Introduction to Variant Discovery

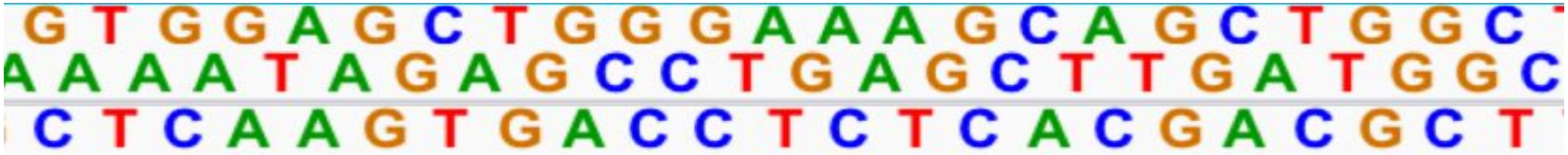
Basic concepts, variant types

and respective workflows

Presenter: Nguyen Le Duc Minh, MD

https://gatk.broadinstitute.org/hc/en-us

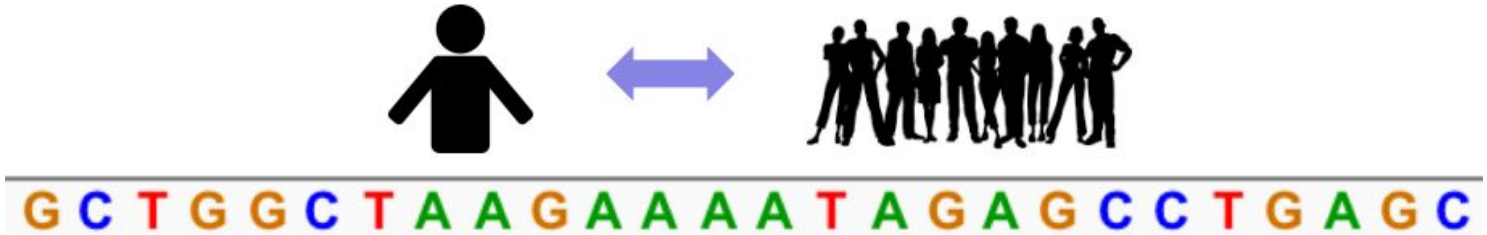
Human genomic variation



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

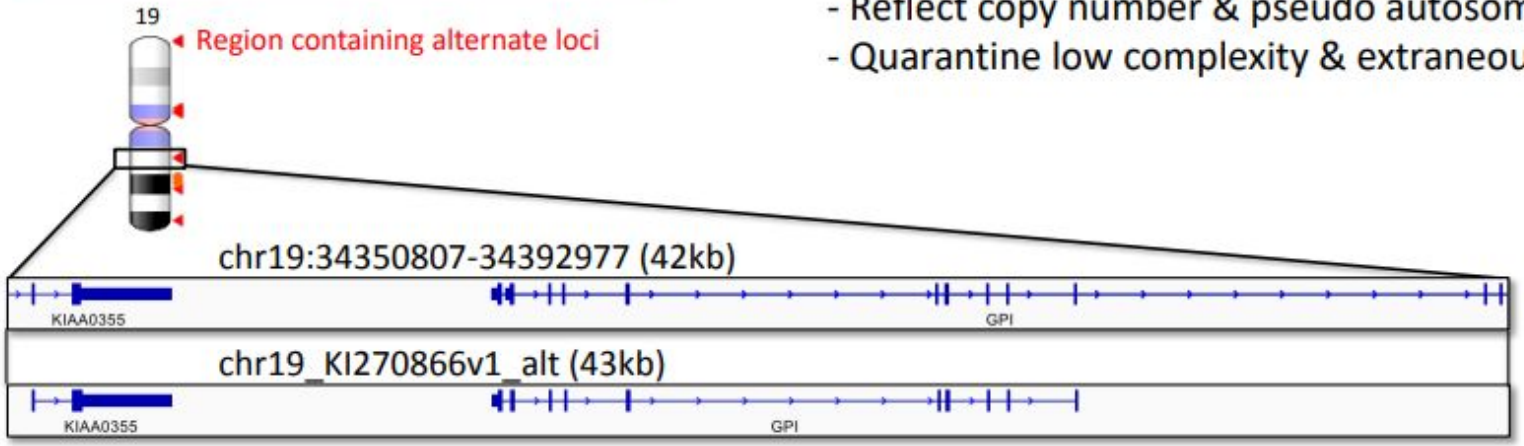


Human genome reference build **GRCh38**

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

http://www.ncbi.nlm.nih.gov/projects/genome/assembly/grc/human/

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.

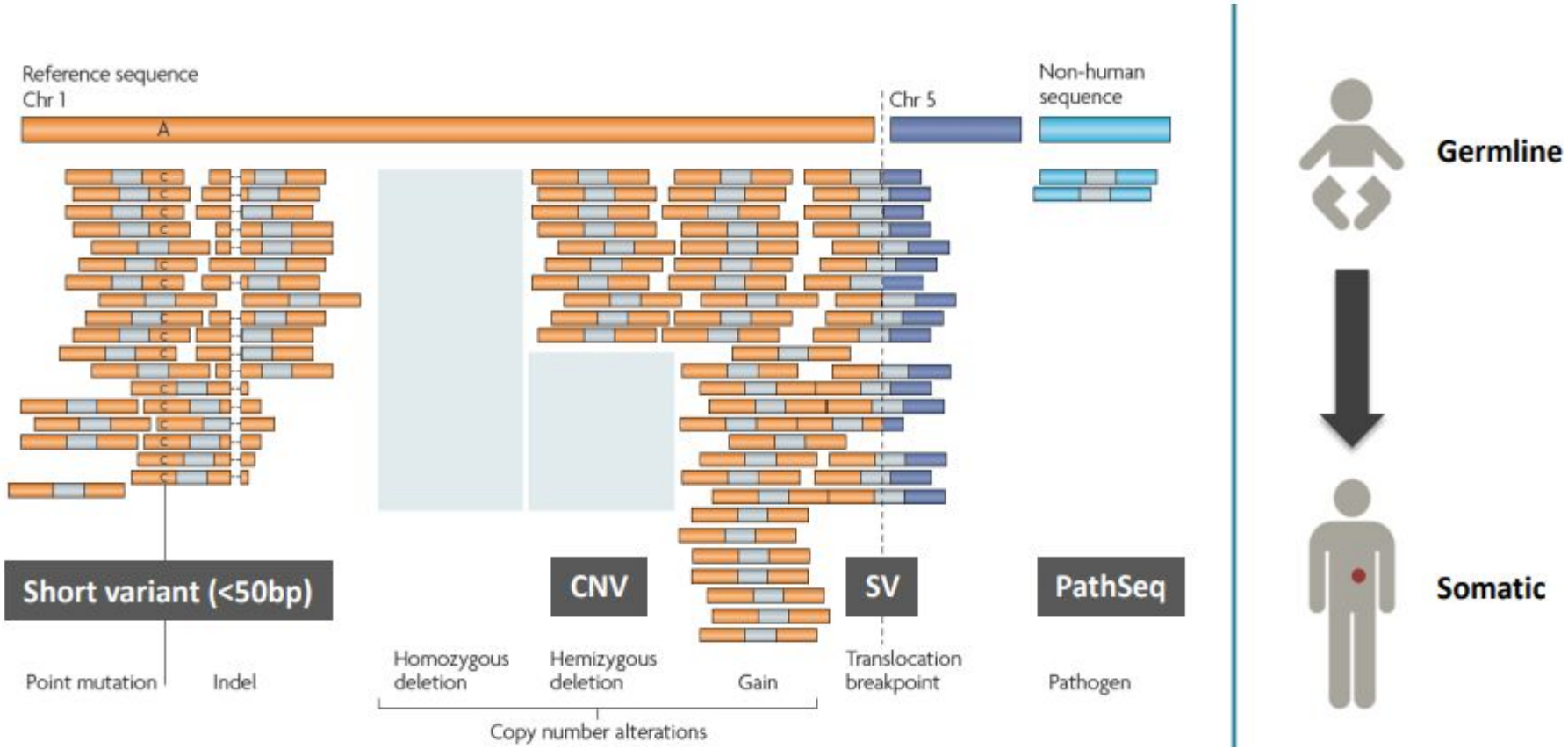
Four main types of variants:

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

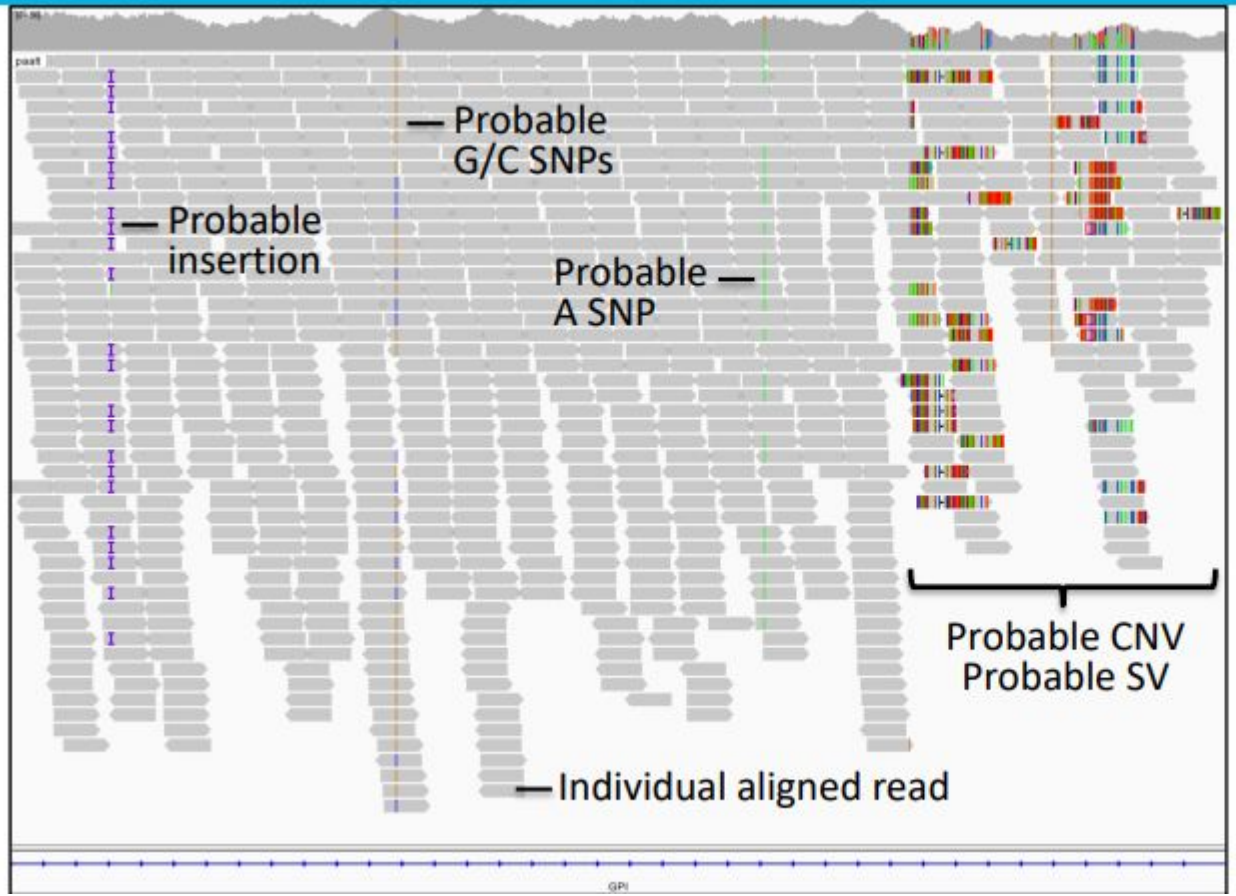
2 main types of variant classifications:

1. Germline variants
2. Somatic variants

Different types of genomic variants



What variants look like in a genome browser



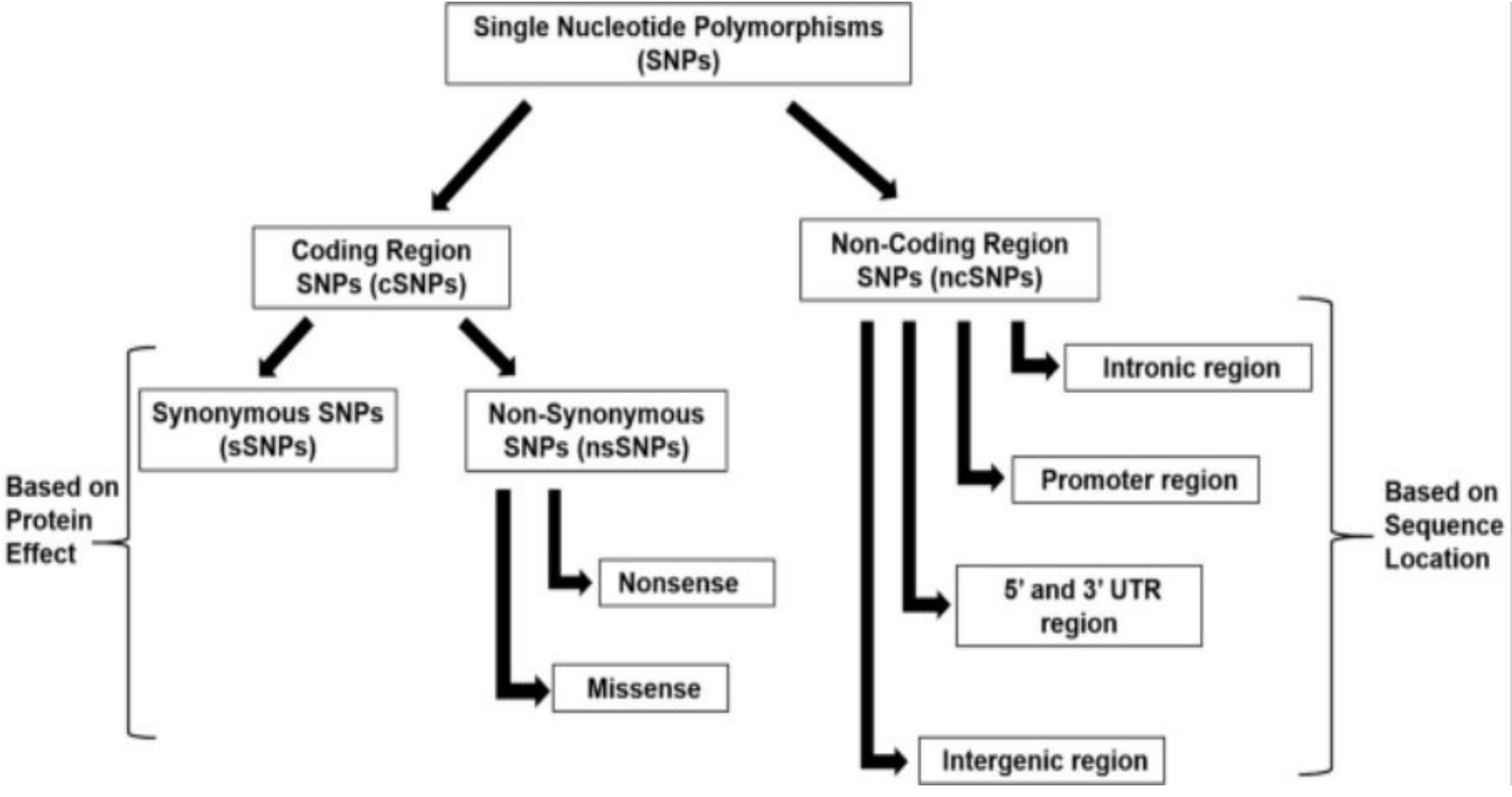
Top track: Depth of coverage

Non-reference bases are colored;
reference bases are grey

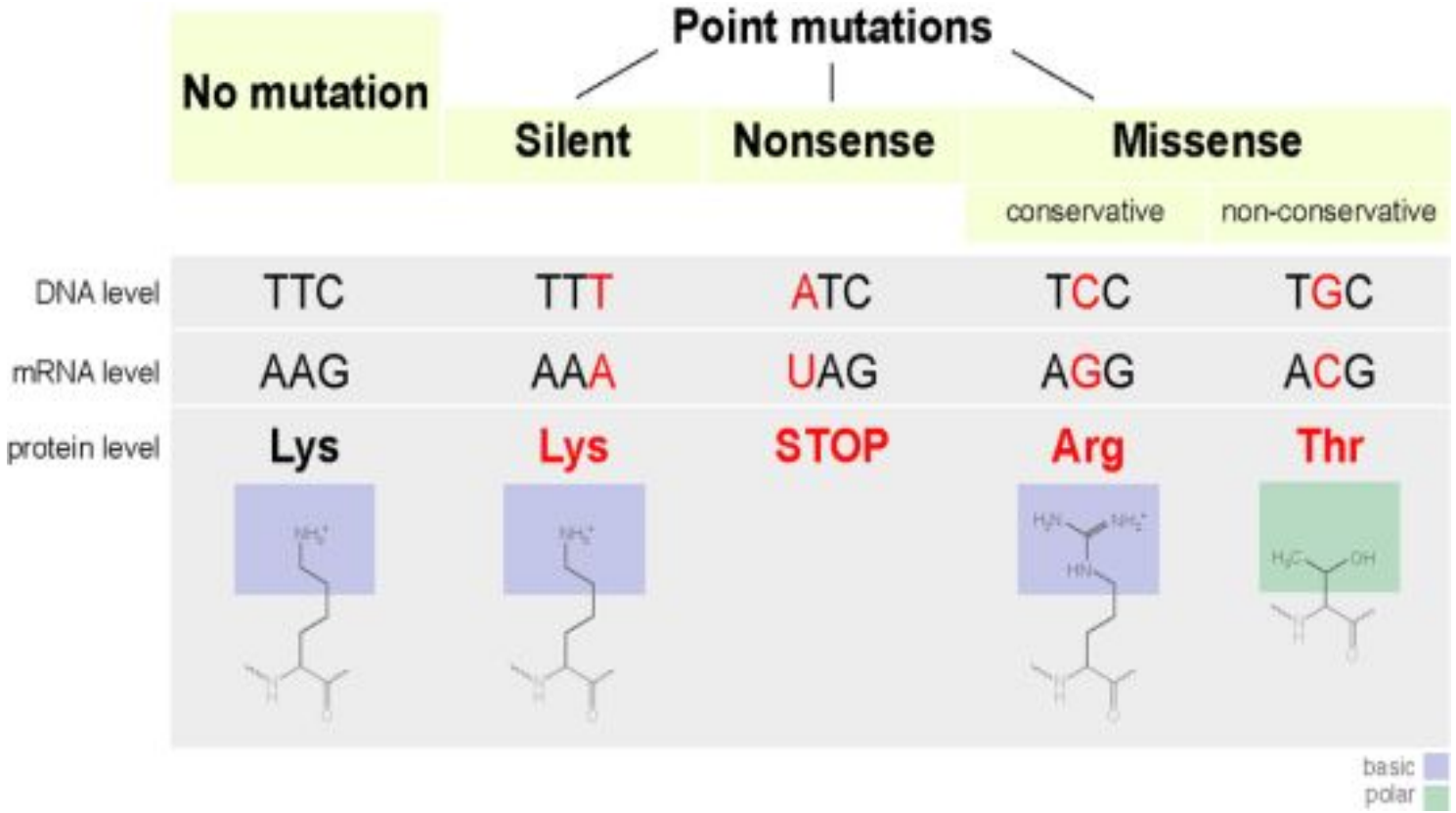
A C G T

Reference genome

SNPs classifications

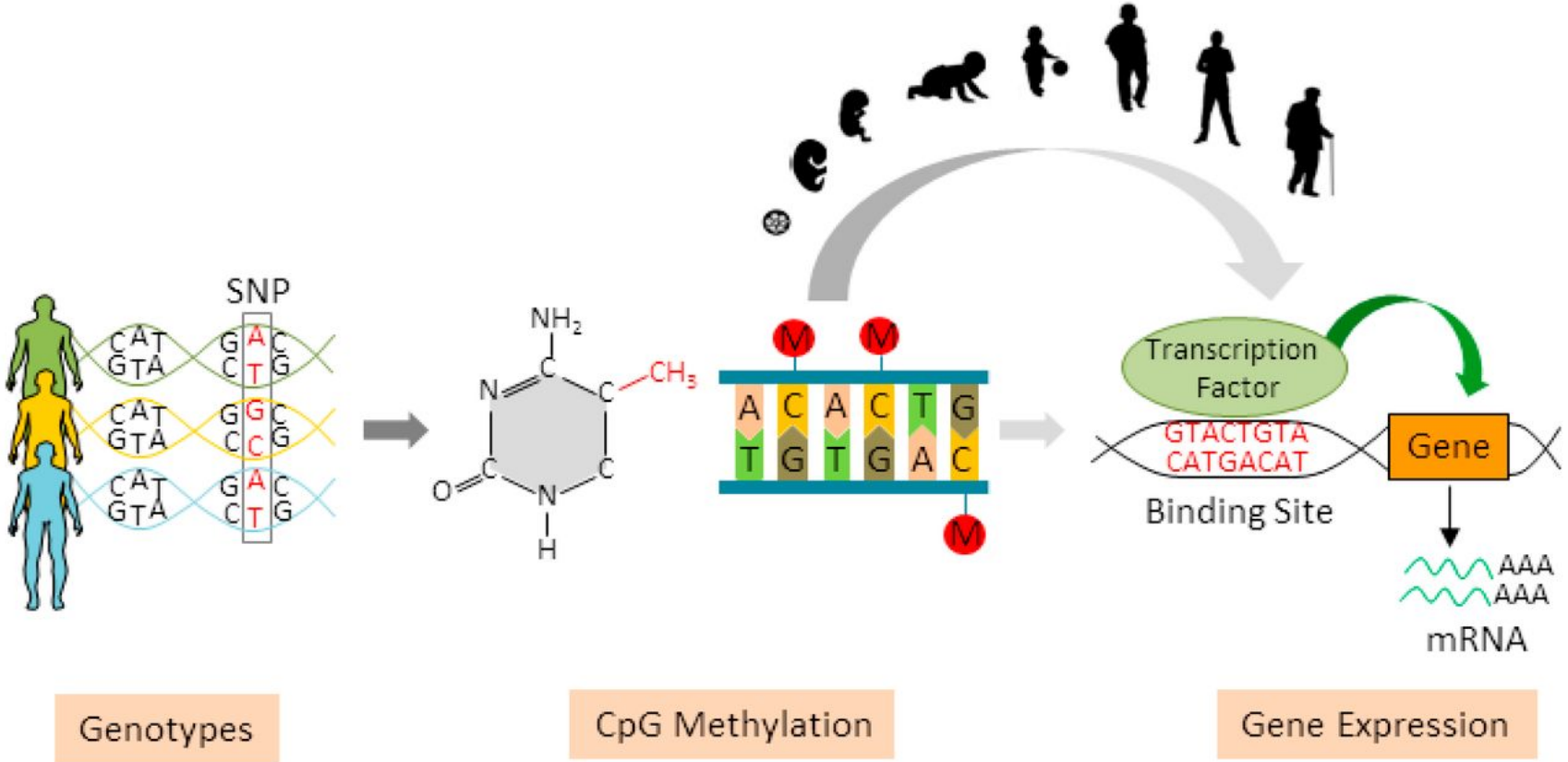


SNP/SNV



https://www.differencebetween.com/what-is-the-difference-between-point-mutations-and-indels/

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

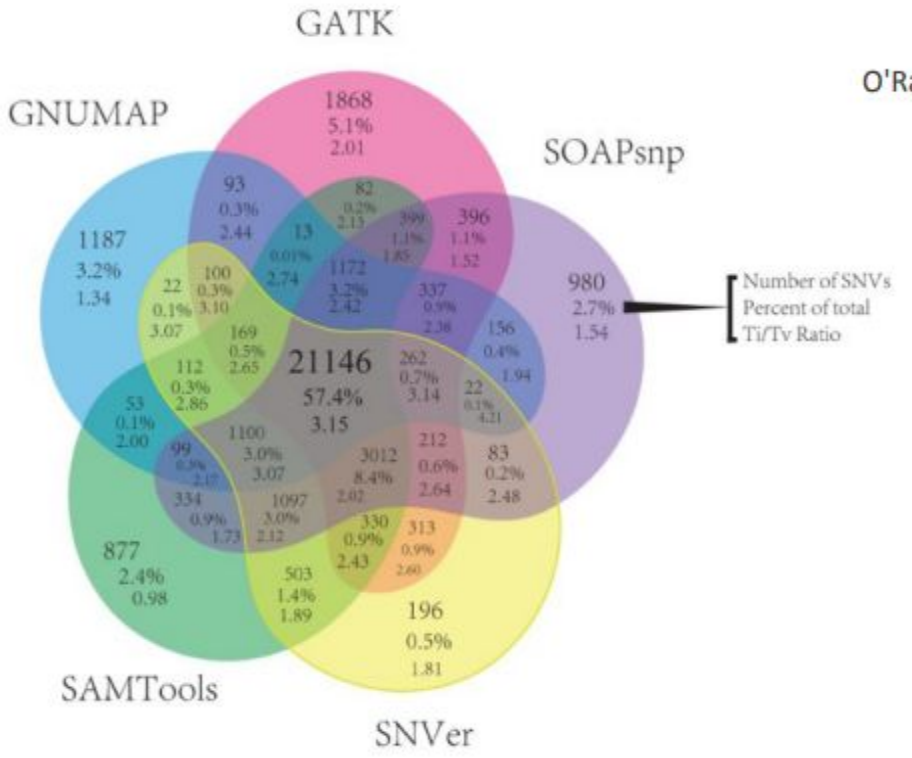


https://en.wikipedia.org/wiki/Single-nucleotide_polymorphism

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



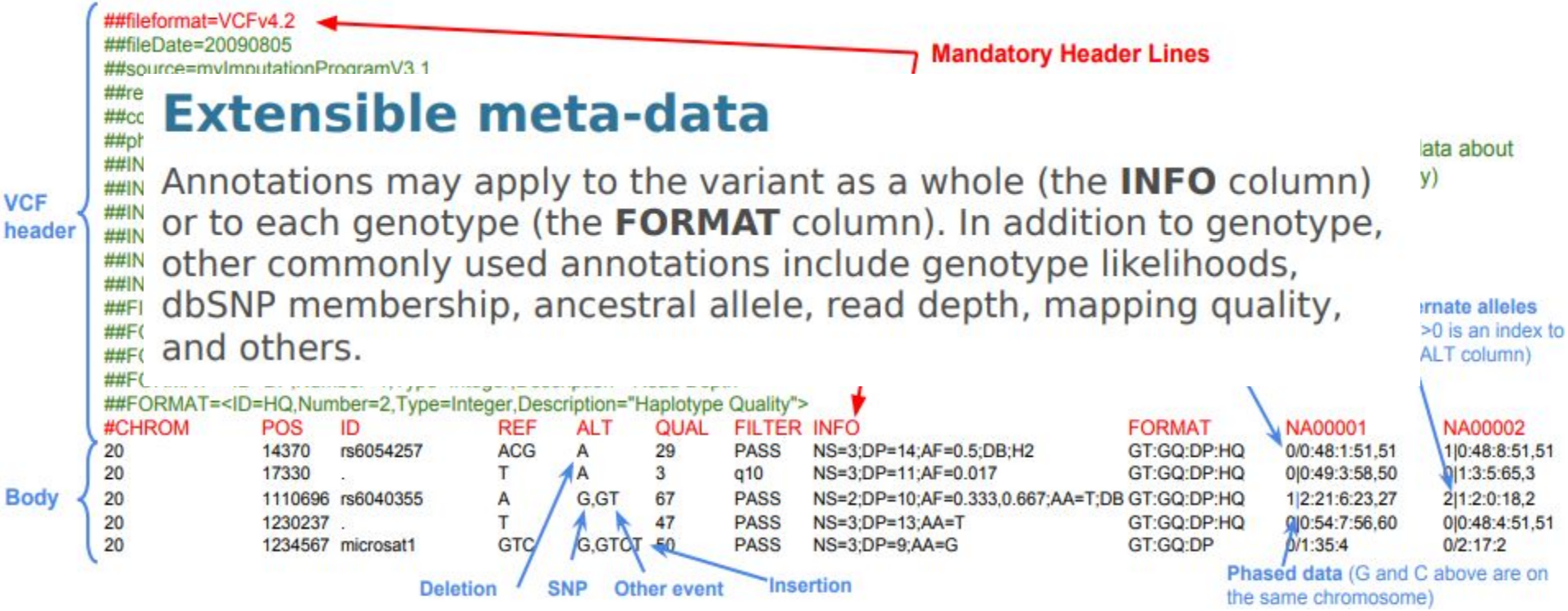
O'Rawe et al., Genome medicine 2013

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.



VCF: Variant Call Format (2)

Types of variants

SNPs

Alignment
```
ACGT
ATGT
```

VCF representation
```
POS REF ALT
2   C   T
```

Insertions

Alignment
```
AC-GT
ACTGT
```

VCF representation
```
POS REF ALT
2   C   CT
```

Deletions

Alignment
```
ACGT
A--T
```

VCF representation
```
POS REF ALT
1   ACG A
```

Complex events

Alignment
```
ACGT
A-TT
```

VCF representation
```
POS REF ALT
1   ACG AT
```

Large structural variants

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

http://vcftools.sourceforge.net/VCF-poster.pdf

VCF format supports CNVs and SVs

```
...
##INFO=<ID=BKPTID,Number=.,Type=String,Description="ID of the assembled alternate allele in the assembly fi
##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

https://samtools.github.io/hts-specs/VCFv4.2.pdf

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 • Reference • Alternate	 0/0 1/1, 2/2, 3/3, …
Missing	./0, ./1, ./., …

VCF specification versions

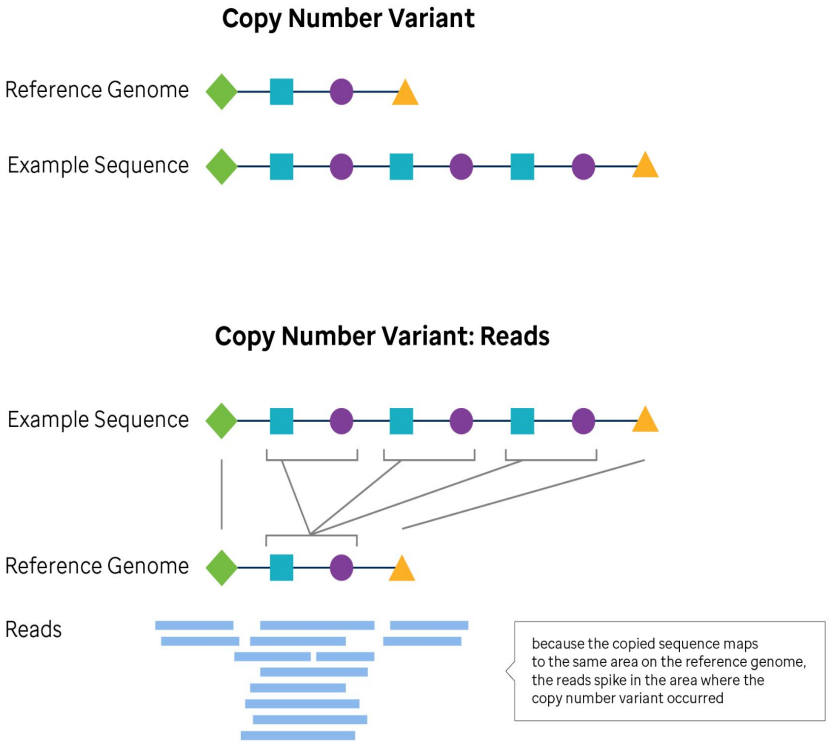
Changes between VCFv4.1 and VCFv4.2:

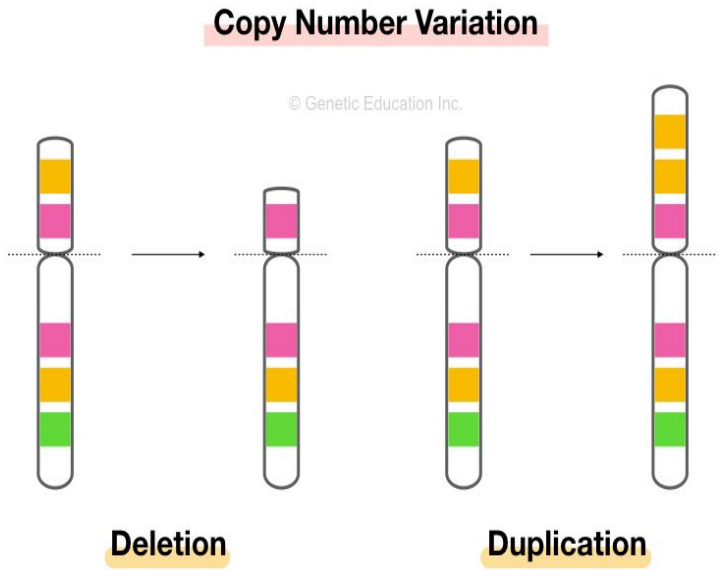
- Information field format: adding source and version as recommended fields.
- INFO field can have one value for each possible allele (code `R`).
- For all of the ##INFO, ##FORMAT, ##FILTER, and ##ALT metainformation, extra fields can be included after the default fields.
- Alternate base (ALT) can include `*`: 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)

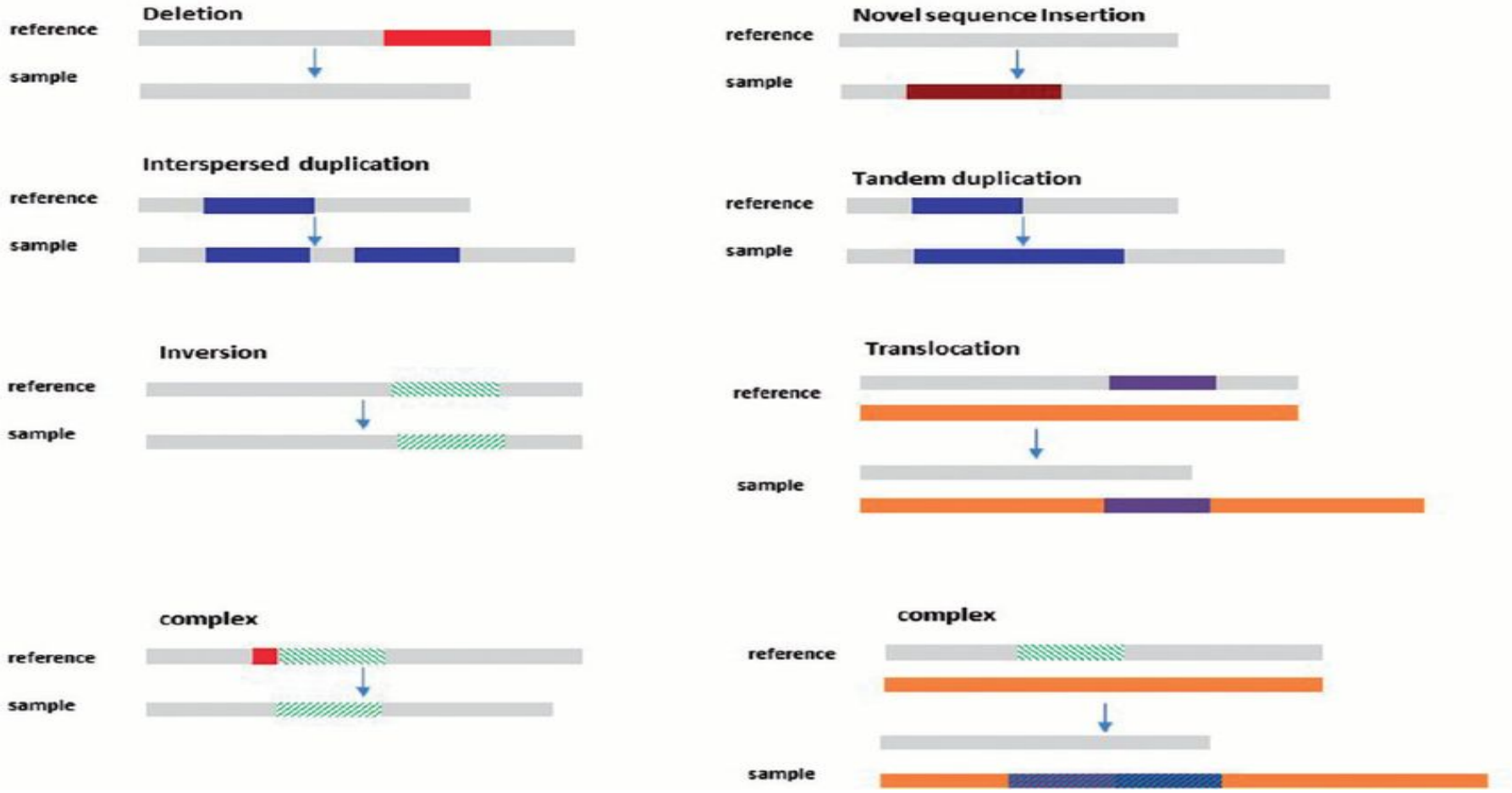




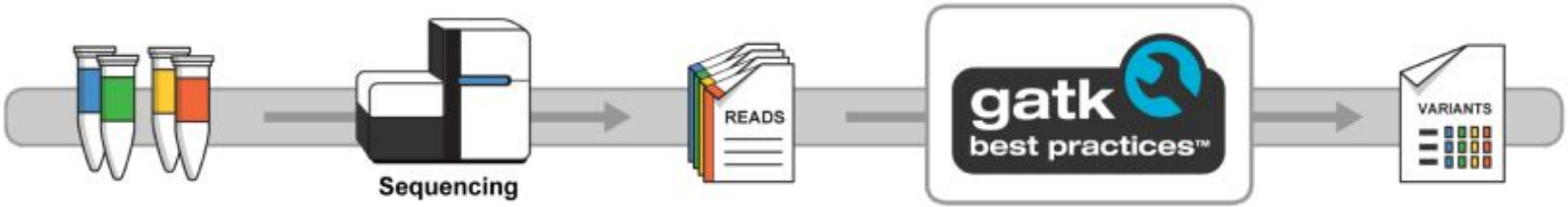
© Copyright 2022 St. Jude Children's Research Hospital, a not-for-profit, section 501(c)(3)

https://learngenomics.dev/docs/genomic-variation/copy-number-variation/

Structural variants



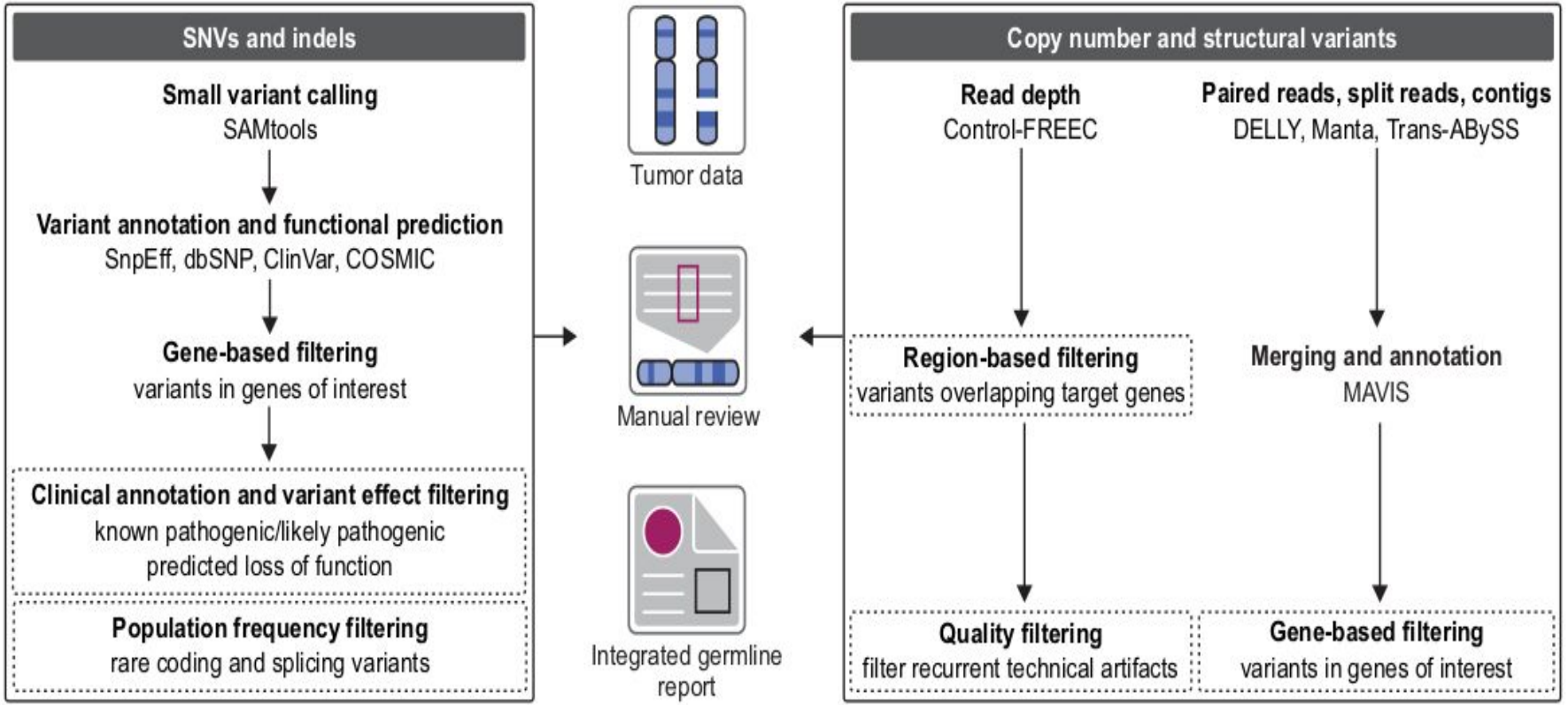
Workflows for all major variant classes



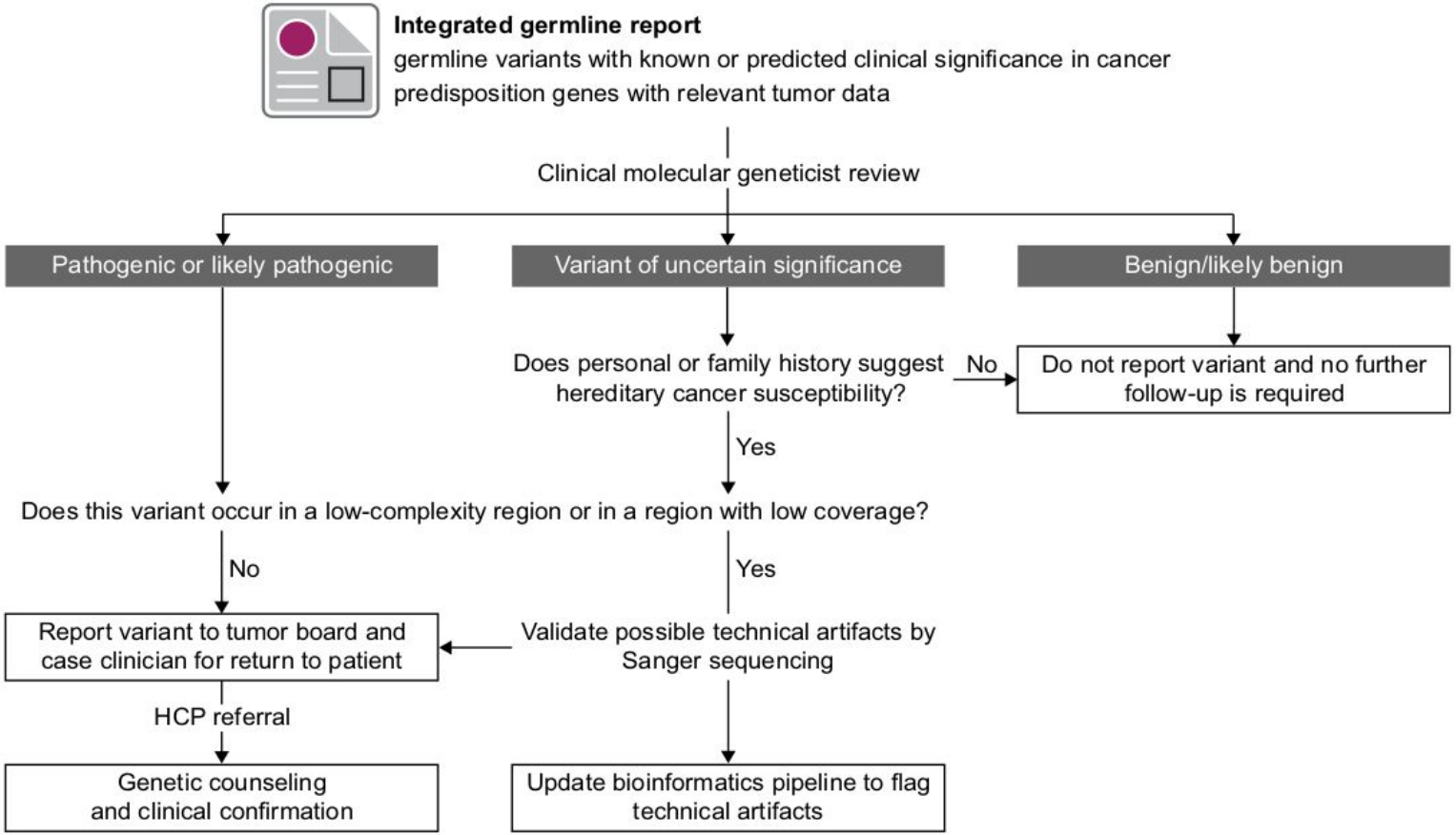
	GERMLINE	SOMATIC
SNPs & INDELs	HaplotypeCaller GVCF	Mutect2
Copy Number	GATK gCNV	GATK CNV + aCNV
Structure Variation	GATK SVDiscovery (beta)	(planned)

https://software.broadinstitute.org/gatk/best-practices

Workflow from variant calling to integrated report



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



Normal - Tumor paired variants calling

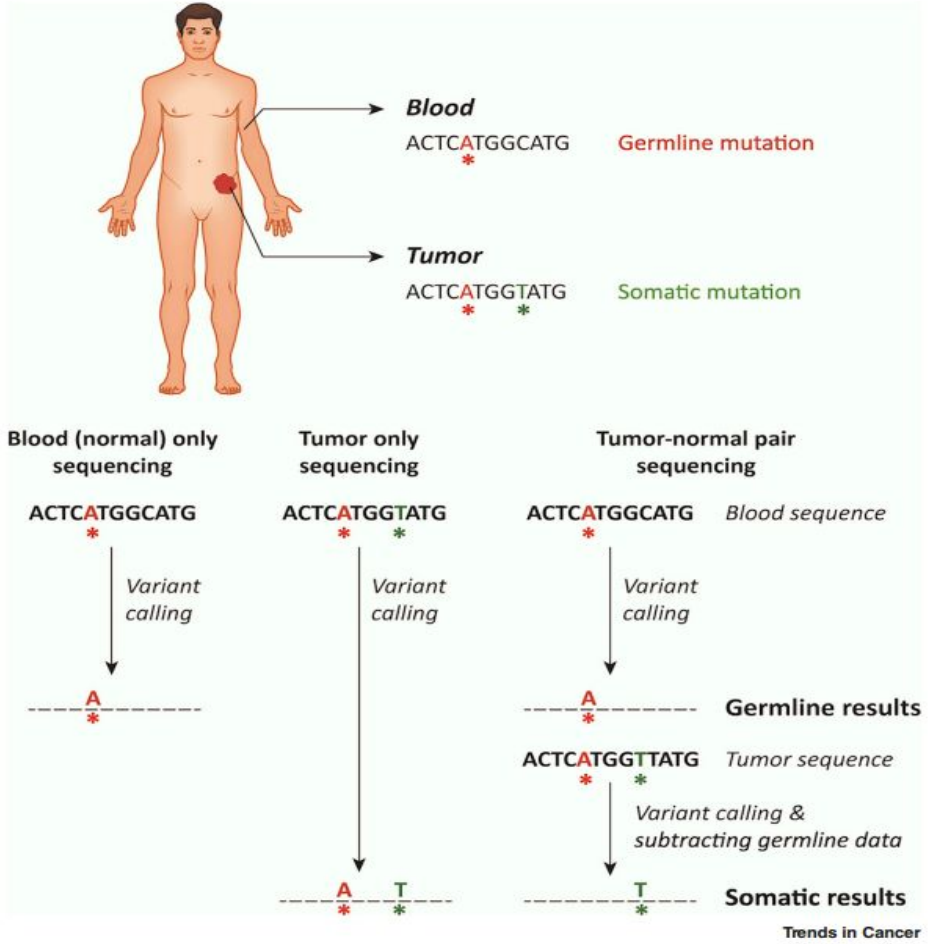
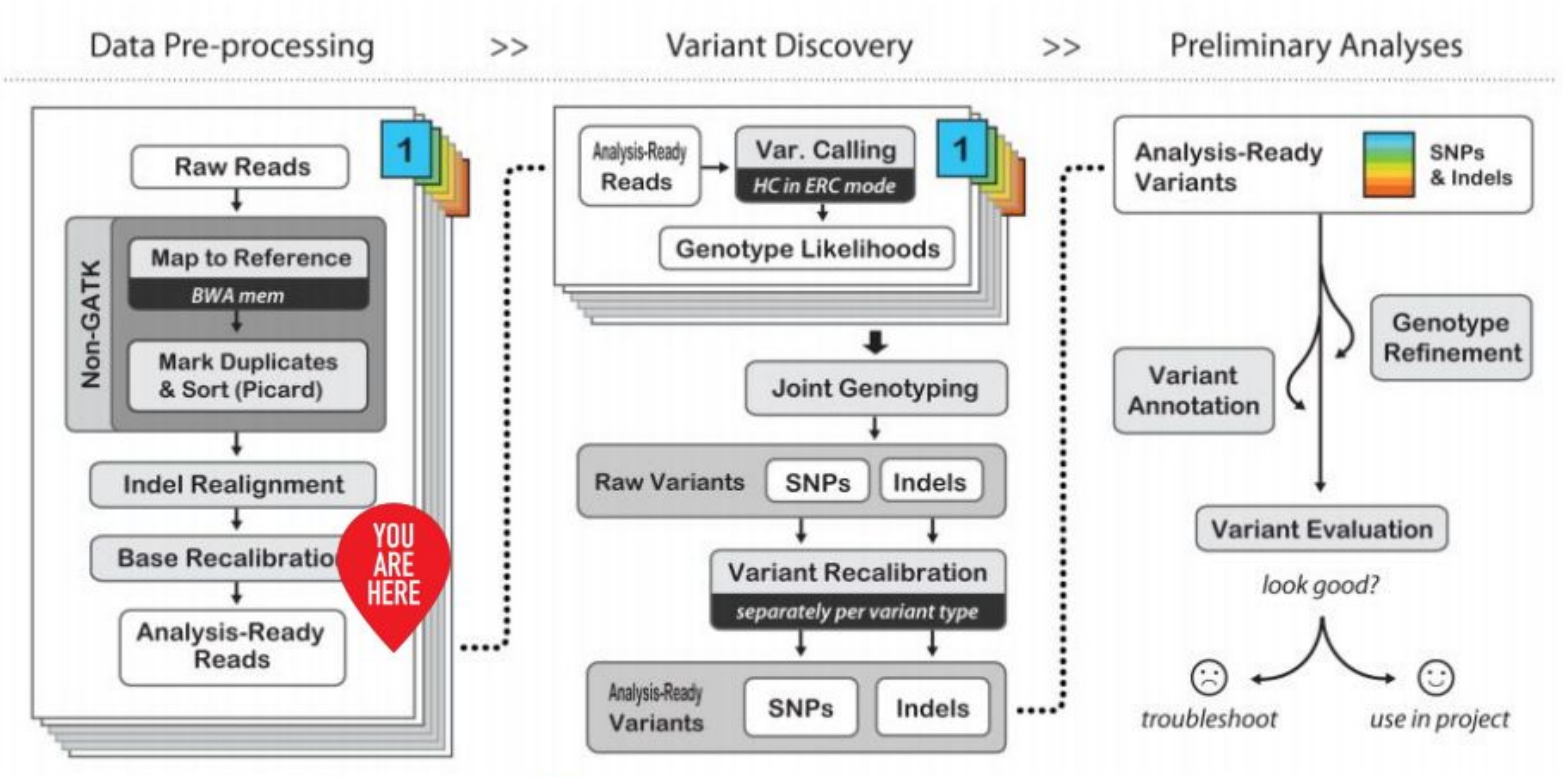


Figure 1. Mutations Reported in Blood-Only, Tumor-Only, and Paired Tumor-Normal Sequencing.

Workflow continues



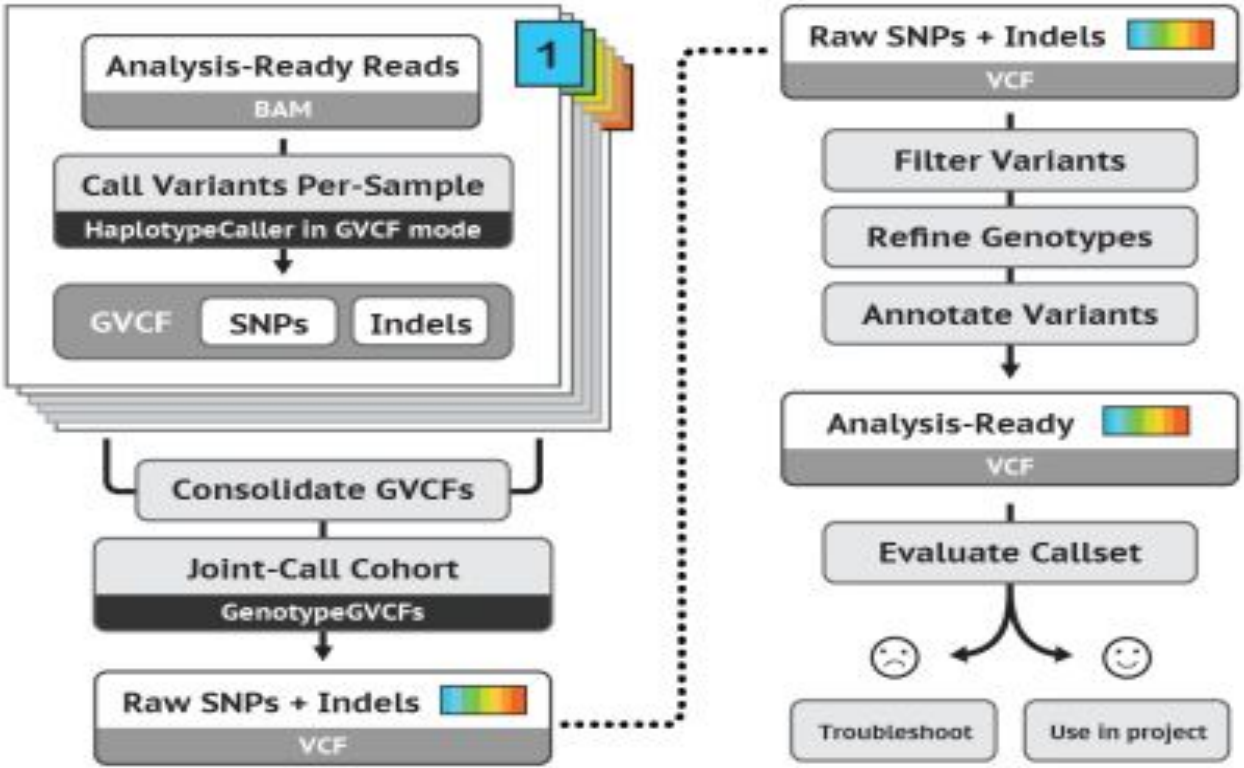
Ready for variant calling!!

Best Practices for Germline SNPs and Indels in Whole Genomes and Exomes - June 2016

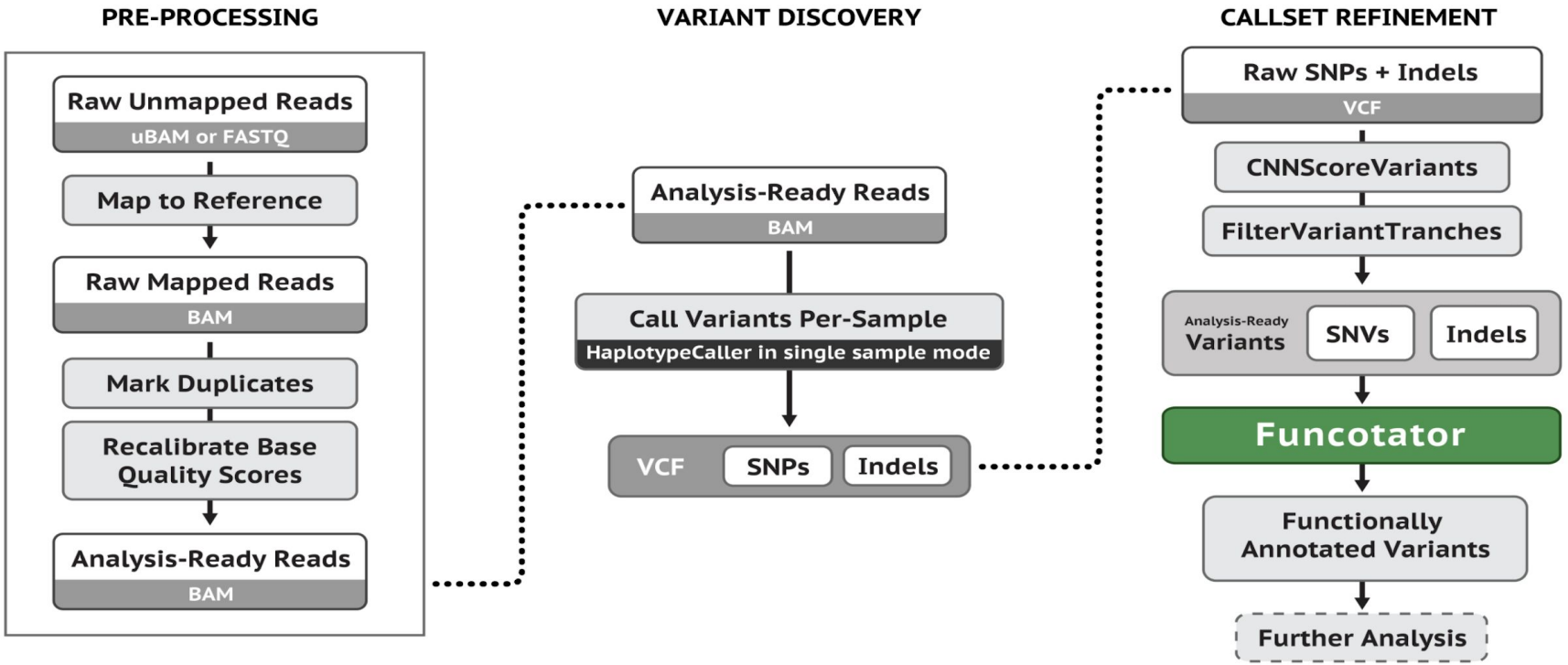
GERMLINE SNPs & INDELs

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

Main steps for Germline Cohort Data



Main steps for Germline Single-Sample Data

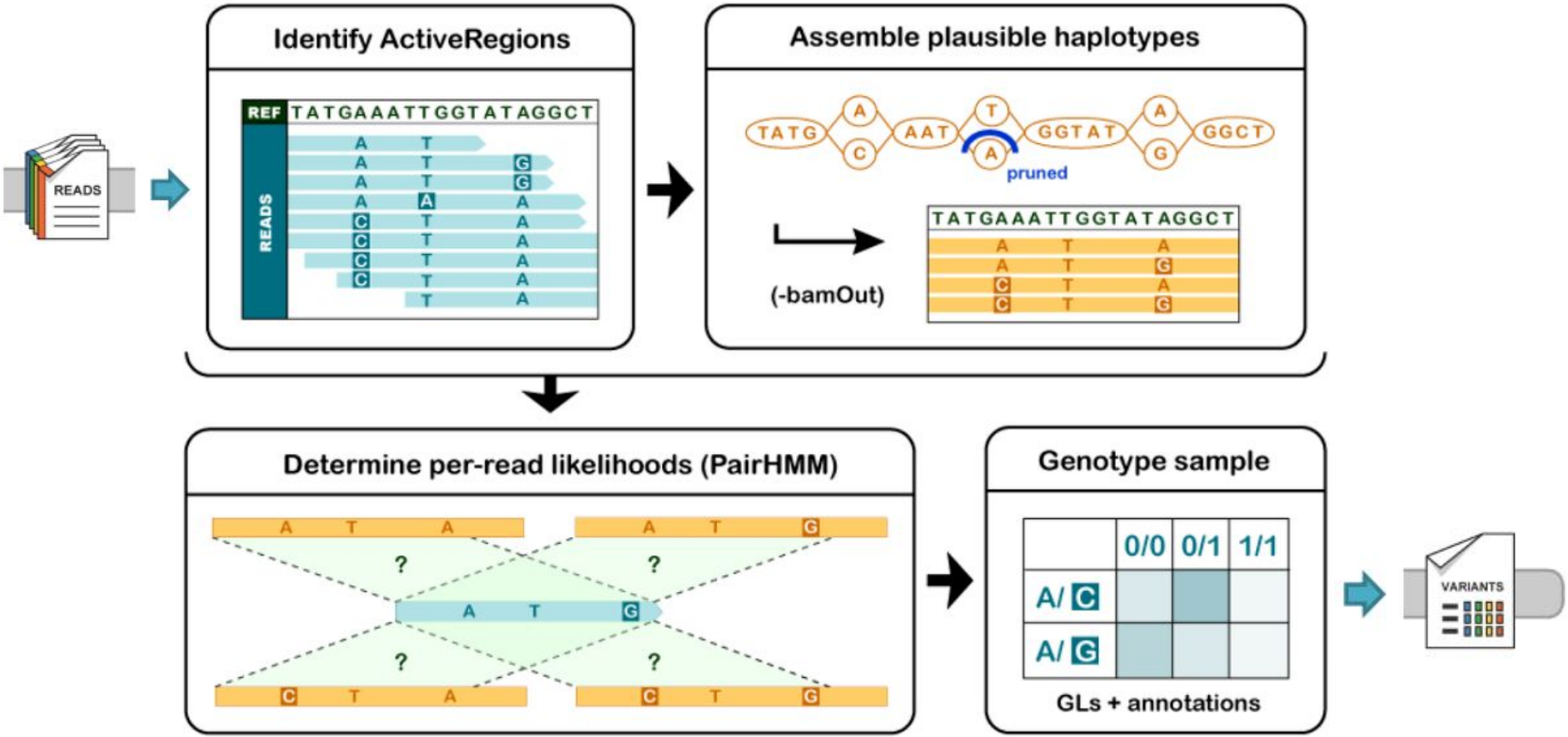


Variant calling with HaplotypeCaller

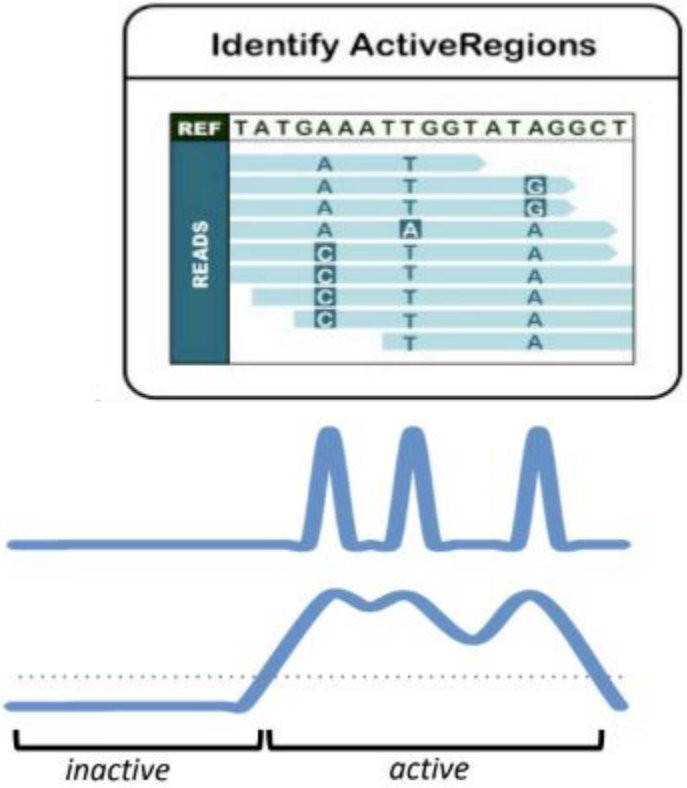
Basic operation and algorithm

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

GATK HaplotypeCaller calls germline short variants



1. Define Active Regions

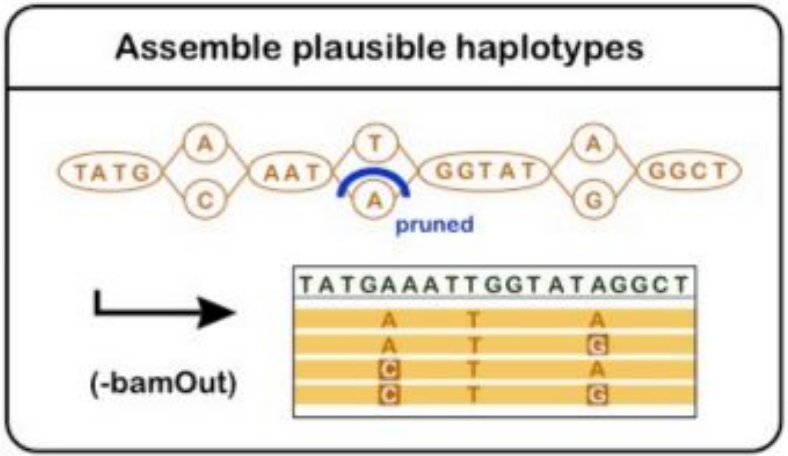


- Sliding window along the reference
- Count mismatches, indels and soft-clips
- Measure of entropy

Trim and continue with ActiveRegions over threshold

2. Assemble plausible haplotypes

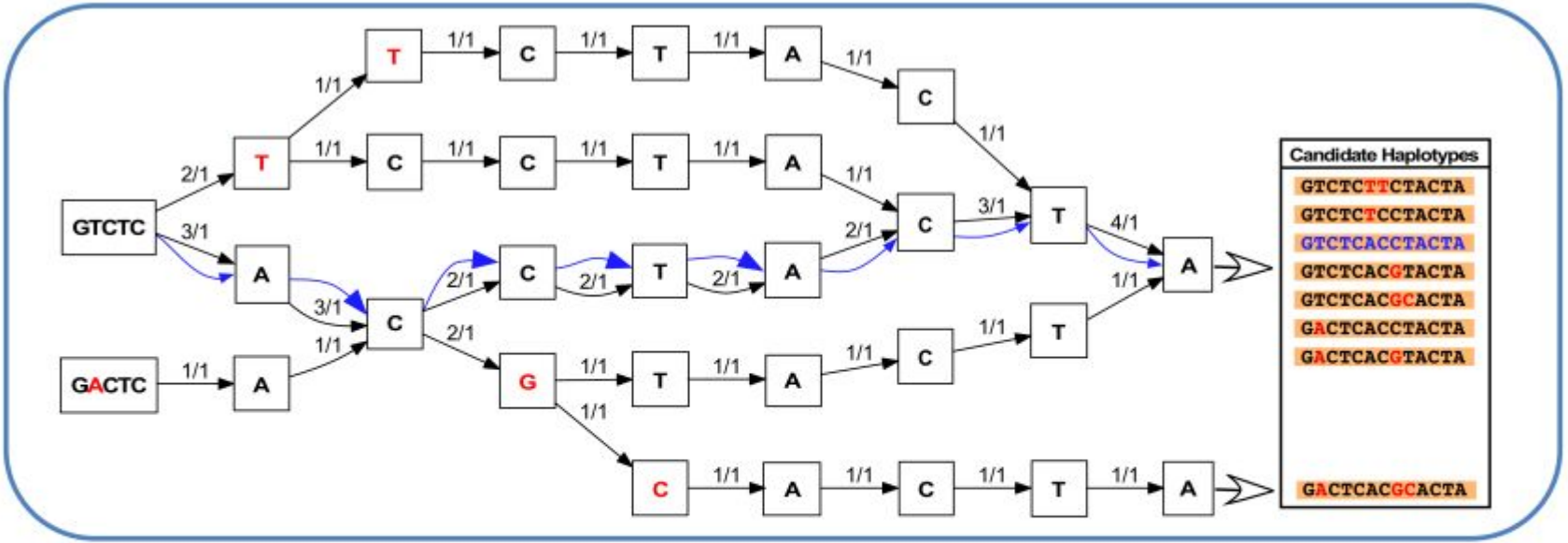
- Local realignment via graph assembly
- Traverse graph to collect most likely haplotypes
- Align haplotypes to reference using Smith-Waterman



Likely haplotypes + candidate variant sites

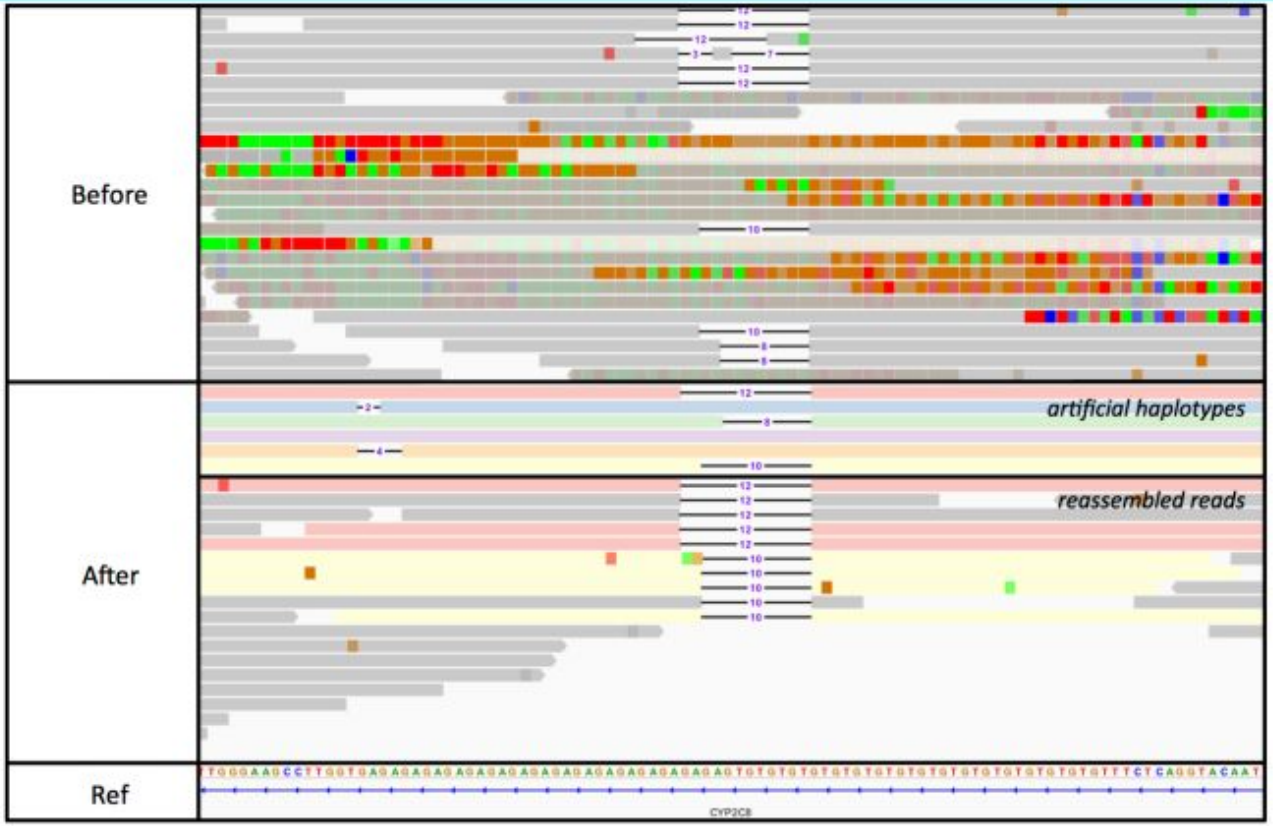
Can make HC output the reassembled reads and selected haplotypes using the –bamOut parameter

Example HaplotypeCaller assembly graph



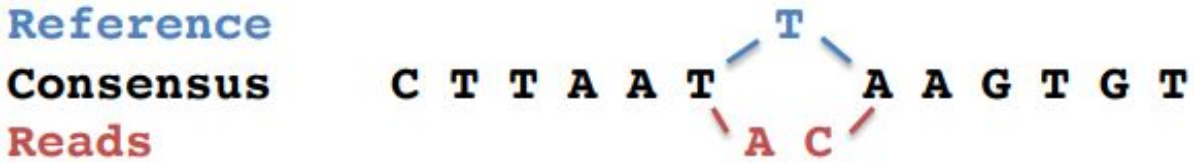
- Ignore previous alignments
- Graph consists of every possible sequence combination based on reads
- Count reads that support paths

Graph assembly recovers indels and removes artifacts

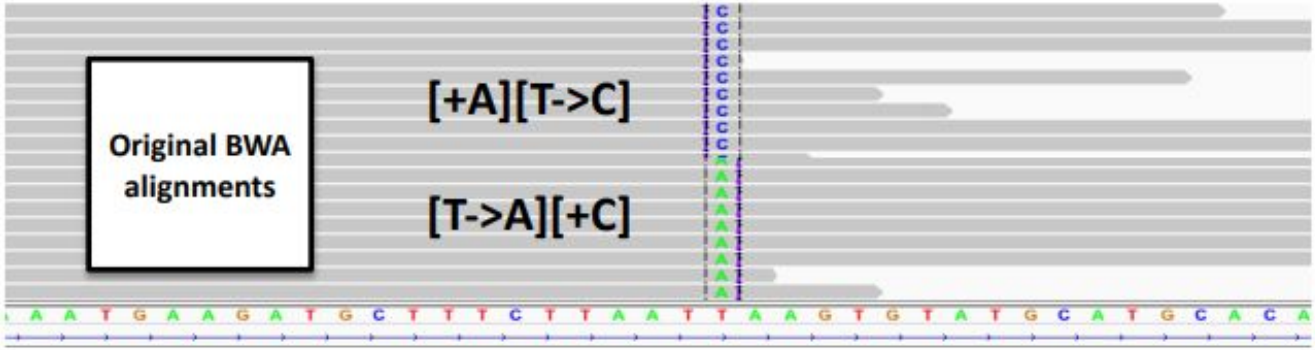


Showing 100bp region starting at 10:96,825,862 for NA12878

Resolves complexity caused by mapper limitations



- Mapper can represent two different ways, at random:



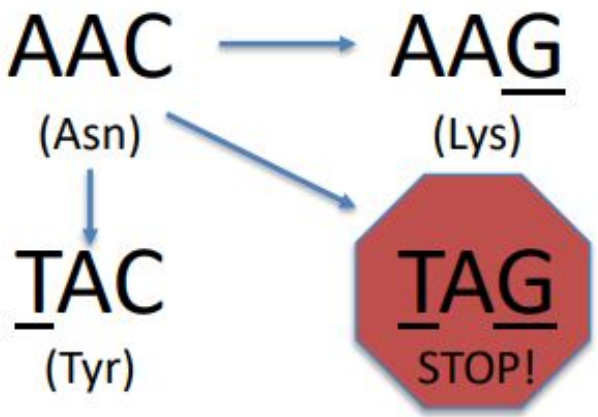
- HaplotypeCaller will settle on one representation -> cleaner output call

Functional implications of variant phasing

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

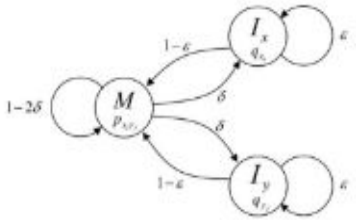
In trans – two copies, each with a missense mutation

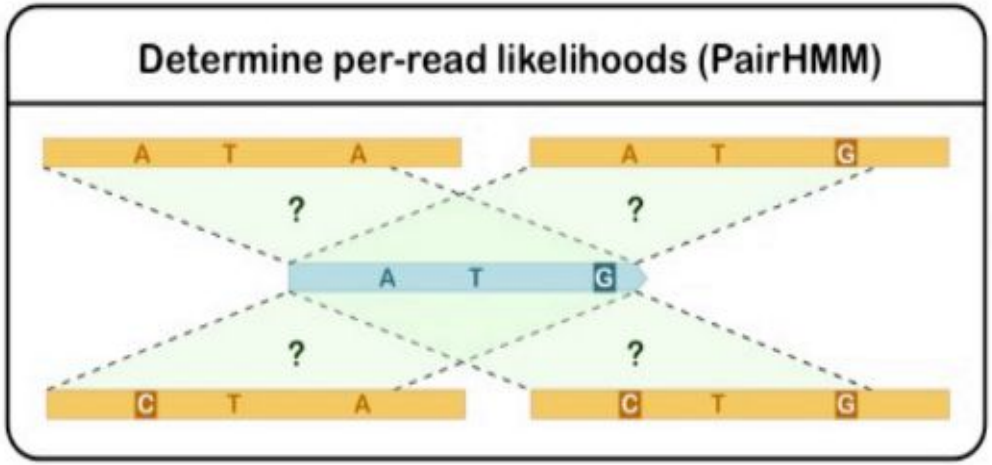
In cis – one functional copy and one loss of function!



3. Score haplotypes using PairHMM

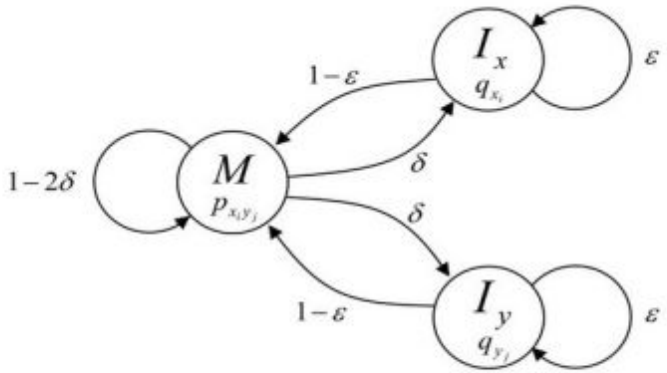
- PairHMM* aligns each read to each haplotype
- Uses base qualities as the estimate of error





→ **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 - ε) = Base precedes an insertion or a deletion

(1 - 2δ) = 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

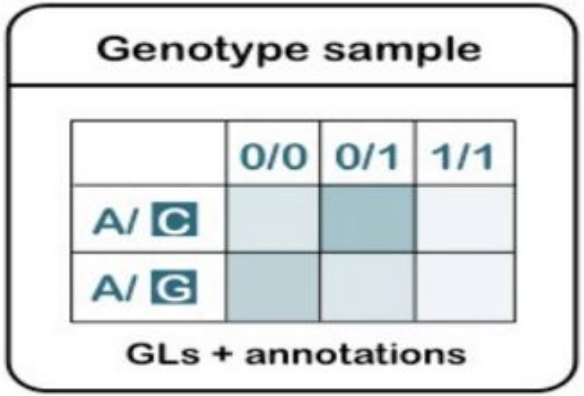
4. Genotype each sample at each potential variant site

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

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

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

$L(R_j \mid H_i)$ Read-haplotype likelihoods



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

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

Likelihood

Genotype prior (constant)

$$L(R \mid G_i) = \prod_j \left(\frac{L(R_j \mid H_1)}{2} + \frac{L(R_j \mid H_2)}{2} \right)$$

$G_i = H_1H_2$ for diploids

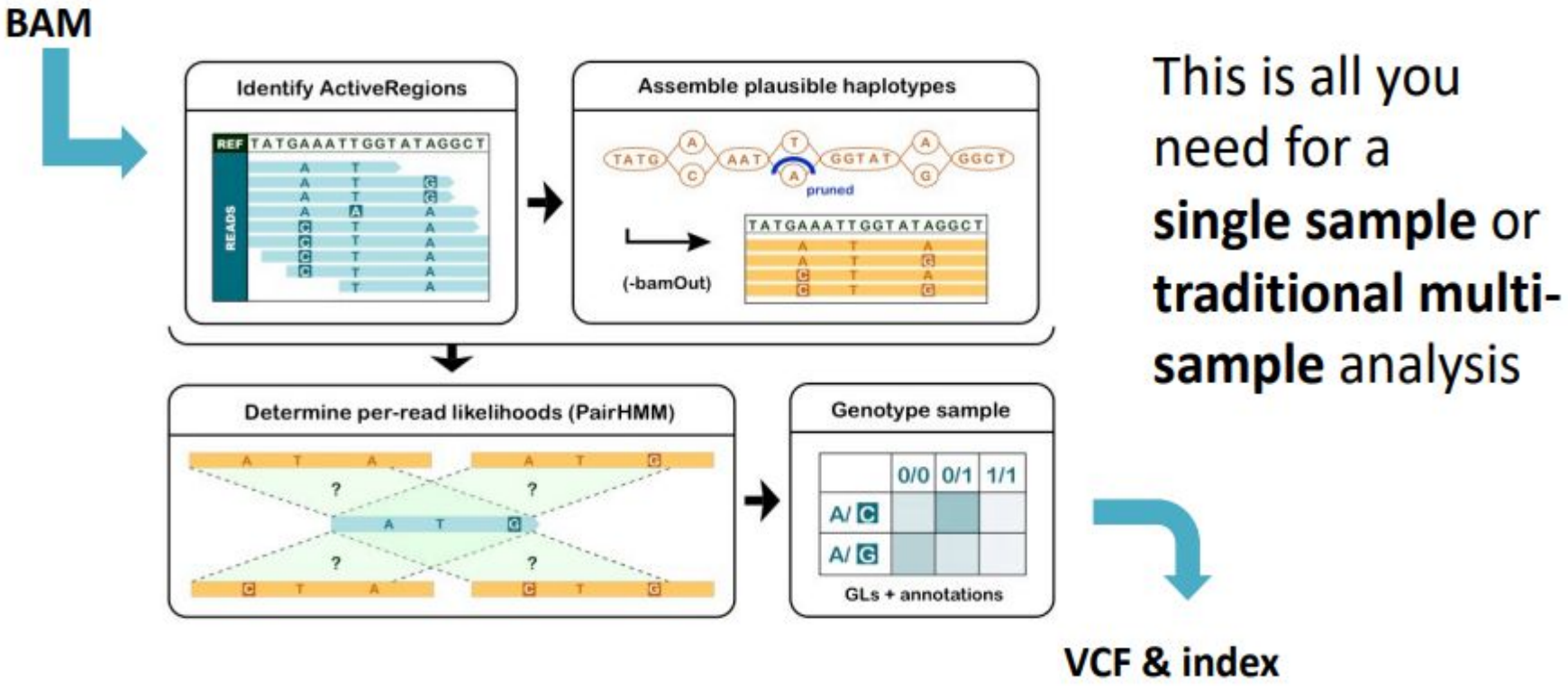
$L(R_j \mid H_i)$ Read-haplotype likelihoods

Plug in the numbers!

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

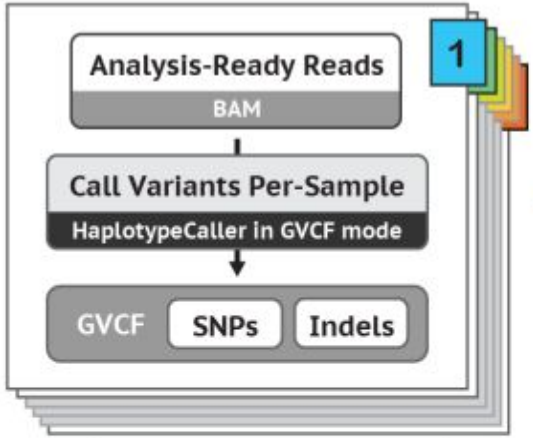
Determines the most likely genotype of the sample at each event in the haplotypes

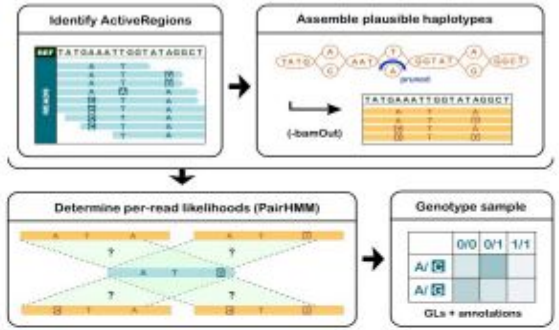
HaplotypeCaller recap: reads in / variants out



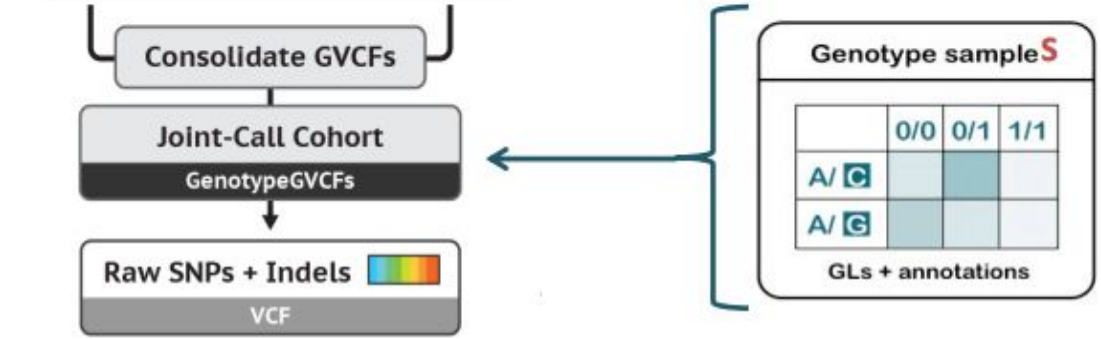
This is all you need for a **single sample** or **traditional multi-sample** analysis

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
```

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

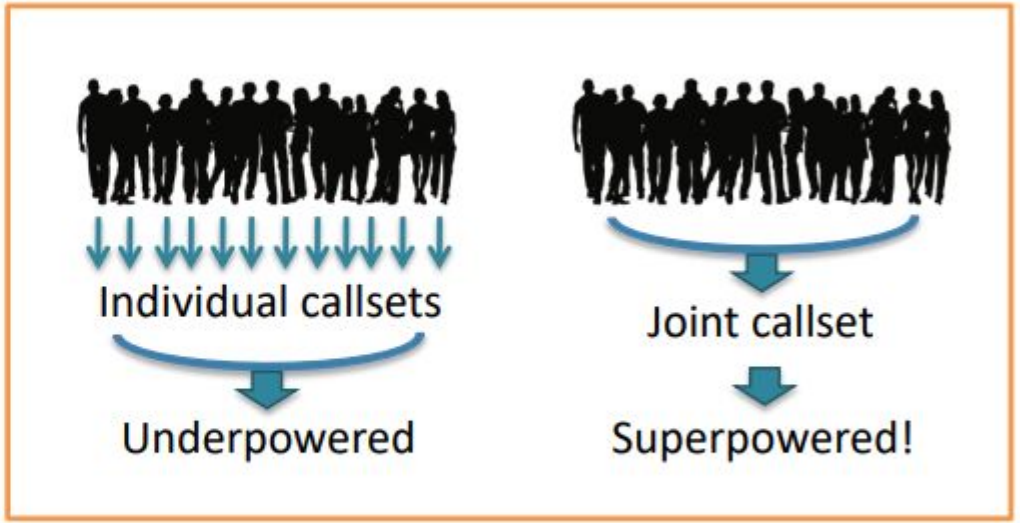
Joint variant calling

GVCF-based workflow

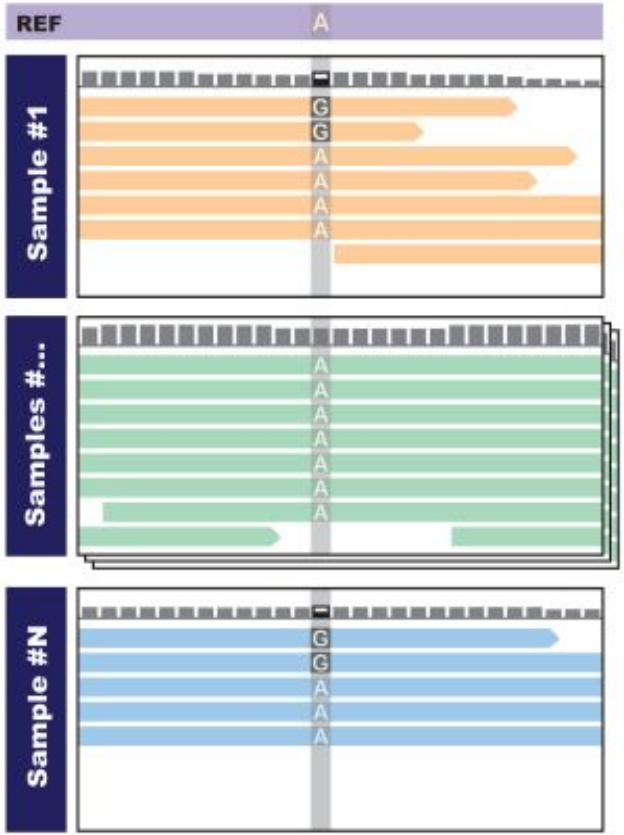
http://software.broadinstitute.org/gatk/

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

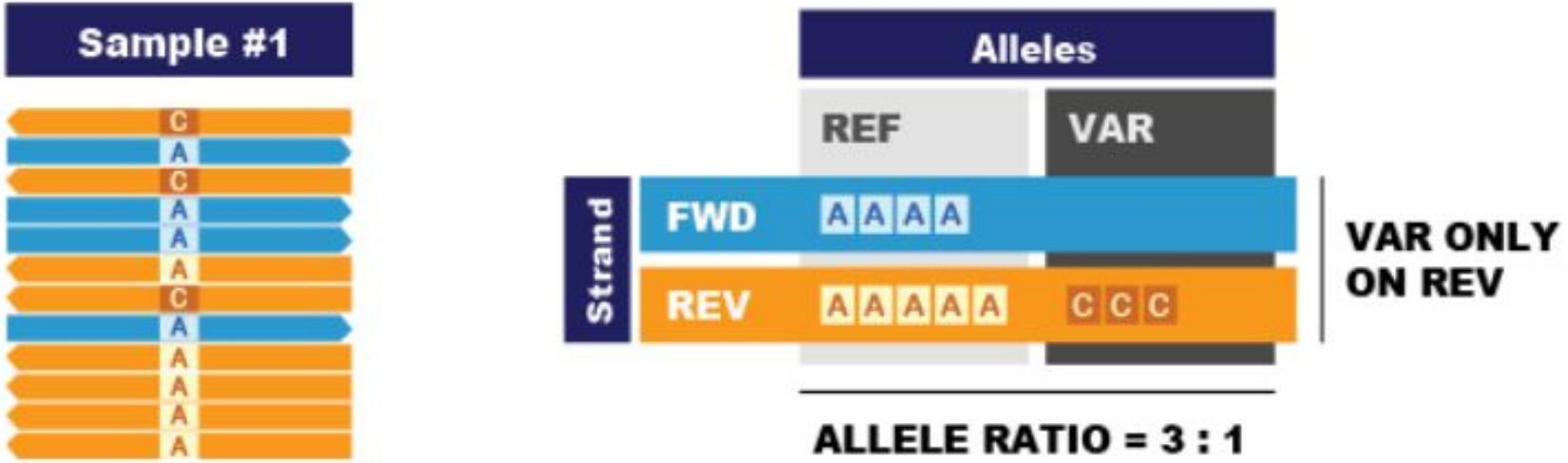


Discovery is empowered at difficult sites



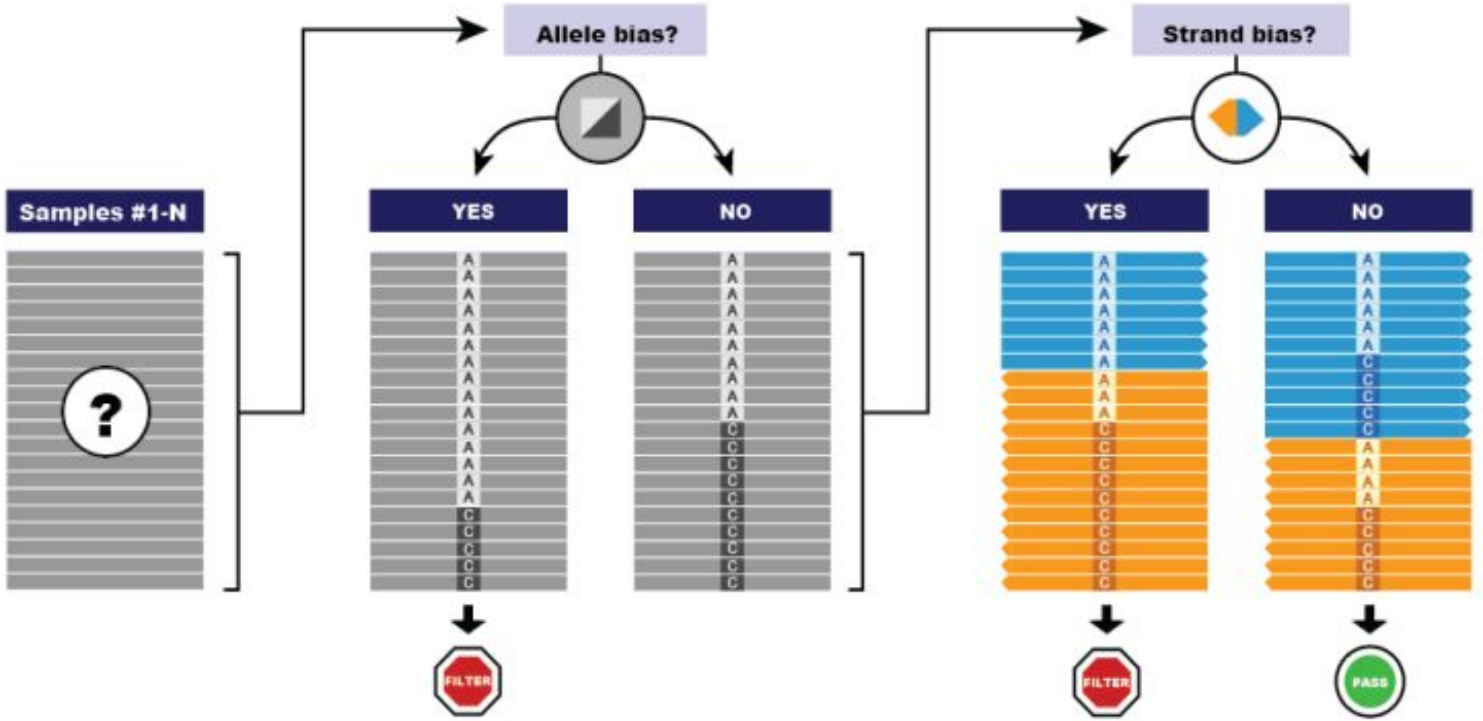
- Sample #1 or Sample #N alone:
  - **weak evidence for variant**
  - **may miss calling the variant**
- Both samples seen together:
  - **unlikely to be artifact**
  - **call the variant more confidently**

Joint analysis helps resolve bias issues (1)



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

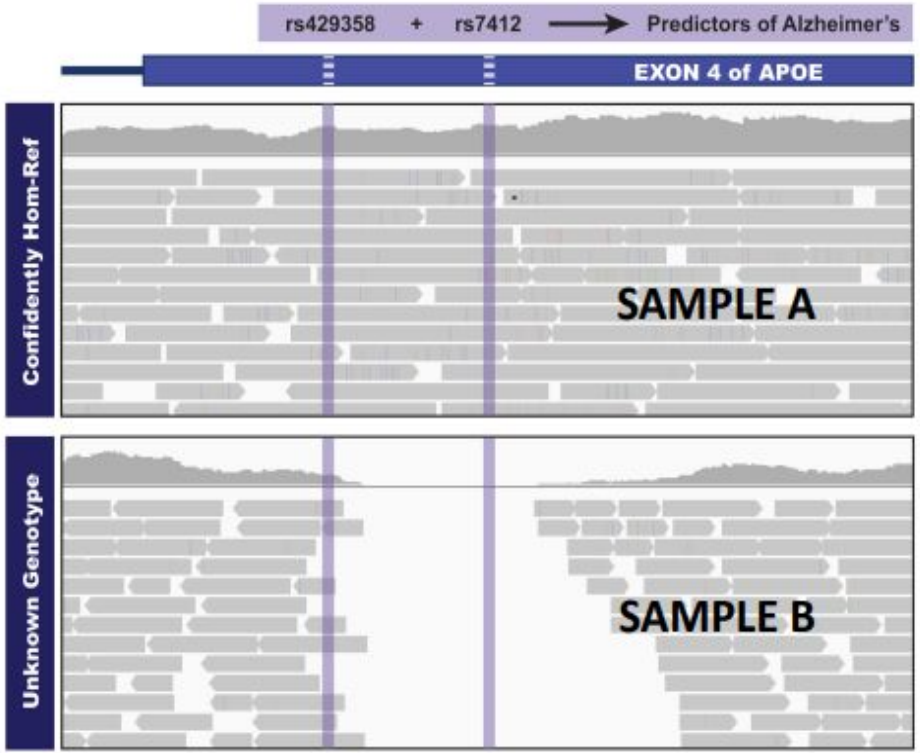
Joint analysis helps resolve bias issues (2)



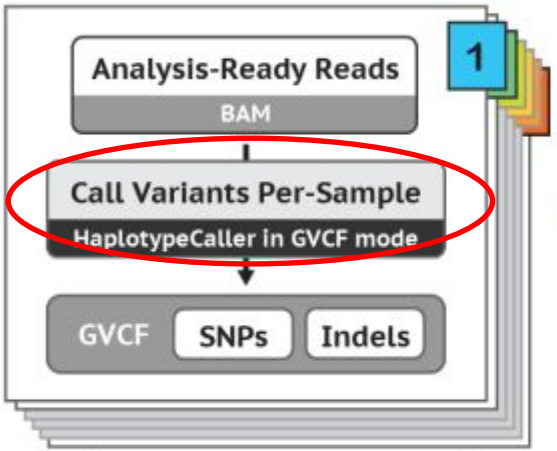
Decision process using evidence from multiple samples to filter out sites showing systematic biases

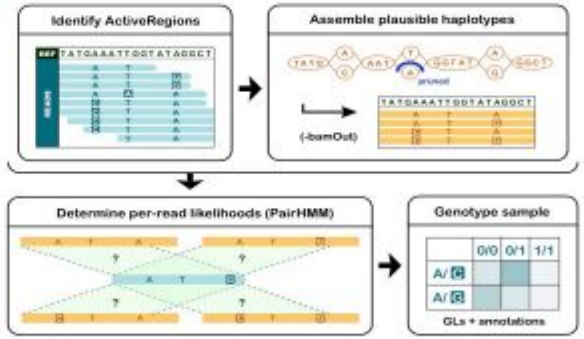
Gather full information at all sites of interest

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

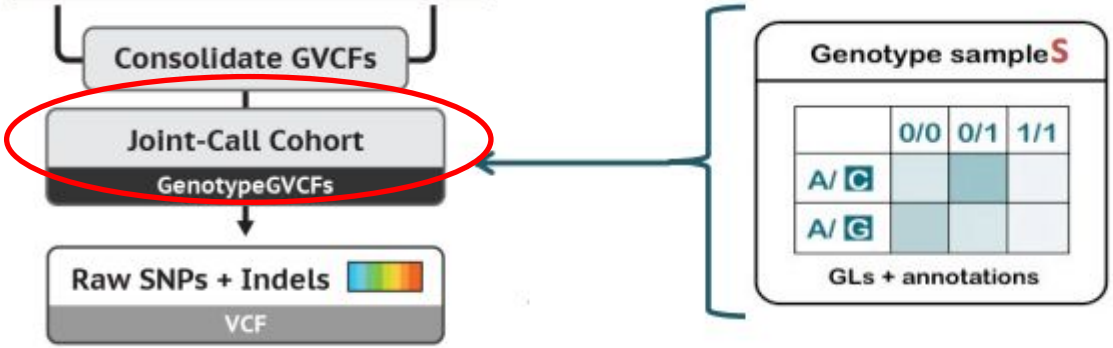


Joint calling implemented as a two-step process for scalability





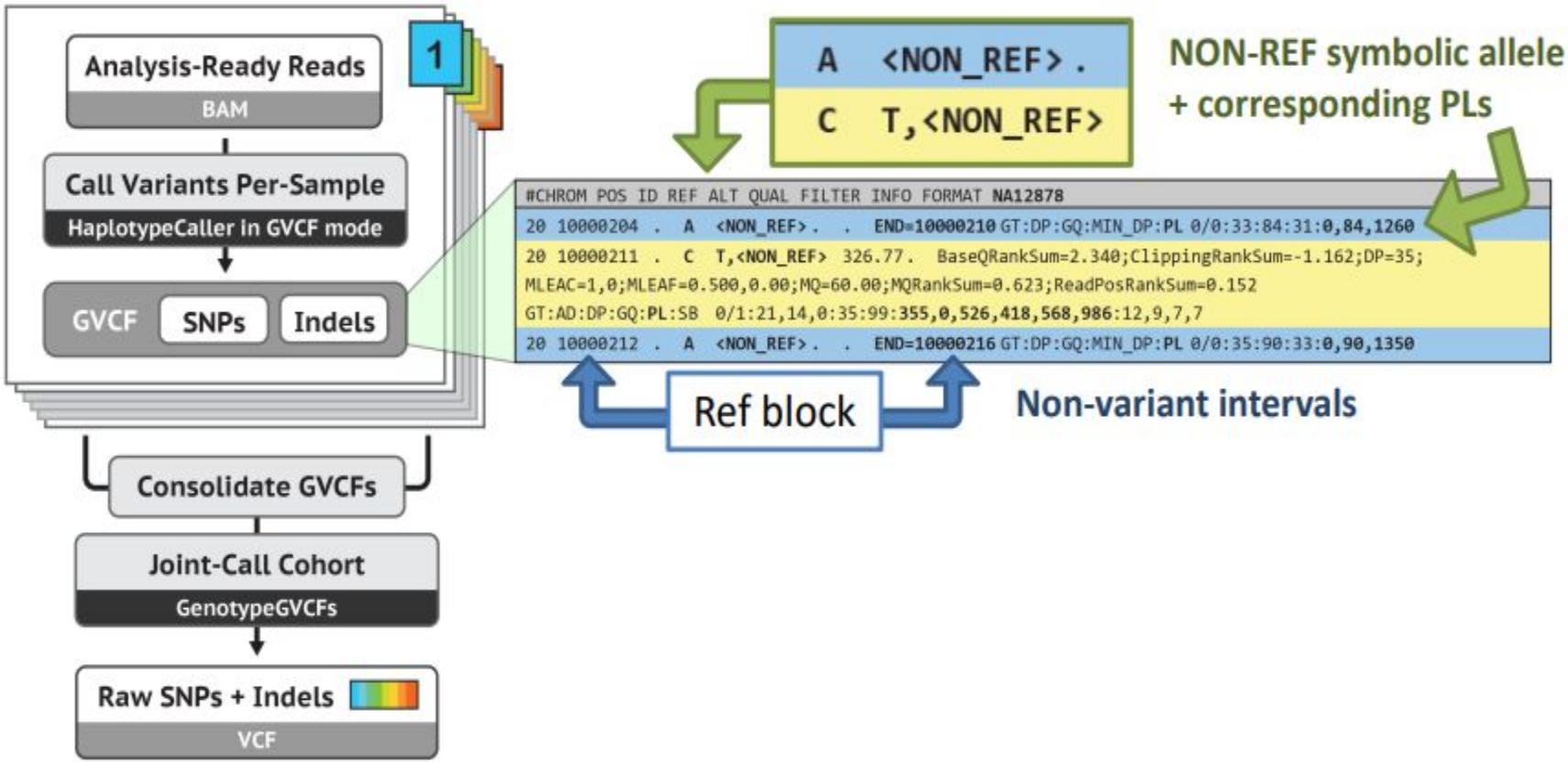
- Run HC in **GVCF mode** to emit GVCF



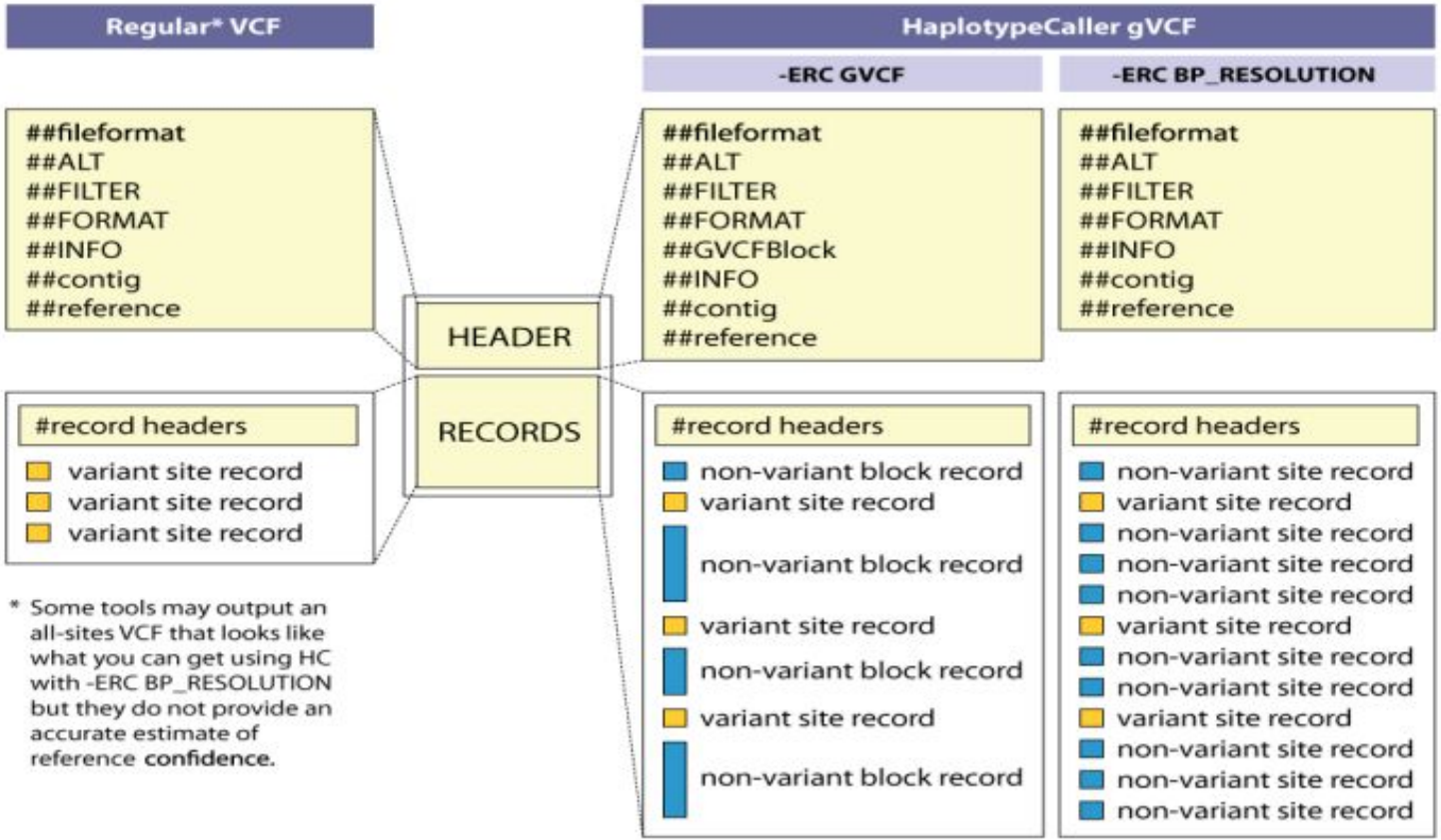
- Run GenotypeGVCFs to re-genotype samples with **multi-sample model**

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

GVCF intermediate contains reference confidence estimate

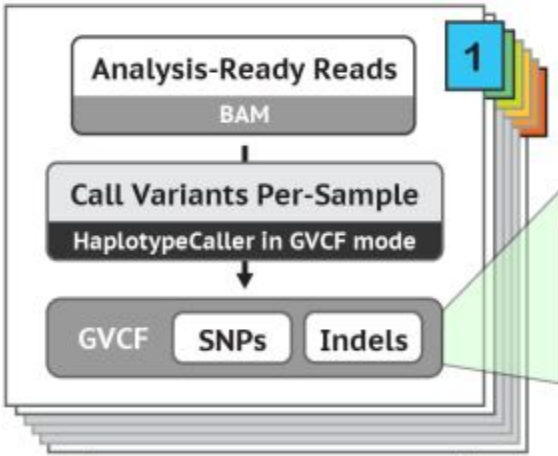


GVCFs are valid VCFs with extra information

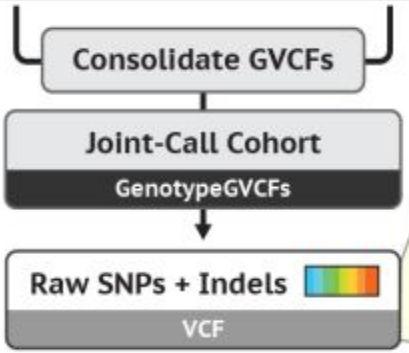


https://gatkforums.broadinstitute.org/gatk/discussion/4017/what-is-a-gvcf-and-how-is-it-different-from-a-regular-vcf

Joint calling produces final multi-sample VCF

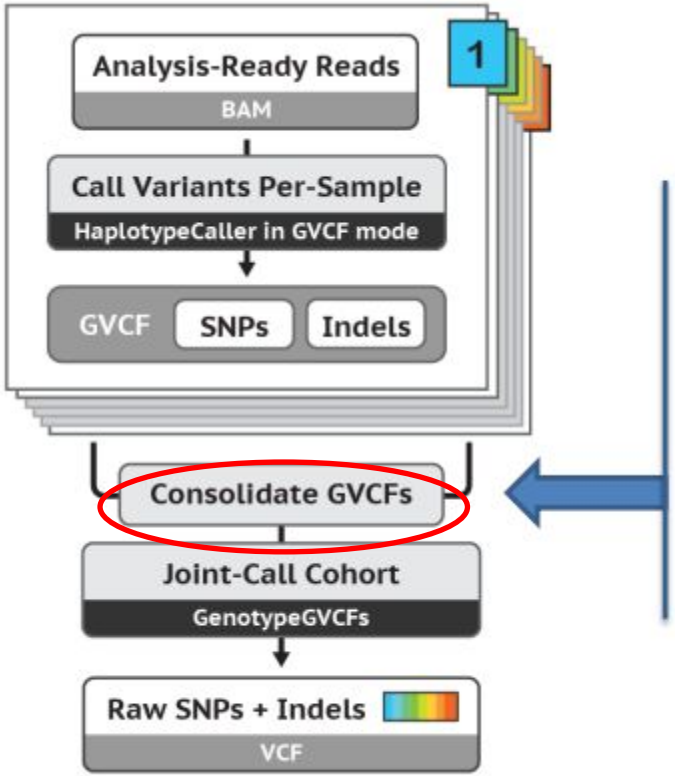



```
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA12878
20 10000204 . A <NON_REF> . . END=10000210 GT:DP:GQ:MIN_DP:PL 0/0:33:84:31:0,84,1260
20 10000211 . C T,<NON_REF> 326.77 . BaseQRankSum=2.340;ClippingRankSum=-1.162;DP=35;
MLEAC=1,0;MLEAF=0.500,0.00;MQ=60.00;MQRankSum=0.623;ReadPosRankSum=0.152
GT:AD:DP:GQ:PL:SB 0/1:21,14,0:35:99:355,0,526,418,568,986:12,9,7,7
20 10000212 . A <NON_REF> . . END=10000216 GT:DP:GQ:MIN_DP:PL 0/0:35:90:33:0,90,1350
```




```
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA12877 NA12878 NA12882
20 10000117 . C T 1606.16 . AC=4;AF=0.667;AN=6;BaseQRankSum=1.66;ClippingRankSum=0.340;DP=85;
FS=5.718;MLEAC=4;MLEAF=0.667;MQ=60.36;MQRankSum=1.45;QD=18.90;ReadPosRankSum=1.62;SOR=1.503
GT:AD:DP:GQ:PL 0/1:17,15:32:99:399,0,439 0/1:11,12:23:99:291,0,292 1/1:0,30:30:90:948,90,0
20 10000211 . C T 1765.16 . AC=4;AF=0.667;AN=6;BaseQRankSum=2.34;ClippingRankSum=-1.147e+00;
DP=97;FS=0.809;MLEAC=4;MLEAF=0.667;MQ=60.00;MQRankSum=1.21;QD=18.58;ReadPosRankSum=0.152;SOR=0.831
GT:AD:DP:GQ:PL 0/1:13,10:23:99:243,0,341 0/1:21,14:35:99:355,0,526 1/1:0,37:37:99:1199,111,0
20 10000439 . T G 1982.13 . AC=5;AF=0.833;AN=6;BaseQRankSum=1.31;ClippingRankSum=0.549;DP=103;
FS=0.000;MLEAC=5;MLEAF=0.833;MQ=60.00;MQRankSum=0.972;QD=19.82;ReadPosRankSum=1.56;SOR=0.839
GT:AD:DP:GQ:PL 0/1:18,12:30:99:208,0,455 1/1:0,29:29:86:795,86,0 1/1:1,40:41:99:1010,110,0
```

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:

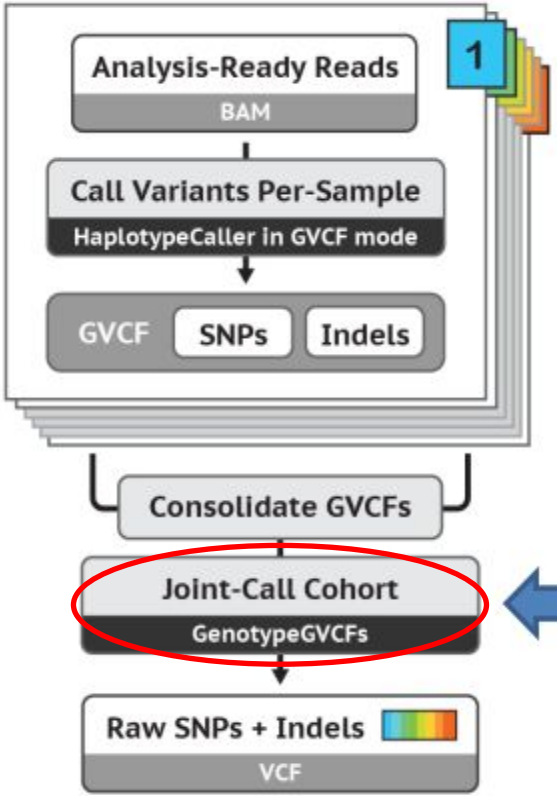
```
gatk CombineGVCFs \
   -R reference.fasta \
   -V sample1.g.vcf \
   -V sample2.g.vcf \
   -O combined.g.vcf
```

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

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

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

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

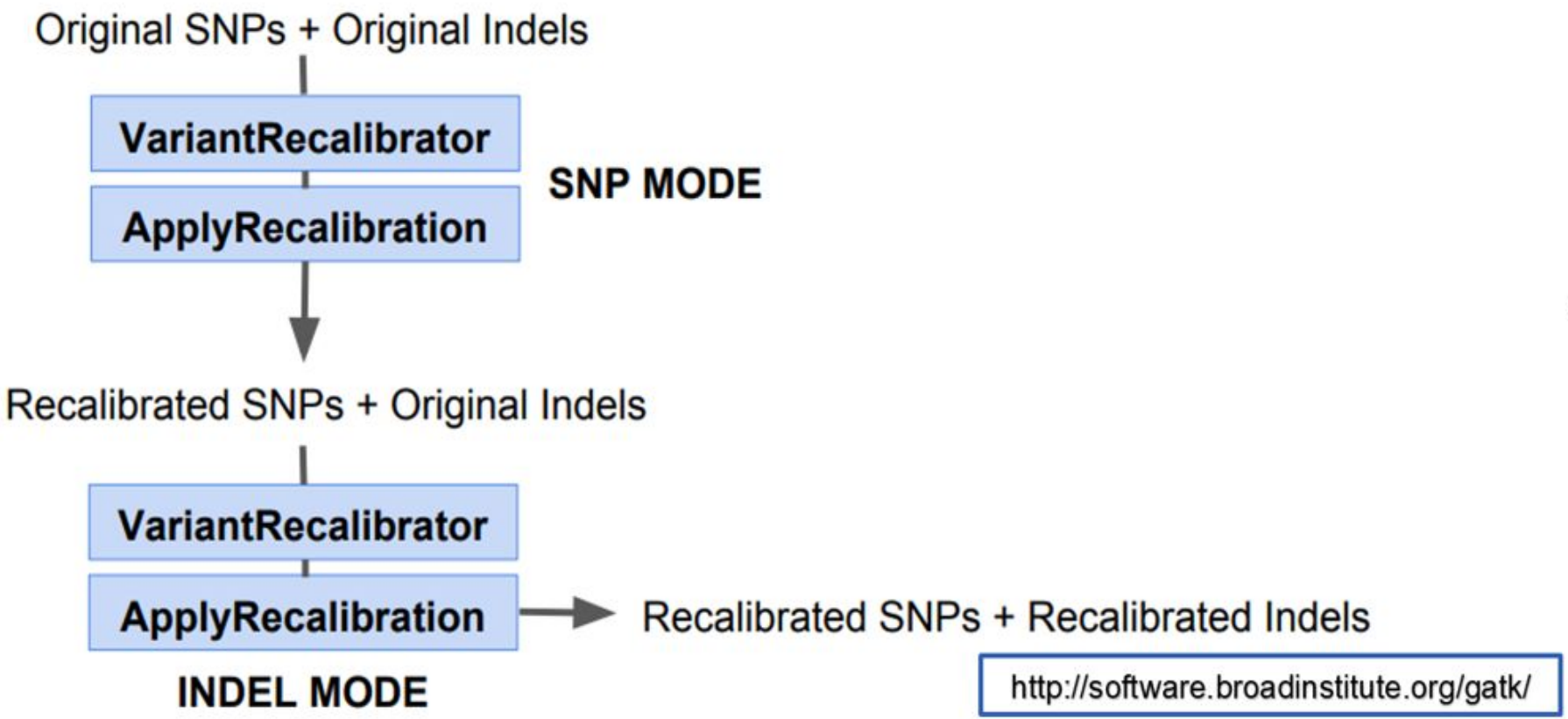
[1] When sample size > 10

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, DP[1], InbreedingCoeff

GATK (broadinstitute.org)

[1] Not to be used for WES

GATK : VQSR Indel human resources

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

Annotations: QD, MD, MQRankSum, ReadPosRankSum, FS, SOR, DP[1], InbreedingCoeff

GATK (broadinstitute.org)

[1] Not to be used for WES

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
```

https://gatk.broadinstitute.org/hc/en-us/articles/13832712299547-CNNScoreVariants

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
```

https://gatk.broadinstitute.org/hc/en-us/articles/13832777777179-FilterVariantTranches

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-


THANK YOU !