Variant Calling and Annotation

Variant Calling and Annotation

1. Bam postprocessing
2. Variant calling
3. Variant calling format
4. Variant normalization
5. Variant filtering
6. Variant annotation and prioritization

Variant Calling and annotation

1. Bam postprocessing
2. Variant calling
3. Variant calling format
4. Variant normalization
5. Variant filtering
6. Variant annotation and prioritization

BAM postprocessing

-- Preliminary steps for variant calling --

- BAM sorting
- PCR duplicates bias
- Indel problems

BAM postprocessing

You already did these steps :

- Fastq demultiplexing
- Fastq trimming (Adapters and quality)
- Quality checking of the reads
- Reads alignment in BAM format
- Quality checking of the alignment

BAM postprocessing - Sorting

- Sorting by read name
  - Default read order from the sequencer
  - Keep the same order as raw fastq file for particular post processing
    - UMI (molecular barcodes)
    - Compare fastq reads with aligned reads
- Sorting by alignment position
  - To accelerate bam processing and variant calling
  - To allow efficient visualisation
  - To have better compression ratio
    - Read order sorted (150x exome) : 11 Go
    - Position sorted (150x exome) : 7.3 Go
  - Sorting by position means you also have an index file (bam+bai)
- Many tools need a position sorted bam file !
- Bam format has a specific tag (SO)

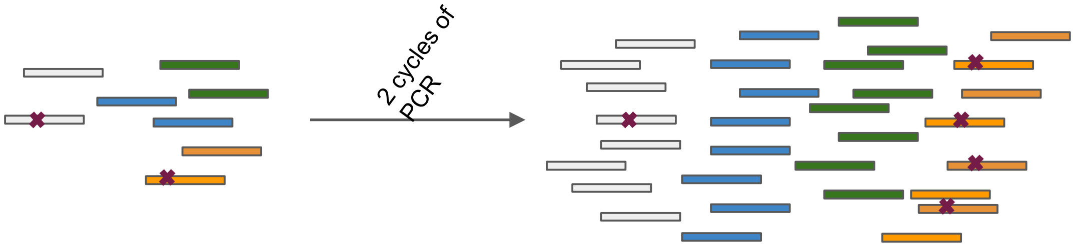
BAM postprocessing - Sorting

Software:

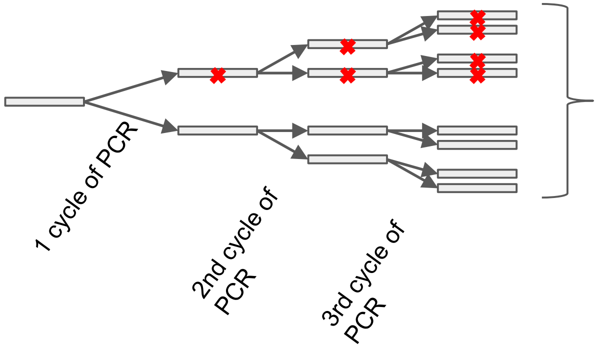
- Samtools ‘sort’
- Picard tools ‘sort sam’
- Sambamba ‘sort’

BAM postprocessing - PCR duplicates

Non uniform amplification: some allele can be preferentially amplified

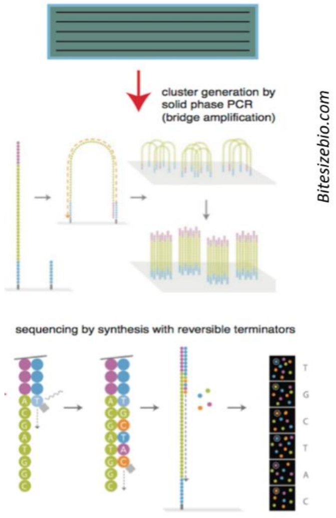


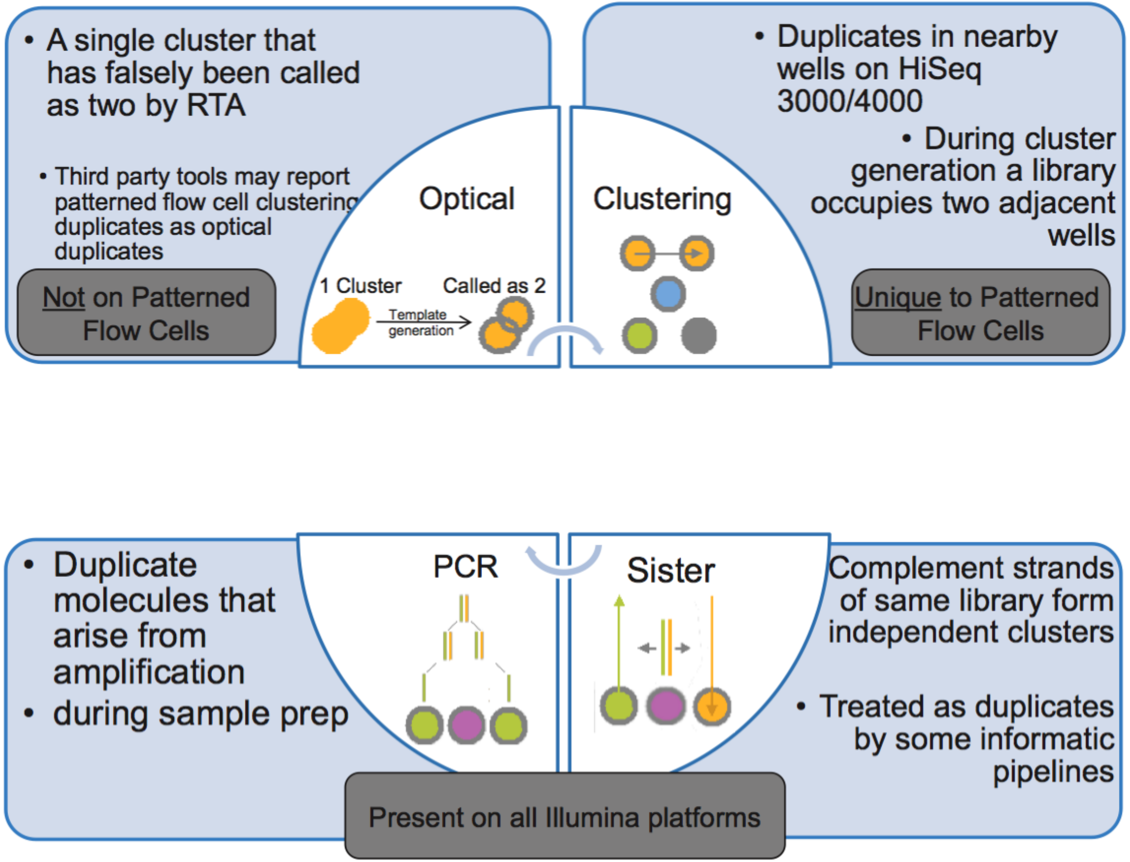
PCR error: an error in first PCR cycle is propagated and amplified



4 variant copies: heterozygous variant with non uniform amplification, low allelic variant (Somatic) or artefact ?

BAM postprocessing - Optical PCR duplicates


cluster generation by solid phase PCR (bridge amplification)
sequencing by synthesis with reversible terminators
Bitesizebio.com



- A single cluster that has falsely been called as two by RTA
- Third party tools may report patterned flow cell clustering duplicates as optical duplicates

Not on Patterned Flow Cells

Optical

1 Cluster

Template generation

Called as 2

Clustering

- Duplicates in nearby wells on HiSeq 3000/4000
- During cluster generation a library occupies two adjacent wells

Unique to Patterned Flow Cells

- Duplicate molecules that arise from amplification
- during sample prep

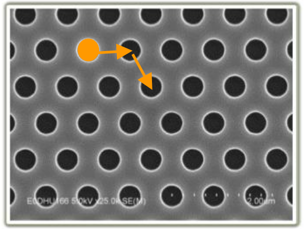
PCR

Sister

Complement strands of same library form independent clusters

- Treated as duplicates by some informatic pipelines

Present on all Illumina platforms



From http://core-genomics.blogspot.fi/2016/01/almost-everything-you-wanted-to-know.html

BAM postprocessing - PCR Duplicates software

Tool : Picard Tools / MarkDuplicate

- Mark duplicate / Do not remove duplicated reads !
- Keep a representative read (<u>best base quality scores</u>, maximum length or random)
- All reads starting at the same oriented position are duplicates
  - Same start of both reads for paired-end
- Optical duplicates (based on distance of the clusters) are flagged as DT:SQ*, others are flagged as DT:LD (Library duplicates)

Tool : Picard Tools / MarkDuplicateWithMateCigar

- Mark duplicate / Do not remove duplicated reads !
- Keep a representative fragment (<u>maximum length</u>, best base quality scores or random)
- All reads starting at the same oriented position with identical CIGAR are duplicates

Tool : Samtools / Rmdup

- Removes duplicate reads only

* SQ stand for sequencing platform artifactual duplicate

BAM postprocessing - PCR Duplicates software

GATK
Best
Practices

Tool : Picard Tools / MarkDuplicate

- Mark duplicate / Do not remove duplicated reads !
- Keep a representative read (best base quality scores, maximum length or random)
- All reads starting at the same oriented position are duplicates
  - Same start of both reads for paired-end
- Optical duplicates (based on distance of the clusters) are flagged as DT:SQ*, others are flagged as DT:LD (Library duplicates)

Tool : Picard Tools / MarkDuplicateWithMateCigar

- Mark duplicate / Do not remove duplicated reads !
- Keep a representative fragment (maximum length, best base quality scores or random)
- All reads starting at the same oriented position with identical CIGAR are duplicates

Tool : Samtools / Rmdup

- Removes duplicate reads only

* SQ stand for sequencing platform artifactual duplicate

BAM postprocessing - Duplicates management

Always MarkDuplicate

… or almost always ...

- Not for Amplicon or Amplicon-like Sequencing (All read starts are of course the same !)
- Not for RNA-sequencing (depending on application)

BAM postprocessing - Indels problem

- The alignment of indels depends on the score of each base-alignment
  - Match, Mismatch, Gap open, Gap extension penality, alignment clipping
- The aligner align read independently (we do not know if the alignment is consistent)

This is due to indels at the end of the reads

Exemple

```
Ref   T A C C C A T T T T T T T C T A A A A G C T
Read1         C C A T T T T T T C T A A A A A C T
```

The best scored pairwise alignment

```
Ref   T A C C C A T T T T T T T C T A A A A G C T
ReadN   A C C C A - T T T T T T C T A A A
```

Another best scored pairwise alignment

(Example from GATK Best Practice for Variant Discovery)

BAM postprocessing - Indels problem

- The alignment of indels depends on the score of each base-alignment
  - Match, Mismatch, Gap open, Gap extension penality, alignment clipping
- The aligner align read independently (we do not know if the alignment is consistent)

This is due to indels at the end of the reads

Exemple

```
Ref   T A C C C A T T T T T T T C T A A A A G C T
Read1         C C A T T T T T T C T A A A A A C T
```

The best scored pairwise alignment

```
Ref   T A C C C A T T T T T T T C T A A A A G C T
ReadN   A C C C A - T T T T T T C T A A A
Read1         C C A T T T T T T C T A A A A A C T
```

Inconsistency of pairwise alignment

(Example from GATK Best Practice for Variant Discovery)

BAM postprocessing - Indels problem

- The alignment of indels depends on the score of each base-alignment
  - Match, Mismatch, Gap open, Gap extension penality, alignment clipping
- The aligner align read independently (we do not know if the alignment is consistent)

This is due to indels at the end of the reads

Exemple

```
Ref    T A C C C A T T T T T T T C T A A A A G C T
Read1          C C A T T T T T T C T A A A A A C T
```

The best scored pairwise alignment

```
Ref    T A C C C A T T T T T T T C T A A A A G C T
ReadN    A C C C A - T T T T T T C T A A A
Read1          C C A T T T T T T C T A A A A A C T
```

Inconsistency of pairwise alignment

```
Ref    T A C C C A T T T T T T T C T A A A A G C T
ReadN    A C C C A - T T T T T T C T A A A
Read1        C C A - T T T T T T C T A A A A A C T
```

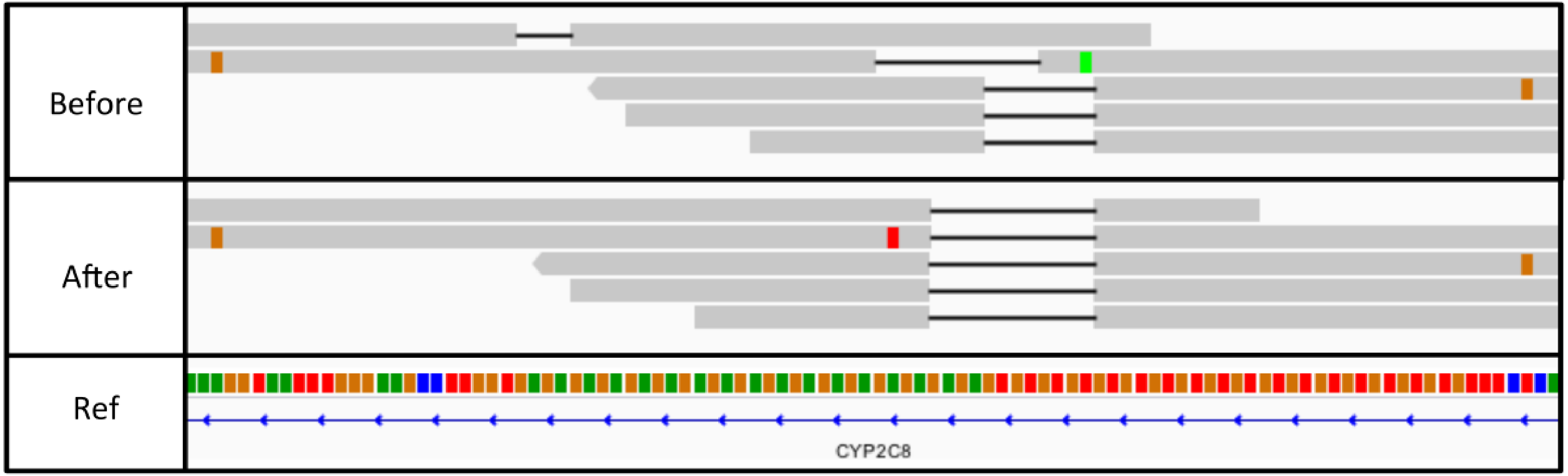
Consistent alignment

From GATK Best Practices for Variant Discovery Presentation, https://software.broadinstitute.org/gatk/download/workshops

BAM postprocessing - Indels problem

Realigning indels taking into account other reads allows:

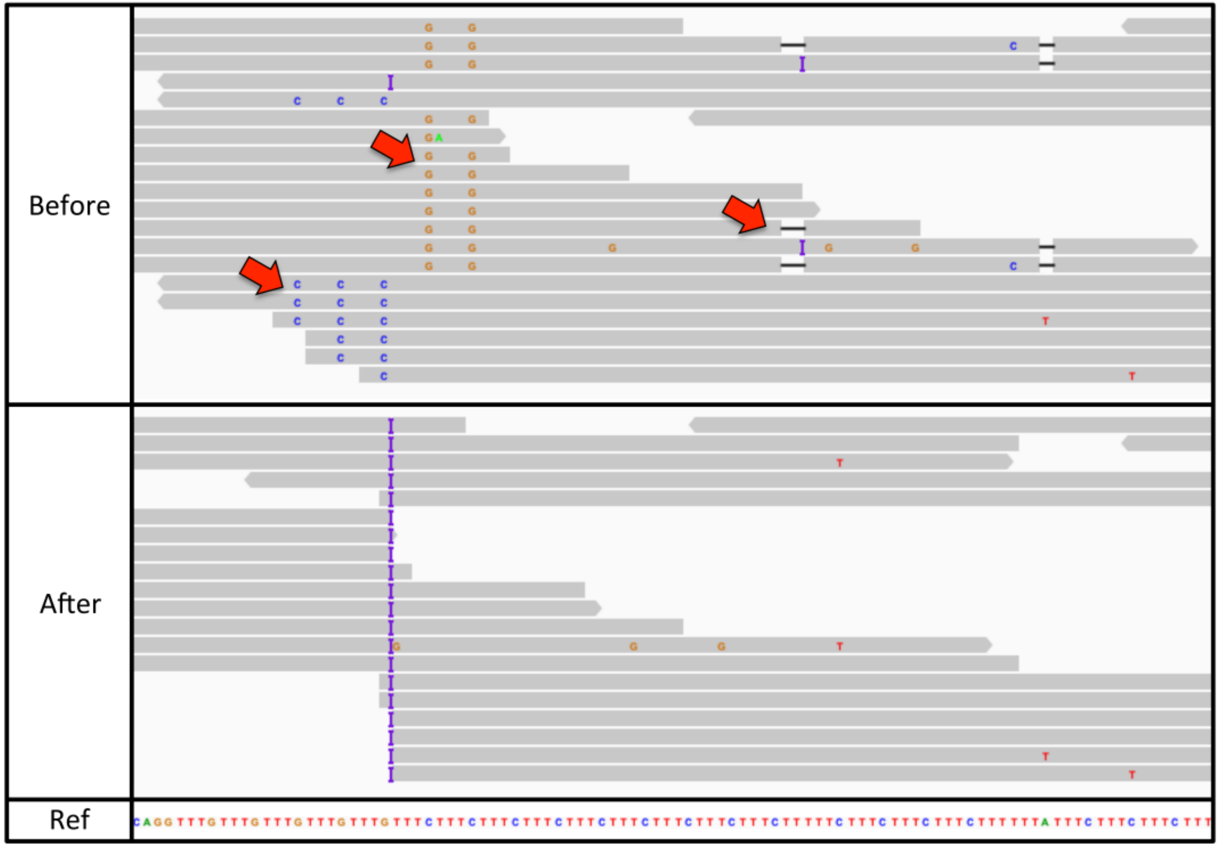
- More precise evaluation of the allelic ratio
- To compare indels between samples (same nomenclature)
- Eliminate artefactual variant due to misalignment !



Reads are subset for only those that undergo realignment.

From GATK Best Practices for Variant Discovery Presentation,
https://software.broadinstitute.org/gatk/download/workshops

Local indel realignment reduces false positive SNPs



Reads are subset for only those that undergo realignment.

From GATK Best Practices for Variant Discovery Presentation,
https://software.broadinstitute.org/gatk/download/workshops

Bam postprocessing - Indels problem

- Not always mandatory since modern callers perform that step internally :
  - Freebayes
  - Genome Analysis ToolKit (GATK, HaplotypeCaller)
  - Platypus
  - MuTect2
  - …
- But still useful for legacy tools :
  - Samtools mpileup
  - Genome Analysis ToolKit (GATK, UnifiedGenotyper)
  - MuTect1
  - ...

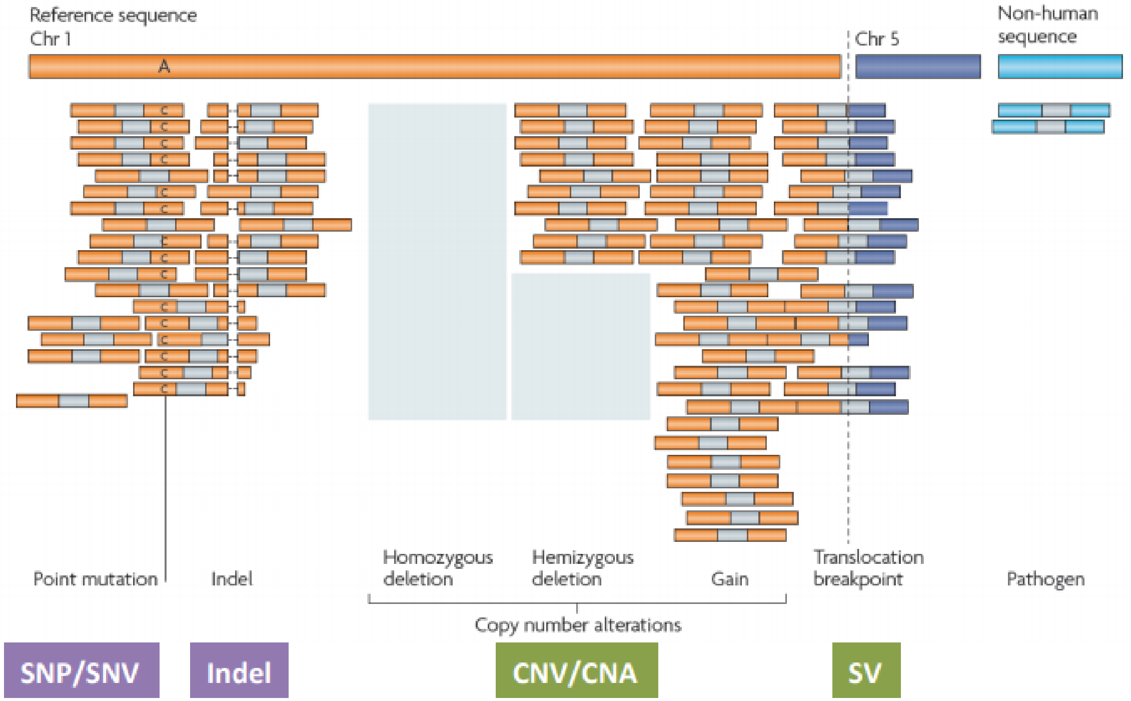
Variant Calling and Annotation

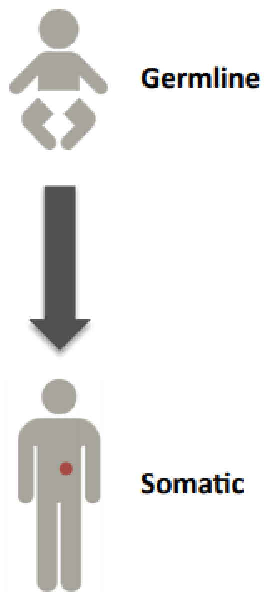
1. Bam postprocessing
2. Variant calling
3. Variant calling format
4. Variant normalization
5. Variant filtering
6. Variant annotation and prioritization

Variant calling in a diploid organism

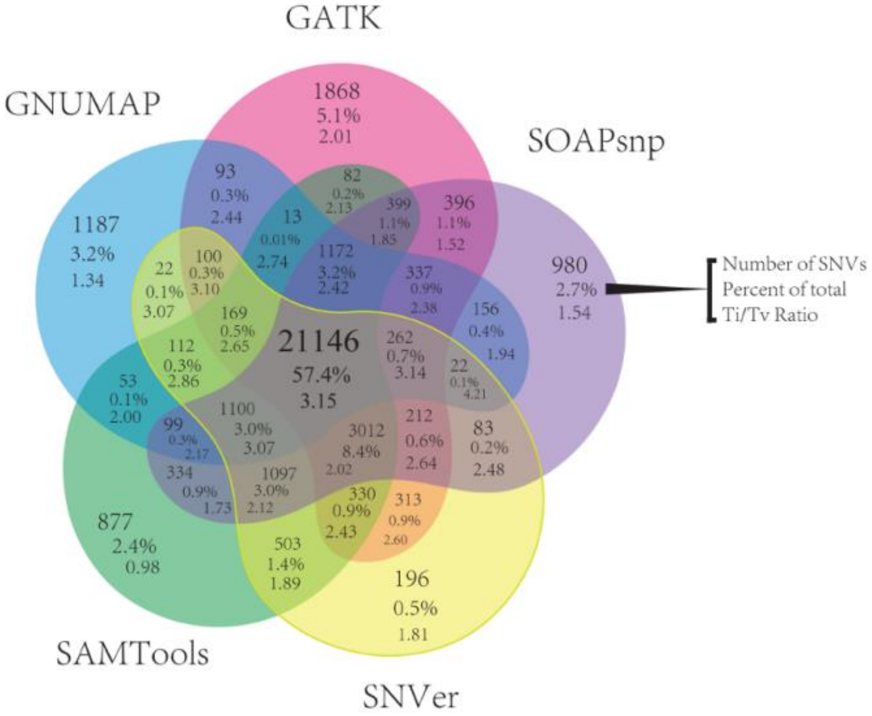
- Genetic changes relative to **a reference genome**
  - Germline (inherited)
  - Somatic (cancer)
- **Reference Genome** = a standardized genomic sequence
- Human reference sequence :
  - hg19/GRCh37/b37 (still broadly used)
  - hg38/GRCh38/b38 (current reference)
- Other organisms :
  - Many have a fully assembled reference sequence (mouse, rat, etc.)
  - Many still do not (plants, etc.)

Different types of variants





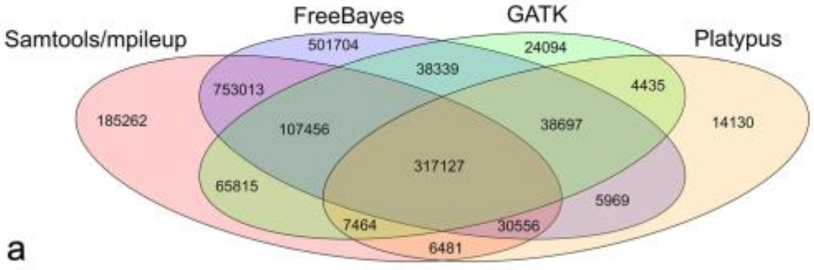
Variants callers are not concordant



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

O'Rawe et al., Genome medicine 2013

Variant callers are not concordant



Callers comparison in large plant re-sequencing (wheat)

Yao et al., BMC bioinformatics 2020

Variant callers - How do they work

Example :

```
REFERENCE: atcatgacggcaGtagcatat
--------------------------------
READ1:     atcatgacggcaGtagcatat
READ2:         tgacggcaGtagcatat
READ3:     atcatgacggcaAtagca
READ4:            cggcaGtagcatat
READ5:     atcatgacggcaGtagc
```

https://training.galaxyproject.org/training-material/topics/variant-analysis/tutorials/dip/tutorial.html

Variant callers - How do they work

Example :

```
REFERENCE: atcatgacggcaGtagcatat
--------------------------------
READ1:     atcatgacggcaGtagcatat
READ2:         tgacggcaGtagcatat
READ3:     atcatgacggcaAtagca
READ4:            cggcaGtagcatat
READ5:     atcatgacggcaGtagc
```

Naïve procedure :

- 20% A (1 read), 80% G (4 reads)
- Call site as heterozygous

BUT only one single read

Possibilities :

- A true variant
- An experimental artifact (library preparation error)
- A base calling error
- An analysis error (misalignment, etc.)

https://training.galaxyproject.org/training-material/topics/variant-analysis/tutorials/dip/tutorial.html

Variant callers - How do they work

Example :

```
REFERENCE: atcatgacggcaGtagcatat
--------------------------------
READ1:     atcatgacggcaGtagcatat
READ2:         tgacggcaGtagcatat
READ3:     atcatgacggcaAtagca
READ4:            cggcaGtagcatat
READ5:     atcatgacggcaGtagc
```

Naïve procedure :

- 20% A (1 read), 80% G (4 reads)
- Call site as heterozygous

BUT only one single read

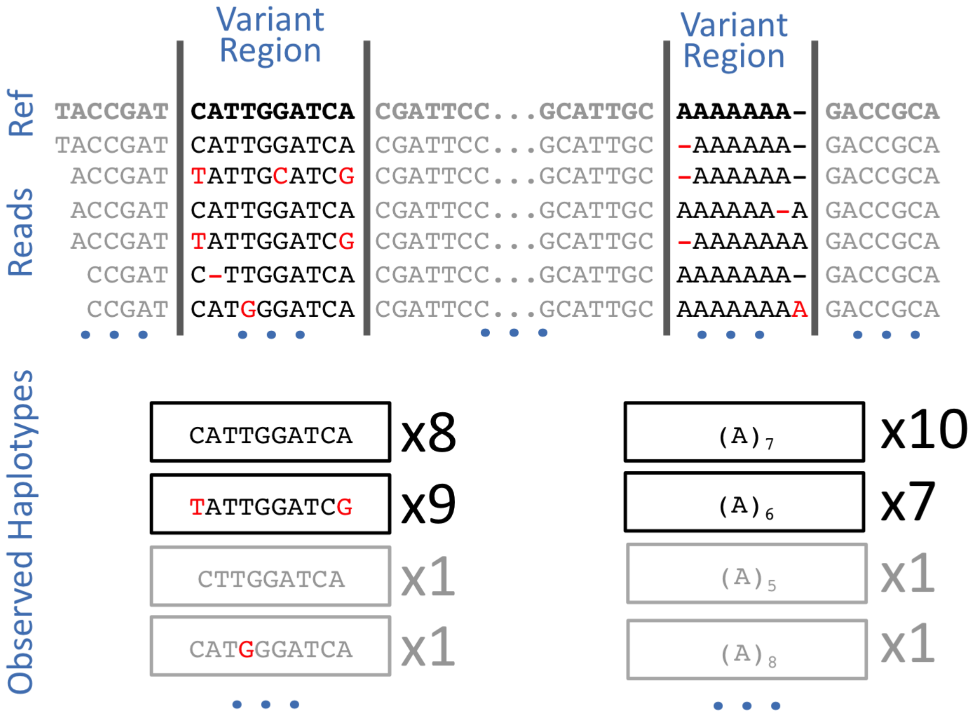
Possibilities :

- A true variant
- An experimental artifact (library preparation error)
- A base calling error
- An analysis error (misalignment, etc.)

Assign reliability estimate for genotype calls (modern callers)

https://training.galaxyproject.org/training-material/topics/variant-analysis/tutorials/dip/tutorial.html

Variant callers : Haplotype based callers



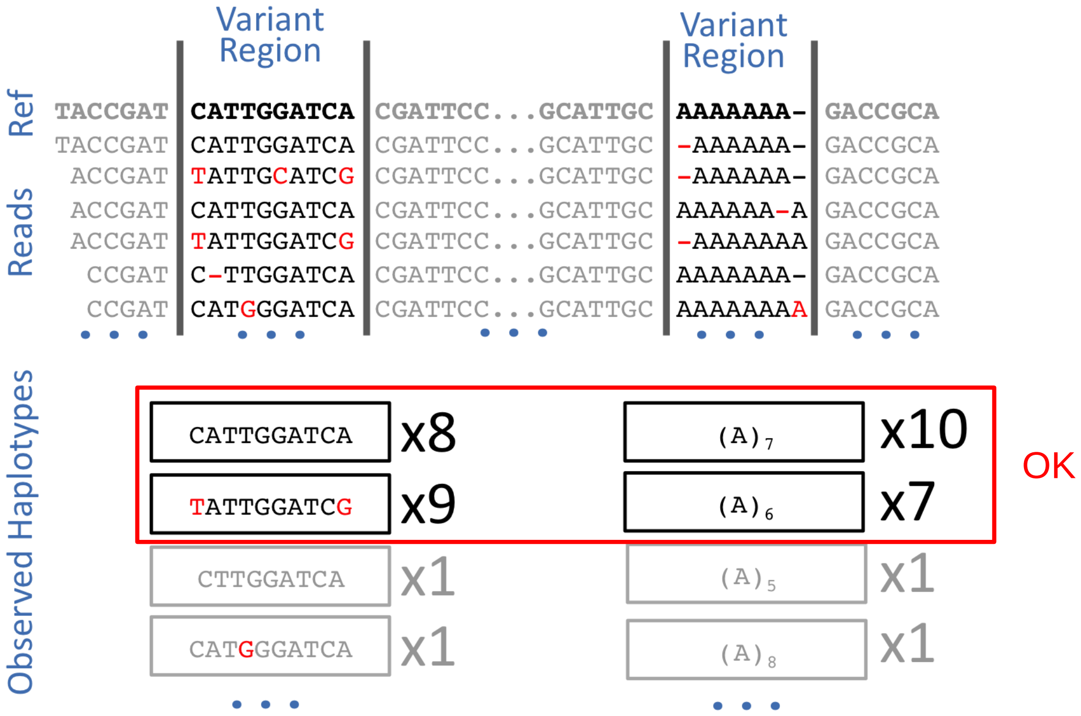
		Variant Region		Variant Region	
Ref	TACCGAT	CATTGGATCA	CGATTCC...GCATTGC	AAAAAAA-	GACCGCA
Reads	TACCGAT	CATTGGATCA	CGATTCC...GCATTGC	-AAAAAA-	GACCGCA
	ACCGAT	TATTGCATCG	CGATTCC...GCATTGC	-AAAAAA-	GACCGCA
	ACCGAT	CATTGGATCA	CGATTCC...GCATTGC	AAAAAA-A	GACCGCA
	ACCGAT	TATTGGATCG	CGATTCC...GCATTGC	-AAAAAAA	GACCGCA
	CCGAT	C-TTGGATCA	CGATTCC...GCATTGC	AAAAAAA-	GACCGCA
	CCGAT	CATGGGATCA	CGATTCC...GCATTGC	AAAAAAAA	GACCGCA
	• • •	• • •	• • •	• • •	• • •

Observed Haplotypes

CATTGGATCA	x8	$(A)_7$	x10
TATTGGATCG	x9	$(A)_6$	x7
CTTGGATCA	x1	$(A)_5$	x1
CATGGGATCA	x1	$(A)_8$	x1
• • •		• • •	

https://training.galaxyproject.org/training-material/topics/variant-analysis/tutorials/dip/tutorial.html

Variant callers : Haplotype based callers


Variant Region
Variant Region
Ref
Reads
TACCGAT CATTGGATCA CGATTCC...GCATTGC AAAAAAA- GACCGCA
TACCGAT CATTGGATCA CGATTCC...GCATTGC -AAAAAA- GACCGCA
ACCGAT TATTGCATCG CGATTCC...GCATTGC -AAAAAA- GACCGCA
ACCGAT CATTGGATCA CGATTCC...GCATTGC AAAAAA-A GACCGCA
ACCGAT TATTGGATCG CGATTCC...GCATTGC -AAAAAAA GACCGCA
CCGAT C-TTGGATCA CGATTCC...GCATTGC AAAAAAA- GACCGCA
CCGAT CATGGGATCA CGATTCC...GCATTGC AAAAAAAA GACCGCA
Observed Haplotypes
CATTGGATCA x8
TATTGGATCG x9
CTTGGATCA x1
CATGGGATCA x1
(A)7 x10
(A)6 x7
(A)5 x1
(A)8 x1
OK

https://training.galaxyproject.org/training-material/topics/variant-analysis/tutorials/dip/tutorial.html

Variant callers : Haplotype based callers

- Genome Analysis ToolKit, GATK HaplotypeCaller
- Platypus
- Freebayes
  - Indel realignment accomplished internally
  - Base recalibration is avoided
  - Variant quality recalibration is avoided
  - Ability to incorporate non-diploid case

Freebayes will be used for the variant calling in the tutorial

https://gatk.broadinstitute.org/hc/en-us
https://github.com/andyrimmer/Platypus
https://github.com/freebayes/freebayes

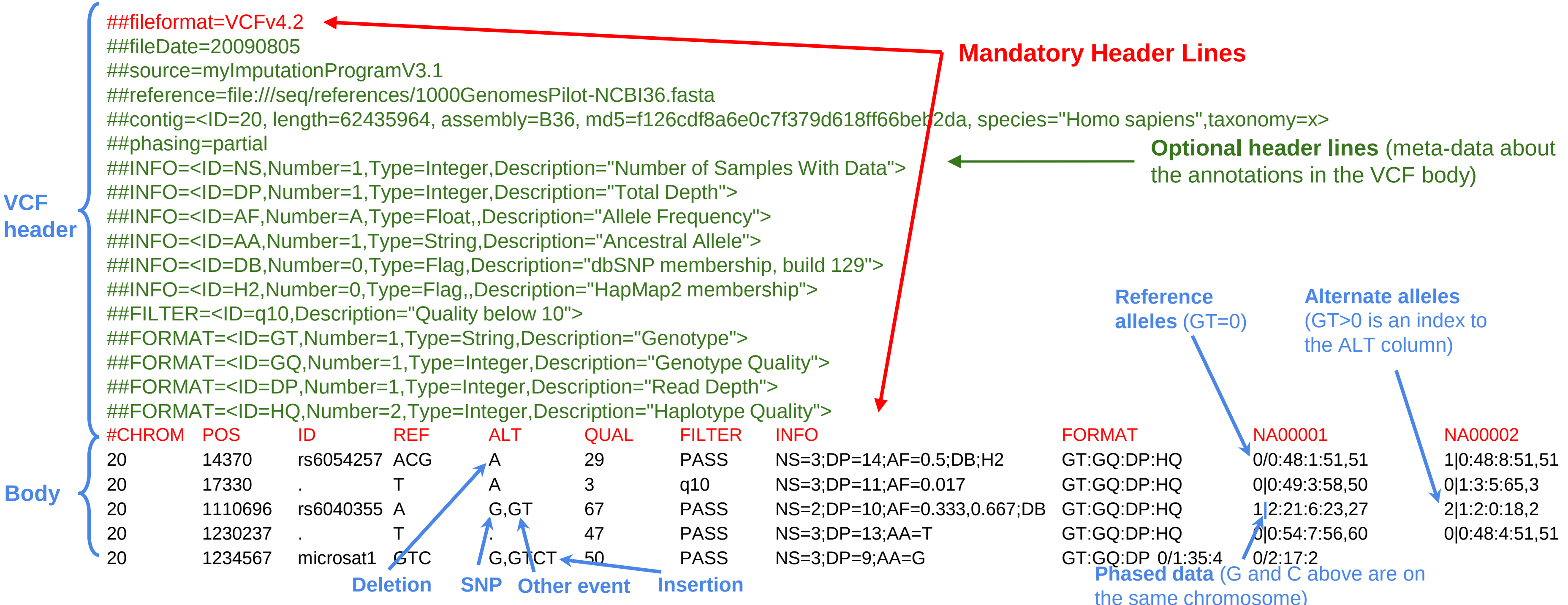
Variant Calling and Annotation

1. Bam postprocessing
2. Variant calling
3. Variant calling format
4. Variant normalization
5. Variant filtering
6. Variant annotation and prioritization

VCF : Variant Call Format

Standardised format for storing the most prevalent types of sequence variations

Text file format in 2 parts : header and body.



```
##fileformat=VCFv4.2
##fileDate=20090805
##source=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">
##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       NA00001         NA00002
20      14370    rs6054257  ACG   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
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
20      1110696  rs6040355  A     G,GT    67    PASS    NS=2;DP=10;AF=0.333,0.667;DB   GT:GQ:DP:HQ  1|2:21:6:23,27  2|1:2:0:18,2
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
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
```

VCF header

Body

Mandatory Header Lines

Optional header lines (meta-data about the annotations in the VCF body)

Reference alleles (GT=0)

Alternate alleles (GT>0 is an index to the ALT column)

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

Deletion

SNP

Other event

Insertion

VCF : Variant Call Format

Types of variants :

SNPs

Alignment	*VCF representation*		
ACGT	POS	REF	ALT
ATGT	2	C	T

Insertions

Alignment	*VCF representation*		
AC-GT	POS	REF	ALT
ACTGT	2	C	CT

Deletions

Alignment	*VCF representation*		
ACGT	POS	REF	ALT
A--T	1	ACG	A

Complex events

Alignment	*VCF representation*		
ACGT	POS	REF	ALT
A-TT	1	ACG	AT

Large structural variants

VCF representation

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

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

VCF : header

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

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

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

VCF representation of genotypes

Zygosity	VCF representation
Heterozygous	0/1, 1/2, 0/2, ...
Homozygous Reference Alternate	 0/0 1/1, 2/2, 3/3, ...
Missing	./0, ./1, ./., ...

0 = Ref   1 = Alt1   2 = Alt2   3 = Alt3   ...

VCF specification versions

VCF specifications evolve through versions!

Changes between VCFv4.1 and VCFv4.2:

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

Changes between VCFv4.2 and VCFv4.3 :

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

Variant Calling and Annotation

1. Bam postprocessing
2. Variant calling
3. Variant calling format
4. Variant normalization
5. Variant filtering
6. Variant annotation and prioritization

Variant normalization - Principles

- Every variant in the human genome has various representations !
- When merging variants from multiple variant callers for the same sample

  ⇒ which variants are common between callers ?

- When comparing variant from the same variant caller but from different samples

  ⇒ which variants are shared between samples ?

A normalized variant is parsimonious and left-aligned

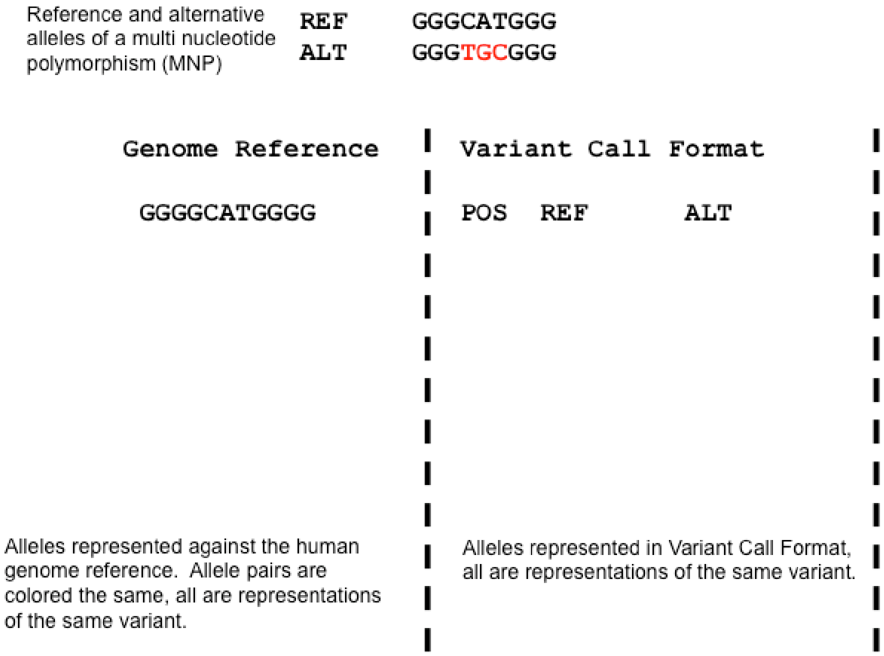
Variant normalization - Parsimony

1. Variant represented in as few nucleotides as possible without an allele of length 0 (e.i : ‘.’)

1. If the leftmost nucleotide of each variant is of the same type and the removal of the allele will not result in an empty allele (e.i : point 1.), remove superfluous nucleotide of his left side

1. The concept is symmetric (left parsimony, right parsimony)

Variant normalization - Parsimony



Reference and alternative alleles of a multi nucleotide polymorphism (MNP)

REF	GGGCATGGG
ALT	GGGTGCGGG

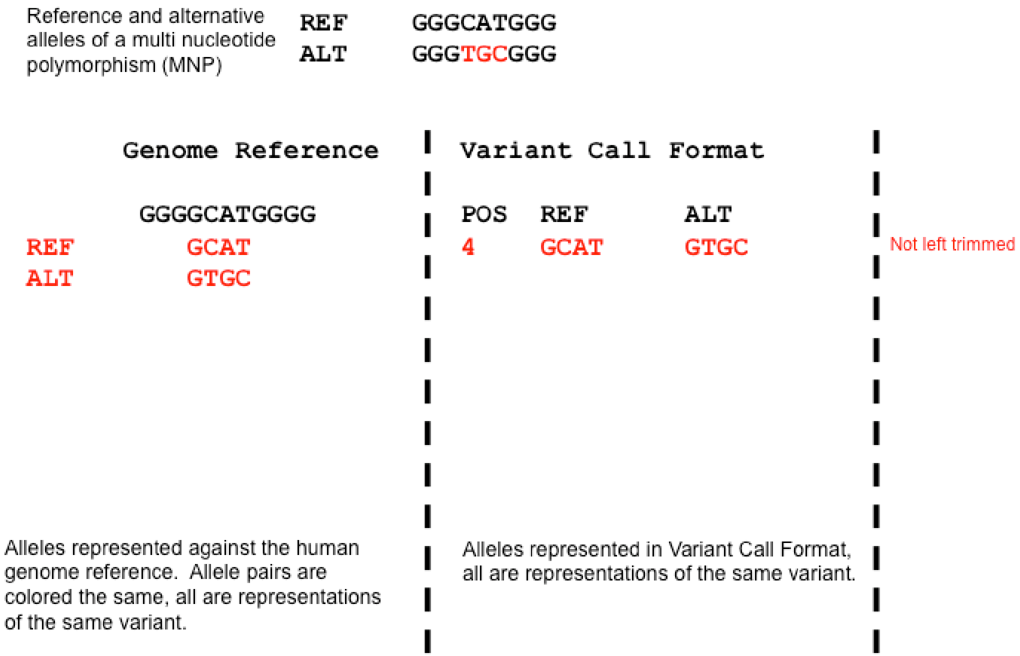
Genome Reference	Variant Call Format		
GGGGCATGGGG	POS	REF	ALT

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Parsimony



Reference and alternative alleles of a multi nucleotide polymorphism (MNP)

```
REF      GGGCATGGG
ALT      GGGTGCGGG
```

Genome Reference

```
            GGGGCATGGGG
REF             GCAT
ALT             GTGC
```

Variant Call Format

POS	REF	ALT
4	GCAT	GTGC

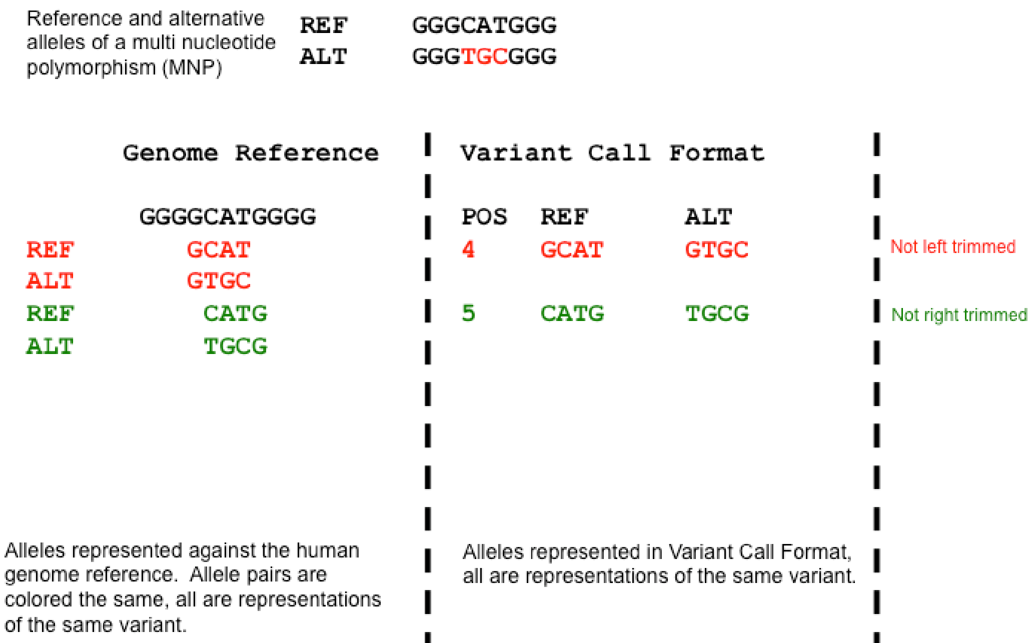
Not left trimmed

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Parsimony



Reference and alternative alleles of a multi nucleotide polymorphism (MNP)

REF	GGGCATGGG
ALT	GGGTGCGGG

Genome Reference

```
        GGGGCATGGGG
REF        GCAT
ALT        GTGC
REF         CATG
ALT         TGCG
```

Variant Call Format

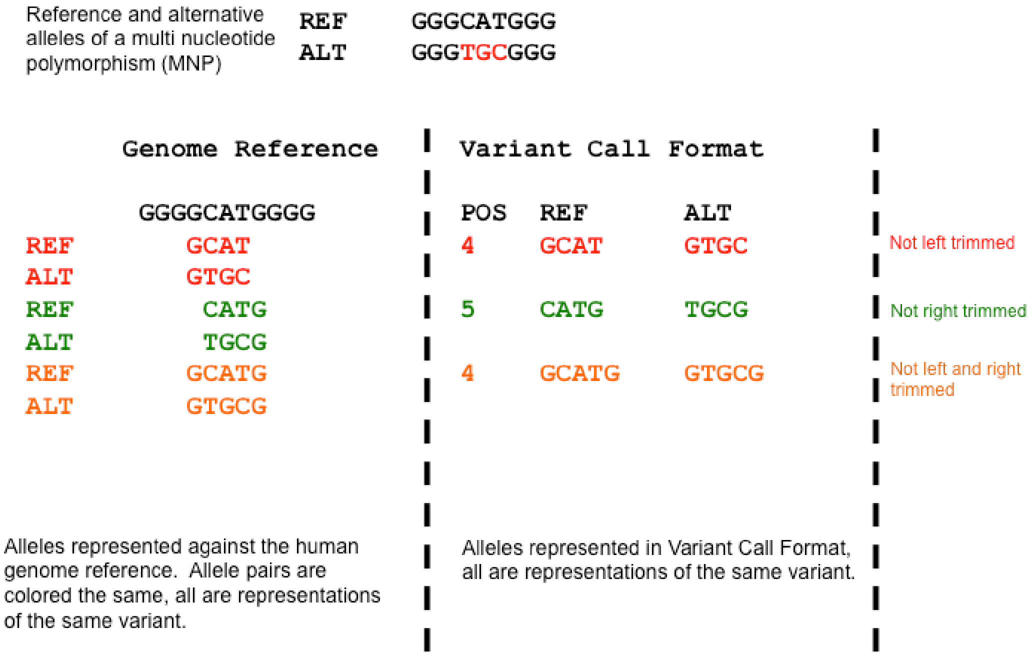
POS	REF	ALT	
4	GCAT	GTGC	Not left trimmed
5	CATG	TGCG	Not right trimmed

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Parsimony



Reference and alternative alleles of a multi nucleotide polymorphism (MNP)

```
REF     GGGCATGGG
ALT     GGGTGCGGG
```

Genome Reference

```
            GGGGCATGGGG
REF            GCAT
ALT            GTGC
REF             CATG
ALT             TGCG
REF            GCATG
ALT            GTGCG
```

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

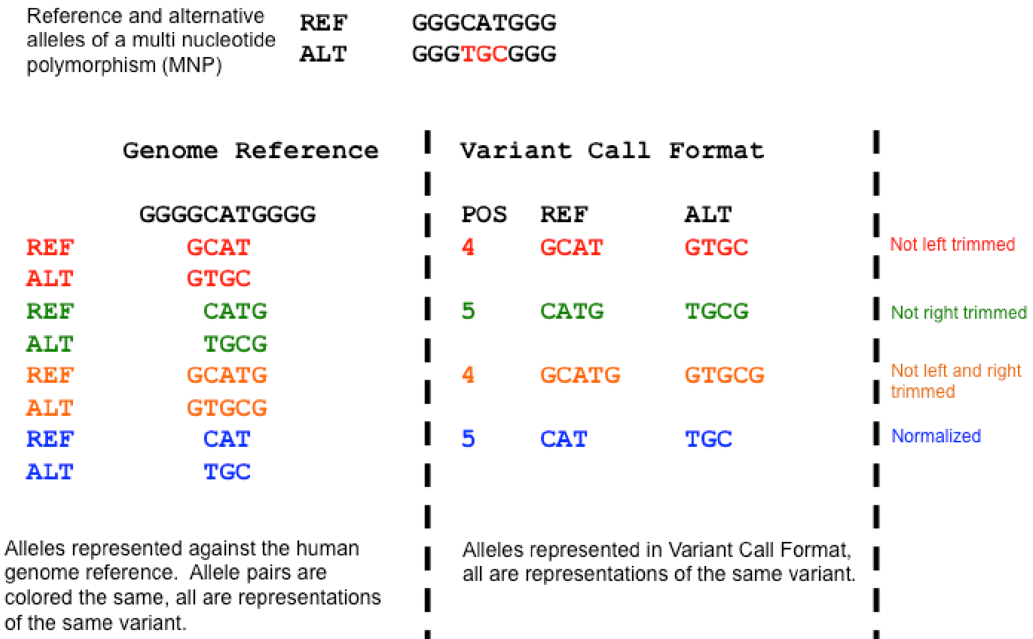
Variant Call Format

POS	REF	ALT	
4	GCAT	GTGC	Not left trimmed
5	CATG	TGCG	Not right trimmed
4	GCATG	GTGCG	Not left and right trimmed

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Parsimony



Reference and alternative alleles of a multi nucleotide polymorphism (MNP)

```
REF    GGGCATGGG
ALT    GGGTGCGGG
```

Genome Reference

```
       GGGGCATGGGG
REF       GCAT
ALT       GTGC
REF        CATG
ALT        TGCG
REF       GCATG
ALT       GTGCG
REF        CAT
ALT        TGC
```

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Variant Call Format

POS	REF	ALT	
4	GCAT	GTGC	Not left trimmed
5	CATG	TGCG	Not right trimmed
4	GCATG	GTGCG	Not left and right trimmed
5	CAT	TGC	Normalized

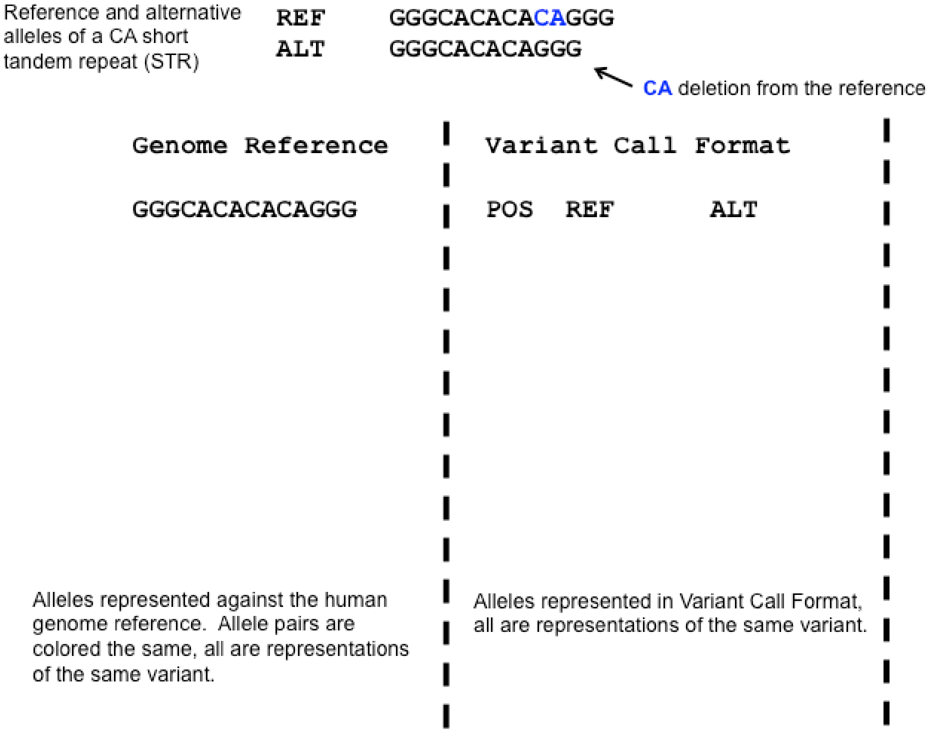
Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Left alignment

A variant is left-aligned if and only if it is no longer possible to shift its position to the left while keeping the length of all its alleles constant

Variant normalization - Left alignment



Reference and alternative alleles of a CA short tandem repeat (STR)

```
REF     GGGCACACACAGGG
ALT     GGGCACACAGGG
```

CA deletion from the reference

Genome Reference

GGGCACACACAGGG

Variant Call Format

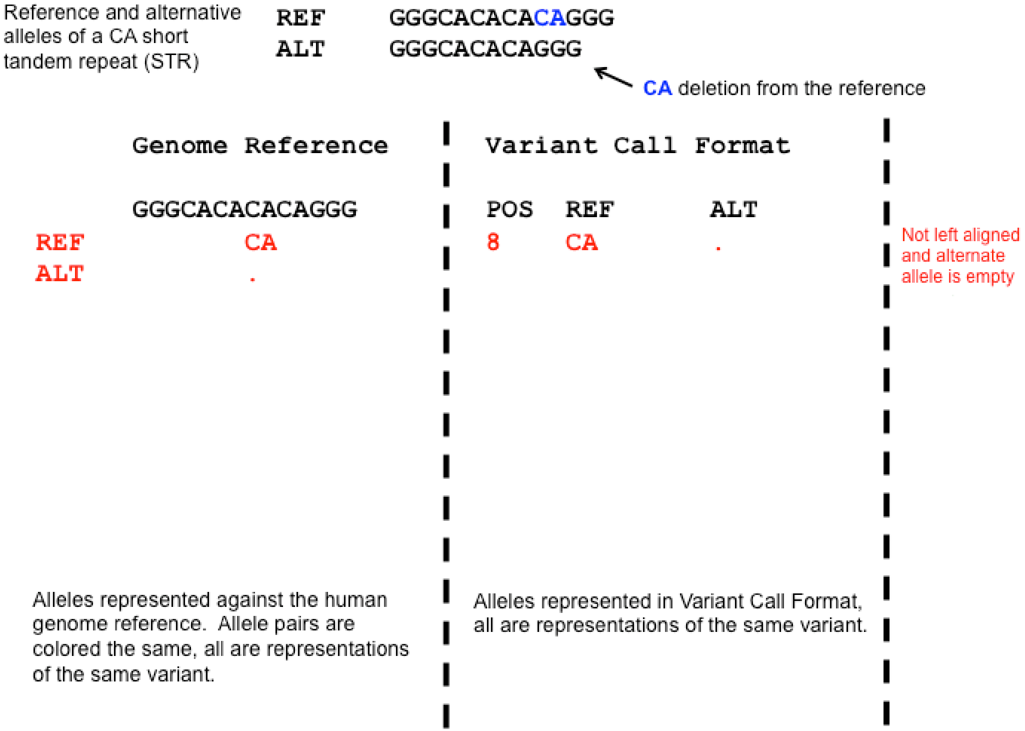
POS REF ALT

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Left alignment



Reference and alternative alleles of a CA short tandem repeat (STR)

```
REF     GGGCACACACAGGG
ALT     GGGCACACAGGG
```

CA deletion from the reference

```
        Genome Reference       | Variant Call Format
                               |
        GGGCACACACAGGG         | POS   REF      ALT
REF               CA           | 8     CA       .
ALT               .            |
```

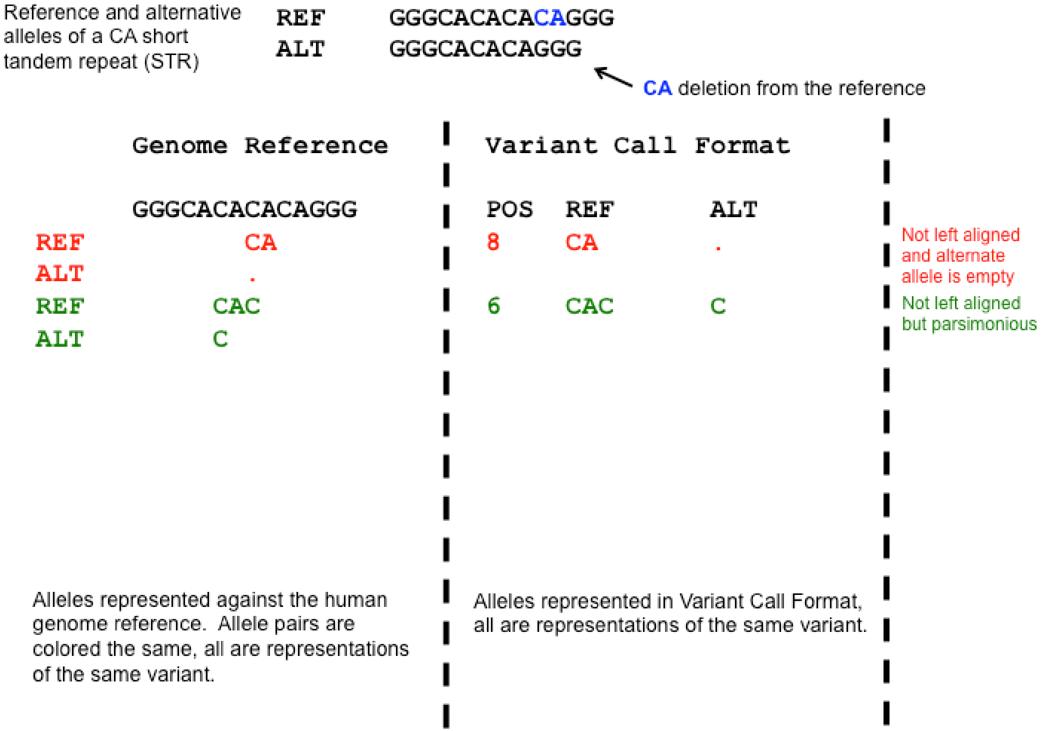
Not left aligned and alternate allele is empty

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Left alignment



Reference and alternative alleles of a CA short tandem repeat (STR)

```
REF     GGGCACACACAGGG
ALT     GGGCACACAGGG
```

CA deletion from the reference

Genome Reference

```
              GGGCACACACAGGG
REF                CA
ALT                 .
REF              CAC
ALT              C
```

Variant Call Format

POS	REF	ALT
8	CA	.
6	CAC	C

Not left aligned and alternate allele is empty

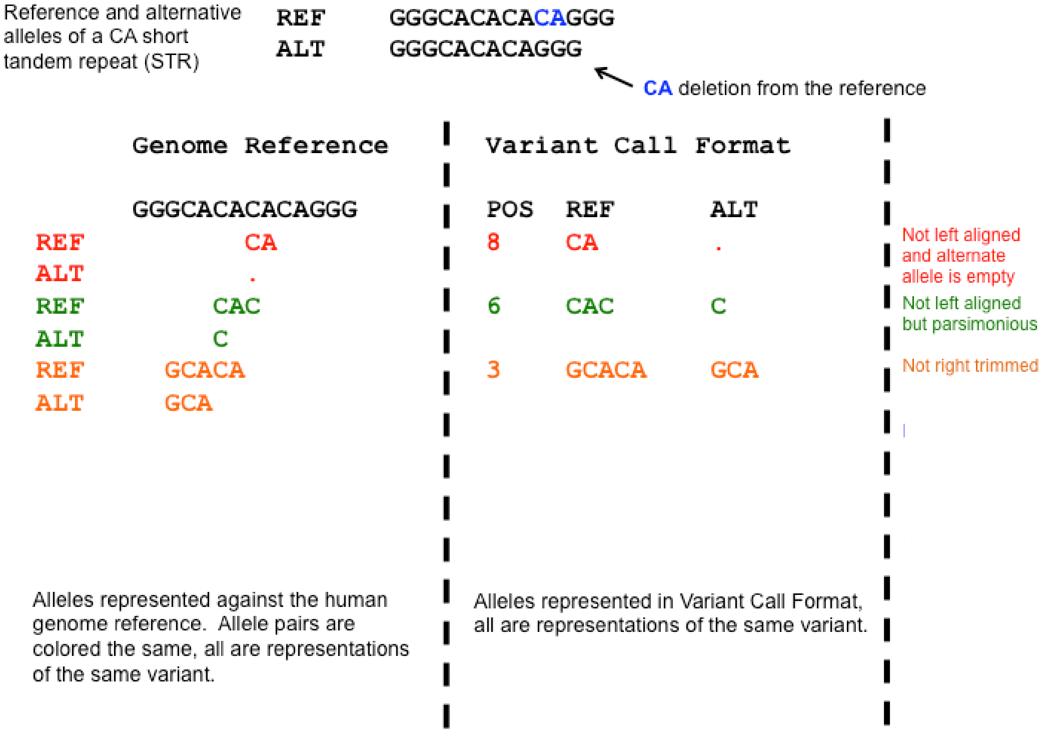
Not left aligned but parsimonious

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Left alignment



Reference and alternative alleles of a CA short tandem repeat (STR)

```
REF    GGGCACACACAGGG
ALT    GGGCACACAGGG
```

CA deletion from the reference

Genome Reference

```
      GGGCACACACAGGG
REF          CA
ALT          .
REF        CAC
ALT        C
REF     GCACA
ALT     GCA
```

Variant Call Format

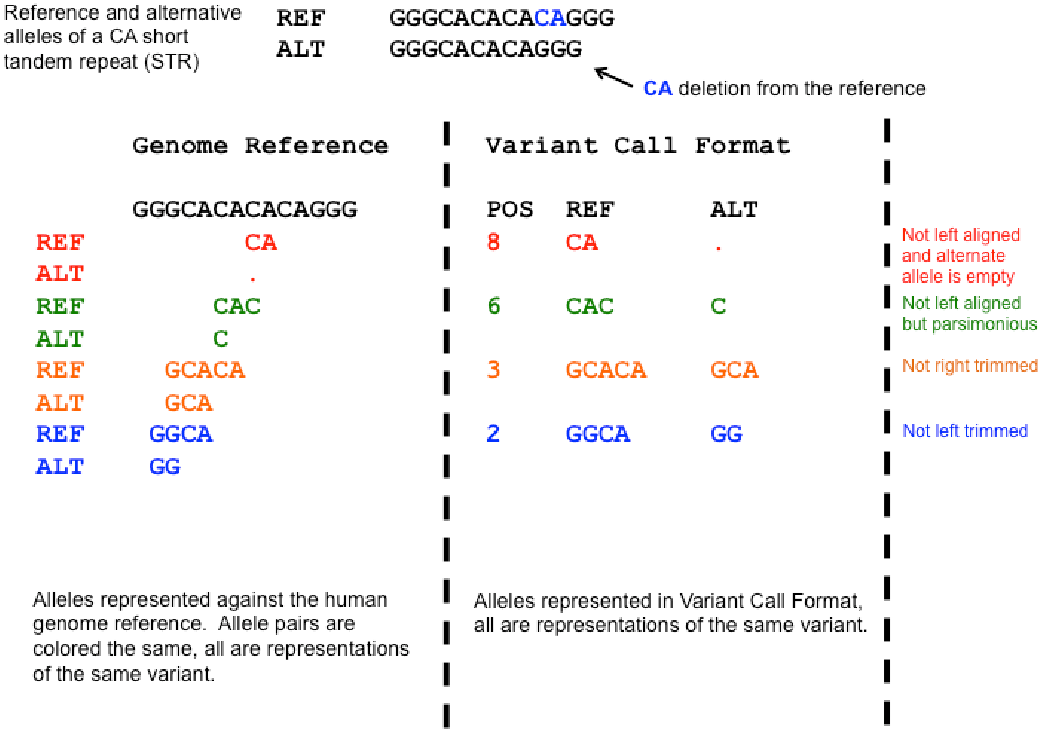
POS	REF	ALT	
8	CA	.	Not left aligned and alternate allele is empty
6	CAC	C	Not left aligned but parsimonious
3	GCACA	GCA	Not right trimmed

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Left alignment



Reference and alternative alleles of a CA short tandem repeat (STR)

```
REF     GGGCACACACAGGG
ALT     GGGCACACAGGG
```

CA deletion from the reference

Genome Reference

```
      GGGCACACACAGGG
REF          CA
ALT          .
REF        CAC
ALT        C
REF     GCACA
ALT     GCA
REF    GGCA
ALT    GG
```

Variant Call Format

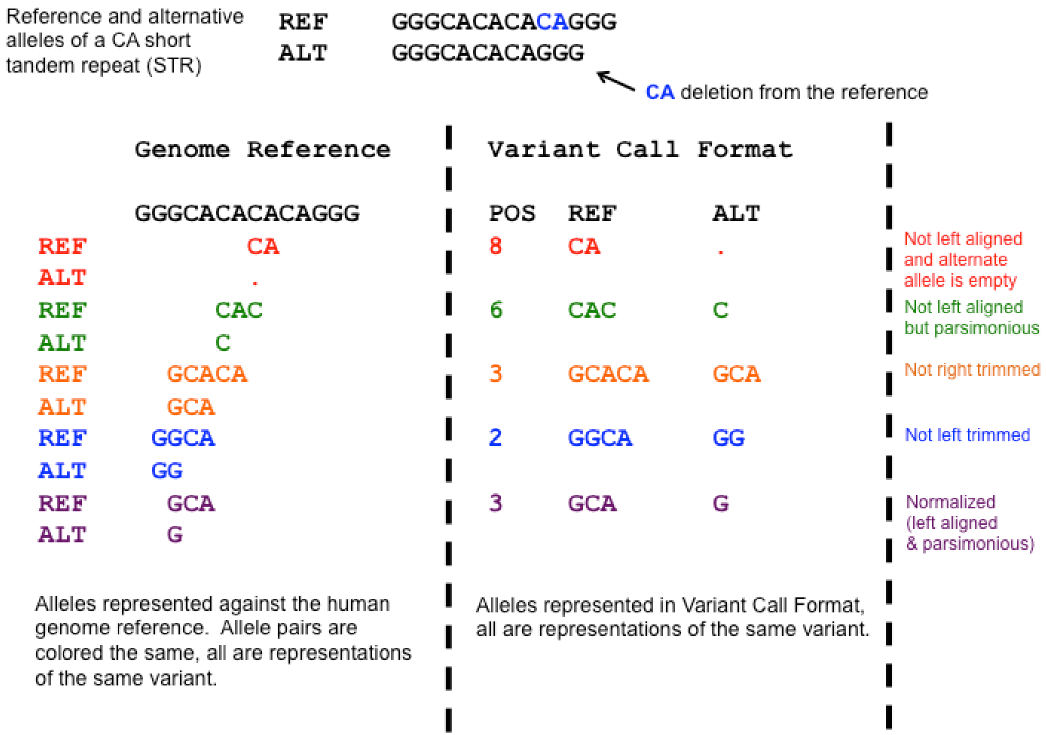
POS	REF	ALT	
8	CA	.	Not left aligned and alternate allele is empty
6	CAC	C	Not left aligned but parsimonious
3	GCACA	GCA	Not right trimmed
2	GGCA	GG	Not left trimmed

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant normalization - Left alignment



Reference and alternative alleles of a CA short tandem repeat (STR)

```
REF     GGGCACACACAGGG
ALT     GGGCACACAGGG
```

CA deletion from the reference

Genome Reference

```
           GGGCACACACAGGG
REF               CA
ALT                .
REF             CAC
ALT             C
REF          GCACA
ALT          GCA
REF         GGCA
ALT         GG
REF          GCA
ALT          G
```

Alleles represented against the human genome reference. Allele pairs are colored the same, all are representations of the same variant.

Variant Call Format

POS	REF	ALT	
8	CA	.	Not left aligned and alternate allele is empty
6	CAC	C	Not left aligned but parsimonious
3	GCACA	GCA	Not right trimmed
2	GGCA	GG	Not left trimmed
3	GCA	G	Normalized (left aligned & parsimonious)

Alleles represented in Variant Call Format, all are representations of the same variant.

https://genome.sph.umich.edu/wiki/Variant_Normalization

Variant Calling and Annotation

1. Bam postprocessing
2. Variant calling
3. Variant calling format
4. Variant normalization
5. Variant filtering
6. Variant annotation and prioritization

Variant filtering - Principles

- Calling algorithms are very permissive
- Calling sets can contain many false positives
- Callers give multiple annotations per variant (INFO field)
- Two filtering approaches :
  - Hard filtering : using thresholds on annotations
  - Variant recalibration using machine learning
- Sensitivity vs Specificity

Variant filtering - Principles

Callers’ annotations represent properties/statistics describing each variant :

- Sequence context
- Depth of coverage
- Number of reads covering each allele
- Proportion of reads in forward/reverse orientation
- ...

Variant filtering - Hard filtering

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

https://gatk.broadinstitute.org/hc/en-us/articles/360035531112?id=6925

Variant filtering - Machine learning methods

- Some callers have integrated methods to ‘score’ variants based on annotations
- Example : GATK3, Variant Quality Score Recalibration, VQSR (based on machine learning)
  - Pros :
    - Easy to perform (if integrated in software)
    - Works well in practice
  - Cons :
    - Requires DNA-seq data (not working on RNA-seq data)
    - Requires well curated training/truth resources (usually not available for non human organisms)
    - Large amount of variants (no targeted gene panels, etc.)
    - > 30 samples for WES data
- Example : GATK4, CNN (base on deep learning, still beta version)

https://gatk.broadinstitute.org/hc/en-us/articles/360035531112?id=6925

Variant Calling and Annotation

1. Bam postprocessing
2. Variant calling
3. Variant calling format
4. Variant normalization
5. Variant filtering
6. Variant annotation and prioritization

Variant annotation and prioritization

Basic idea : for each variant, add annotations to help the prioritization for further analyses (biological/bioinfomatics) :

- Frequency (reference panels)
- Genomic context (gene, regulatory region, etc.)
- Impact (missense, stop codon, splice event, etc.)
- Clinical context (known disease/phenotype association, etc.)
- Conservation across species
- Pathogenicity prediction
- ...

Annotation Databases

Eilbeck, Karen & Quinlan, Aaron & Yandell, Mark. (2017). Settling the score: variant prioritization and Mendelian disease. Nature Reviews Genetics. . 10.1038/nrg.2017.52.

Genomic data repositories

- 1000 Genomes

  The 1000 Genomes Project (abbreviated as 1KGP), launched in January 2008, was an international research effort to establish by far the most detailed catalogue of human genetic variation.

- ESP (NHLBI Exome Sequencing Project)

  Exists in 3 flavours : evs annotation data was generated from approximately 2500 exomes, evs_5400 from approximately 5400 exomes and the last one, evs_6500 from approximately 6500 exomes

- ExAC (Exome Aggregation Consortium)

  Coalition of investigators seeking to aggregate and harmonize exome sequencing data from a variety of large-scale sequencing projects, and to make summary data available for the wider scientific community.

- gnomAD (Genome Aggregation Database)

  Developed by an international coalition of investigators, with the goal of aggregating and harmonizing both exome and genome sequencing data from a wide variety of large-scale sequencing projects.

- FREX (The French Exome Project Database)

  A reference panel of exomes from French regions

Annotation Databases

Eilbeck, Karen & Quinlan, Aaron & Yandell, Mark. (2017). Settling the score: variant prioritization and Mendelian disease. Nature Reviews Genetics. . 10.1038/nrg.2017.52.

Databases of variant-disease and gene-disease associations

- ClinVar
  - ClinVar is a freely accessible, public archive of reports of the relationships among human variations and phenotypes hosted by the National Center for Biotechnology Information (NCBI) and funded by intramural National Institutes of Health (NIH) funding.

- dbSNP
  - The Single Nucleotide Polymorphism Database (dbSNP) is a free public archive for genetic variation within and across different species developed and hosted by the National Center for Biotechnology Information (NCBI) in collaboration with the National Human Genome Research Institute (NHGRI).
  - Despite the name, not only SNP
  - The quality of the data found on dbSNP has been questioned by many research groups

- Gencode
  - set of annotations including all protein-coding loci with alternatively transcribed variants, non-coding loci with transcript evidence, and pseudogenes.

Annotation Databases

Eilbeck, Karen & Quinlan, Aaron & Yandell, Mark. (2017). Settling the score: variant prioritization and Mendelian disease. Nature Reviews Genetics. . 10.1038/nrg.2017.52.

Databases of variant-disease and gene-disease associations

- HGMD Public
  - The Human Gene Mutation Database (HGMD®) represents an attempt to collate known (published) gene lesions responsible for human inherited disease.
- COSMIC
  - COSMIC (Catalogue of Somatic Mutations in Cancer) is a data resource that is designed to store and display somatic mutation information and related details and contains information relating to human cancers.
  - Data in COSMIC is curated from known Cancer Genes Literature and Systematic Screens.
- dbNSFP
  - Annotation database for non-synonymous SNPs assembled by Xiaoming Liu from the University of Texas School of Public Health (see citation below). 2 flavours : the **dbNSFP** database or **dbNSFP-light** (a version with fewer features)

HGMD®
The Human Gene Mutation Database
Cardiff

COSMIC
Catalogue Of Somatic Mutations In Cancer

dbNSFP

Genotype-Phenotype Databases

Eilbeck, Karen & Quinlan, Aaron & Yandell, Mark. (2017). Settling the score: variant prioritization and Mendelian disease. Nature Reviews Genetics. . 10.1038/nrg.2017.52.

Databases of variant-disease and gene-disease associations

- GA4GH Beacon Project
  - The Global Alliance for Genomics and Health (GA4GH) Beacon Project 108 allows researchers to search for a particular variant across a host of individual hospital and research facilities using the same interface.
- Geno$_2$MP
  - Genotype to Mendelian Phenotype is a service that houses anonymized and aggregated data that enable phenotypic querying
- MyGene2
  - Allows researchers and clinicians to identify and contact other researchers, clinicians or families who have shared both raw data and summary information about the same rare condition or candidate.
- OMIM
  - Online Mendelian Inheritance in Man. OMIM is a comprehensive, authoritative compendium of human genes and genetic phenotypes that is freely available and updated daily. The full-text, referenced overviews in OMIM contain information on all known mendelian disorders and over 15,000 genes.

Variant prioritization tools

Eilbeck, Karen & Quinlan, Aaron & Yandell, Mark. (2017). Settling the score: variant prioritization and Mendelian disease. Nature Reviews Genetics. . 10.1038/nrg.2017.52.

Conservation and pathogenicity prediction

- SIFT
  - Sorts Intolerant From Tolerant. The degree of protein sequence conservation is used to predict the impact of a missense variant
- PolyPhen2
  - Polymorphism phenotyping version 2 uses protein sequence and structure to predict the impact of a missense variant
- CADD
  - Integration of conservation metrics, functional data and scores such as SIFT and PolyPhen2 to predict the deleteriousness of nucleotide or short indel change in the genome
- GERP++
  - Measures sequence conservation in the human genome through alignments to 43 other vertebrate genome
- REVEL
  - Combination of 13 prediction tools into a single score

Just because a variant is predicted to be damaging by tools does not mean that it is pathogenic !

Variant prioritization tools

Eilbeck, Karen & Quinlan, Aaron & Yandell, Mark. (2017). Settling the score: variant prioritization and Mendelian disease. Nature Reviews Genetics. . 10.1038/nrg.2017.52.

Software/Frameworks

- Variant Effect Predictor (VEP)
  - Efficient tool from ensembl which incorporates many databases and plugins for different genomes. Provides a web interface.
- **SnpEff** / ANNOVAR
  - efficient software tool to utilize update-to-date information to functionally annotate genetic variants detected from diverse genomes
- seqr
  - an open source web interface for rare disease genomics to make research productive, accessible, and user-friendly while leveraging resources and infrastructure at the Broad Institute.
- VAAST/VAAST2/pVAAST
  - combines variant frequency data with AAS (Amino Acid Substitution) information on a feature-by-feature basis. Uses the likelihood ratio to search for damaged genes by comparing the variants in a set of disease genomes (cases) to those in a set of healthy genomes (controls).
- **GEMINI**
  - flexible framework for exploring genetic variation in the context of the wealth of genome annotations available for the human genome. Provides a simple, flexible, and powerful system for exploring genetic variation for disease and population genetics.

Ve!P

ANNOVAR

BROAD INSTITUTE

seqr

An open source software platform for rare disease genomics

VAAST

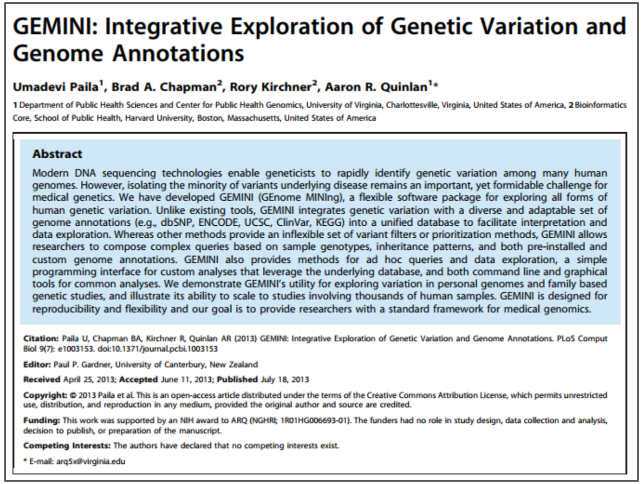
GEMINI

A FRAMEWORK FOR MINING GENOME VARIATION

GEMINI presentation

GEnome MINIng

- Software package for exploring genetic variation
- Integrates annotations from many different sources (ClinVar, dbSNP, ENCODE, UCSC, 1000 Genomes, ESP, KEGG, etc.)
- Load a VCF into an “easy to use” database
- Query (fetch data) from database based on annotations or subject genotypes
- Analyze simple genetic models
- More advanced pathway, protein-protein interaction analyses



GEMINI: Integrative Exploration of Genetic Variation and Genome Annotations

Umadevi Paila[1], Brad A. Chapman[2], Rory Kirchner[2], Aaron R. Quinlan[1]*

[1] Department of Public Health Sciences and Center for Public Health Genomics, University of Virginia, Charlottesville, Virginia, United States of America, [2] Bioinformatics Core, School of Public Health, Harvard University, Boston, Massachusetts, United States of America

Abstract

Modern DNA sequencing technologies enable geneticists to rapidly identify genetic variation among many human genomes. However, isolating the minority of variants underlying disease remains an important, yet formidable challenge for medical genetics. We have developed GEMINI (GEnome MINIng), a flexible software package for exploring all forms of human genetic variation. Unlike existing tools, GEMINI integrates genetic variation with a diverse and adaptable set of genome annotations (e.g., dbSNP, ENCODE, UCSC, ClinVar, KEGG) into a unified database to facilitate interpretation and data exploration. Whereas other methods provide an inflexible set of variant filters or prioritization methods, GEMINI allows researchers to compose complex queries based on sample genotypes, inheritance patterns, and both pre-installed and custom genome annotations. GEMINI also provides methods for ad hoc queries and data exploration, a simple programming interface for custom analyses that leverage the underlying database, and both command line and graphical tools for common analyses. We demonstrate GEMINI's utility for exploring variation in personal genomes and family based genetic studies, and illustrate its ability to scale to studies involving thousands of human samples. GEMINI is designed for reproducibility and flexibility and our goal is to provide researchers with a standard framework for medical genomics.

Citation: Paila U, Chapman BA, Kirchner R, Quinlan AR (2013) GEMINI: Integrative Exploration of Genetic Variation and Genome Annotations. PLoS Comput Biol 9(7): e1003153. doi:10.1371/journal.pcbi.1003153

Editor: Paul P. Gardner, University of Canterbury, New Zealand

Received April 25, 2013; **Accepted** June 11, 2013; **Published** July 18, 2013

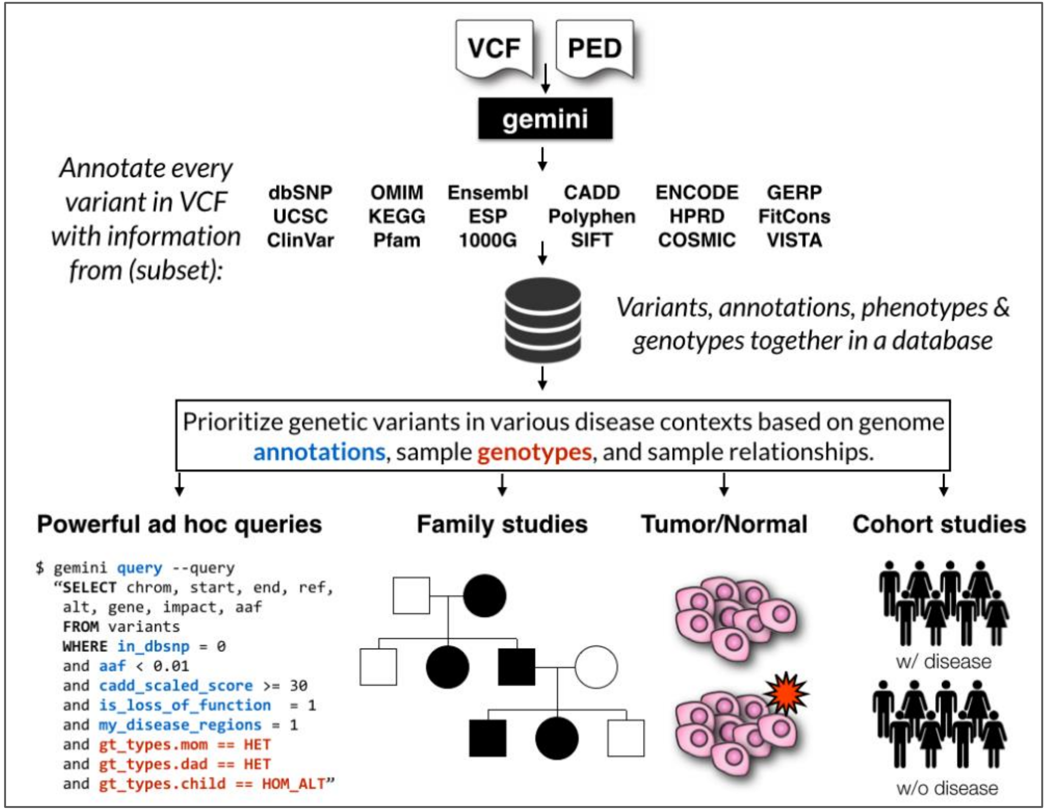
Funding: This work was supported by an NIH award to ARQ (NGHRI; 1R01HG006693-01). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: arq5x@virginia.edu

Paila U, Chapman BA, Kirchner R, Quinlan AR (2013)
GEMINI: Integrative Exploration of Genetic Variation and Genome Annotations.
PLoS Comput Biol 9(7): e1003153. doi:10.1371/journal.pcbi.1003153

GEMINI presentation



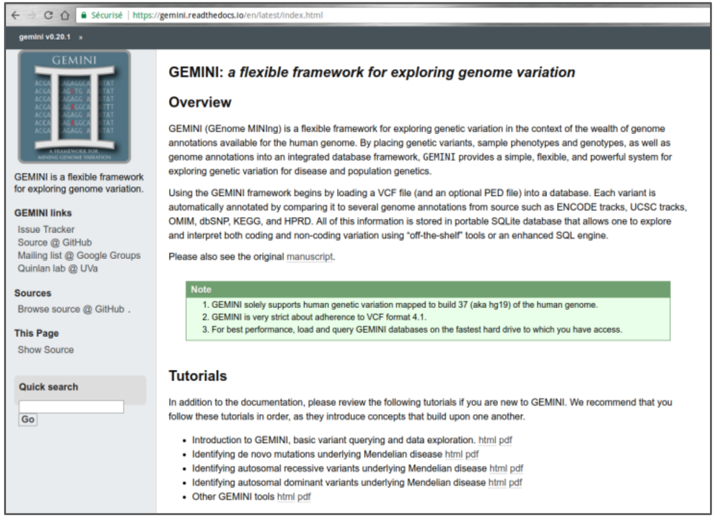
Annotate every variant in VCF with information from (subset):

Variants, annotations, phenotypes & genotypes together in a database

Prioritize genetic variants in various disease contexts based on genome annotations, sample genotypes, and sample relationships.

```
$ gemini query --query
  “SELECT chrom, start, end, ref,
   alt, gene, impact, aaf
   FROM variants
   WHERE in_dbsnp = 0
   and aaf < 0.01
   and cadd_scaled_score >= 30
   and is_loss_of_function  = 1
   and my_disease_regions = 1
   and gt_types.mom == HET
   and gt_types.dad == HET
   and gt_types.child == HOM_ALT”
```

Documentation :
http://gemini.readthedocs.io
Tutorials :
https://speakerdeck.com/arq5x/



GEMINI is a flexible framework for exploring genome variation.

GEMINI links

Issue Tracker
Source @ GitHub
Mailing list @ Google Groups
Quinlan lab @ UVa

Sources

Browse source @ GitHub .

This Page

Show Source

Quick search

GEMINI: *a flexible framework for exploring genome variation*

Overview

GEMINI (GEnome MINIng) is a flexible framework for exploring genetic variation in the context of the wealth of genome annotations available for the human genome. By placing genetic variants, sample phenotypes and genotypes, as well as genome annotations into an integrated database framework, GEMINI provides a simple, flexible, and powerful system for exploring genetic variation for disease and population genetics.

Using the GEMINI framework begins by loading a VCF file (and an optional PED file) into a database. Each variant is automatically annotated by comparing it to several genome annotations from source such as ENCODE tracks, UCSC tracks, OMIM, dbSNP, KEGG, and HPRD. All of this information is stored in portable SQLite database that allows one to explore and interpret both coding and non-coding variation using “off-the-shelf” tools or an enhanced SQL engine.

Please also see the original manuscript.

Note

1. GEMINI solely supports human genetic variation mapped to build 37 (aka hg19) of the human genome.
2. GEMINI is very strict about adherence to VCF format 4.1.
3. For best performance, load and query GEMINI databases on the fastest hard drive to which you have access.

Tutorials

In addition to the documentation, please review the following tutorials if you are new to GEMINI. We recommend that you follow these tutorials in order, as they introduce concepts that build upon one another.

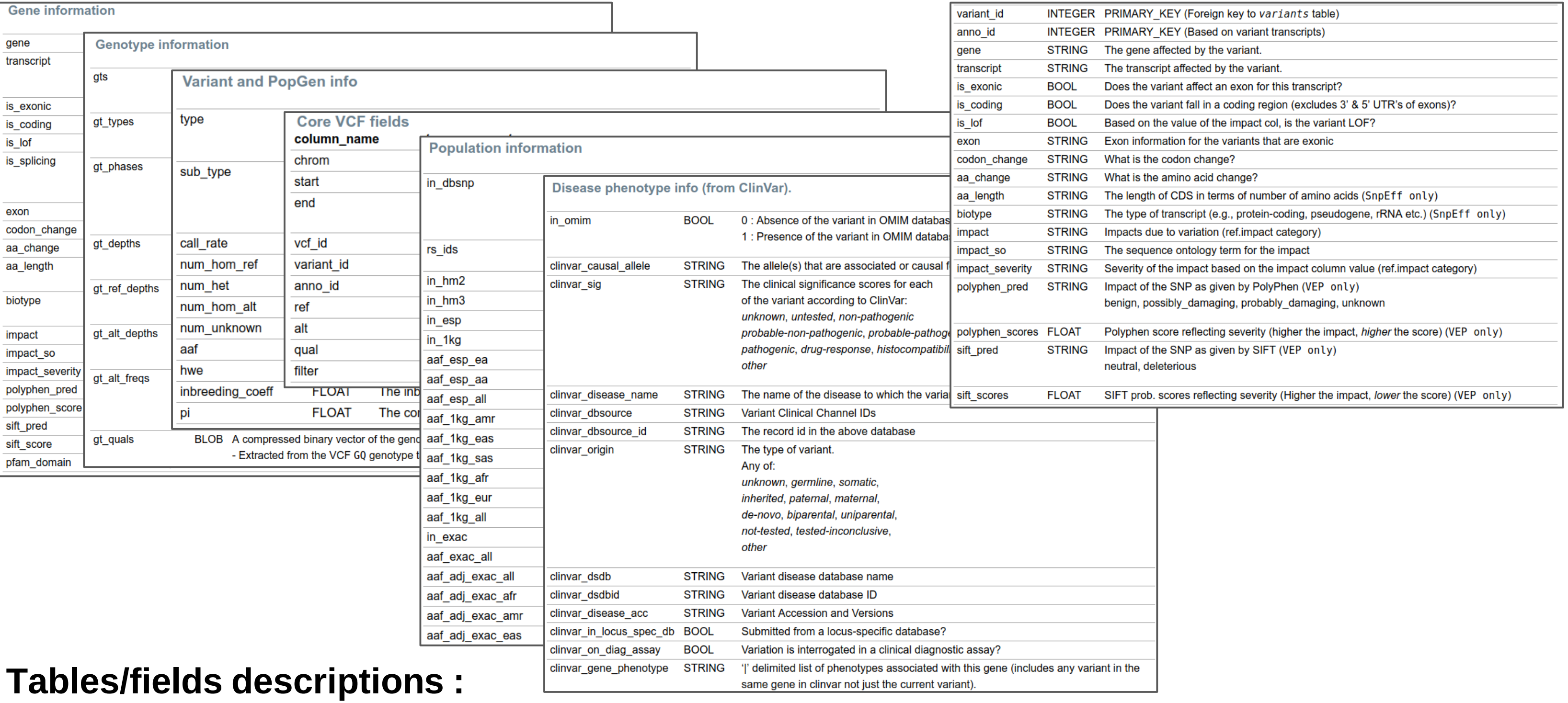
- Introduction to GEMINI, basic variant querying and data exploration. html pdf
- Identifying de novo mutations underlying Mendelian disease html pdf
- Identifying autosomal recessive variants underlying Mendelian disease html pdf
- Identifying autosomal dominant variants underlying Mendelian disease html pdf
- Other GEMINI tools html pdf

GEMINI database overview

The variants table

The variant_impacts table

Etc.



Gene information

gene
transcript
is_exonic
is_coding
is_lof
is_splicing
exon
codon_change
aa_change
aa_length
biotype
impact
impact_so
impact_severity
polyphen_pred
polyphen_score
sift_pred
sift_score
pfam_domain

Genotype information

gts		
gt_types		
gt_phases		
gt_depths		
gt_ref_depths		
gt_alt_depths		
gt_alt_freqs		
gt_quals	BLOB	A compressed binary vector of the geno - Extracted from the VCF GQ genotype t

Variant and PopGen info

type		
sub_type		
call_rate		
num_hom_ref		
num_het		
num_hom_alt		
num_unknown		
aaf		
hwe		
inbreeding_coeff	FLOAT	The inb
pi	FLOAT	The co

Core VCF fields

column_name
chrom
start
end
vcf_id
variant_id
anno_id
ref
alt
qual
filter

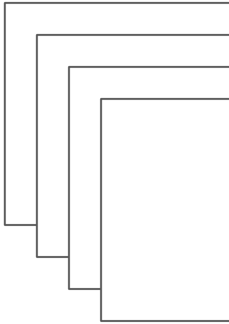
Population information

in_dbsnp
rs_ids
in_hm2
in_hm3
in_esp
in_1kg
aaf_esp_ea
aaf_esp_aa
aaf_esp_all
aaf_1kg_amr
aaf_1kg_eas
aaf_1kg_sas
aaf_1kg_afr
aaf_1kg_eur
aaf_1kg_all
in_exac
aaf_exac_all
aaf_adj_exac_all
aaf_adj_exac_afr
aaf_adj_exac_amr
aaf_adj_exac_eas

Disease phenotype info (from ClinVar).

in_omim	BOOL	0 : Absence of the variant in OMIM databas 1 : Presence of the variant in OMIM databa
clinvar_causal_allele	STRING	The allele(s) that are associated or causal f
clinvar_sig	STRING	The clinical significance scores for each of the variant according to ClinVar: *unknown*, *untested*, *non-pathogenic* *probable-non-pathogenic*, *probable-pathog* *pathogenic*, *drug-response*, *histocompatibil* *other*
clinvar_disease_name	STRING	The name of the disease to which the varia
clinvar_dbsource	STRING	Variant Clinical Channel IDs
clinvar_dbsource_id	STRING	The record id in the above database
clinvar_origin	STRING	The type of variant. Any of: *unknown*, *germline*, *somatic*, *inherited*, *paternal*, *maternal*, *de-novo*, *biparental*, *uniparental*, *not-tested*, *tested-inconclusive*, *other*
clinvar_dsdb	STRING	Variant disease database name
clinvar_dsdbid	STRING	Variant disease database ID
clinvar_disease_acc	STRING	Variant Accession and Versions
clinvar_in_locus_spec_db	BOOL	Submitted from a locus-specific database?
clinvar_on_diag_assay	BOOL	Variation is interrogated in a clinical diagnostic assay?
clinvar_gene_phenotype	STRING	'|' delimited list of phenotypes associated with this gene (includes any variant in the same gene in clinvar not just the current variant).

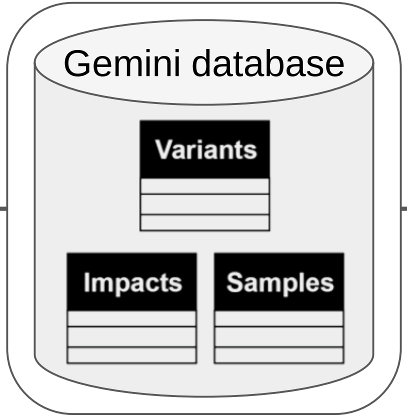
variant_id	INTEGER	PRIMARY_KEY (Foreign key to `variants` table)
anno_id	INTEGER	PRIMARY_KEY (Based on variant transcripts)
gene	STRING	The gene affected by the variant.
transcript	STRING	The transcript affected by the variant.
is_exonic	BOOL	Does the variant affect an exon for this transcript?
is_coding	BOOL	Does the variant fall in a coding region (excludes 3' & 5' UTR's of exons)?
is_lof	BOOL	Based on the value of the impact col, is the variant LOF?
exon	STRING	Exon information for the variants that are exonic
codon_change	STRING	What is the codon change?
aa_change	STRING	What is the amino acid change?
aa_length	STRING	The length of CDS in terms of number of amino acids (`SnpEff only`)
biotype	STRING	The type of transcript (e.g., protein-coding, pseudogene, rRNA etc.) (`SnpEff only`)
impact	STRING	Impacts due to variation (ref.impact category)
impact_so	STRING	The sequence ontology term for the impact
impact_severity	STRING	Severity of the impact based on the impact column value (ref.impact category)
polyphen_pred	STRING	Impact of the SNP as given by PolyPhen (`VEP only`) benign, possibly_damaging, probably_damaging, unknown
polyphen_scores	FLOAT	Polyphen score reflecting severity (higher the impact, *higher* the score) (`VEP only`)
sift_pred	STRING	Impact of the SNP as given by SIFT (`VEP only`) neutral, deleterious
sift_scores	FLOAT	SIFT prob. scores reflecting severity (Higher the impact, *lower* the score) (`VEP only`)



Tables/fields descriptions :

http://gemini.readthedocs.io/en/latest/content/database_schema.html

How to use GEMINI



ad hoc data exploration

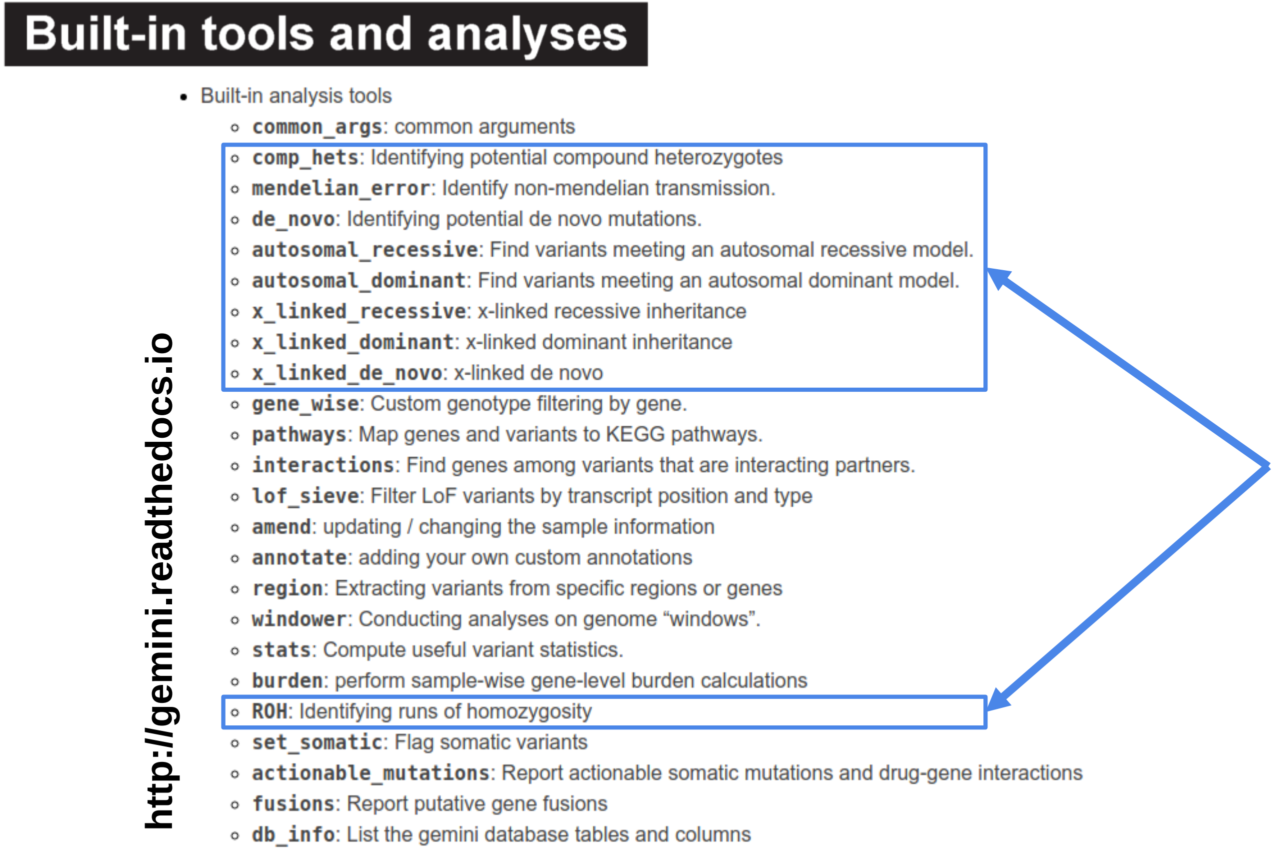
```
gemini query
--query
“select chrom, start, end,
      ref, alt, gene,
      impact, aaf, gts.proband
 from variants
 where in_dbsnp = 0
 and aaf < 0.01
 and is_lof  = 1
 and my_disease_regions = 1”
--gt-filter
“gt_types.mom == HET
 and
 gt_types.dad == HET
 and
 gt_types.proband == HOM_ALT”
```

Built-in tools and analyses

gemini

Tool	*Description*
region	*extract variants from specific genomic intervals or genes*
stats	*compute variant statisics (SFS, Ts/Tv, counts, etc.)*
annotate	*add new columns based on custom annotations*
windower	*compute variant statistics across genome “windows”*
comp_hets	*identify candidate compund heterozygotes*
pathways	*maps genes and variants to KEGG pathways*
lof_sieve	*prioritize candidate loss-of-function variants*
interact	*find protein interactions for genes/variants/samples*
auto_rec	*identify variants meeting an autosomal recessive model*
auto_dom	*identify variants meeting an autosomal dominant model*
de_novo	*identify candidate de novo mutations*
browser	*launch the interactive gemini web browser interface*

GEMINI usages



Built-in tools and analyses

- Built-in analysis tools
  - **common_args**: common arguments
  - **comp_hets**: Identifying potential compound heterozygotes
  - **mendelian_error**: Identify non-mendelian transmission.
  - **de_novo**: Identifying potential de novo mutations.
  - **autosomal_recessive**: Find variants meeting an autosomal recessive model.
  - **autosomal_dominant**: Find variants meeting an autosomal dominant model.
  - **x_linked_recessive**: x-linked recessive inheritance
  - **x_linked_dominant**: x-linked dominant inheritance
  - **x_linked_de_novo**: x-linked de novo
  - **gene_wise**: Custom genotype filtering by gene.
  - **pathways**: Map genes and variants to KEGG pathways.
  - **interactions**: Find genes among variants that are interacting partners.
  - **lof_sieve**: Filter LoF variants by transcript position and type
  - **amend**: updating / changing the sample information
  - **annotate**: adding your own custom annotations
  - **region**: Extracting variants from specific regions or genes
  - **windower**: Conducting analyses on genome “windows”.
  - **stats**: Compute useful variant statistics.
  - **burden**: perform sample-wise gene-level burden calculations
  - **ROH**: Identifying runs of homozygosity
  - **set_somatic**: Flag somatic variants
  - **actionable_mutations**: Report actionable somatic mutations and drug-gene interactions
  - **fusions**: Report putative gene fusions
  - **db_info**: List the gemini database tables and columns

http://gemini.readthedocs.io

```
gemini comp_hets
gemini mendelian_error
gemini denovo
gemini autosomal_recessive
gemini autosomal_dominant
gemini ROH
```

inheritance tools