GULBENKIAN SCIENCE

PO21: Precision Oncology Course

Variant Detection

BU CNIO BIOINFORMATICS UNIT

cnio stop cancer

Outline

- Definition and types of genomic variants
- Genomic variants in cancer
- Steps for variant detection after alignment
- Algorithms for variant calling
- Pipeline for targeted DNA sequencing: varca

Genomic Variants

Definition, relevance and types

What are genomic variants?

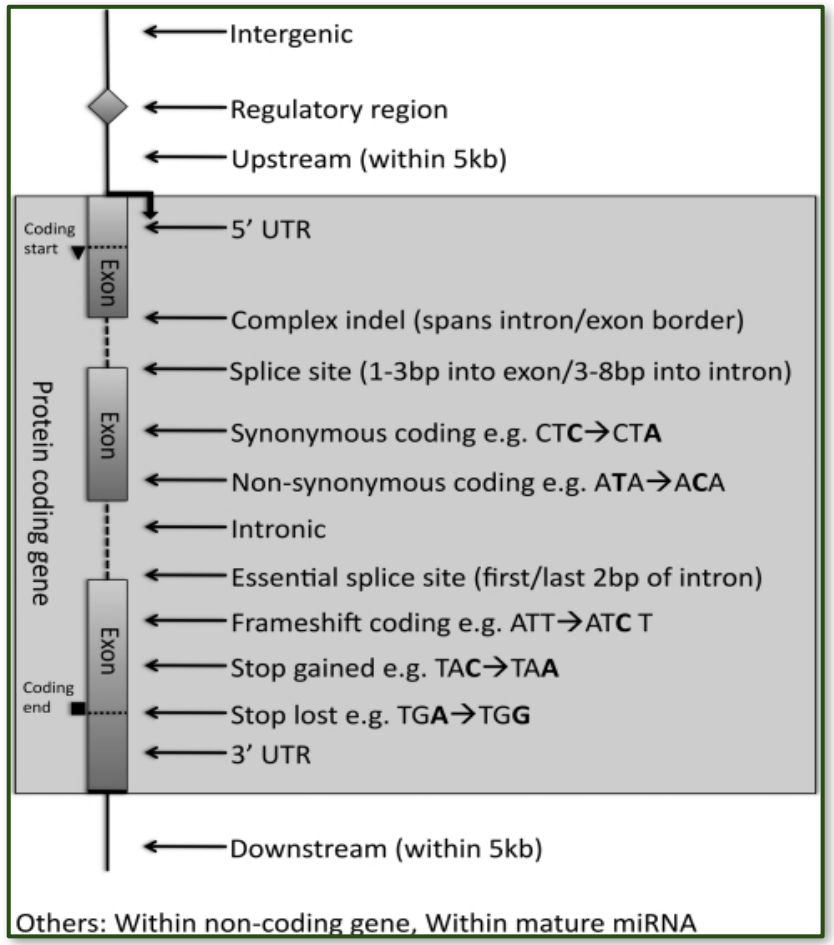
- Genomic variants are **permanent** changes in the DNA sequence of an organism.

- They can **emerge by different mechanisms**:
  - Recombination during gametes formation.
  - Errors during the DNA replication.
  - External factors like radiation, viruses, transposons, tobacco, UV light.

Types of genomic variants

- Different types of variants according to different criteria:
  - Variant **size**
  - Variant **position** in the DNA sequence
  - **Consequence** of the variant in transcription and translation
  - **Clinical implication**

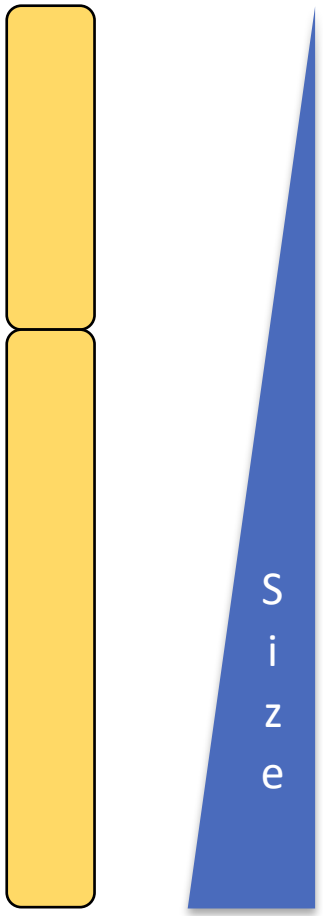
Classification of variants according to the sequence position



McLaren et al. Bioinformatics. 2010.

- Intergenic
- In regulatory regions
- Upstream
- Downstream
- In genes
  - Untranslated regions: 5'UTR y 3'UTR
  - Exons
  - Introns
  - Splicing regions

Classification of variants according to size



SNVs Single Nucleotide Variants (SNP)
Indels Small Insertion and Deletions

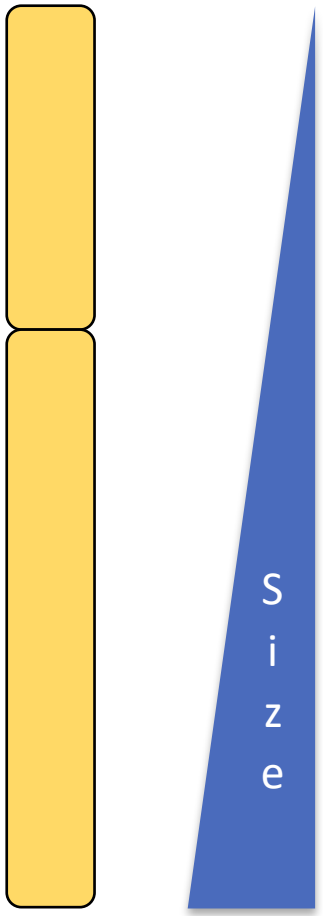
VNTRs (Micro, Minisatellites)
Variable Number of Tandem Repetitions

CNVs Copy Number Variations
Translocations, Inversions

Structural variants

Aneuploidies

Classification of variants according to size



SNVs Single Nucleotide Variants (SNP)
Indels Small Insertion and Deletions

Small scale variants

VNTRs (Micro, Minisatellites)
Variable Number of Tandem Repetitions

CNVs Copy Number Variations
Translocations, Inversions

Aneuploidies

Genomic Variants in Cancer

Intra-tumoral heterogeneity

Variant Allele Frequency

Proportion of DNA molecules in the sample carrying the variant

$$VAF = \frac{sequence\ reads\ with\ a\ DNA\ variant}{overall\ coverage\ at\ that\ locus}$$

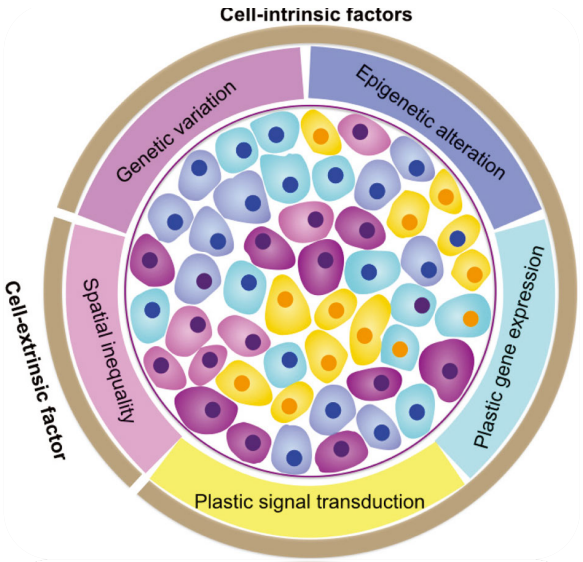
For a diploid organism:

- **heterozygous loci** should be near 0.5 VAF
- **homozygous loci** should be near 1 VAF
- **reference loci** should be near 0 VAF

doi: 10.28092/j.issn.2095-3941.2016.0004

Clonal composition in cancer

changes 0.5/1.0 diploid variant allele frequencies



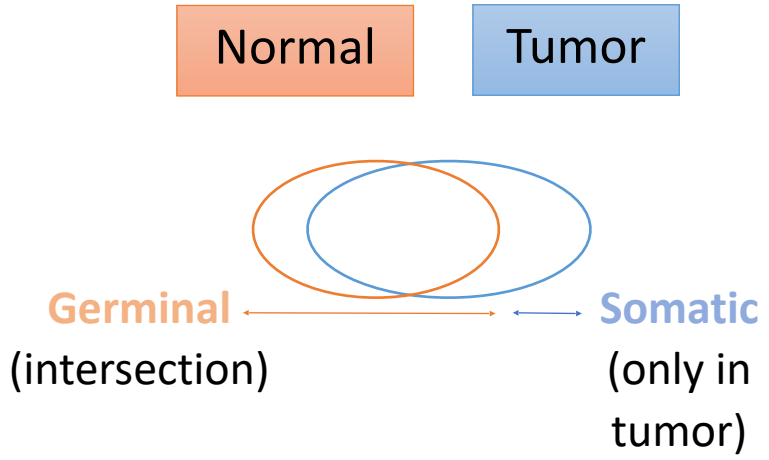
doi: 10.1038/aps.2015.92

Somatic vs Germline variants

Germline: appear in gametes
Inheritable
Affect to future generations
e.g.: variants involved in rare diseases

Somatic: appear in different from germline cells
Acquired
Only affect to the linage of the affected cell
e.g.: variants causing cancer

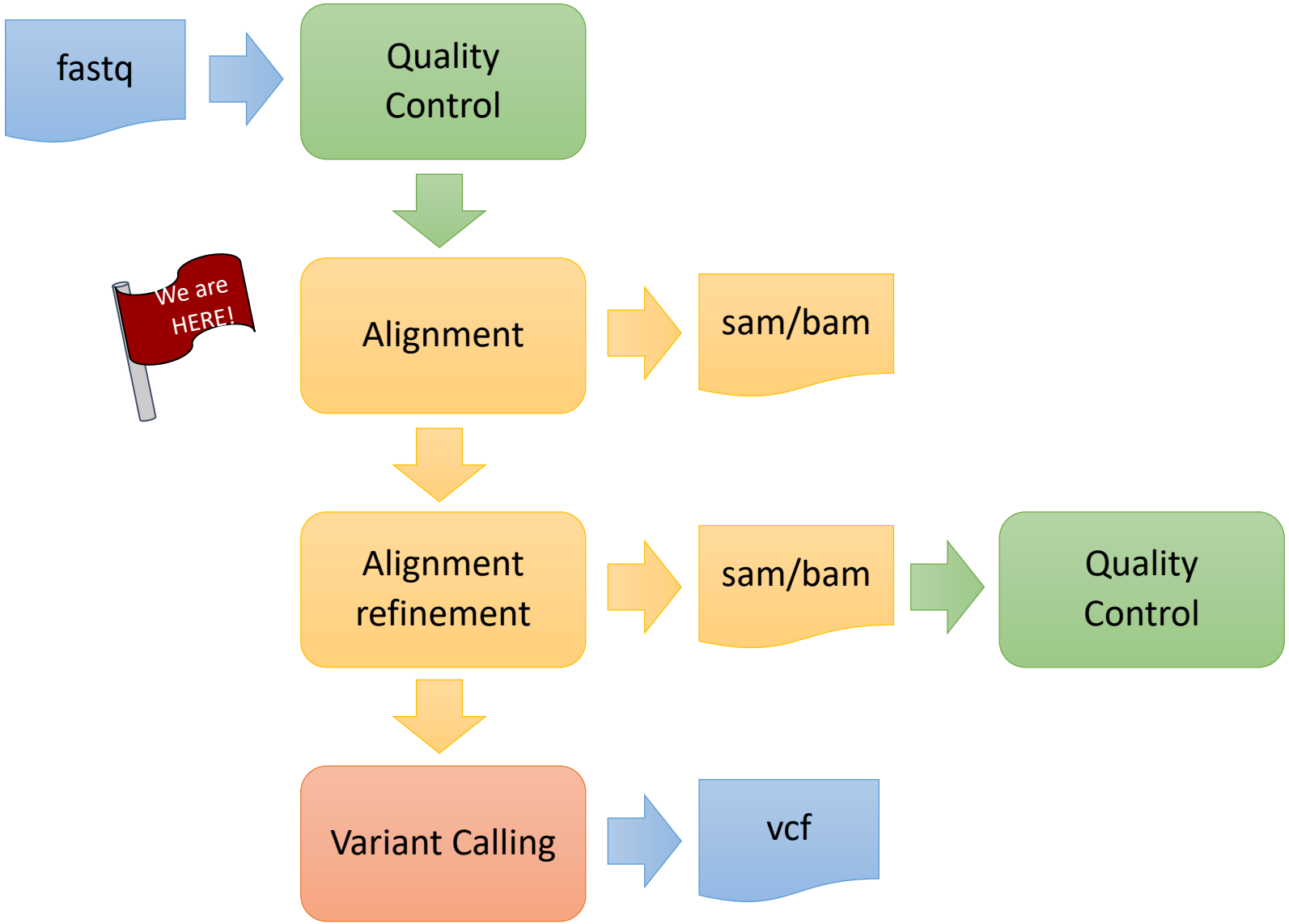
Identified by comparison:



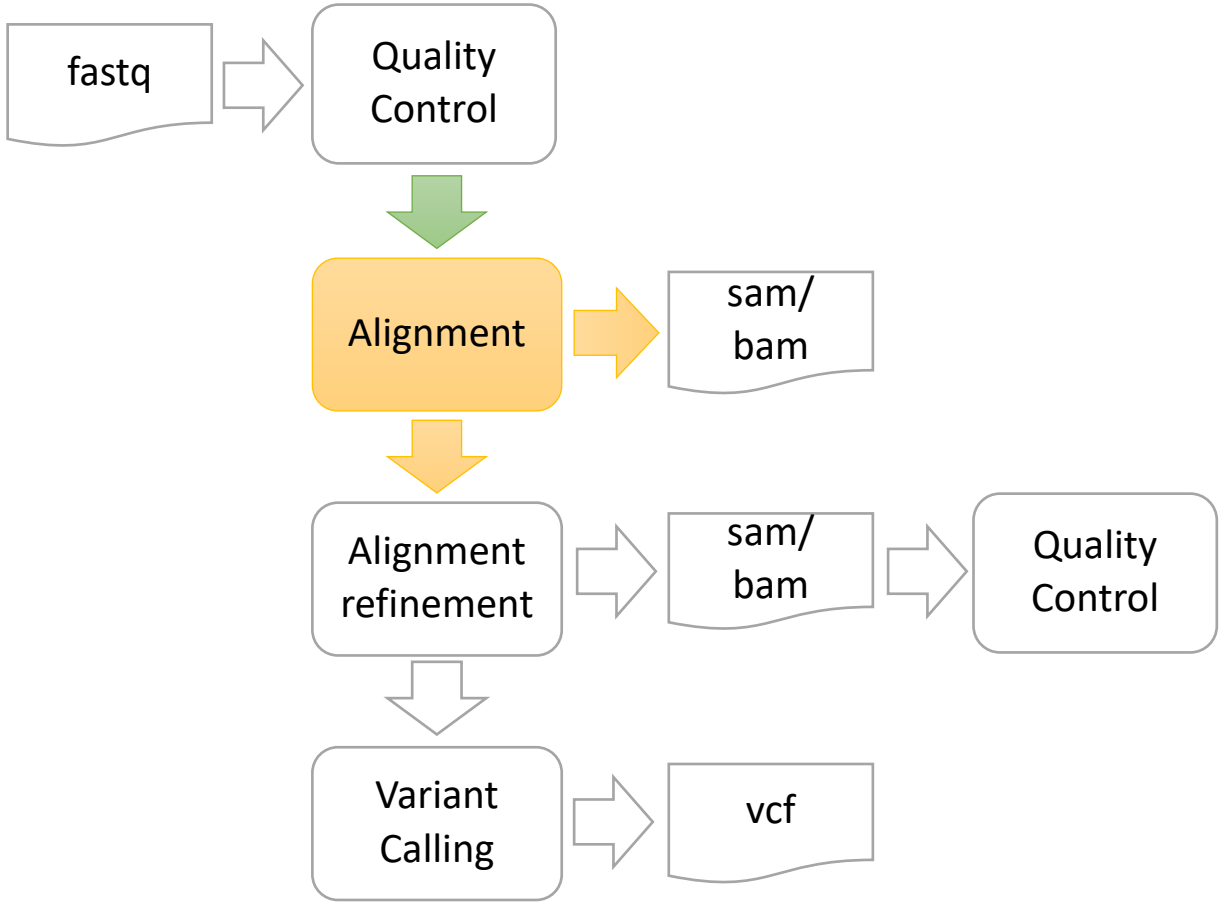
Variant detection

Sequencing and bioinformatics process

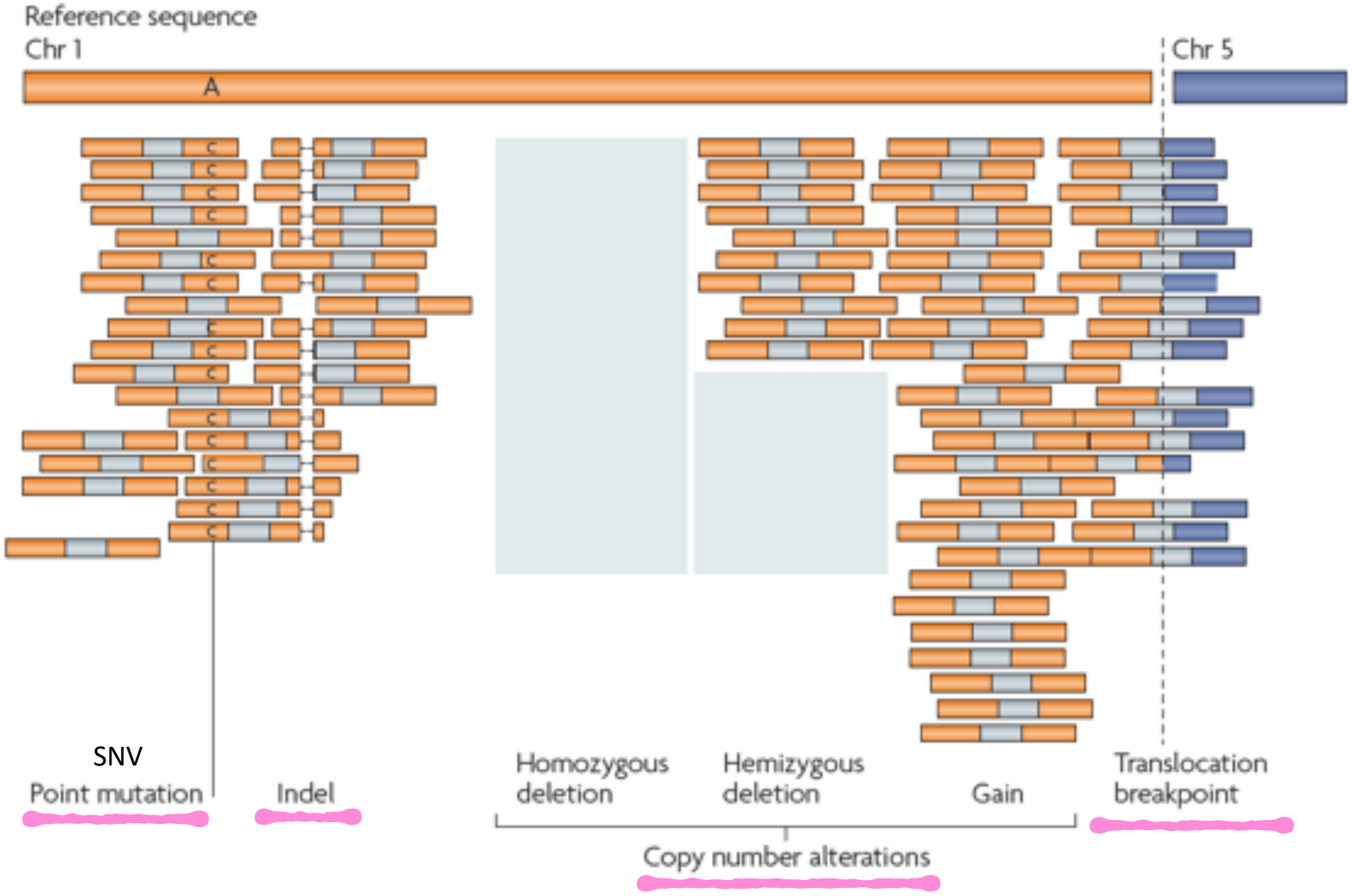
General steps



Alignment

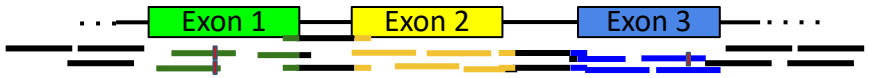

fastq
Quality Control
Alignment
sam/ bam
Alignment refinement
sam/ bam
Quality Control
Variant Calling
vcf

Alignment of reads uncover potential variant sites

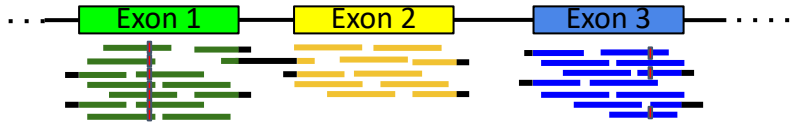


DNA-seq strategies – Sequencing coverage

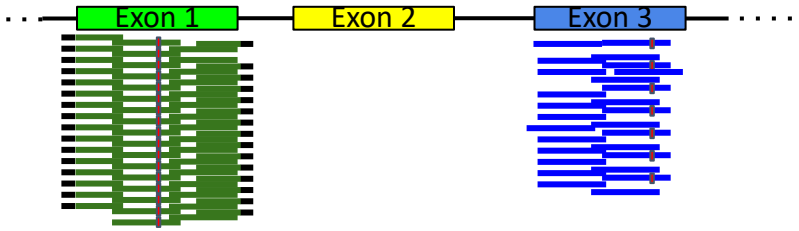
Whole-Genome Sequencing

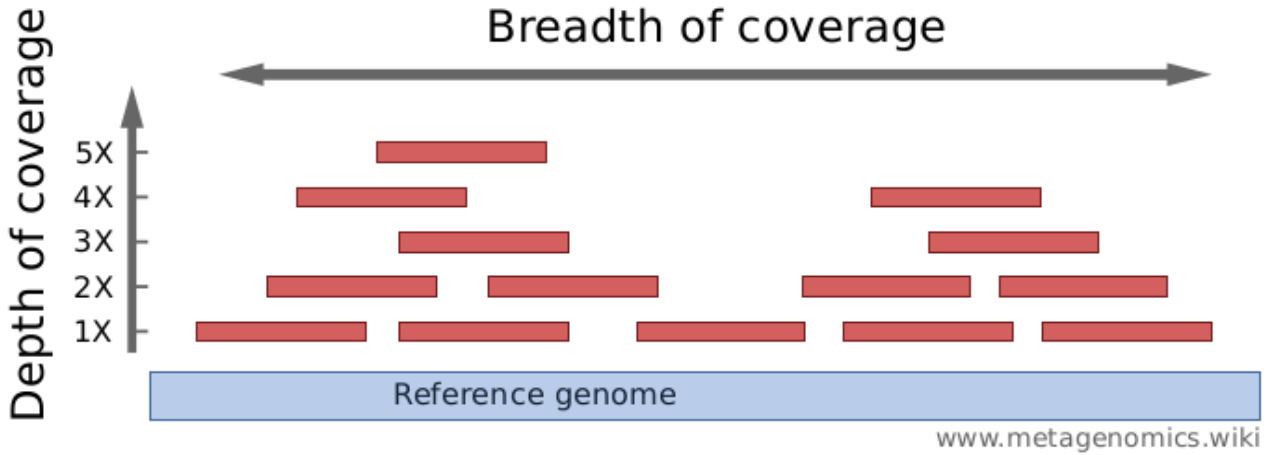


Whole-Exome Sequencing



Panel of genes by amplicon/PCR approach





Depth: times that a base is sequenced

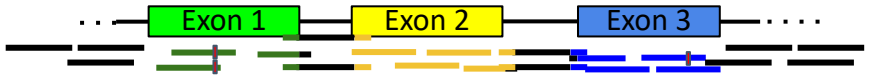
On average:

$$\frac{\sum \textit{Number of times a sequenced base is covered by reads}}{\textit{Length of the sequenced genome}}$$

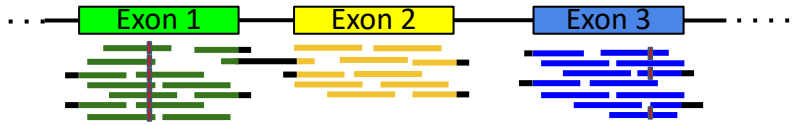
Breadth: percentage of the sequenced genome covered by the reads (at a certain depth)

DNA-seq strategies – Type of variants

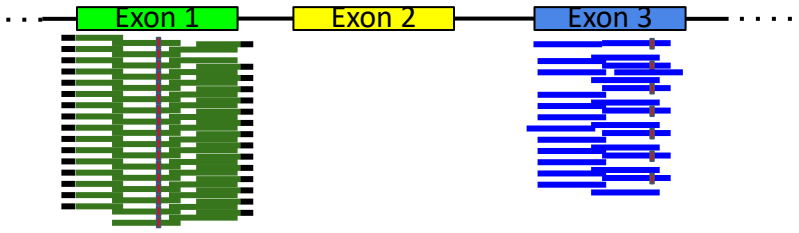
Whole-Genome Sequencing



Whole-Exome Sequencing



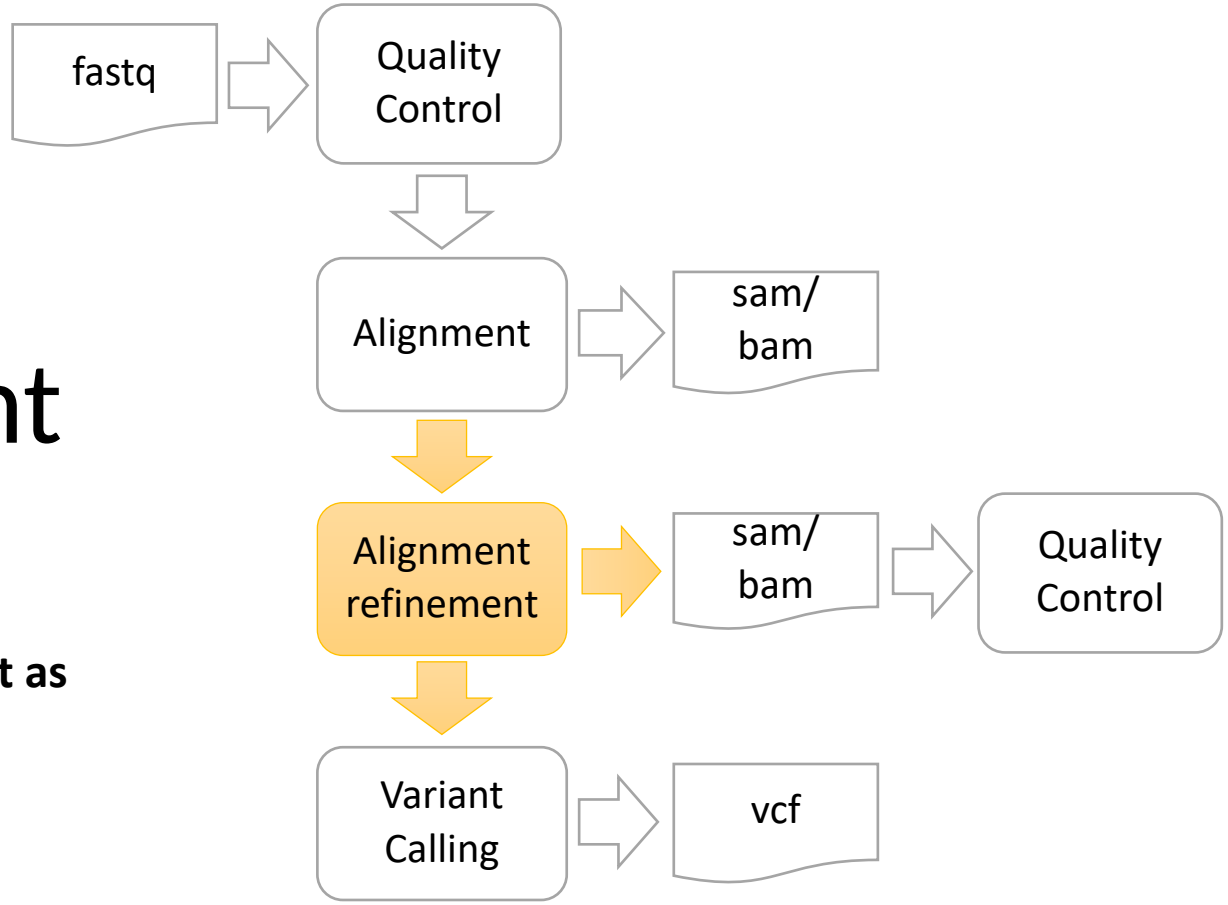
Panel of genes by amplicon/PCR approach



Target	Type of variants discovered	Avg depth per pos	Cost
Entire genome	Coding variants, intronic and regulatory sites. Structural variants #Variants= 3M - 4M.	> 30x Most uniform	High
2% of the genome	Coding variants Some intronic and regulatory sites. Issues in the detection of structural variants **#**Variants= 20k - 60k.	> 50x - 100x	Low
Variable	Depends on the design (customizable) Challenging detection of structural variants # variants = *ND*	> 500x	Lower

Alignment refinement

Variant calling requires the most perfect alignment as possible to avoid False Positives.



Mark/remove duplicates

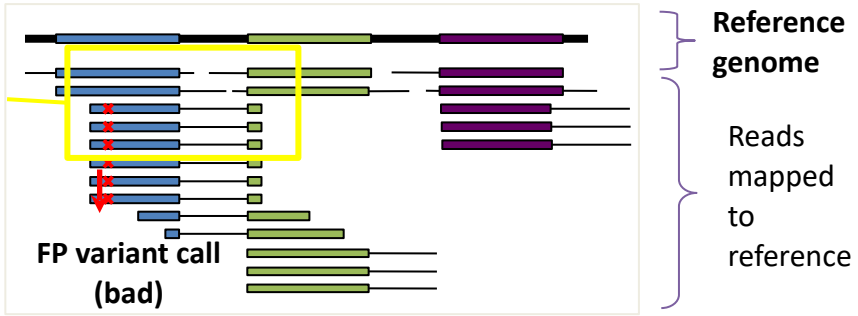
- Duplicates derive from **PCR amplification** (library preparation): one fragment is sequenced multiple times.
- An error at the beginning of the PCR (first steps) is propagated.

- Duplicates in hybrid-seq are **worthless** for the subsequent analysis:
  Duplicates are source of False Positives calls while only provide redundancy.

Solution: retrieve the best one, discard the duplicates:

METHOD: by **Picard-tools**
http://broadinstitute.github.io/picard/
(alternatives : **samtools**)

Duplicates share the **same alignment properties** : sequence, start and end positions



✖ = sequencing error propagated in duplicates

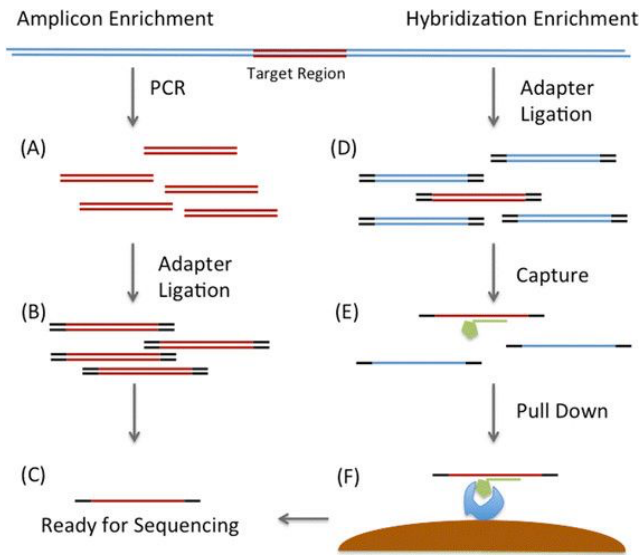
MARK/REMOVE DUPLICATES

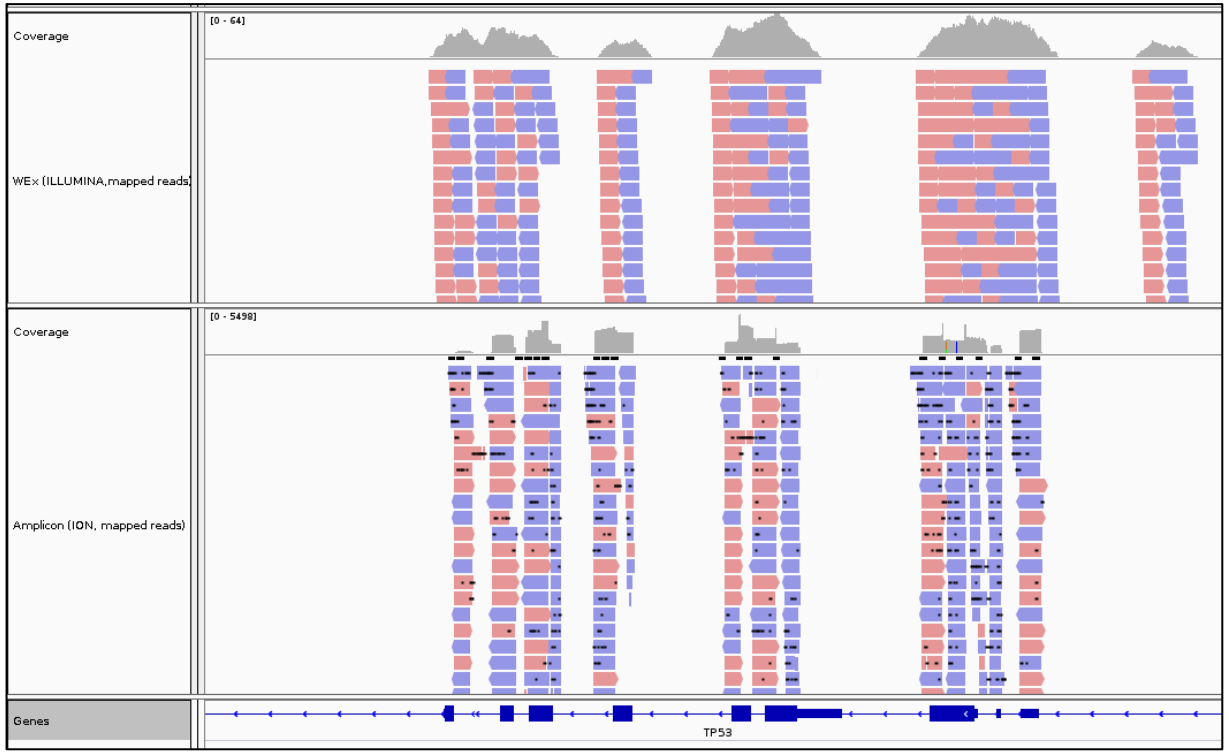
After marking/removing duplicates, the variant caller will only see :

… and thus be more likely to make the right call

Adapted from GATK

Mark/remove duplicates: Amplicon seq





WARNING: **Do NOT remove duplicates in data derived from** amplicon techniques (**Ion Torrent**).
More info.: https://github.com/broadgsa/gatk/blob/master/doc_archive/tutorials/(How_to)_Mark_duplicates_with_MarkDuplicates_or_MarkDuplicatesWithMateCigar.md

Indel realignment

- Algorithms align reads very fast with high accuracy, but not perfectly.

  During alignment, ***penalties on mismatches are much cheaper than gaps*** *(indels).*

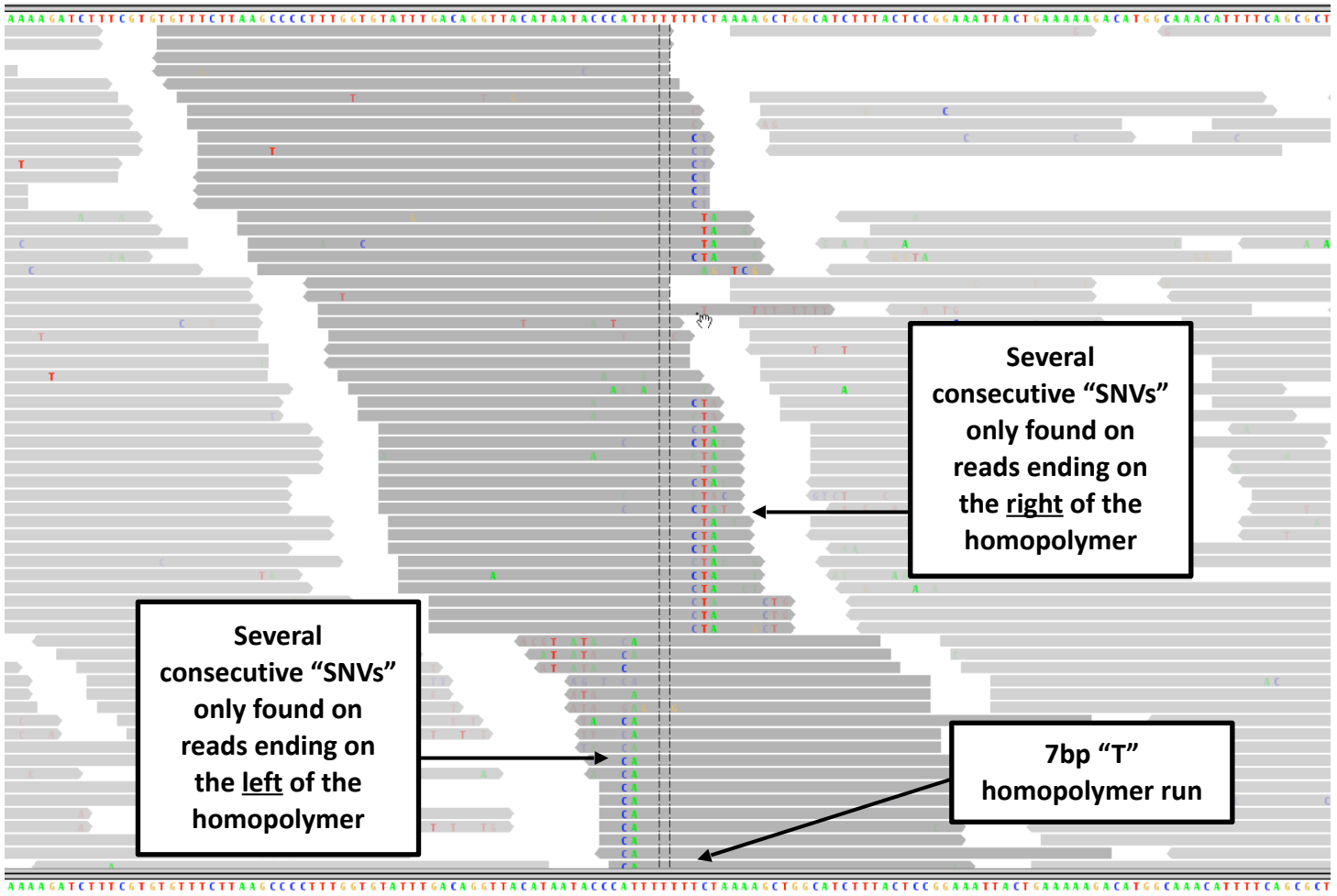
- Also, there are sometimes multiple solutions (alignments) for a given read. **Aligners can choose one randomly**.

- Reads are aligned separately (one by one).

- **Indels can be no properly identified** in the alignment of the read.

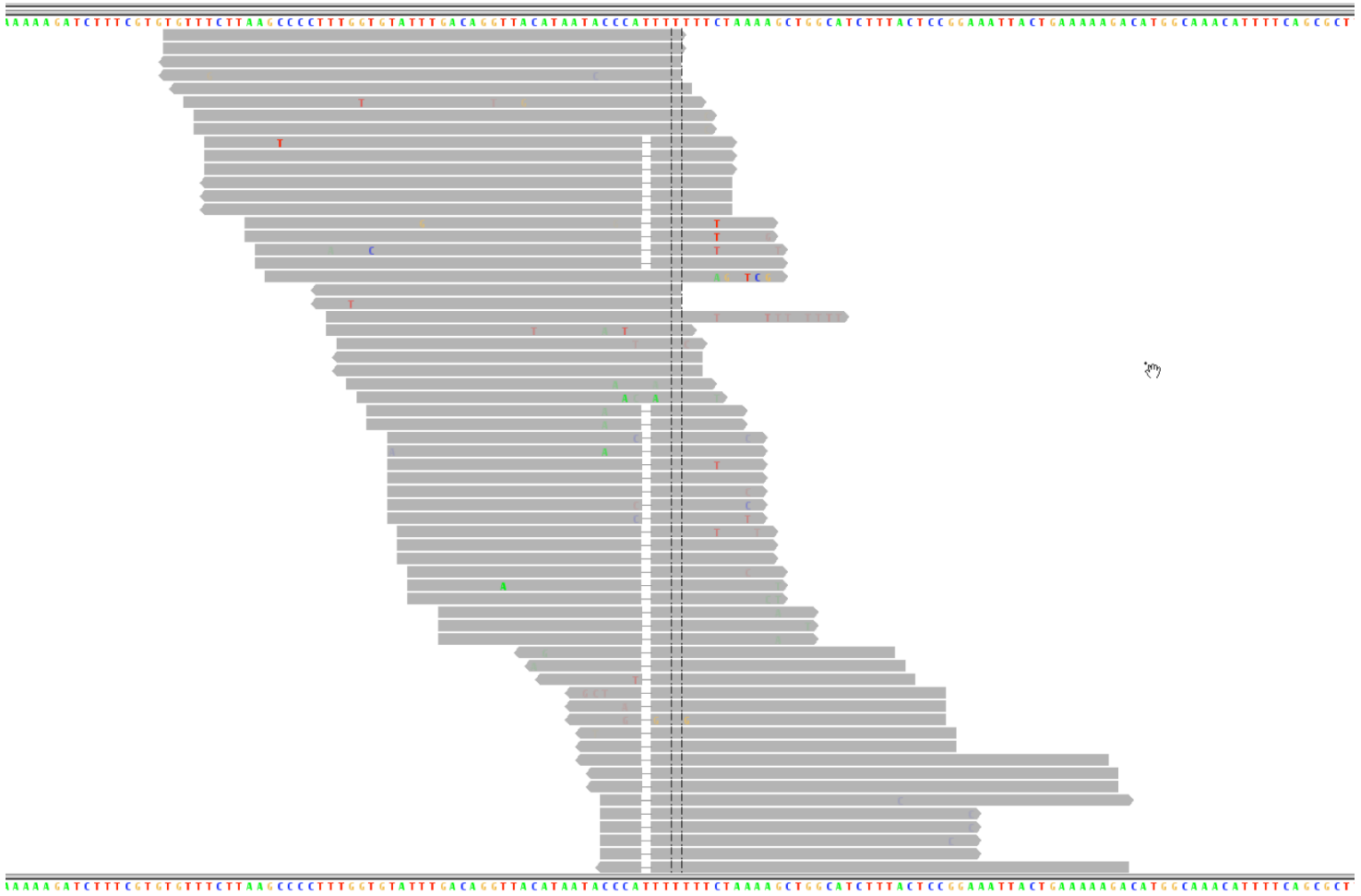
METHOD: by **GATK**
https://github.com/broadinstitute/gatk-docs/blob/master/gatk3-tutorials/(howto)_Perform_local_realignment_around_indels.md

Indel realignment



Taken from GATK team

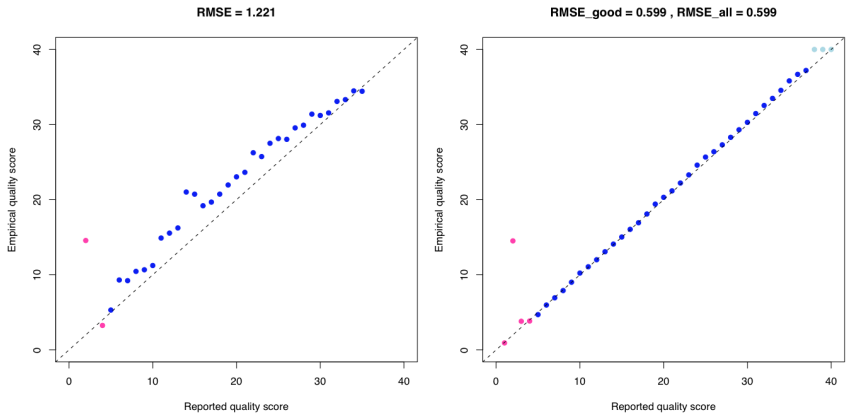
Indel realignment



Taken from GATK team

Base Quality Score Recalibration

- **Phred Quality score**: each position of the sequence has its particular **base Quality score**.
- The individual quality measures **are crucial during Variant calling**.
- Different NGS technologies have their particular **bias in Quality Score** depending on the context. Recalibration **correct empirically** these biases.

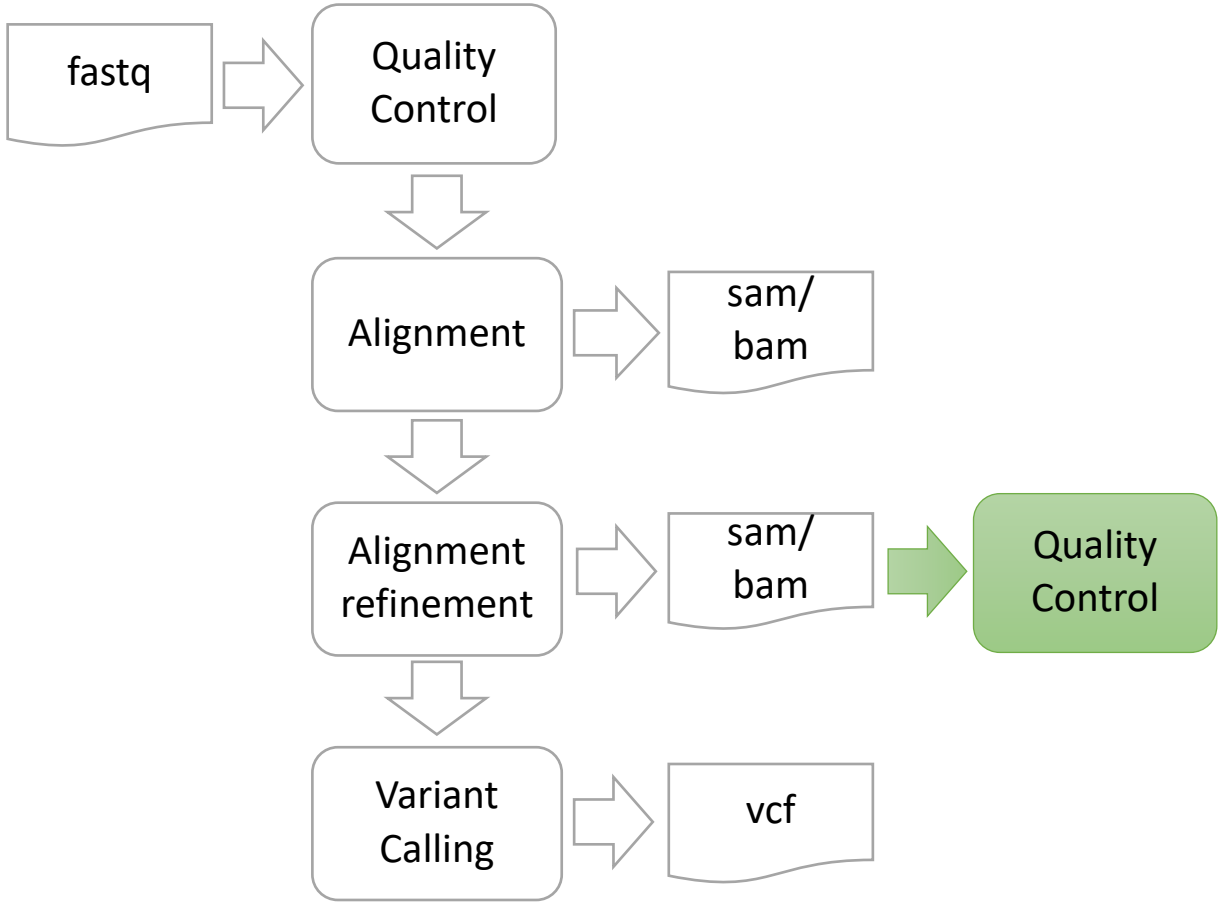


Original

After BQSR recalibration

METHOD: by **GATK**
https://gatk.broadinstitute.org/hc/en-us/articles/360035890531-Base-Quality-Score-Recalibration-BQSR-

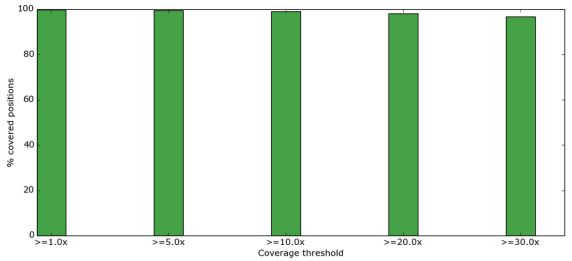
Alignment Quality Control

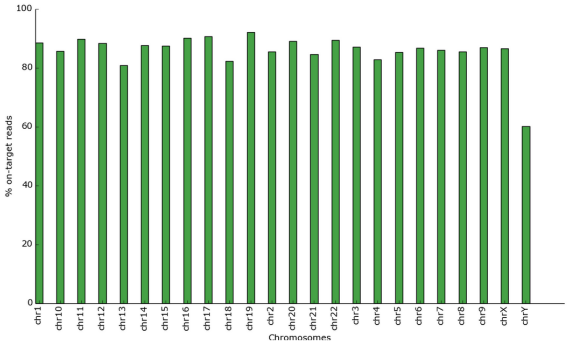


Alignment Quality Control

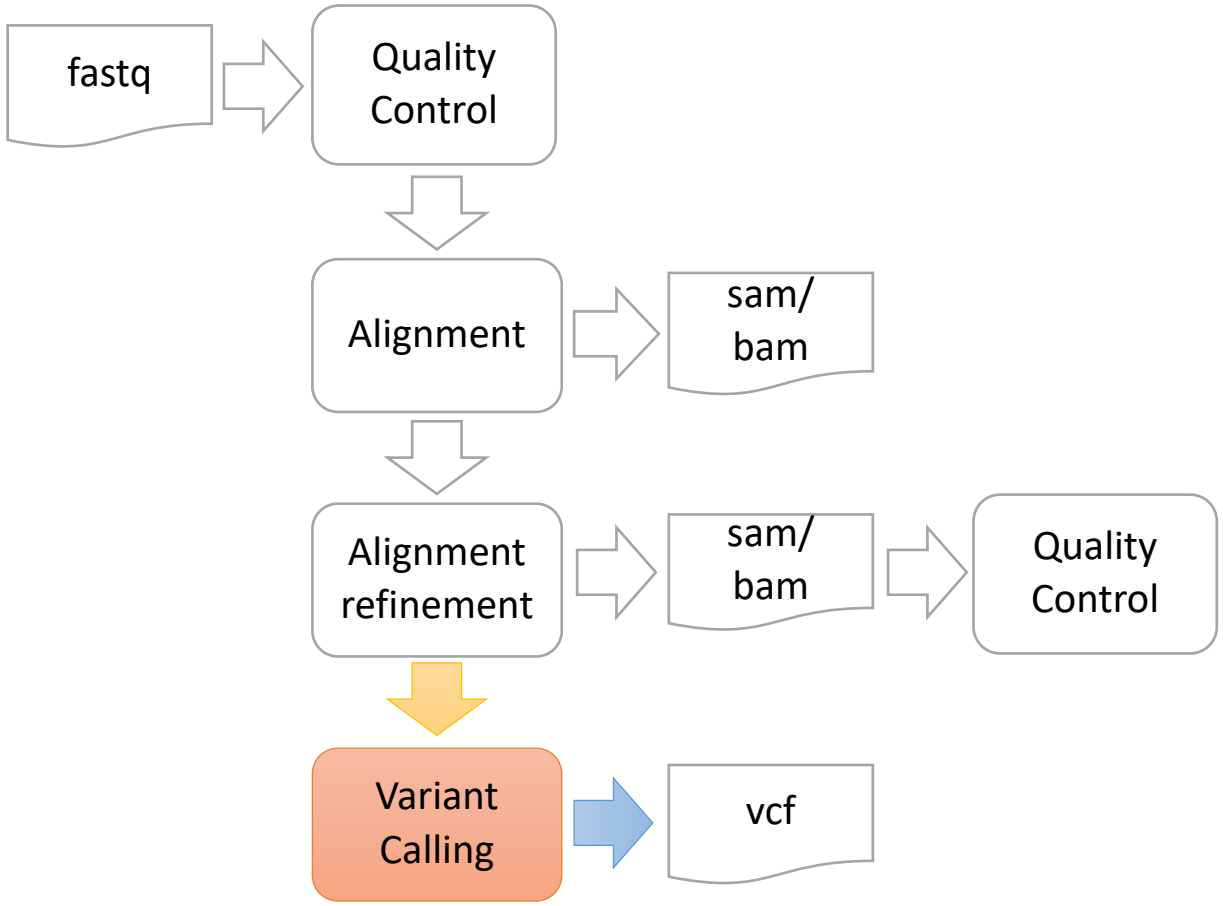
- Mean sequencing depth
- Is there enough coverage in regions of interest?
- Are the reads on-target?

- Software:
  - ngsCAT
  - QualiMap

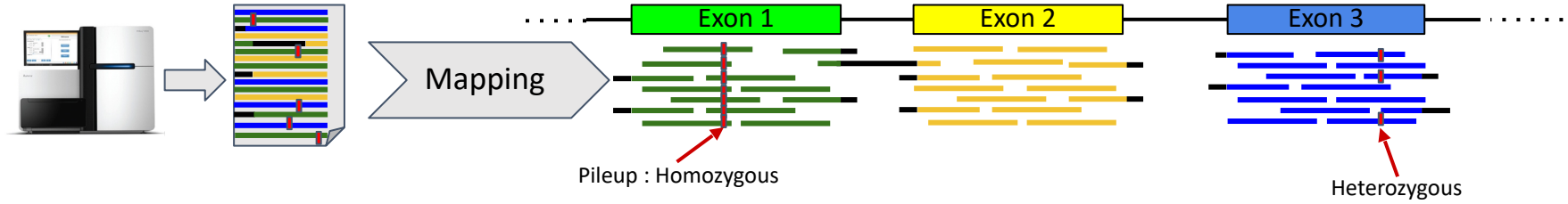




Variant calling



Fundamentals of Variant Calling



1 Identify the most likely **genotype for each genomic position** using statistical methods.

2 **Identify the differences** by comparing with the reference genome.

What is Crucial in Variant calling

- For clinical practices, the use of **gold standard methods** and **reproducible analysis** are mandatory.
- The analysis is based on the comparison against the reference genome:

A single consensus sequence for the whole genome. It was built up from a high quality set of representative samples of the species (from different populations).
It is the first-line comparison during analysis.

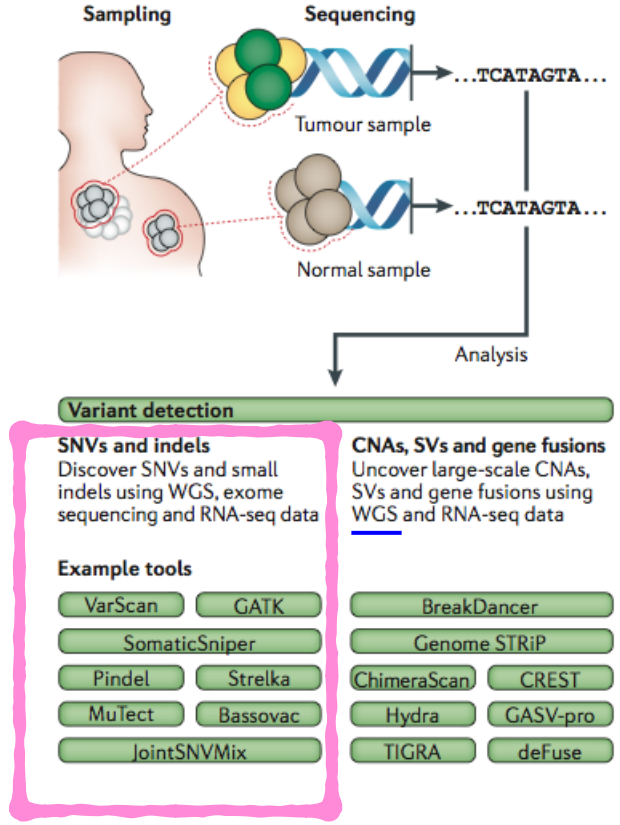
By **Genome Reference Consortium (GRC)** (http://www.ncbi.nlm.nih.gov/projects/genome/assembly/grc/)

- **Human assemblies (Versions):**
  - **GRCh37/hg19** : former version. Released in 2012. It is still used for analysis.
  - **CRCh38/hg38** : current version (Sep. 2017). Released in 2014. More accurate, comprehensive (includes Haplotypes) and sophisticated.

We must **keep consistency in the Genome Reference Version** through the variant analysis.

- We must know what **regions along the genome were sequenced** in the experiment, that is, the sequencing library.

Algorithms for Variant Calling



SNVs and Indels

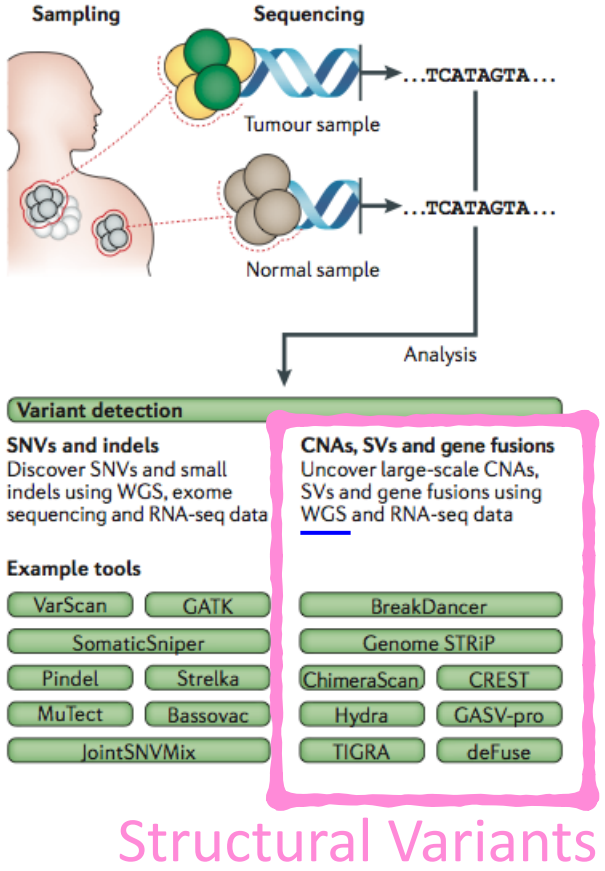
Several Methods have been published.

Further reading:

Ding, L., Wendl, M. C., McMichael, J. F. & Raphael, B. J. Nat Rev Genet – (2014). doi:10.1038/nrg3767

Nam J.N. et al. Evaluation of somatic copy number estimation tools for whole-exome sequencing data. Brief. Bioinformatics (2015)

Algorithms for Variant Calling



Structural Variants

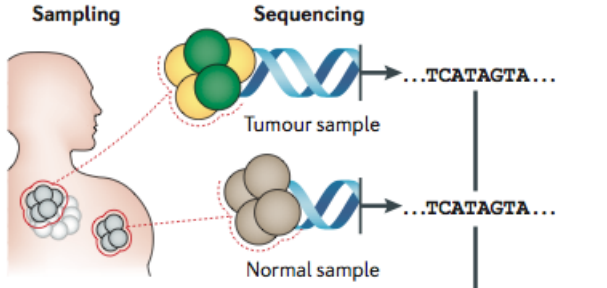
Several Methods have been published.

Further reading:

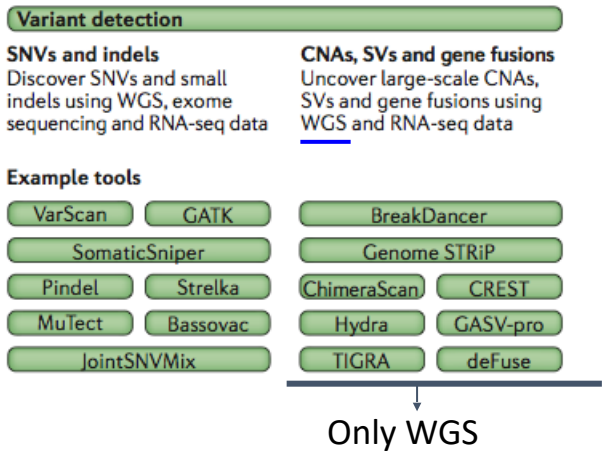
Ding, L., Wendl, M. C., McMichael, J. F. & Raphael, B. J. Nat Rev Genet – (2014). doi:10.1038/nrg3767

Nam J.N. et al. Evaluation of somatic copy number estimation tools for whole-exome sequencing data. Brief. Bioinformatics (2015)

Algorithms for Variant Calling



Several Methods have been published.



Only WGS

More on CNAs

Tool	Year	Language	Paired or pooled data	Segmentation	Feature
ADTEx	2014	Python, R	Both	HMM	Noise reduction Ploidy estimation
CONTRA	2012	Python, R	Both	CBS	GC correction
Control-FREEC	2011	C++, R	Paired	LASSO	GC correction, mappability
EXCAVATOR	2013	Perl, R	Both	HSLM	GC correction, mappability, exon-size correction
ExomeCNV	2011	R	Paired	CBS	GC correction, mappability
Varscan2	2012	Java, Perl, R	Paired	CBS	GC correction

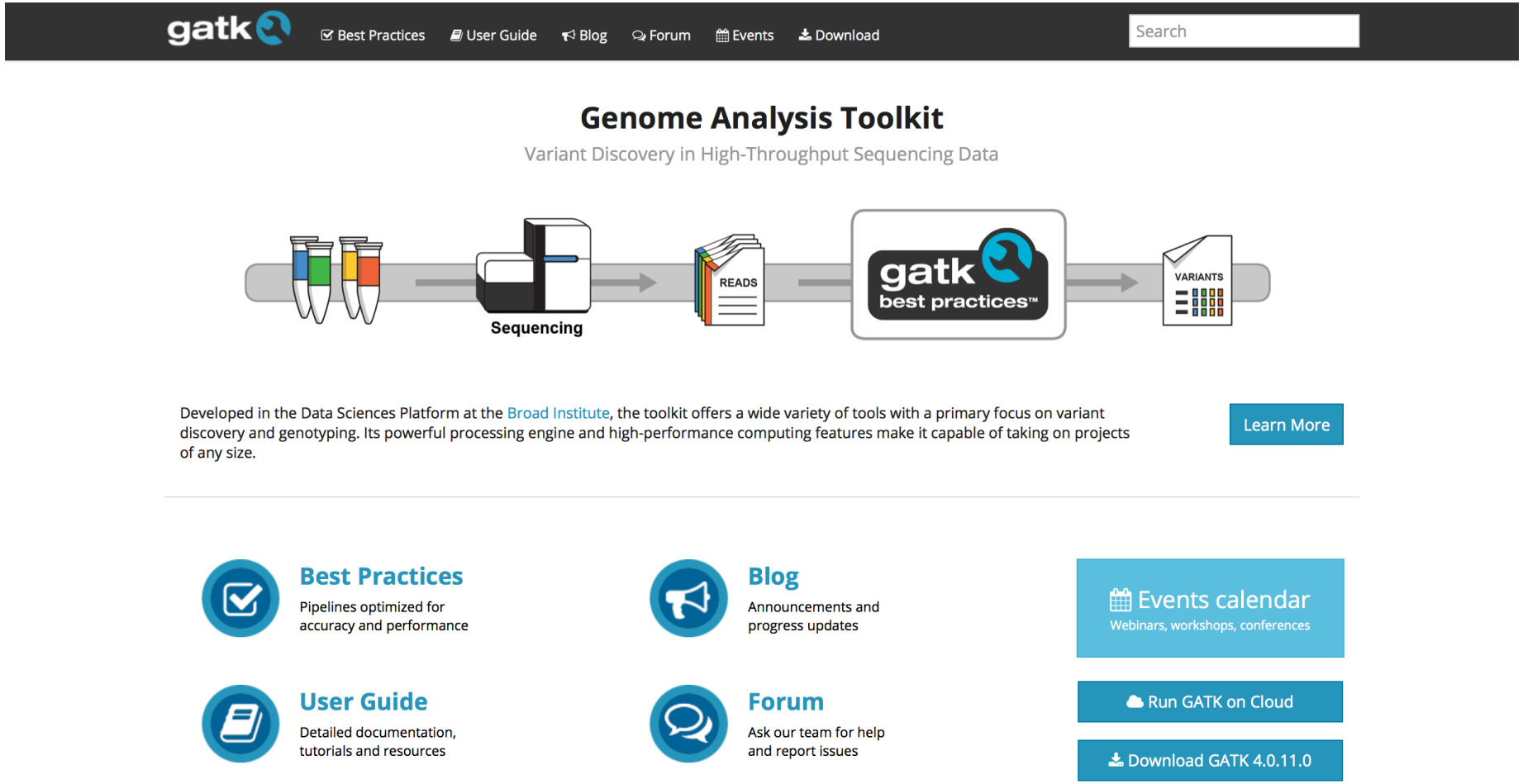
Appropriate methods for Whole-Exome seq

Further reading:

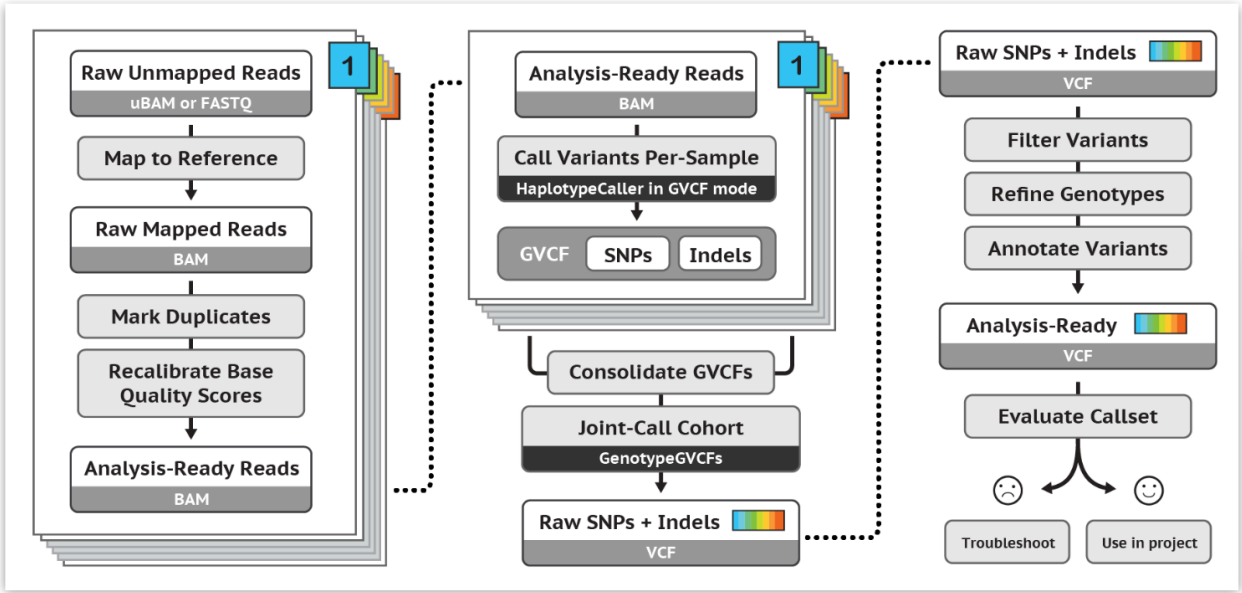
Ding, L., Wendl, M. C., McMichael, J. F. & Raphael, B. J. Nat Rev Genet – (2014). doi:10.1038/nrg3767

Nam J.N. et al. Evaluation of somatic copy number estimation tools for whole-exome sequencing data. Brief. Bioinformatics (2015)

GATK for variant calling analysis

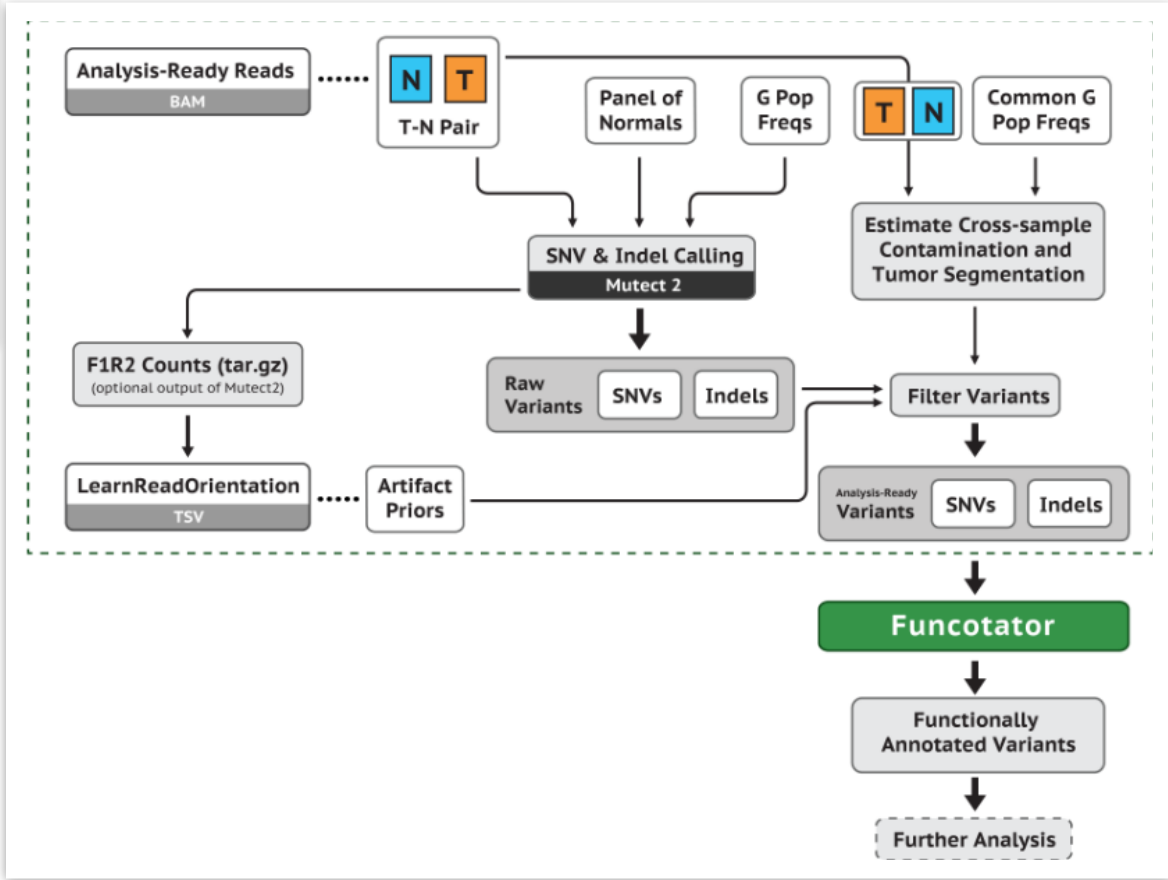


GATK Best Practices



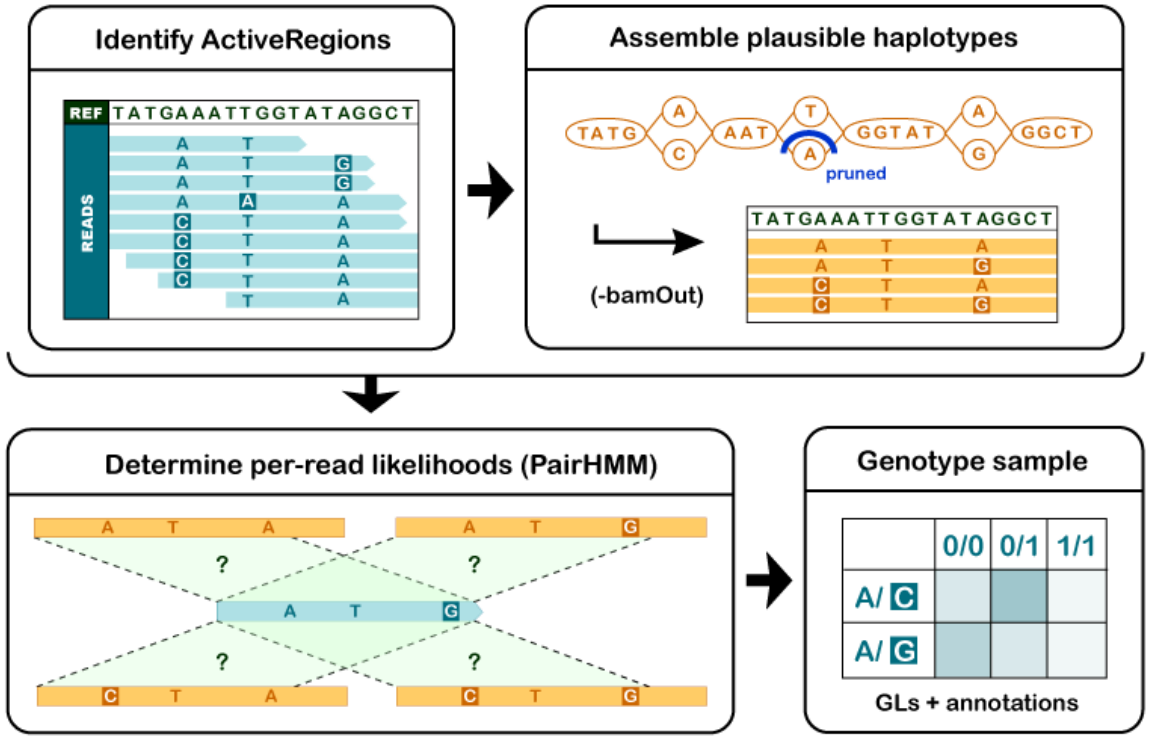
Germline small-scale variants

Somatic small-scale variants



Variant Calling for SNVs and Indels

Haplotype Caller : Variant calling based on the calculation of genotype likelihoods:



Assumptions: It bases the calling in the indicated ploidy (e.g. 2n)

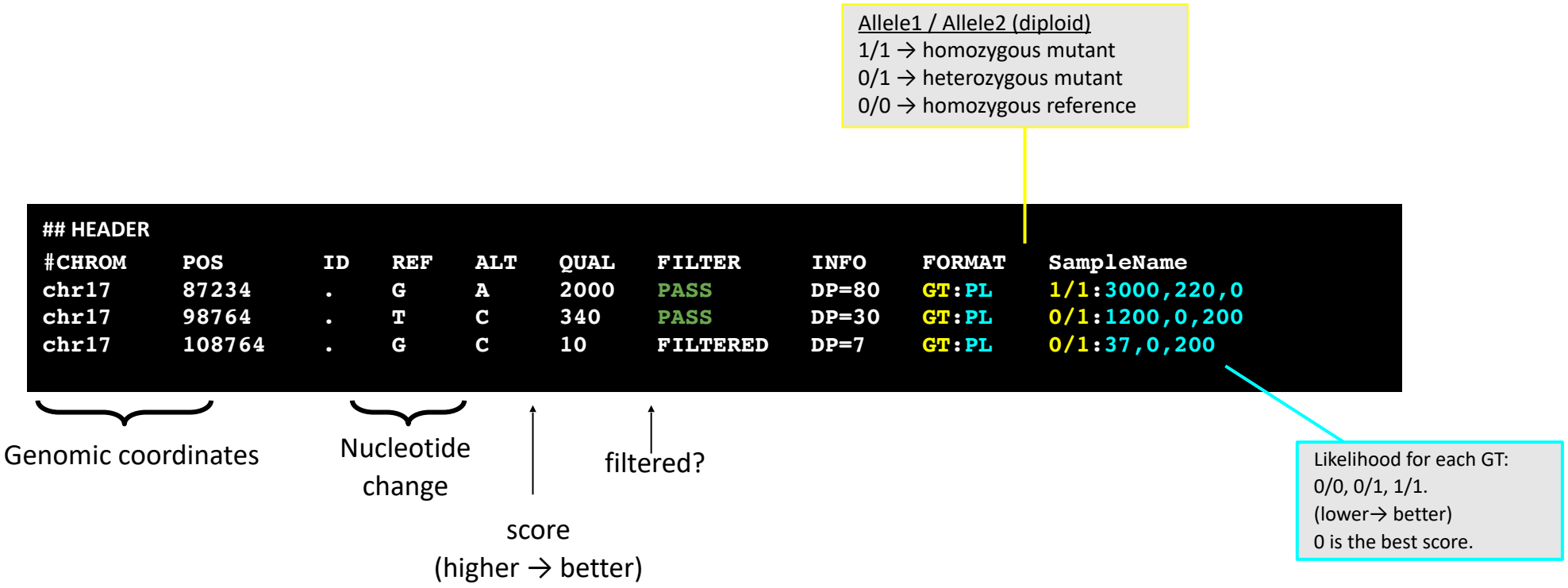
Limited detection of low allele frequencies.

Further reading:

https://github.com/broadgsa/gatk/blob/master/doc_archive/methods/HC_overview:_How_the_HaplotypeCaller_works.md

https://gatk.broadinstitute.org/hc/en-us/sections/360007226771?name=methods

VCF file



More info.:

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

https://gatk.broadinstitute.org/hc/en-us/articles/360035531692-VCF-Variant-Call-Format

Variant Calling for somatic variants: MuTect2

SNV and Indel caller**.**

Similar logic to Haplotype Caller but:

- It **allows variable allele frequencies**.
- It **includes logic to avoid germline variants**.

CGA
Cancer Genome Analysis

Log-in

Search this site:
Search...

CGA
- Home
- Help Forum
- ABSOLUTE
- BreakPointer
- ChainFinder
- ContEst
- Copynumber Pipeline
- D-ToxoG
- dRanger
- Firehose
- GISTIC
- HAPSEG
- Indelocator
- InVEx
- MuTect
  - How do I get MuTect?
  - How do I run MuTect?
  - How do I get help?
- MutSig
- Oncotator
- PathSeq
- RNA-SeQC
- SegSeq
- TranspoSeq

BROAD INSTITUTE
© 2013 Broad Institute

For site help, please contact cga-webmaster@broadinstitute.org

Home ›

MuTect

What is MuTect?

MuTect is a method developed at the Broad Institute for the reliable and accurate identification of somatic point mutations in next generation sequencing data of cancer genomes.

For complete details, please see our publication in Nature Biotechnology:

Cibulskis, K. et al. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. Nat Biotechnology (2013).doi:10.1038/nbt.2514

How does it work?

In brief, muTect consists of three steps.

1. Preprocessing the aligned reads in the tumor and normal sequencing data. In this step we ignore reads with too many mismatches or very low quality scores since these represent noisy reads that introduce more noise than signal.
2. A statistical analysis that identifies sites that are likely to carry somatic mutations with high confidence. The statistical analysis predicts a somatic mutation by using two Bayesian classifiers – the first aims to detect whether the tumor is non-reference at a given site and, for those sites that are found as non-reference, the second classifier makes sure the normal does not carry the variant allele. In practice the classification is performed by calculating a LOD score (log odds) and comparing it to a cutoff determined by the log ratio of prior probabilities of the considered events. For the tumors we calculate $LOD_T = \log_{10}\left(\frac{P(\text{observed data in tumor} \mid \text{site is mutated})}{P(\text{observed data in tumor} \mid \text{site is reference})}\right)$, and for the normal $LOD_N = \log_{10}\left(\frac{P(\text{observed data in normal} \mid \text{site is reference})}{P(\text{observed data in normal} \mid \text{site is mutated})}\right)$.

   Since we expect somatic mutations to occur at a rate of ~1 in a Mb, we require $LOD_T > \log_{10}\left(0.5 \times 10^{-6}\right) = 6.3$ which guarantees that our false positive rate, due to noise in the tumor, is less than half of the somatic mutation rate. In the normal, not in dbSNP sites, we require $LOD_N > \log_{10}\left(0.5 \times 10^{-2}\right) = 2.3$ since non-dbSNP germline variants occur roughly at a rate of 100 in a Mb. This cutoff guarantees that the false positive somatic call rate, due to missing the variant in the normal, is also less than half the somatic mutation rate.

3. Post-processing of candidate somatic mutations to eliminate artifacts of next-generation sequencing, short read alignment and hybrid capture. For example, sequence context can cause hallucinated alternate alleles but often only in a single direction. Therefore, we test that the alternate alleles supporting the mutations are observed in both directions.

As muTect attempts to call mutations it also generates a coverage file (in a wiggle file format, which indicates for every base whether it is sufficiently covered in the tumor and normal to be sensitive enough to call mutations). We currently use cutoffs of at least 14 reads in the tumor and at least 8 in the normal (these cutoffs are applied after removing noisy reads in the preprocessing step). In addition, wiggle files can also be generated of the observed depth in the tumor and in the normal.

Most cancer genome studies at the Broad Institute have made use of MuTect and have validated the mutation calls as a part of their cancer biology papers, showing that MuTect has a very low false positive rate. A summary of validation rates from these papers are show below:

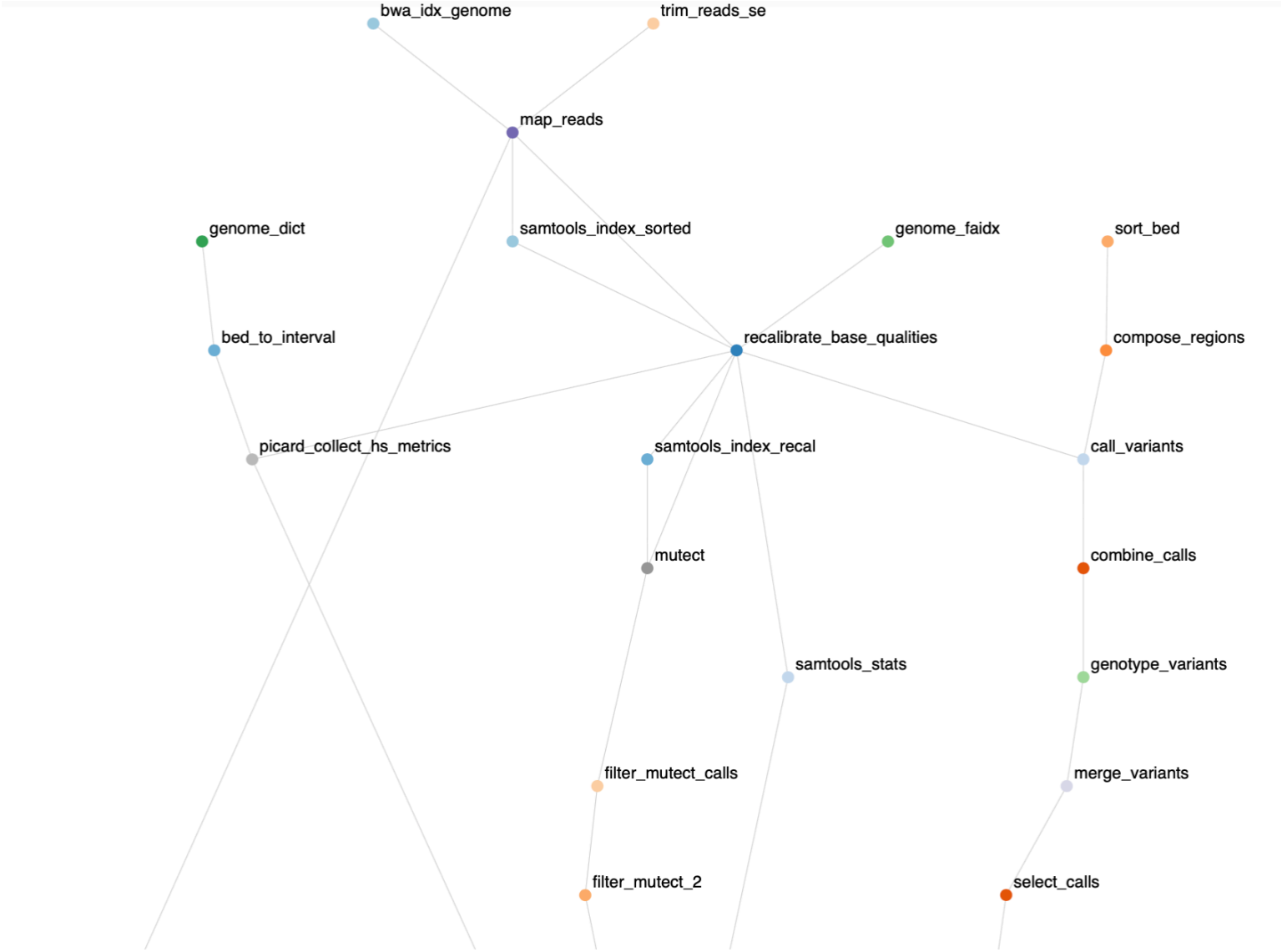
publication	technology	candidates	validated	no result	validation rate
Multiple Myeloma[1]	Sequenom	97	92	5	94.85%
Ovarian[2]	Sequenom/PCR/454	1655	1483	172	89.61%
Ovarian[2]	Capture/Illumina	6497	6232	265	95.92%
Head and Neck[3]	Sequenom	321	288	33	89.72%
Breast[4]	Sequenom/PCR/454	455	428	0	94.07%

https://gatk.broadinstitute.org/hc/en-us/articles/360037593851-Mutect2

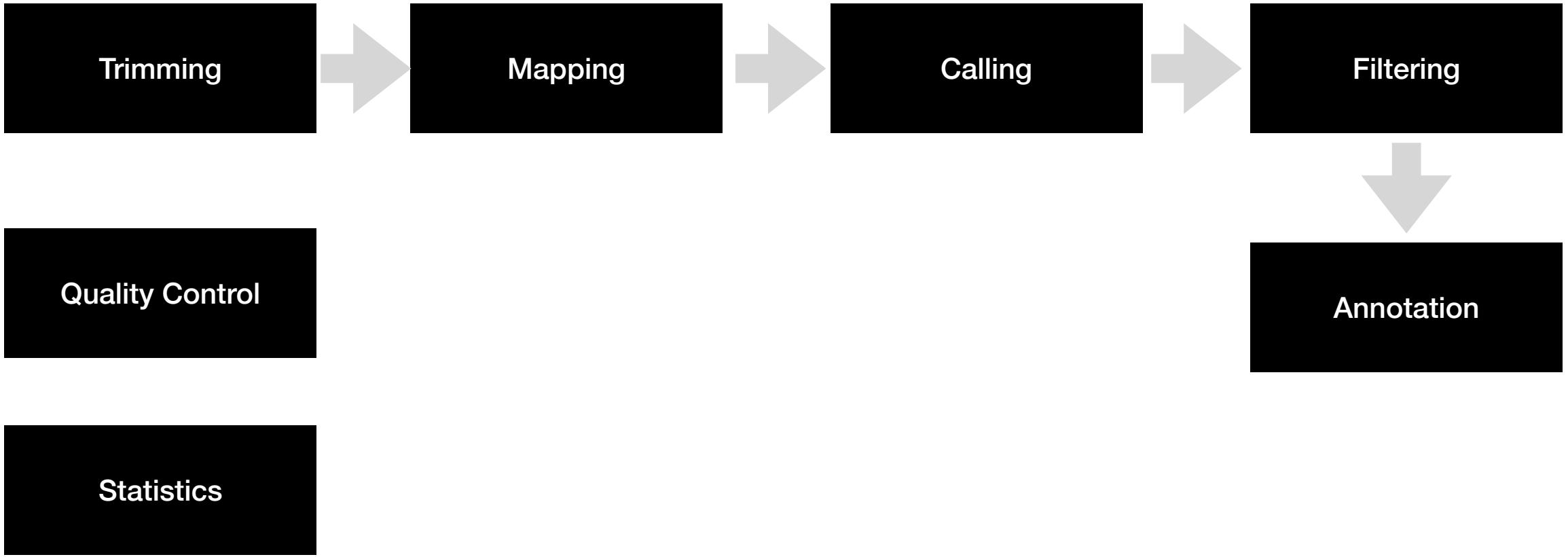
Cibulskis, K. et al.
Nat Biotechnology (2013).doi:10.1038/nbt.2514

varca

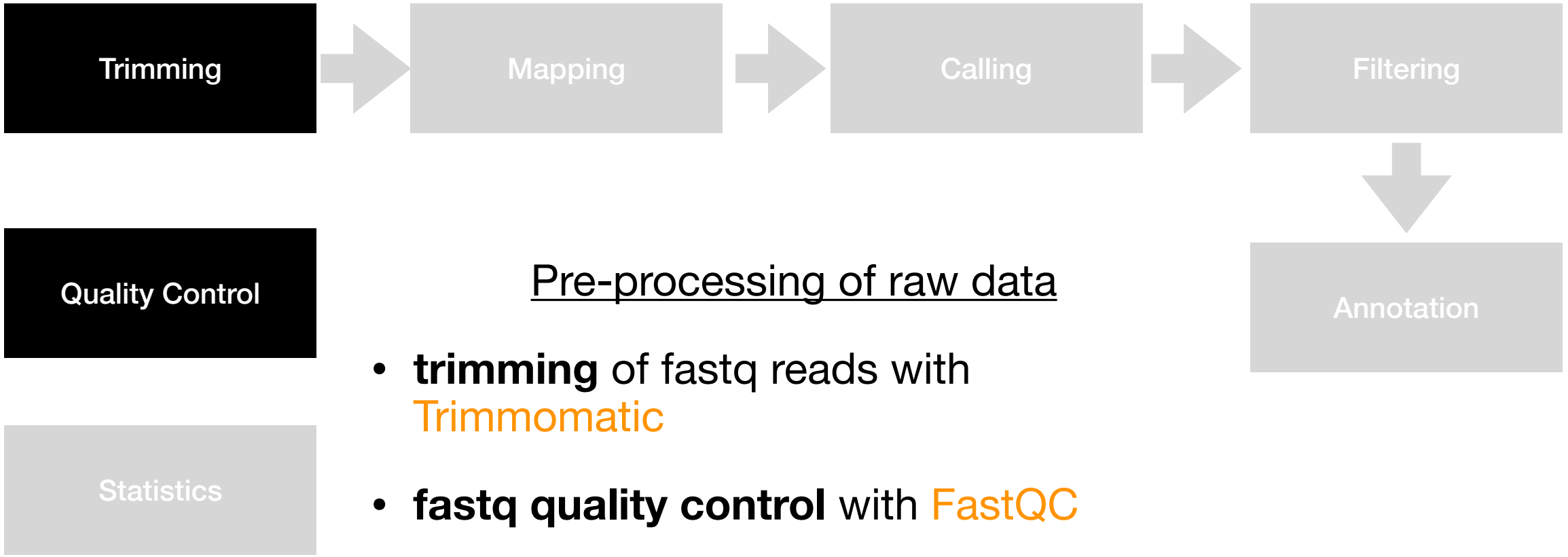
https://gitlab.com/bu_cnio/varca



Steps in the pipeline



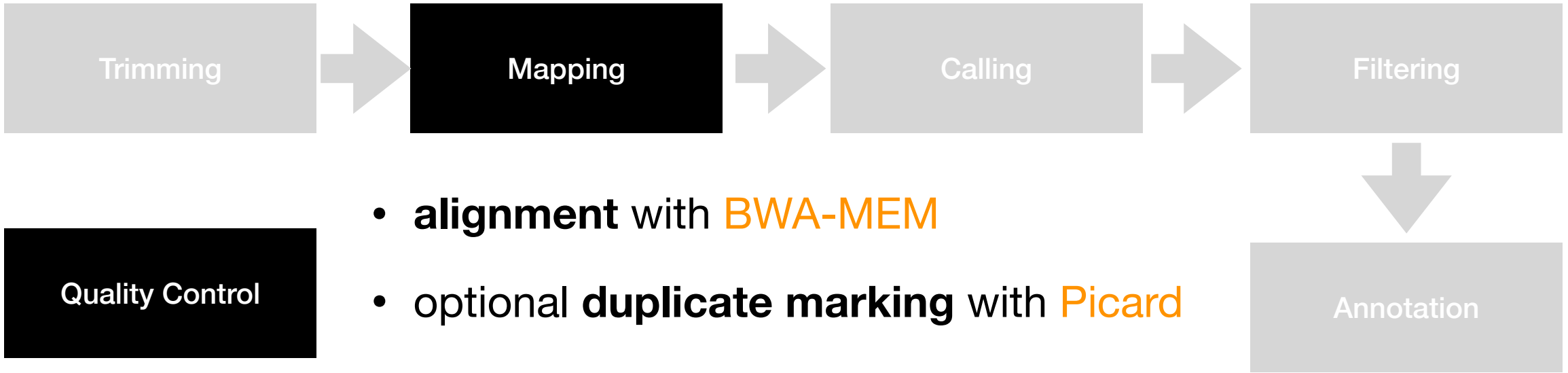
Steps in the pipeline



Pre-processing of raw data

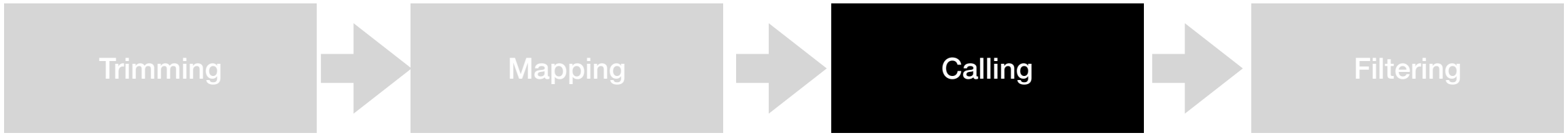
- **trimming** of fastq reads with Trimmomatic
- **fastq quality control** with FastQC

Steps in the pipeline



- **alignment** with BWA-MEM
- optional **duplicate marking** with Picard
- **base quality score recalibration** with GATK
- **collection of metrics from bam** with samtools and Picard

Steps in the pipeline

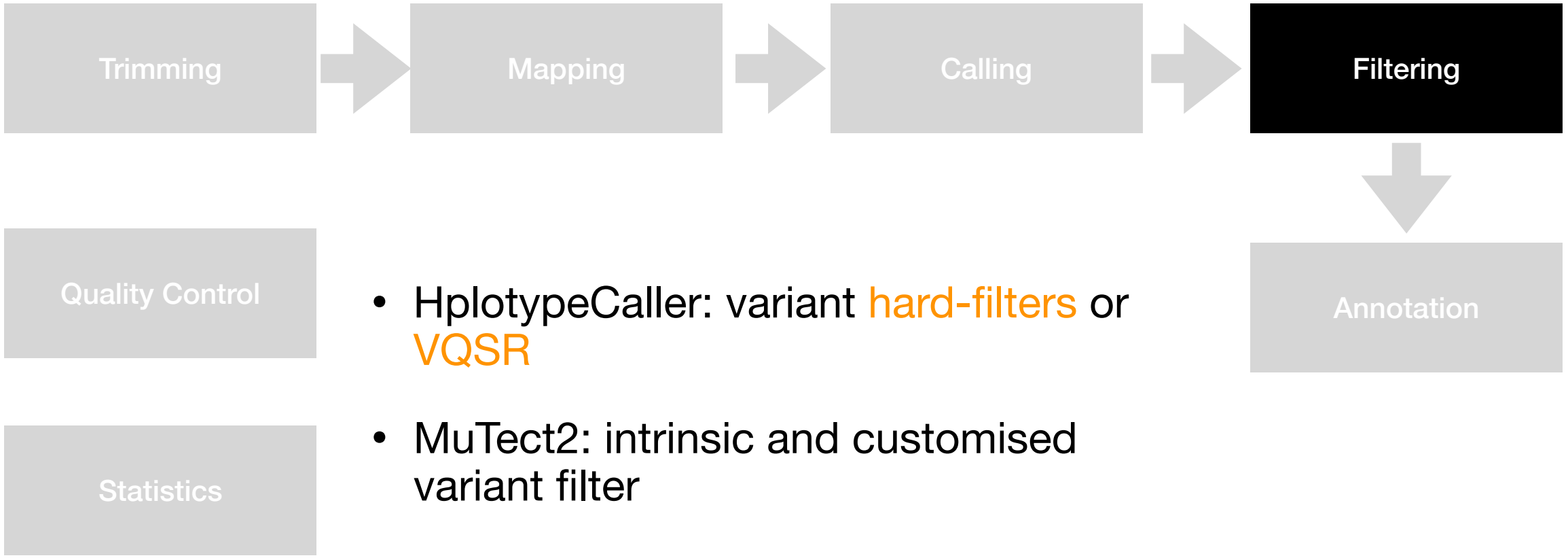


Quality Control

Statistics

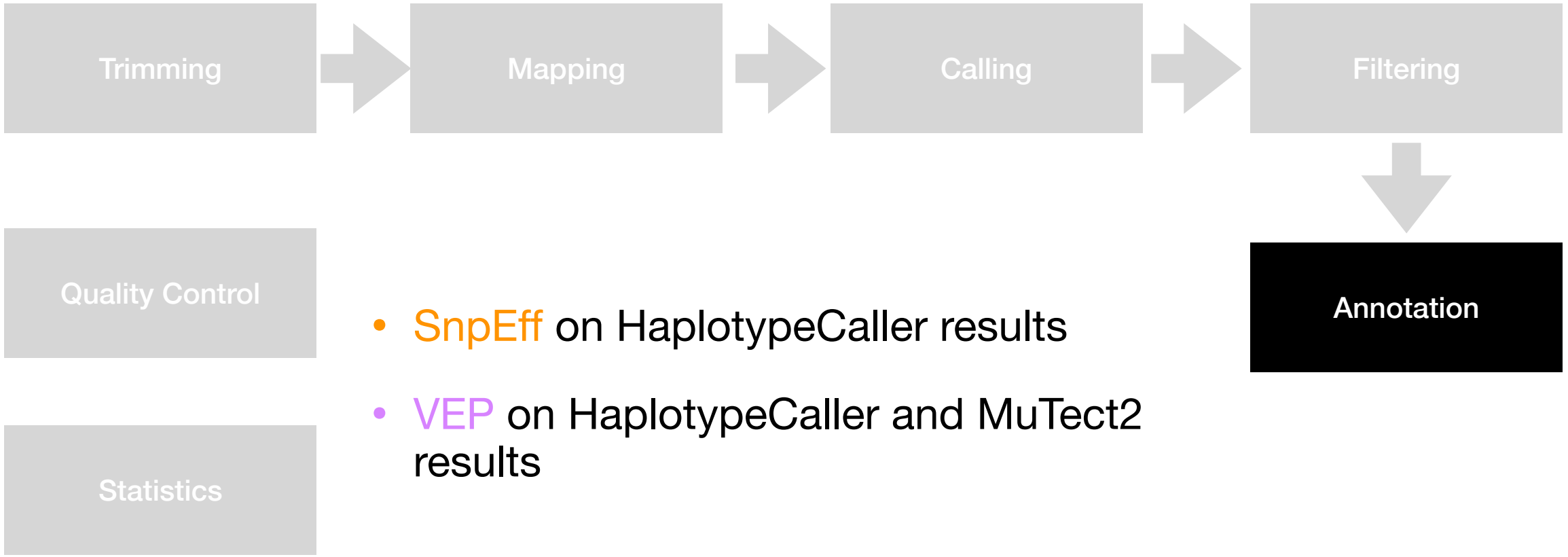
- Germline and medium-to-high allele frequency somatic variants with HaplotypeCaller (GATK)
- Optional extension to somatic low-frequency variants with MuTect2

Steps in the pipeline



- HplotypeCaller: variant hard-filters or VQSR
- MuTect2: intrinsic and customised variant filter

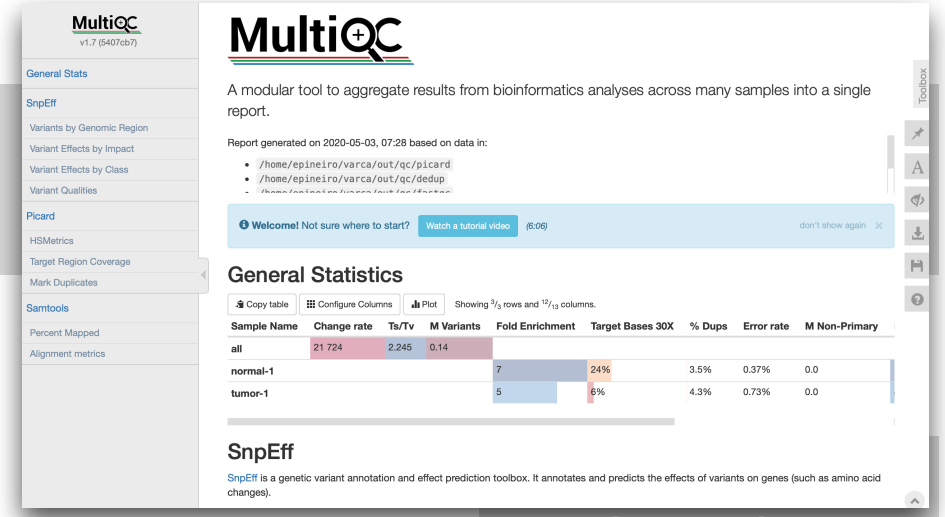
Steps in the pipeline



- SnpEff on HaplotypeCaller results
- VEP on HaplotypeCaller and MuTect2 results

Steps in the pipeline

Trimming → Mapping → Annotation



Quality Control

- Quality report with MultiQC

Statistics

- Statistic Plots: allele frequencies and depth of sequencing

Configuration files

2 configuration files with the **samples definition**

1 configuration file with the **contigs** to analyse

1 configuration file with **paths** to additional files used in the process **and parameters**

Configuration files

samples.tsv and units.tsv

```
1  sample  control
2  A       B
3  B       -
```

unable MuTect2 calling on the sample

samples

list of samples (sample column)

establishes the MuTect2 calling mode (control column)

- No MuTect2 execution:

sample	control
A	-

- MuTect2 execution in tumor-only mode:

sample	control
A	A

- MuTect2 execution in tumor-normal mode:

sample	control
A	B
B	-

being A:tumor, B:normal

Configuration files

samples.tsv and units.tsv

sample	control
A	B
B	-

unable MuTect2 calling on the sample

samples

list of samples (sample column)

establishes the MuTect2 calling mode (control column)

paired-end and single-end definition

sample	unit	platform	fq1	fq2
A	1	ILLUMINA	data/reads/a.chr21.1.fq	data/reads/a.chr21.2.fq
B	1	ILLUMINA	data/reads/b.chr21.1.fq	data/reads/b.chr21.2.fq
B	2	ILLUMINA	data/reads/b.chr21.1.fq	

units

sequential to identify multiple sequences for the same sample

Configuration files

contigs.tsv

```
chr1
chr2
chr3
chr4
chr5
chr6
chr7
chr8
chr9
chr10
chr11
chr12
chr13
chr14
chr15
chr16
chr17
chr18
chr19
chr20
chr21
chr22
chrX
chrY
chrM
```

~3300 contigs in hg38

```
@SQ     SN:chr1_KI270708v1_random       LN:127682       M5:1e95e047b98ed92148dd84d6c037158c
@SQ     SN:chr1_KI270709v1_random       LN:66860        M5:4e2db2933ea96aee8dab54af60ecb37d
@SQ     SN:chr1_KI270710v1_random       LN:40176        M5:9949f776680c6214512ee738ac5da289
@SQ     SN:chr1_KI270711v1_random       LN:42210        M5:af383f98cf4492c1f1c4e750c26cbb40
@SQ     SN:chr1_KI270712v1_random       LN:176043       M5:c38a0fecae6a1838a405406f724d6838
@SQ     SN:chr1_KI270713v1_random       LN:40745        M5:cb78d48cc0adbc58822a1c6fe89e3569
@SQ     SN:chr1_KI270714v1_random       LN:41717        M5:42f7a452b8b769d051ad738ee9f00631
@SQ     SN:chr2_KI270715v1_random       LN:161471       M5:b65a8af1d7bbb7f3c77eea85423452bb
@SQ     SN:chr2_KI270716v1_random       LN:153799       M5:2828e63b8edc5e845bf48e75fbad2926
@SQ     SN:chr3_GL000221v1_random       LN:155397       M5:3238fb74ea87ae857f9c7508d315babb
@SQ     SN:chr4_GL000008v2_random       LN:209709       M5:a999388c587908f80406444cebe80ba3
@SQ     SN:chr5_GL000208v1_random       LN:92689        M5:aa81be49bf3fe63a79bdc6a6f279abf6
@SQ     SN:chr9_KI270717v1_random       LN:40062        M5:796773a1ee67c988b4de887addbed9e7
@SQ     SN:chr9_KI270718v1_random       LN:38054        M5:b0c463c8efa8d64442b48e936368dad5
@SQ     SN:chr9_KI270719v1_random       LN:176845       M5:cd5e932cfc4c74d05bb64e2126873a3a
@SQ     SN:chr9_KI270720v1_random       LN:39050        M5:8c2683400a4aeeb40abff96652b9b127
@SQ     SN:chr11_KI270721v1_random      LN:100316       M5:9654b5d3f36845bb9d19a6dbd15d2f22
@SQ     SN:chr14_GL000009v2_random      LN:201709       M5:862f555045546733591ff7ab15bcecbe
@SQ     SN:chr14_GL000225v1_random      LN:211173       M5:63945c3e6962f28ffd469719a747e73c
@SQ     SN:chr14_KI270722v1_random      LN:194050       M5:51f46c9093929e6edc3b4dfd50d803fc
@SQ     SN:chr14_GL000194v1_random      LN:191469       M5:6ac8f815bf8e845bb3031b73f812c012
@SQ     SN:chr14_KI270723v1_random      LN:38115        M5:74a4b480675592095fb0c577c515b5df
```

Configuration files

config.yaml

```yaml
samples: samples.tsv
units: units.tsv
contigs: contigs.tsv

outdir: "out"
logdir: "log"

ref:
  # Genome database of snpeff to be used in the annotation with this resource. Available databases can be checked with java -ja
  name: GRCh38.86
  # Path to the reference genome, ideally as it is provided by the GATK bundle.
  genome: /home/epineiro/REFERENCES/Homo_sapiens/HG38/hg38.fa
  # Path to any database of known variants, ideally as it is provided by the GATK bundle.
  known-variants: /home/epineiro/REFERENCES/Homo_sapiens/HG38/dbsnp_151.hg38.vcf

filtering:
  # Set to true in order to apply machine learning based recalibration of
  # quality scores instead of hard filtering.
  vqsr: false
  hard:
    # hard filtering as outlined in GATK docs
    # (https://gatkforums.broadinstitute.org/gatk/discussion/2806/howto-apply-hard-filters-to-a-call-set)
    snvs:
      "QD < 2.0 || QUAL < 100.0 || DP < 50.0 || SOR > 3.0 || FS > 60.0 || MQ < 40.0 || MQRankSum < -12.5 || ReadPosRankSum < -8
    indels:
      "QD < 2.0 || QUAL < 100 || DP < 10.0 || FS > 200.0 || ReadPosRankSum < -20.0"
  #depth of coverage threshold to apply to variants identified with MuTect2
  depth: "DP < 30"
```

Reference Genome and known variants file

Variant filtering system and parameters for hard filtering

Configuration files

config.yaml

```
processing:
  remove-duplicates: true
  # see https://gatkforums.broadinstitute.org/gatk/discussion/4133/when-should-i-use-l-to-pass-in-a-list-of-intervals.
  restrict-regions: /home/epineiro/REFERENCES/Homo_sapiens/HG38/S04380219_Padded_v5_hg38_varca.bed
  # If regions are restricted, uncomment this to enlarge them by the given value in order to include
  # flanking areas.
  # region-padding: 100

params:
  gatk:
    HaplotypeCaller: ""
    BaseRecalibrator: ""
    GenotypeGVCFs: ""
    #If vqsr set to true, fill the following section with the required information (see VariantRecalibrator documentation in GA
    VariantRecalibrator:
      #paths to the necessary resources
      hapmap: "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_hapmap_3.3.hg38.vcf.gz"
      omni: "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_1000G_omni2.5.hg38.vcf.gz"
      g1k: "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_1000G_phase1.snps.high_confidence.hg38.vcf.gz"
      dbsnp: "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_Homo_sapiens_assembly38.dbsnp138.vcf"
      #paths to the resource indexes
      aux: ["/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_hapmap_3.3.hg38.vcf.gz.tbi",
                  "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_1000G_omni2.5.hg38.vcf.gz.tbi",
                              "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_1000G_phase1.snps.high_conf
                                          "/home/epineiro/REFERENCES/Homo_sapiens/HG38/resources_broad_hg38_v0_Homo_sapiens_a
      #set definition for each resource
      parameters: {"hapmap": {"known": False, "training": True, "truth": True, "prior": 15.0},
                   "omni":   {"known": False, "training": True, "truth": False, "prior": 12.0},
                   "g1k":    {"known": False, "training": True, "truth": False, "prior": 10.0},
                   "dbsnp":  {"known": True, "training": False, "truth": False, "prior": 2.0}}
      #Names of the annotations used for calculations
```

Remove duplicates in the alignment?

Restrict variants to specific regions?

Configuration parameters for trimming, deduplication and GATK variant calling, vqsr

Configuration files

config.yaml

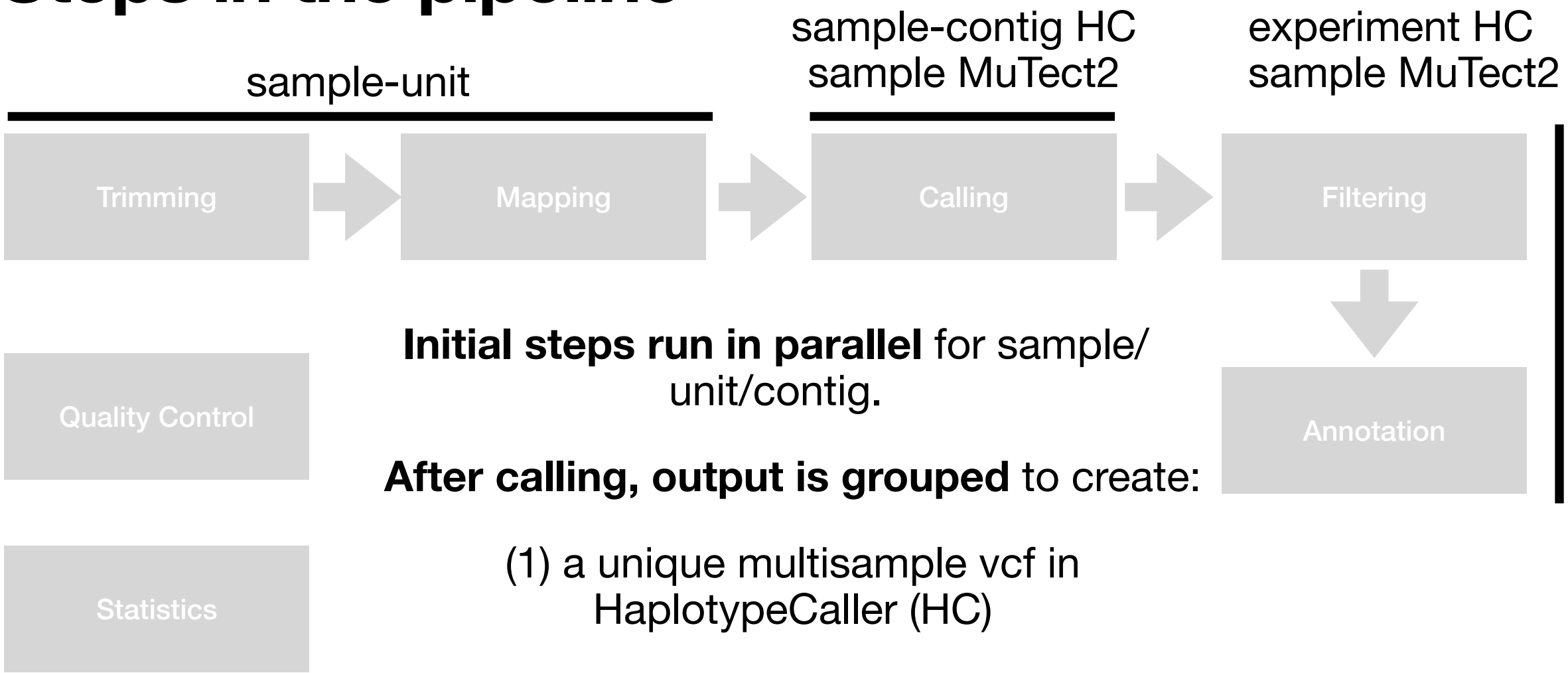
Parameters for the annotation with VEP

```
annotation:
  vep:
    # If set to true, a cache directory must be created using the instructions of VEP. Using a cache allows for a fastest and
    cache: true
    #Indicate the version of the cache if cache is set to true. e.g. 100. This version must match the one indicated in the vep.
    cache_version: 100
    #Indicate the path to the cache files if cache is set to true
    cache_directory: /home/epineiro/Programs/VEP/VEP100/.vep/
    #Indicate the assembly version of the reference genome
    assembly: GRCh38
    #Indicate the annotations to include with the VEP execution and additional parameters. See the documentation (https://www.
    annotations: "--sift b --polyphen b --ccds --uniprot --hgvs --symbol --numbers --domains --regulatory --canonical --protei
```

Machine requirements

```
resources:
    default:
        threads: 1
        mem: 32000
        walltime: 720
    snpeff:
        mem: 8000
        walltime: 480
    call_variants:
        mem: 10000
        threads: 4
        walltime: 480
    combine_calls:
        mem: 8000
        threads: 1
        walltime: 480
    recalibrate_calls:
        mem: 8000
        walltime: 480
```

Steps in the pipeline



Initial steps run in parallel for sample/unit/contig.

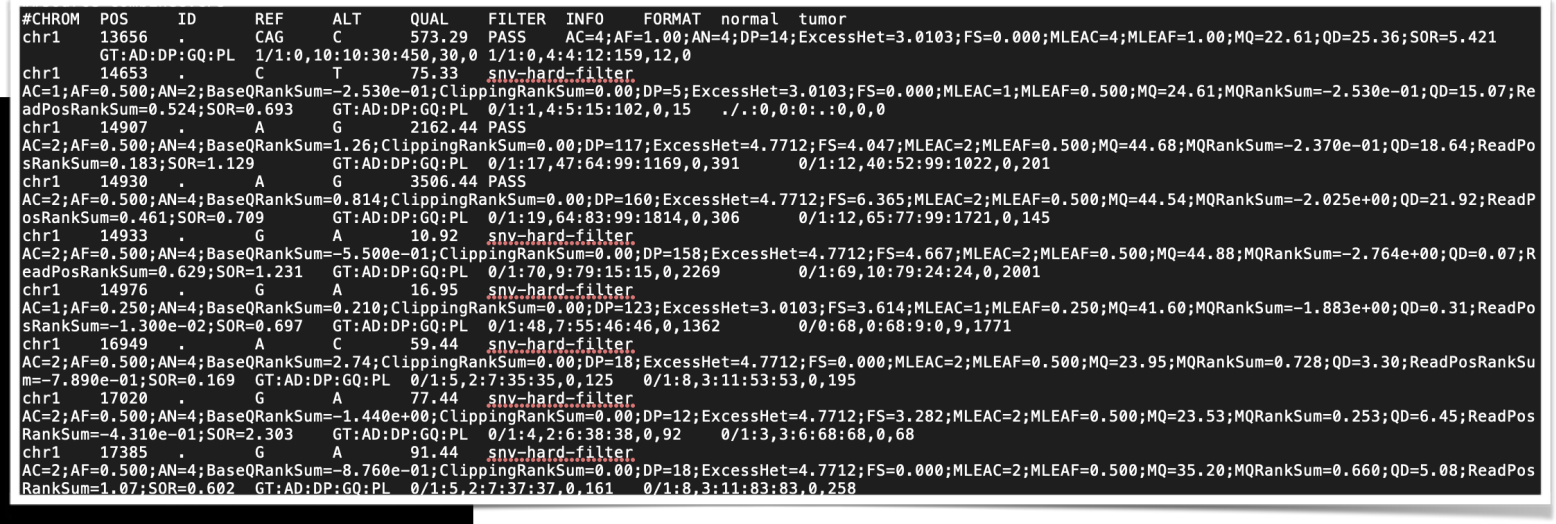
After calling, output is grouped to create:

(1) a unique multisample vcf in HaplotypeCaller (HC)

(2) a vcf file for each tumor sample in MuTect2

Result's files

VCF multisample



```
#CHROM  POS     ID      REF     ALT     QUAL    FILTER  INFO    FORMAT  normal  tumor
chr1    13656   .       CAG     C       573.29  PASS    AC=4;AF=1.00;AN=4;DP=14;ExcessHet=3.0103;FS=0.000;MLEAC=4;MLEAF=1.00;MQ=22.61;QD=25.36;SOR=5.421
        GT:AD:DP:GQ:PL  1/1:0,10:10:30:450,30,0 1/1:0,4:4:12:159,12,0
chr1    14653   .       C       T       75.33   snv-hard-filter
AC=1;AF=0.500;AN=2;BaseQRankSum=-2.530e-01;ClippingRankSum=0.00;DP=5;ExcessHet=3.0103;FS=0.000;MLEAC=1;MLEAF=0.500;MQ=24.61;MQRankSum=-2.530e-01;QD=15.07;Re
adPosRankSum=0.524;SOR=0.693    GT:AD:DP:GQ:PL  0/1:1,4:5:15:102,0,15    ./.:0,0:0:.:0,0,0
chr1    14907   .       A       G       2162.44 PASS
AC=2;AF=0.500;AN=4;BaseQRankSum=1.26;ClippingRankSum=0.00;DP=117;ExcessHet=4.7712;FS=4.047;MLEAC=2;MLEAF=0.500;MQ=44.68;MQRankSum=-2.370e-01;QD=18.64;ReadPo
sRankSum=0.183;SOR=1.129        GT:AD:DP:GQ:PL  0/1:17,47:64:99:1169,0,391      0/1:12,40:52:99:1022,0,201
chr1    14930   .       A       G       3506.44 PASS
AC=2;AF=0.500;AN=4;BaseQRankSum=0.814;ClippingRankSum=0.00;DP=160;ExcessHet=4.7712;FS=6.365;MLEAC=2;MLEAF=0.500;MQ=44.54;MQRankSum=-2.025e+00;QD=21.92;ReadP
osRankSum=0.461;SOR=0.709       GT:AD:DP:GQ:PL  0/1:19,64:83:99:1814,0,306      0/1:12,65:77:99:1721,0,145
chr1    14933   .       G       A       10.92   snv-hard-filter
AC=2;AF=0.500;AN=4;BaseQRankSum=-5.500e-01;ClippingRankSum=0.00;DP=158;ExcessHet=4.7712;FS=4.667;MLEAC=2;MLEAF=0.500;MQ=44.88;MQRankSum=-2.764e+00;QD=0.07;R
eadPosRankSum=0.629;SOR=1.231   GT:AD:DP:GQ:PL  0/1:70,9:79:15:15,0,2269        0/1:69,10:79:24:24,0,2001
chr1    14976   .       G       A       16.95   snv-hard-filter
AC=1;AF=0.250;AN=4;BaseQRankSum=0.210;ClippingRankSum=0.00;DP=123;ExcessHet=3.0103;FS=3.614;MLEAC=1;MLEAF=0.250;MQ=41.60;MQRankSum=-1.883e+00;QD=0.31;ReadPo
sRankSum=-1.300e-02;SOR=0.697   GT:AD:DP:GQ:PL  0/1:48,7:55:46:46,0,1362        0/0:68,0:68:9:0,9,1771
chr1    16949   .       A       C       59.44   snv-hard-filter
AC=2;AF=0.500;AN=4;BaseQRankSum=2.74;ClippingRankSum=0.00;DP=18;ExcessHet=4.7712;FS=0.000;MLEAC=2;MLEAF=0.500;MQ=23.95;MQRankSum=0.728;QD=3.30;ReadPosRankSu
m=-7.890e-01;SOR=0.169  GT:AD:DP:GQ:PL  0/1:5,2:7:35:35,0,125    0/1:8,3:11:53:53,0,195
chr1    17020   .       G       A       77.44   snv-hard-filter
AC=2;AF=0.500;AN=4;BaseQRankSum=-1.440e+00;ClippingRankSum=0.00;DP=12;ExcessHet=4.7712;FS=3.282;MLEAC=2;MLEAF=0.500;MQ=23.53;MQRankSum=0.253;QD=6.45;ReadPos
RankSum=-4.310e-01;SOR=2.303    GT:AD:DP:GQ:PL  0/1:4,2:6:38:38,0,92     0/1:3,3:6:68:68,0,68
chr1    17385   .       G       A       91.44   snv-hard-filter
AC=2;AF=0.500;AN=4;BaseQRankSum=-8.760e-01;ClippingRankSum=0.00;DP=18;ExcessHet=4.7712;FS=0.000;MLEAC=2;MLEAF=0.500;MQ=35.20;MQRankSum=0.660;QD=5.08;ReadPos
RankSum=1.07;SOR=0.602  GT:AD:DP:GQ:PL  0/1:5,2:7:37:37,0,161    0/1:8,3:11:83:83,0,258
```

Output directory

trimmed
mapped
dedup
recal -> bam files
called
genotyped
filtered -> vcf file with all HC variants and PASS label
snpeff
annotated -> annotated vcf
qc
mutect
mutect_filter-> vcf files with MuTect2 variants and PASS label

Elena Piñeiro-Yáñez
epineiro@cnio.es
Coral Fustero-Torre
cfustero@cnio.es
María José Jiménez-Santos
mjjimenez@cnio.es
Fátima Al-Shahrour
falshahrour@cnio.es