

# Variant Calling

MEDT32/33

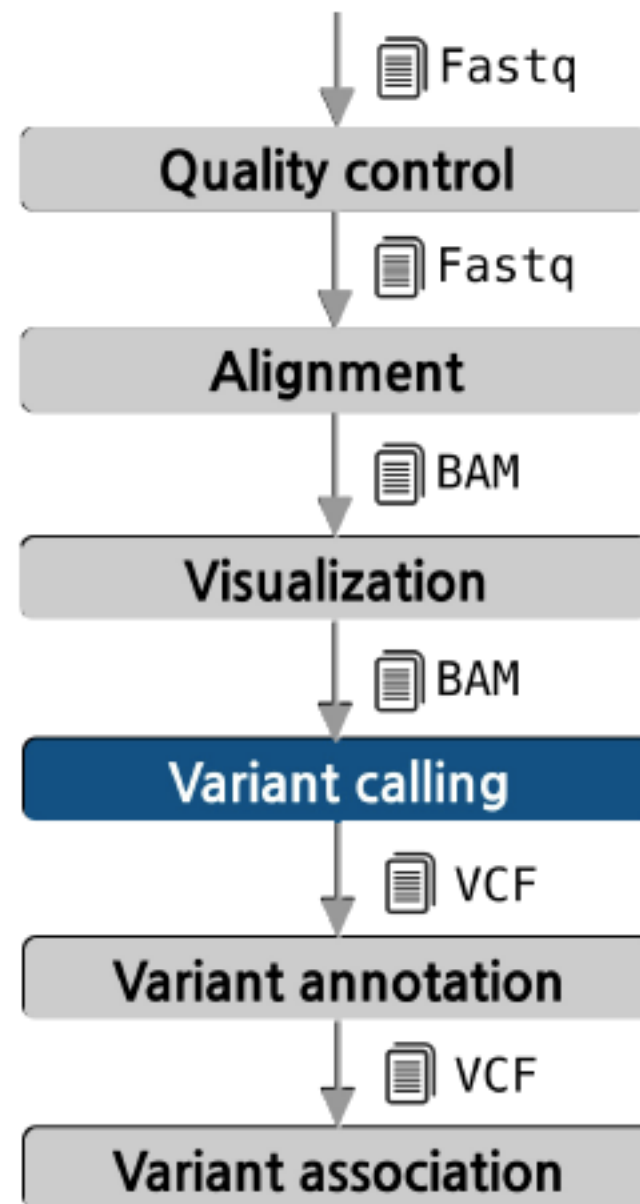
[http://genomemedicine.com/content/  
5/3/28/abstract](http://genomemedicine.com/content/5/3/28/abstract)

materials also from Marta Bleda  
Latorre

[mb2033@cam.ac.uk](mailto:mb2033@cam.ac.uk)

# The pipeline

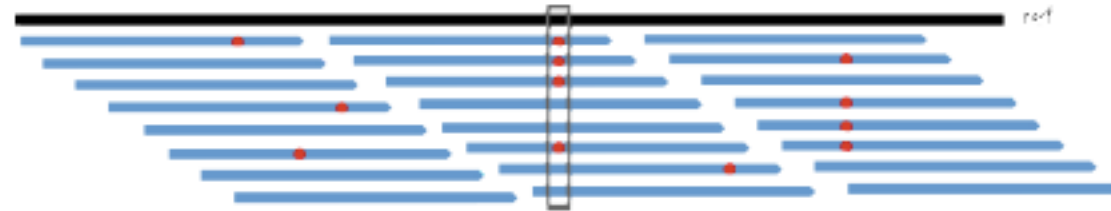
---



# Objective

---

Assign a genotype to each position



## Problems

Some variation observed in BAM files is caused by mapping and sequencing artifacts:

- **PCR artifacts:**
  - Mismatches due to errors in early PCR rounds
  - PCR duplicates
- **Sequencing errors:** erroneous call, either for physical reasons or to properties of the sequenced DNA
- **Mapping errors:** often happens around repeats or other low-complexity regions

Separate **true variation** from machine artifacts

# Variant calling process pipeline

---

## 1. Mark duplicates

Duplicates should not be counted as additional evidence

## 2. Local realignment around INDELS

Reads mapping on the edges of INDELS often get mapped with mismatching bases introducing false positives

## 3. Base quality score recalibration (BQSR)

Quality scores provided by sequencing machines are generally inaccurate and biased

## 4. Variant calling

Discover variants and their genotypes

# 1. Mark duplicates

---

- The same DNA molecule can be **sequenced several times during PCR**
- **Not informative**
- **Not** to be counted as **additional evidence** for or against a putative variant
- Can result in **false variant calls**

## Tools

- Samtools: `samtools rmdup`
- Picard: `MarkDuplicates`

# 1. Mark duplicates

The reason why duplicates are bad

✗ = sequencing error propagated in duplicates



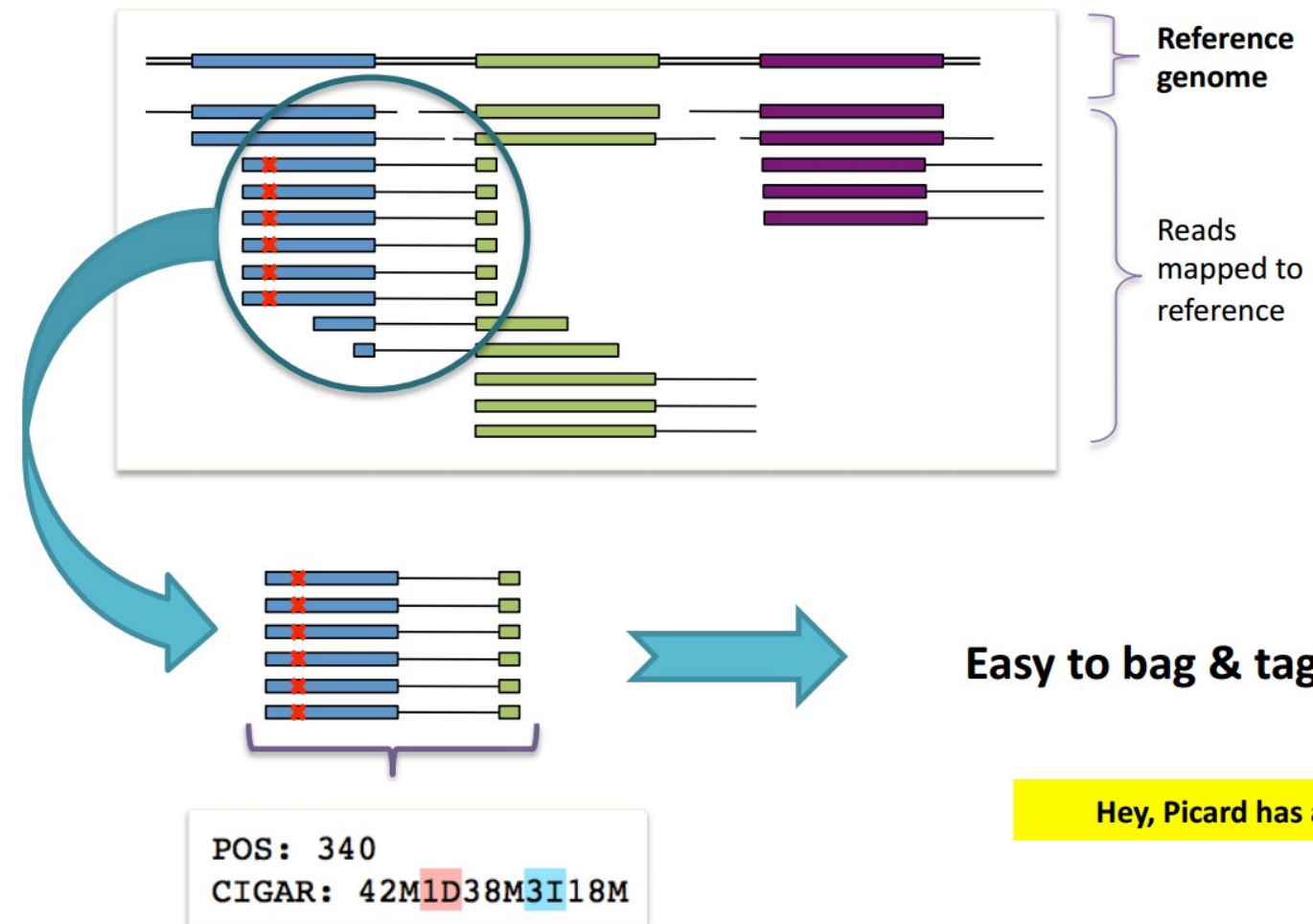
After marking duplicates, the GATK will only see :



# 1. Mark duplicates

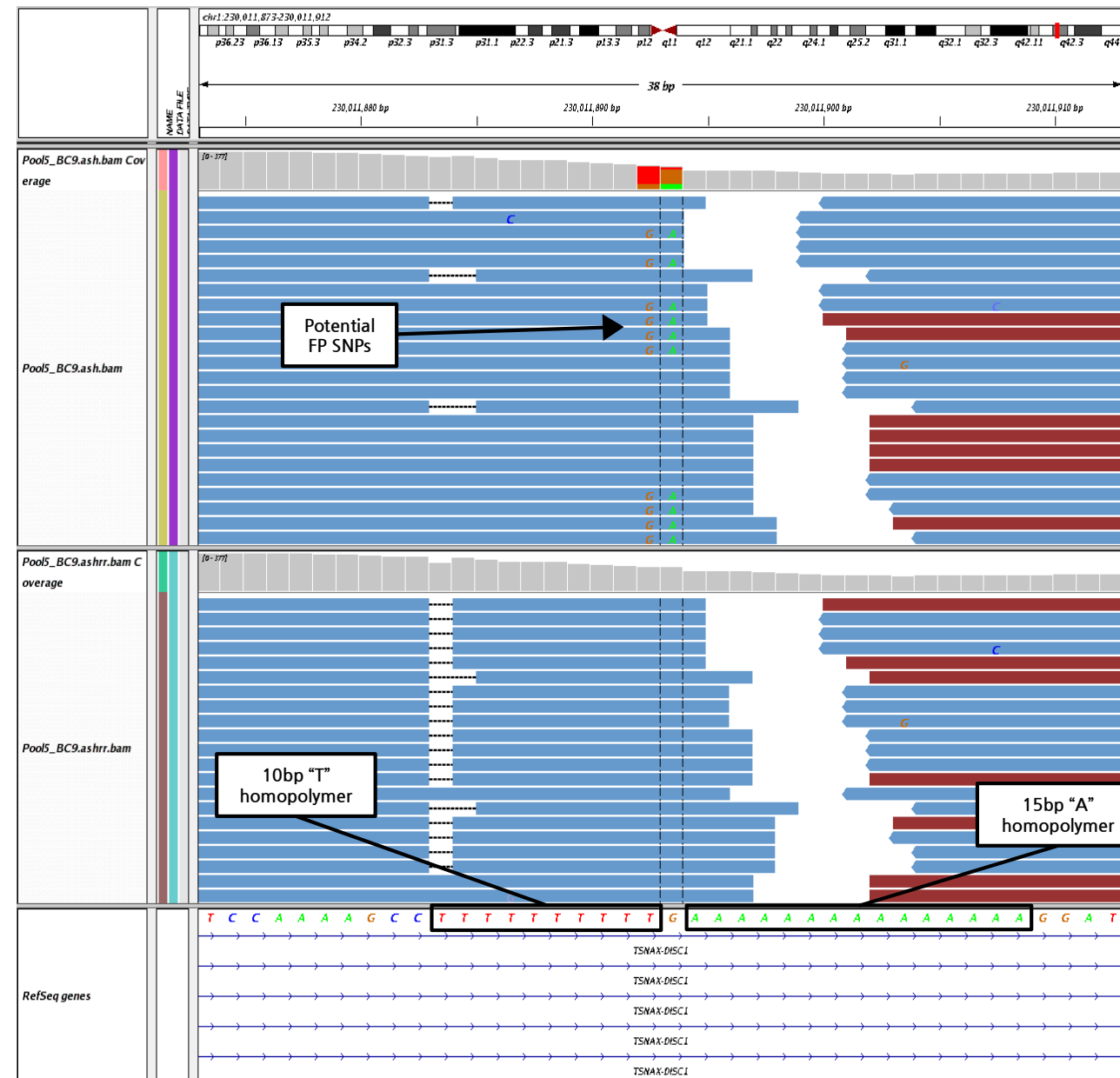
## Duplicate identification

Duplicates have the **same starting position** and the **same CIGAR string**



## 2. Local realignment around INDELS

- Reads **near INDELS** are mapped with mismatches
- **Realignment** can identify the most consistent placement for these reads
  1. **Identify** problematic regions
  2. **Determine the optimal** consensus sequence
- **Minimizes mismatches** with the reference sequence
- **Refines** location of **INDELS**

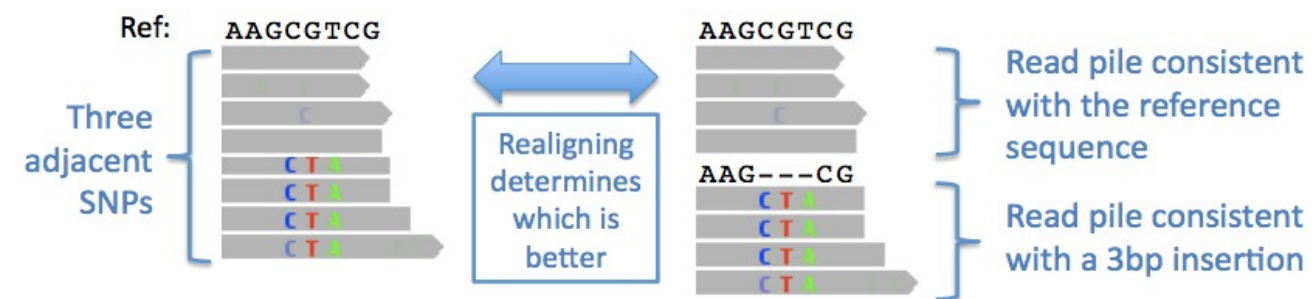




## 2. Local realignment around INDELS

---

- Reads **near INDELS** are mapped with mismatches
- **Realignment** can identify the most consistent placement for these reads
  1. **Identify** problematic regions
  2. **Determine the optimal** consensus sequence
- **Minimizes mismatches** with the reference sequence
- **Refines** location of **INDELS**



### 3. Base quality score recalibration

---

- **Calling algorithms** rely heavily on the **quality scores** assigned to the individual base calls in each sequence read
- Unfortunately, the scores produced by the machines are subject to various sources of **systematic error**, leading to over- or under-estimated base quality scores in the data

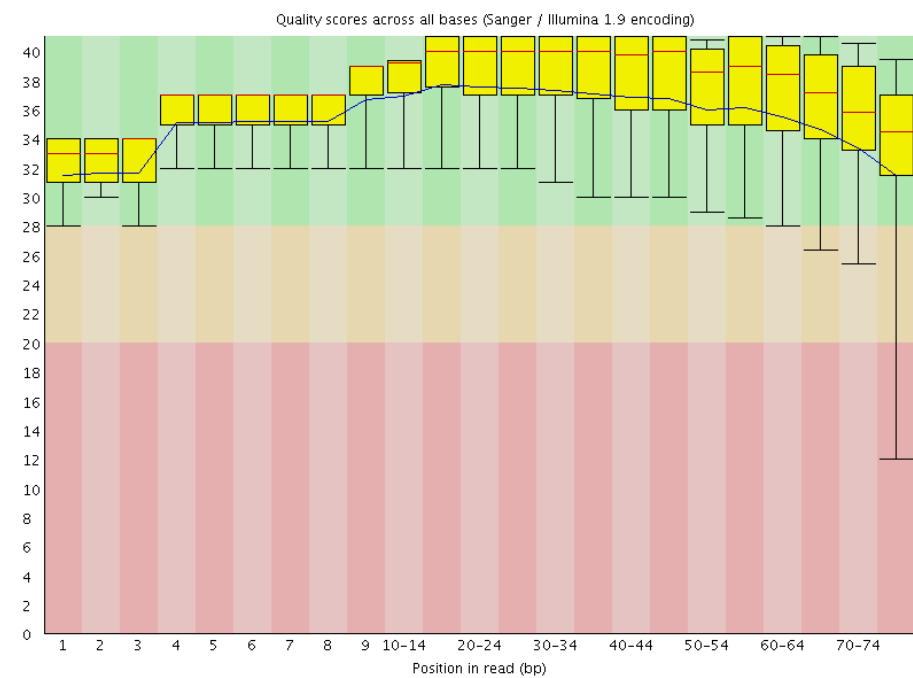
#### How?

1. **Analyze covariation** among several features of a base:
  - Reported quality score
  - Position within the read
  - Preceding and current nucleotide
2. Use a set of **known variants** (i.e.: dbSNP) to model error properties of real polymorphism and determine the **probability that novel sites are real**
3. **Adjust** the quality scores of all reads in a BAM file

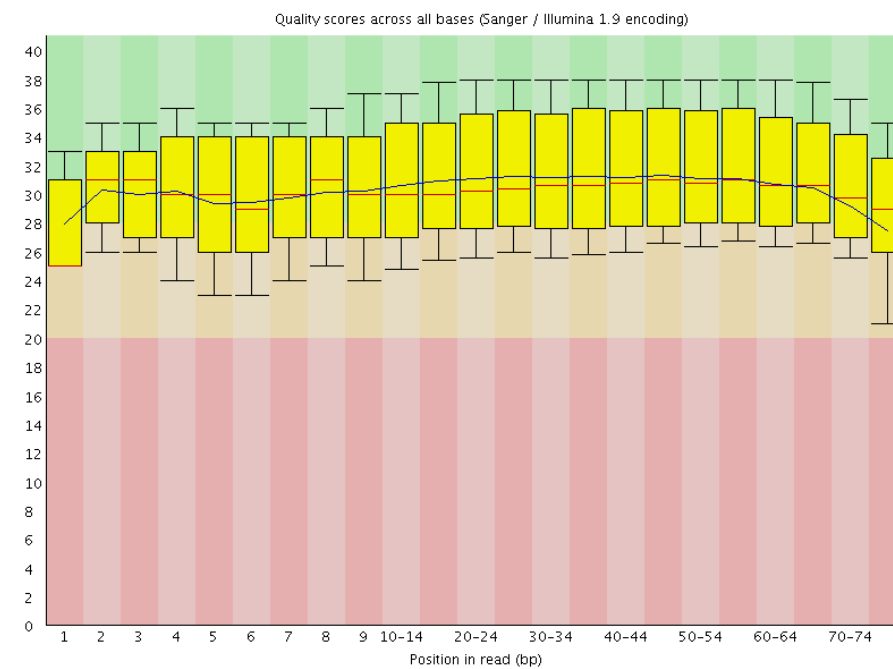
# 3. Base quality score recalibration

---

Before



After



Phred Quality score:

$$Q_{\text{Phred}} = -10 \log_{10} P(\text{error}).$$

A score of 20 corresponds to 1 % error rate in base calling

## 4. Variant calling

### Variant discovery process

---

#### Steps

1. **Variant calling:** Identify the positions that differ from the reference
2. **Genotype calling:** calculate the genotypes for each sample at these sites

#### Initial approach

**Independent** base assumption

Counting the number of times each allele is observed

#### Evolved approach

**Bayesian inference** → Compute genotype likelihood

Advantages:

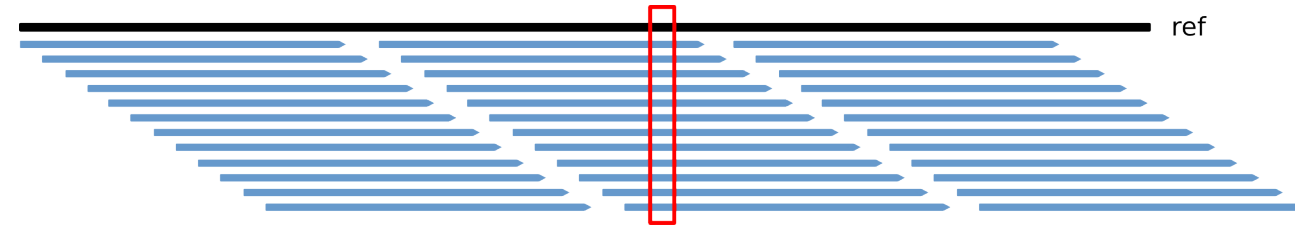
Provide statistical measure of **uncertainty**

Lead to **higher accuracy** of genotype calling

## 4. Variant calling

Variant discovery process

---



Reference = A

## 4. Variant calling

Variant discovery process



Reference = A

AAAAAAAAAAAAAAAAAAAAAAAAAAAAA

$N=30$ ,  $X=0$

$N$  = nucleotides  
 $G$  = true genotype  
 $R$  = reference base  
 $V$  = variant base  
 $X$  = variant nucleotides

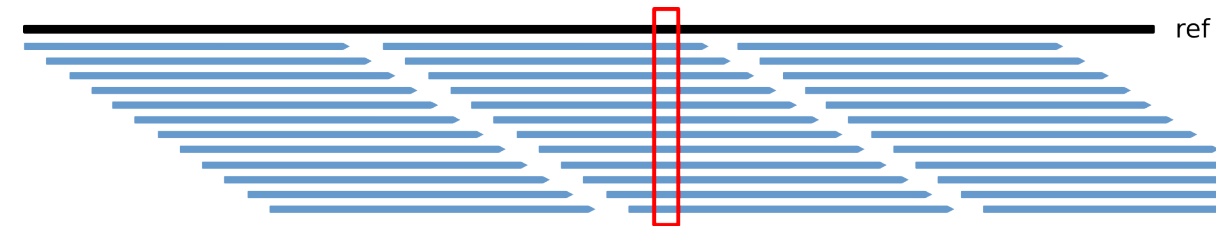
Outcomes:

RR RV VV

## 4. Variant calling

Variant discovery process

---



Reference = A

AAAAAAAAAAAAAAAAAAAAAAAAAAAA

$N=30$ ,  $X=0$

GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG

$N=30$ ,  $X=30$

$N$  = nucleotides  
 $G$  = true genotype  
 $R$  = reference base  
 $V$  = variant base  
 $X$  = variant nucleotides

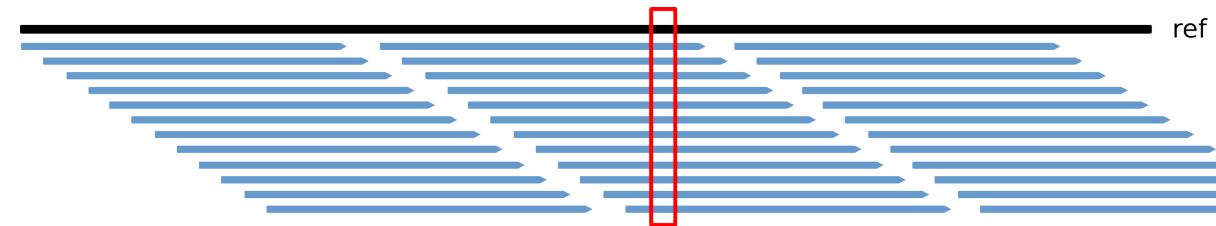
Outcomes:

RR RV VV

## 4. Variant calling

Variant discovery process

---



Reference = A

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

$N=30$ ,  $X=0$

GGGGGGGGGGGGGGGGGGGGGGGGGGGGGG

$N=30$ ,  $X=30$

AAAAAAAAAAAAAAAAAGGGGGGGGGGGGGGGG

$N=30$ ,  $X=15$

$N$  = nucleotides  
 $G$  = true genotype  
 $R$  = reference base  
 $V$  = variant base  
 $X$  = variant nucleotides

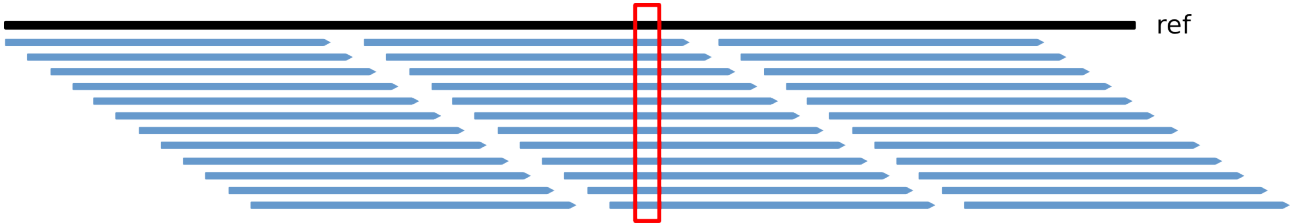
Outcomes:

RR RV VV



# 4. Variant calling

Variant discovery process



Reference = A

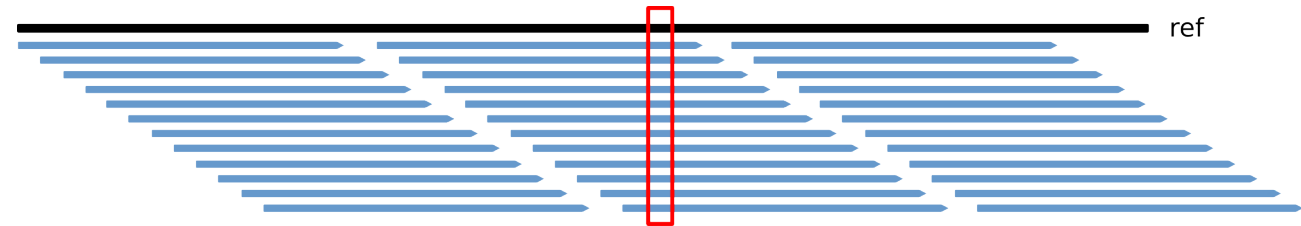
AAAAAAAAAAAAAAAAAAAAAAAAAAAA	$N=30,$	$X=0$
GGGGGGGGGGGGGGGGGGGGGGGGGGGG	$N=30,$	$X=30$
AAAAAAAAAAAAAAAAAGGGGGGGGGGGGGGGG	$N=30,$	$X=15$
AAAAAAAAAAAAAAAAAGGGGGGGGGGGGGGCT	$N=30,$	$X=12$

$N$  = nucleotides  
 $G$  = true genotype  
 $R$  = reference base  
 $V$  = variant base  
 $X$  = variant nucleotides

Outcomes:  
RR RV VV

# 4. Variant calling

Variant discovery process



Reference = A

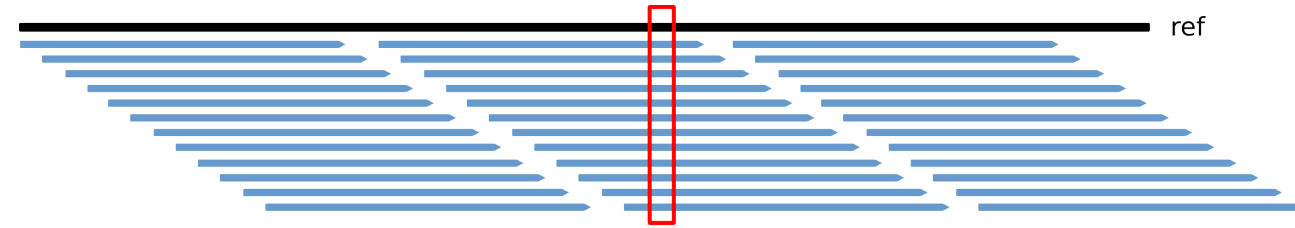
AAAAAAAAAAAAAAAAAAAAAAAAAAAAA	$N=30$ , $X=0$
GGGGGGGGGGGGGGGGGGGGGGGGGGGG	$N=30$ , $X=30$
AAAAAAAAAAAAAAAAAGGGGGGGGGGGGGG	$N=30$ , $X=15$
AAAAAAAAAAAAAAAAAGGGGGGGGGGGGGCT	$N=30$ , $X=12$
AAAGGGCCTT	$N=10$ , $X=3$

$N$  = nucleotides  
 $G$  = true genotype  
 $R$  = reference base  
 $V$  = variant base  
 $X$  = variant nucleotides

Outcomes:  
RR RV VV

# 4. Variant calling

## Variant discovery process



Reference = A

AAAAAAAAAAAAAAAAAAAAAAAAAAAA	$N=30, \quad X=0$
GGGGGGGGGGGGGGGGGGGGGGGGGG	$N=30, \quad X=30$
AAAAAAAAAAAAAAAAAAGGGGGGGGGGGGGGG	$N=30, \quad X=15$
AAAAAAAAAAAAAAAAAAGGGGGGGGGGGGGCT	$N=30, \quad X=12$
AAAGGGCCTT	$N=10, \quad X=3$

Cutoff for  $X \rightarrow$  value or proportion

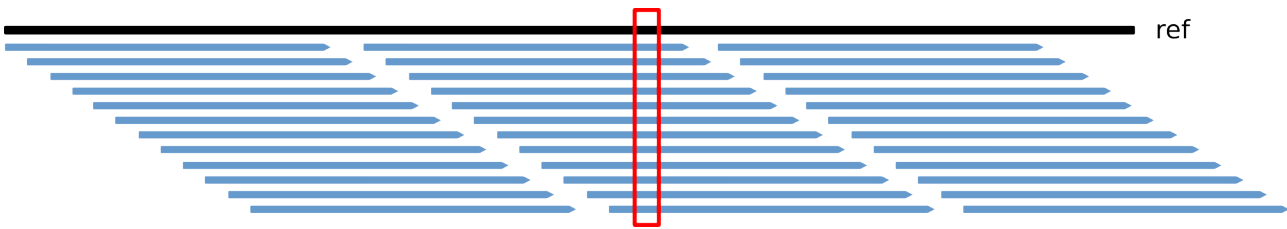
- $c = 30\% \quad X \leq c \rightarrow \mathbf{RR}, \quad X > c \rightarrow \mathbf{RV}$

$N$  = nucleotides  
 $G$  = true genotype  
 $R$  = reference base  
 $V$  = variant base  
 $X$  = variant nucleotides

Outcomes:  
RR RV VV

# 4. Variant calling

## Variant discovery process



Reference = A

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA	$N=30, \quad X=0 \rightarrow \mathbf{RR}$
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGG	$N=30, \quad X=30 \rightarrow \mathbf{VV}$
AAAAAAAAAAAAAAAAAAGGGGGGGGGGGGGG	$N=30, \quad X=15 \rightarrow \mathbf{RV}$
AAAAAAAAAAAAAAAAAAGGGGGGGGGGGGGCT	$N=30, \quad X=12 \rightarrow \mathbf{RV}$
AAAGGGCCTT	$N=10, \quad X=3 \rightarrow \mathbf{RV?}$

Cutoff for  $X \rightarrow$  value or proportion

- $c = 30\% \quad X \leq c \rightarrow \mathbf{RR}, \quad X > c \rightarrow \mathbf{RV}$
- $c_1 = 10\%, \quad c_2 = 30\% \quad \begin{array}{ll} X \leq c_1 & \rightarrow \mathbf{RR} \\ c_1 < X < c_2 & \rightarrow \mathbf{RV} \\ X \geq c_2 & \rightarrow \mathbf{RR} \end{array}$

$N$  = nucleotides  
 $G$  = true genotype  
 $R$  = reference base  
 $V$  = variant base  
 $X$  = variant nucleotides

Outcomes:  
RR RV VV

# VCF file format

- Specification defined by the 1000 genomes (current version **4.2**):  
<http://www.1000genomes.org/wiki/Analysis/Variant%20Call%20Format/vcf-variant-call-format-version-41>
- Commonly **compressed and indexed** with bgzip/tabix
- Single-sample or multi-sample VCF

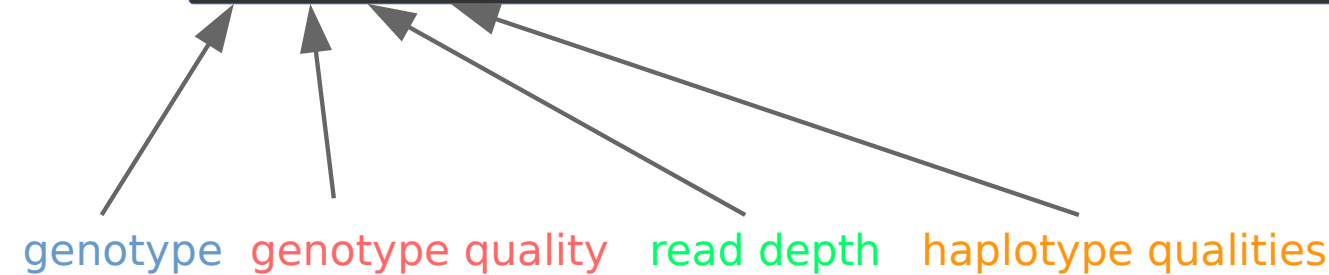
```
##fileformat=VCFv4.1
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA000001 NA000002 NA000003
20 14370 rs6054257 G A 29 PASS NS=3;DP=14;AF=0.5;DB;H2 GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51 1/1:43:5:.,.
20 17330 . T A 3 q10 NS=3;DP=11;AF=0.017 GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3 0/0:41:3
20 1110696 rs6040355 A G,T 67 PASS NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|1:2:0:18,2 2/2:35:4
20 1230237 . T . 47 PASS NS=3;DP=13;AA=T GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20 1234567 microsat1 GTC G,GTCT 50 PASS NS=3;DP=9;AA=G GT:GQ:DP 0/1:35:4 0/2:17:2 1/1:40:3
```

# VCF file format

---

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO
20	14370	rs6054257	G	A	29	PASS	NS=3;DP=14;AF=0.5;DB;H2
FORMAT		NA00001	NA00002		NA00003		
GT:GQ:DP:HQ		0 0:48:1:51,51	1 0:48:8:51,51		1/1:43:5:.,.		

genotype genotype quality read depth haplotype qualities



- **CHROM**: chromosome
  - **POS**: position
  - **ID**: identifier
  - **REF**: reference base(s)
  - **ALT**: non-reference allele(s)
  - **QUAL**: quality score of the calls (phed scale)
  - **FILTER**: “PASS” or a filtering tag
  - **INFO**: additional information
  - **FORMAT**: describes the information given by sample
-

# Software

Software	Available from	Calling method	Prerequisites	Comments	Refs
SOAP2	<a href="http://soap.genomics.org.cn/index.html">http://soap.genomics.org.cn/index.html</a>	Single-sample	High-quality variant database (for example, dbSNP)	Package for NGS data analysis, which includes a single individual genotype caller (SOAPsnp)	15
realSFS	<a href="http://128.32.118.212/thorfinn/realSFS/">http://128.32.118.212/thorfinn/realSFS/</a>	Single-sample	Aligned reads	Software for SNP and genotype calling using single individuals and allele frequencies. Site frequency spectrum (SFS) estimation	-
Samtools	<a href="http://samtools.sourceforge.net/">http://samtools.sourceforge.net/</a>	Multi-sample	Aligned reads	Package for manipulation of NGS alignments, which includes a computation of genotype likelihoods (samtools) and SNP and genotype calling (bcftools)	53
GATK	<a href="http://www.broadinstitute.org/gsa/wiki/index.php/The_Genome_Analysis_Toolkit">http://www.broadinstitute.org/gsa/wiki/index.php/The_Genome_Analysis_Toolkit</a>	Multi-sample	Aligned reads	Package for aligned NGS data analysis, which includes a SNP and genotype caller (Unified Genotyper), SNP filtering (Variant Filtration) and SNP quality recalibration (Variant Recalibrator)	32,33
Beagle	<a href="http://faculty.washington.edu/browning/beagle/beagle.html">http://faculty.washington.edu/browning/beagle/beagle.html</a>	Multi-sample LD	Candidate SNPs, genotype likelihoods	Software for imputation, phasing and association that includes a mode for genotype calling	42
IMPUTE2	<a href="http://mathgen.stats.ox.ac.uk/impute/impute_v2.html">http://mathgen.stats.ox.ac.uk/impute/impute_v2.html</a>	Multi-sample LD	Candidate SNPs, genotype likelihoods	Software for imputation and phasing, including a mode for genotype calling. Requires fine-scale linkage map	44
QCall	<a href="ftp://ftp.sanger.ac.uk/pub/rd/QCALL">ftp://ftp.sanger.ac.uk/pub/rd/QCALL</a>	Multi-sample LD	'Feasible' genealogies at a dense set of loci, genotype likelihoods	Software for SNP and genotype calling, including a method for generating candidate SNPs without LD information (NLDA) and a method for incorporating LD information (LDA). The 'feasible' genealogies can be generated using Margarita ( <a href="http://www.sanger.ac.uk/resources/software/margarita">http://www.sanger.ac.uk/resources/software/margarita</a> )	54
MaCH	<a href="http://genome.sph.umich.edu/wiki/Thunder">http://genome.sph.umich.edu/wiki/Thunder</a>	Multi-sample LD	Genotype likelihoods	Software for SNP and genotype calling, including a method (GPT_Freq) for generating candidate SNPs without LD information and a method (thunder_glf_freq) for incorporating LD information	-

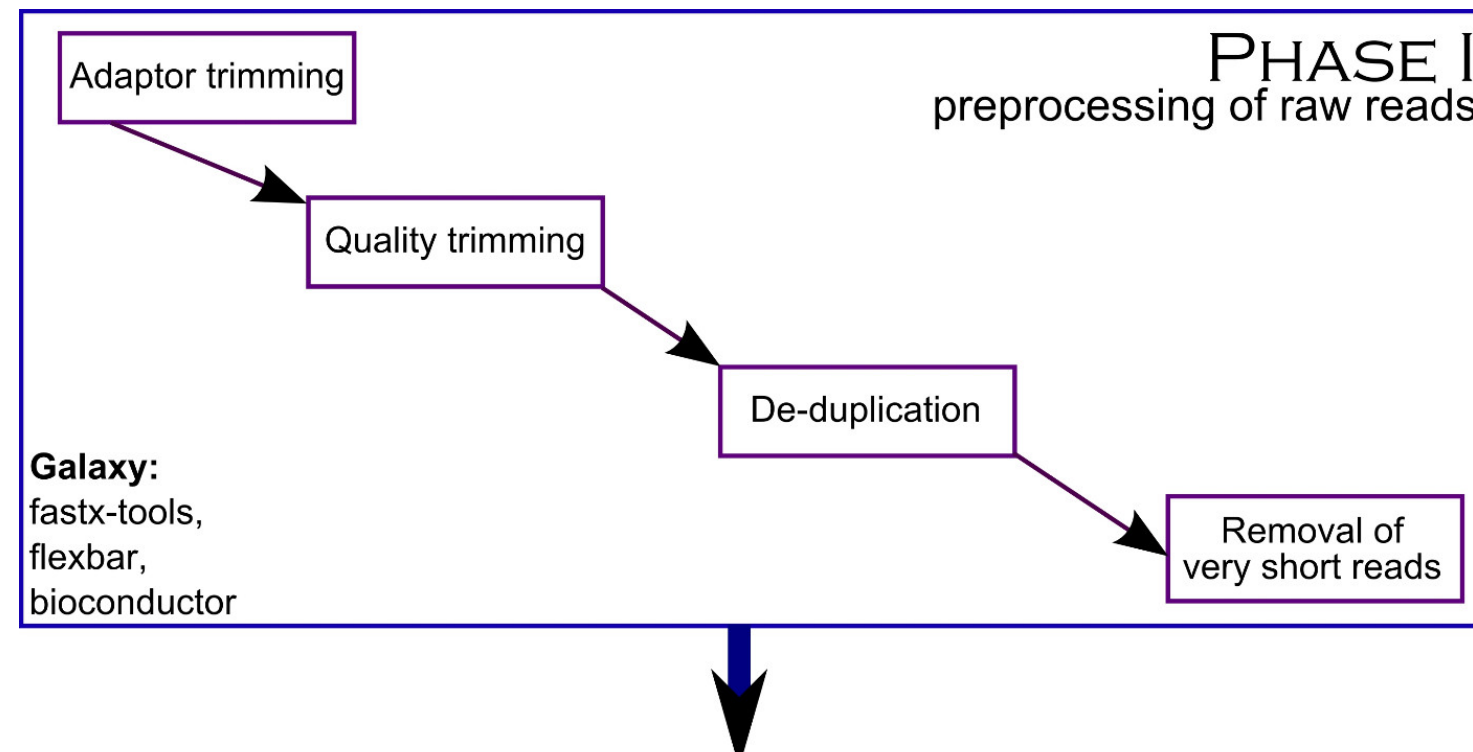
A more complete list is available from <http://seqanswers.com/wiki/Software/list>. LD, linkage disequilibrium; NGS, next-generation sequencing.

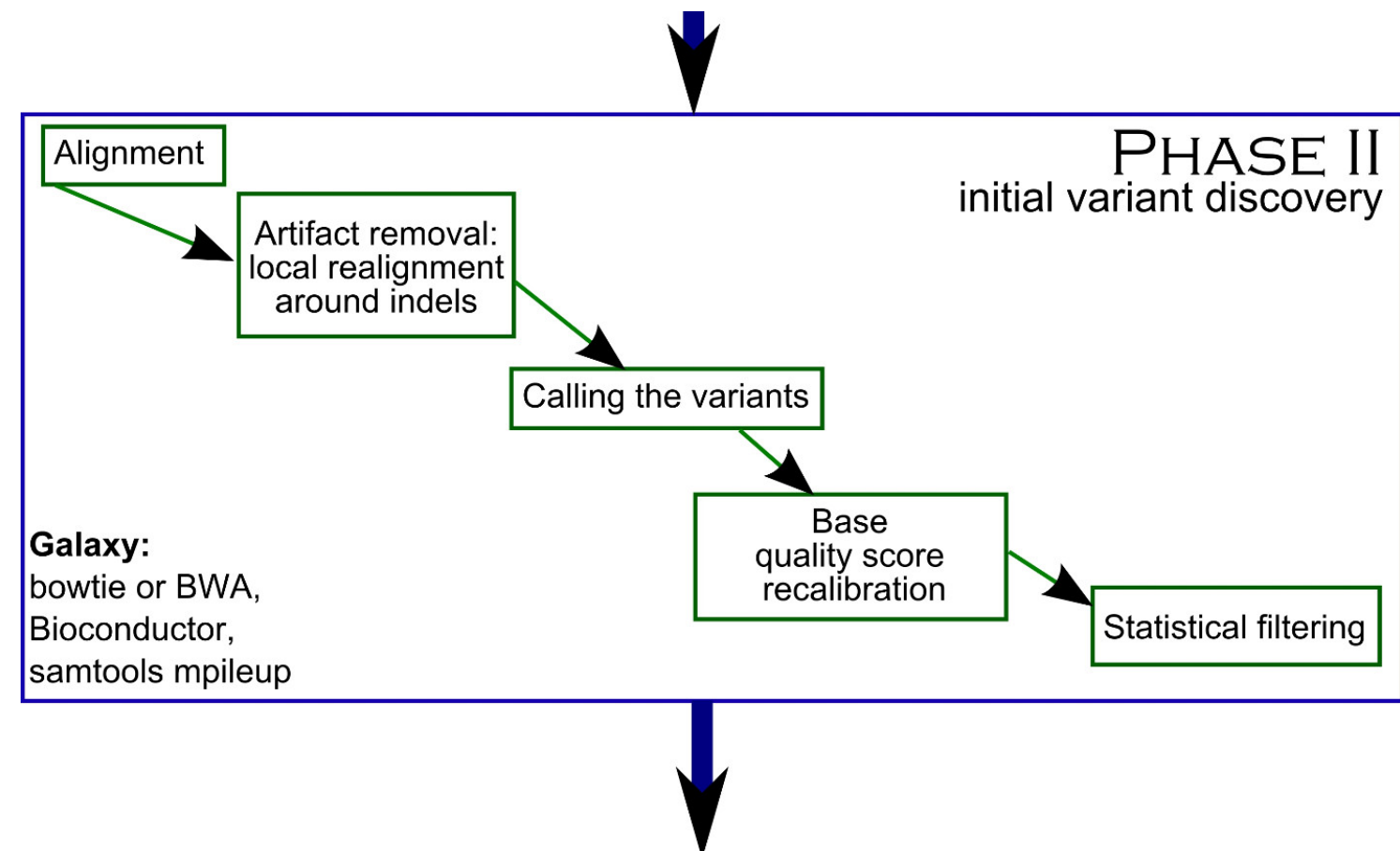
# *Variant calling*

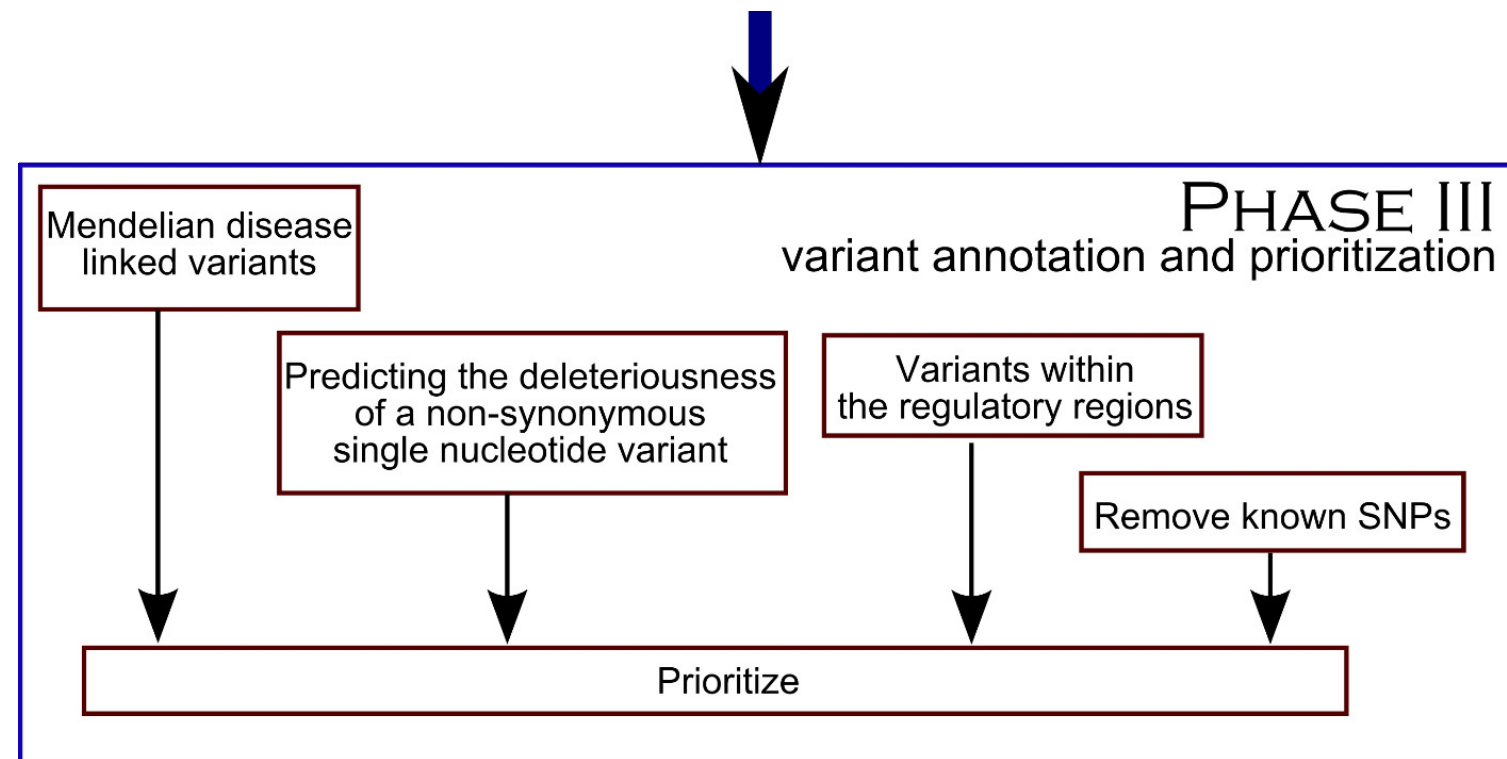
- challenging due to rapidly changing algorithms
- relatively low concordance between methods
- comparing approaches:
  - <http://bcbio.wordpress.com/2013/05/06/framework-for-evaluating-variant-detection-methods-comparison-of-aligners-and-callers/>



# overview







- Pabinger et al. (2013) provides a good discussion of the common tools and approaches for variant calling.
- Also see the older Nielsen et al. (2011).

S Pabinger, A Dander, M Fischer, R Snajder, M Sperk, M Efremova, B Krabichler, MR. Speicher, J Zschocke, Z Trajanoski (2013) A survey of tools for variant analysis of nextgeneration genome sequencing data. Briefings in Bioinformatics (21 January 2013).  
Nielsen R, Paul JS, Albrechtsen A, Song YS (2011) Genotype and SNP calling from nextgeneration sequencing data. Nature Reviews 12:443451.

# Adaptor trimming

- (2-90% of read)
  - fastq format (Fastq 2013),
  - Fastxtoolkit (fastx\_clipper), Bioconductor (ShortRead package), Flexbar (Dodt et al. 2012)
  - (<http://bioscholar.com/genomics/toolsremoveadaptersequencesnextgenerationsequencingdata/>)  
.
- Selection of the tool to use depends on the amount of adaptor sequence leftover in the data.

- (2)
- Most aligners do a good job of hard and soft clipping reads now so it's not as crucial. Best practice is to throw things right in and only trim if absolutely necessary since alignment errors and variants will get filtered downstream.

- Selection of the tool to use depends on the amount of adaptor sequence leftover in the data.

# Quality trimming

- inspect reads in bulk
- quality tends to drop off toward one end of the read. PrinSeq

Schmieder R and Edwards R (2011) Quality control and preprocessing of metagenomic datasets. *Bioinformatics* 27:863864.

# Removal of very short reads

- short reads are likely to align to multiple (wrong) locations on reference



# De-duplication

- Duplicately sequenced molecules should not be counted as additional evidence for or against a putative variant – they must be removed prior to the analysis
- Picard: MarkDuplicates,
- FASTXToolkit: has fastx\_collapser

blog post by Eric Vallabh Minikel  
2012

pre-alignment processing does help reduce number of problem reads but it also doesn't scale well to multiple samples.

- D All of the fastq manipulation is disk intensive, since - reading and writing big files, also gets slow trying to do a large number in parallel.
- P
- F

blog post by Eric Vallabh Minikel  
2012

# Initiation Variants Alignment

## GATK

- Reads that align on the edges of indels often get mapped with mismatching bases that might look like evidence for SNPs. We look for the most consistent placement of the reads with respect to the indel in order to clean up these artifacts.

# Alignment

- bwa for shorter read (< 75bp)
- bwa mem for longer reads
- outperform bowtie2 in most alignment comparisons
- novoalign (<http://www.novocraft.com/main/index.php>) requires a license. It is slower than bwa but more comprehensive since it does full smith-waterman. Newer version (3.0+) handle both short and long reads.

# Base quality score recalibration

- The per base estimate of error (base quality score)
- estimates provided by the sequencing machines are often inaccurate
- empirically accurate error model through recalibration

GATK: QSR

DePristo MA, Banks E, Poplin R, Garimella KV, Maguire JR, Hartl C, Philippakis AA, del Angel G, Rivas MA, Hanna M, McKenna A, Fennell TJ, Kernytsky AM, Sivachenko AY, Cibulskis K, Gabriel SB, Altshuler D, Daly MJ. (2011) A framework for variation discovery and genotyping using nextgeneration DNA sequencing data. Nat Genet. 43(5):4918.

# Calling

- Only those variants should be kept that have a high confidence score:
- minimum Q30 for deep coverage data ( $>10\times$  per sample) and
- minimum Q4 (if  $\leq 100$  samples) or Q10 (if  $> 100$  samples) for shallower coverage.
- The variant calls are usually produced in form of VCF files (1020 M of hard disk space per file),

<http://www.1000genomes.org/wiki/Analysis/vcf4.0>

- quality of variant call sets is expected to increase substantially if multiple variant callers are used

- GATK HaplotypeCaller
- - FreeBayes (<https://github.com/ekg/freebayes>, <http://gkno.me/>)
- - GATK UnifiedGenotyper
- - samtools (performs poorly on indels)



# Statistical filtering

- raw VCF files frequently have many sites that are not really genetic variants
- separate out the false positive machine artifacts from the true positive genetic variants
- variant quality score recalibration

[http://gatkforums.broadinstitute.org/discussion/39/  
variantqualityscorecalibrationvqsr](http://gatkforums.broadinstitute.org/discussion/39/variantqualityscorecalibrationvqsr)

# Variant annotation and prioritization

- Mendelian disease linked variants:VARMD,
- KGGSeq, FamSeq (Peng et al 2013).
- Predicting the deleteriousness of a nonsynonymous single nucleotide variant: dbNSFP,
- HuVariome, SeattleSeq,ANNOVAR,VAAST, snpEff
- Identifying variants within the regulatory regions: RegulomeDB (Boyle et al 2012)
- **GEMINI**

## GEMINI - integrative exploration of genetic variation and genome annotations.

- Annotate Functional Impact of Each Variant
- integrated genetic variation (from VCF files) with a wealth of genome annotations into a unified database framework
- 1,000,000 variants times 1,000 samples yields one billion genotypes

<http://gemini.readthedocs.org/en/latest/>

- (<https://github.com/arq5x/gemini>, <http://gemini.readthedocs.org/en/latest/>),
- works after standard annotators like snpEff and provides a database of variants associated with external annotations (dbSNP, ENCODE, ClinVar, OMIM, KEGG)

- distinguish synonymous, non-synonymous, conserved splice site, 5'UTR, 3'UTR and other variants
- [http://genome.sph.umich.edu/wiki/  
Generic\\_Exome\\_Analysis\\_Plan](http://genome.sph.umich.edu/wiki/Generic_Exome_Analysis_Plan)

# Challenges

- Variants being called individually and not in a haplotype aware manner
- GC rich regions or regions that produce PCR or sequencing artifacts
- Regions of the genome with low mappability or high homology to other regions

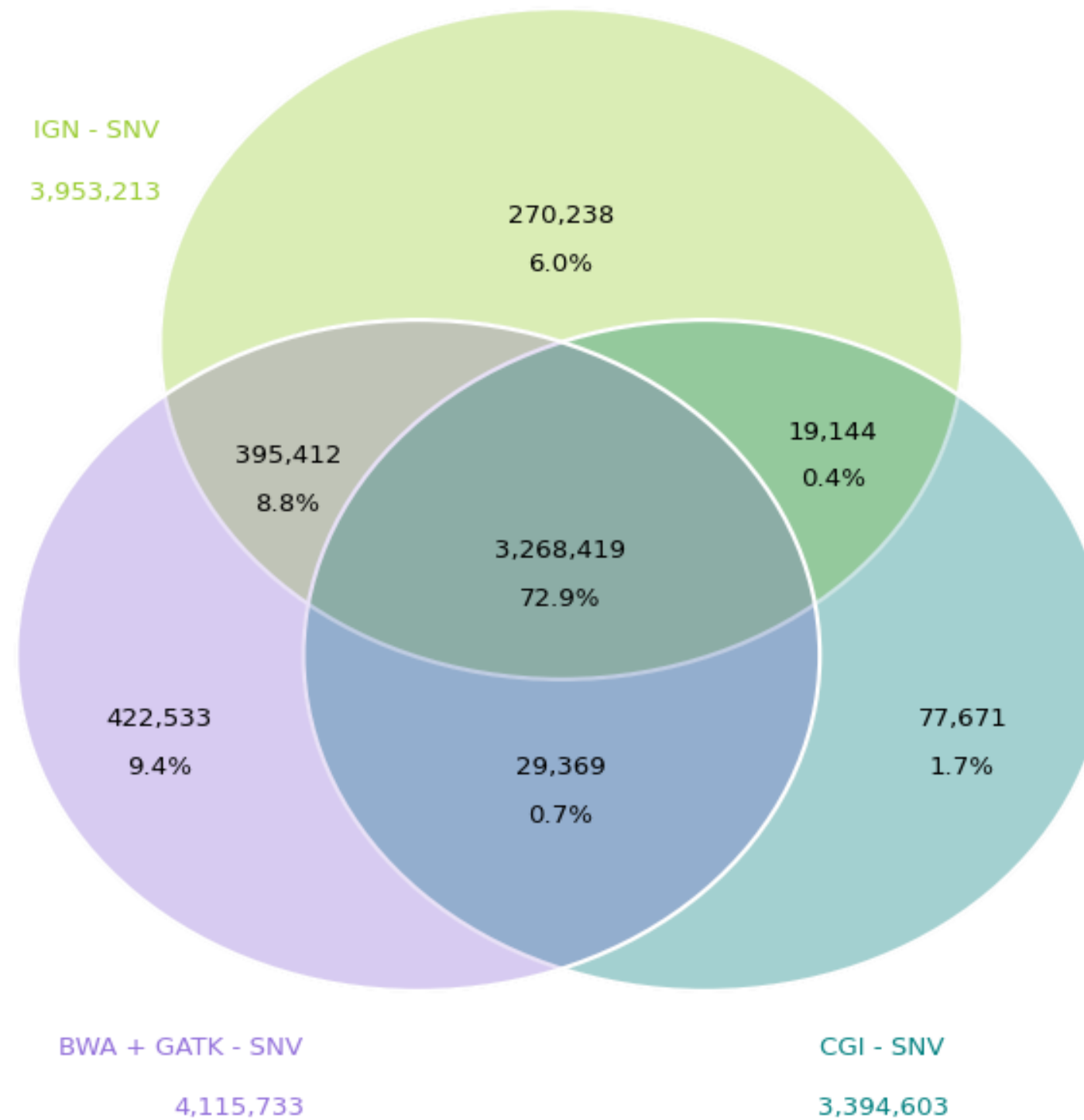
Poor similarity between the sample's genome and the reference.

- Structural variants : inversions or translocations
- tandem repeats or mobile element insertions
- reference sequence is unable to capture the allelic diversity of the population and the sample

# Heterozygote call

- 30x genome : heterozygote on average to be covered by 15 reads to support the mutant allele
- loci where a heterozygote legitimately exists but it gets sampled only once or twice
  - 1.74 times when calling 4 million variants





State of Variant calling : don;t panic!  
<http://blog.goldenhelix.com/?p=1725>

## Germline variant calling

bcbio implements configurable SNP, indel and structural variant calling for germline populations. We include whole genome and exome evaluations against reference calls from the [Genome in a Bottle](#) consortium and [Illumina Platinum Genomes](#) project, enabling continuous assessment of new alignment and variant calling algorithms. We regularly report on these comparisons and continue to improve approaches as the community makes new tools available. Here is some of the research that contributes to the current implementation:

- An introduction to the [variant evaluation framework](#). This includes a comparison of the [bwa mem](#) and [novoalign](#) aligners. We also compared the [FreeBayes](#), [GATK HaplotypeCaller](#) and [GATK UnifiedGenotyper](#) variant callers.
- An in-depth evaluation of [FreeBayes](#) and [BAM post-alignment processing](#). We found that FreeBayes quality was equal to GATK HaplotypeCaller. Additionally, a lightweight post-alignment preparation method using only de-duplication was equivalent to GATK's recommended Base Quality Score Recalibration (BQSR) and realignment around indels, when using good quality input datasets and callers that do local realignment.
- Additional work to [improve variant filtering](#), providing methods to remove low complexity regions (LCRs) that can bias indel results. We also tuned [GATK's Variant Quality Score Recalibrator](#) (VQSR) and compared it with hard filtering. VQSR requires a large number of variants and we use it in bcbio with GATK HaplotypeCaller when your [Algorithm parameters](#) contains high depth samples ( `coverage_depth` is not low) and you are calling on the whole genome ( `coverage_interval` is genome) or have more than 50 regional or exome samples called concurrently.
- An [evaluation of joint calling](#) with GATK HaplotypeCaller, FreeBayes, Platypus and samtools. This validates the joint calling implementation, allowing scaling of large population germline experiments. It also demonstrates improved performance of new callers: samtools 1.0 and Platypus.
- Support for [build 38 of the human genome](#), improving precision of detection thanks to the improved genome representation.

BCBIO-NEXTGEN