THE UNIVERSITY OF
MELBOURNE

Melbourne Bioinformatics
BIOINFORMATICS + DATA SERVICES + INFRASTRUCTURE, FOR LIFE SCIENCES TODAY

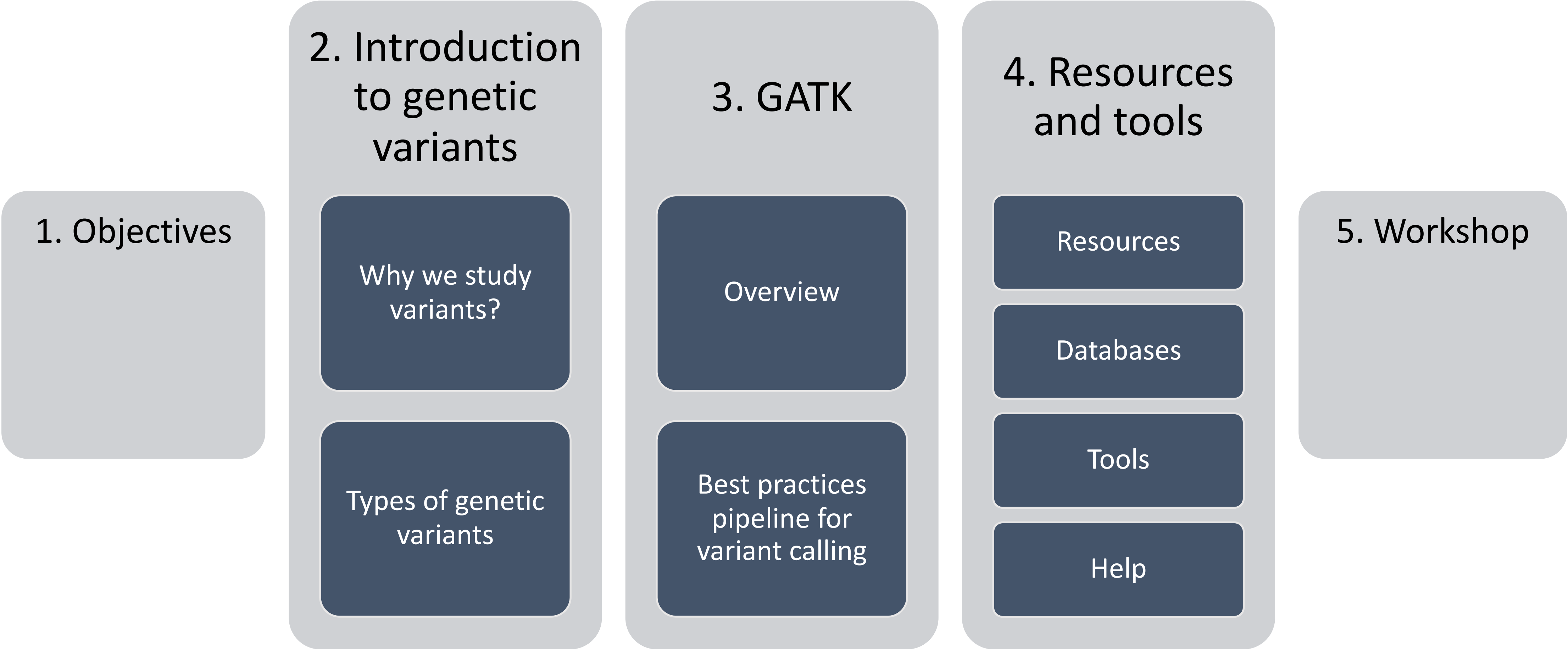
Variant calling using GATK

Khalid Mahmood

August 28, 2024

https://www.melbournebioinformatics.org.au/tutorials/tutorials/variant_calling_gatk1/variant_calling_gatk1/

Workshop overview



1. Objectives

2. Introduction to genetic variants
- Why we study variants?
- Types of genetic variants

3. GATK
- Overview
- Best practices pipeline for variant calling

4. Resources and tools
- Resources
- Databases
- Tools
- Help

5. Workshop

1. Objectives

- We aim to cover:
  - Perform QC of sequencing data
  - Align raw reads to reference sequences
  - Perform alignment metric and generating a QC report
  - Prepare alignment data for variant calling
  - Identify simple variants using GATK HaplotypeCaller
  - Visualise simple variant data (VCF files)
  - Perform basic variant filtering

2. Introduction to genetic variants

- There are approximately 3 billion base pairs in the human genome.
- Humans share 99.5% of DNA with other humans.
- A **variant** is a difference between similar genomes.
- In most cases – this means a difference between DNA sequences compared to a **reference genome**.
- In this context - a variant is described by its location (genomic coordinates) and genetic change.

e.g. chr2 9834 A→G

2. Introduction to genetic variants

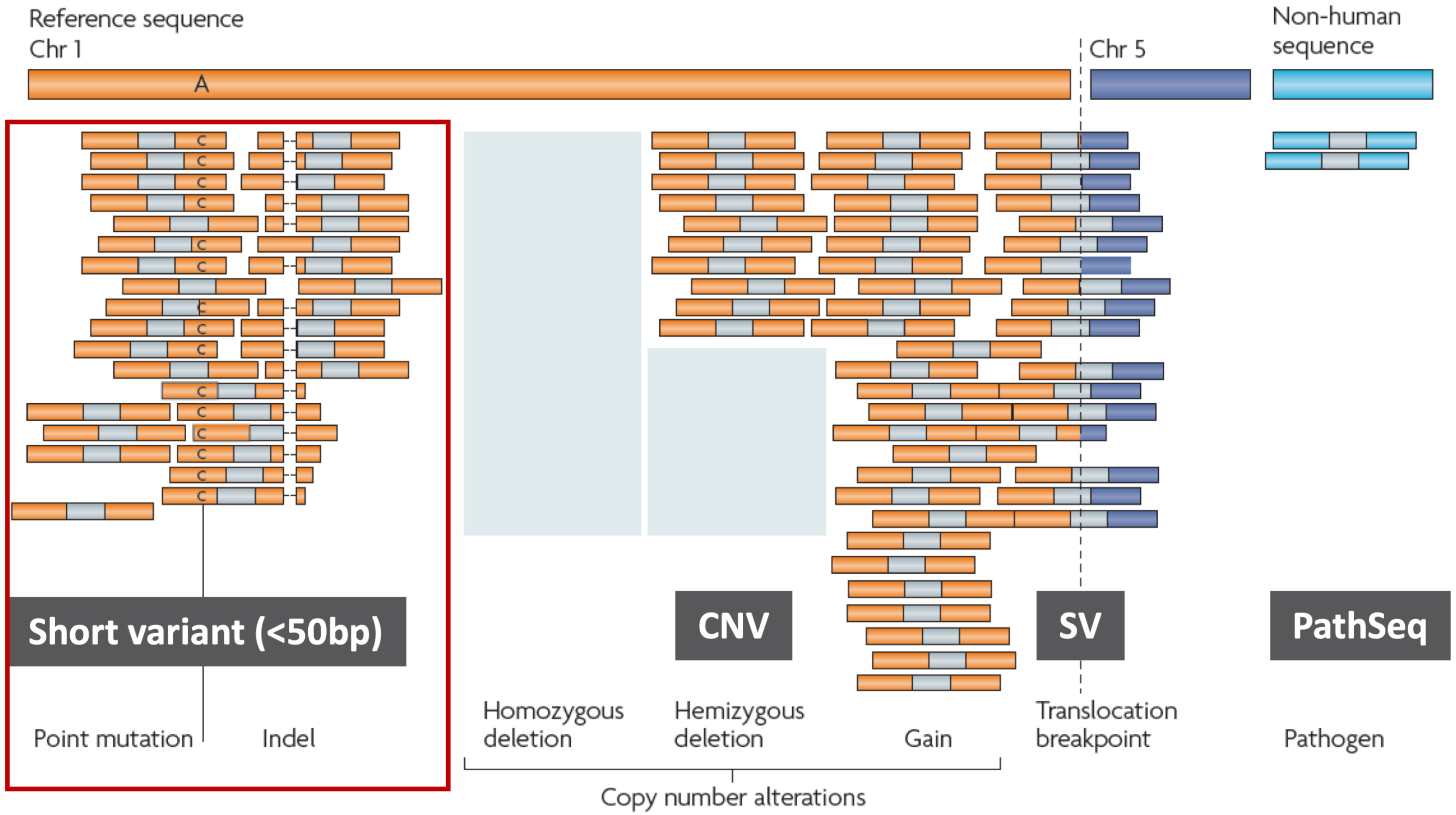
There is high degree of similarity but the human genome is large ~3 billion nucleotides.

This results in approximately 4-5 million variants between any individual and the reference genome.

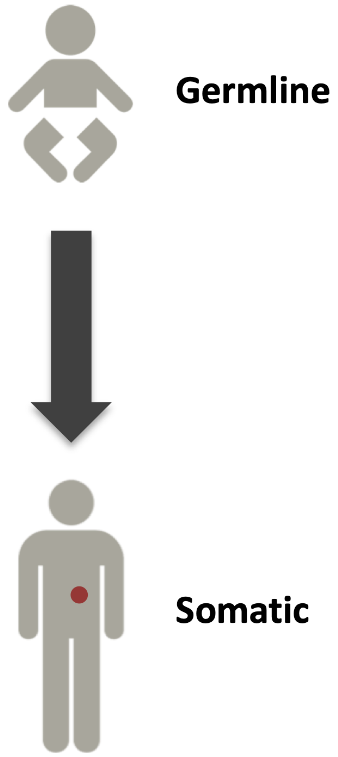
These, seemingly small number of variations likely explains a significant proportion of phenotypic diversity among humans.

2. Types of genetic variants

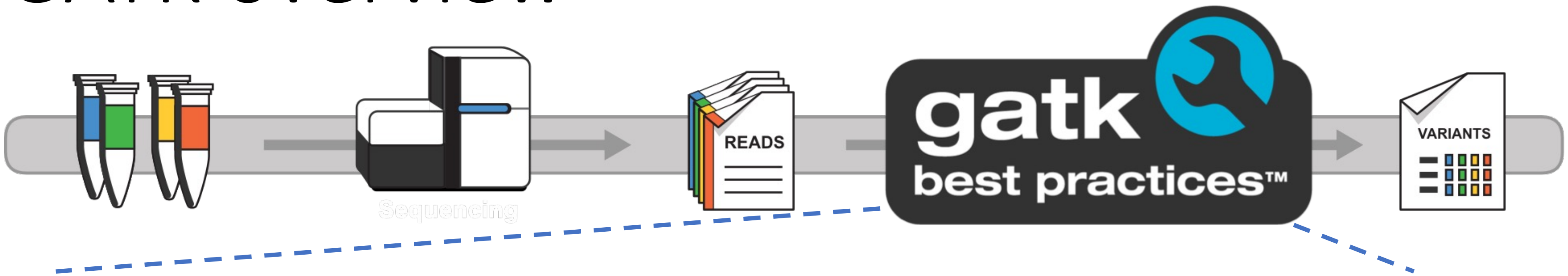
Types of genetic variants:



Focus of this workshop:
Calling short germline variants



3. GATK overview



Pre-processing

- Prepare analysis ready bam files

Variant discovery

- Variant discovery

Refinement

- Use metadata from variant calling to improve genotype accuracy

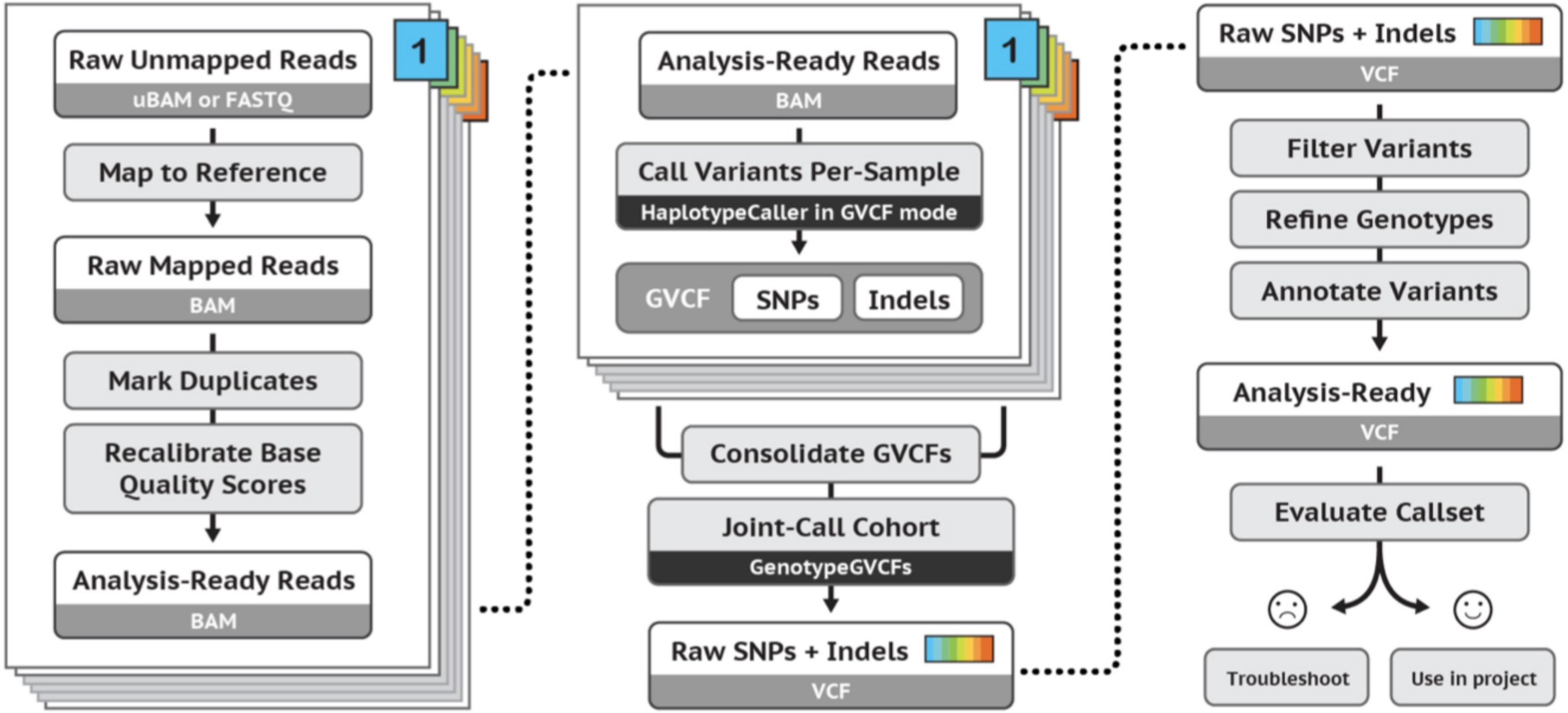
- Genome Analysis Toolkit (GATK): software package to analyze high-throughput sequencing data

3. GATK overview

- Download available from
- https://github.com/broadinstitute/gatk/releases
- Tutorial version: GATK 4.2.0.0
- Current version: GATK 4.4.0.0
- Explore GATK website - gatk.broadinstitute.org
    - Tool index – provides tools usage instructions
    - Technical documentation – provides details on for example Algorithms
    - Forum – provides access to Q&As and community discussions

Melbourne Bioinformatics
BIOINFORMATICS + DATA SERVICES + INFRASTRUCTURE, FOR LIFE SCIENCES TODAY

3. GATK Best practices pipeline

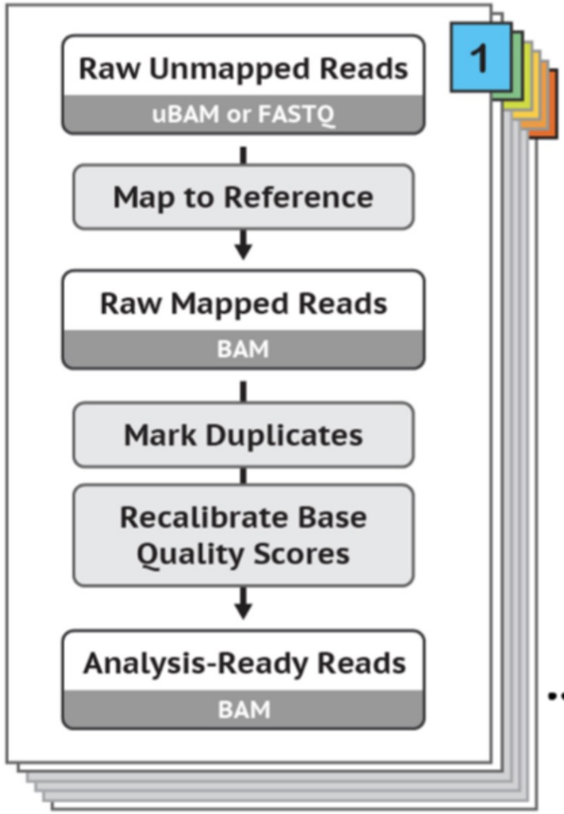


3.1 Pre-processing

3.2 Variant discovery

3.3 Refinement

3.1 Pre-processing



1

Raw Unmapped Reads
uBAM or FASTQ

Map to Reference

Raw Mapped Reads
BAM

Mark Duplicates

Recalibrate Base Quality Scores

Analysis-Ready Reads
BAM

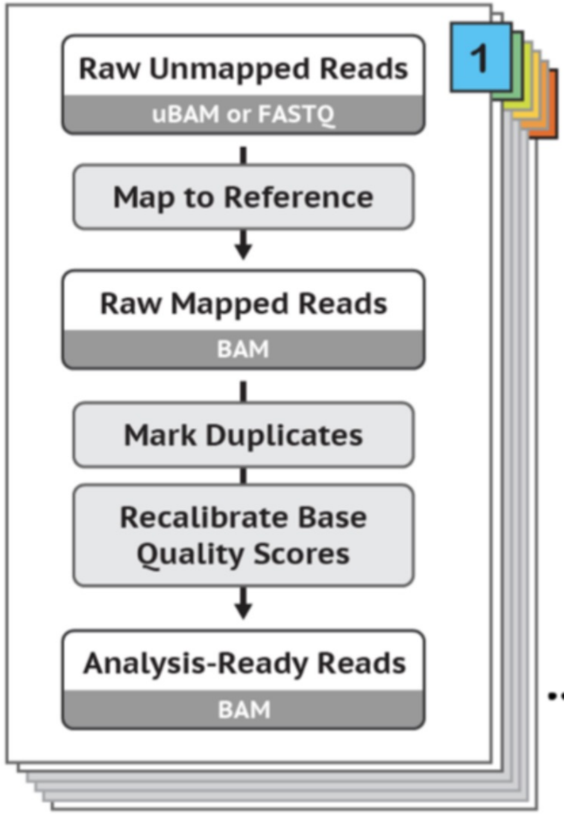
Pre-processing

- A sequencing experiment results in a large volume of sequencing reads

- Reads are not mapped to a reference
- Reads can contain errors and technical artifacts
- e.g. a molecule sequenced multiple times will result in duplicate reads

- We need to filter and prepare the reads and the alignment data – ready for variant calling

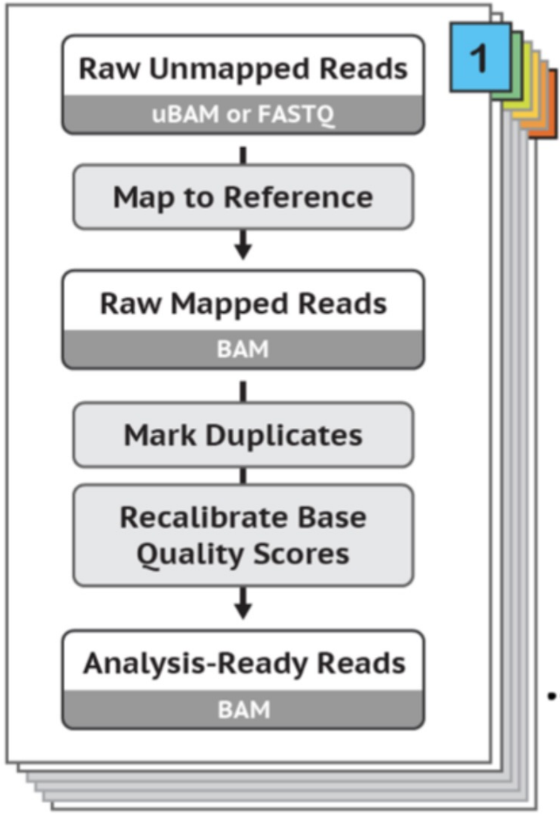
3.1 Map to reference



Pre-processing

- **BWA-MEM**
- `bwa mem -M -t 4 -R "@RG\tID:SRR622461.7\tSM:NA12878\tLB:ERR194147\tPL:ILLUMINA" <reference> sample_1.fastq sample_2.fastq > alignment.sam`
- -M: inserts a tag to the alignment if non-primary alignment (required by GATK)
- -R: read group
- -t: threads or number of cpus
- <reference>: path to reference genome in fasta format and the BWA index files

3.1 Map to reference



Pre-processing

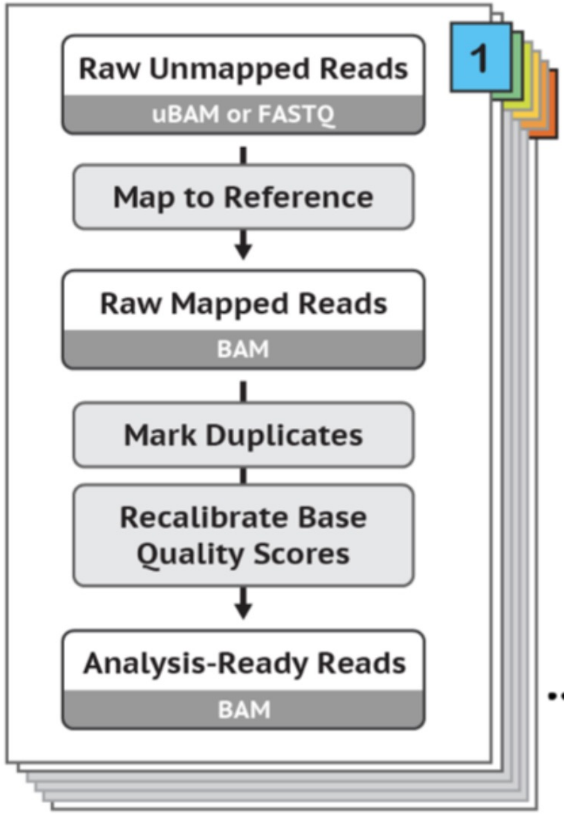
- **BWA-MEM**

```
bwa mem -M -t 4 -R
"@RG\tID:SRR622461.7\tSM:NA12878\tLB:ERR194147\tPL:ILLUMINA"
<reference> sample_1.fastq sample_2.fastq > alignment.sam
```

- -R: read group contains information such as the sample name, library and flow cell.
- Refers to a set of reads generated from a single sequencing run in particular machine

```
@RG     ID:SRR622461.7  SM:NA12878      LB:ERR194147    PL:ILLUMINA
```

3.1 Map to reference



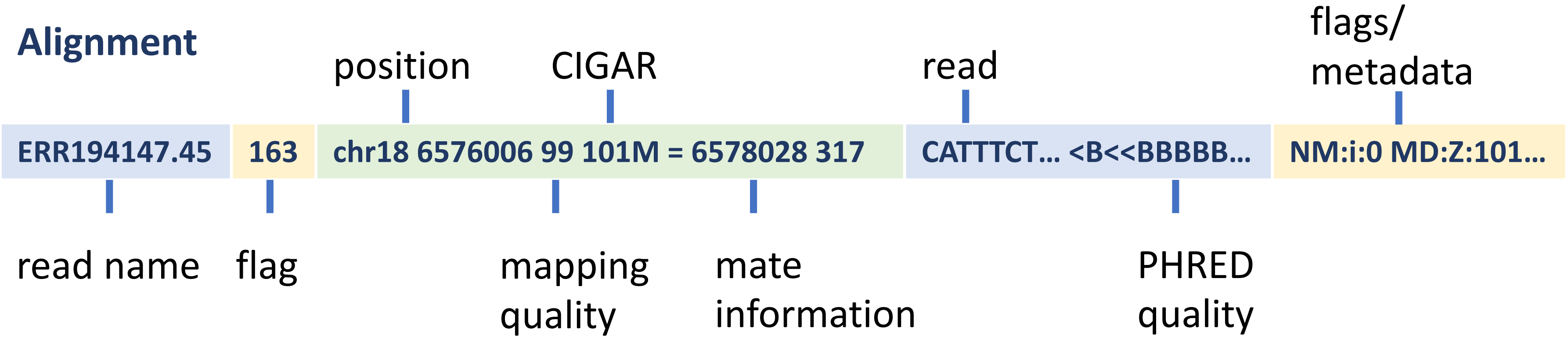
Pre-processing

- Output is a SAM/BAM file.
- SAM file specifications: https://samtools.github.io/hts-specs/SAMv1.pdf

Header

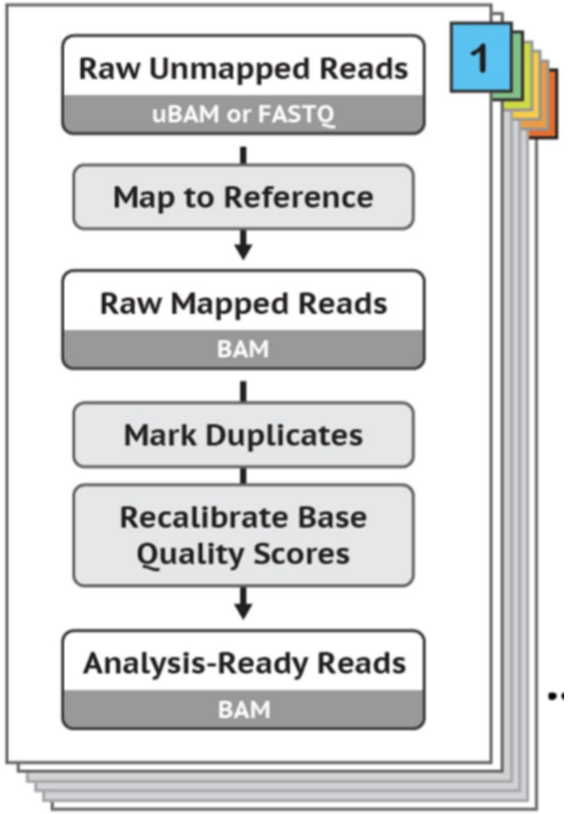
```
@HD    VN:1.5  SO:coordinate
@RG    ID:SRR622461.7  SM:NA12878    LB:ERR194147   PL:ILLUMINA
@PG    ID:bwa  PN:bwa  VN:0.7.17-r1188 CL:bwa mem -M -t 4 -R
```

Alignment



ERR194147.45 163 chr18 6576006 99 101M = 6578028 317 CATTTCT… <B<<BBBBB… NM:i:0 MD:Z:101…

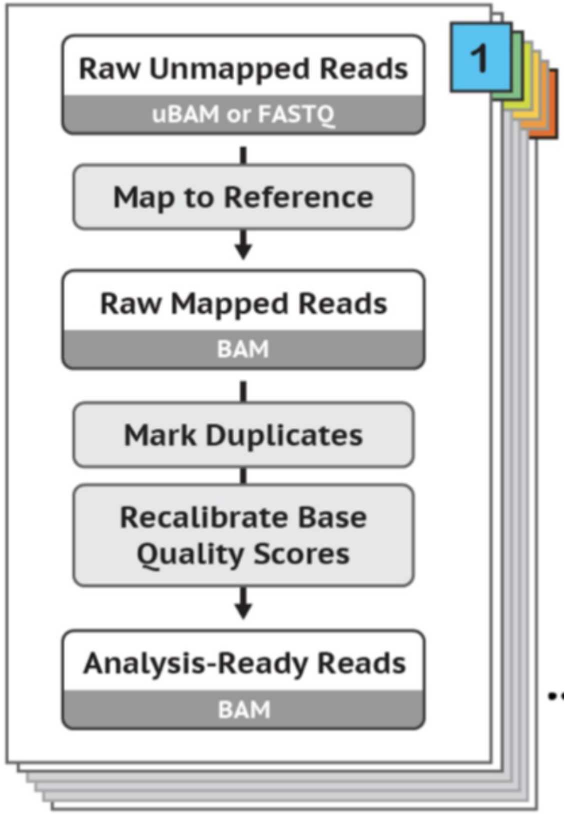
3.1 Mark duplicates



Pre-processing

- **Mark Duplicates**
- Identify reads that are non-independent measurement of sequence fragment
  - Same template of DNA sampled multiple times
  - PCR duplicates
- High sequence identify
- Align to same reference position

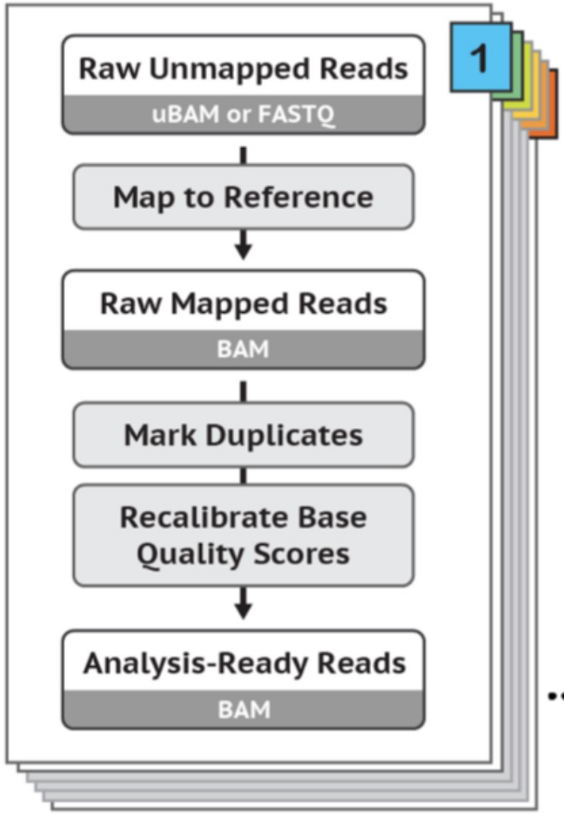
3.1 Mark duplicates



Pre-processing

- **Mark Duplicates**
  - `gatk MarkDuplicates –I sample.bam –O sample.dedup.bam –M sample.dedup.metrics.txt`
- Recommended to be performed on reads per library or lane
- SAM flags are used to mark reads as duplicates
- Downstream GATK tools depend on these flags to assess support for variants and alleles

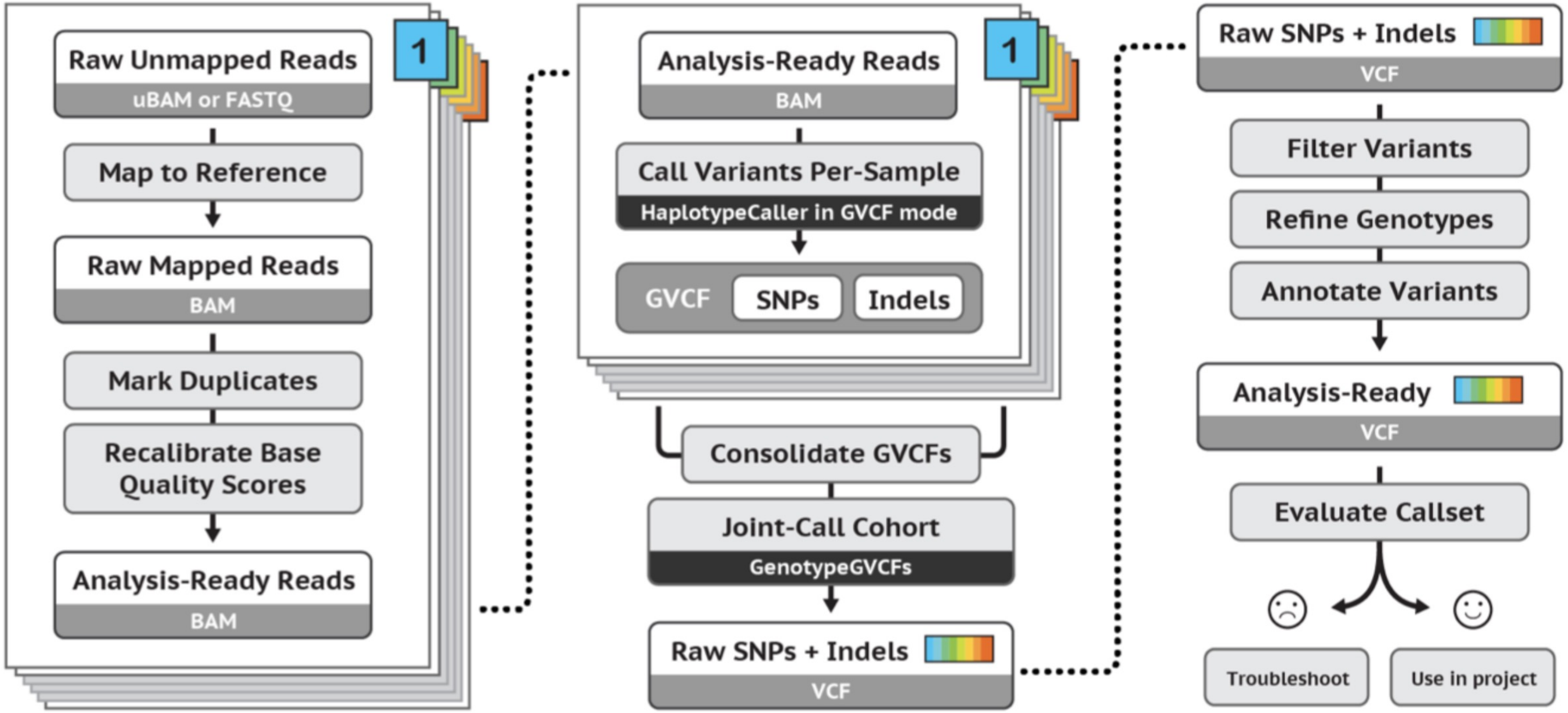
3.1 Base recalibration



Pre-processing

- **Base recalibration**
- `gatk tools BaseRecalibrator and ApplyRecalibration`
- Performed per-sample to detect and correct for patterns of systematic errors in base quality scores.
- Evidenced by calculating metrics based on known variant locations
- Important for building reliable evidence for downstream analysis.

3. GATK Best practices pipeline



Raw Unmapped Reads
uBAM or FASTQ
Map to Reference
Raw Mapped Reads
BAM
Mark Duplicates
Recalibrate Base Quality Scores
Analysis-Ready Reads
BAM

Analysis-Ready Reads
BAM
Call Variants Per-Sample
HaplotypeCaller in GVCF mode
GVCF SNPs Indels
Consolidate GVCFs
Joint-Call Cohort
GenotypeGVCFs
Raw SNPs + Indels
VCF

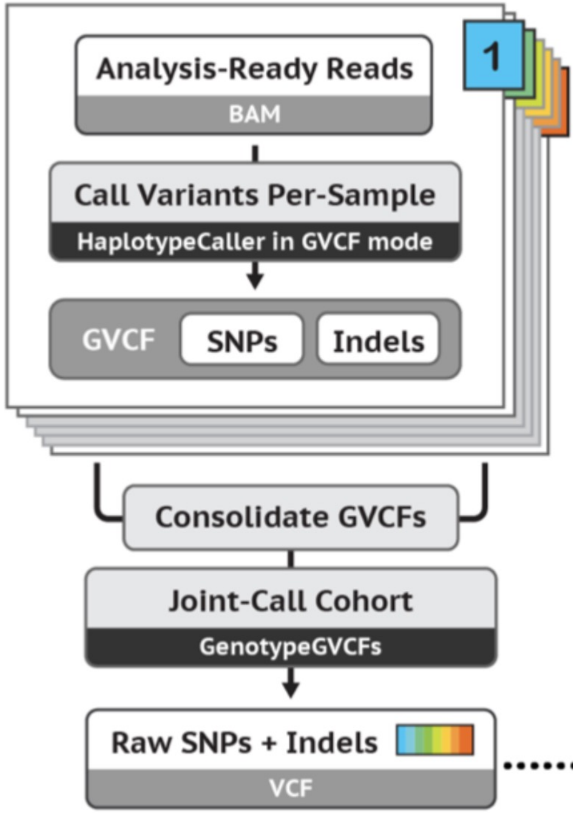
Raw SNPs + Indels
VCF
Filter Variants
Refine Genotypes
Annotate Variants
Analysis-Ready
VCF
Evaluate Callset
☹ Troubleshoot
☺ Use in project

3.1 Pre-processing

3.2 Variant discovery

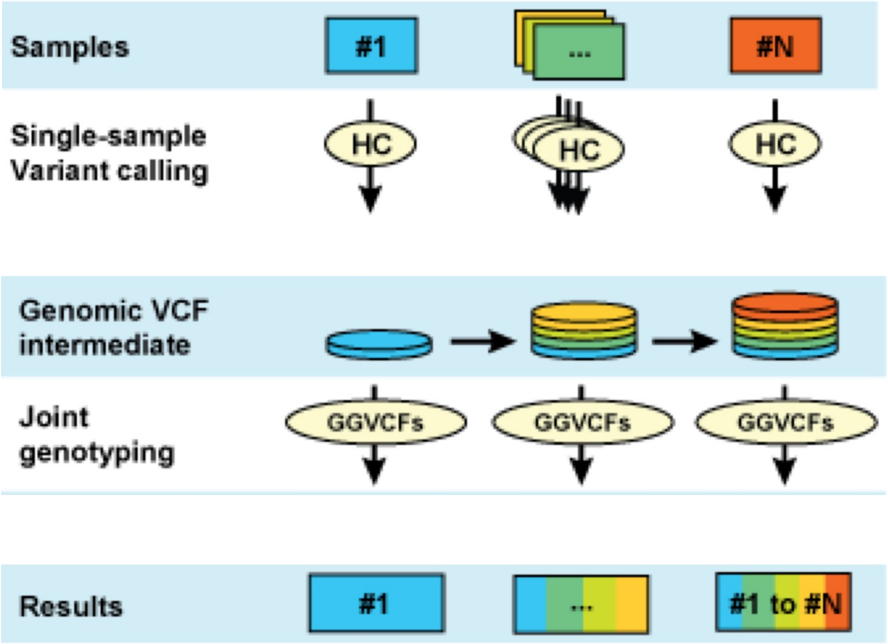
3.3 Refinement

3.2 Variant discovery



Variant discovery

- **Software**

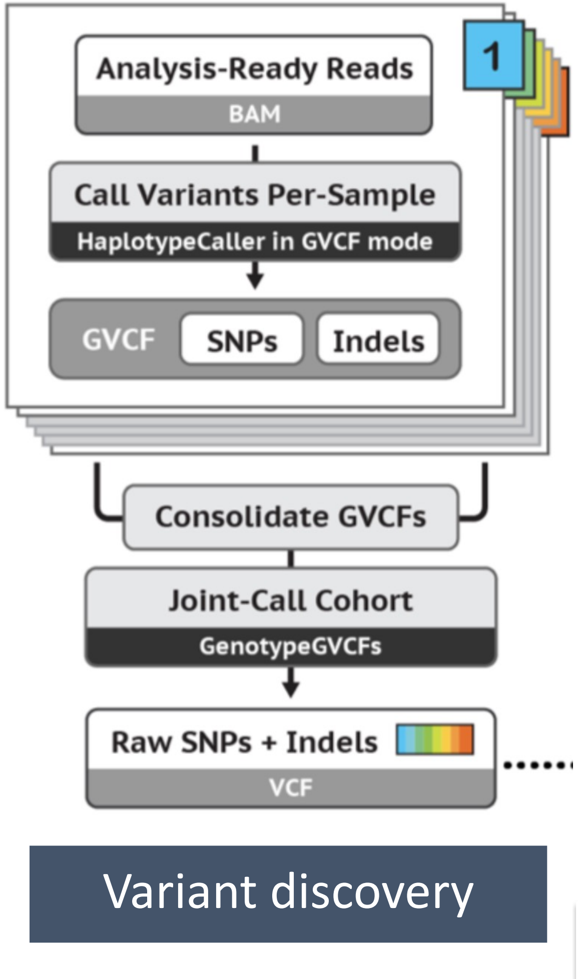


HaplotypeCaller

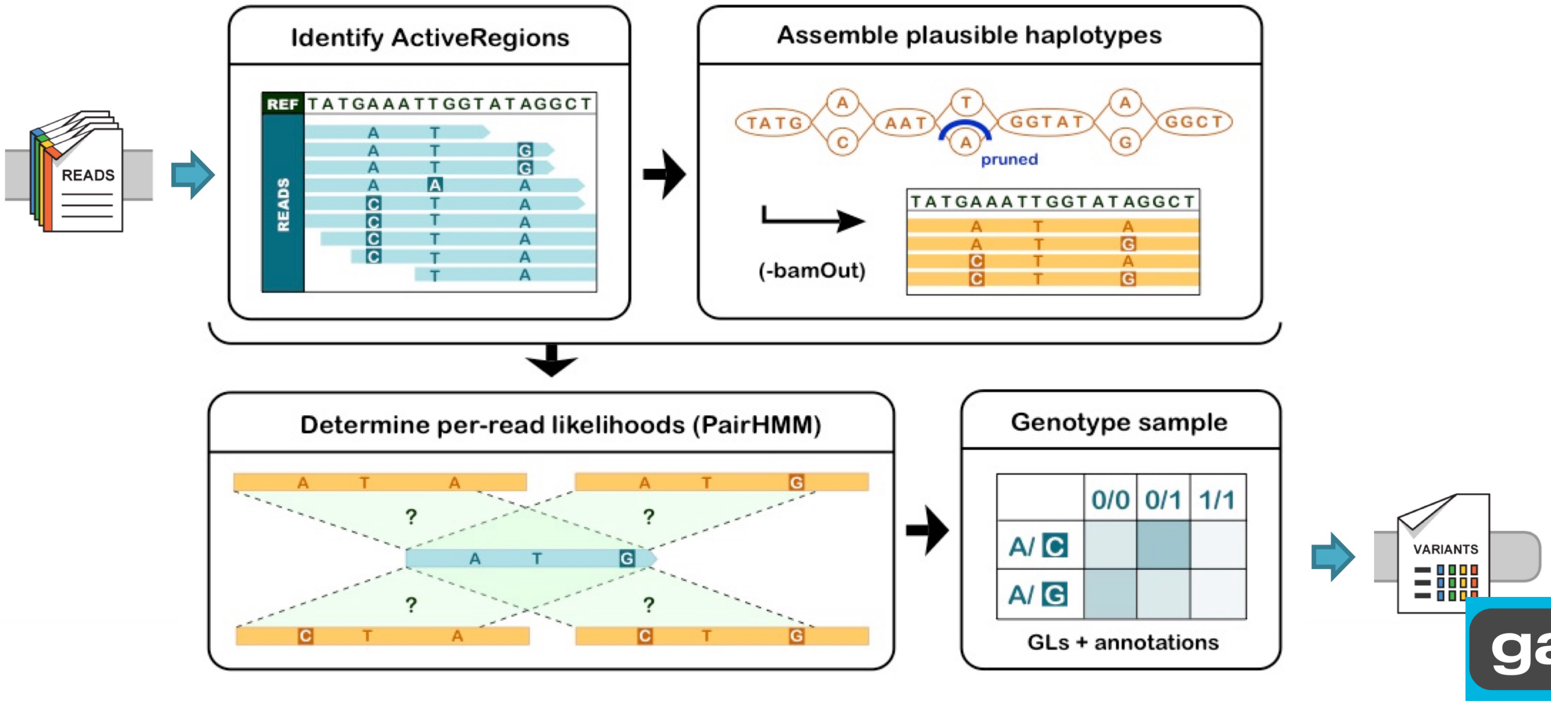
CombineGVCFs/ GenomicsDBImport

GenotypeGVCFs

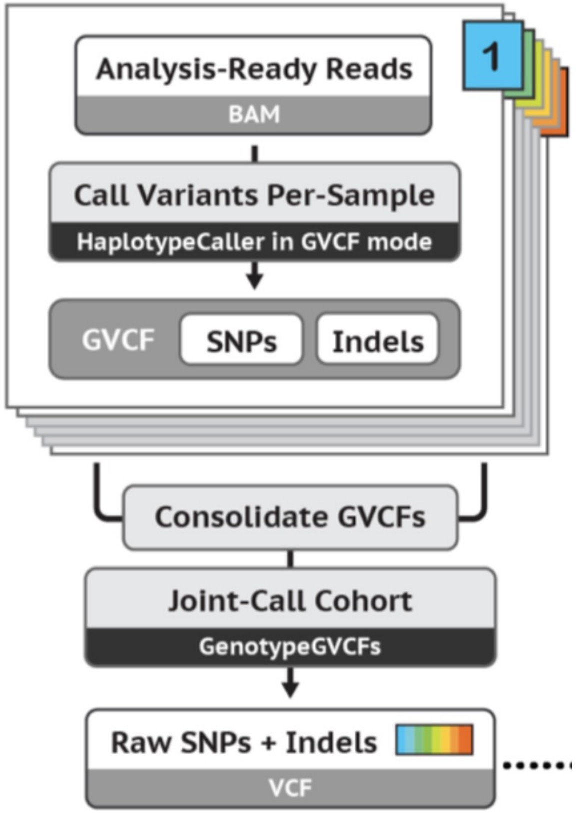
3.2 Variant discovery



- **HaplotypeCaller**
  - `gatk --java-options "-Xmx4g" HaplotypeCaller -R <reference.fa> -I input.bam -O output.g.vcf.gz -ERC GVCF`



3.2 Variant discovery



Analysis-Ready Reads

BAM

Call Variants Per-Sample

HaplotypeCaller in GVCF mode

GVCF

SNPs

Indels

1

Consolidate GVCFs

Joint-Call Cohort

GenotypeGVCFs

Raw SNPs + Indels

VCF

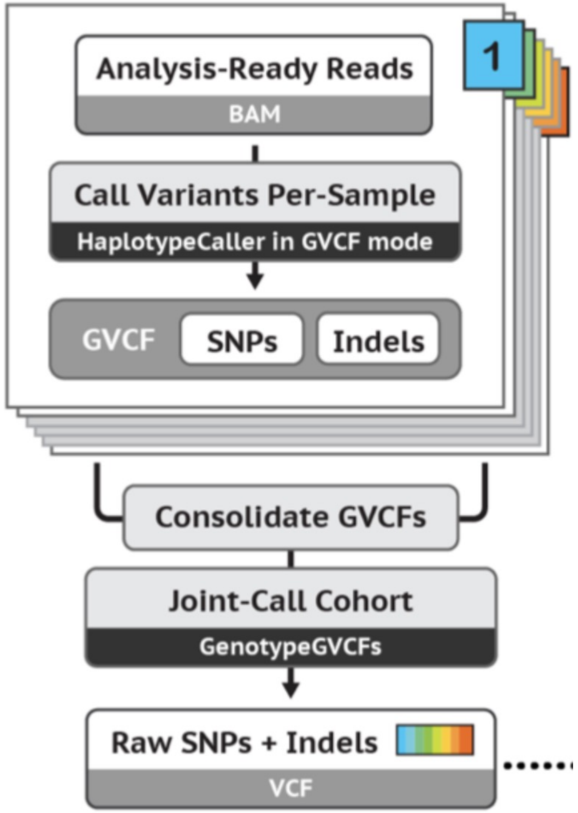
Variant discovery

- **CombineGVCFs**

- `gatk CombineGVCFs R <reference.fa> --variant sample1.g.vcf.gz --variant sample2.g.vcf.gz –O cohort.g.vcf.gz`

- Combine per samples gVCF files (produced by HaplotypeCaller) into a multi-sample gVCF file.

3.2 Variant discovery



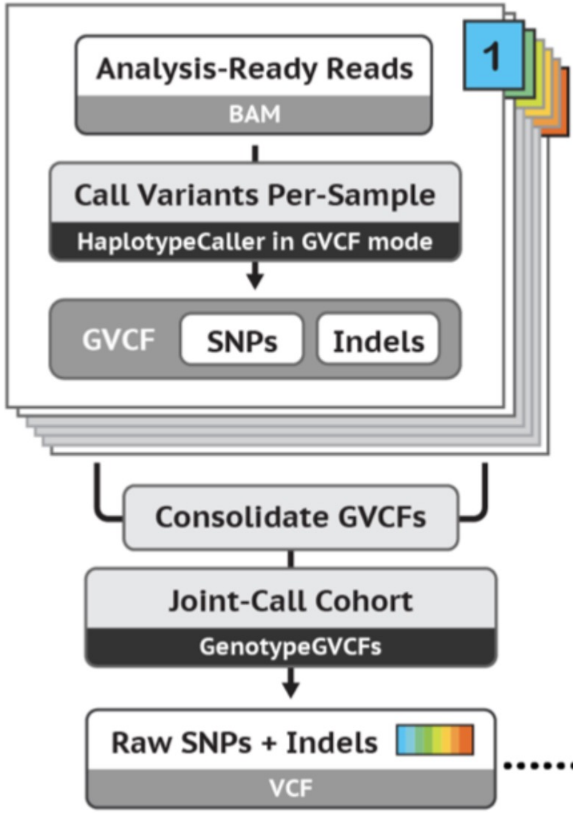
Variant discovery

- **GenotypeGVCFs**

- `gatk --java-options "-Xmx4g" GenotypeGVCFs –R <reference.fa> -V cohort.g.vcf.gz -O output.vcf.gz`

- Combine per samples gVCF files (produced by HaplotypeCaller) into a multi-sample gVCF file.

3.2 Variant discovery



Analysis-Ready Reads
BAM
Call Variants Per-Sample
HaplotypeCaller in GVCF mode
GVCF
SNPs
Indels
1
Consolidate GVCFs
Joint-Call Cohort
GenotypeGVCFs
Raw SNPs + Indels
VCF

Variant discovery

- Output is a VCF file
- VCF file specifications https://samtools.github.io/hts-specs/VCFv4.2.pdf

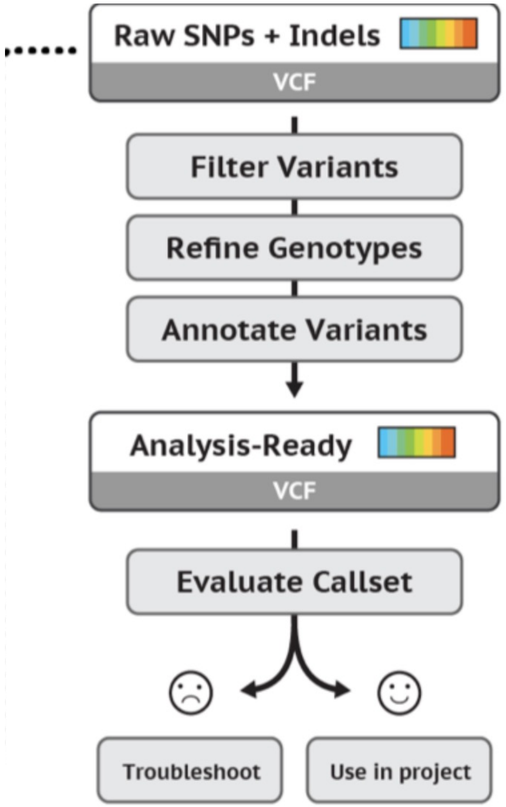
Header

```
#fileformat=VCFv4.2
##FILTER=<ID=PASS,Description="All filters passed">
##contig=<ID=1,length=249250621>
##FORMAT=<ID=AD,Number=R,Type=Integer,Description="Allelic depths for the ref and alt alleles in the order listed">
```

Variant record

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	Sample1
1	567376	.	G	A	146.3	PASS	AC=1;DP=55	GT:AD:DP	0/1:30,25:55

3.3 Variant Refinement



Refinement

- Variant callers are sensitive
- The aim here is to identify potential false positives and apply filters to remove those less likely to be real variants. Strategies include:

1. Variant quality score recalibration (using known sites)
2. Hard filtering on quality criteria
3. Annotation features

Variant record

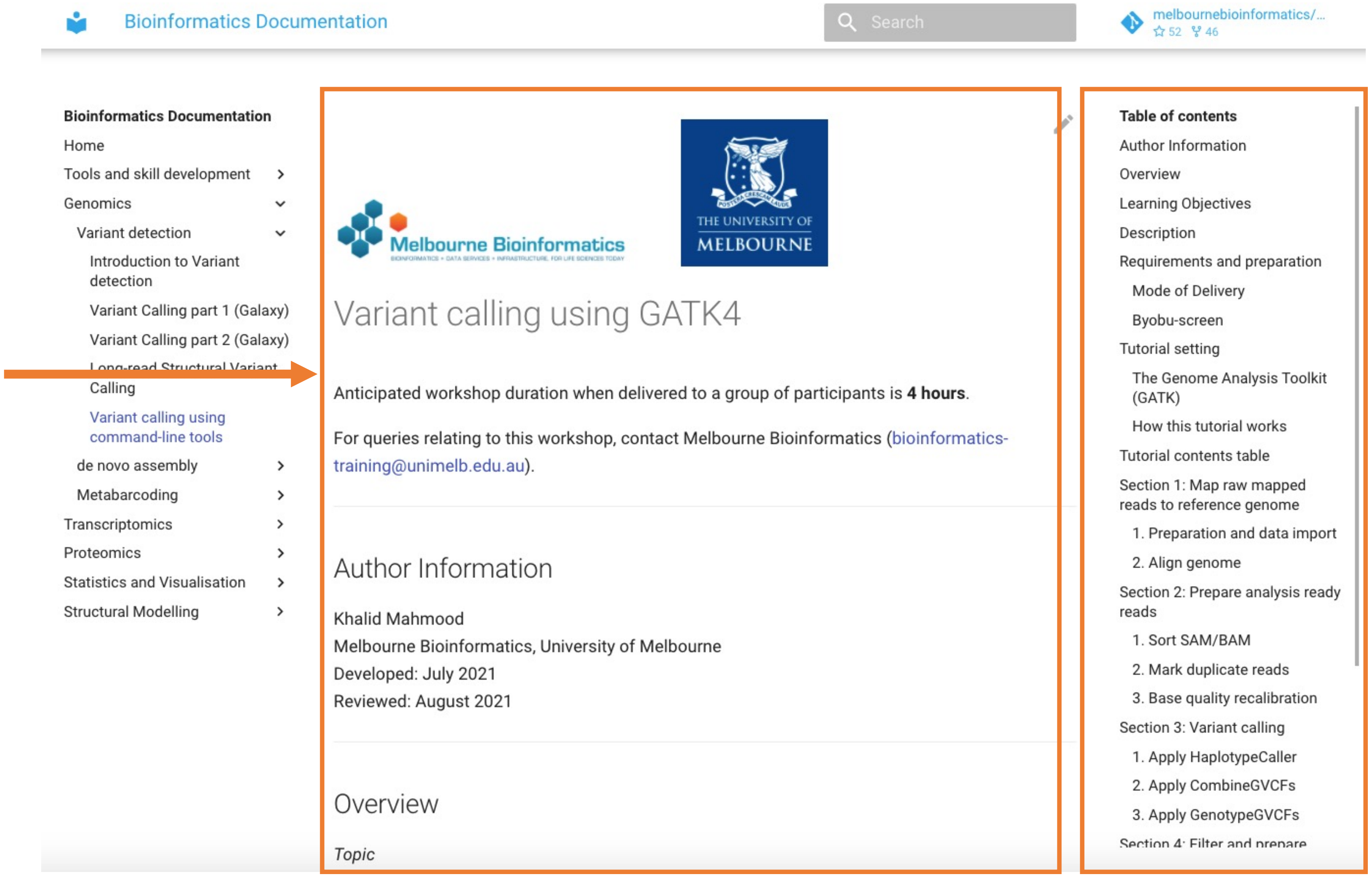
#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	Sample
1	567376	.	G	A	146.3	PASS	AC=1;DP=55	GT:AD:DP	0/1:30,25:55

Workshop structure

Workshop structure

Tutorial content

Tutorial navigation



Bioinformatics Documentation

Search

melbournebioinformatics/...
☆ 52 ⑂ 46

Bioinformatics Documentation

- Home
- Tools and skill development
- Genomics
  - Variant detection
    - Introduction to Variant detection
    - Variant Calling part 1 (Galaxy)
    - Variant Calling part 2 (Galaxy)
    - Long-read Structural Variant Calling
    - Variant calling using command-line tools
  - de novo assembly
  - Metabarcoding
- Transcriptomics
- Proteomics
- Statistics and Visualisation
- Structural Modelling

Melbourne Bioinformatics

THE UNIVERSITY OF MELBOURNE

Variant calling using GATK4

Anticipated workshop duration when delivered to a group of participants is **4 hours**.

For queries relating to this workshop, contact Melbourne Bioinformatics (bioinformatics-training@unimelb.edu.au).

Author Information

Khalid Mahmood
Melbourne Bioinformatics, University of Melbourne
Developed: July 2021
Reviewed: August 2021

Overview

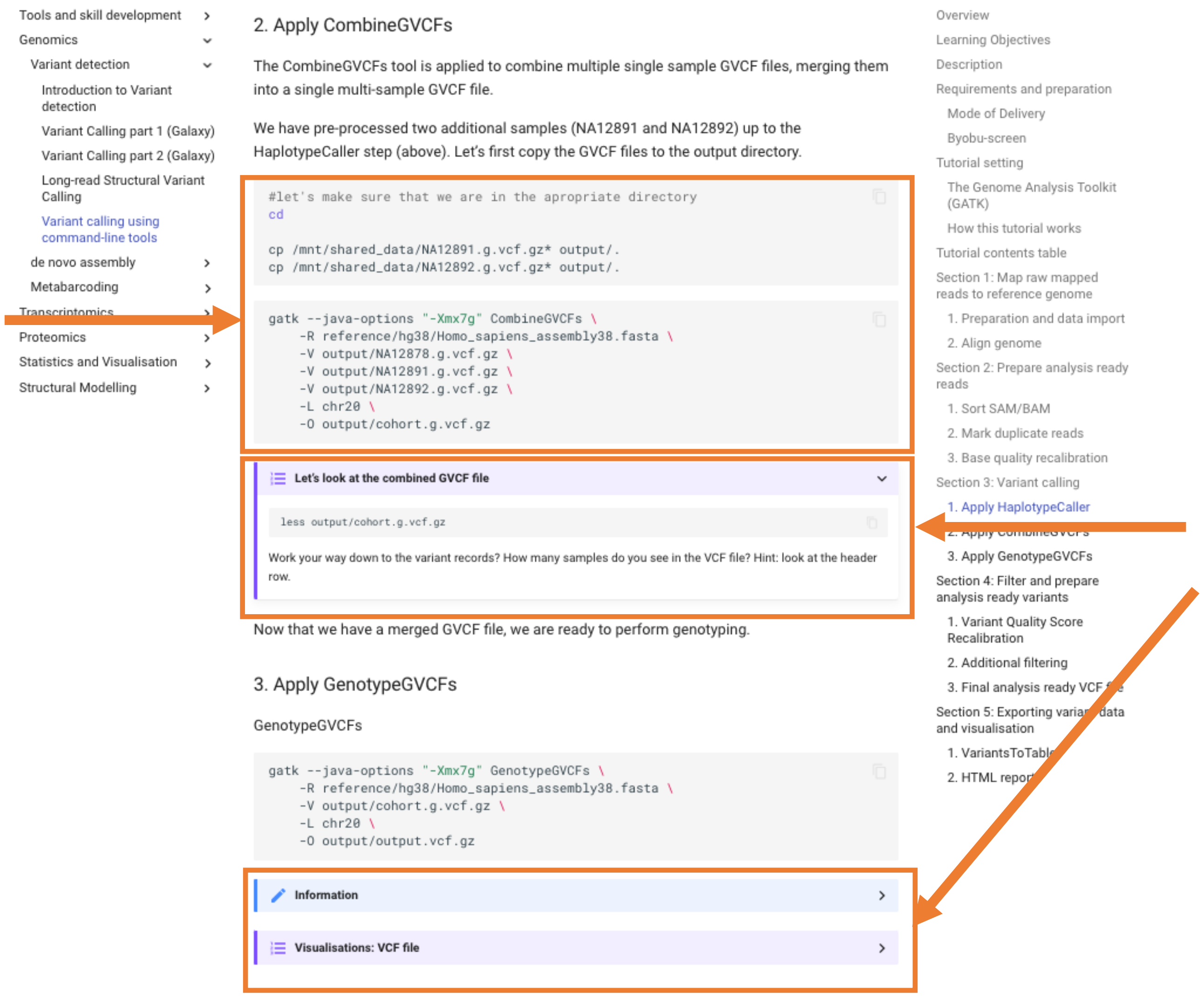
Topic

Table of contents

Melbourne Bioinformatics
BIOINFORMATICS + DATA SERVICES + INFRASTRUCTURE, FOR LIFE SCIENCES TODAY

Workshop structure

Command and output blocks
'#' comments - do not run


Tools and skill development
Genomics
Variant detection
Introduction to Variant detection
Variant Calling part 1 (Galaxy)
Variant Calling part 2 (Galaxy)
Long-read Structural Variant Calling
Variant calling using command-line tools
de novo assembly
Metabarcoding
Transcriptomics
Proteomics
Statistics and Visualisation
Structural Modelling

2. Apply CombineGVCFs

The CombineGVCFs tool is applied to combine multiple single sample GVCF files, merging them into a single multi-sample GVCF file.

We have pre-processed two additional samples (NA12891 and NA12892) up to the HaplotypeCaller step (above). Let's first copy the GVCF files to the output directory.

```
#let's make sure that we are in the apropriate directory
cd

cp /mnt/shared_data/NA12891.g.vcf.gz* output/.
cp /mnt/shared_data/NA12892.g.vcf.gz* output/.
```

```
gatk --java-options "-Xmx7g" CombineGVCFs \
    -R reference/hg38/Homo_sapiens_assembly38.fasta \
    -V output/NA12878.g.vcf.gz \
    -V output/NA12891.g.vcf.gz \
    -V output/NA12892.g.vcf.gz \
    -L chr20 \
    -O output/cohort.g.vcf.gz
```

```
less output/cohort.g.vcf.gz
```

Work your way down to the variant records? How many samples do you see in the VCF file? Hint: look at the header row.

Now that we have a merged GVCF file, we are ready to perform genotyping.

3. Apply GenotypeGVCFs

GenotypeGVCFs

```
gatk --java-options "-Xmx7g" GenotypeGVCFs \
    -R reference/hg38/Homo_sapiens_assembly38.fasta \
    -V output/cohort.g.vcf.gz \
    -L chr20 \
    -O output/output.vcf.gz
```

Information
Visualisations: VCF file
Overview
Learning Objectives
Description
Requirements and preparation
Mode of Delivery
Byobu-screen
Tutorial setting
The Genome Analysis Toolkit (GATK)
How this tutorial works
Tutorial contents table
Section 1: Map raw mapped reads to reference genome
1. Preparation and data import
2. Align genome
Section 2: Prepare analysis ready reads
1. Sort SAM/BAM
2. Mark duplicate reads
3. Base quality recalibration
Section 3: Variant calling
1. Apply HaplotypeCaller
2. Apply CombineGVCFs
3. Apply GenotypeGVCFs
Section 4: Filter and prepare analysis ready variants
1. Variant Quality Score Recalibration
2. Additional filtering
3. Final analysis ready VCF
Section 5: Exporting varia data and visualisation
1. VariantsToTabl
2. HTML repor

Interactive sections
Notes, hints, exercises

Workshop structure

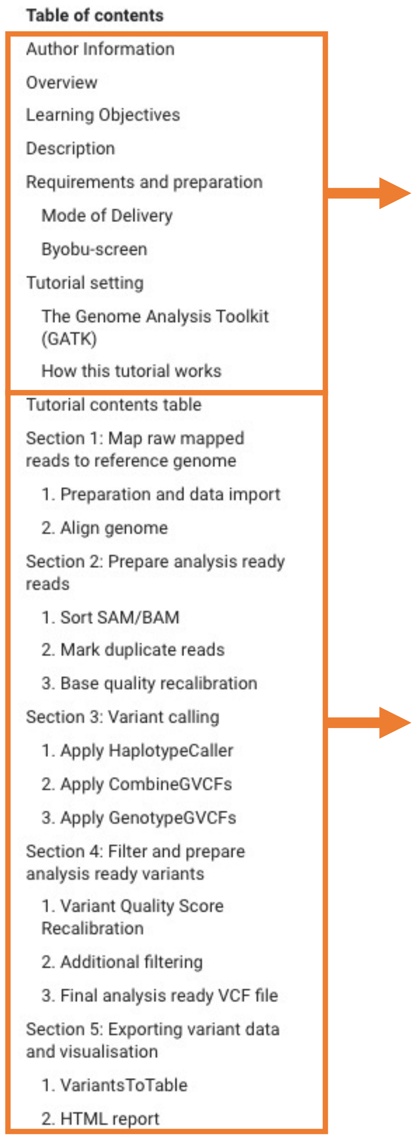


Table of contents

Introductory material
Tutorial delivery and some instructions

Workshop content:

- 5 sections
- Each section covers a stage in the variant calling pipeline
- Each section has a text explain the process and links to relevant material
- Sections have multiple steps. Mostly have an input and an expected output file.
- This is a pipeline: input to a step is the output from a previous step

Workshop computers

- We will be conducting the workshop on virtual machines

- Hosted on the University of Melbourne Research Cloud service and the ARDC Nectar Research Cloud infrastructure.

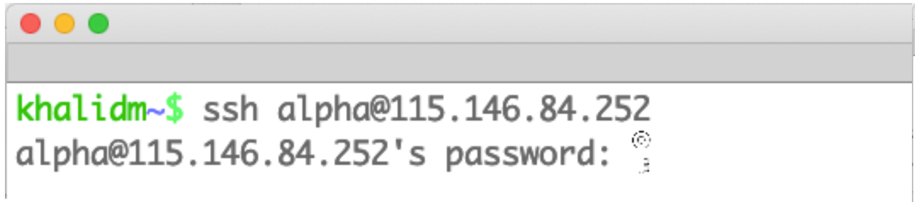
- Infrastructure for development and setup of the workshops machines by Catherine Bromhead and Simon Gladman

Workshop computers

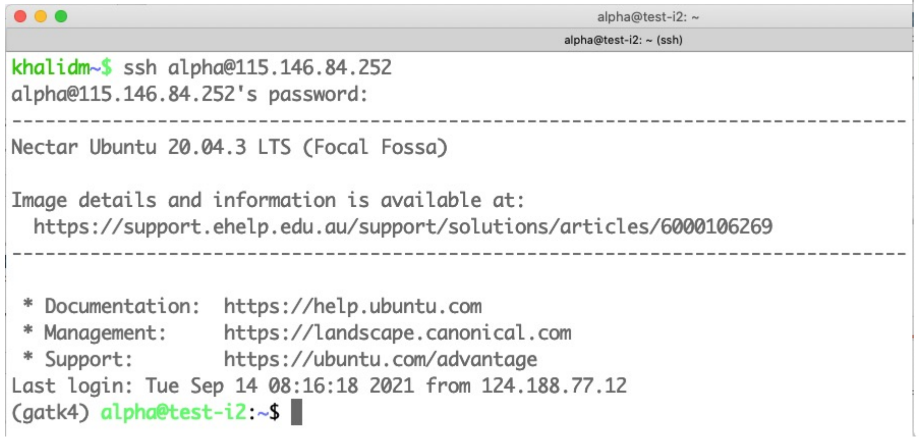
- Each participant should have a username and a password
- Each participant will be assigned a log in to one of the VM machines:
  - Follow the google sheet link for more details
- Configuration:

Log on to the VMs

- Open a terminal window and on the command prompt type and enter:



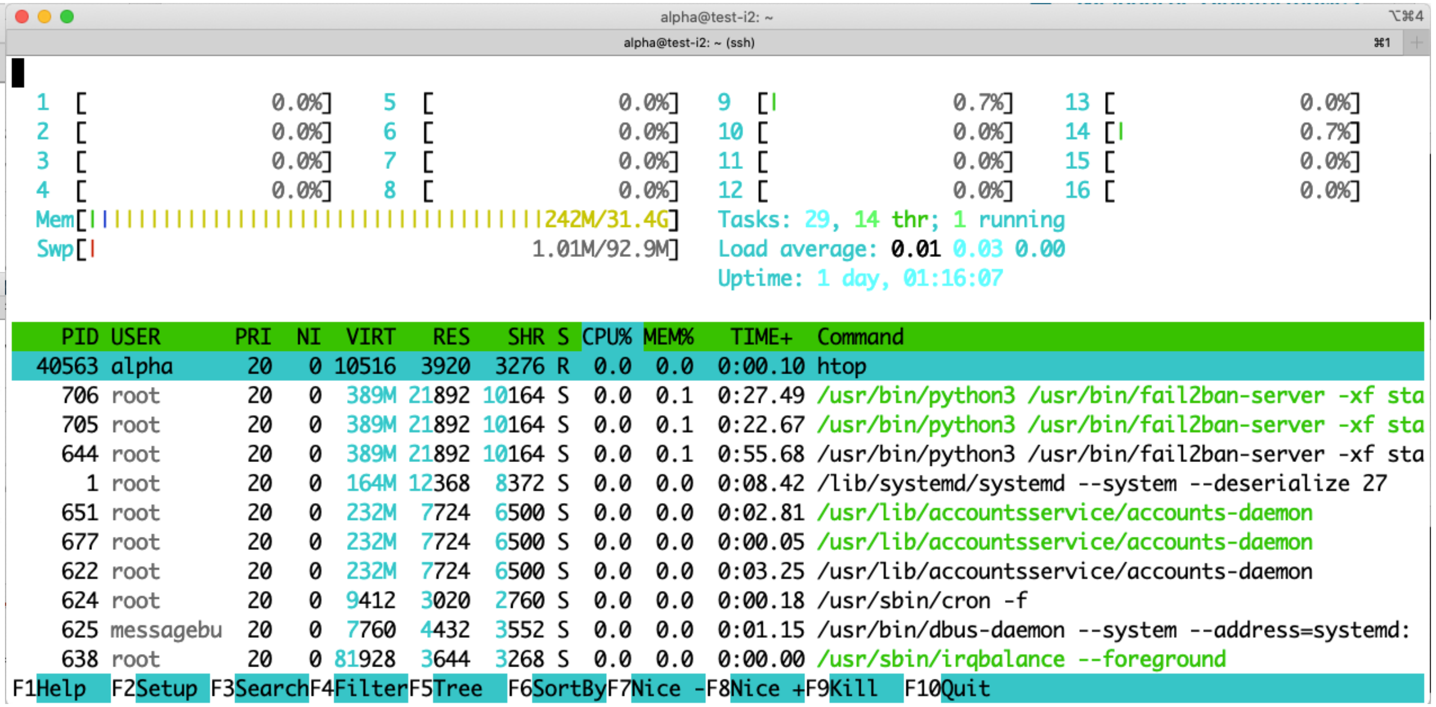
```
khalidm~$ ssh alpha@115.146.84.252
alpha@115.146.84.252's password:
```



```
khalidm~$ ssh alpha@115.146.84.252
alpha@115.146.84.252's password:
-------------------------------------------------------------------------------
Nectar Ubuntu 20.04.3 LTS (Focal Fossa)

Image details and information is available at:
  https://support.ehelp.edu.au/support/solutions/articles/6000106269
-------------------------------------------------------------------------------

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
Last login: Tue Sep 14 08:16:18 2021 from 124.188.77.12
(gatk4) alpha@test-i2:~$
```



```
  1  [                     0.0%]   5  [                     0.0%]   9  [|                    0.7%]   13 [                     0.0%]
  2  [                     0.0%]   6  [                     0.0%]   10 [                     0.0%]   14 [|                    0.7%]
  3  [                     0.0%]   7  [                     0.0%]   11 [                     0.0%]   15 [                     0.0%]
  4  [                     0.0%]   8  [                     0.0%]   12 [                     0.0%]   16 [                     0.0%]
  Mem[|||||||||||||||||||||||||||||||||||||||242M/31.4G]   Tasks: 29, 14 thr; 1 running
  Swp[|                                  1.01M/92.9M]   Load average: 0.01 0.03 0.00
                                                         Uptime: 1 day, 01:16:07

    PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
  40563 alpha      20   0 10516  3920  3276 R  0.0  0.0  0:00.10 htop
    706 root       20   0  389M 21892 10164 S  0.0  0.1  0:27.49 /usr/bin/python3 /usr/bin/fail2ban-server -xf sta
    705 root       20   0  389M 21892 10164 S  0.0  0.1  0:22.67 /usr/bin/python3 /usr/bin/fail2ban-server -xf sta
    644 root       20   0  389M 21892 10164 S  0.0  0.1  0:55.68 /usr/bin/python3 /usr/bin/fail2ban-server -xf sta
      1 root       20   0  164M 12368  8372 S  0.0  0.0  0:08.42 /lib/systemd/systemd --system --deserialize 27
    651 root       20   0  232M  7724  6500 S  0.0  0.0  0:02.81 /usr/lib/accountsservice/accounts-daemon
    677 root       20   0  232M  7724  6500 S  0.0  0.0  0:00.05 /usr/lib/accountsservice/accounts-daemon
    622 root       20   0  232M  7724  6500 S  0.0  0.0  0:03.25 /usr/lib/accountsservice/accounts-daemon
    624 root       20   0  9412  3020  2760 S  0.0  0.0  0:00.18 /usr/sbin/cron -f
    625 messagebu  20   0  7760  4432  3552 S  0.0  0.0  0:01.15 /usr/bin/dbus-daemon --system --address=systemd:
    638 root       20   0 81928  3644  3268 S  0.0  0.0  0:00.00 /usr/sbin/irqbalance --foreground
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

Useful Linux commands

- Autofill on command line: **Tab key**
- Abort command: **Ctrl-c**
- List contents of a directory: **ls -l**
- What's the path to my current directory: **pwd**
- Change directory: **cd <path/to/destination>**
- Create a directory: **mkdir <directory name>**
- Copy a file: **cp <source file> <destination path/name>**
- Remove a directory: **rmdir <directory name>**
- Remove a file: **rm <file name>**
- Rename/move a file (this is not copying a file): **mv <source file> <destination file>**
- Open a text editor: **nano**
- Print file content (small files): **cat <file name>**
- Print file content (quick view): **less <file name>**
- Print file content (quick view/first 10 lines of a file): **head <file name>**
- Print file content (quick view/last 10 lines of a file): **tail <file name>**
- curl or wget: download a file from a URL (you will see this in other QIIME2 tutorials)
- Documentation for a command line tool: try **man <tool name**> OR **<command_name> --help**

Workshop data

- Primary data: paired-end sequencing reads from the chr20
    - chr20:2677705-6631126
- Whole genome sequencing data
    - Female
    - Utah resident (European ancestry)
    - 1000 genomes project (NA12878)
- Other data from
    - A male and female
    - Utah resident (European ancestry)
    - 1000 genomes project (NA12891 and NA12892)

Byrska-Bishop, Marta et al. "High coverage whole genome sequencing of the expanded 1000 Genomes Project cohort including 602 trios". *bioRxiv*. (2021).

Byobu-screen

- A terminal multiplexer or a tool to to create multiple ‘windows’ in a single screen
- Improves stability of terminal sessions when connected to a remove computer

- List screen sessions: byobu-screen -ls
- Start new session: byobu-screen -S workshop
- Detach from screen to original window: Ctrl-a-d

- More details:
- https://www.melbournebioinformatics.org.au/tutorials/tutorials/variant_calling_gatk1/variant_calling_gatk1/#byobu-screen

Workshop

https://www.melbournebioinformatics.org.au/tutorials/tutorials/variant_calling_gatk1/variant_calling_gatk1/

Break…

4. Resources and tools

- GATK resources bundle: collection of files for GATK based analysis working with human sequencing data.

- ftp://gsapubftp-anonymous@ftp.broadinstitute.org/bundle/hg38

1000G_omni2.5.hg38.vcf.gz
1000G_phase1.snps.high_confidence.hg38.vcf.gz
Axiom_Exome_Plus.genotypes.all_populations.poly.hg38.vcf.gz
dbsnp_146.hg38.vcf.gz
hapmap_3.3_grch38_pop_stratified_af.vcf.gz
hapmap_3.3.hg38.vcf.gz
Homo_sapiens_assembly38.dict
Homo_sapiens_assembly38.fasta
Homo_sapiens_assembly38.fasta.gz
Mills_and_1000G_gold_standard.indels.hg38.vcf.gz

4. Resources and tools

- BWA-MEM index
- bwa index Homo_sapiens_assembly38.fasta

```
Homo_sapiens_assembly38.fasta
Homo_sapiens_assembly38.fasta.amb
Homo_sapiens_assembly38.fasta.ann
Homo_sapiens_assembly38.fasta.bwt
Homo_sapiens_assembly38.fasta.pac
Homo_sapiens_assembly38.fasta.sa
```

Melbourne Bioinformatics
BIOINFORMATICS + DATA SERVICES + INFRASTRUCTURE, FOR LIFE SCIENCES TODAY

4. Resources and tools

Tools name	function
FastQC	QC tools for raw sequencing reads
MultiQC	QC report aggregator (generates an HTML report)
GATK	Set of tools for variant calling
Picard	A command line tool to analysis and manipulate sequencing files
Samtools	Suite of tools for interacting with mapped sequencing reads (SAM/BAM/CRAM format)
BCFtools	Suite of tools for interacting with variant data (VCF/BCF formats)

4. Genetic variant resources

- dbSNP
  - An archive of genetic variations – contains ~700 million variants
  - ~90% have a recorded population frequency
- gnomAD
  - Aggregation of variants derived from re-analysis of >125k WES and WGS
- ClinVar
  - Aggregates genetic variations and its relationships with phenotypes
- UCSC genome browser
- UniProt

4. Help

- Tool documentation
- GATK forum
- Online resources (e.g. Biostar)
- GitHub for technical issues/discussions

THE UNIVERSITY OF
MELBOURNE

Melbourne Bioinformatics

BIOINFORMATICS + DATA SERVICES + INFRASTRUCTURE, FOR LIFE SCIENCES TODAY