Variant Calling Applications in Department of medcal genetics, OUS

Diagnostic on the inherited diseases: nuclear DNA, mitochondrial DNA:

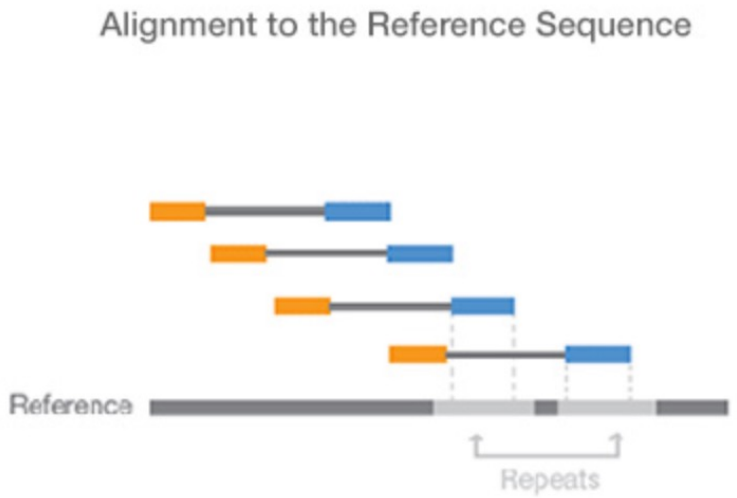
- Inherited general disease
- Inherited cancer disease
- Inherited heart disease

Now process 5,000 whole genome sequencing samples and 5,000 targeted sequencing samples each year!

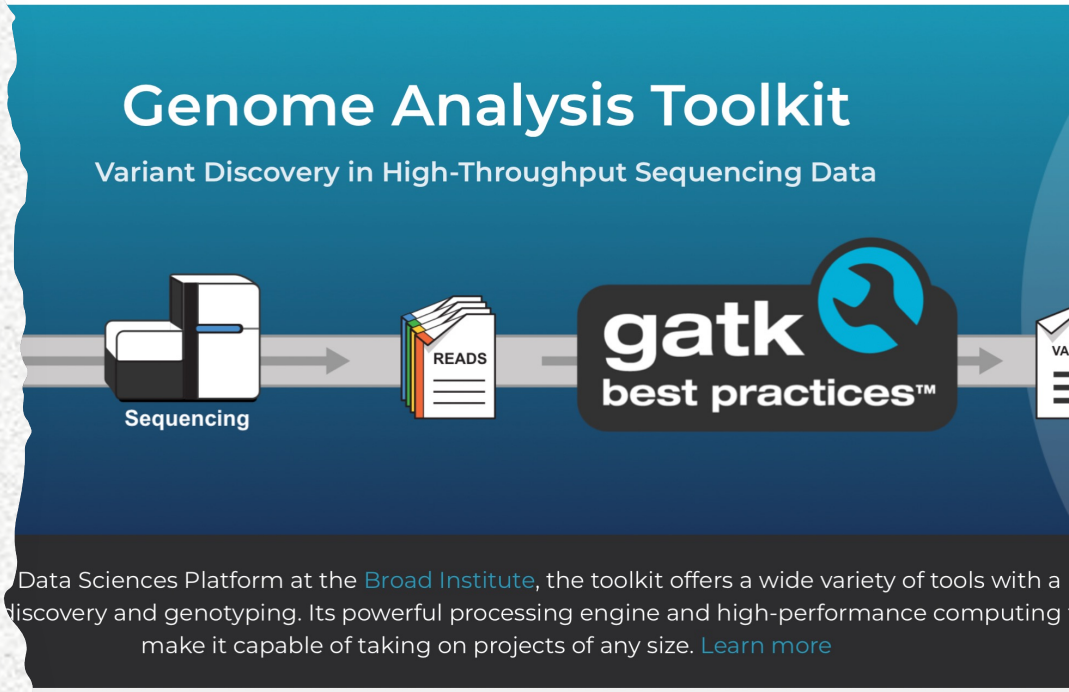
Join at menti.com | use code 2830 9543

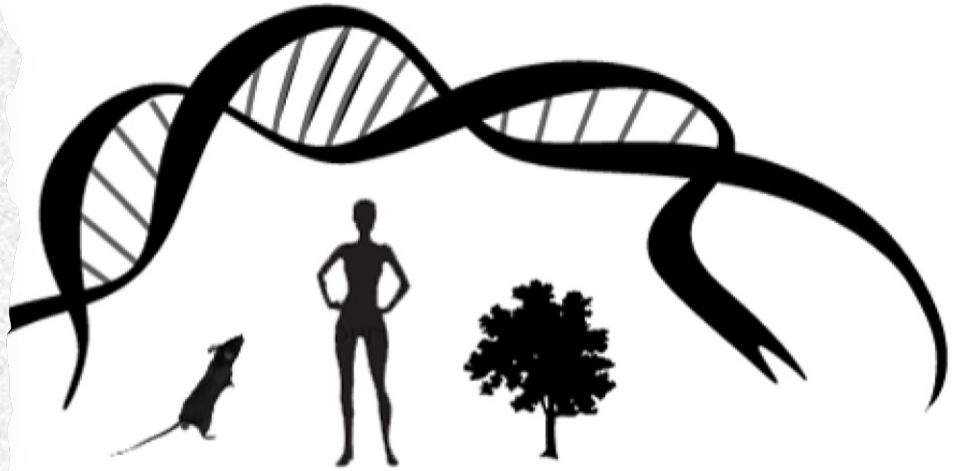
Concepts

- **Short** (higher yield, lower error rate, lower cost) or long read sequencing
- Single end or **paired-end** sequencing
- **Germline** or somatic variant calling
- Target (some regions), exome (all coding regions) or genome sequencing



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

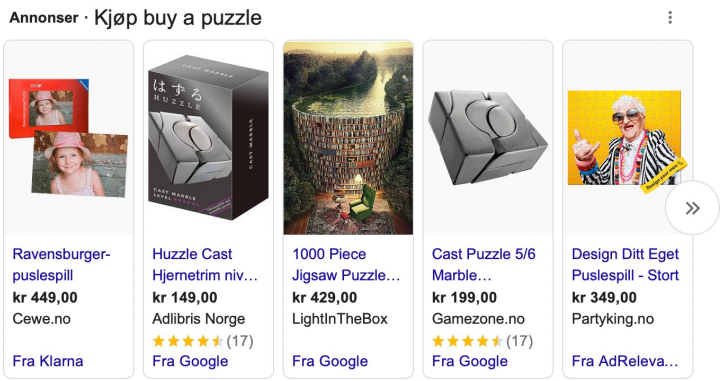




The GATK, designed for human genome and exome analysis and extended to handle other organisms.

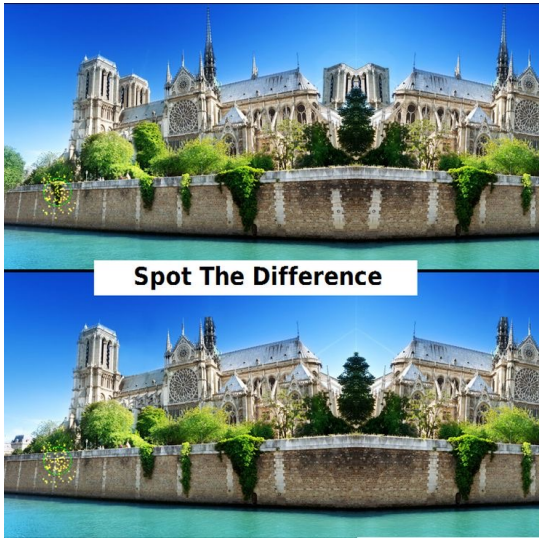
Variant Calling - steps

Carry out sequencing to create FASTQ files

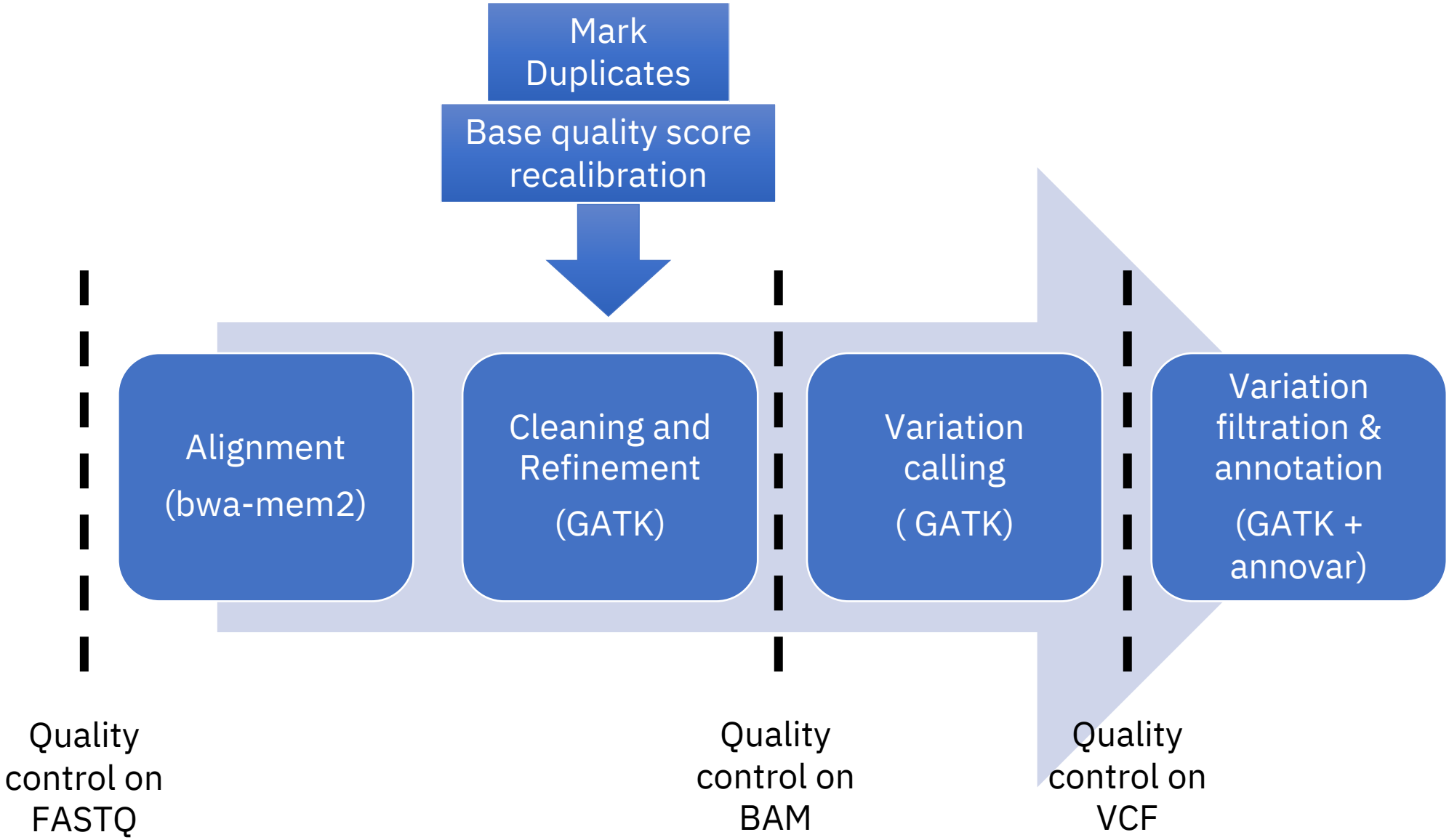


Align the sequences to a reference genome creating BAM or CRAM files

Identify where the aligned reads differ from the reference genome and write to a VCF file

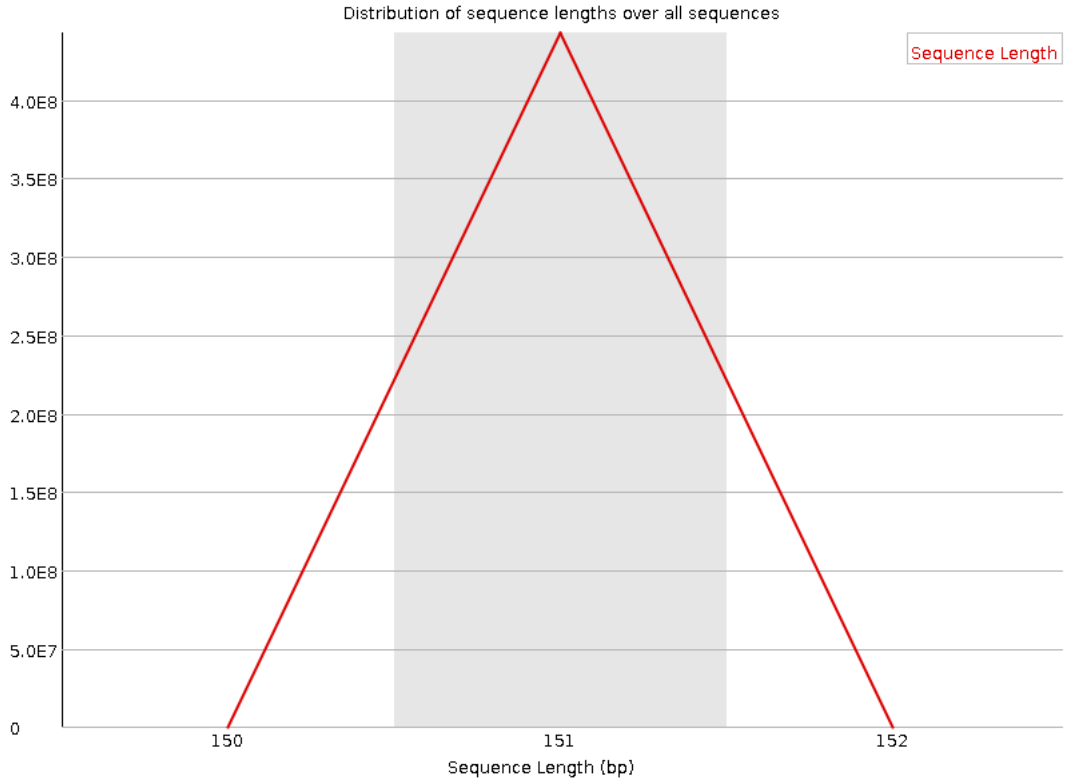


Overview of variant calling

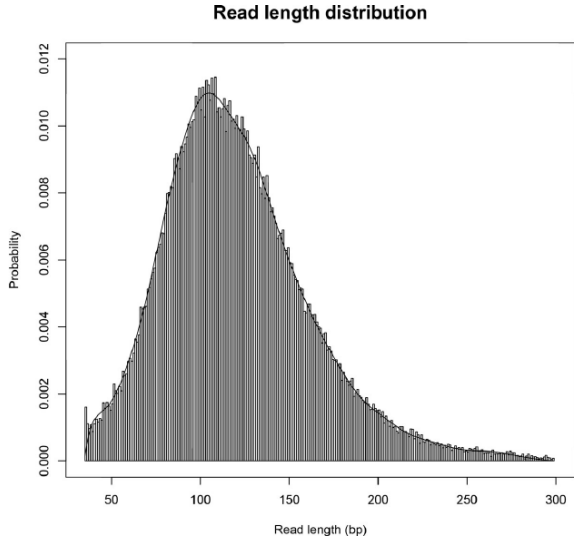


Quality control on fastq files (fastqc)

Sequence length distribution



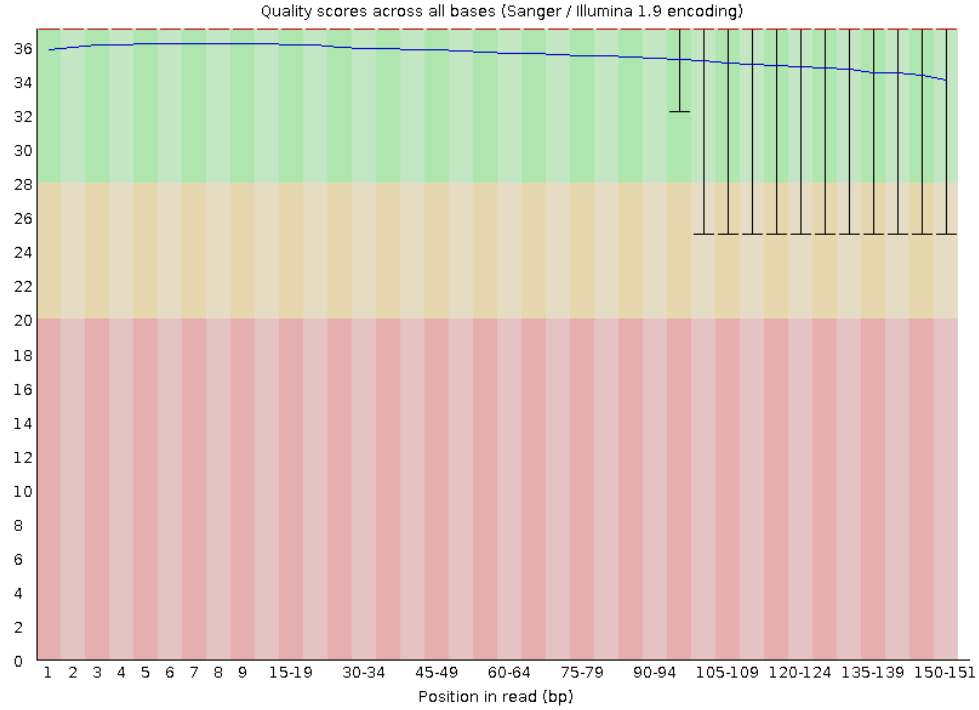
Illumina

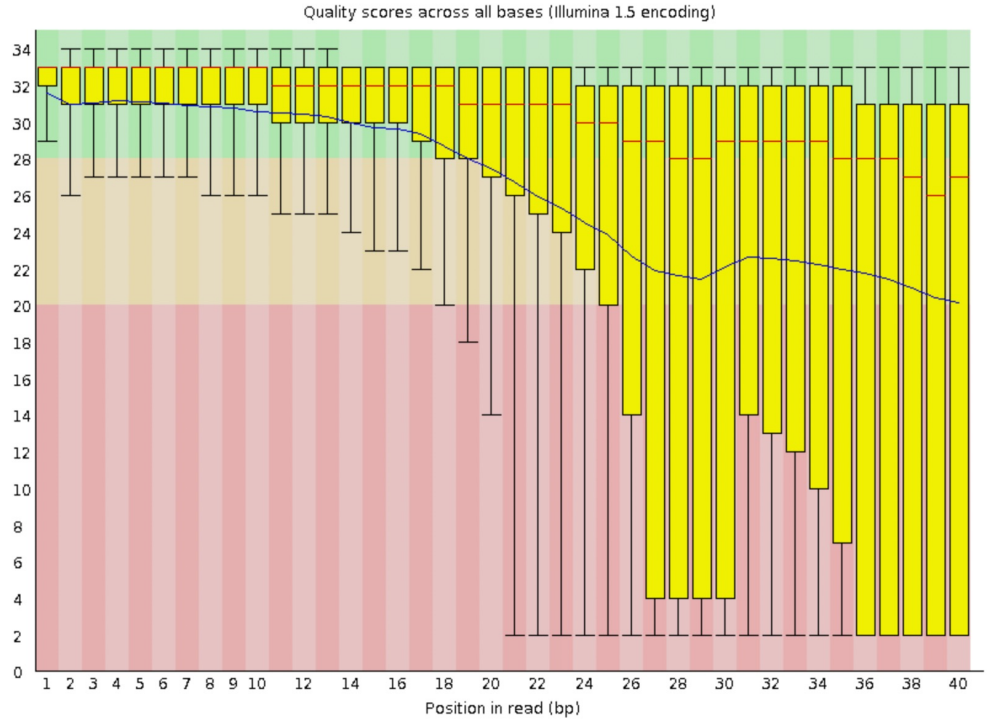


Ion Torrent

Quality control on fastq files (fastqc)

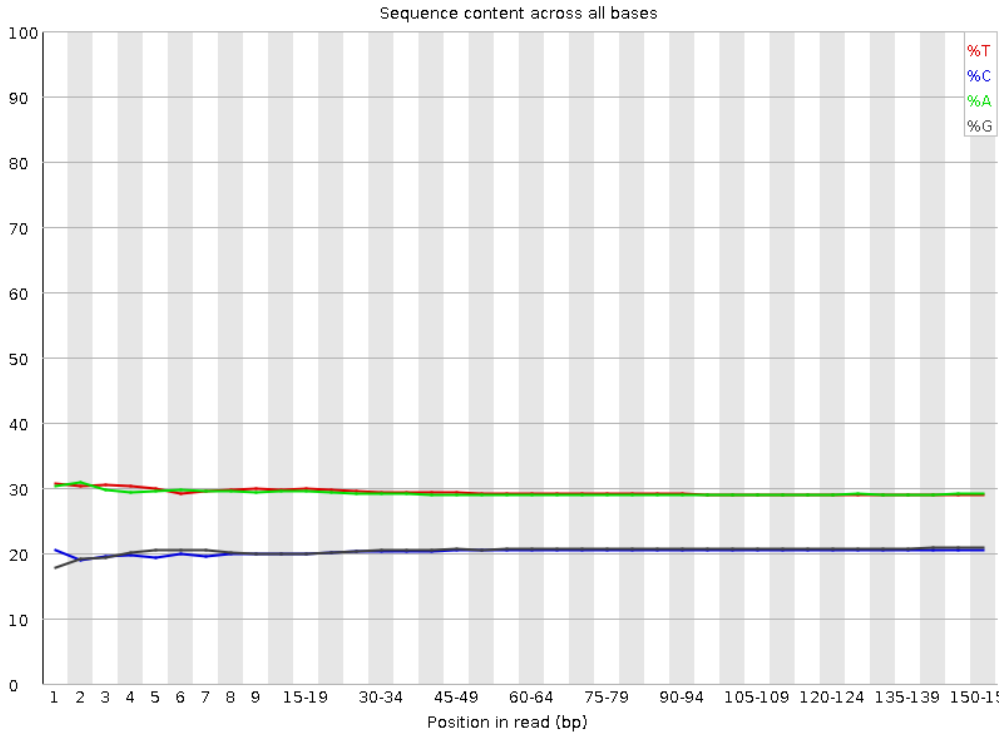
Per Base Sequence Quality

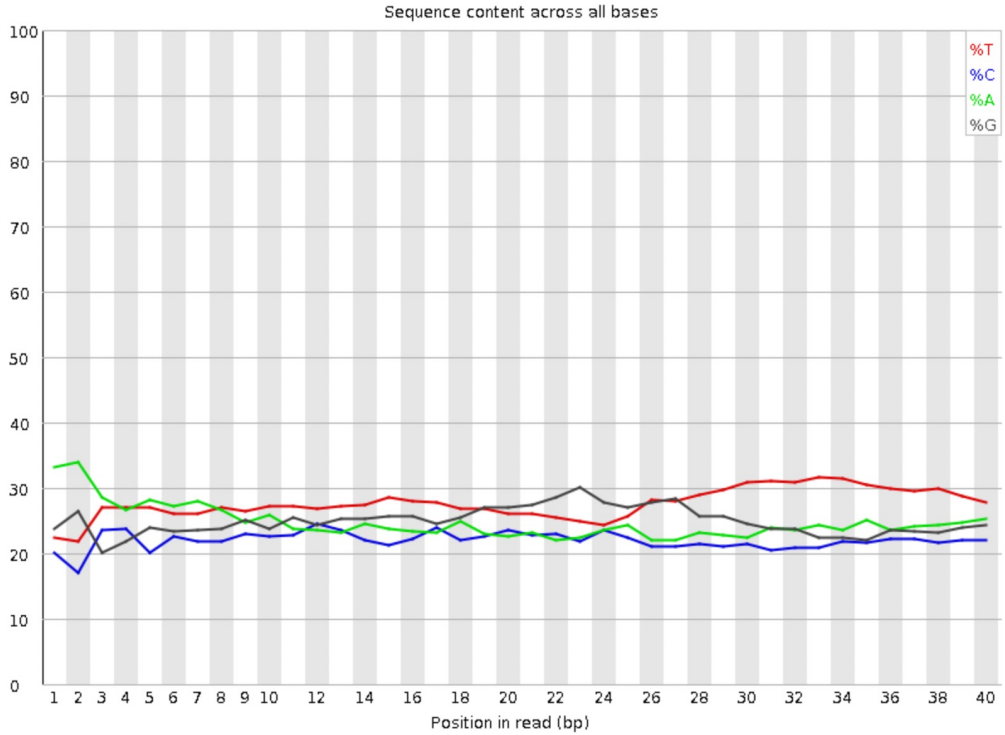




Quality control on fastq files (fastqc) – Cont.

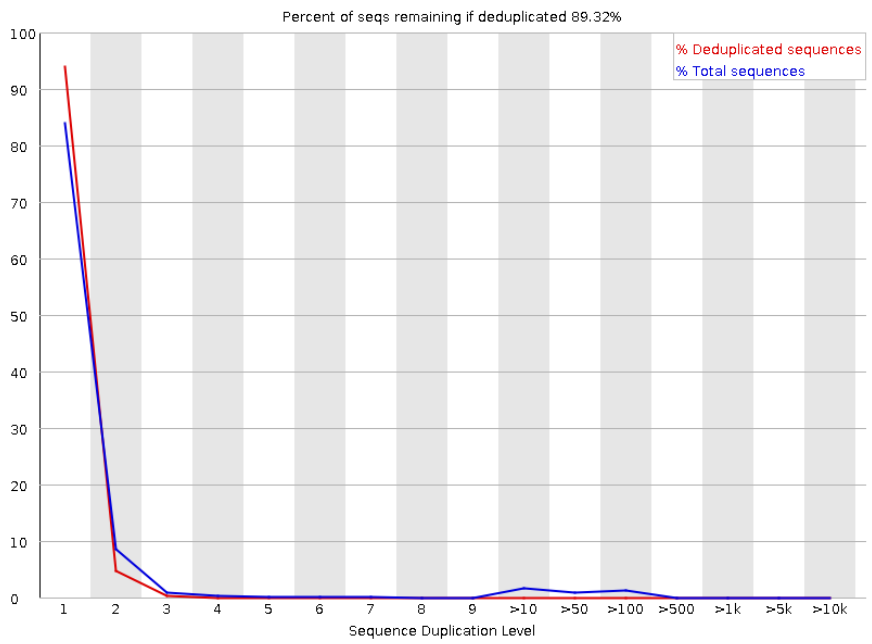
Per Base Sequence Content

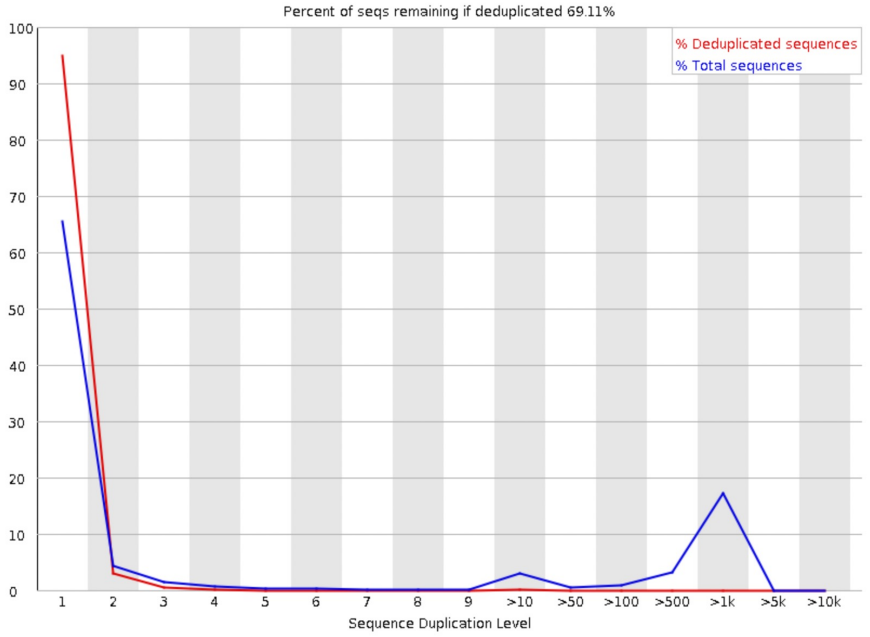




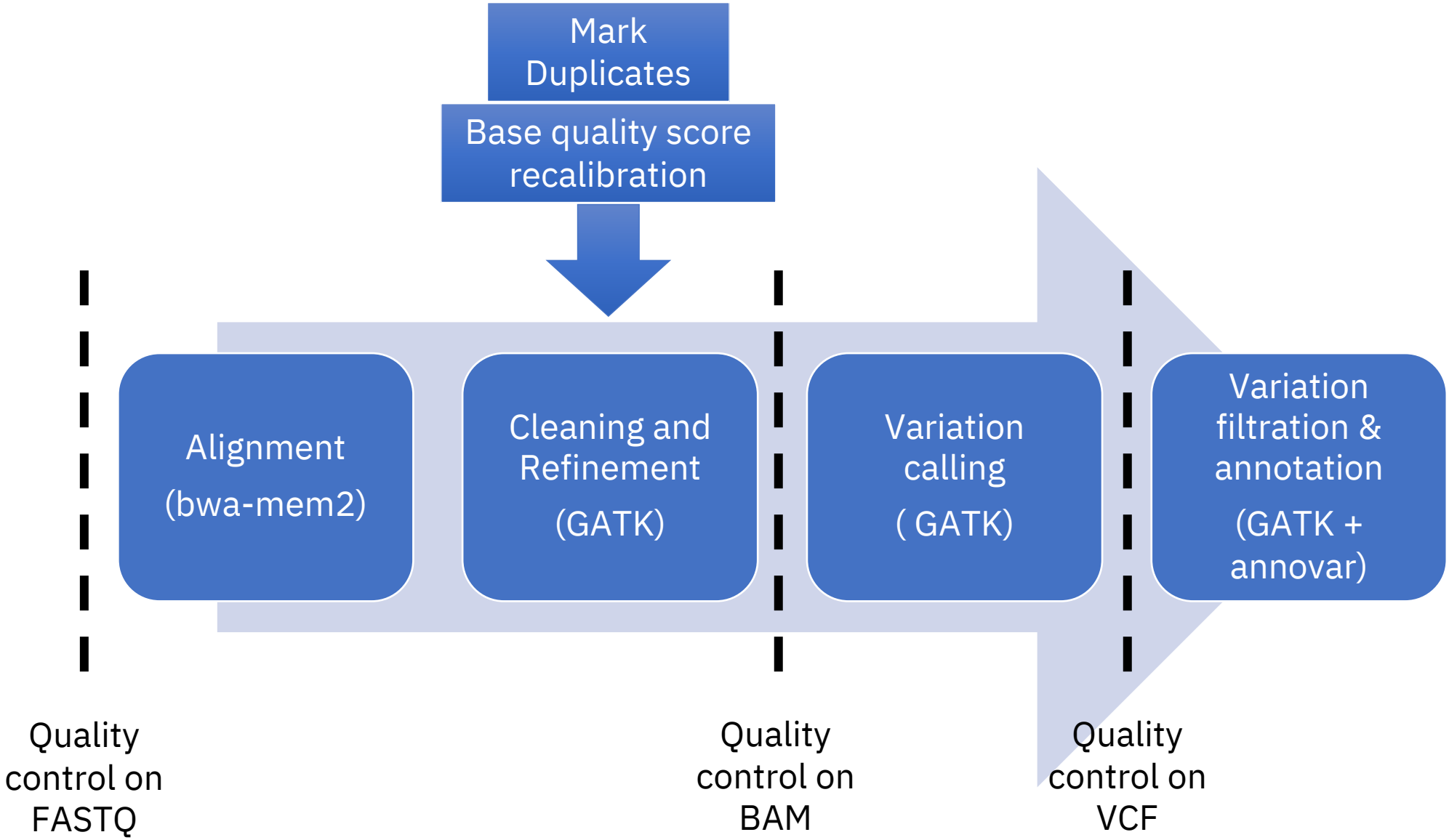
Quality control on fastq files (fastqc) – Cont.

Sequence duplication level





Overview of variant calling



Alignment

Find **genomic locations** (genomic coordinates) of the sequences in a **reference genome** and write into a **SAM/BAM/CRAM file**.

Requirements:
- Short read aligner: bwa-mem2 (mapping DNA sequences against a large reference genome)
- Reference genome (fasta and its index file)
- Fastq file(s)

Preparation before alignment: index the reference genome by the aligner
bwa-mem2 index human.fasta

Alignment – reference genome

What is a good reference genome: No gap, no error

e.g. Human genome
- **It contains all chromosomes**: chromosomes 1–22 (chr1–chr22, **1-22**), X (chrX, **X**), Y (chrY, **Y**) and Mitochondrial (chrM, **MT**).

- **And *unlocalized sequence:*** on a specific chromosome but with unknown order or orientation. Identify by _random suffix.

- **And *unplaced* sequence:** on an unknown chromosome. Identify by chrU_ prefix.

Because the aligner tries to place the read to a location in the reference genome:
No matter of sizes of the capture, always use the whole genome as the reference genome!

https://gatk.broadinstitute.org/hc/en-us/articles/360035890711-GRCh37-hg19-b37-humanG1Kv37-Human-Reference-Discrepancies#humanG1Kv37

Alignment – read group – marker for the reads

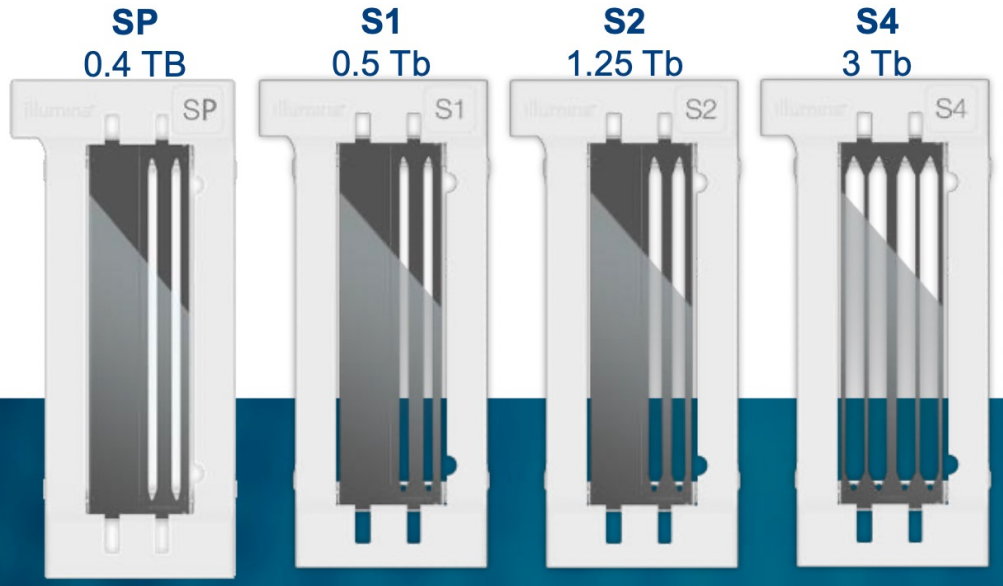
A read group: all the reads derived from the same:
library preparation && biological sample && lane && flow cell

Used to not only **differentiate samples**, but also **various technical features associated with artefacts**. Can be used to correct errors in downstream analysis, e.g. duplicate marking and base quality score recalibration.

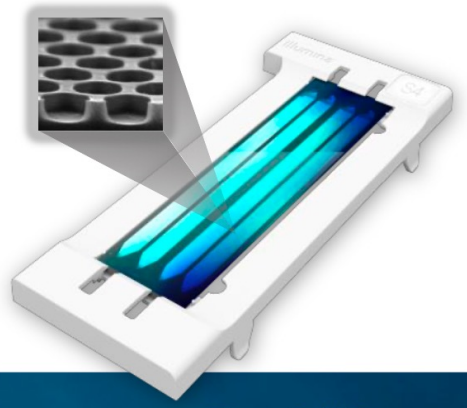
Tags in SAM/BAM/CRAM (connected by tabs, SAM format specification):
@RG
ID : Read group identifier
LB : Library
PL : Platform/technology used to produce the reads (Valid values: ILLUMINA, SOLID, LS454, HELICOS and PACBIO)
CN : Name of sequencing center producing the read
PU : Platform unit (e.g. flowcell-barcode.lane.sample_barcode)
SM : Sample
DS : Description
DT : Date the run was produced, ISO8601 date or date/time, e.g. 2022-10-24
@RG\tID:NA12878\tLB:NA12878\tPL:ILLUMINA\tPU:BH35C5DSX5.1\tSM:NA12878

NovaSeq 6000 Flow Cell Configurations

Max Output / Flow Cell:



Patterned Flow Cell



	Flow Cell Type			
	SP	**S1**	**S2**	**S4**
Lane Width	3.8mm	3.8mm	5mm	7mm
Lane Pairs	Single			Dual
Surfaces	Single	Dual		

Alignment – commands

```
bwa-mem2 mem \ # mem is one of the functions in bwa-mem2
-R ‘Read group string’ \ # read group information
-v 2 \ # how much log informaiton print out in the screen, 2 means only show warnings/error
-M \ # Mark shorter split hits as secondary (for Picard compatibility).
-t CPUs \ # number of CPUs want to used in the calculation
-Y indexed reference genome \ # e.g. human_g1k_v37_decoy.fasta
R1_FASTQ R2_FASTQ | \
samtools view –@ CPUs –Sb - | \ # convert SAM to BAM file
samtools sort –@ CPUs –o sorted.bam –T sorted – # sort the BAM file by genomic coordinates

samtools index -@ CPUs sorted.bam # index the BAM file
```

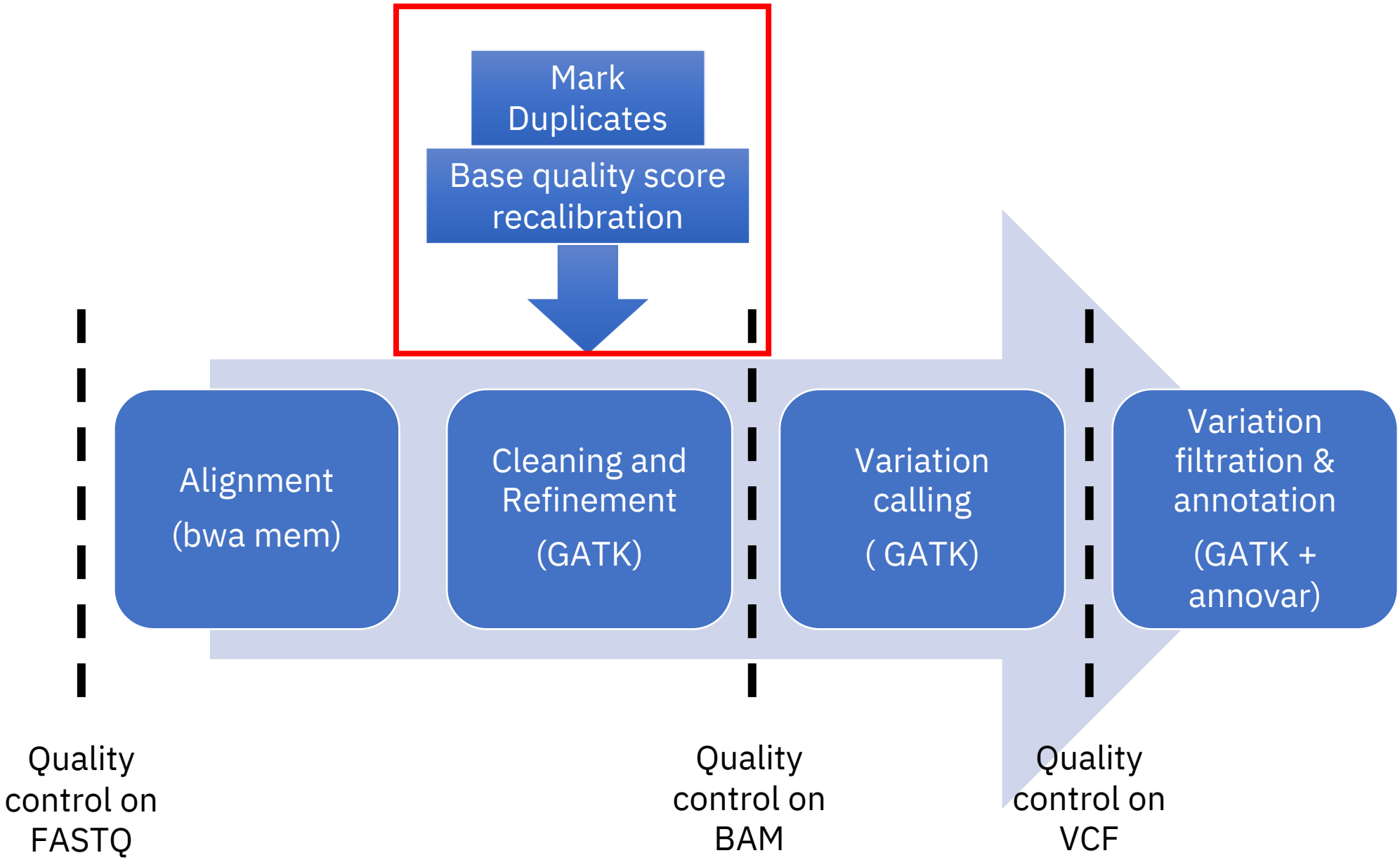
Results will be: sorted.bam and sorted.bam.bai

Refinement of alignment – data clean up

NO PERFECT WORLD!

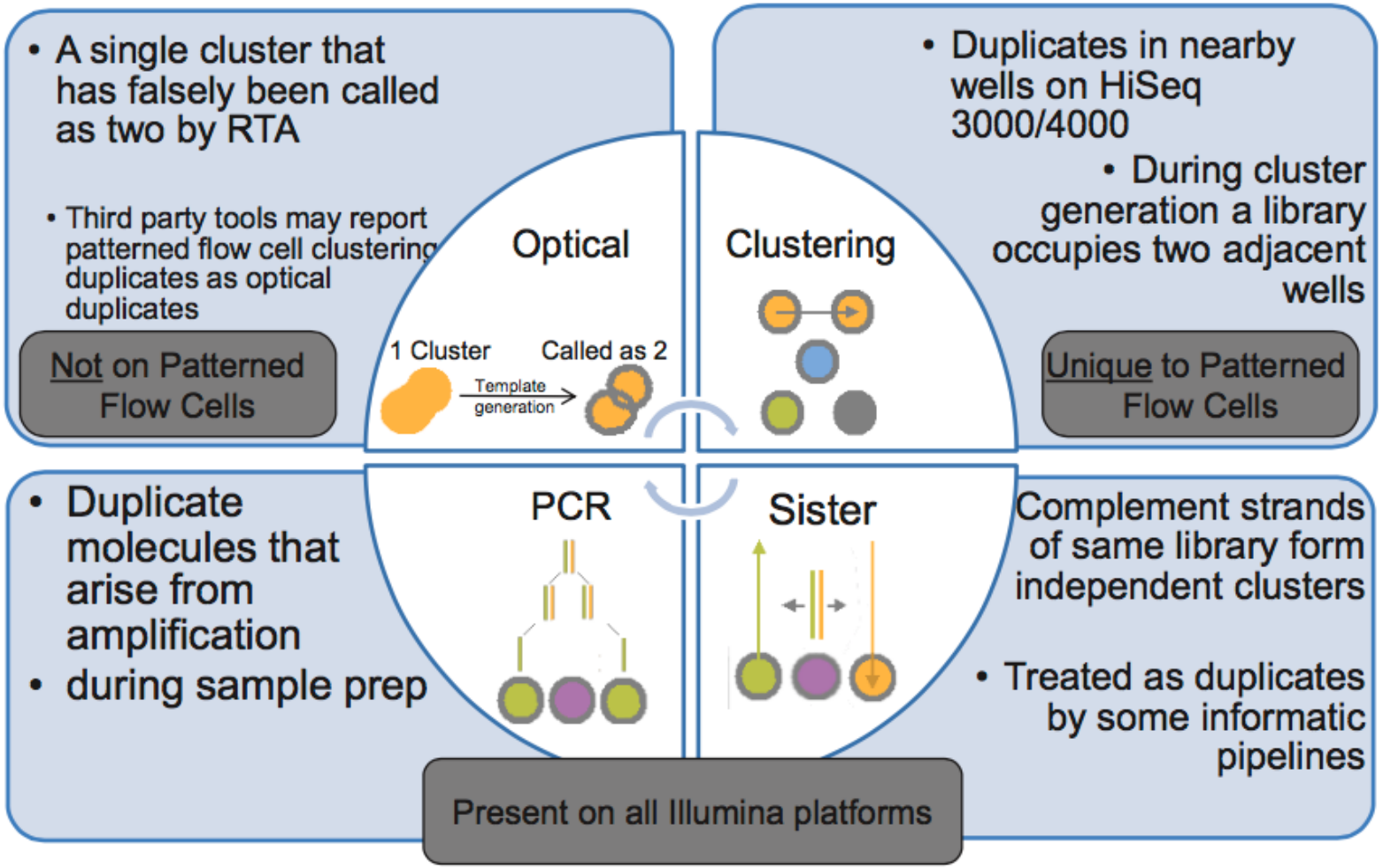
- The sample preparation is not perfect – Mark duplicates
- The sequencer is not perfect - Mark duplicates, BQSR
- The aligner is not perfect – re-assembly in variant caller

Overview of variant calling



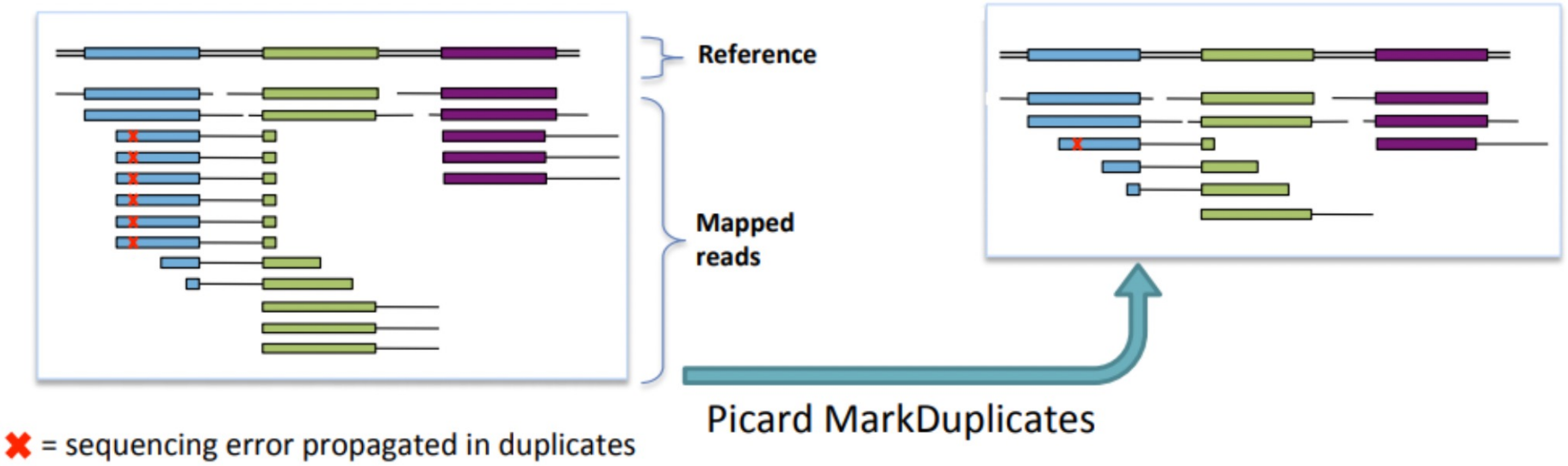
Refinement of alignment – Duplicates

Originating from a single fragment of DNA, but recognized as two reads in the sequencing - PCR duplicates, Optical duplicates



Refinement of alignment – Duplicates - algorithm

Duplicated reads could create fake coverage, which create false variants



By comparing sequences in the 5 prime positions of both reads and read-pairs in a SAM/BAM file.

The tool differentiates the primary and duplicate reads using an algorithm that ranks reads by the sums of their base-quality scores.

main output is a new SAM or BAM file, Duplicates are marked with the hexadecimal value of 0x0400, which corresponds to a decimal value of 1024.

Refinement of alignment – Duplicates - command

```
gatk MarkDuplicates \
-INPUT ${input_file}.bam \
-OUTPUT "${output_file}.bam" \
-METRICS_FILE "${output_file}.markduplicates.metrics" \
-CREATE_INDEX true \
-CREATE_MD5_FILE true \
-VALIDATION_STRINGENCY STRICT
```

Refinement of alignment – Base Quality Score Recalibration

Base quality score (in fastq file): tell how much we can trust the base said by the sequencer. It is important for variant calling.

However, there is systematic (non-random) technical error from sequencer, leading to over- or under-estimated base quality scores in the data.

Steps:

BaseRecalibrator divided bases into bins based on the following features:

- read group
- quality score from the sequencer
- machine cycle
- current base + previous base (dinucleotide)

and calculate error rate by (# mismatches + 1) / (# bases + 2) per bin. The known variants will not be counted as mismatches.

ApplyBQSR adjust each base's score based on which bins it falls in.

Refinement of alignment – BQSR – command 1

```
gatk BaseRecalibrator \
-I ${input_file}.bam \
-R ${reference} \
-L calling_region.interval_list \
--known-sites ${dbsnp_data} \
--known-sites 1000G_phase1.indels.b37.vcf \
--known-sites Mills_and_1000G_gold_standard.indels.b37.vcf \
-O ${output_file}.recal_data.table
```

Refinement of alignment – BQSR – command 2

```
gatk ApplyBQSR \
-I ${input_file}.bam \
-R ${reference} \
--bqsr-recal-file ${output_file}.recal_data.table \
--create-output-bam-md5 true \
-O ${output_file}.bam
```

With --emit-original-quals, the original base quality score is stored under OQ tag in the bam file.

What steps needed for which sequencing

	Target sequencing	Exome sequencing	Genome sequencing
Mark duplicates	✓	✓	✓
BQSR	✗	✓	(✓)

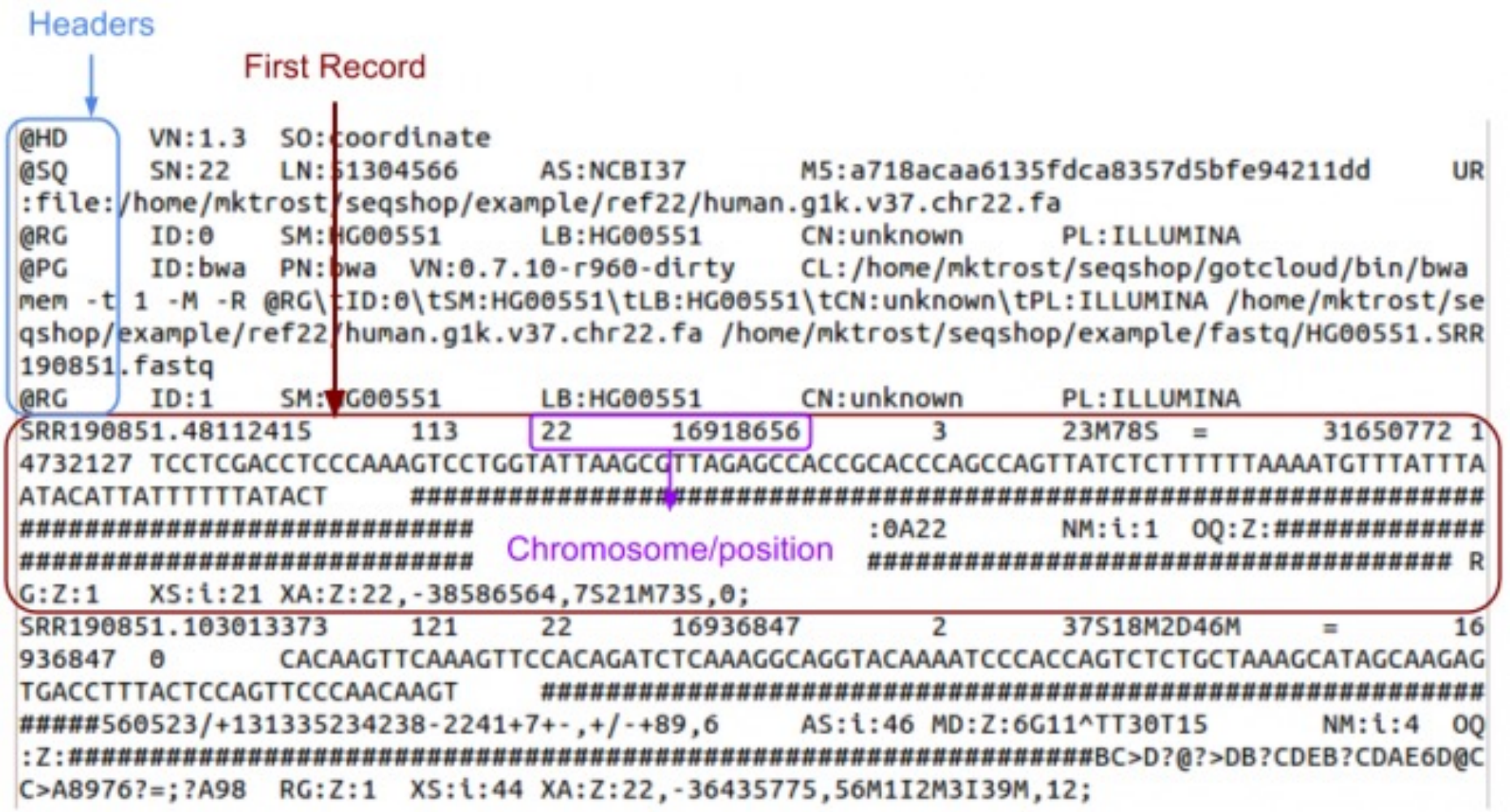
Alignment result

Reads not mapped: unmapped reads, contaminations, bad quality bases, structural variations in the sample

Reads mapped to the unique location: mapping quality > 0

Reads mapped to the multiple locations equally: mapping quality = 0, repeats, pseudogenes

Alignment result: SAM/BAM file format




```
@HD     VN:1.3  SO:coordinate
@SQ     SN:22   LN:51304566     AS:NCBI37       M5:a718acaa6135fdca8357d5bfe94211dd     UR
:file:/home/mktrost/seqshop/example/ref22/human.g1k.v37.chr22.fa
@RG     ID:0    SM:HG00551      LB:HG00551      CN:unknown      PL:ILLUMINA
@PG     ID:bwa  PN:bwa  VN:0.7.10-r960-dirty    CL:/home/mktrost/seqshop/gotcloud/bin/bwa
mem -t 1 -M -R @RG\tID:0\tSM:HG00551\tLB:HG00551\tCN:unknown\tPL:ILLUMINA /home/mktrost/se
qshop/example/ref22/human.g1k.v37.chr22.fa /home/mktrost/seqshop/example/fastq/HG00551.SRR
190851.fastq
@RG     ID:1    SM:HG00551      LB:HG00551      CN:unknown      PL:ILLUMINA
SRR190851.48112415      113     22      16918656        3       23M78S  =       31650772 1
4732127 TCCTCGACCTCCCAAAGTCCTGGTATTAAGCGTTAGAGCCACCGCACCCAGCCAGTTATCTCTTTTTTAAAATGTTTATTTA
ATACATTATTTTTTATACT     ##################################################################
##############################                    :0A22       NM:i:1  OQ:Z:##############
##############################                    ################################### R
G:Z:1   XS:i:21 XA:Z:22,-38586564,7S21M73S,0;
SRR190851.103013373     121     22      16936847        2       37S18M2D46M     =       16
936847  0       CACAAGTTCAAAGTTCCACAGATCTCAAAGGCAGGTACAAAATCCCACCAGTCTCTGCTAAAGCATAGCAAGAG
TGACCTTTACTCCAGTTCCCAACAAGT      ################################################################
#####560523/+131335234238-2241+7+-,+/-+89,6     AS:i:46 MD:Z:6G11^TT30T15       NM:i:4  OQ
:Z:##################################################################BC>D?@?>DB?CDEB?CDAE6D@C
C>A8976?=;?A98  RG:Z:1  XS:i:44 XA:Z:22,-36435775,56M1I2M3I39M,12;
```

FLAG (nr. 2): 4 (unmapped reads), 1024 (duplicated reads)
https://broadinstitute.github.io/picard/explain-flags.html

Exercise I

- Copy the scripts to your local folder

```
mkdir DIR_SHOULD_BE
cd DIR_SHOULD_BE
cp -r /projects/ec34/in-biosx000/human_vc/scripts .
```

- Create folder structure of the exercise

```
bash scripts/00_prepare.bash
```

- Log in to int-{1,2,3,4}, check the resource available by:

```
cat /etc/motd
```

- Running the 01 and 02 scripts (scripts/01_..., scripts/02_...) to do alignment, refinement of alignment (see commands on the terminal)

Exercise I – continue

- View the BAM file in the terminal

```
samtools view sorted.bam
```

```
samtools view -H sorted.bam
```

Exercise I – continue

- Download IGV (Integrative Genomics Viewer):

https://software.broadinstitute.org/software/igv/download (latest version)
https://data.broadinstitute.org/igv/projects/downloads/ (previous version)
https://igv.org/app/ (web version)

- Download the BAM files to view in IGV (File->Load from File, make sure the Genome is 'Human (GRCh37/hg19)'):

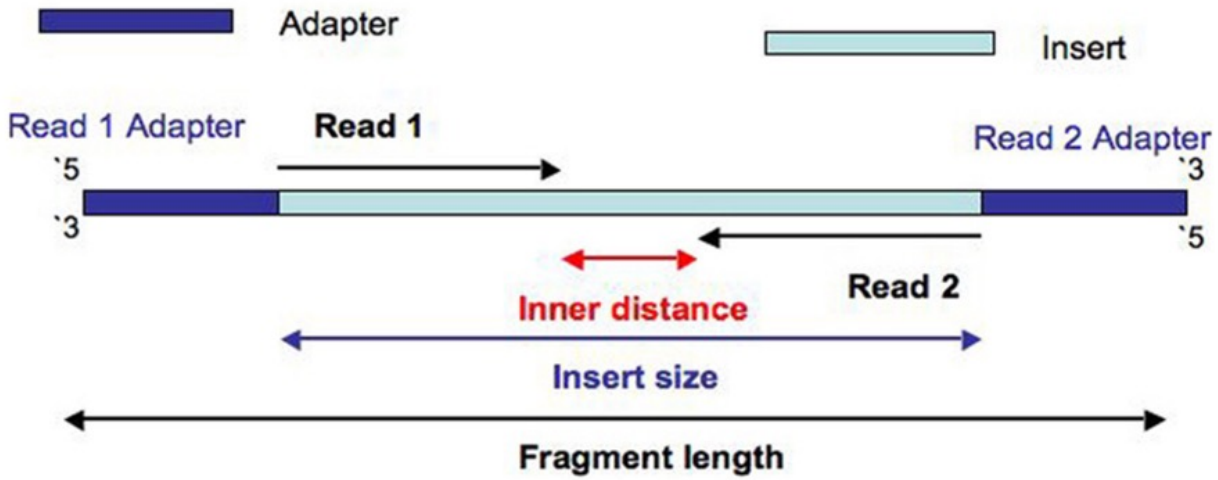
http://bit.ly/3C1FgDH

- View the bam file in IGV on duplicates, read pairs, reads mapped to multiple locations (click the reads, base, 7:6,021,962-6,023,036, view as pairs, variants 1:17,380,476-17,380,516 (SNP), 2:48,032,854-48,032,893(deletion), 11:108,151,687-108,151,726(insertion))

Quality Control on the BAM file

- **How many reads well mapped to the reference genome**
  Adaptors
  Sequences not from the species you sequence

- **The length of the DNA fragment in the sample preparation (insert size)**



- **How many reads mapped on each genomic position (coverage)**
  Median coverage, percentage of 10X, percentage of 20X, coverage uneveness

Quality Control on the BAM – Supported interval list format

.interval_list, 1-based

```
@HD     VN:1.0  SO:coordinate
@SQ     SN:1    LN:249250621    AS:GRCh37       UR:http://www.broadinstitute.org/ftp/pub/seq
@SQ     SN:2    LN:243199373    AS:GRCh37       UR:http://www.broadinstitute.org/ftp/pub/seq
1       30366   30503   +       target_1
1       69089   70010   +       target_2
1       367657  368599  +       target_3
1       621094  622036  +       target_4
1       861320  861395  +       target_5
1       865533  865718  +       target_6
```

.list or .intervals, 1-based (no sequence dict required)

<chr>:<start>-<end>

.bed, 0-based

<chr>\t<start>\t<end>

Quality Control on the BAM – CollectAlignmentSummaryMetrics

```
gatk CollectAlignmentSummaryMetrics \
R=${reference} \
I=${bam_file} \
O=CollectAlignmentSummaryMetrics.txt \
MAX_INSERT_SIZE=600 # used to define chimeras
```

Important results:
- Total number of reads (total, no exclusions)
- High quality aligned PF reads (high quality == mapping quality >= 20)
- High quality aligned PF Q20 bases (subset of above where base quality >= 20)
- Reads aligned in pairs (vs. reads aligned with mate unaligned/not present)

Metrics are written for the first read of a pair, the second read, and combined for the pair.

Quality Control on the BAM file – CollectInsertSizeMetrics

```
gatk CollectInsertSizeMetrics \
R=${reference} \
I=${input_file}.bam \
O=CollectInsertSizeMetrics.txt \
HISTOGRAM_FILE=CollectInsertSizeMetrics-histogram.pdf
```

Important results:
- MEDIAN_INSERT_SIZE
- MEDIAN_ABSOLUTE_DEVIATION

(Require R in the PATH)

Quality Control on the BAM file – HsMetrics

For capture sequencing samples (Find out the efficiency of the capture kit):

```
gatk CollectHsMetrics \
R=${reference} \
I=${input_file}.bam \
O=CollectHsMetrics.txt \
BAIT_INTERVALS=${bait_list} \
TARGET_INTERVALS=${target_list} \
PER_TARGET_COVERAGE=CollectHsMetrics-per-target-coverage.txt
```

For whole genome sequencing samples:

```
gatk CollectWgsMetrics \
R=${reference}
I=${input_file}.bam\
O=collect_wgs_metrics.txt
```

Quality Control on the BAM file – HsMetrics

BAIT_DESIGN_EFFICIENCY: TARGET/BAIT, 1 indicates a perfect design

OFF_BAIT_BASES: the number of PF_BASES_ALIGNED that are mapped away from any baited region.

MEDIAN_TARGET_COVERAGE: The median coverage of a target region.

PCT_USABLE_BASES_ON_BAIT/TARGET: The number of aligned, de-duped, on-bait/target bases out of the PF bases available.

PCT_EXC_DUPE/MAPQ/BASEQ/OVERLAP/OFF_TARGET: percentage of reads excluded from the coverage calculation because of different reasons

PCT_TARGET_BASES_10X/20X: The fraction of all target bases achieving 10X/20x or greater coverage.

Quality Control on the BAM file – Coverage problems

- Median coverage is low, reason:
  The number of reads from sequencing is low,
  High duplication rate,
  High level of unmapped reads
  High off target reads

- The coverage is not even, you will see:
  The number of reads from sequencing is probably OK,
  the median coverage is also OK,
  Percentage of positions covered by more than 10 reads: bad
  Percentage of positions covered by more than 20 reads: bad

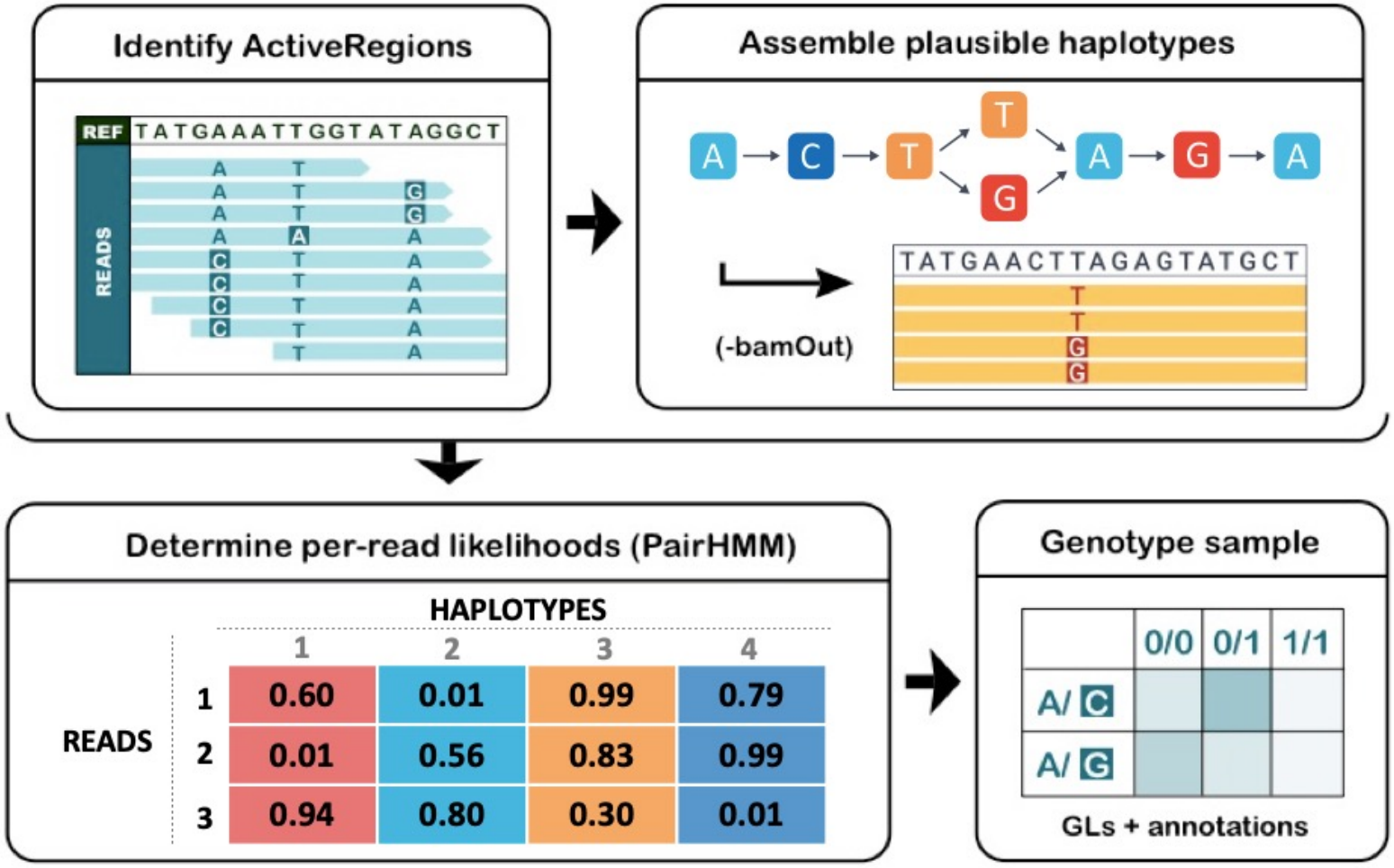
Variant calling – germline variants VS somatic variants

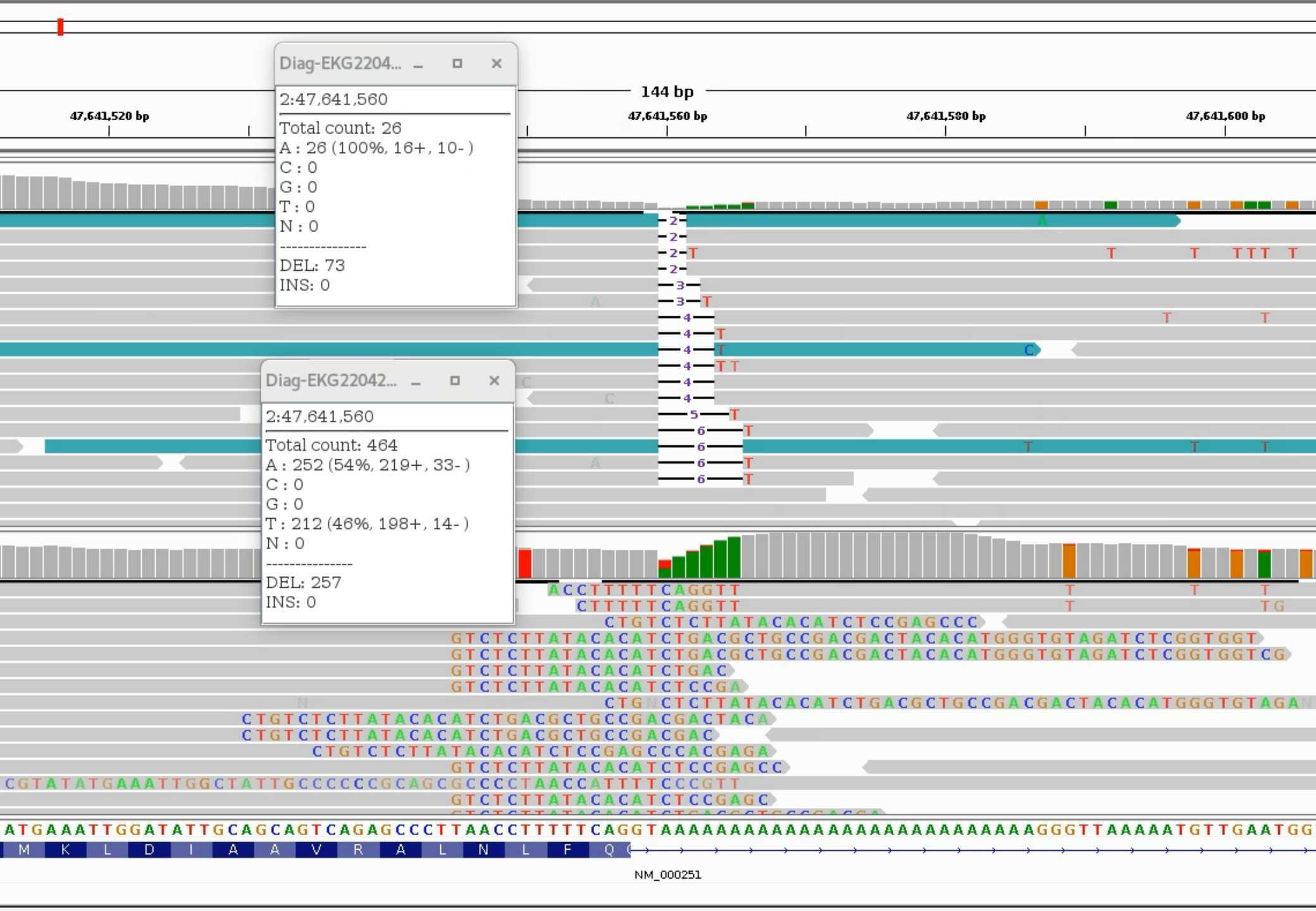
Variant calling is the process by which we identify variants from sequence data.

Germline variants: in all cells, for diploid, heterozygosity (Aa) allele ratio should be 50%.

Somatic variants: only in certain cells, heterozygosity (Aa) allele ratio could be very low depends on whether you pick up the right tissue.

Germline Variant Calling




Diag-EKG2204...
2:47,641,560
Total count: 26
A : 26 (100%, 16+, 10-)
C : 0
G : 0
T : 0
N : 0
DEL: 73
INS: 0
Diag-EKG22042...
2:47,641,560
Total count: 464
A : 252 (54%, 219+, 33-)
C : 0
G : 0
T : 212 (46%, 198+, 14-)
N : 0
DEL: 257
INS: 0
144 bp
47,641,520 bp
47,641,560 bp
47,641,580 bp
47,641,600 bp
NM_000251

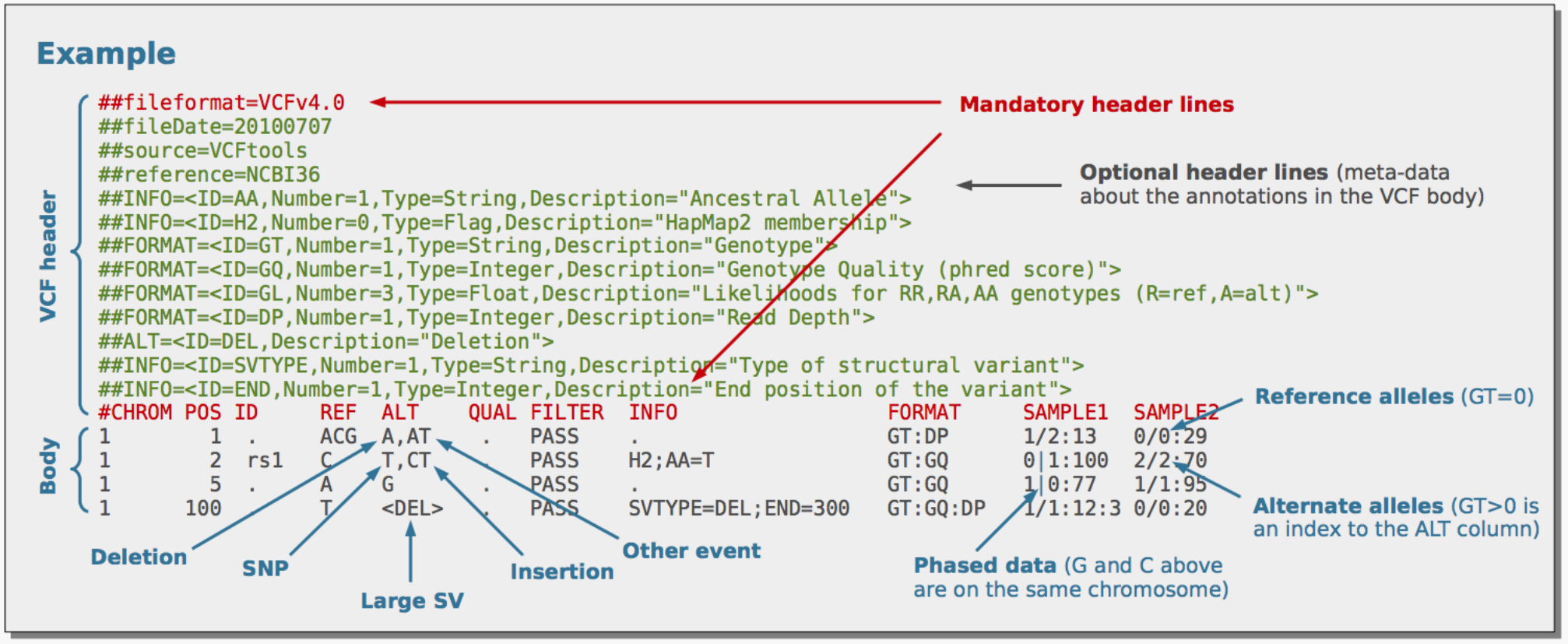
Germline Variant Calling - command

```
gatk HaplotypeCaller \
-R ${reference} \
-I ${input_file}.bam \
--sample-ploidy INT\ # number of chromosomes per sample.
--read-filter OverclippedReadFilter \ # too short after
soft-clipping
--dbsnp ${dbsnp} \
--emit-ref-confidence GVCF \
-L ${intervals} \
-O ${output_file}.g.vcf.gz # Genomic Variant Call Format
```

Germline Variant Calling - command

```
gatk GenotypeGVCFs \
-R ${reference} \
--variant ${output_file}.g.vcf.gz \
--dbsnp ${dbsnp} \
-O ${output_file}.raw.vcf.gz
```

VCF file format



GVCF - Genomic Variant Call Format

Basically, the same format with the VCF format, but with information on all sites, no matter whether there is a variant. It is used for joint variant calling.

```
#CHROM  POS ID  REF ALT QUAL    FILTER  INFO    FORMAT  NA12878
20  10001567    .   A   <NON_REF>   .   .   END=10001616    GT:DP:GQ:MIN_DP:PL  0/0:38:99:34
20  10001617    .   C   A,<NON_REF> 493.77  .   BaseQRankSum=1.632;ClippingRankSum=0.000;DP=
20  10001618    .   T   <NON_REF>   .   .   END=10001627    GT:DP:GQ:MIN_DP:PL  0/0:39:99:37
20  10001628    .   G   A,<NON_REF> 1223.77 .   DP=37;ExcessHet=3.0103;MLEAC=2,0;MLEAF=1.00,
20  10001629    .   G   <NON_REF>   .   .   END=10001660    GT:DP:GQ:MIN_DP:PL  0/0:43:99:38
```

Quality Control on the VCF file

Whether there are too many false positive variants

- **ti/tv ratio** (transition/tranversion ratio): human exome 3.0, genome 2.0

Transition: purine to purine or pyrimidine to pyrimidine, A<->G or C<->T

Tranversion: purine to pyrimidine or vice versa, A<->C or G<->T

- **Number of variants** : human exome (35000 - 55000), genome (4.7 M)
- **Contamination:** whether there are sequences from other individual in the sample (het: allele ratio < 0.25 or >0.75)

Variant Quality Filtration – hard filtering and soft filtering

Hard-filtering consists of choosing specific thresholds for one or more annotations and throwing out any variants that have annotation values above or below the set thresholds. Limiting but possible on small set of variants. – used for target or exome sequencing data

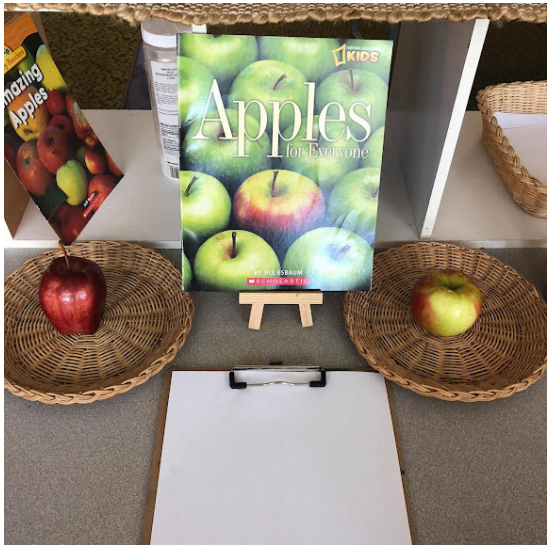
Soft-filtering: uses machine-learning algorithms to learn from the data what are the annotation profiles of good variants (true positives) and of bad variants (false positives) in a particular dataset. - Variant quality score recalibration (VQSR) - **requires a large number of variants and well-curated known variant resources.**

Hard - filtering

>= 2.0 kg PASS

< 2.0 kg FAIL

Soft - filtering



Variant Quality Filtration

Little different for the features and thresholds used for SNPs and Indels.

GATK: SelectVariants

GATK: MergeVcfs

Variant Quality Filtration – hard filtering

One of the most helpful ways to approach hard-filtering is to visualize the distribution of annotation values for a truth set called using a particular pipeline.

Quality, strand bias, mapping quality, base position

Several features could be considered:
QualByDepth (QD): QUAL/AD (Allele depth). normalized the variant score.
FisherStrand (FS): the Phred-scaled probability that there is strand bias at the site.
StrandOddsRatio (SOR): another way to estimate strand bias using a test similar to the symmetric odds ratio test.
RMSMappingQuality (MQ): the root mean square mapping quality over all the reads at the site.
MappingQualityRankSumTest (MQRankSum): It compares the mapping qualities of the reads supporting the reference allele and the alternate allele. A positive value means the mapping qualities of the reads supporting the alternate allele are higher than those supporting the reference allele; a negative value indicates the mapping qualities of the reference allele are higher than those supporting the alternate allele.
ReadPosRankSumTest (ReadPosRankSum): It compares whether the positions of the reference and alternate alleles are different within the reads.

```
1	17359676	rs1022580	C	A	8569.06	.	AC=2;AF=1.00;AN=2;DB;DP=303;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=29.05;SOR=1.358	GT:AD:DP:GQ:PL	1/1:0,295:295:99:8583,881,0
1	17380497	rs2746462	G	T	11539.06	.	AC=2;AF=1.00;AN=2;DB;DP=450;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=26.11;SOR=0.711	GT:AD:DP:GQ:PL	1/1:0,442:442:99:11553,1307,0
1	45796269	rs3219493	G	C	11072.06	.	AC=2;AF=1.00;AN=2;DB;DP=410;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=27.14;SOR=0.796	GT:AD:DP:GQ:PL	1/1:0,408:408:99:11086,1219,0
1	45797505	rs3219489	C	G	6354.64	.	AC=1;AF=0.500;AN=2;BaseQRankSum=0.818;DB;DP=597;ExcessHet=0.0000;FS=0.521;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=11.13;ReadPosRankSum=0.303;SOR=0.769	GT:AD:DP:GQ:PL	0/1:281,290:571:99:6362,0,6018
1	45798555	rs3219487	T	C	18604.06	.	AC=2;AF=1.00;AN=2;DB;DP=697;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=27.40;SOR=0.875	GT:AD:DP:GQ:PL	1/1:0,679:679:99:18618,2027,0
1	183086757	rs2296292	A	C	4754.64	.	AC=1;AF=0.500;AN=2;BaseQRankSum=-9.380e-01;DB;DP=529;ExcessHet=0.0000;FS=0.000;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=9.16;ReadPosRankSum=1.08;SOR=0.707	GT:AD:DP:GQ:PL	0/1:280,239:519:99:4762,0,6002
2	30381505	rs1137288	T	C	3278.64	.	AC=1;AF=0.500;AN=2;BaseQRankSum=0.671;DB;DP=362;ExcessHet=0.0000;FS=1.251;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=9.08;ReadPosRankSum=1.56;SOR=0.786	GT:AD:DP:GQ:PL	0/1:206,155:361:99:3286,0,4598
2	47601106	rs1126497	T	C	4930.64	.	AC=1;AF=0.500;AN=2;BaseQRankSum=-7.860e-01;DB;DP=548;ExcessHet=0.0000;FS=1.106;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=9.20;ReadPosRankSum=0.268;SOR=0.694	GT:AD:DP:GQ:PL	0/1:296,240:536:99:4938,0,6525
2	47601257	rs4399765	T	C	2189.64	.	AC=1;AF=0.500;AN=2;BaseQRankSum=0.076;DB;DP=253;ExcessHet=0.0000;FS=1.000;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=8.97;ReadPosRankSum=-1.461e+00;SOR=0.764	GT:AD:DP:GQ:PL	0/1:139,105:244:99:2197,0,3160
2	47635523	rs201372136	CT	C	519.60	.	AC=1;AF=0.500;AN=2;BaseQRankSum=1.26;DB;DP=392;ExcessHet=0.0000;FS=0.488;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=1.57;ReadPosRankSum=1.23;SOR=0.640	GT:AD:DP:GQ:PL	0/1:281,50:356:99:527,0,5535
2	47641559	.	TAAAAAAAAAA	T	68.60	.	AC=1;AF=0.500;AN=2;DP=20;ExcessHet=0.0000;FS=0.000;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=17.15;SOR=0.916	GT:AD:DP:GQ:PL	0/1:2,2:5:76:76,0,100
```

Variant Quality annotation – hard filtering - command

```
gatk VariantFiltration \
-R ${reference} \
--variant ${output_file}.raw.snp/indel.vcf.gz \
--filter-expression "QD < 2.0" --filter-name "QD_failed” \
--filter-expression "FS > 60.0" --filter-name "FS_failed” \
......
-O "variants.${mode}.filtered.vcf"
```

```
1	17359676	rs1022580	C	A	8569.06	.	AC=2;AF=1.00;AN=2;DB;DP=303;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=29.05;SOR=1.358	GT:AD:DP:GQ:PL	1/1:0,295:295:99:8583,881,0
1	17380497	rs2746462	G	T	11539.06	.	AC=2;AF=1.00;AN=2;DB;DP=450;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=26.11;SOR=0.711	GT:AD:DP:GQ:PL	1/1:0,442:442:99:11553,1307,0
1	45796269	rs3219493	G	C	11072.06	.	AC=2;AF=1.00;AN=2;DB;DP=410;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=27.14;SOR=0.796	GT:AD:DP:GQ:PL	1/1:0,408:408:99:11086,1219,0
1	45797505	rs3219489	C	G	6354.64	.	AC=1;AF=0.500;AN=2;BaseQRankSum=0.818;DB;DP=597;ExcessHet=0.0000;FS=0.521;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=11.13;ReadPosRankSum=0.303;SOR=0.769	GT:AD:DP:GQ:PL	0/1:281,290:571:99:6362,0,6018
1	45798555	rs3219487	T	C	18604.06	.	AC=2;AF=1.00;AN=2;DB;DP=697;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=27.40;SOR=0.875	GT:AD:DP:GQ:PL	1/1:0,679:679:99:18618,2027,0
1	183086757	rs2296292	A	C	4754.64	.	AC=1;AF=0.500;AN=2;BaseQRankSum=-9.380e-01;DB;DP=529;ExcessHet=0.0000;FS=0.000;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=9.16;ReadPosRankSum=1.08;SOR=0.707	GT:AD:DP:GQ:PL	0/1:280,239:519:99:4762,0,6002
2	30381505	rs1137288	T	C	3278.64	.	AC=1;AF=0.500;AN=2;BaseQRankSum=0.671;DB;DP=362;ExcessHet=0.0000;FS=1.251;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=9.08;ReadPosRankSum=1.56;SOR=0.786	GT:AD:DP:GQ:PL	0/1:206,155:361:99:3286,0,4598
```

```
1	17359676	rs1022580	C	A	8569.06	PASS	AC=2;AF=1.00;AN=2;DB;DP=303;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=29.05;SOR=1.358	GT:AD:DP:GQ:PL	1/1:0,295:295:99:8583,881,0
1	17380497	rs2746462	G	T	11539.06	PASS	AC=2;AF=1.00;AN=2;DB;DP=450;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=26.11;SOR=0.711	GT:AD:DP:GQ:PL	1/1:0,442:442:99:11553,1307,0
1	45796269	rs3219493	G	C	11072.06	PASS	AC=2;AF=1.00;AN=2;DB;DP=410;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=27.14;SOR=0.796	GT:AD:DP:GQ:PL	1/1:0,408:408:99:11086,1219,0
1	45797505	rs3219489	C	G	6354.64	PASS	AC=1;AF=0.500;AN=2;BaseQRankSum=0.818;DB;DP=597;ExcessHet=0.0000;FS=0.521;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=11.13;ReadPosRankSum=0.303;SOR=0.769	GT:AD:DP:GQ:PL	0/1:281,290:571:99:6362,0,6018
1	45798555	rs3219487	T	C	18604.06	PASS	AC=2;AF=1.00;AN=2;DB;DP=697;ExcessHet=0.0000;FS=0.000;MLEAC=2;MLEAF=1.00;MQ=60.00;QD=27.40;SOR=0.875	GT:AD:DP:GQ:PL	1/1:0,679:679:99:18618,2027,0
1	183086757	rs2296292	A	C	4754.64	PASS	AC=1;AF=0.500;AN=2;BaseQRankSum=-9.380e-01;DB;DP=529;ExcessHet=0.0000;FS=0.000;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=9.16;ReadPosRankSum=1.08;SOR=0.707	GT:AD:DP:GQ:PL	0/1:280,239:519:99:4762,0,6002
2	30381505	rs1137288	T	C	3278.64	PASS	AC=1;AF=0.500;AN=2;BaseQRankSum=0.671;DB;DP=362;ExcessHet=0.0000;FS=1.251;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=9.08;ReadPosRankSum=1.56;SOR=0.786	GT:AD:DP:GQ:PL	0/1:206,155:361:99:3286,0,4598
2	47601106	rs1126497	T	C	4930.64	PASS	AC=1;AF=0.500;AN=2;BaseQRankSum=-7.860e-01;DB;DP=548;ExcessHet=0.0000;FS=1.106;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=9.20;ReadPosRankSum=0.268;SOR=0.694	GT:AD:DP:GQ:PL	0/1:296,240:536:99:4938,0,6525
2	47601257	rs4399765	T	C	2189.64	PASS	AC=1;AF=0.500;AN=2;BaseQRankSum=0.076;DB;DP=253;ExcessHet=0.0000;FS=1.000;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=8.97;ReadPosRankSum=-1.461e+00;SOR=0.764	GT:AD:DP:GQ:PL	0/1:139,105:244:99:2197,0,3160
2	47635523	rs201372136	CT	C	519.60	QD_failed	AC=1;AF=0.500;AN=2;BaseQRankSum=1.26;DB;DP=392;ExcessHet=0.0000;FS=0.488;MLEAC=1;MLEAF=0.500;MQ=60.00;MQRankSum=0.00;QD=1.57;ReadPosRankSum=1.23;SOR=0.640	GT:AD:DP:GQ:PL	0/1:281,50:356:99:527,0,5535
```

Variant Quality annotation – soft filtering (VQSR)

Steps:

- **VariantRecalibrator builds the model(s)**

This model attempts to describe the relationship between variant annotations (e.g. QD, MQ and ReadPosRankSum) and the probability that a variant is a true genetic variant versus a sequencing or data processing artifact.

- **ApplyVQSR applies a filtering threshold**

This adaptive error model can then be applied to both known and novel variation discovered in the call set of interest to evaluate the probability that each call is real. The result is a score called the VQSLOD that gets added to the INFO field of each variant. This score is the log odds of being a true variant versus being false under the trained Gaussian mixture model.

Variant Quality annotation – soft filtering - command

```
gatk VariantRecalibrator \
-R ${reference} \
--variant ${output_file}.raw.snp/indel.vcf.gz \
--resource:hapmap,known=false,training=true,truth=true,prior=15.0
${hapmap_ressource} \
...... (other resources)
-an QD \
-an MQRankSum \
-an ReadPosRankSum \
...... (other features)
-tranche 100.0 -tranche 99.5 -tranche 99.0 -tranche 90.0 \
-O ${output_file}.vqsr.output.recal \
--tranches-file ${output_file}.vqsr.output.tranches \
--rscript-file ${output_file}.vqsr.output.plots.R \
--mode SNP/INDEL
```

Variant Quality annotation – soft filtering - command

```
gatk ApplyVQSR \
-R ${reference} \
--variant ${output_file}.raw.snp/indel.vcf.gz \
-O variants.snp/indel.filtered.vcf \
--truth-sensitivity-filter-level 99.0 \
--tranches-file ${output_file}.vqsr.output.tranches \
--recal-file ${output_file}.vqsr.output.recal \
--mode SNP/INDEL
```

Exercise II

- Running the 03,04,05 scripts to do bam quality control, variant calling and variant filtration
- Find PCT_PF_READS_ALIGNED in NA12878F.CollectAlignmentSummaryMetrics.easy.txt
- Find MEDIAN_INSERT_SIZE, MEDIAN_ABSOLUTE_DEVIATION in NA12878F.CollectInsertSizeMetrics.easy.txt
- Find BAIT_DESIGN_EFFICIENCY, OFF_BAIT_BASES , MEDIAN_TARGET_COVERAGE, PCT_TARGET_BASES_10X, PCT_TARGET_BASES_20X in NA12878F.CollectHsMetrics.easy.txt
- Find how many variants in 04_variantfiltration/NA12878F.final.vcf (grep –v '^#' NA12878F.final.vcf|wc -l)
- Find how many PASS variants in 04_variantfiltration/ NA12878F.final.vcf (grep 'PASS' NA12878F.final.vcf|wc -l)

Exercise II - continue

- Revisit the variants from the Exercise I and check how it looks like in the vcf file:

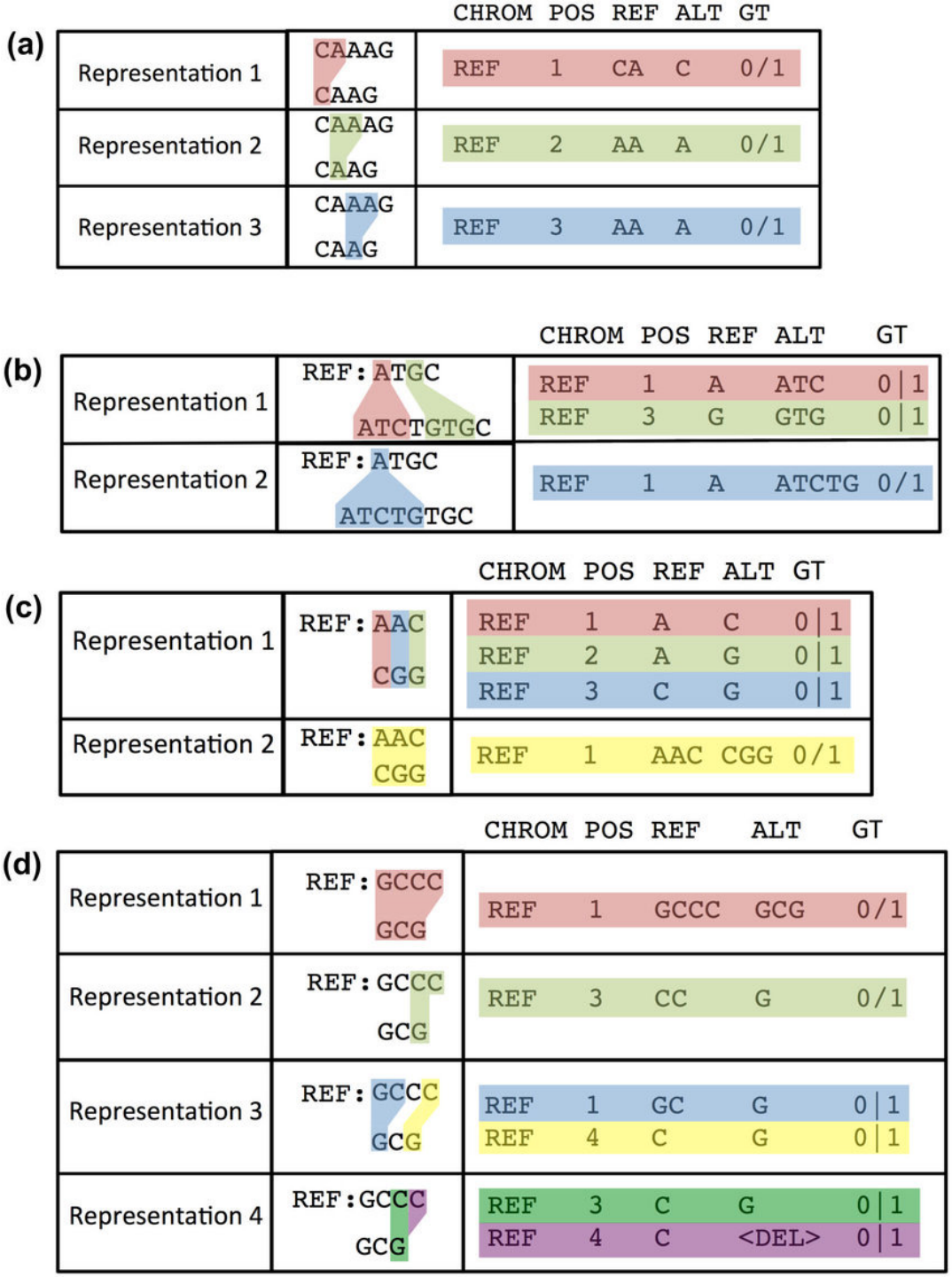
1:17,380,476-17,380,516 (SNP),

2:48,032,854-48,032,893(deletion),

11:108,151,687-108,151,726(insertion))

Variant Functional annotation – preprocess - normalization

Standardize indel representation:
the left-most and right-most positions at which a particular indel could be represented in a VCF file

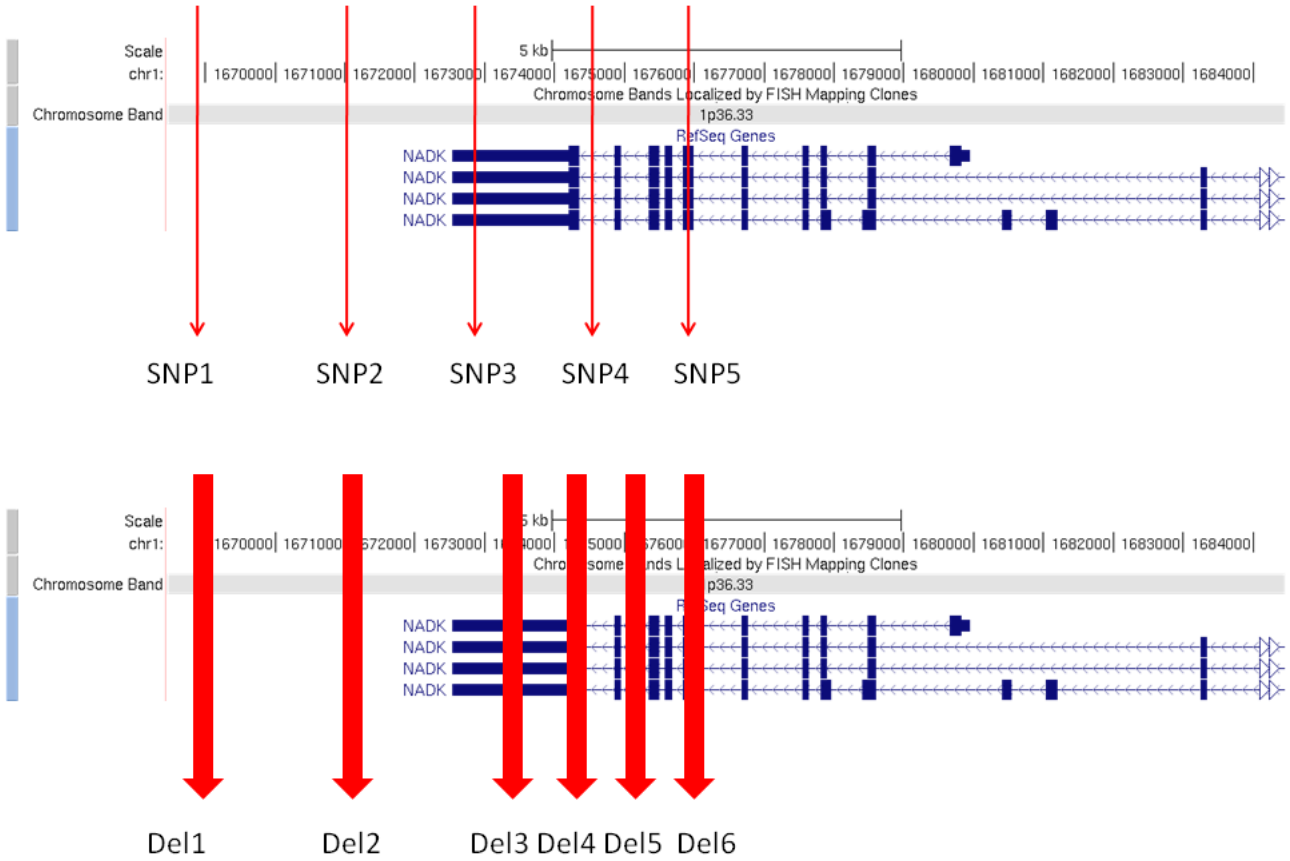


Variant Functional annotation – preprocess - decomposition

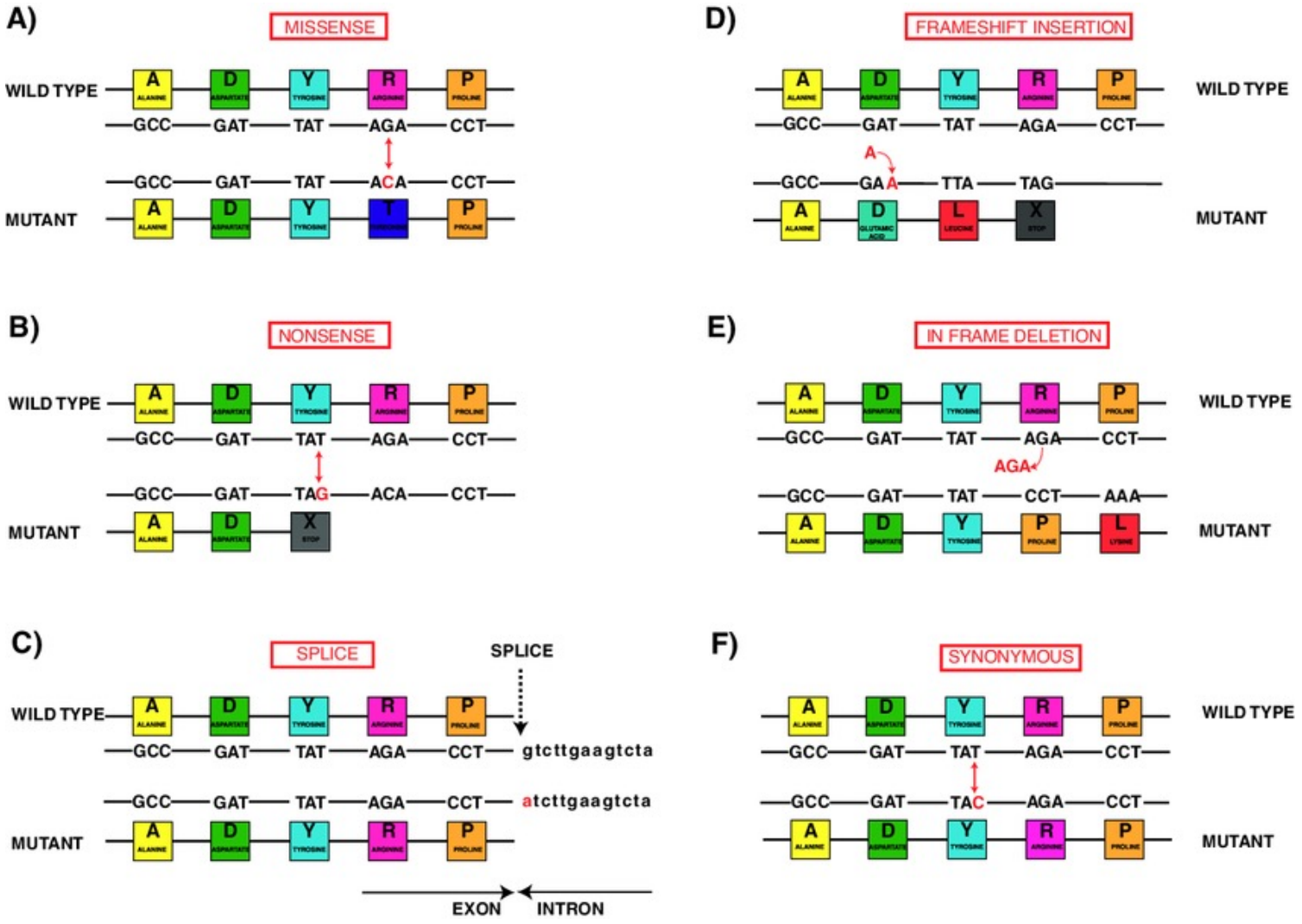
```
  #before decomposition
  #CHROM  POS     ID   REF     ALT         QUAL   FILTER  INFO                  FORMAT    S1                                          S2
  1       3759889 .    TA      TAA,TAAA,T  .      PASS    AF=0.342,0.173,0.037  GT:DP:PL          1/2:81:281,5,9,58,0,115,338,46,116,809
0/0:86:0,30,323,31,365,483,38,291,325,567

  #after decomposition
  #CHROM  POS     ID   REF     ALT         QUAL   FILTER  INFO                                          FORMAT   S1               S2
  1       3759889 .    TA      TAA         .      PASS    OLD_MULTIALLELIC=1:3759889:TA/TAA/TAAA/T      GT:PL    1/.:281,5,9
0/0:0,30,323
  1       3759889 .    TA      TAAA        .      .       OLD_MULTIALLELIC=1:3759889:TA/TAA/TAAA/T      GT:PL    ./1:281,58,115
0/0:0,31,483
  1       3759889 .    TA      T           .      .       OLD_MULTIALLELIC=1:3759889:TA/TAA/TAAA/T      GT:PL    ./.:281,338,809
0/0:0,38,567
```

Variant Functional annotation – Annovar - Position

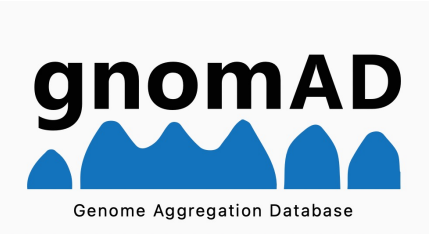
A tool (written with perl) with providing different data source. It can be used to functionally annotate genetic variants detected from diverse genomes, human as well as mouse, worm, fly, yeast, SARS-CoV-2, and many others.



Variant Functional annotation – Annovar - Consequence



Variant Functional annotation – frequencies - gnomAD



The **Genome Aggregation Database** (gnomeAD):

- V2.1.1 (GRCh37/hg19) contains 125,748 healthy exome sequences and 157,078 healthy genome sequencing data
- Free to use
- Analyze all samples with the same pipeline and joint variant calling to improve consistency across the projects
- Quality control on each variant
- It also contains structural variants, mitochondrial variants
- It contains population variant frequencies (e.g. European, East Asian etc.)

https://gnomad.broadinstitute.org/news/2018-10-gnomad-v2-1/

Variant Functional annotation – frequencies – in-house database

If you have large number of samples (> 1000 individuals):

- Prepared with the same sample preparation method
- Sequenced in the same sequencer
- Analyzed with the same pipeline

Build in-house database

Advantage:

Filter out artefacts in your settings.

Variant Functional annotation – classification – Clinvar

- Archive of interpretations of variants relative to conditions
- Variant-level interpretations
  - Assertion/Clinical significance/Interpretation/Classification/
- Fully public and freely available
- Submission-driven database
- Curation support from NCBI staff

CLINSIG: clinical significant

Variant Functional annotation – classification – HGMD

The Human Gene Mutation Database (HGMD) represents an attempt to collate all known (published) gene lesions responsible for human inherited disease.

New release every quarter.

Free (3 year old data, no downloading, limited, accessible from VEP) and licence version.

https://www.hgmd.cf.ac.uk/ac/all.php

Variant Functional annotation – variant functional effect predictors

SIFT (Sorting Intolerant from Tolerant) scores: SIFT uses sequence homology to compute the likelihood that an amino acid substitution will have an adverse effect on protein function. The underlying assumption is that evolutionarily conserved regions tend to be less tolerant of mutations, and hence amino acid substitutions or insertions/deletions in these regions are more likely to affect function **(D: Deleterious; T: tolerated)**

PolyPhen-2 (Polymorphism Phenotyping v2) scores: predicts the possible impact of amino acid substitutions on the stability and function of human proteins using structural and comparative evolutionary considerations. **(D: Probably damaging, P: possibly damaging; B: benign)**

……

CADD (Combined Annotation Dependent Depletion) scores: a method for objectively integrating many diverse annotations into a single measure (C score) for each variant. C scores correlate with allelic diversity, annotations of functionality, pathogenicity, disease severity, experimentally measured regulatory effects and complex trait associations, precompute C scores for all 8.6 billion possible human single-nucleotide variants and enable scoring of short insertions-deletions. (The higher the worst consequence)

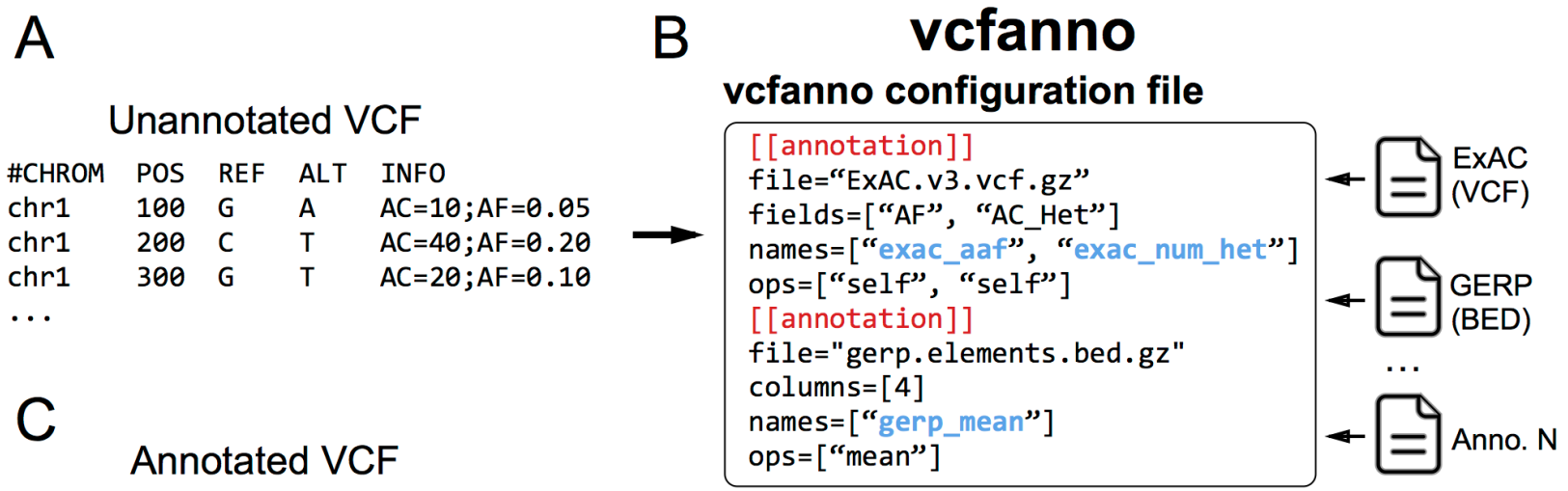
Variant Functional annotation – annovar command

```
perl annovar_2020June07/table_annovar.pl \
        ${input_vcf} \
        annovar_2017Jul16/humandb/ \
        --vcfinput \
        --remove \ # remove all temporary files
        --buildver hg19 \
        --outfile ${output_marker} \
        --otherinfo \ # print out columns after QUAL
        --gff3dbfile repeatMasker_hg19_all.gff3 \
        -protocol
"gff3,refGene,avsnp150,dbnsfp33aReduced,clinvar_20210123,c
add13gt10,gnomad_exomePOPMAX,gnomad_genomePOPMAX" \
       --operation 'r,g,f,f,f,f,f,f’ \
```

View the annotated vcf file under 05_variantannotation (41203088):

Variant Functional annotation – vcfanno

vcfanno allows you to quickly annotate your VCF with any number of INFO fields from any number of VCFs or BED files. It uses a simple conf file to allow the user to specify the source annotation files and fields and how they will be added to the info of the query VCF.



Joint Variant Calling

Variant Calling on multiple samples – higher sensitivity and better genotype

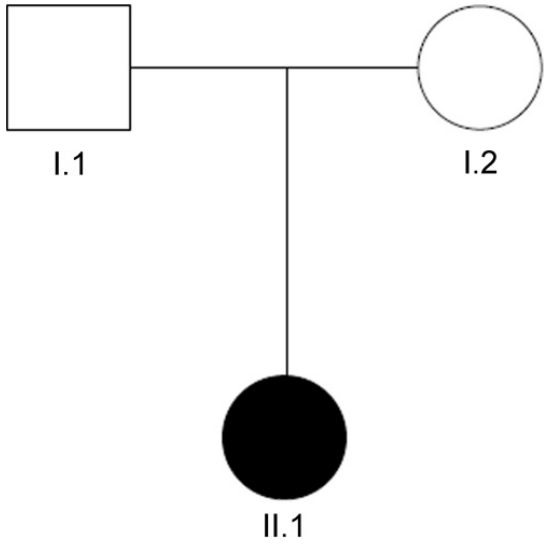
1. Clearer distinction between homozygous reference sites and sites with missing data
Batch-calling does not output a genotype call at sites where no member in the batch has evidence for a variant; it is thus impossible to distinguish such sites from locations missing data. In contrast, joint calling emits genotype calls at every site where any individual in the call set has evidence for variation.

2. Greater sensitivity for low-frequency variants
By sharing information across all samples, joint calling makes it possible to “rescue” genotype calls at sites where a carrier has low coverage but other samples within the call set have a confident variant at that location. However, this does not apply to singletons, which are unique to a single sample. To minimize the chance of missing singletons, we increase the cohort size -- so that singletons themselves have less chance of happening in the first place.

3. Greater ability to filter out false positives (VQSR works better with bigger data)

Joint Variant Calling – trio pedigree file

A pedigree is a structured description of the familial relationships between samples. One row = one person

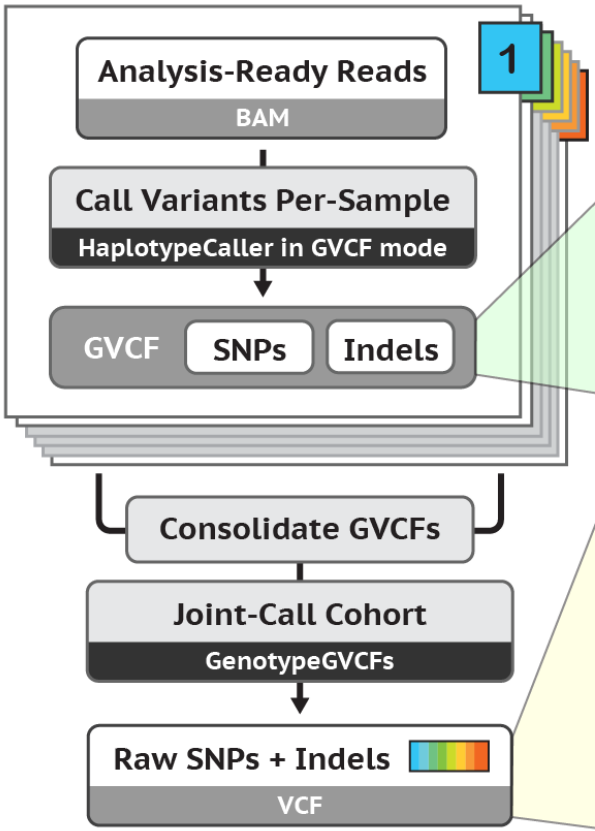


Family ID	Individual ID	Paternal ID	Maternal ID	Sex	Phenotype
Family_1	II.1	I.1	I.2	2	2
Family_1	I.1	0	0	1	1
Family_1	I.2	0	0	2	1

Family_1 II.1 I.1 I.2 2 2
Family_1 I.1 0 0 1 1
Family_1 I.2 0 0 2 1

Sex: 1=male; 2=female; other=unknown
Phenotype: 1=unaffected, 2=affected, 0,-9=missing

Joint Variant Calling - workflow




```
#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
```

Joint Variant Calling - command

g.vcf.gz per individual

```
gatk CombineGVCFs \
-R ${reference} \
--variant proband.g.vcf.gz \
--variant father.g.vcf.gz \
--variant mother.g.vcf.gz \
-ped family_1.ped \
-O family_1.combined.g.vcf.gz
```

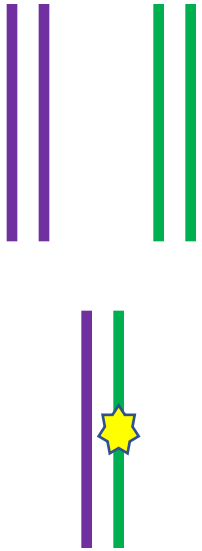
```
gatk GenotypeGVCFs \
-R ${reference} \
--variant family_1.combined.g.vcf.gz \
--dbsnp ${dbsnp} \
--variant mother.g.vcf.gz \
-ped family_1.ped \
-O family_1.raw.vcf.gz
```

Trio analysis - De novo variants and Recessive variants

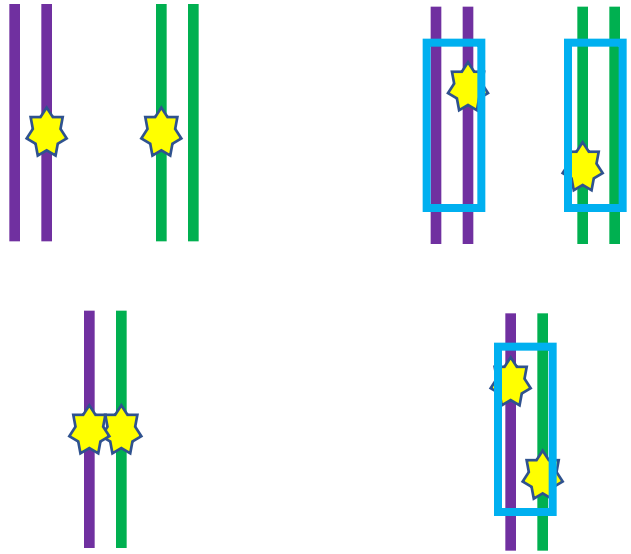
Suppose both parents are healthy, the child has the disease. Try to find differences between the child and the parents.

Parents don’t have: De novo
Parents just have one while the child have two: recessive variants, compound heterzygosity

De novo variants



Recessive variants



Pedigree check and gender detection – Quality Control

Pedigree check: whether the three individuals coming from the same family, by comparing with genotypes, e.g. AA+BB=AB

Gender detection: Coverage on chrX,Y compared those to autosomes

View the joint variant calling vcf file under: 06_jointvc/HG002.filter.vcf

Improvement on alignment

Development on human reference genome:

Hg19: released in 2009, cover ~ 92.5% of human genome

Hg38: released in 2013, cover ~ 95% of human genome

T2T-CHM13: released in 2022, first gapless version of human genome

https://www.youtube.com/watch?v=swNtGe9QWAQ

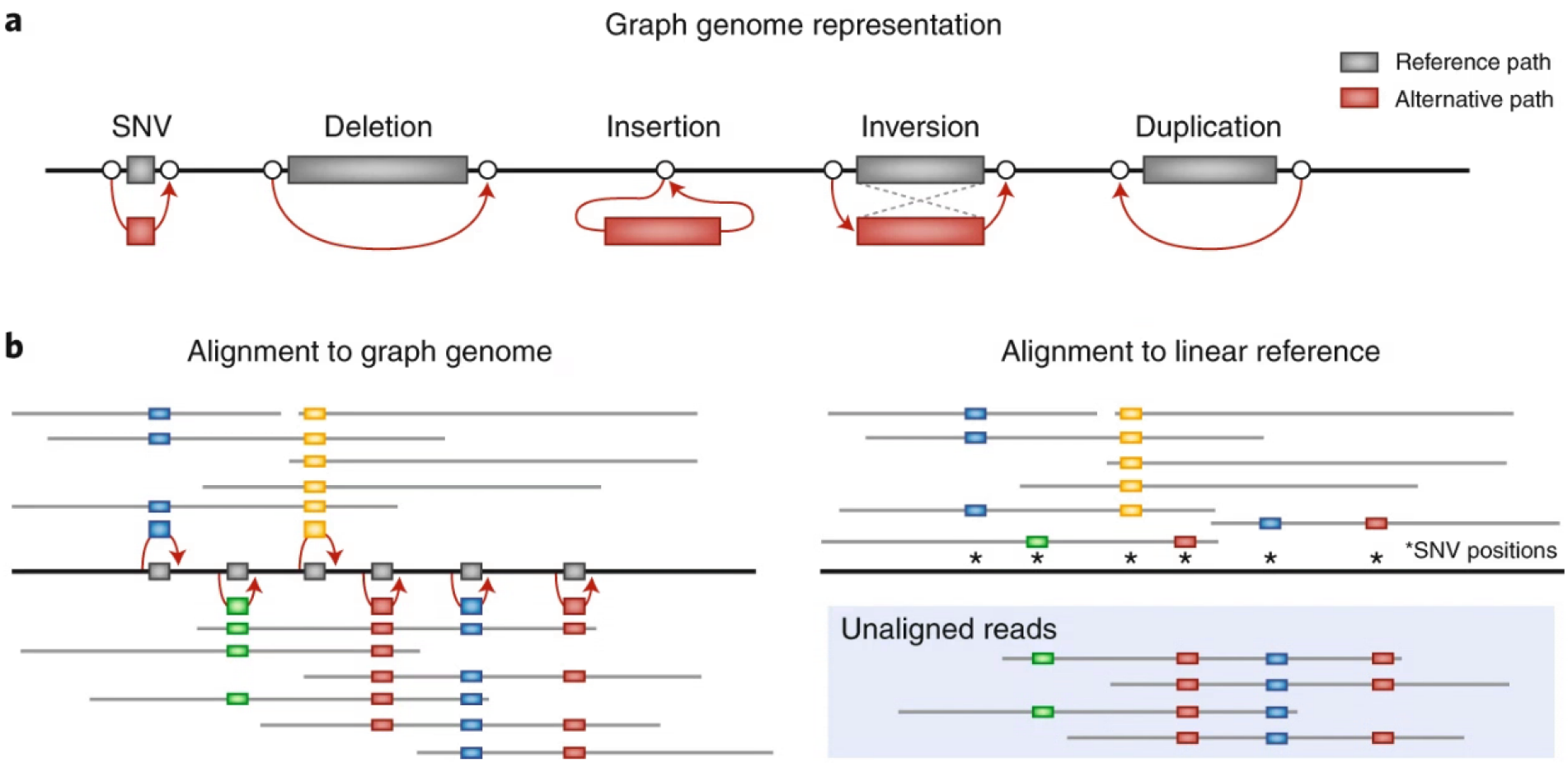
Development on alignment method:

- Graph reference genome
- Alt-aware alignment

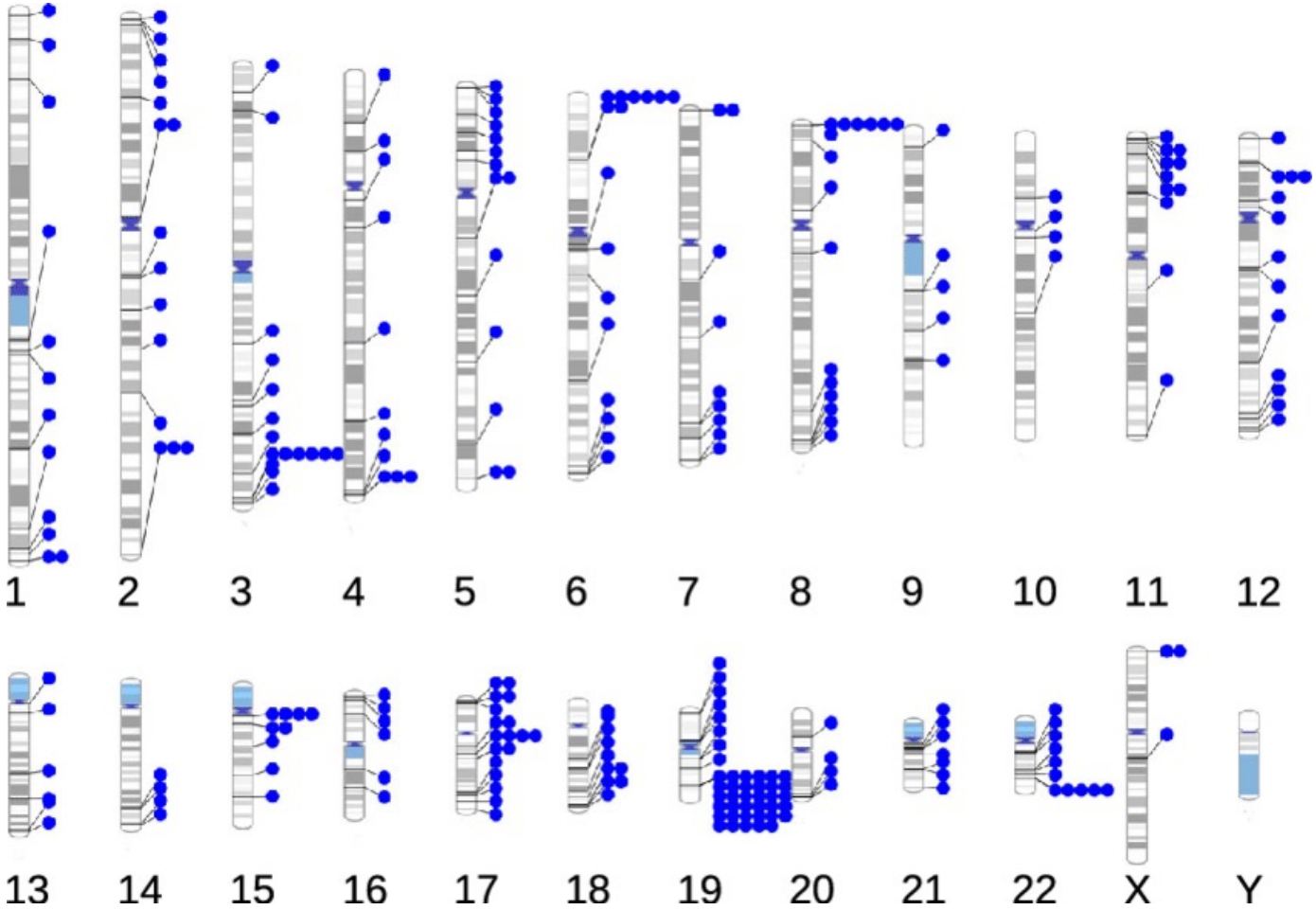
Improvement on alignment – graph reference genome

Fig. 1: Genome graphs and graph alignments.

From: Goodbye reference, hello genome graphs



Improvement on alignment – alt-aware alignment



hg19:
39 alternative loci

hg38:
261 alternative loci

T2T-CHM13:
No alternative loci,
but more complete

Storage, Pipeline, etc.

Storage (human)

	exome	genome
Fastq files	15 G	80 G
BAM files	10 G	80 G

Compression tool can compress file into 1/5 of the original size.
fastq -> ora or genozip, bam -> cram (require reference for decompression)

Pipelines: instead of running commands step by step, it runs the whole workflow without interaction, also consider paralization and catching errors.

Pipeline language: nextflow (nf-core) and WDL (GATK)

Genome in a bottle (GIAB) – control samples and validation

https://www.nist.gov/programs-projects/genome-bottle

Benchmark (or "High-confidence") variant calls and regions.

For the same control samples, collect variant calls from different sequencing platform and different analysis pipelines. Find consensus calls and confident regions to create a ‘true’ set.

Free for downloading.

Have both single samples and trio samples.