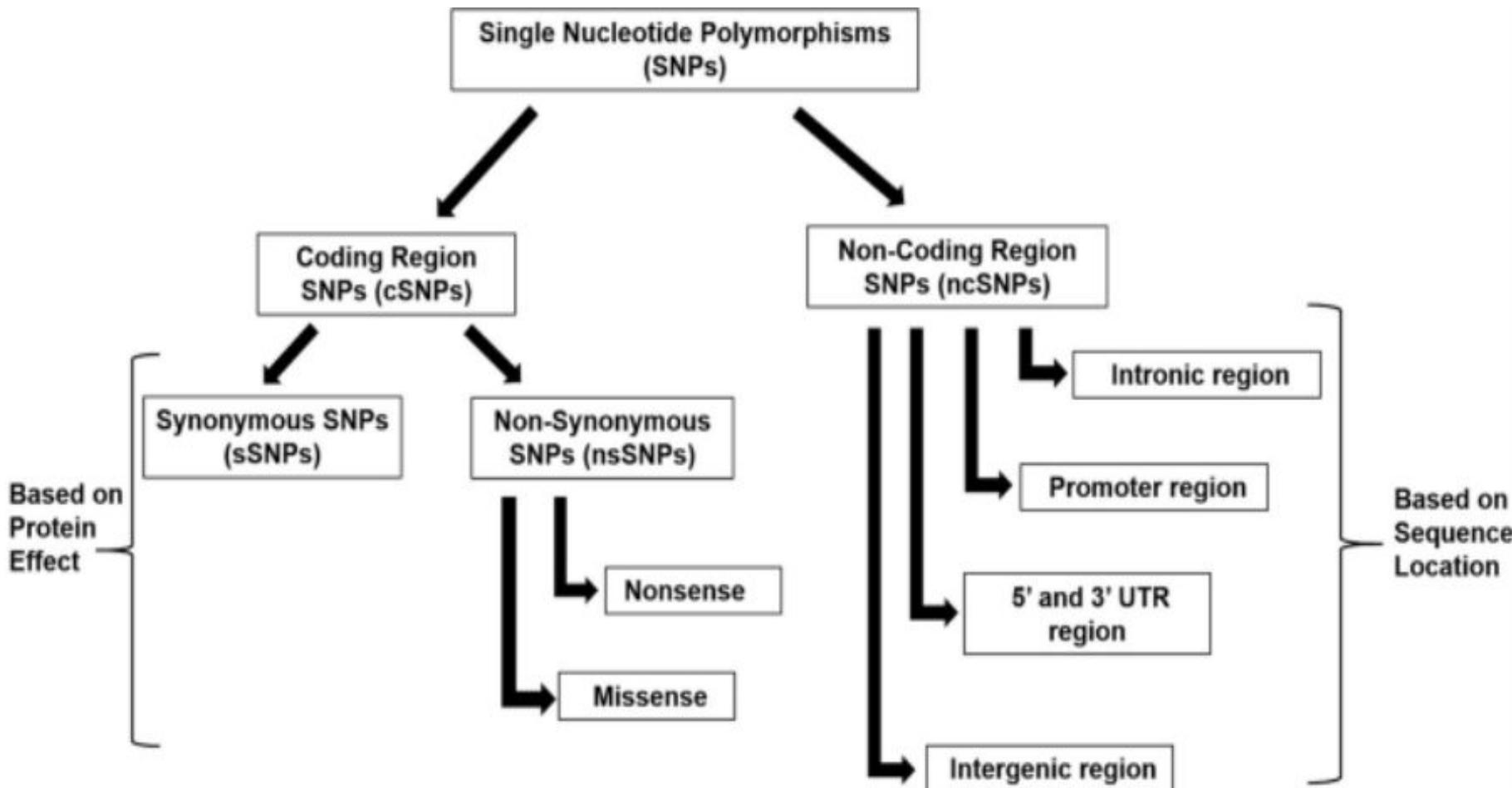
Different types of genomic variants



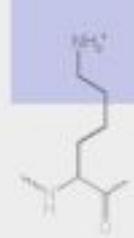
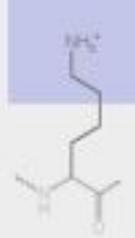
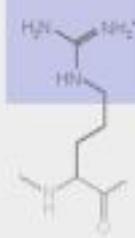
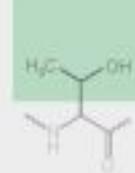
What variants look like in a genome browser



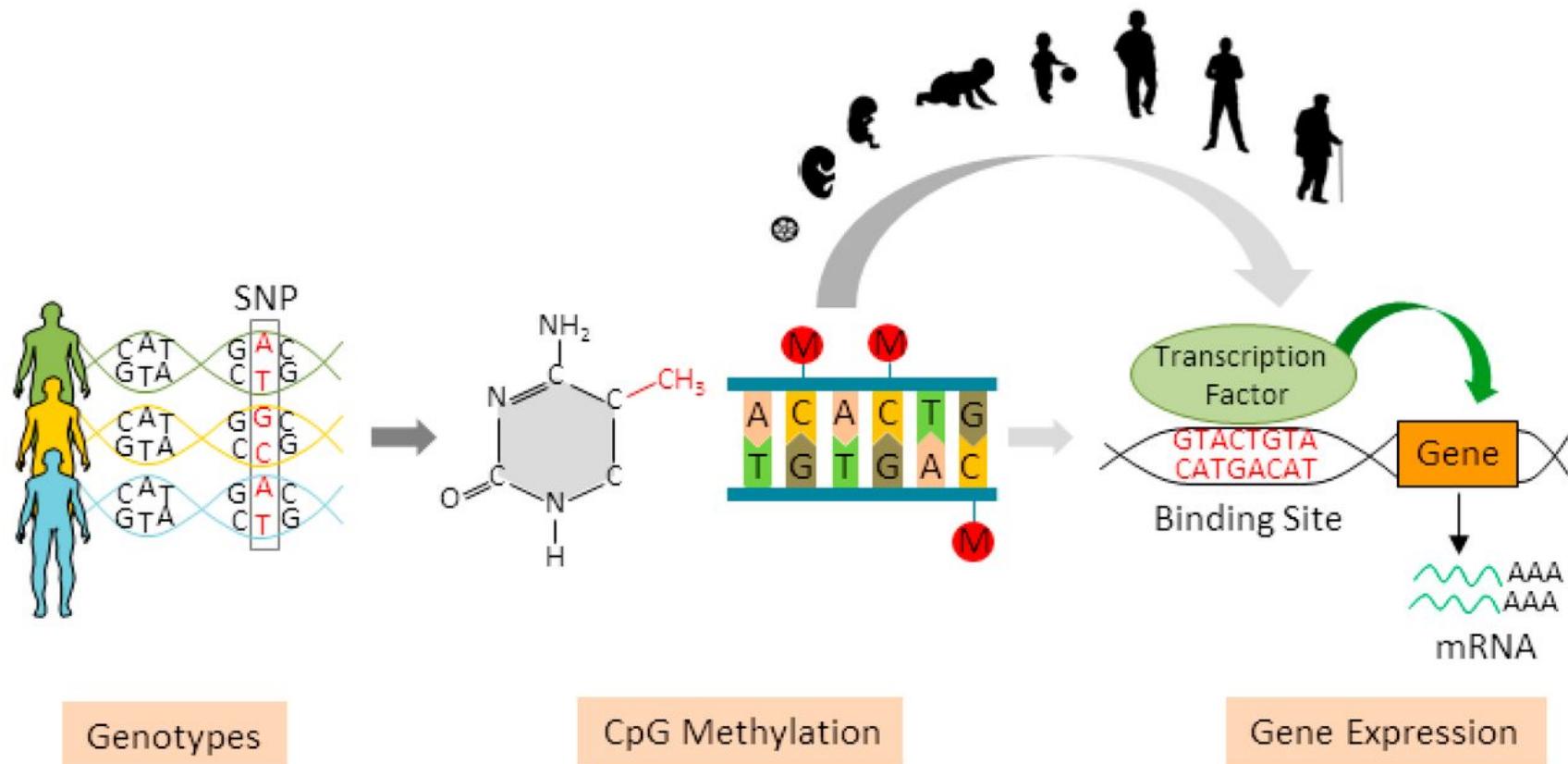
SNPs classifications



SNP/SNV

No mutation	Point mutations			non-conservative	
	Silent	Nonsense	Missense		
DNA level	TTC	TTT	ATC	TCC	TGC
mRNA level	AAG	AAA	UAG	AGG	ACG
protein level	Lys	Lys	STOP	Arg	Thr
					
					basic polar

Associations between SNPs, methylation patterns and gene expression of biological traits



Genotypes

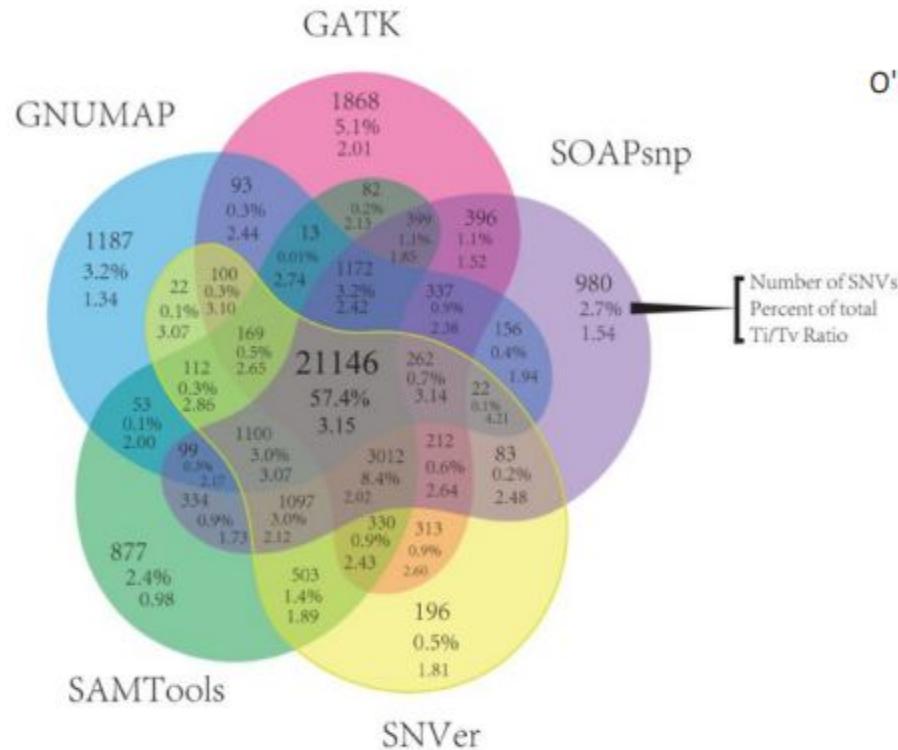
CpG Methylation

Gene Expression

Variant calling tools

	Germline	Somatic
SNPs/Indels	Haplotypecaller, FreeBayes, Strelka, DeepVariant, mpileup	Mutect2, FreeBayes, Strelka
CNV	CNVKit	ASCAT, CNVKit, Control-FREEC
Structural variants		Manta, TIDDIT
Microsatellite Instability	NA	MSIsensorpro

Variant callers are not concordant



O'Rawe et al., Genome medicine 2013

Mean single-nucleotide variants (SNV) concordance over 15 exomes between five alignment and variant-calling pipelines

Variants are reported in VCF (Variant Call Format)

Standardised format for storing the most prevalent types of sequence variations

Text file format in 2 parts : header and body.

```
##fileformat=VCFv4.2  
##fileDate=20090805  
##source=mvImputationProgramV3.1
```

Mandatory Header Lines

Extensible meta-data

Annotations may apply to the variant as a whole (the **INFO** column) or to each genotype (the **FORMAT** column). In addition to genotype, other commonly used annotations include genotype likelihoods, dbSNP membership, ancestral allele, read depth, mapping quality, and others.

##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">							
#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO
20	14370	rs6054257	ACG	A	29	PASS	NS=3;DP=14;AF=0.5;DB;H2
20	17330	.	T	A	3	q10	NS=3;DP=11;AF=0.017
20	1110696	rs6040355	A	G,GT	67	PASS	NS=2;DP=10;AF=0.333,0.667;AA=T;DB
20	1230237	.	T		47	PASS	GT:GQ:DP:HQ
20	1234567	microsat1	GTC	G,GTOT	50	PASS	NS=3;DP=9;AA=G

Annotations:

- Deletion (rs6054257)
- SNP (rs6040355)
- Other event (microsat1)
- Insertion (rs6054257)

Phased data (G and C above are on the same chromosome)

Annotations:

- FORMAT (rs6054257)
- FORMAT (rs6040355)
- FORMAT (microsat1)

VCF: Variant Call Format (2)

Types of variants

SNPs

Alignment	VCF representation		
ACGT	POS	REF	ALT
A T G _T	2	C	T

Insertions

Alignment	VCF representation		
AC-GT	POS	REF	ALT
A C TG _T	2	C	CT

Deletions

Alignment	VCF representation		
ACGT	POS	REF	ALT
A - _T	1	ACG	A

Complex events

Alignment	VCF representation		
ACGT	POS	REF	ALT
A - _{TT}	1	ACG	AT

Large structural variants

VCF representation
POS REF ALT INFO
100 T SVTYPE=DEL;END=300

VCF format supports CNVs and SVs

```
...
##INFO<ID=BKPTID,Number=.,Type=String>Description="ID of the assembled alternate allele in the assembly file"
##INFO<ID=CIEND,Number=2,Type=Integer>Description="Confidence interval around END for imprecise variants">
##INFO<ID=CIPOS,Number=2,Type=Integer>Description="Confidence interval around POS for imprecise variants">
##INFO<ID=END,Number=1,Type=Integer>Description="End position of the variant described in this record">
###INFO<ID=SVTYPE,Number=1,Type=String>Description="Type of structural variant">
##ALT=<ID=DEL,Description="Deletion">
##ALT=<ID=DUP,Description="Duplication">
##ALT=<ID=INS,Description="Insertion of novel sequence">
##ALT=<ID=INV,Description="Inversion">
##ALT=<ID=CNV,Description="Copy number variable region">
##FORMAT=<ID=GT,Number=1,Type=String>Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Float>Description="Genotype quality">
##FORMAT=<ID=CN,Number=1,Type=Integer>Description="Copy number genotype for imprecise events">
##FORMAT=<ID=CNQ,Number=1,Type=Float>Description="Copy number genotype quality for imprecise events">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001
1 2827694 rs2376870 CGTGGATGCGGGGAC C . PASS SVTYPE=DEL;END=2827708;HOMLEN=1;HOMSEQ=G;SVLEN=-14 GT:GQ
2 321682 . T <DEL> 6 PASS SVTYPE=DEL;END=321887;SVLEN=-205;CIPOS=-56,20;CIEND=-10,
3 12665100 . A <DUP> 14 PASS SVTYPE=DUP;END=12686200;SVLEN=21100;CIPOS=-500,500;CIEN
```



Symbolic allele in angle-bracketed ID

VCF header

- Lines that start with #
- Some mandatory lines : file format, column header.
- Optional header lines contain meta-data about annotations in the vcf body



Meta-data may vary a lot from a variant caller to another one!

INFO versus FORMAT :

- INFO = annotations on variant as a whole
- FORMAT = annotations that apply to each genotype

VCF representation of genotypes

Zygosity	VCF presentation
Heterozygous	0/1, 1/2, 0/2, ...
Homozygous <ul style="list-style-type: none">• Reference• Alternate	0/0 1/1, 2/2, 3/3, ...
Missing	./0, ./1, ./., ...

VCF specification versions

Changes between VCFv4.1 and VCFv4.2:

- Information field format: adding source and version as recommended fields.
- INFO field can have one value for each possible allele (code R).
- For all of the ##INFO, ##FORMAT, ##FILTER, and ##ALT metainformation, extra fields can be included after the default fields.
- Alternate base (ALT) can include *: missing due to a upstream deletion.
- Quality scores, a sentence removed: *High QUAL scores indicate high confidence calls. Although traditionally people use integer phred scores, this field is permitted to be a floating point to enable higher resolution for low confidence calls if desired.*
- Examples changed a bit.

Changes between VCFv4.2 and VCFv4.3 :

- VCF compliant implementations must support both LF and CR+LF newline conventions
- INFO and FORMAT tag names must match the regular expression ^[A-Za-z][0-9A-Za-z .]*\$
- Spaces are allowed in INFO field values
- Characters with special meaning (such as ';' in INFO, ':' in FORMAT, and '%' in both) can be encoded using the percent encoding (see Section 1.2) • The character encoding of VCF files is UTF-8. 35
- The SAMPLE field can contain optional DOI URL for the source data file
- Introduced ##META header lines for defining phenotype metadata
- New reserved tag "CNP" analogous to "GP" was added. Both CNP and GP use 0 to 1 encoding, which is a change from previous phred-scaled GP.
- In order for VCF and BCF to have the same expressive power, we state explicitly that Integers and Floats are 32-bit numbers. Integers are signed.
- We state explicitly that zero length strings are not allowed, this includes the CHROM and ID column, INFO IDs, FILTER IDs and FORMAT IDs. Meta-information lines can be in any order, with the exception of ##fileformat which must come first.
- All header lines of the form ##key= must have an ID value that is unique for a given value of "key". All header lines whose value starts with "<" must have an ID field. Therefore, also ##PEDIGREE newly requires a unique ID.
- We state explicitly that duplicate IDs, FILTER, INFO or FORMAT keys are not valid.
- A section about gVCF was added, introduced the <*> symbolic allele.

Copy number variations (CNV)

Copy Number Variant

Reference Genome

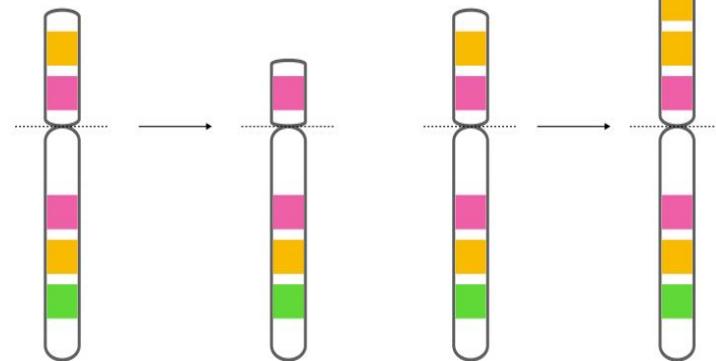


Example Sequence



Copy Number Variation

© Genetic Education Inc.



Copy Number Variant: Reads

Example Sequence



Reference Genome



Reads

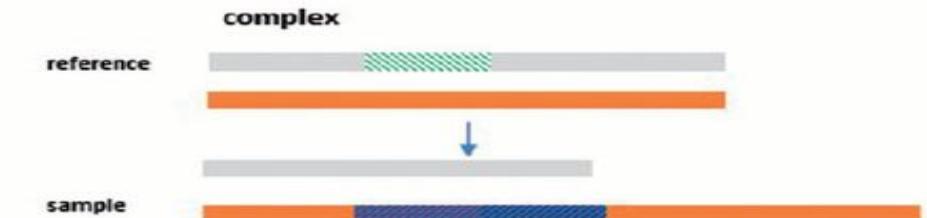
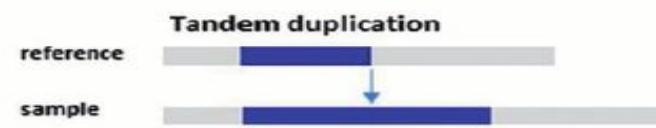
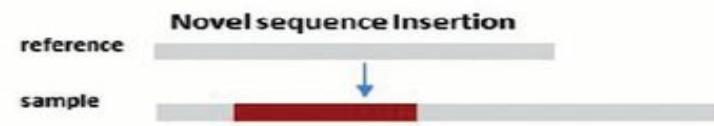
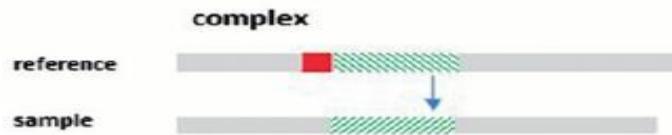
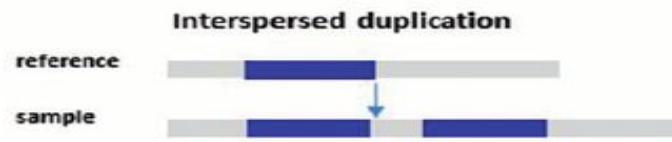


because the copied sequence maps to the same area on the reference genome, the reads spike in the area where the copy number variant occurred

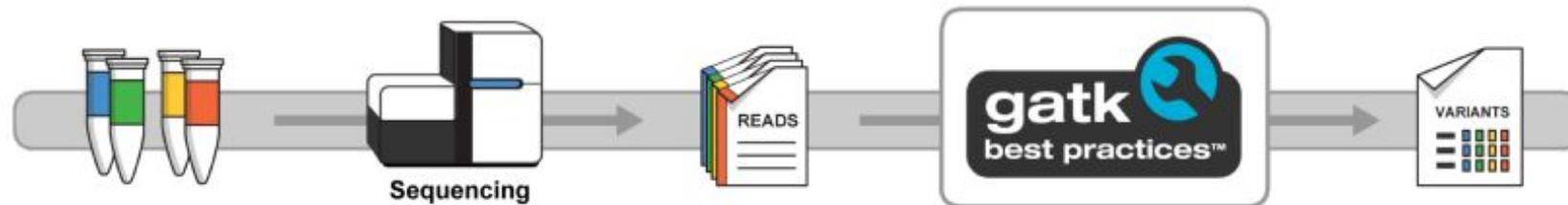
Deletion

Duplication

Structural variants

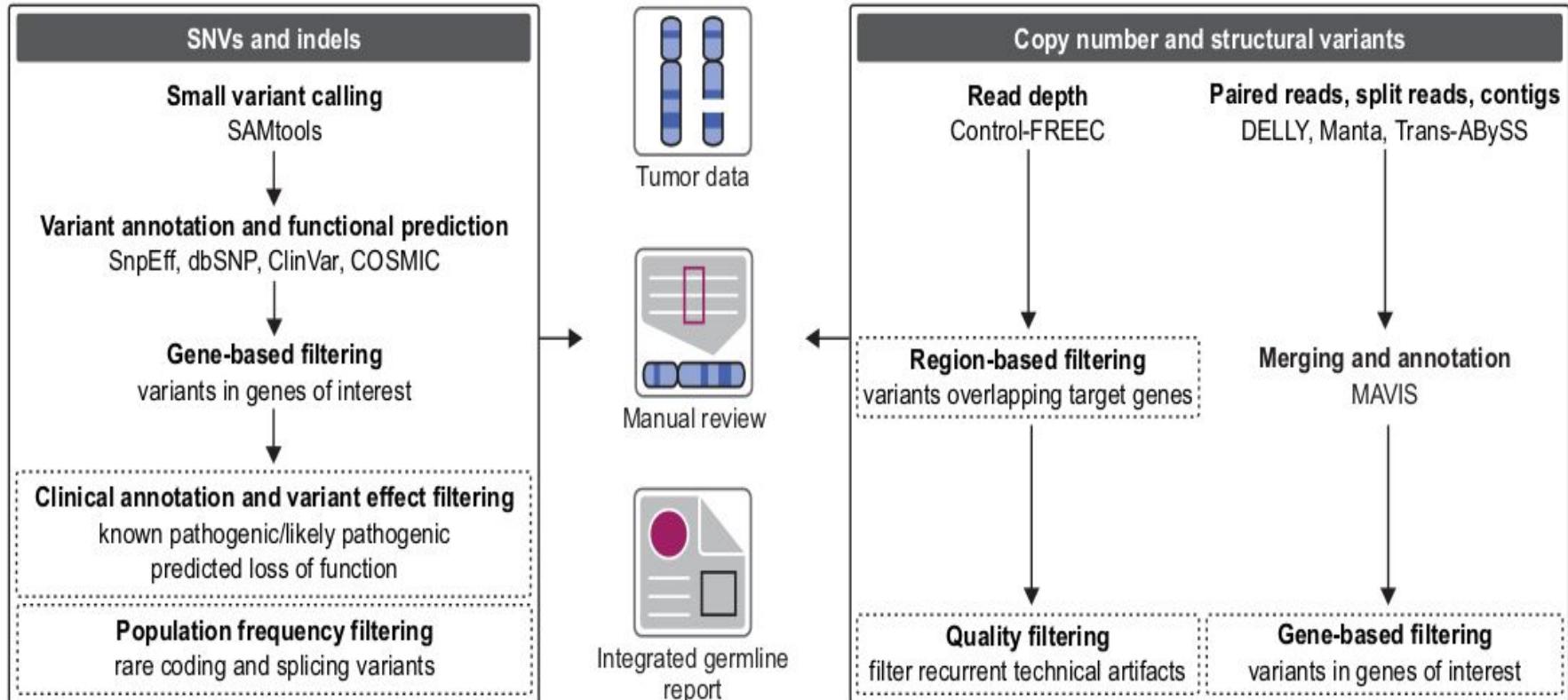


Workflows for all major variant classes

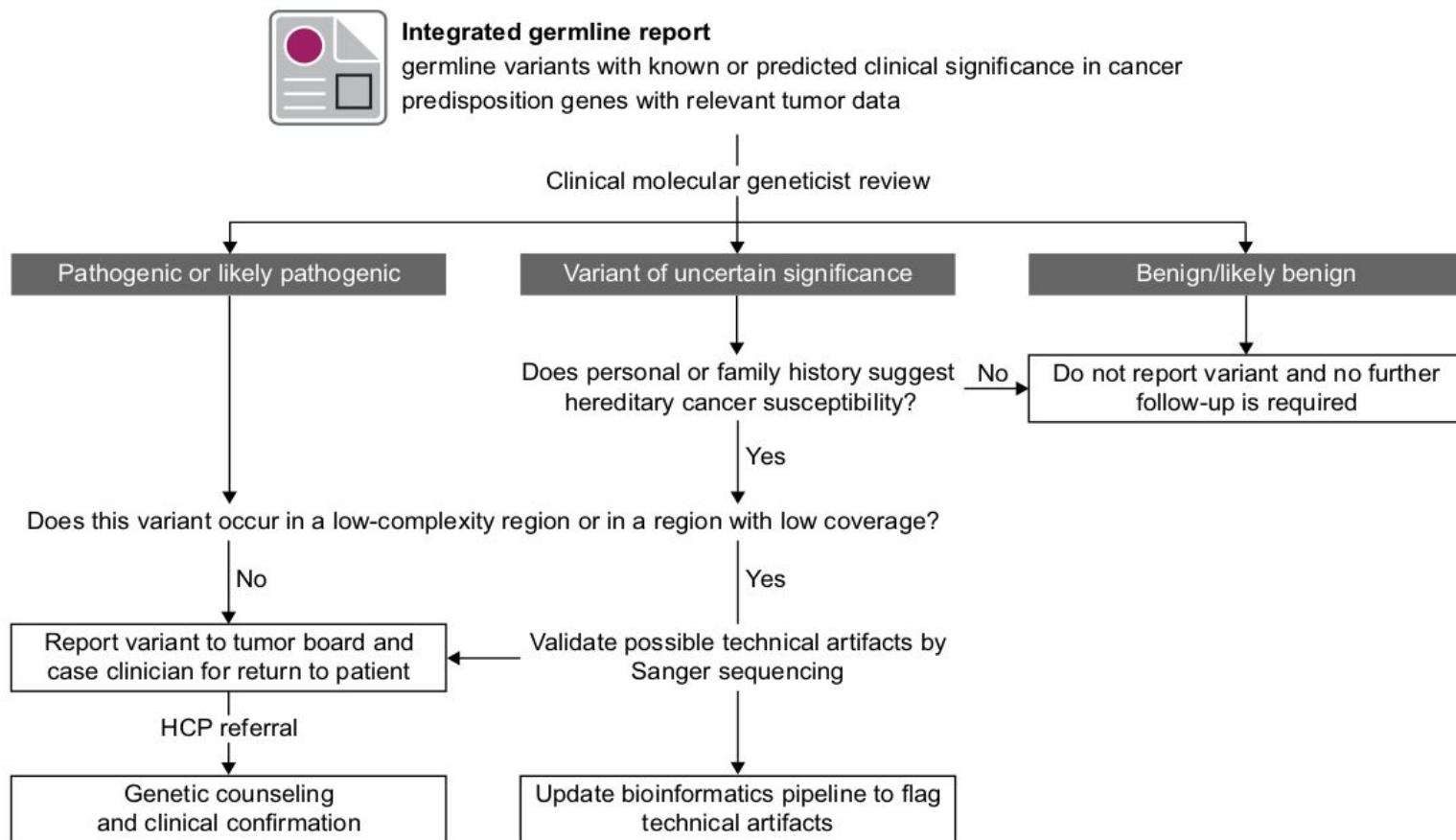


	GERMLINE	SOMATIC
SNPs & INDELs	HaplotypeCaller GVCF	Mutect2
Copy Number	GATK gCNV	GATK CNV + aCNV
Structure Variation	GATK SVDISCOVERY (beta)	(planned)

Workflow from variant calling to integrated report



Standard procedure for the review, reporting, and clinical translation of germline variants



Normal - Tumor paired variants calling

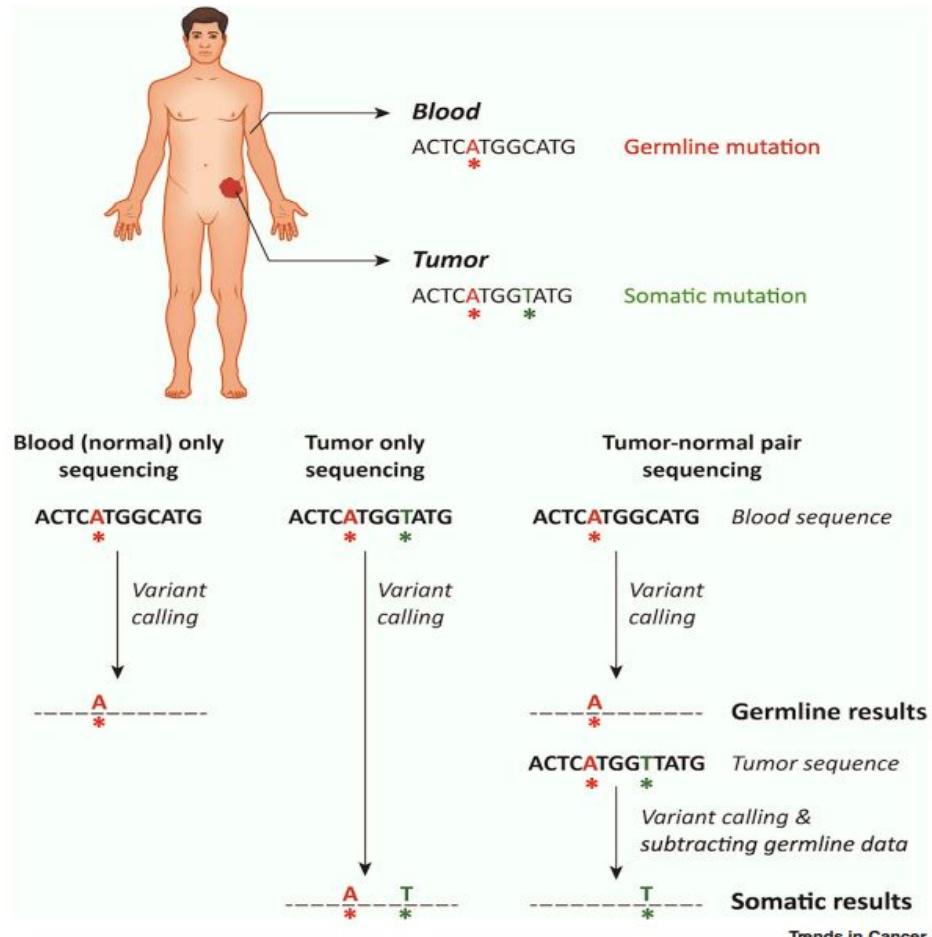
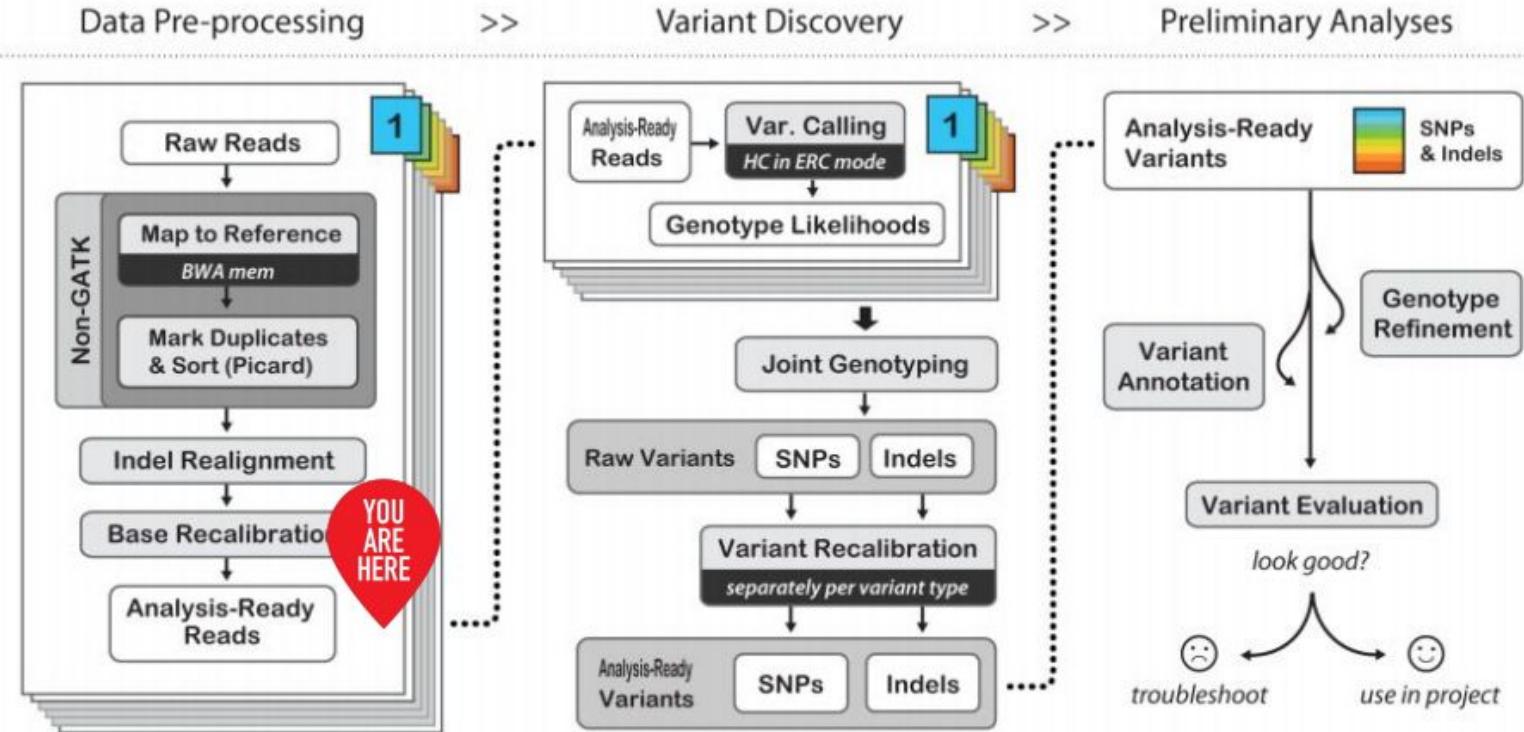


Figure 1. Mutations Reported in Blood-Only, Tumor-Only, and Paired Tumor-Normal Sequencing.

Workflow continues

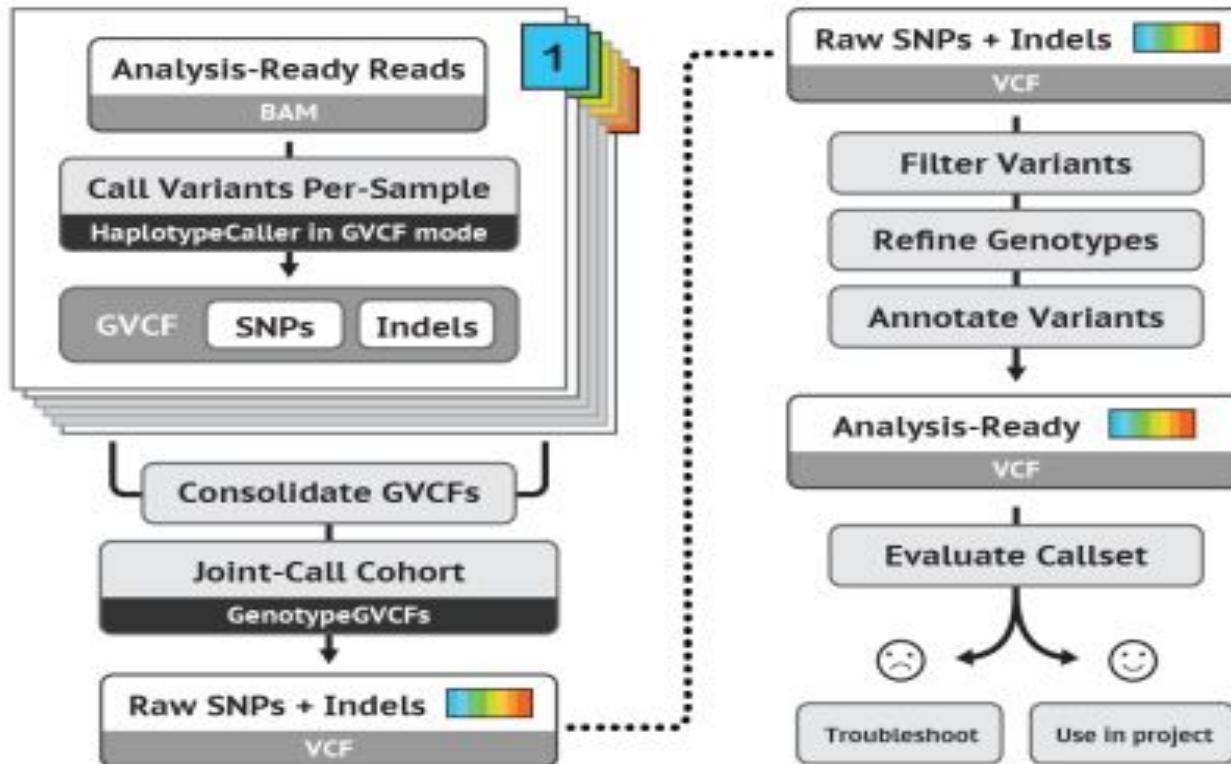


Ready for variant calling!!

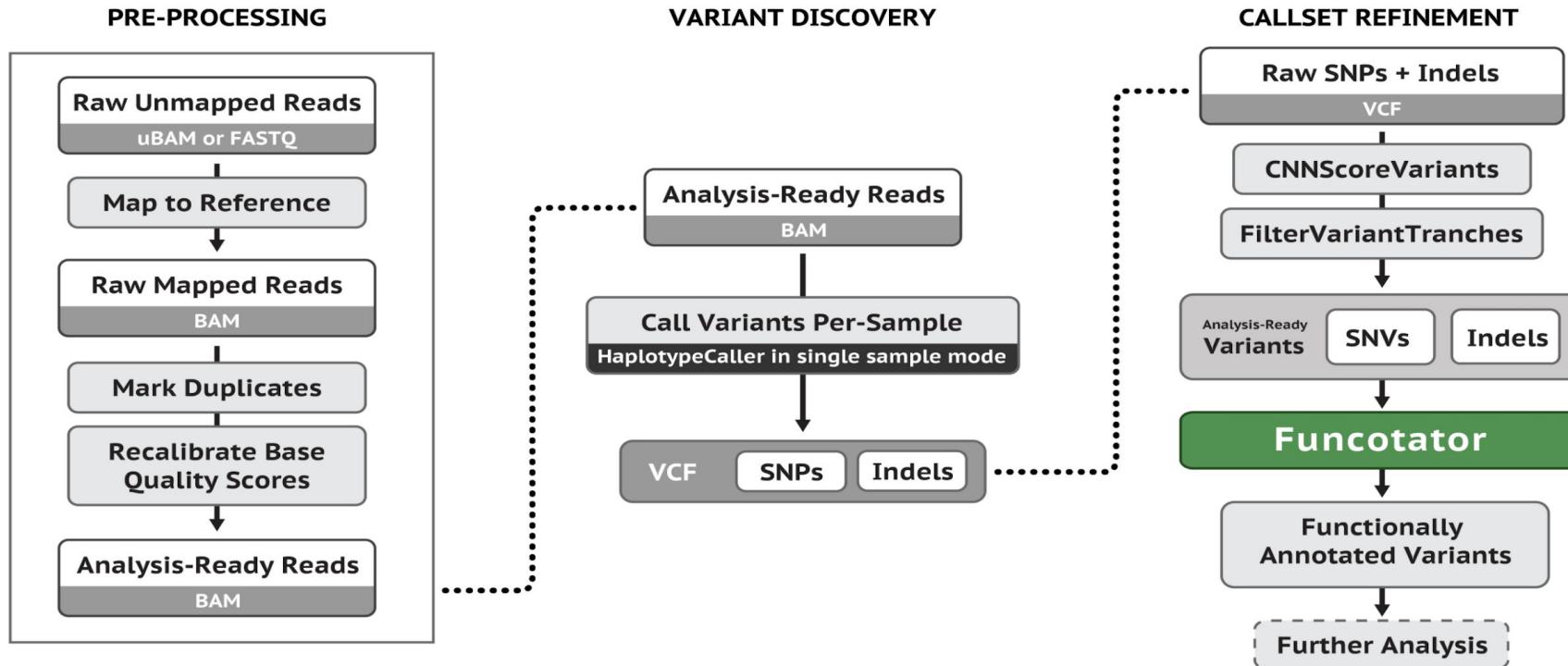
GERMLINE SNPs & INDELS

<http://software.broadinstitute.org/gatk/>

Main steps for Germline Cohort Data



Main steps for Germline Single-Sample Data

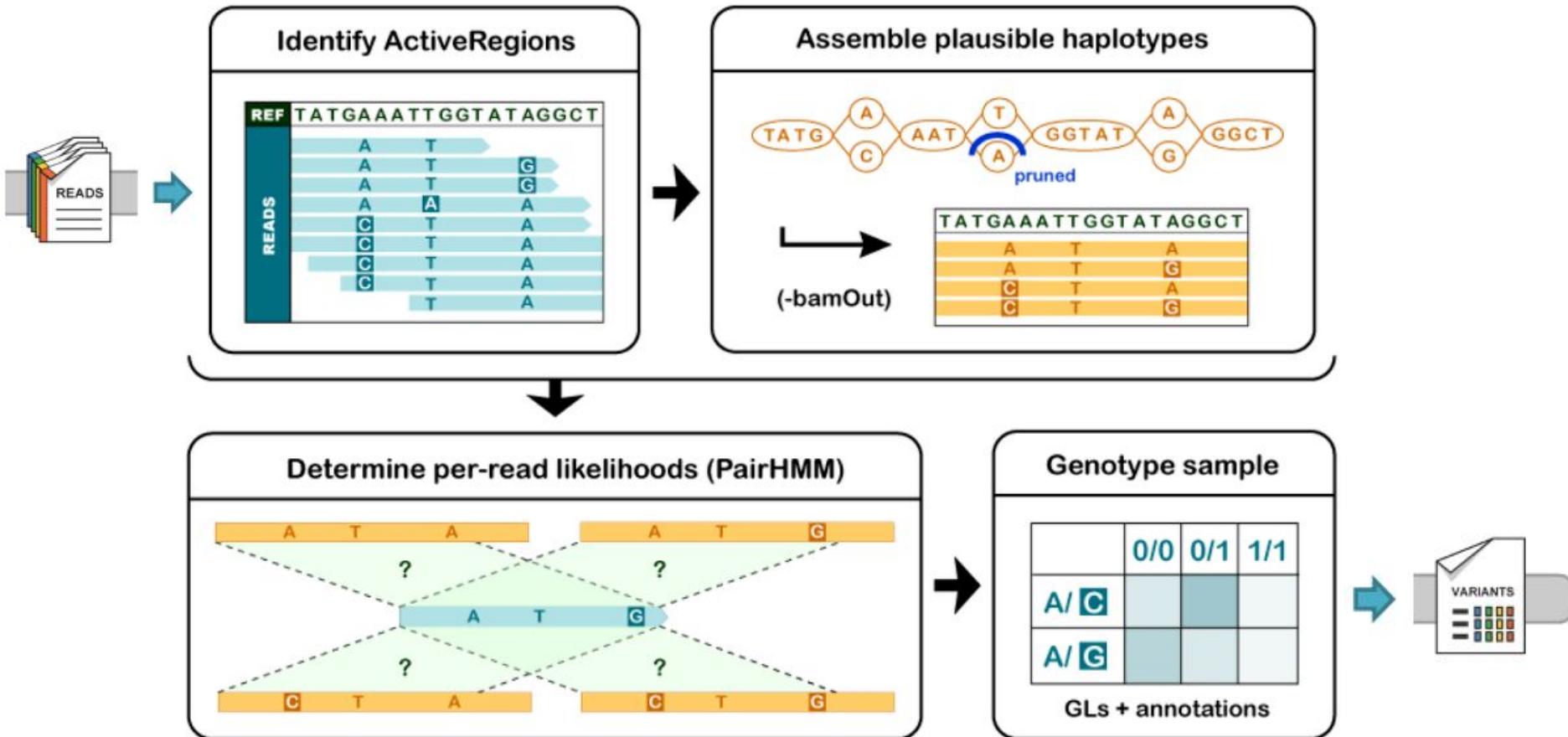


Variant calling with HaplotypeCaller

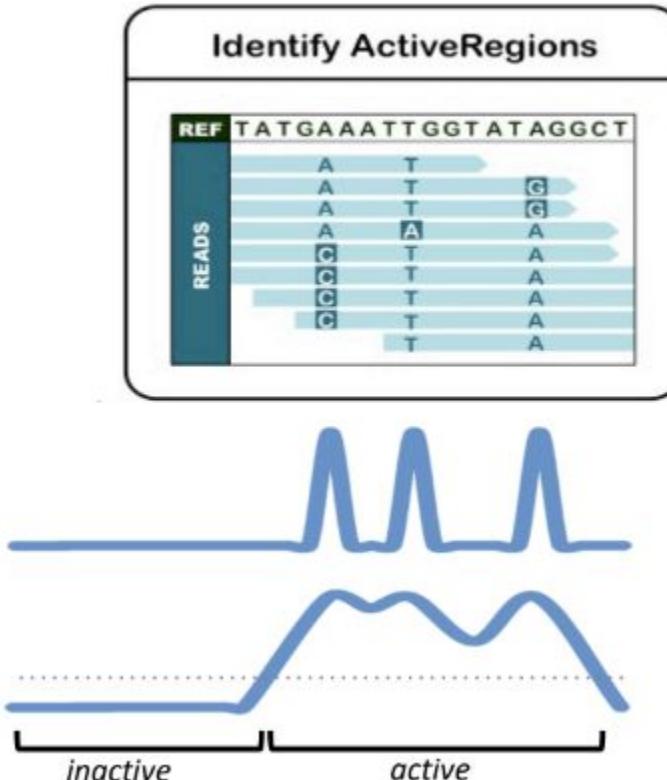
Basic operation and algorithm

<http://software.broadinstitute.org/gatk/>

GATK HaplotypeCaller calls germline short variants



1. Define Active Regions

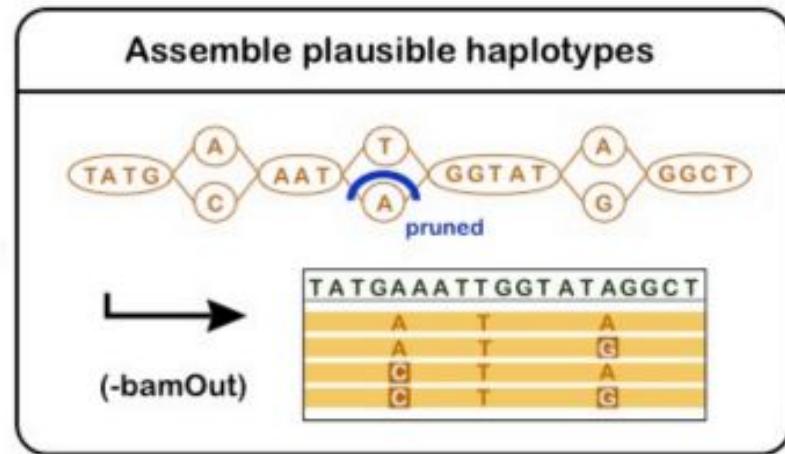


- Sliding window along the reference
- Count mismatches, indels and soft-clips
- Measure of entropy

Trim and continue with ActiveRegions over threshold

2. Assemble plausible haplotypes

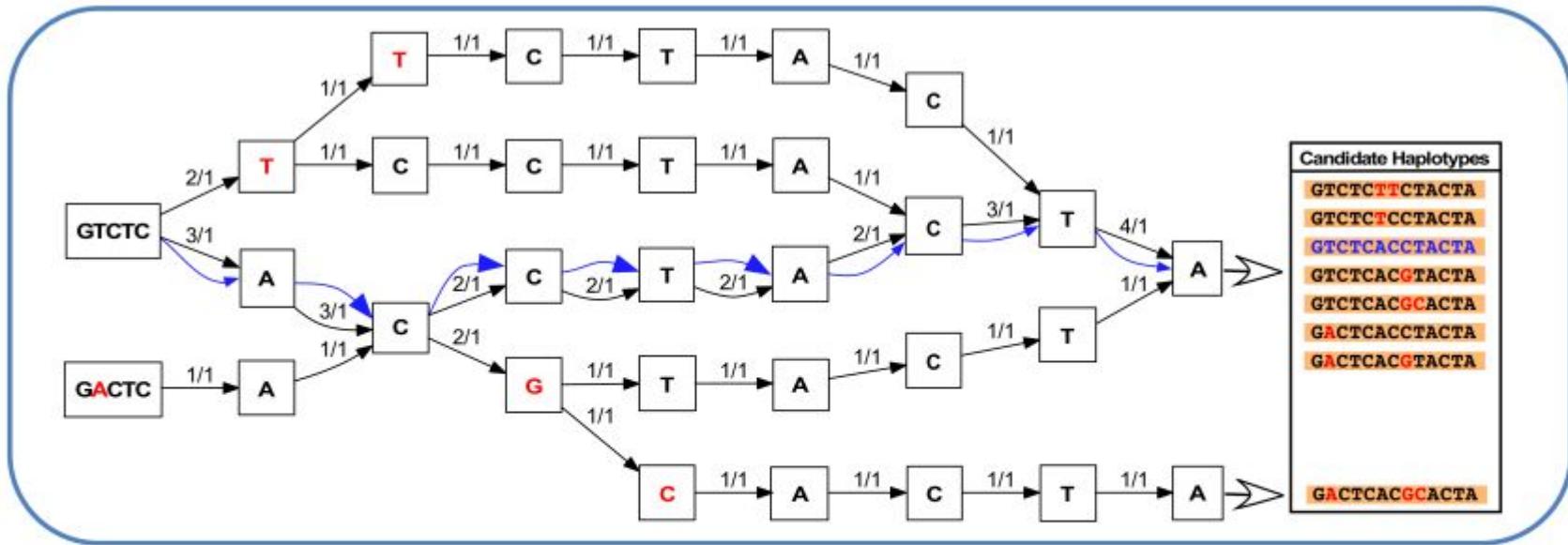
- Local realignment via graph assembly
- Traverse graph to collect most likely haplotypes
- Align haplotypes to reference using Smith-Waterman



Likely haplotypes + candidate variant sites

Can make HC output the reassembled reads and selected haplotypes using the -bamOut parameter

Example HaplotypeCaller assembly graph



- Ignore previous alignments
- Graph consists of every possible sequence combination based on reads
- Count reads that support paths

Graph assembly recovers indels and removes artifacts



Showing 100bp region starting at 10:96,825,862 for NA12878

Resolves complexity caused by mapper limitations

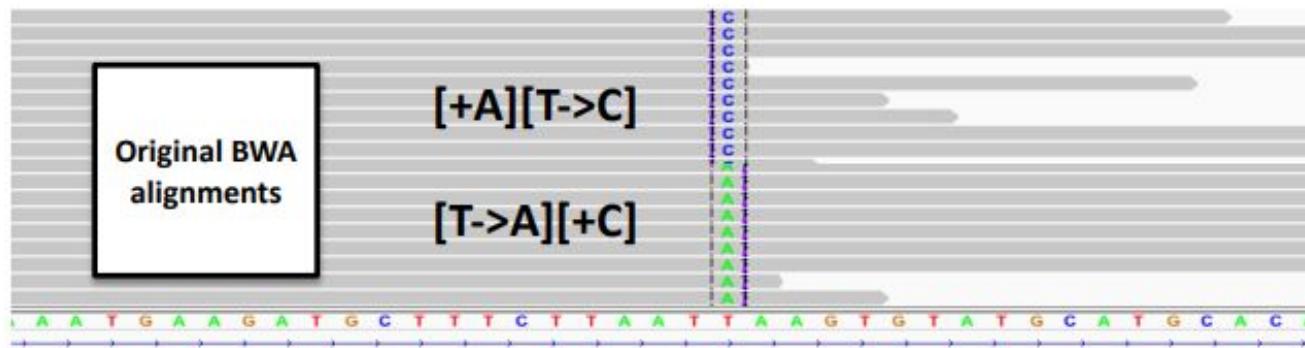
Reference

Consensus

Reads

C T T A A T
A A G T G T
T
A C

- Mapper can represent two different ways, at random:



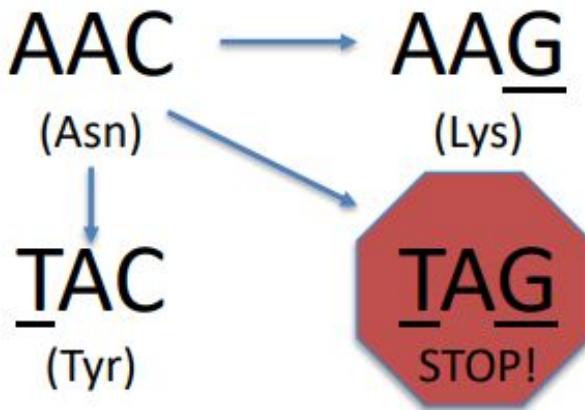
- HaplotypeCaller will settle on one representation -> cleaner output call

Functional implications of variant phasing

Two SNPs in the same codon: A > T and C > G

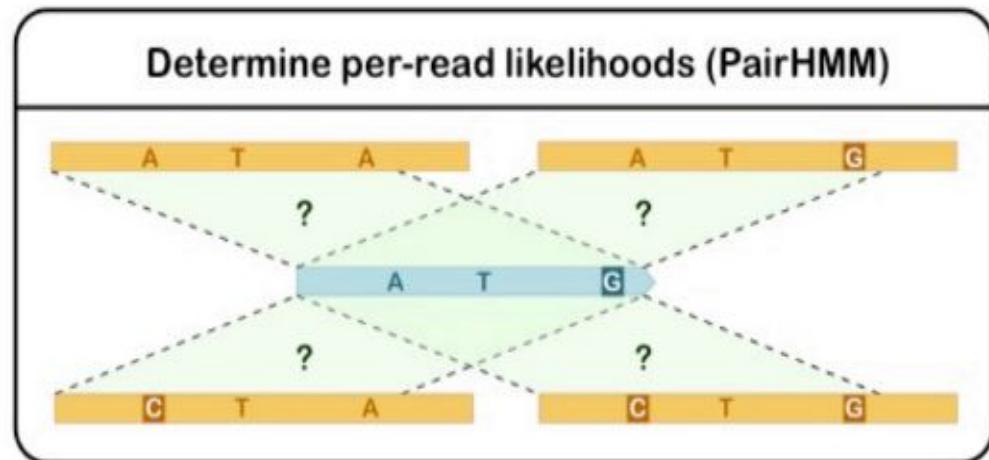
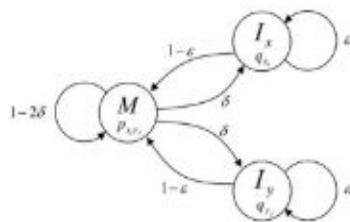
In trans – two copies, each with a missense mutation

In cis – one functional copy and one loss of function!



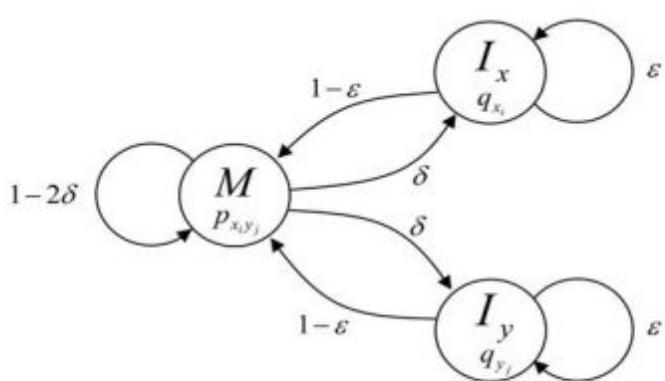
3. Score haplotypes using PairHMM

- PairHMM* aligns each read to each haplotype
- Uses base qualities as the estimate of error



Likelihoods of the haplotypes given reads

PairHMM uses base qualities to score alignments



State
 (M) Match
 (I_x) Insertion
 (I_y) Deletion

Transition probabilities
 (ε) = Gap continuation
 (δ) = Gap open penalty
 $(1 - \varepsilon)$ = Base precedes an insertion or a deletion
 $(1 - 2\delta)$ = Base matches and continues

Haplotypes

Reads $\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & & & A_{2n} \\ \vdots & & & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{bmatrix}$ A_{ij} = probability of haplotype-read pair

→ Matrix contains likelihoods of the haplotypes given the reads

4. Genotype each sample at each potential variant site

- Determine most likely combination of allele(s) for each site
- Based on allele likelihoods (from PairHMM)
- Apply Bayes' theorem with ploidy assumption*

$$P(G_i | R) = \frac{P(R|G_i)P(G_i)}{\sum_k P(R|G_k)P(G_k)} \propto L(R|G_i)P(G_i)$$

$$L(R|G_i) = \prod_j \left(\frac{L(R_j | H_1)}{2} + \frac{L(R_j | H_2)}{2} \right) \quad G_i = H_1H_2 \text{ for diploids}$$

$L(R_j | H_i)$ Read-haplotype likelihoods

Genotype sample			
	0/0	0/1	1/1
A/ C			
A/ G			
GLs + annotations			



Genotype calls

* Default is diploid; can set desired ploidy in command line

Finally, Bayesian math for genotype probability

Posterior probability of the genotype given the reads

$$P(G_i | R) = \frac{P(R|G_i)P(G_i)}{\sum_k P(R|G_k)P(G_k)} \propto \frac{Likelihood}{Genotype\ prior\ (constant)}$$
$$L(R|G_i) = \prod_j \left(\frac{L(R_j | H_1)}{2} + \frac{L(R_j | H_2)}{2} \right)$$

$L(R_j | H_i)$ Read-haplotype likelihoods

Plug in the numbers!

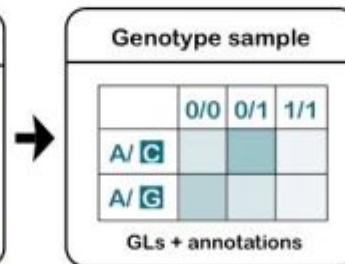
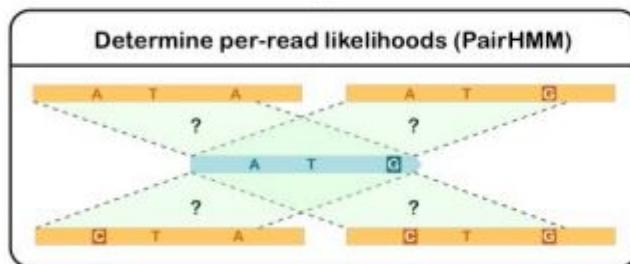
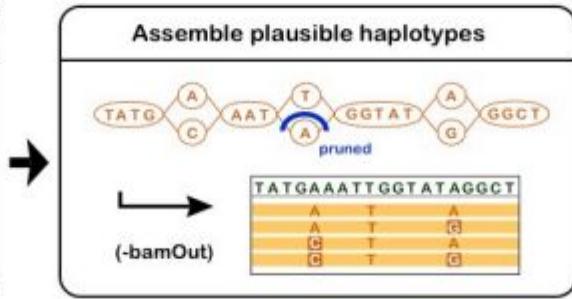
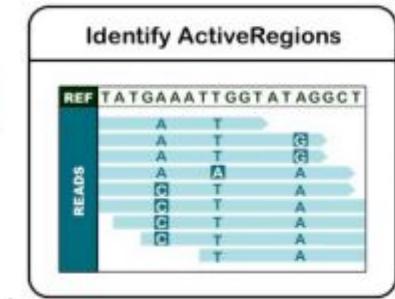
Reads	Alleles	
	C	A
1	0.10	0.06
2	0.10	0.09
3	0.12	0.05



Determines the most likely genotype of the sample at each event in the haplotypes

HaplotypeCaller recap: reads in / variants out

BAM

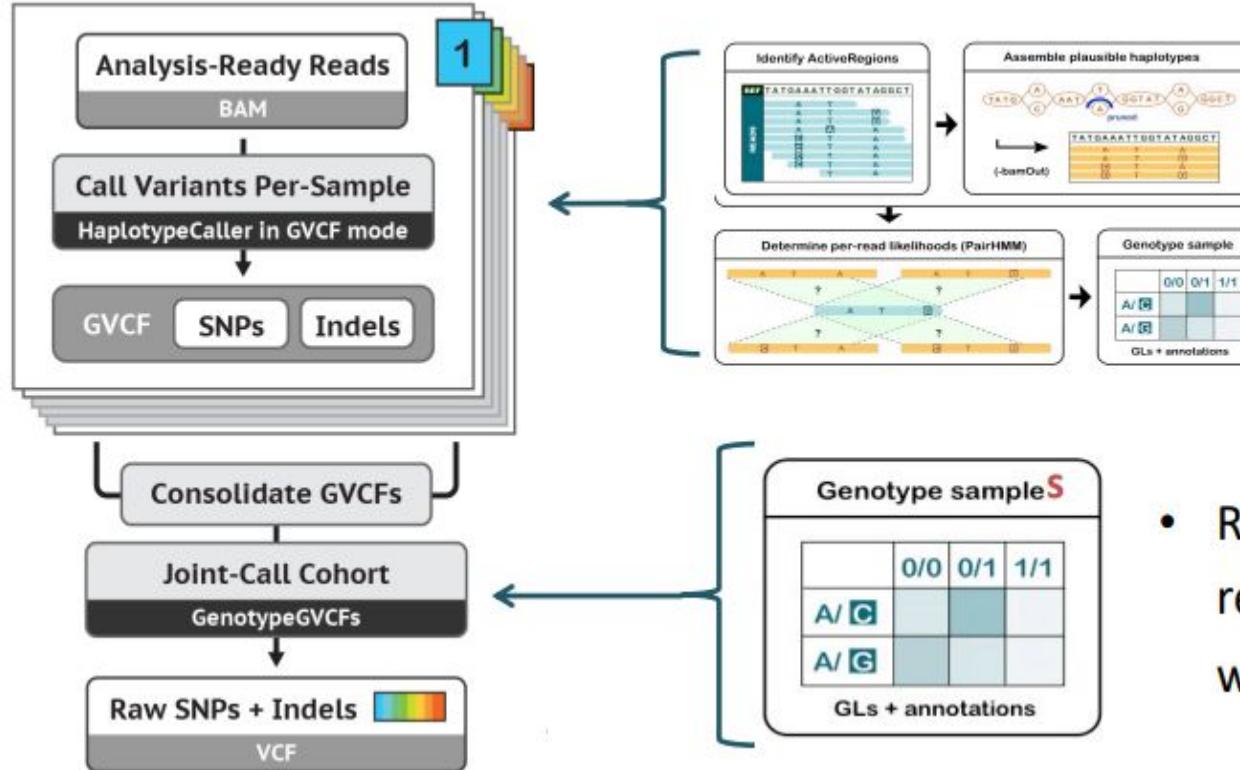


This is all you
need for a
single sample or
traditional multi-
sample analysis



VCF & index

For scalable analysis: emit GVCF + add joint calling step



- Run HC in **GVCF mode** to emit GVCF
- Run GenotypeGVCFs to re-genotype samples with **multi-sample model**

Running HaplotypeCaller

Basic mode (no GVCF):

```
gatk HaplotypeCaller \
    -R reference.fasta \
    -I preprocessed_reads.bam \
    -O germline_variants.vcf
```

To produce a block-compressed GVCF, substitute output filename and add:

```
-O germline_variants.g.vcf \
-ERC GVCF
```

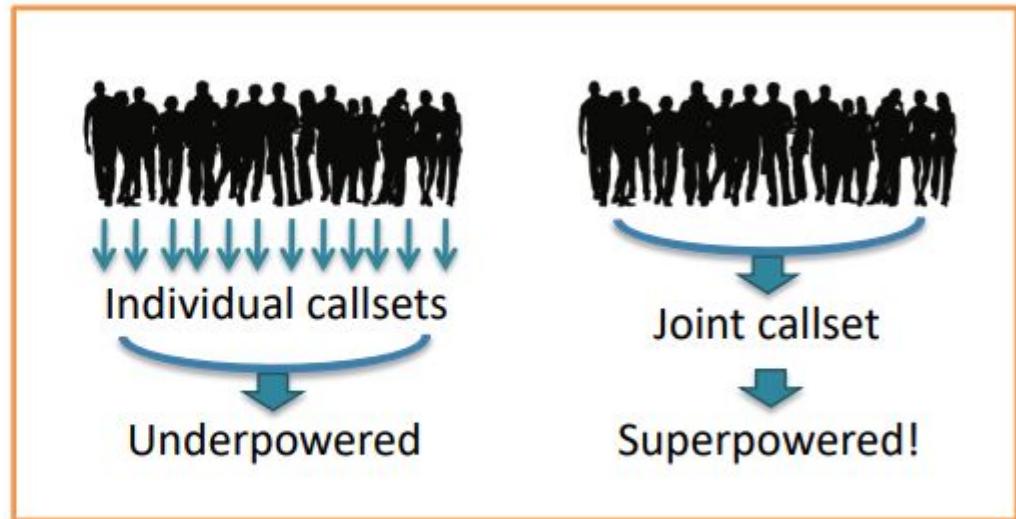
Joint variant calling

GVCF-based workflow

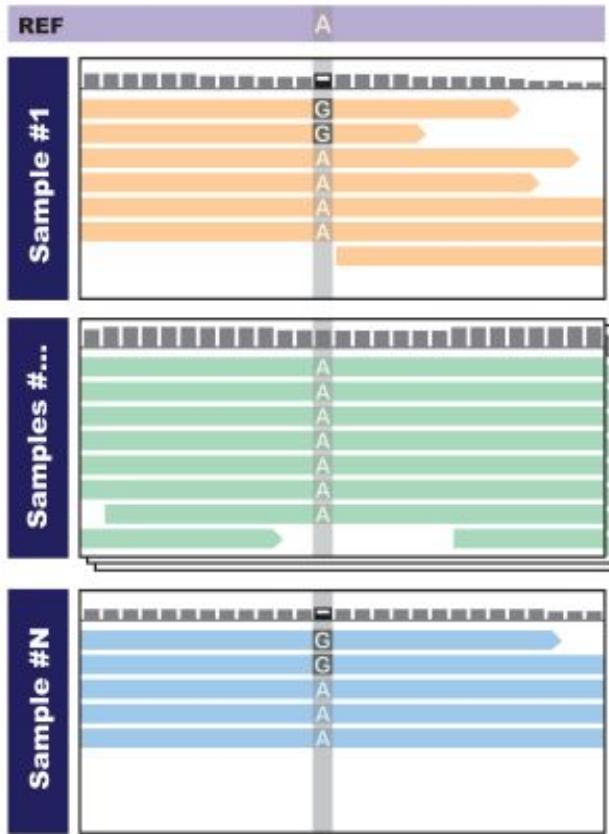
<http://software.broadinstitute.org/gatk/>

Joint analysis empowers discovery

- Single genome in isolation: almost never useful
- Family or population data add valuable information
 - rarity of variants
 - *de novo* mutations
 - ethnic background

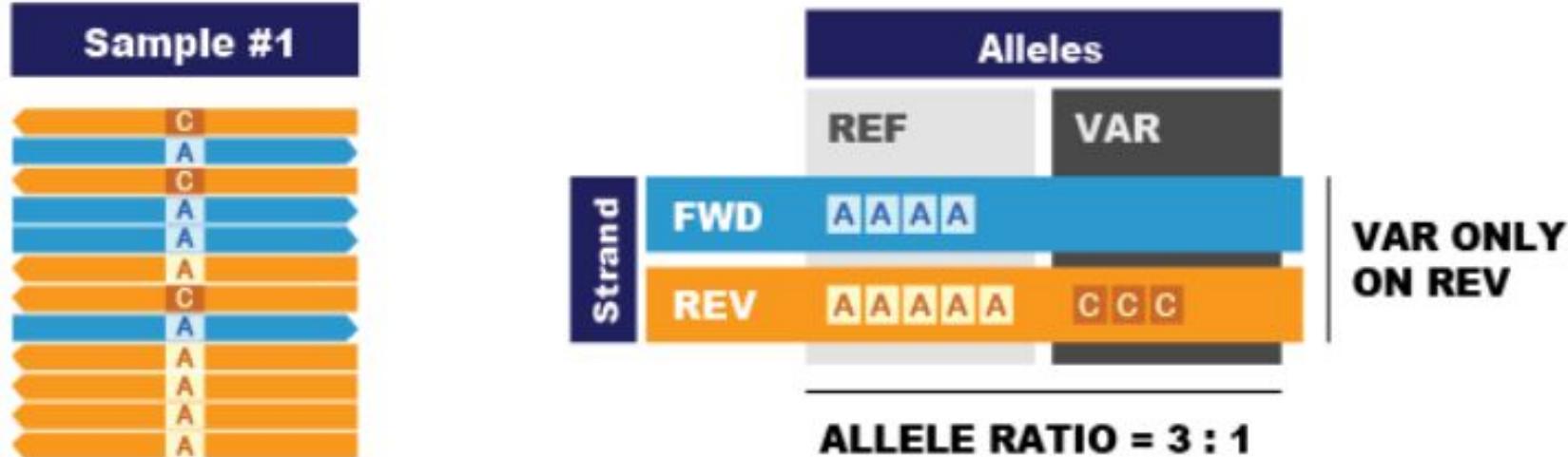


Discovery is empowered at difficult sites



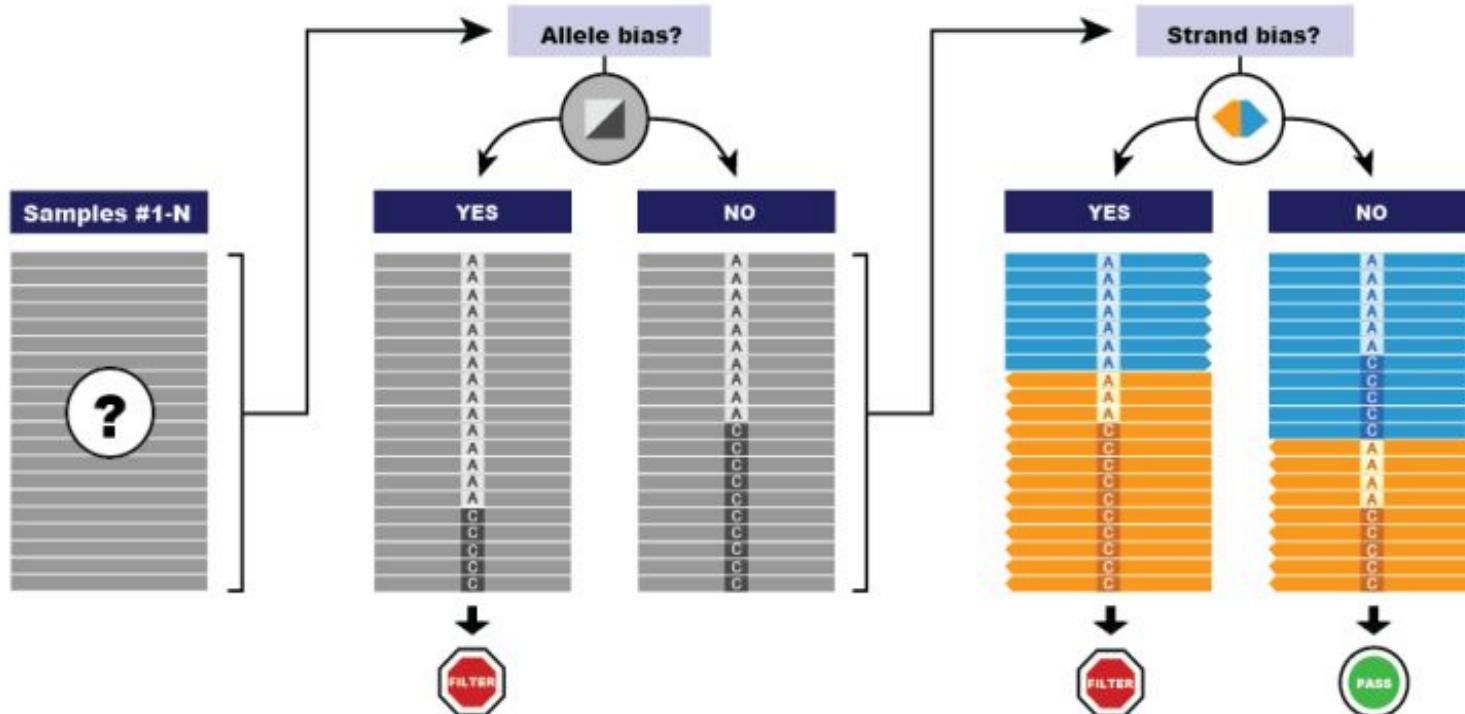
- Sample #1 or Sample #N alone:
 - **weak evidence for variant**
 - **may miss calling the variant**
- Both samples seen together:
 - **unlikely to be artifact**
 - **call the variant more confidently**

Joint analysis helps resolve bias issues (1)



Single sample showing strand and allelic biases – would you call it?

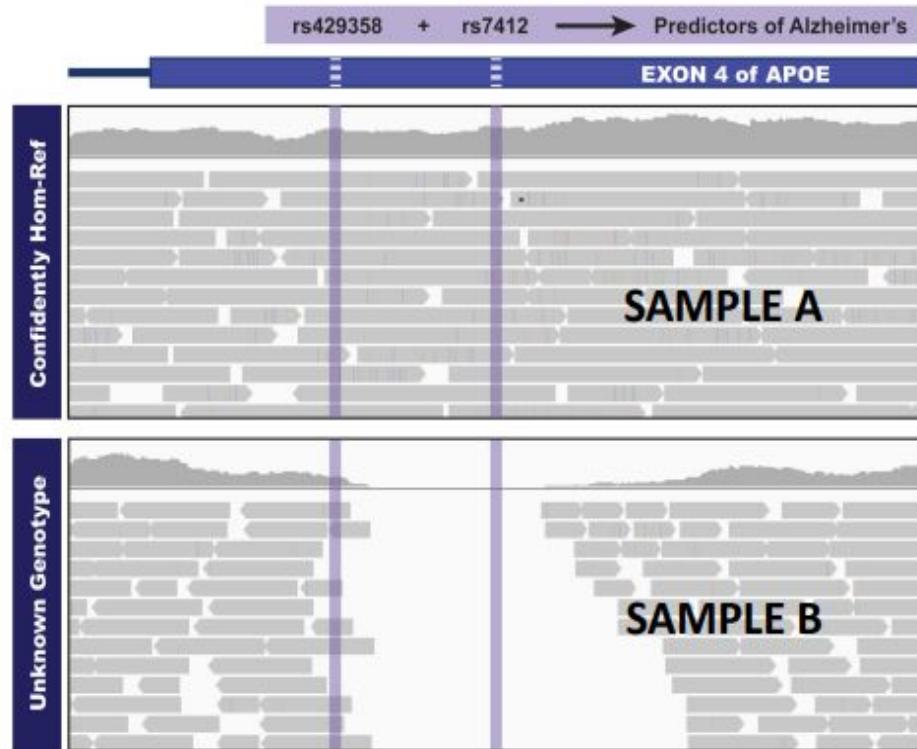
Joint analysis helps resolve bias issues (2)



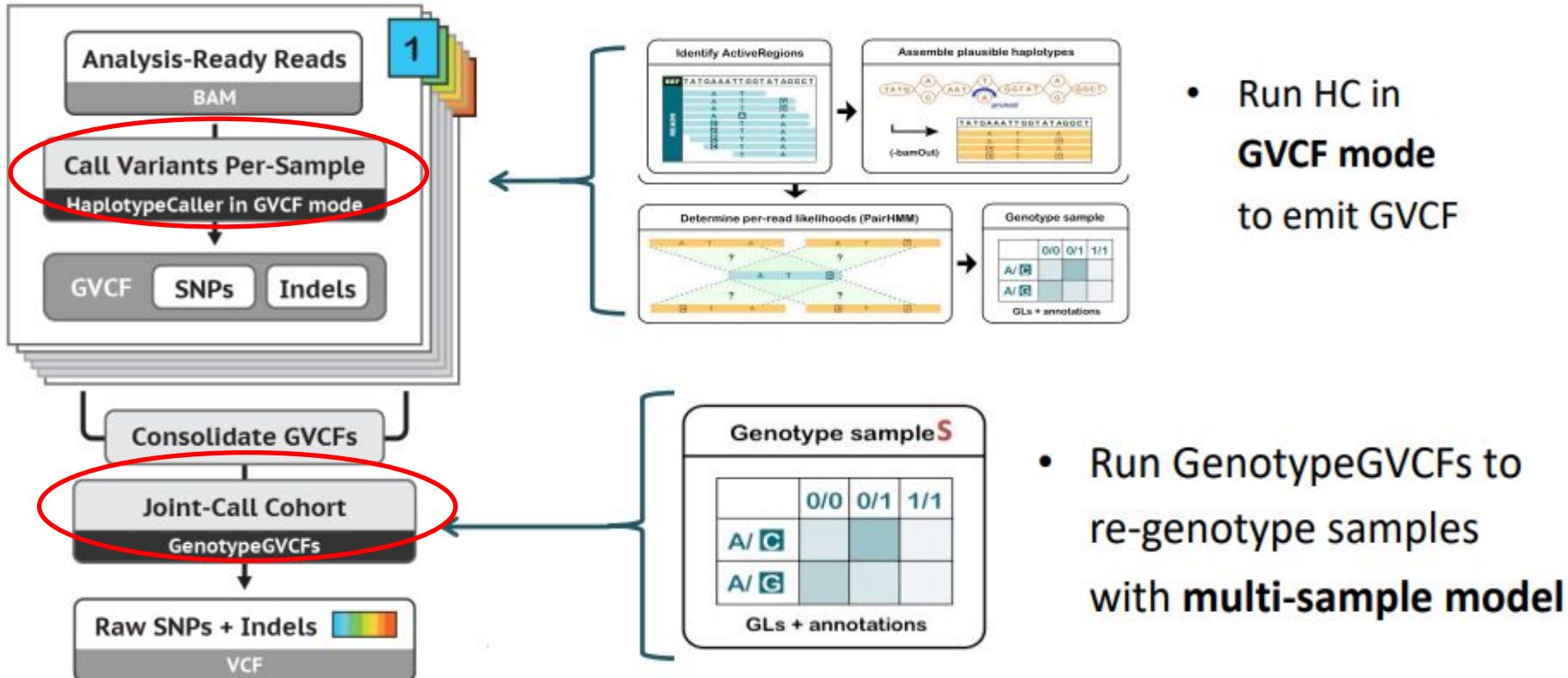
Decision process using evidence from multiple samples to filter out sites showing systematic biases

Gather full information at all sites of interest

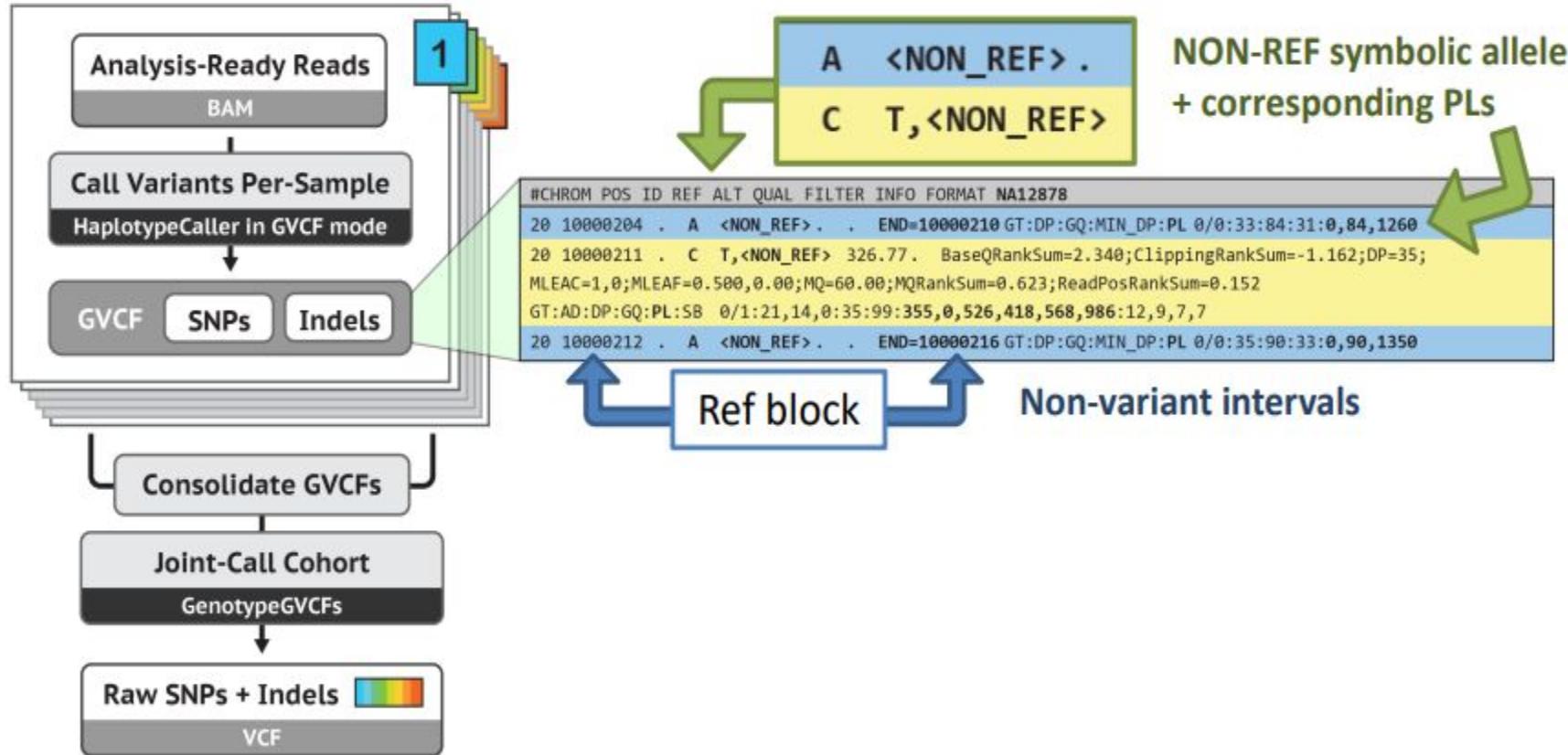
- **Analyzed individually:**
 - No call for either sample
 - Very different reasons!
- **In joint analysis with other samples:**
 - Hom-ref call and no-call genotypes emitted



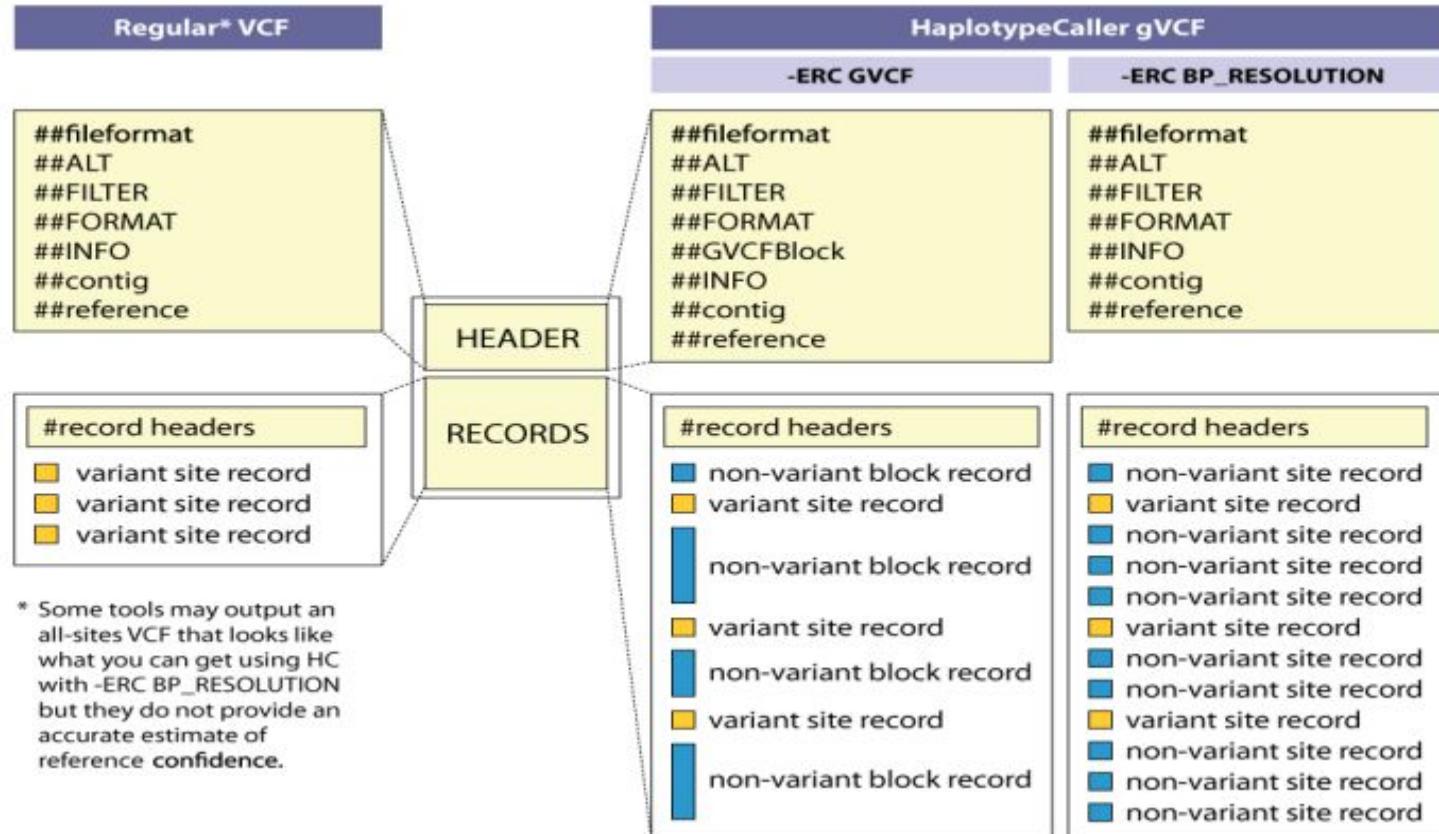
Joint calling implemented as a two-step process for scalability



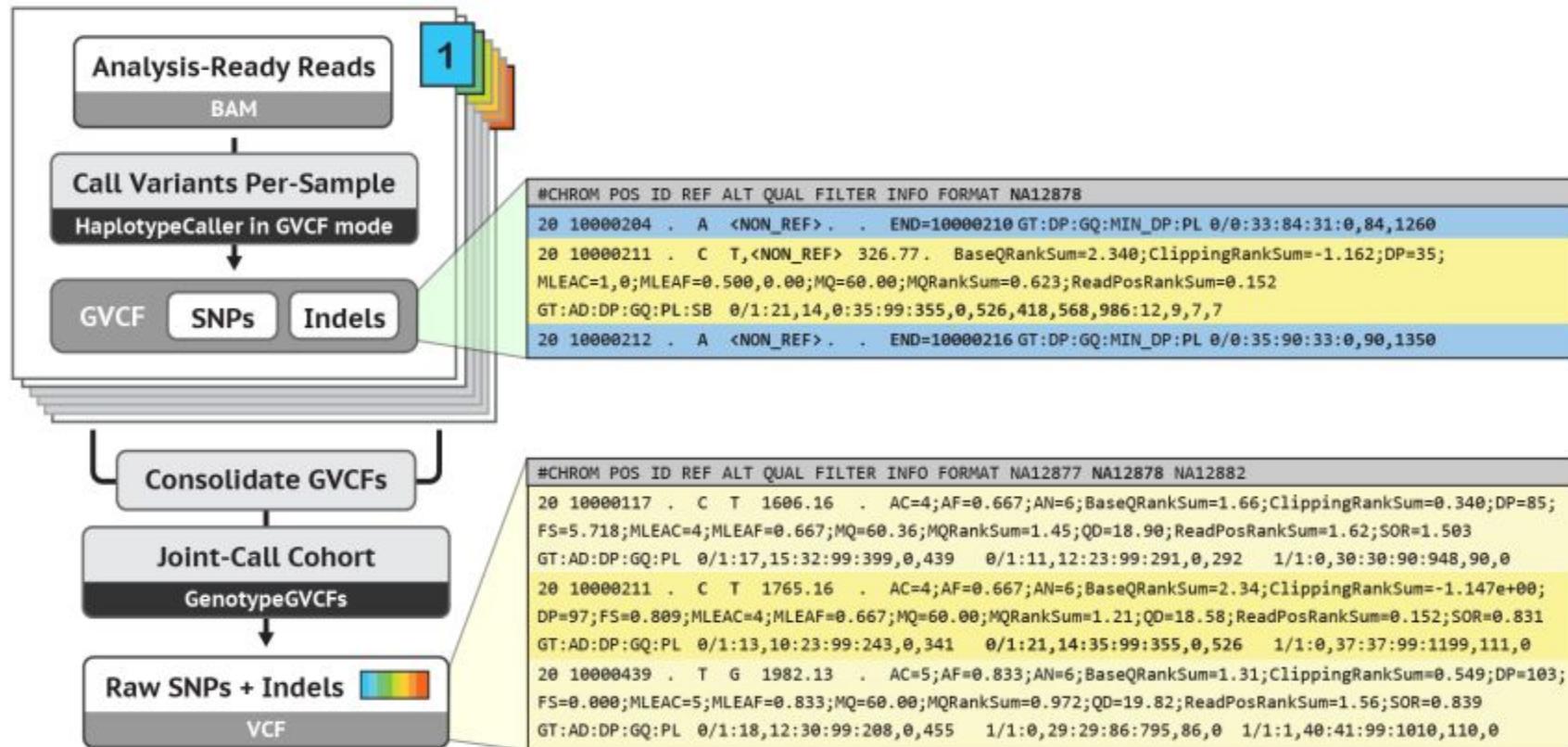
GVCF intermediate contains reference confidence estimate



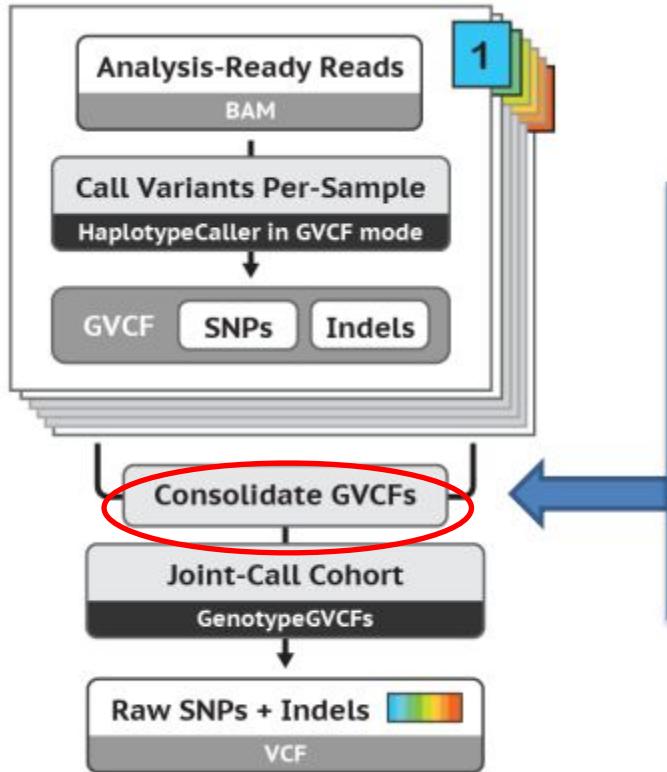
GVCFs are valid VCFs with extra information



Joint calling produces final multi-sample VCF



Consolidate GVCFs before joint calling!



Necessary for efficient scaling

- In GATK 3.x : **CombineGVCFs**

Hierarchical merge on batches of 200 samples max;
outputs GVCF

- In GATK 4.x : **GenomicsDBImport**

All samples processed in a single command;
outputs datastore

With **CombineGVCFs**:

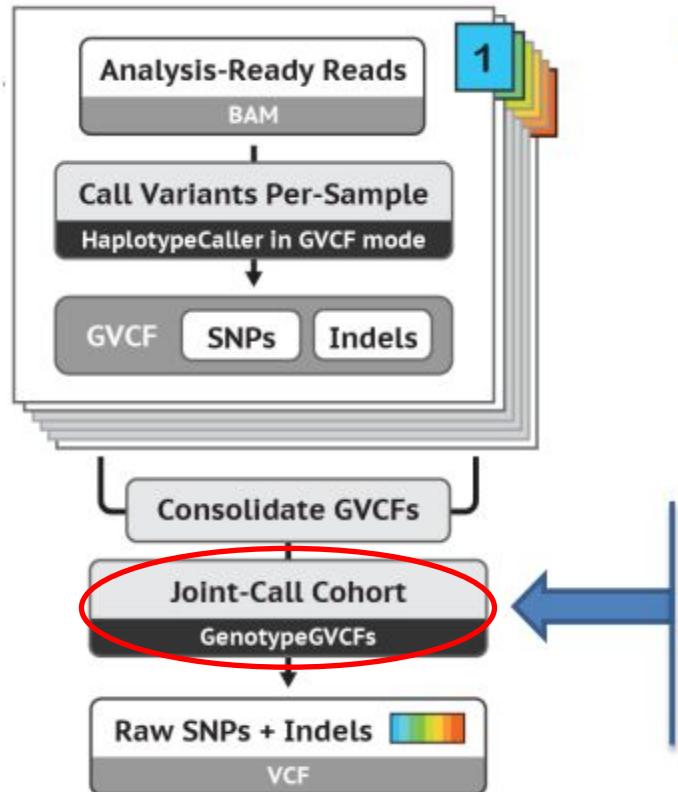
```
gatk CombineGVCFs \
-R reference.fasta \
-V sample1.g.vcf \
-V sample2.g.vcf \
-O combined.g.vcf
```

With **GenomicsDBImport**:

```
gatk GenomicsDBImport \
-R reference.fasta \
-V sample1.g.vcf \
-V sample2.g.vcf \
-L chr20,chr21 \
--genomicsdb-workspace-path gvcfs_db
```

CombineGVCFs does not scale well

Joint calling with GenotypeGVCFs



On a single- or multi-sample GVCF:

```
gatk GenotypeGVCFs \
-R reference.fasta \
-V variants.g.vcf \
-O final_variants.vcf
```

On a GenomicsDB workspace:

```
gatk GenotypeGVCFs \
-R reference.fasta \
-V gendb://gvcfs_db \
-O final_variants.vcf
```

GenotypeGVCFs cannot take multiple inputs (unlike the GATK3 version)

- **GenotypeGVCFs** can take either a **single GVCF file** (can be a merged multi-sample GVCF from `CombineGVCFs`) or a **GenomicsDB datastore**
- No more multiple inputs! (unlike GATK3)

Variant Filtering

Assigning accurate confidence scores to
each putative mutation call

Variant Context Annotations Describe the Observed Data

Each variant has a diverse set of statistics associated with it:

VCF record for an A/G SNP at 22:49582364

22	49582364	.	A	G	198.96	.																									
AC=3;	AF=0.50;	AN=6;	DP=87;	MLEAC=3;	MLEAF=0.50;	MQ=51.31;																									
MQ0=22;	QD=2.29;	SB=-31.76	GT:DP:GQ	0/1:12:99	0/1:11:89	0/1:28:37																									
INFO field		<table><tbody><tr><td>AC</td><td>No. chromosomes carrying alt allele</td><td>MLEAF</td><td>Max likelihood AF</td><td></td></tr><tr><td>AN</td><td>Total no. of chromosomes</td><td>MQ</td><td>RMS MAPQ of all reads</td><td></td></tr><tr><td>AF</td><td>Allele frequency</td><td>MQ0</td><td>No. of MAPQ 0 reads at locus</td><td></td></tr><tr><td>DP</td><td>Depth of coverage</td><td>QD</td><td>QUAL score over depth</td><td></td></tr><tr><td>MLEAC</td><td>Max likelihood AC</td><td></td><td></td><td></td></tr></tbody></table>					AC	No. chromosomes carrying alt allele	MLEAF	Max likelihood AF		AN	Total no. of chromosomes	MQ	RMS MAPQ of all reads		AF	Allele frequency	MQ0	No. of MAPQ 0 reads at locus		DP	Depth of coverage	QD	QUAL score over depth		MLEAC	Max likelihood AC			
AC	No. chromosomes carrying alt allele	MLEAF	Max likelihood AF																												
AN	Total no. of chromosomes	MQ	RMS MAPQ of all reads																												
AF	Allele frequency	MQ0	No. of MAPQ 0 reads at locus																												
DP	Depth of coverage	QD	QUAL score over depth																												
MLEAC	Max likelihood AC																														

GATK: Filtering variants

- Calling algorithms are very permissive
- Calling sets contain many false positives
- Two filtering approaches :
 - Hard filtering : using thresholds on annotations
 - Variant recalibration using machine learning
- Sensitivity vs Specificity

GATK: Hard filtering

- Suitable for all experiments (targeted gene, WES, small sample size, etc.)
- Goal: define annotations and thresholds to filter bad variants
- Pros:
 - Easy to perform
- Cons:
 - Hard to define annotations to use
 - Hard to define threshold
 - May filter good variants, may keep bad variants

QualByDepth

- The QUAL field of the VCF file is defined as a Phred score that reflects the variant quality.
- The QualByDepth (QD) score is the QUAL score divided by the allele depth of the variant (i.e., the ALT allele depth).
- There is no “normal” range for this value, but a QD under **2** is considered poor quality.

FisherStrand

- This parameter is an estimate of strand bias, a kind of sequencing bias in which one strand is favored over the other.
- The higher the value for FS, the more likely there is to be bias or false-positive calls.
- Values of FS over **60** are taken to be strong evidence for strand bias.

RMSMappingQuality

- The root mean square of the mapping quality provides an estimation of the overall mapping quality of reads supporting a variant call.
- The RMS is based on the mapping qualities of the n reads that support variant call.
- The threshold suggested by GATK for MQ is **40**

MappingQualityRankSumTest

- The rank sum test for mapping qualities of REF reads versus ALT reads.
- Compares the mapping qualities of the reads supporting the reference allele with those supporting the alternate allele.
- For variant calling, we are interested in whether there is evidence that the quality of the data supporting the alternate allele is comparatively low.
- GATK suggests filtering if MQRankSum is less than **-12.5**.

ReadPosRankSumTest

- The rank sum test for relative positioning of REF versus ALT alleles within reads.
- Tests whether there is evidence of bias in the genomic position of reference and alternate alleles within the reads that support them.
- If a variant is called only near the ends of reads, can be an indication of error.
- GATK suggests filtering if ReadPosRankSum is less than **-8**.

Strand Odds Ratio (SOR)

- The strand odds ratio measures the ratio of the odds of the variant being observed on the forward strand versus the reverse strand.
- A high SOR value suggests that the variant is more likely to be real, since it is observed on both strands and is less likely to be a sequencing artifact.
- A low SOR value suggests that the variant may be an artifact or that there may be a bias in the sequencing or genotyping process.

Inbreeding Coefficient (InbreedingCoeff)

- A filter option applied to indel(insertion/deletion) variants.
- It is calculated based on the observed and expected number of homozygous genotypes in a population.
- A positive inbreeding coefficient suggests an excess of homozygotes, indicating potential inbreeding or relatedness, while a negative coefficient suggests a deficit of homozygotes.
- Variants with extreme positive or negative inbreeding coefficients may be more likely to be artifacts or sequencing errors and can be filtered out.

GATK: Hard filtering recommendations

- Filtering SNPs where any:
 - **QD** < 2.0
 - **MQ** < 40.0
 - **FS** > 60.0
 - **SOR** > 3.0
 - **MQRankSum** < -12.5
 - **ReadPosRankSum** < -8.0
- Filtering Indels where any:
 - **QD** < 2.0
 - **ReadPosRankSum** < -20.0
 - **InbreedingCoeff**¹ < -0.8
 - **FS** > 200.0
 - **SOR** > 10.0

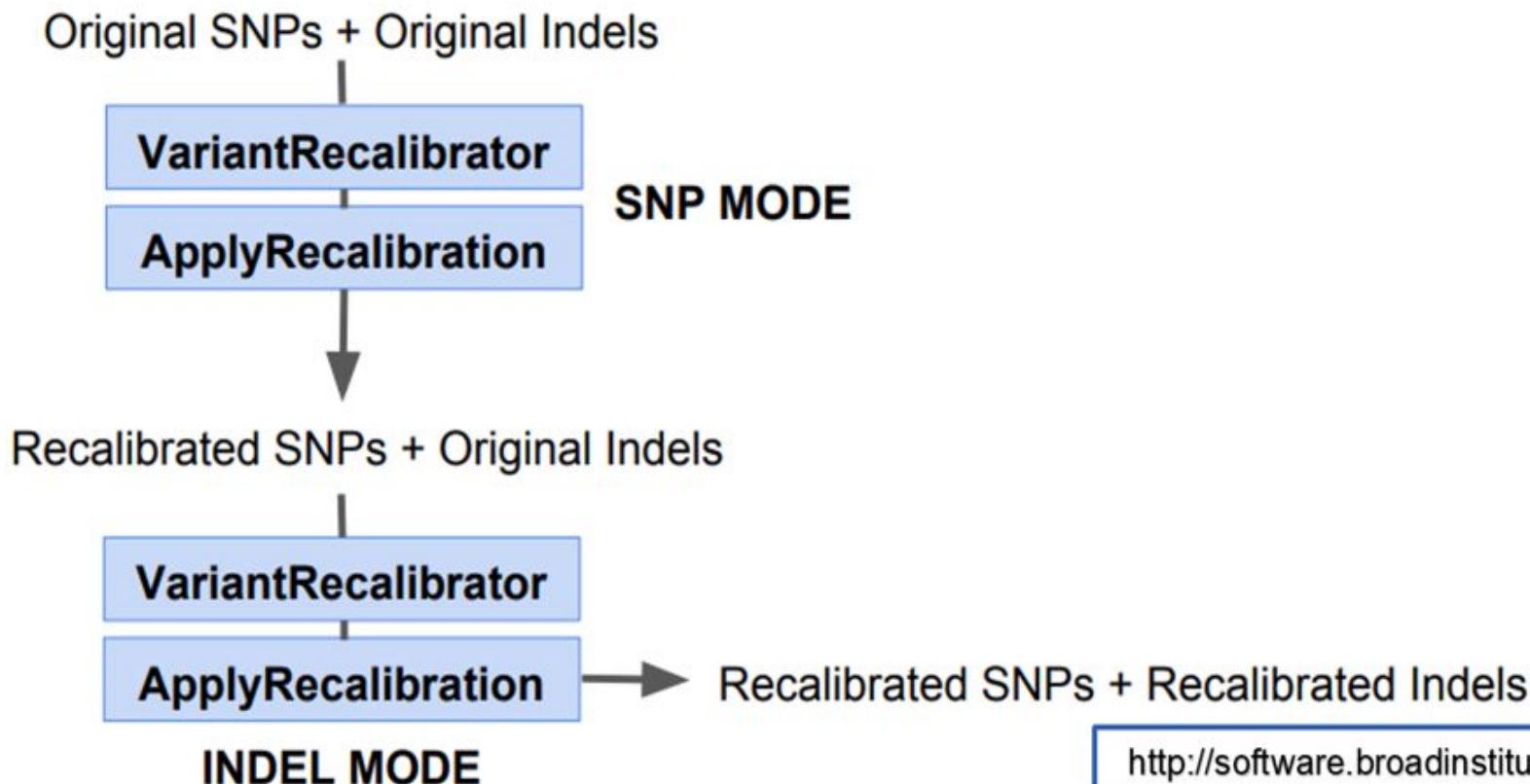
¹ When sample size > 10

Warning: Threshold on maximum depth should not be used for WES data

GATK : Variant Quality Score Recalibration (VQSR)

- Preferred method
- Requires:
 - DNA-seq data (not working on RNA-seq data)
 - Well curated training/truth resources (usually not available for non human organisms)
 - Large amount of variants (no targeted gene panels, etc.)
 - > 30 samples for WES data (1000G WES samples can be added if needed but not optimal)
- Based on machine learning

GATK : VQSR workflow



GATK : VQSR SNP human resources

- **Hapmap**
 - Training
 - Truth
 - Prior = 15
- **Omni**
 - Training
 - Truth
 - Prior = 12
- **1000G SNPs High confidence**
 - Training
 - Prior = 10
- **dbSNP**
 - Known
 - Prior = 2

Annotations: QD, MD, MQRankSum, ReadPosRankSum, FS, SOR, DP¹, InbreedingCoeff

GATK : VQSR Indel human resources

- Mills Indels
 - Training
 - Truth
 - Prior = 12
- dbSNP
 - Known
 - Prior = 2

Annotations: QD, MD, MQRankSum, ReadPosRankSum, FS, SOR, DP¹, InbreedingCoeff

Convolutional Neural Net (CNN)

- Annotate a VCF with scores from a Convolutional Neural Network (CNN).
- The default model should not be used on VCFs with annotations from joint call-sets.
- Two ways to score variants
 - 1D Model
 - Variant annotations
 - Reference
 - 2D Model
 - Variant annotations
 - Reference
 - Read information

```
gatk CNNScoreVariants \  
-V vcf_to_annotate.vcf.gz \  
-R reference.fasta \  
-O annotated.vcf
```

```
gatk CNNScoreVariants \  
-I aligned_reads.bam \  
-V vcf_to_annotate.vcf.gz \  
-R reference.fasta \  
-O annotated.vcf \  
-tensor-type read-tensor
```

Filter Variants with FilterVariantTranches

- Apply tranche filtering to VCF based on scores from an annotation in the INFO field.
- Tranches are specified in **percent sensitivity** to the variants in the resource files.
- Higher tranches = More sensitive, less precise (lower variant scores)
- Lower tranches = Less sensitive, higher precision
- The default tranche filtering threshold for SNPs is **99.95** and for INDELS it is **99.4**.

```
gatk FilterVariantTranches \
-V input.vcf.gz \
--resource hapmap.vcf \
--resource mills.vcf \
--info-key CNN_1D \
--snp-tranche 99.95 \
--indel-tranche 99.4 \
-O filtered.vcf
```

```
gatk FilterVariantTranches \
-V input.vcf.gz \
--resource hapmap.vcf \
--resource mills.vcf \
--info-key CNN_2D \
--snp-tranche 99.95 \
--indel-tranche 99.4 \
--invalidate-previous-filters \
-O filtered.vcf
```

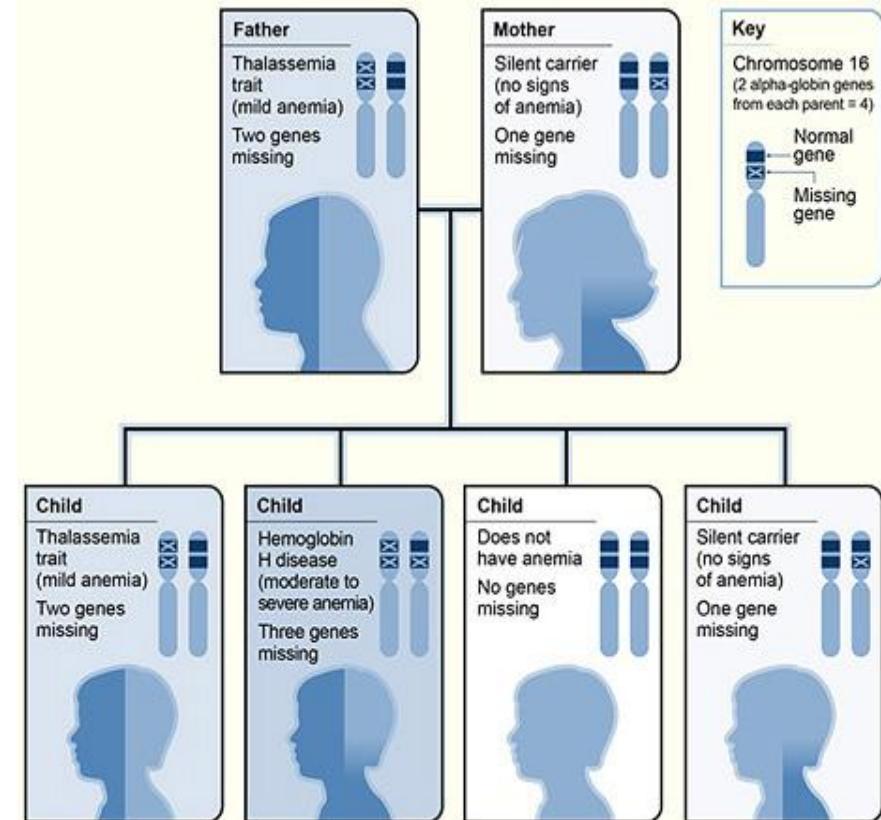
Genotype Refinement Workflow

Using additional data to improve genotype
calls and likelihoods

<http://software.broadinstitute.org/gatk/>

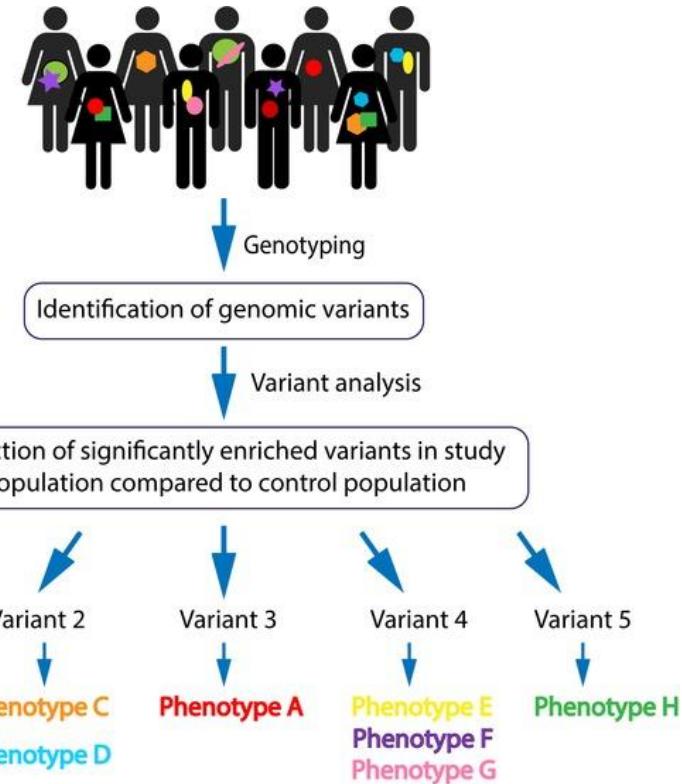
Genotypes for patients

- Medical geneticists need genotypes for patients
 - Do any patients have two copies of a LOF mutation?
 - Are the parents of a diseased child likely to have more afflicted children?



Genotypes for association studies

- Population geneticists need genotypes for association studies
 - How does the number of copies of an allele affect the phenotype?



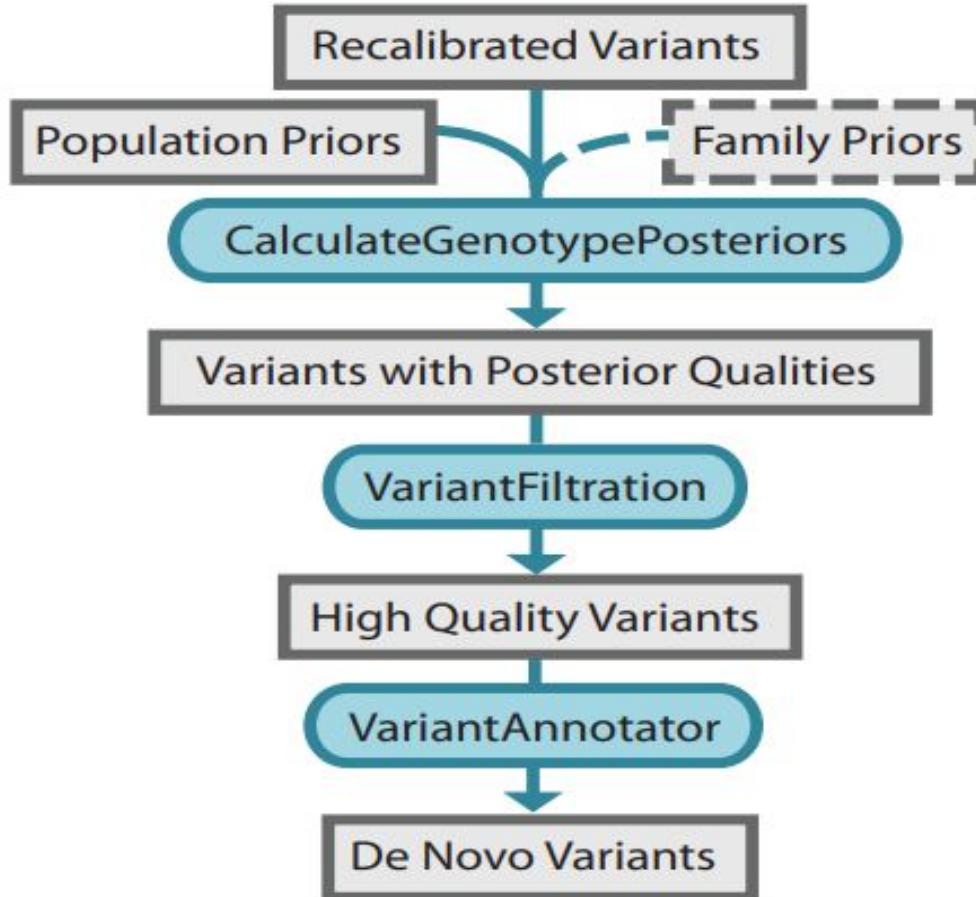
Variant call vs. Genotype call



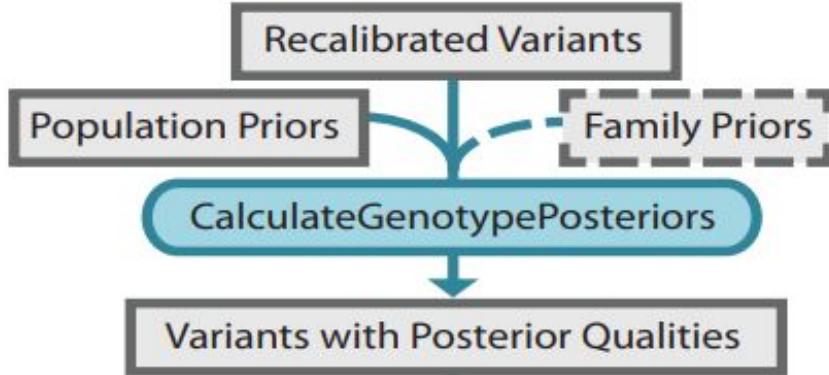
Genotype call quality is crucial !

- Some sites/samples have poor genotype calls
 - Can be ambiguous due to low confidence
 - Might be entirely wrong!
- Can additional (independent) data improve genotype calls?
 - Use high quality data (like 1000G) as priors
 - Use pedigree (if available)
 - Calculate posterior genotype probabilities

Genotype Refinement Workflow

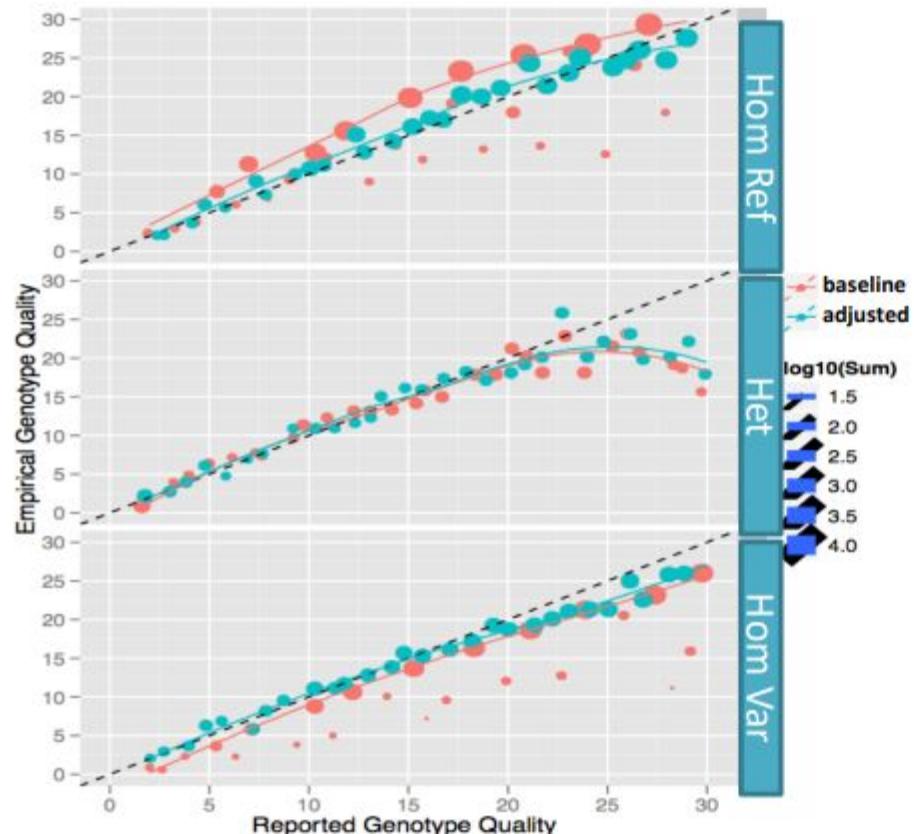


CalculateGenotypePosteriors



```
gatk CalculateGenotypePosteriors \
-R reference.fasta \
-V input.vcf \
-ped family.ped \
-supporting population.vcf \
-O output.vcf
```

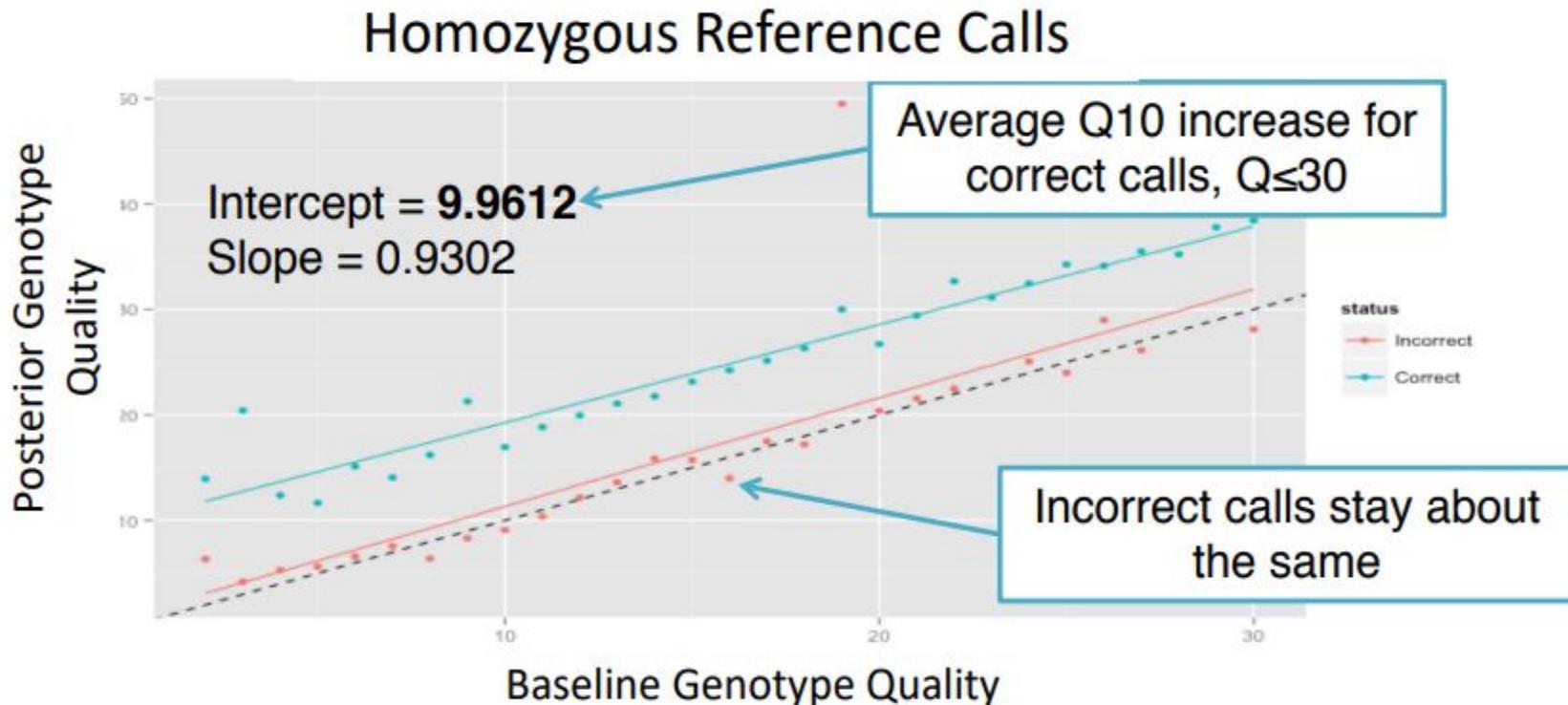
Population priors improve genotype confidence



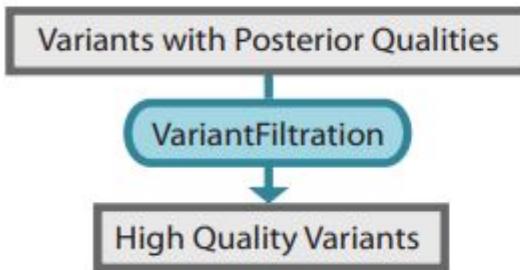
Baseline HomRef calls are under confident, but posterior calls are more accurate

Baseline HomVar calls are over confident, but posterior calls are improved

Assessing confidence and correctness



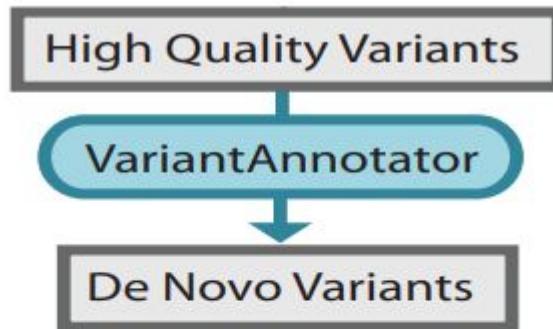
Filter low confidence GQs



```
gatk VariantFiltration \
    -R reference.fasta \
    -V input.vcf \
    --genotype-filter-expression "GQ<20" \
    --genotype-filter-name "lowGQ" \
    -O output.vcf
```

- Use VariantFiltration to filter ambiguous, low-confidence calls
- Recommend threshold is GQ= 20
 - GQ 20 is Phred-scaled 99% confidence
- Restrict further analysis to high-quality data

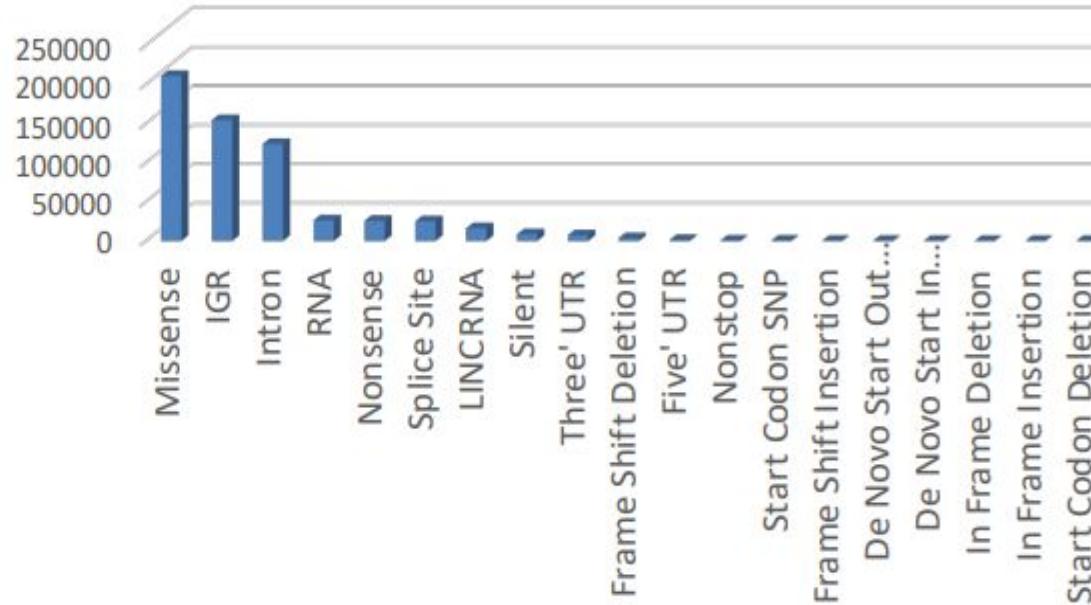
VariantAnnotator



```
gatk VariantAnnotator \
    -R reference.fasta \
    -V input.vcf \
    -A PossibleDeNovo \
    -O output.vcf
```

Genotype refinement yields more high-quality genotypes

- Initial genotype calls may be ambiguous or wrong
- Applying population + family priors improves confidence
- More high confidence genotypes -> more data for downstream analysis!
- Tools can be used for further variant annotation (e.g. Funcotator)



Callset Evaluation

Comparing statistics between
your callset and external resources

<http://software.broadinstitute.org/gatk/>

What do callset evaluation methods aim to determine?

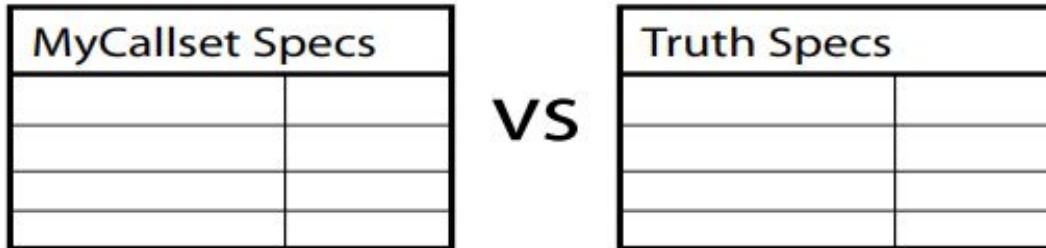
IDEAL	OKAY	TERRIBLE
Your variant calls perfectly match the underlying biological truth	You found many real variants and called few false positives	You didn't find any real variants and only called artifacts!

Where are you on this spectrum?

How do I figure out how good/bad my callset is?



Compare key statistics from callset and truth set stats

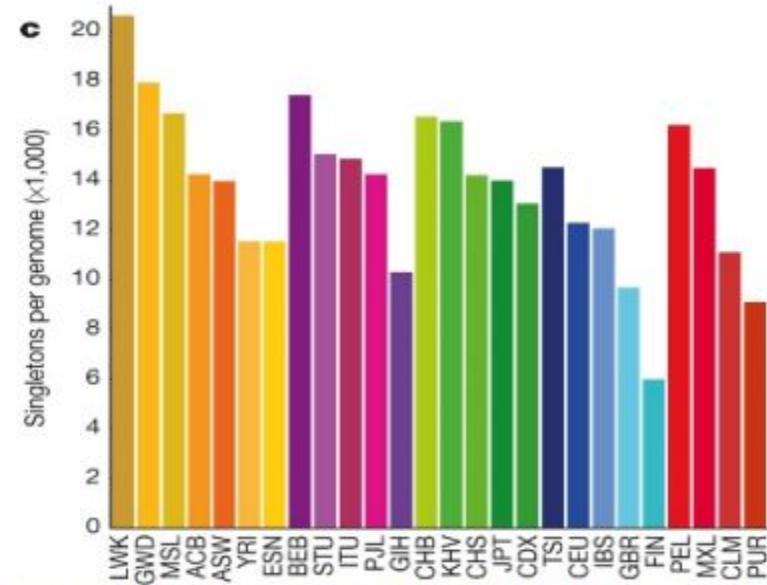


Guiding principle: divergence is indicative of error

Key assumption: truth set is representative / comparable

- Important to match dataset properties!
 - Population ethnicity (European, African, etc.)
 - Sequencing / exp. design (WGS vs. WES)
 - Cohort size

Not easy! You might need to use sub-cohorts (of both sets) to match all three.



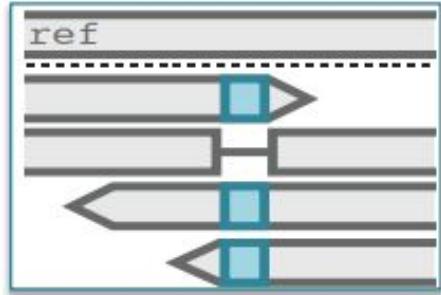
<http://www.nature.com/nature/journal/v526/n7571/full/nature15393.html>

Commonly used truth sets

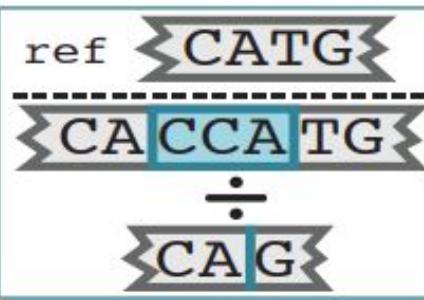
- **dbSNP**
 - All previously reported variation (lots of junk!)
- **ExAC and GnomAD**
 - Extensive catalog of human variation built by aggregating results from many studies
- **HapMap**
 - Highly validated common human variants
- **OMNI**
 - Common variation validated by array
- **NIST's Genome in a bottle, or illumina's Platinum Genomes.**
 - high confidence callsets from a handful of common benchmarking samples

Recommended metrics for callset evaluation

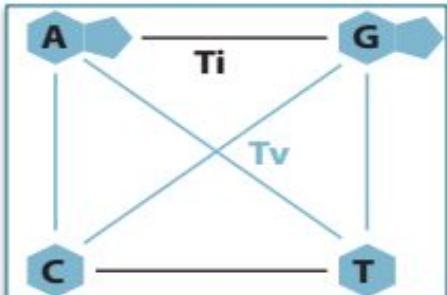
Number of Indels & SNPs



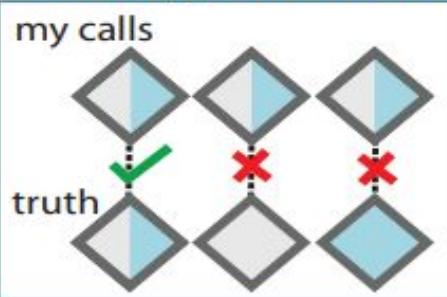
Indel Ratio



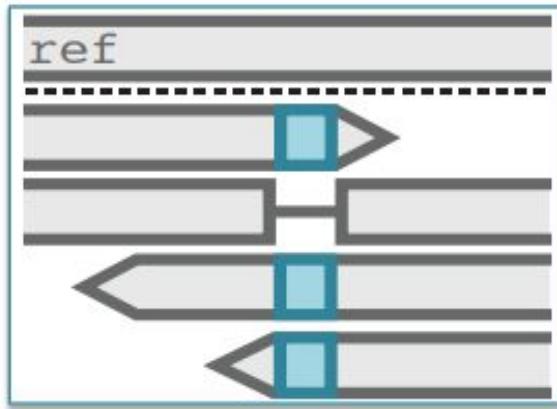
TiTv Ratio



Genotype Concordance



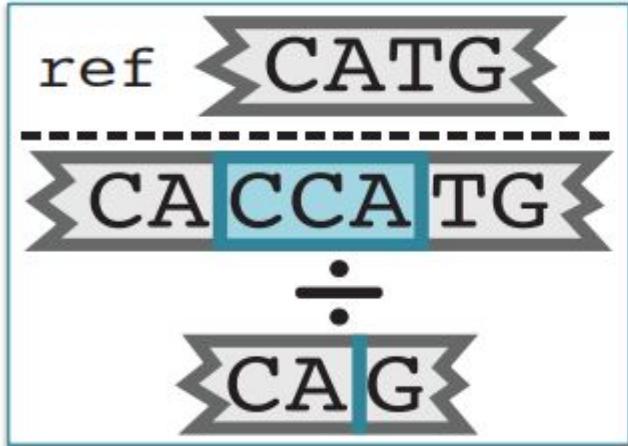
Number of Indels & SNPs



- Variants = Indels + SNPs
- Useful for order-of-magnitude sanity check
- Vary by size and diversity of cohort

Sequencing Type	# of Variants (in 1 sample)
WGS	~4.4 M
WES	~21 k

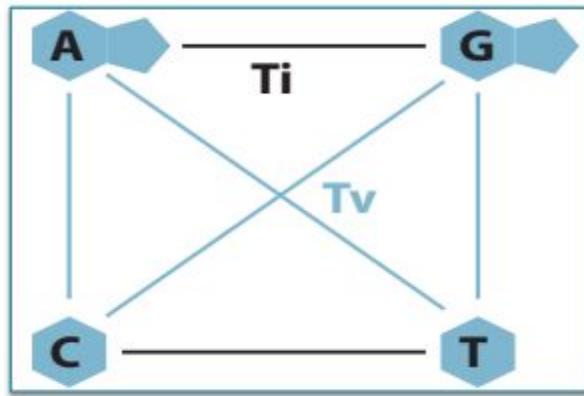
Indel Ratio



- Ratio of **insertions** to **deletions**
- Varies by allele frequency: common ("known") vs. rare ("novel")

Variant prevalence	Indel Ratio
Common	~1
Rare	0.2-0.5

TiTv Ratio (Transitions/Transversions)

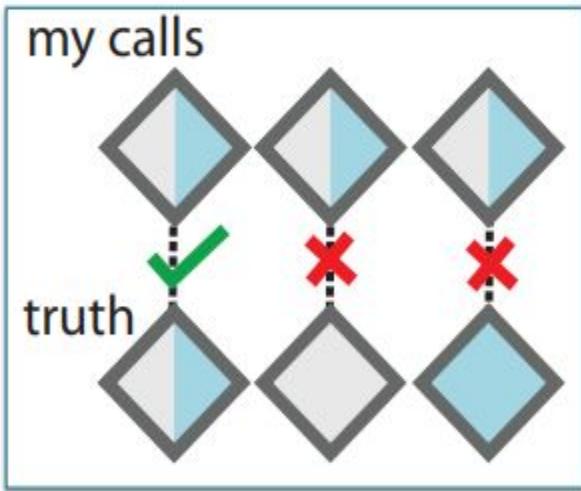


Sequencing Type	TiTv Ratio
WGS	2.0-2.1
WES	3.0-3.3

In Humans...

- Used for SNPs only
- If variation were random: expect ratio of 0.5 as there are twice as many possible transversions vs transitions!
- Low TiTv ratio indicates high rate of false positives

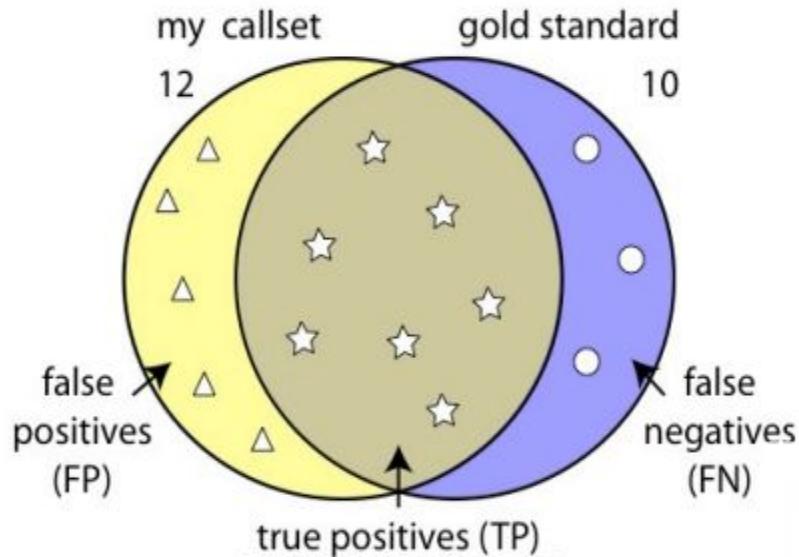
Genotype Concordance



- Most appropriate truth set is genotyping chip for same sample
- % Genotype calls in callset that match calls in truth set
- Unmatched variants considered false positives

Concordance metrics

Variant Concordance



SENSITIVITY

$$\frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{7}{7 + 3} = 70\%$$

FALSE DISCOVERY RATE

$$\frac{\text{FP}}{\text{FP} + \text{TP}} = \frac{5}{5 + 7} = 42\%$$



GT CONCORDANCE

$$\frac{\sum \text{matches}}{\text{TP}} = \frac{4}{7} = 57\%$$

So how to generate these metrics?

Variant Level Evaluation

```
gatk VariantEval \  
    -R reference.b37.fasta \  
    -eval callset.vcf \  
    --comp truthset.vcf \  
    -o results.eval.grp
```

Genotype Level Evaluation

```
java -jar picard.jar \  
    GenotypeConcordance \  
    CALL_VCF=input.vcf \  
    TRUTH_VCF=truth_set.vcf \  
    CALL_SAMPLE=sample_name \  
    TRUTH_SAMPLE=sample_in_truth \  
    O=gc_concordance.vcf \  
    
```

<https://gatk.broadinstitute.org/hc/en-us/articles/13832653055387-VariantEval-BETA>

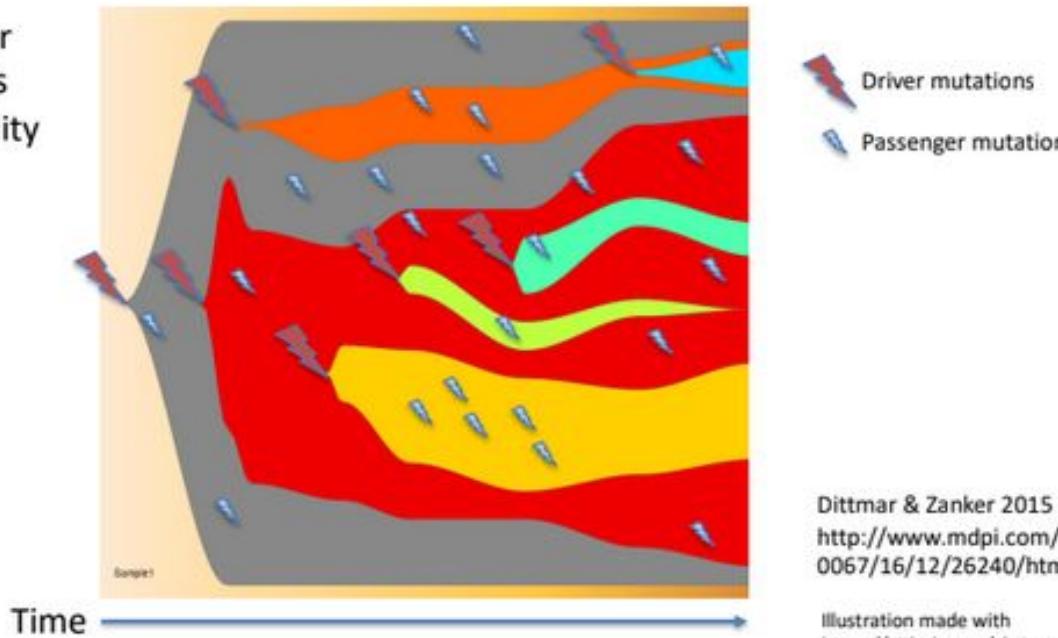
<https://gatk.broadinstitute.org/hc/en-us/articles/13832626568987-GenotypeConcordance-Picard>

Introduction to Somatic Variant Discovery

Key considerations and workflow logic

Role of mutation events in tumor progression

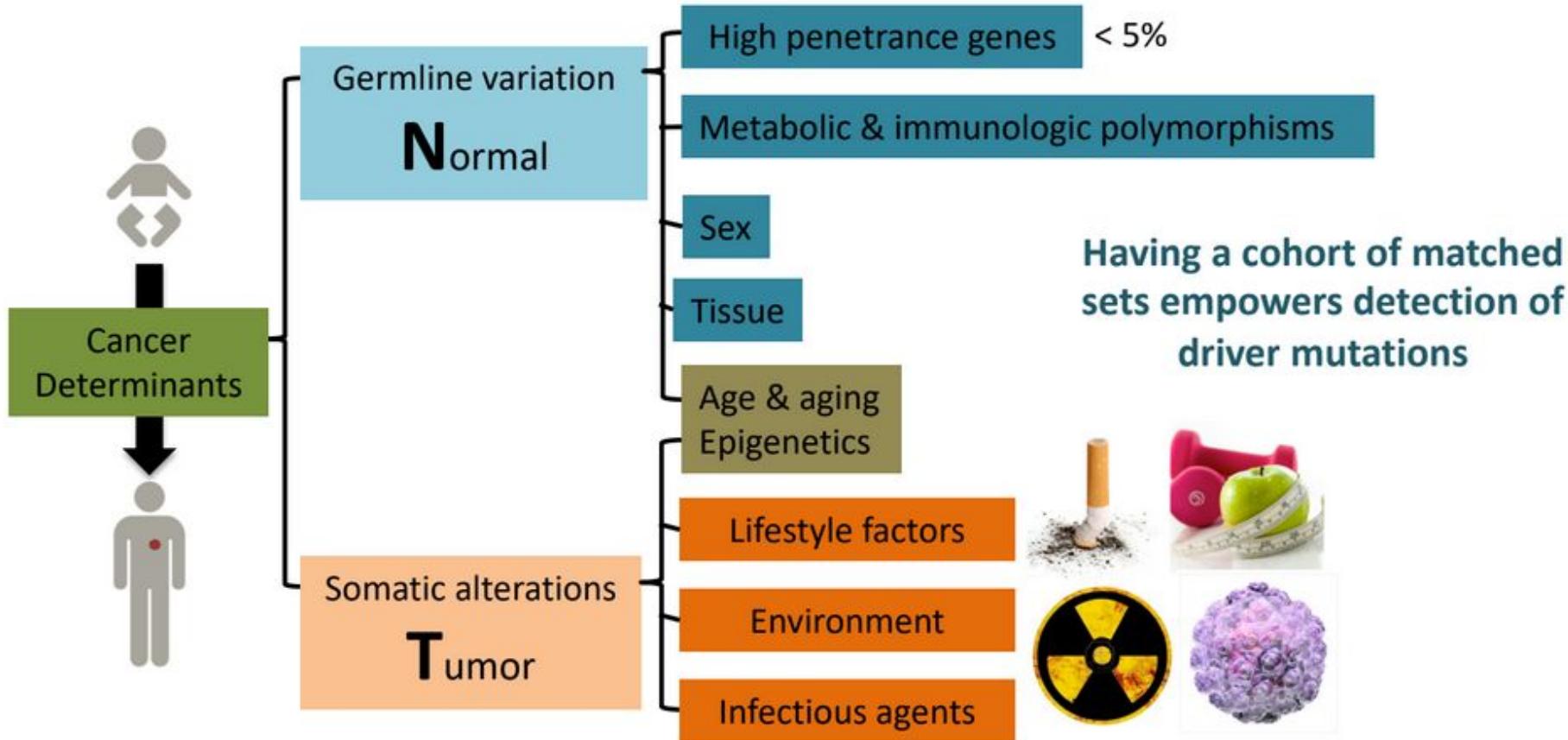
Increasing tumor heterogeneity as genomic instability increases



Dittmar & Zanker 2015
<http://www.mdpi.com/1422-0067/16/12/26240/htm>

Illustration made with
<https://github.com/chrisamiller/fishplot>

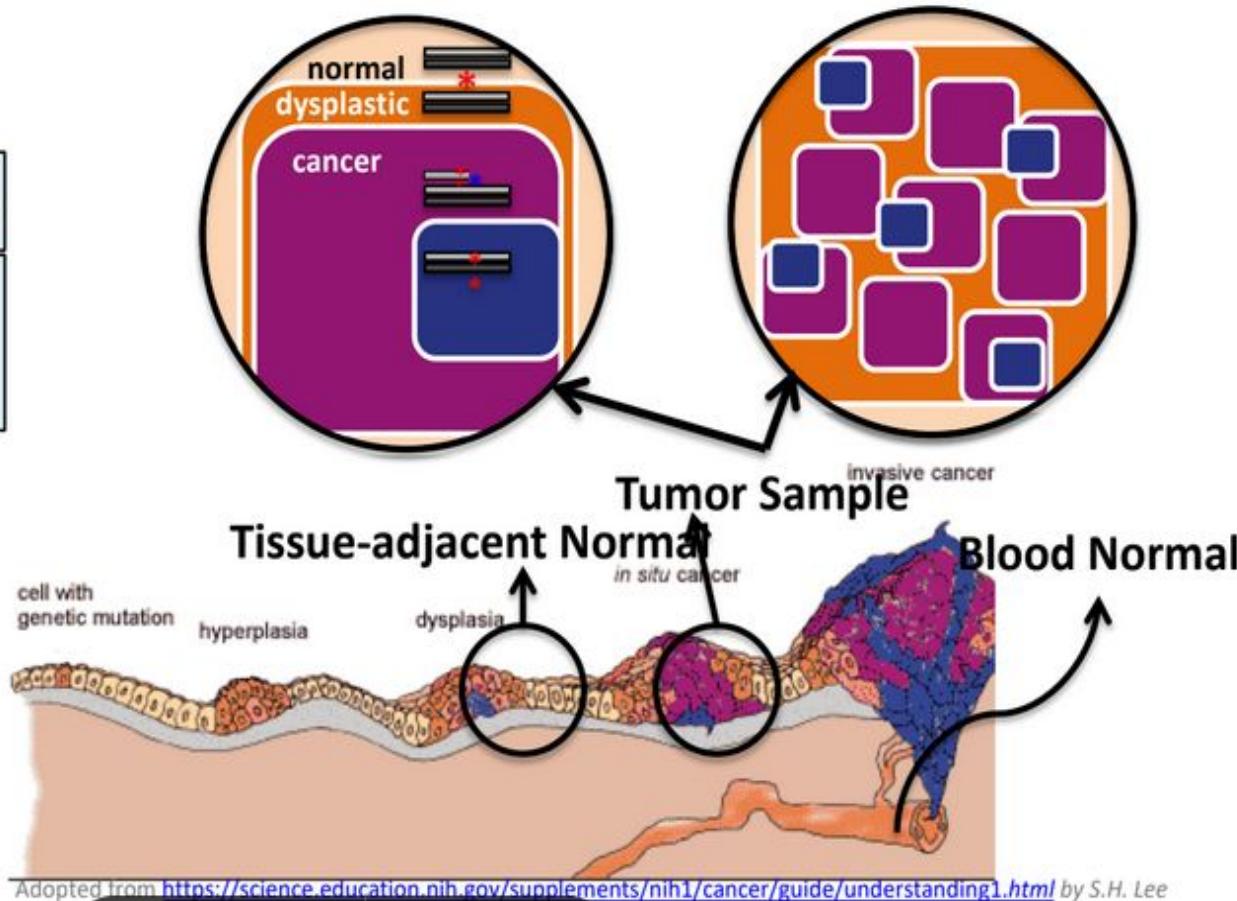
Cohorts of paired T-N data to detect driver mutations



Tumor and normal contamination and heterogeneity

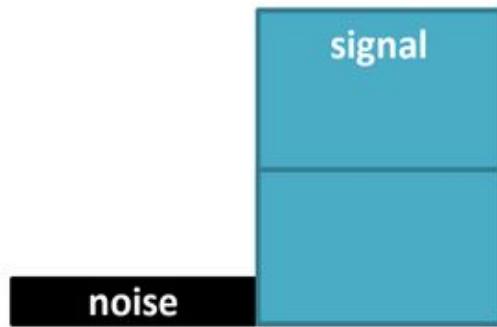
$$\text{Tumor purity} = \frac{(\text{tumor cells})}{(\text{normal} + \text{tumor cells})}$$

Tumor heterogeneity is based on polygenomic populations, segregated or intermixed, due to ongoing subclonal evolution.

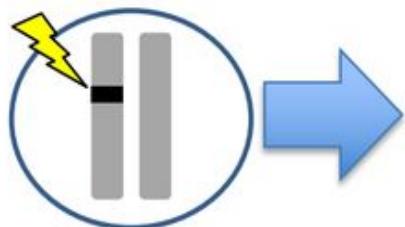


Amount of signal may be comparable to noise

Expectation for germline variants

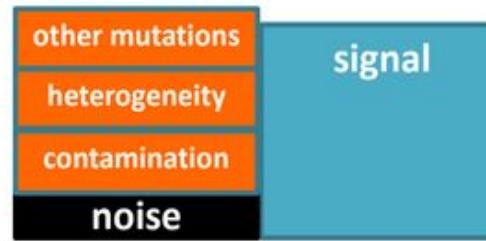


+ AF expected to follow ploidy

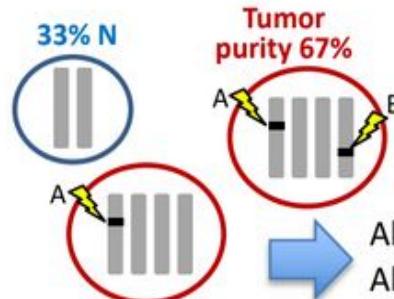


About 50% of reads will support the alternate allele

Expectation for somatic variants



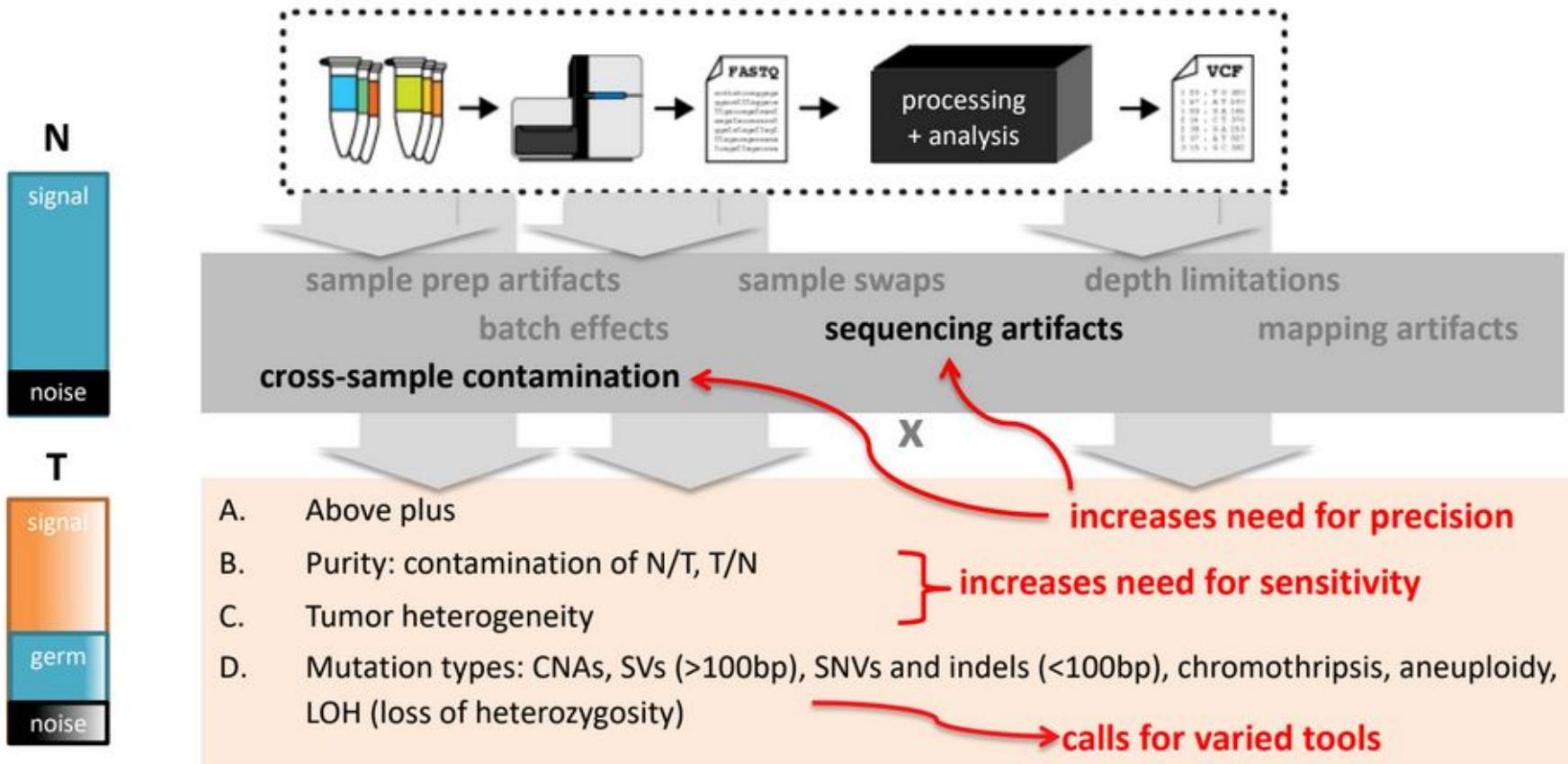
+ no reliance on ploidy for AF



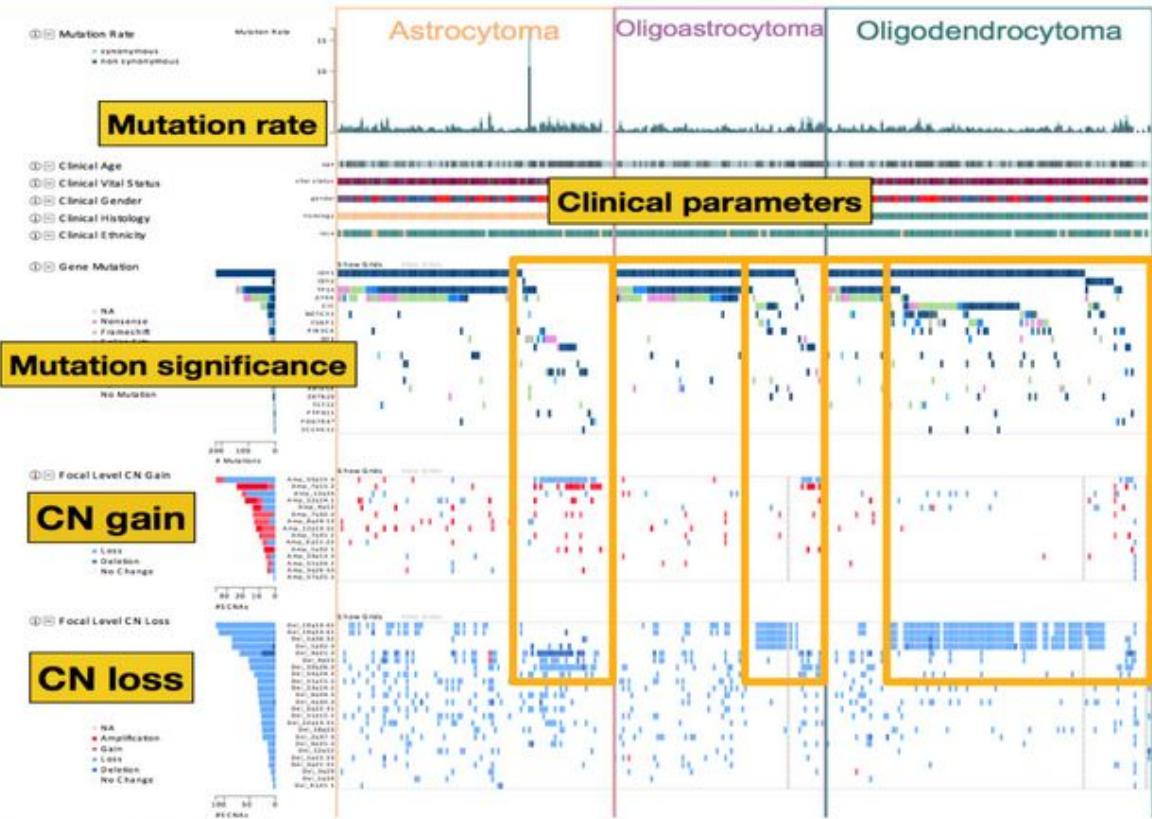
- Example tumor sample:
- Clonal SNV (A)
 - Clonal copy number duplication
 - Subclone of 50% of cancer cells (heterogeneity)
 - Subclonal SNV (B)

About 20% of reads will support A
About 10% of reads will support B

Cancer-specific challenges confound analyses



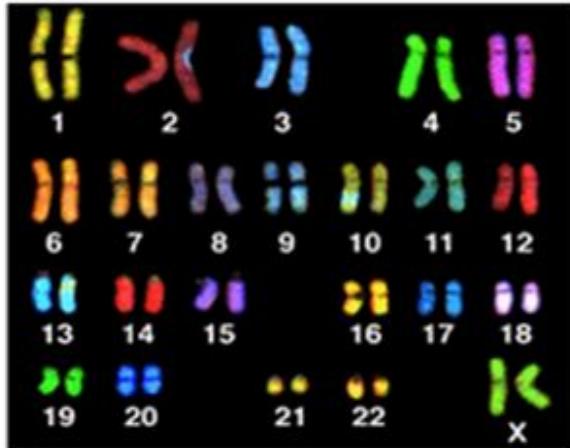
A tumor's genomic alterations are multilayered



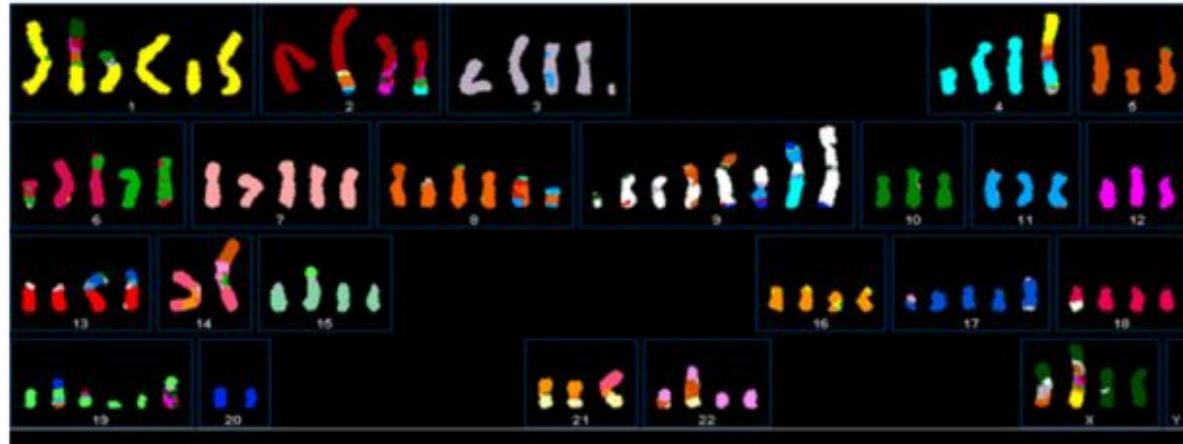
- Brain lower grade glioma (LGG) iCoMut plot from FireBrowse
- Patients lacking characteristic mutations in IDH1/2, TP53 and ATRX have increased focal copy number alterations
- Cancer analysis must take into account a wide spectrum of alterations

Somatic alterations can be dramatic

Normal Cell



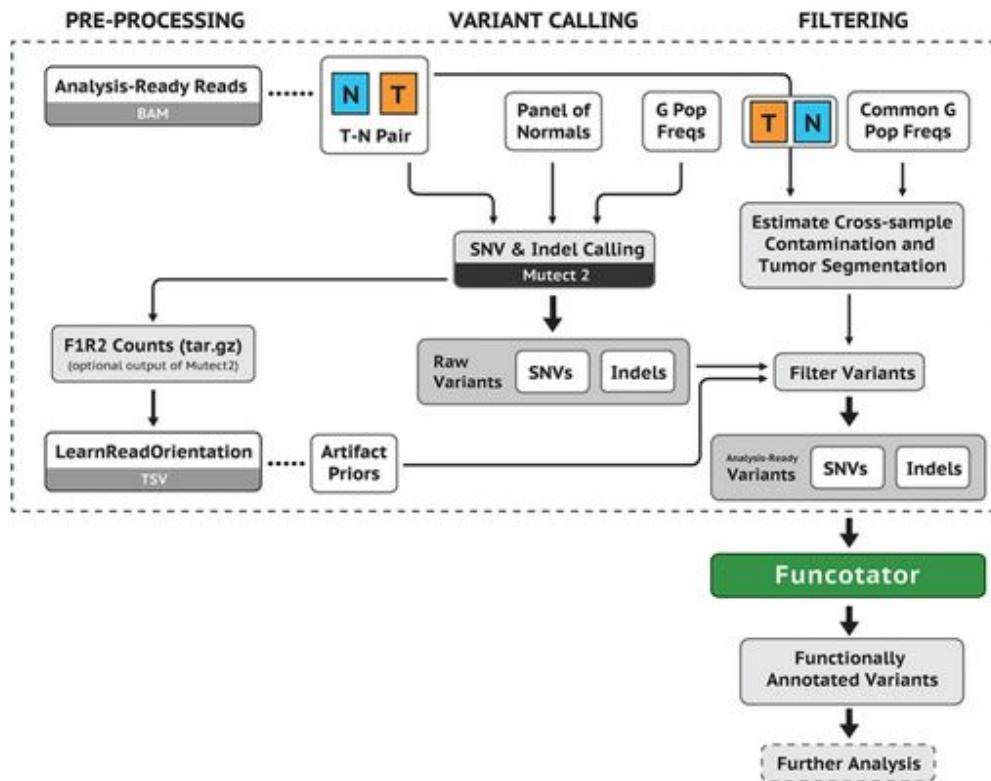
Cancer Cell Line HCC1954



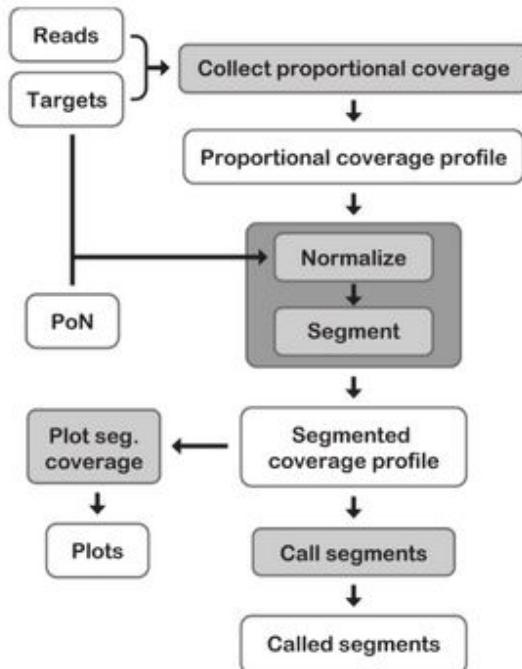
- Spectral karyotyping paints each chromosome pair with a color
- Alterations can vary dramatically between cancers and within cancers

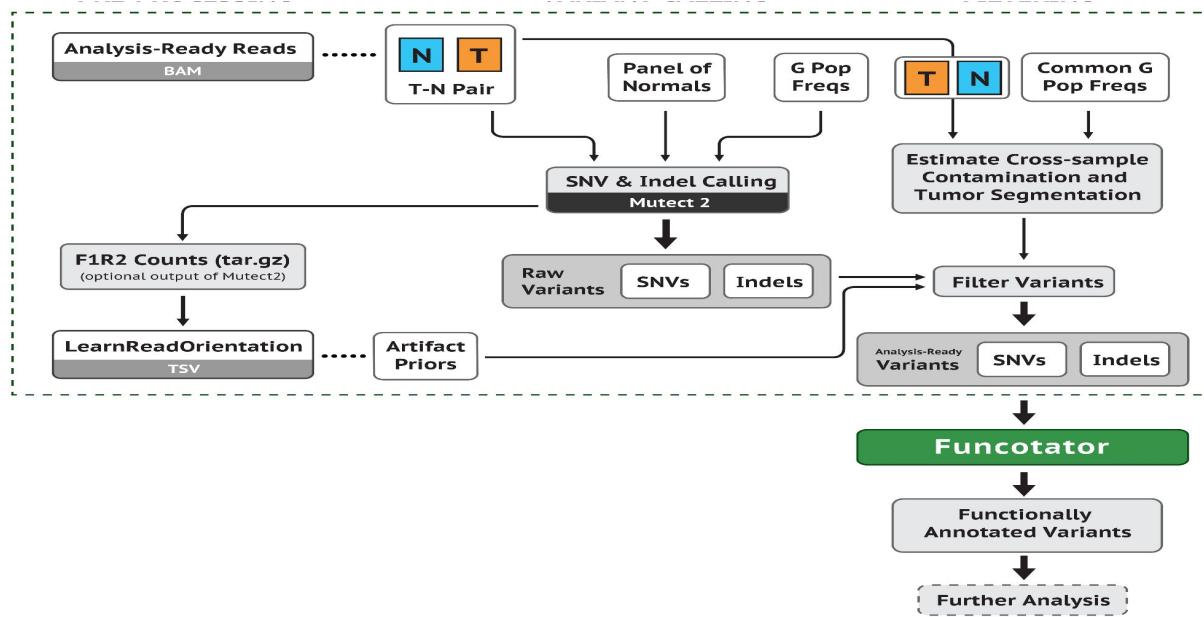
Somatic variant discovery workflow in GATK4

Somatic SNV and Indel Discovery



Somatic CNV Discovery





SOMATIC SNVs & INDELS

Logic of the Tumor-Normal workflow

Comparison to matched normal → subtraction of germline background



Tumor-only analysis

Panel of Normal (PON)

- It is possible to run the workflow without a matched normal in “tumor only mode” (normally used for PON creation).
- MUST have a good PON to eliminate common germline variation.
- Will still require extra filtering.

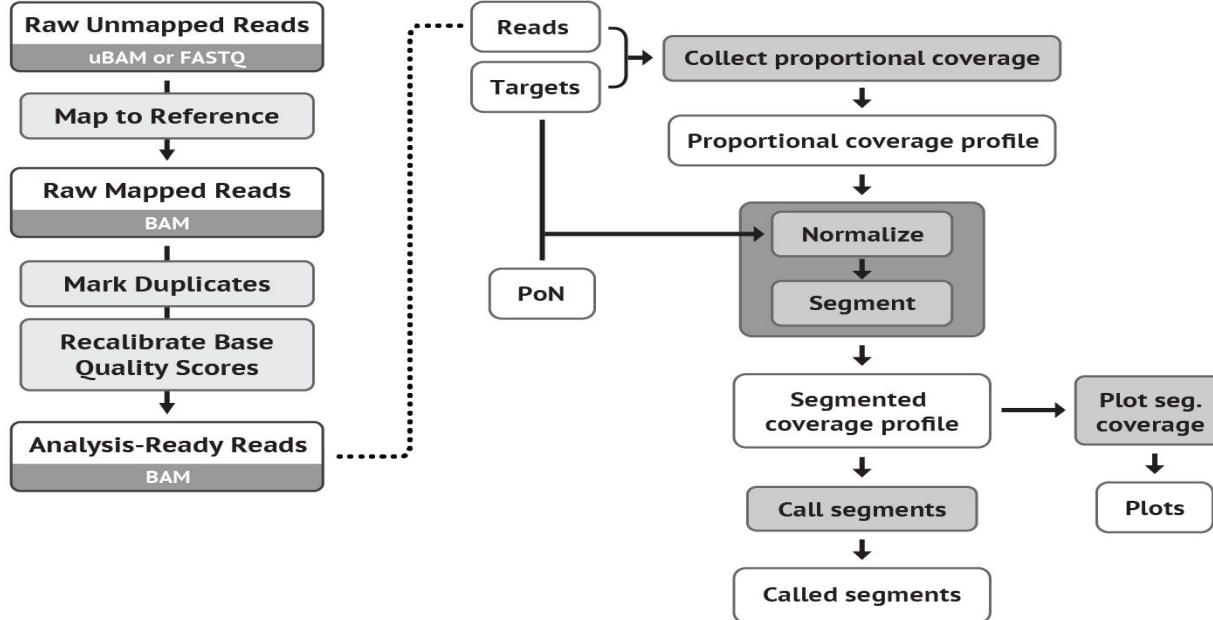
Panel of Normals for SNVs and Indels

- VCF of calls made from a set of *unrelated* “normal” samples.
- Main purpose:

Eliminate common/recurring *technical artifacts*.

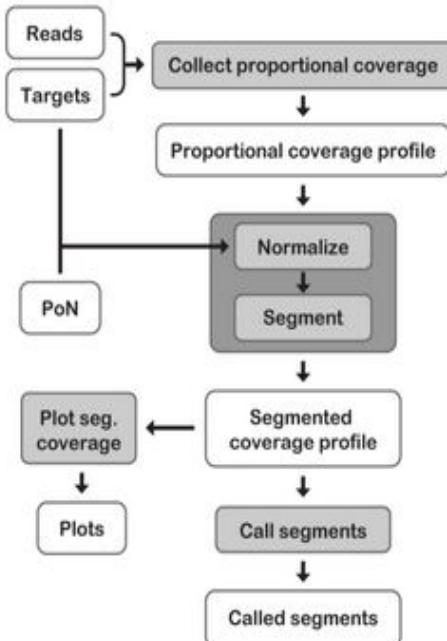
-> should use normals made using the same data generation techniques (e.g same capture kit for exomes, same sequencing platform, etc.)

- Secondary purpose: also eliminate *germline variants* not called in the matched normal (or approximate the normal if none is available)

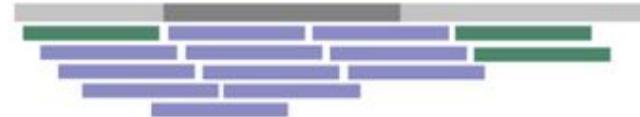


SOMATIC CNVs

Copy number: Coverage and Normalization

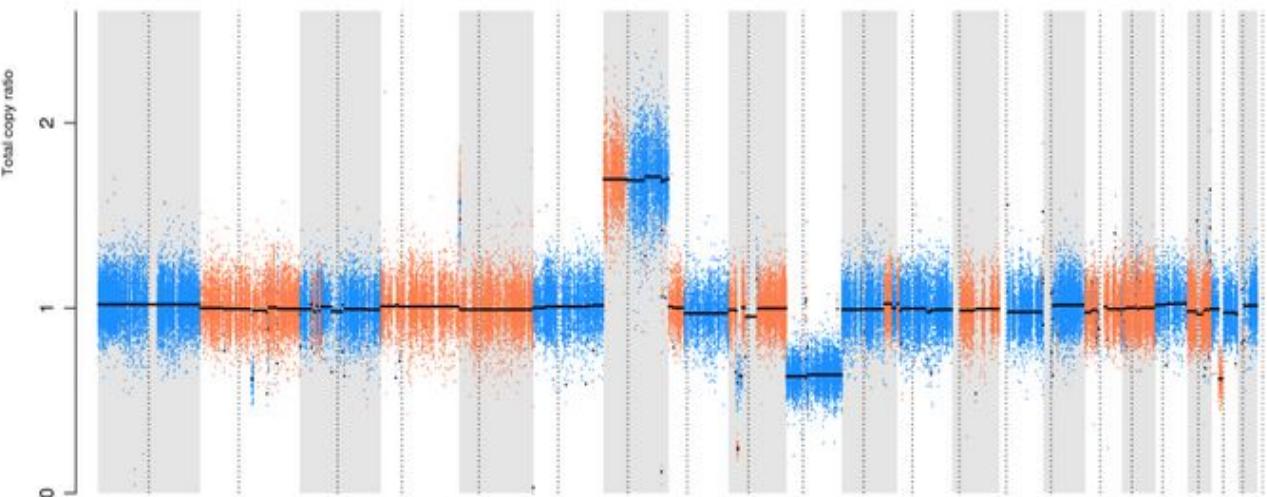


Collect proportional coverage

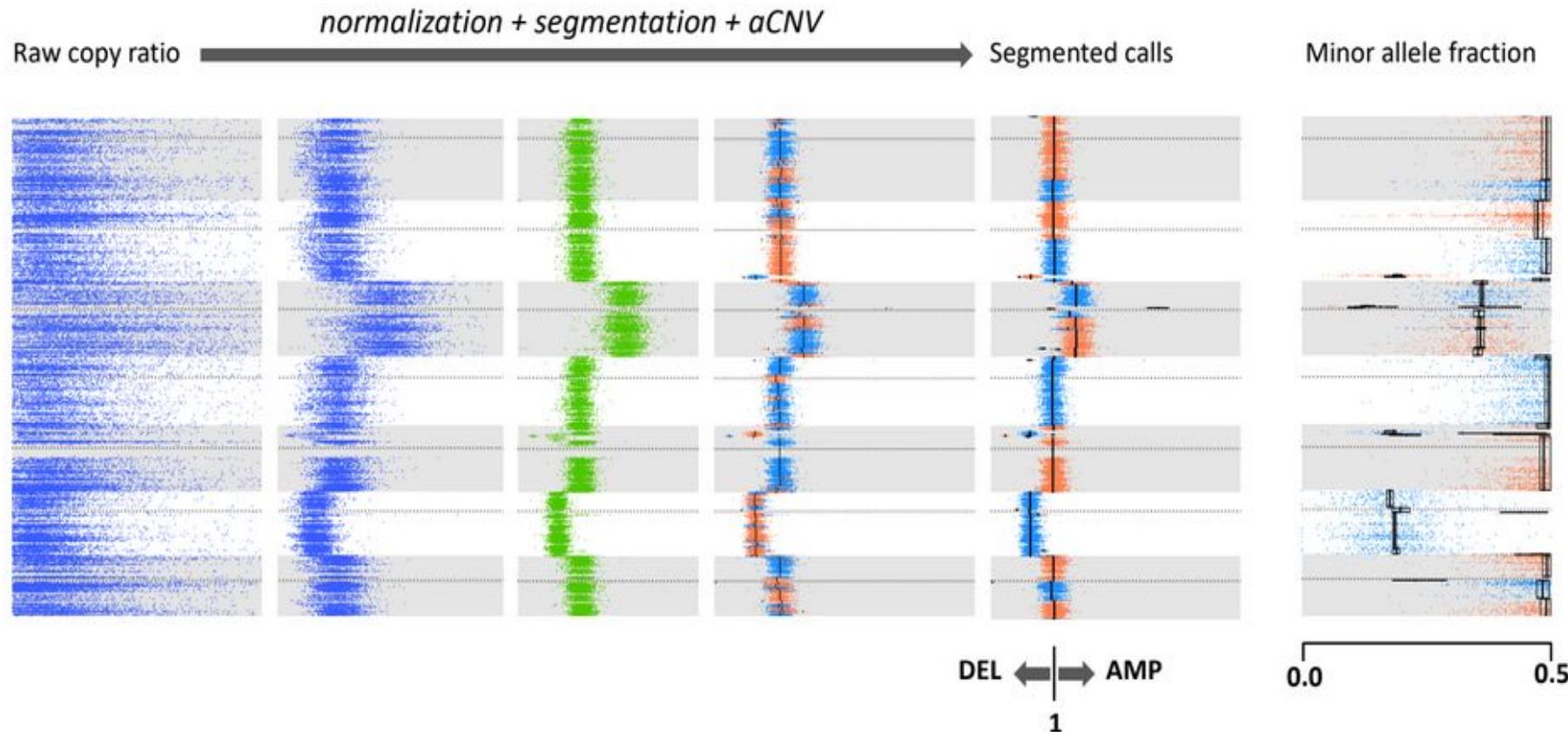


Normalize to remove noise

Identify segment boundaries



“Denoising” normalization process is essential



Panel of Normals for CNVs

- Made from a set of unrelated “normal” samples BUT **not** a vcf (unlike SNVs and Indels).
- Main purpose:
“Denoising” to compensate for variability in coverage.
-> should use normals made using the same data generation techniques (e.g same capture kit for exomes, same sequencing platform, etc.)

References

- Robinson, P.N., Piro, R.M., & Jager, M. (2017). Computational Exome and Genome Analysis (1st ed.). Chapman and Hall/CRC. <https://doi.org/10.1201/9781315154770>
- <https://gatk.broadinstitute.org/hc/en-us>
- Official Github of GATK : <https://github.com/broadinstitute/gatk>
- Germline short variant discovery:
<https://gatk.broadinstitute.org/hc/en-us/articles/360035535932-Germline-short-variant-discovery-SNPs-Indels->
- Somatic short variant discovery:
<https://gatk.broadinstitute.org/hc/en-us/articles/360035894731-Somatic-short-variant-discovery-SNVs-Indels->

A close-up photograph of a bundle of optical fibers. The fibers are thin, dark strands that radiate outwards from a central point, creating a starburst effect. At the end of each fiber is a small, bright, glowing white dot, representing light being transmitted through the fiber. The background is a deep, solid black, making the bright light at the ends of the fibers stand out sharply.

THANK YOU!