Bioinformatics Workshop

Week 07

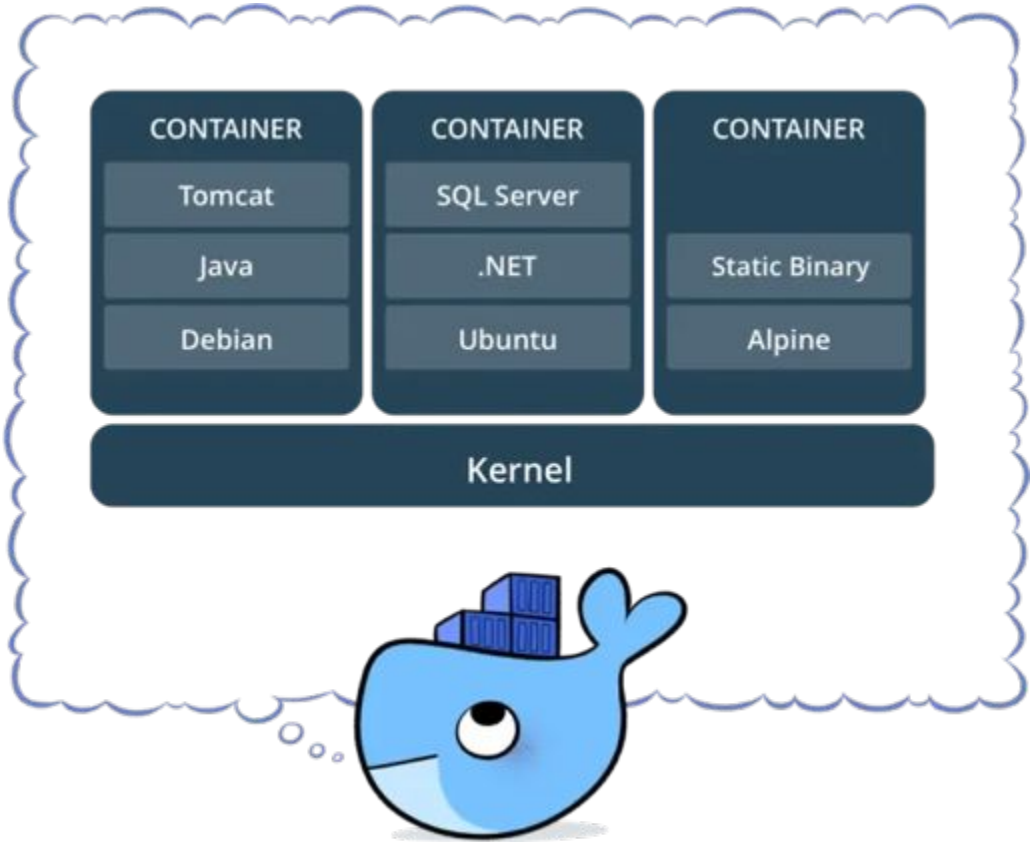
Docker and Germline Variant Calling

Chris Miller and Alex Paul

Computing Environments

- Laptop
  - You administer
  - You control completely
- Shared compute cluster
  - A sysadmin or group administers it
  - You control very little
- Docker (containers)
  - Sysadmins handle the hardware
  - You control the software almost completely

Docker containers



Finding docker images

- Search engines
  - “docker bedtools”
- Repositories - Bioconda/Quay.io/Dockerhub
  - “docker mosdepth quay”
- Slack - ask around
- Building your own

Building docker images

```
# stage 1 for constructing the GATK zip
FROM broadinstitute/gatk:gatkbase-2.3.0 AS gradleBuild
LABEL stage=gatkIntermediateBuildImage

RUN ls .
ADD . /gatk
WORKDIR /gatk

RUN add-apt-repository universe && apt update
RUN apt-get --assume-yes install git-lfs
RUN git lfs install

RUN git lfs pull

RUN export GRADLE_OPTS="-Xmx4048m -Dorg.gradle.daemon=false" && /gatk/gradlew clean collectBundleIntoD
RUN cp -r $( find /gatk/build -name "*bundle-files-collected" )/ /gatk/unzippedJar/
RUN unzip -o -j $( find /gatk/unzippedJar -name "gatkPython*.zip" ) -d /gatk/unzippedJar/scripts

# Using OpenJDK 8
FROM broadinstitute/gatk:gatkbase-2.3.0

WORKDIR /gatk

# Location of the unzipped gatk bundle files
COPY --from=gradleBuild /gatk/unzippedJar .

#Setup linked jars that may be needed for running gatk
RUN ln -s $( find /gatk -name "gatk*local.jar" ) gatk.jar
RUN ln -s $( find /gatk -name "gatk*local.jar" ) /root/gatk.jar
RUN ln -s $( find /gatk -name "gatk*spark.jar" ) gatk-spark.jar

WORKDIR /root
```

This can be intimidating!

But it’s just a recipe

Building docker images

1. The foundation: **FROM**

    - Tells what operating system you want your stuff to run in

Starting from scratch

```
FROM ubuntu:focal

FROM alpine:3.5
```

Building docker images

1. The foundation: **FROM**

    - Tells what operating system you want your stuff to run in

Starting from scratch

```
FROM ubuntu:focal
```

```
FROM alpine:3.5
```

Standing on the shoulders...

```
FROM: broadinstitute/gatk:4.1.8.1
```

```
FROM: continuumio/miniconda3
```

Building docker images

2. Do stuff: **RUN**

- These commands get passed to the OS to install your programs

```
RUN apt-get install build-essential python3

RUN wget http://github.com/path/to/software.zip && unzip software.zip
```

Building docker images

3. Add stuff: **COPY**

- Take your own local files and include them in the image

On your computer:

```
$ ls mydocker/

 Dockerfile
 myscript.py
```

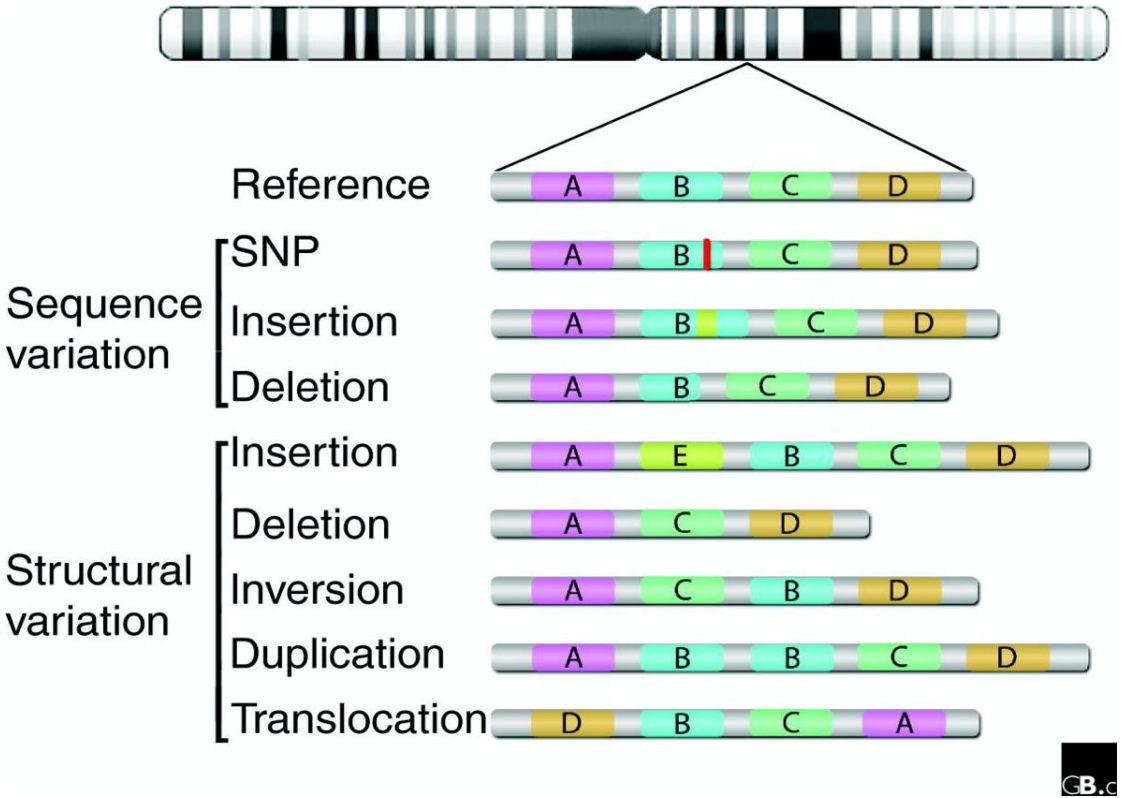
In your Dockerfile

```
COPY myscript.py /usr/bin/
```

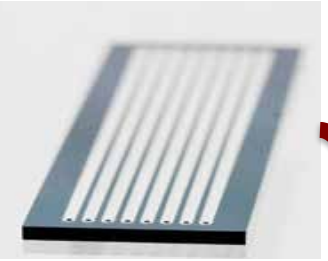
Understanding docker images

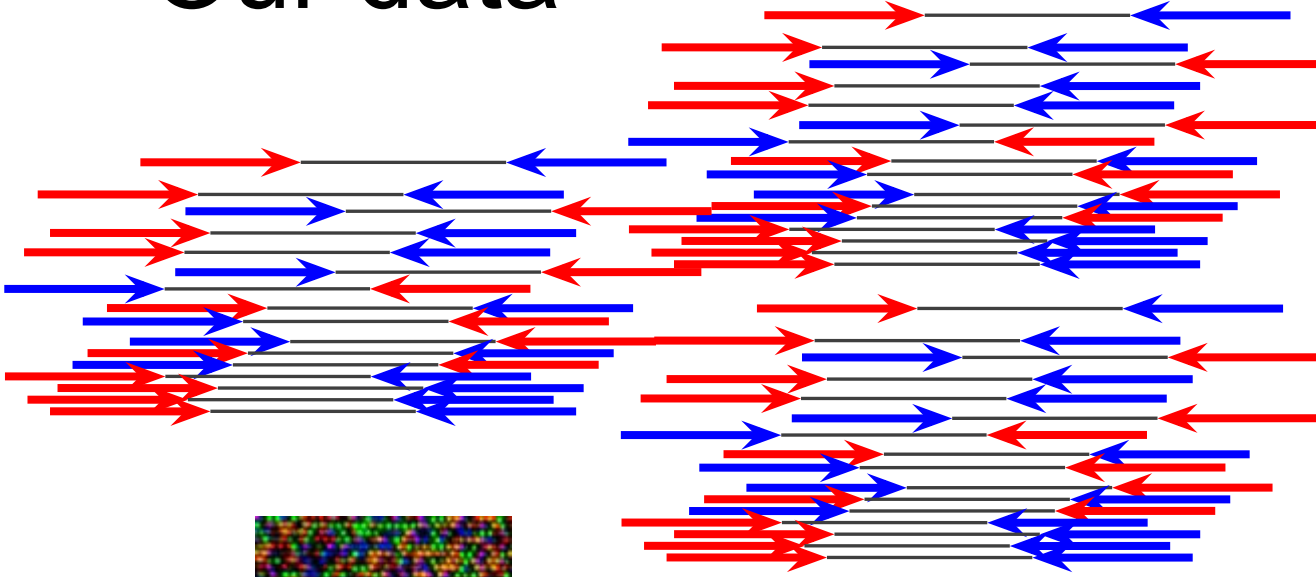
```
$ cat docker-somatic-llr-filter/Dockerfile

FROM continuumio/miniconda3
RUN pip install vcfpy pysam
COPY somatic_llr_filter.py /usr/bin/somatic_llr_filter.py
```

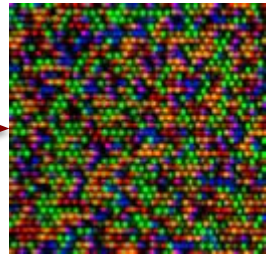
Small Variant Calling



Our data

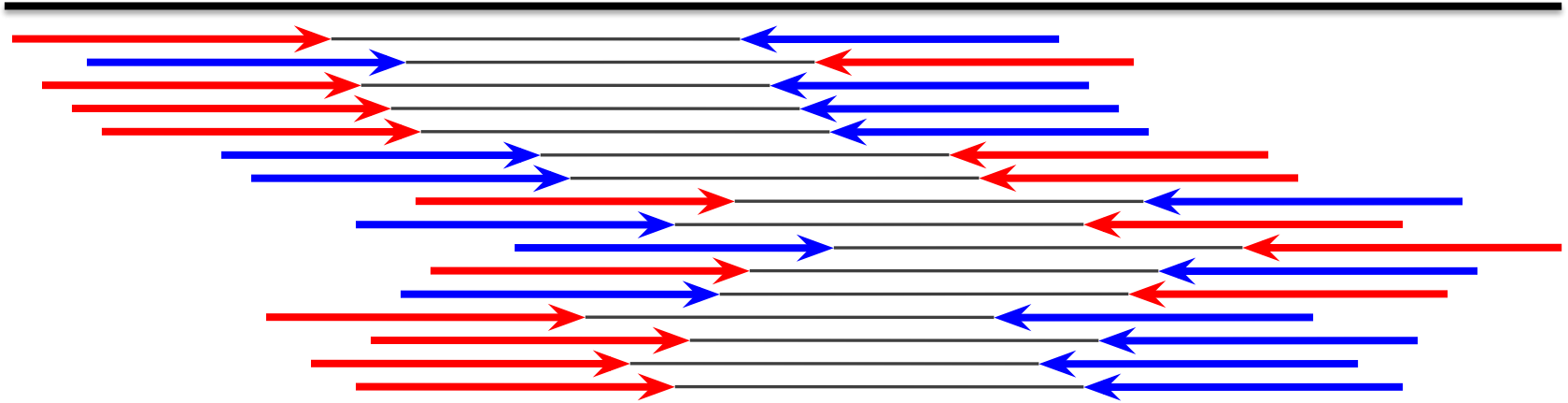






Mapping

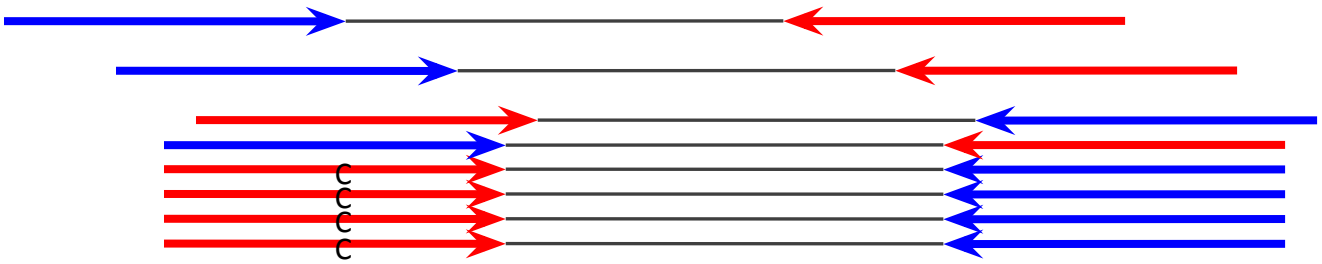
Genome Reference Sequence



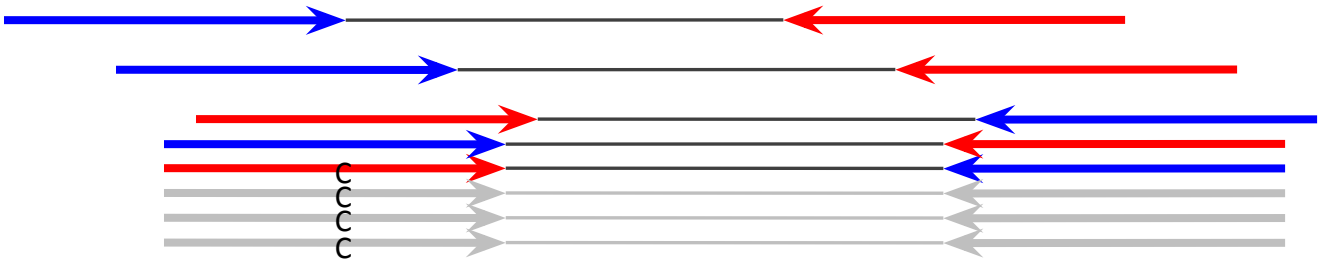
- Single-end reads can be longer, less unique depending on sequence context
- Paired-end reads can span repetitive regions, provide additional information
- Mapping has gotten quite fast, <24 hours for 120 Gbp of sequence
- Split-read alignments are the norm (BWA mem)

Deduplication

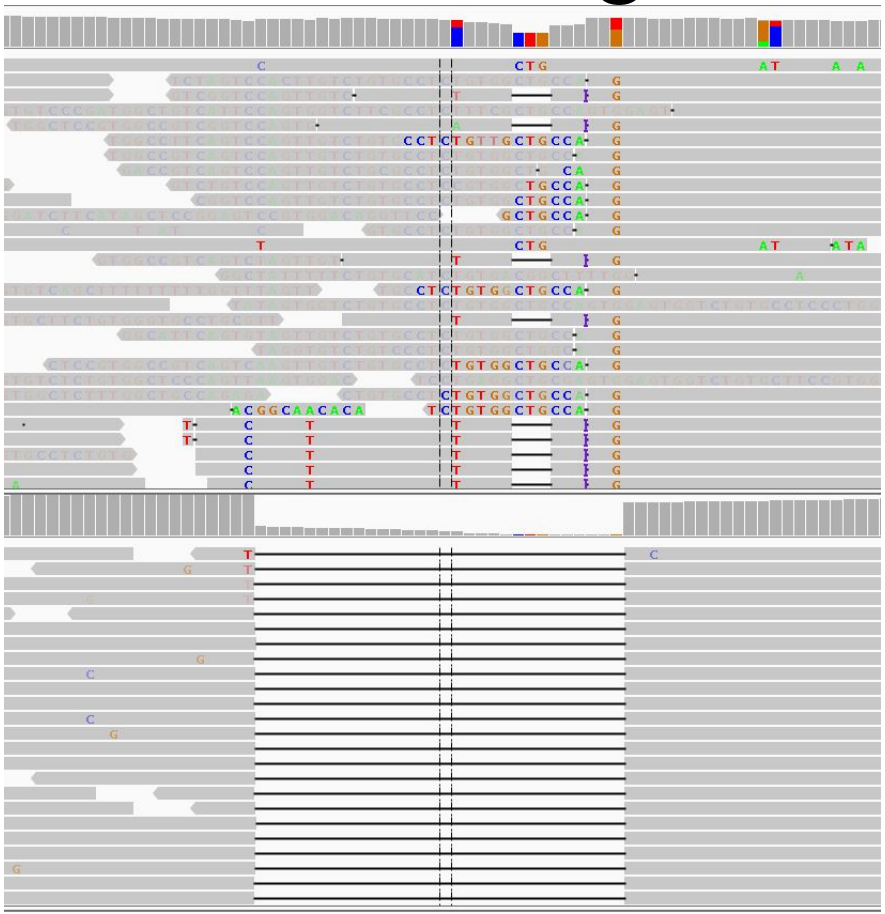
Deduplication



Deduplication



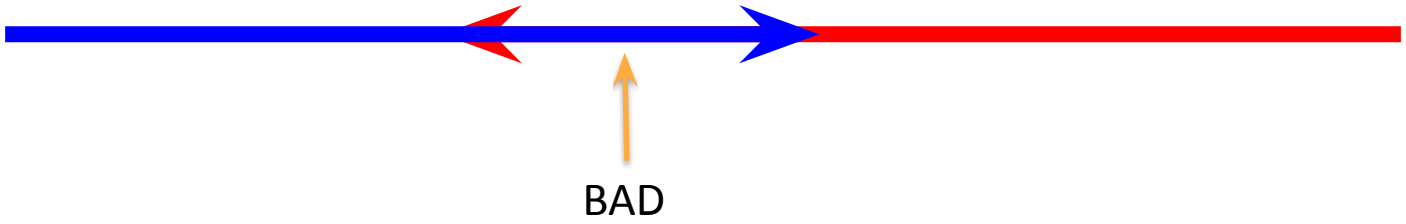
Realignment



https://github.com/mozack/abra

Overlapping reads

Overlapping reads



Overlapping reads

Every aspect of this process is fraught with error

- Base calling is not perfect: *0.5 - 1% error on average*
- Mapping is not perfect*: the reads are short*
- The reference sequence is not perfect

We have a little help

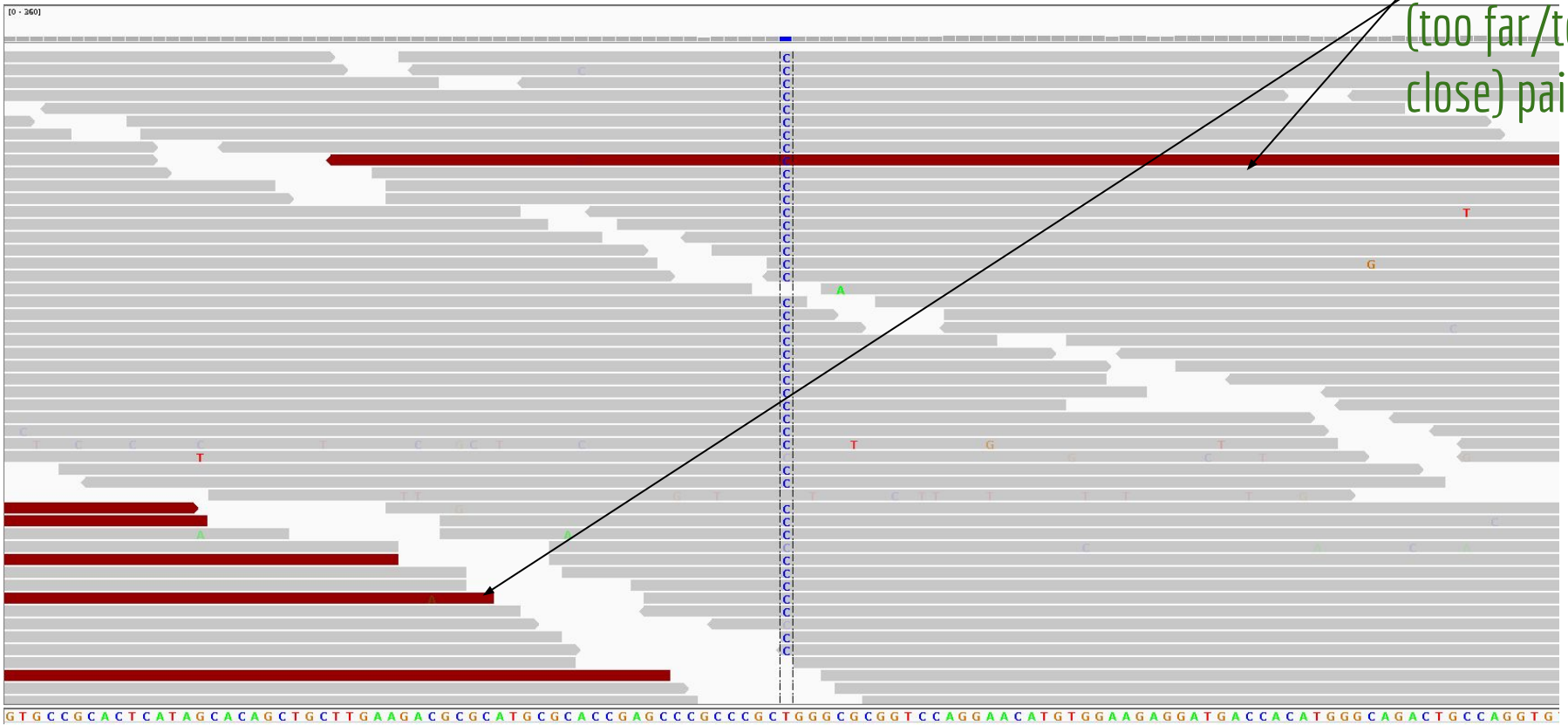
- Some uncertainty is encapsulated in quality scores
  - the rate at which the data is expected to be wrong
- Each base call (ACTGN) comes with a quality
  - Phred-scaled (-10 * $\log_{10}$ of quality)
  - A base call with quality of 20 is wrong 1 out of every 100 times.
- Read mapping has quality too
  - These are also Phred-scaled

Goals of a Variant Caller

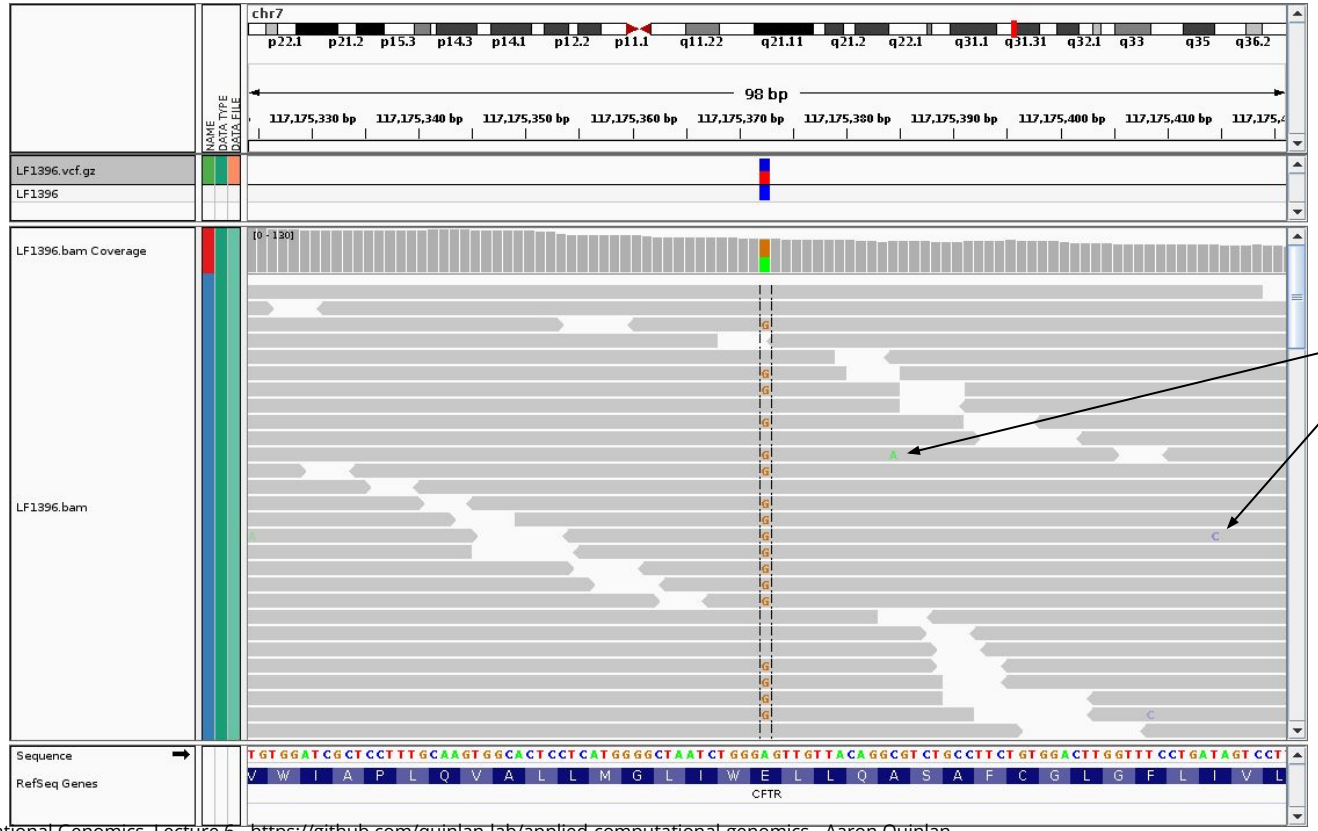
- Sensitively detect mutations
- Precisely detect mutations
  - Confounded by the error we just talked about
  - FDR must be very low as we're looking across a very large space!

    - **An FDR of 0.001 = 3.2 million false positives!**

Homozygous for the "C" allele



Sequencing errors fall out as noise (most of the time)



Sequencing errors

adapted from: Applied Computational Genomics, Lecture 6 - https://github.com/quinlan-lab/applied-computational-genomics - Aaron Quinlan

https://jchoigt.files.wordpress.com/2012/07/igv_e217g_snapshot.png

It is not always so easy

Random versus systematic error

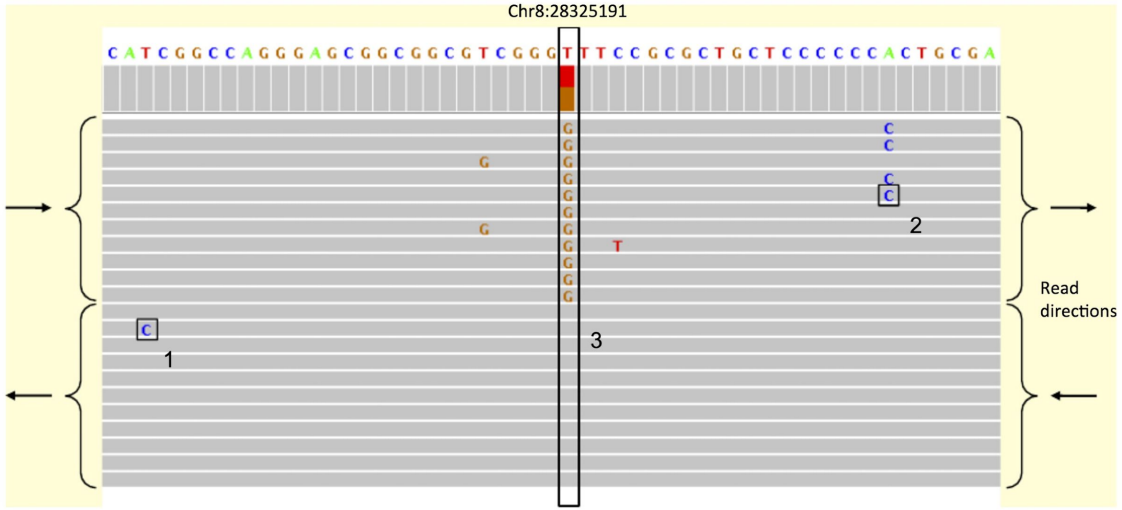
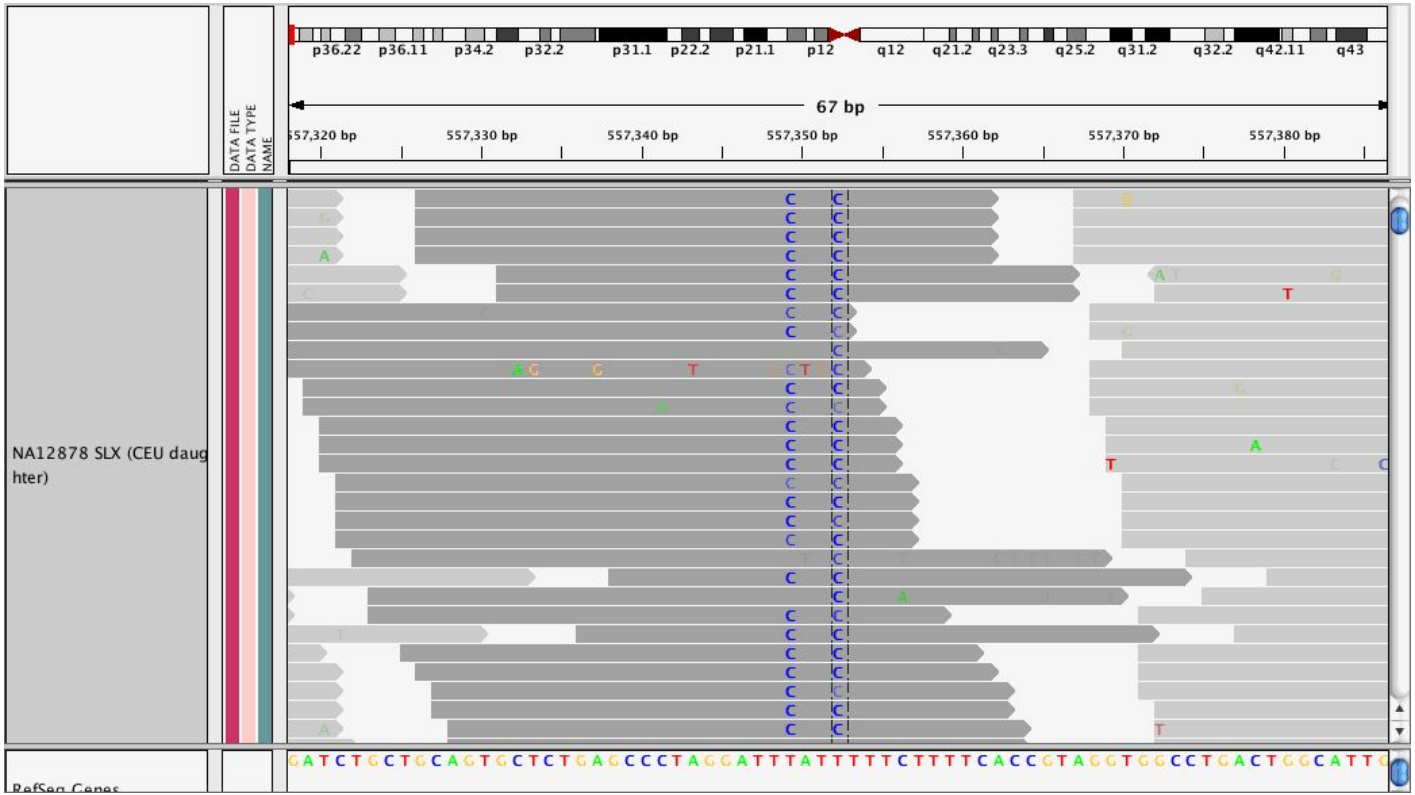


Figure 1 Types of errors. A screenshot from the IGV browser [21] showing three types of error in reads from an Illumina sequencing experiment: (1) A random error likely due to the fact that the *position* is close to the end of the read. (2) Random error likely due to *sequence specific* error- in this case a sequence of Cs are probably inducing errors at the end of the low complexity repeat. (3) *Systematic error*: although it is likely that the GGT sequence motif and the GGC motifs before it created phasing problems leading to the errors, the extent of error is not explained by a random error model. In this case, all the base calls in one direction are wrong as revealed by the 11 overlapping mate-pairs. In particular, all differences from the reference genome are base-call errors, verified by the mate-pair reads, which do not differ from the reference. Given the background error rate, the probability of observing 11 *error-pairs* at a single location, given that 11 mate-pair reads overlap the location, is 1.5×10^{-26}. Moreover, given the presence of such errors at a single location, the probability that all of the errors occur on the same strand (i.e., on the forward mate pair) is $\frac{1}{1024} = 0.00098$. Note that the IGV browser made an incorrect SNP call at the systematic error site (colored bar in top panel).

adapted from: Applied Computational Genomics, Lecture 6 - https://github.com/quinlan-lab/applied-computational-genomics - Aaron Quinlan

https://www.researchgate.net/publication/51814240_Identification_and_correction_of_systematic_error_in_high-throughput_sequence_data

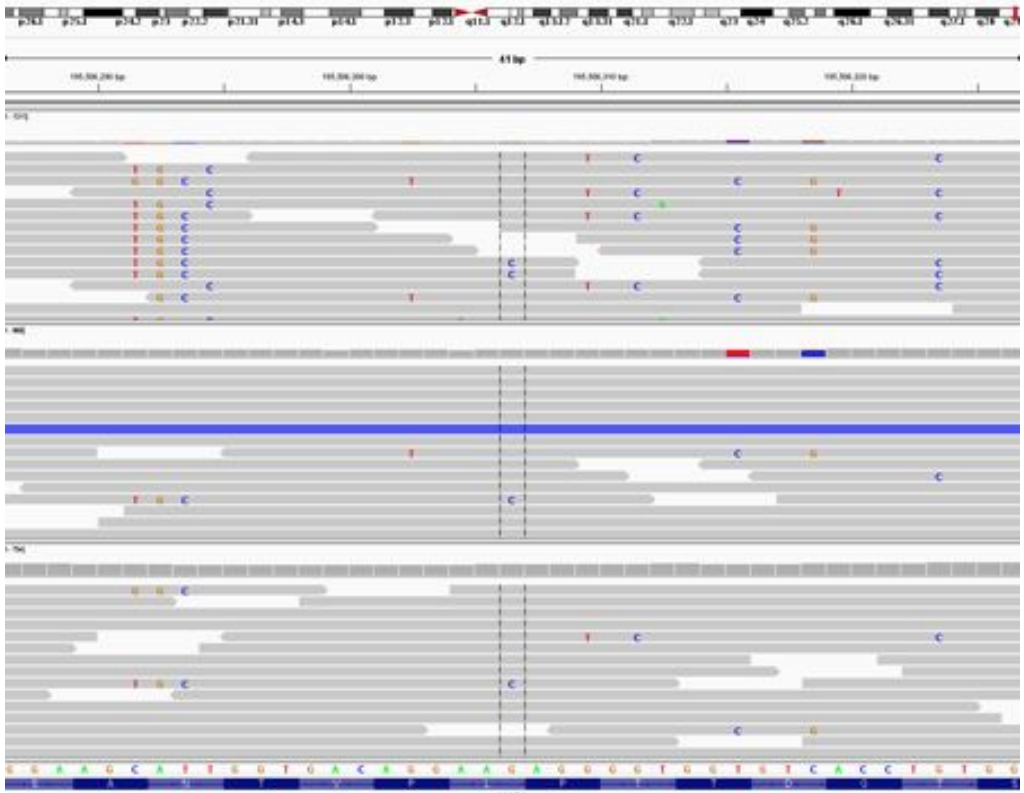
Strand bias from PCR



adapted from: Applied Computational Genomics, Lecture 6 - https://github.com/quinlan-lab/applied-computational-genomics - Aaron Quinlan

http://gatkforums.broadinstitute.org/gatk/discussion/49/using-variant-annotator

Pileups of many differences from paralogy



RESEARCH ARTICLE | OPEN ACCESS

FLAGS, frequently mutated genes in public exomes

Casper Shyr, Maja Tarailo-Graovac, Michael Gottlieb, Jessica JY Lee, Clara van Karnebeek and Wyeth W Wasserman

BMC Medical Genomics 2014 7:64 | DOI: 10.1186/s12920-014-0064-y | © Shyr et al.; licensee BioMed Central Ltd. 2014

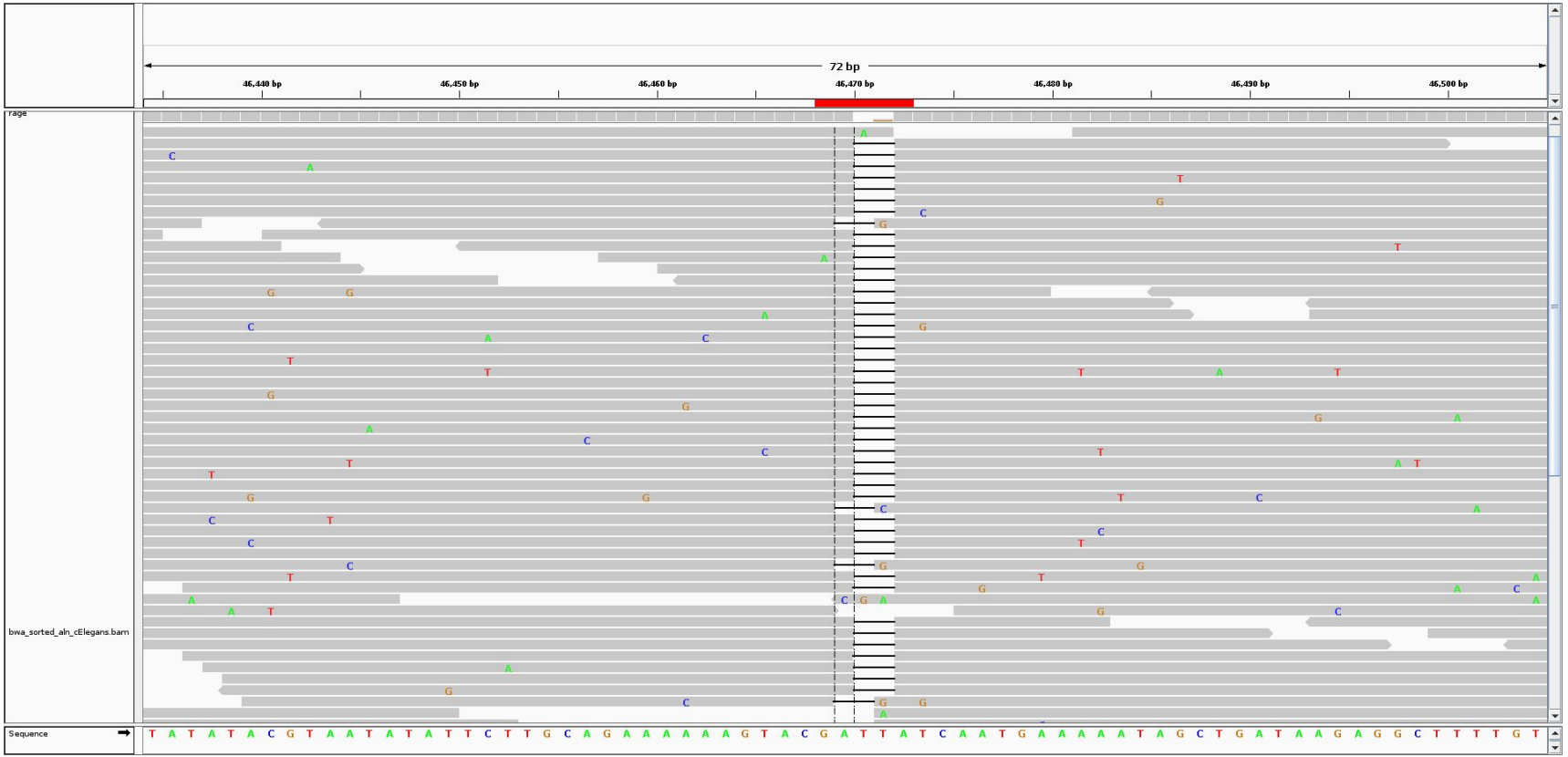
Received: 16 June 2014 | Accepted: 24 October 2014 | Published: 3 December 2014

Open Peer Review reports

adapted from: Applied Computational Genomics, Lecture 6 - https://github.com/quinlan-lab/applied-computational-genomics - Aaron Quinlan

http://massgenomics.org/2013/06/ngs-false-positives.html

Calling INDELs is _much_ harder than SNPs



adapted from: Applied Computational Genomics, Lecture 6 - https://github.com/quinlan-lab/applied-computational-genomics - Aaron Quinlan

INDEL "realignment"

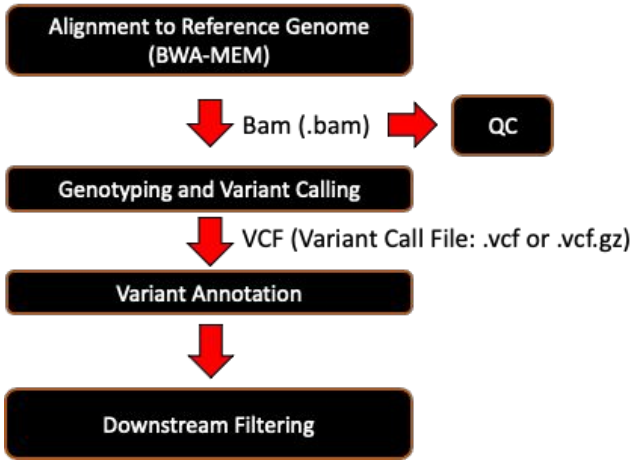
adapted from: Applied Computational Genomics, Lecture 6 - https://github.com/quinlan-lab/applied-computational-genomics - Aaron Quinlan

https://genomebiology.biomedcentral.com/articles/10.1186/gb-2010-11-10-r99

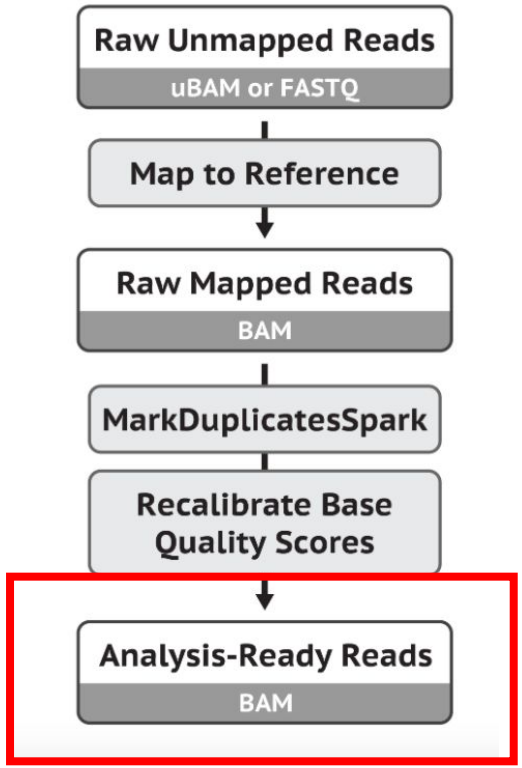
Germline SNV and Indel Calling

How do we identify Germline SNVs and Indels?

1. Align Reads to Reference
2. Call Genotypes
3. Annotate variants
4. Filter Variants
    1. Annotations
        1. Mapping Quality (MQ)
        2. Read Depth (DP)
        3. Genotype Quality (GQ)
    2. Allele Frequency
    3. Region of Interest
5. Final QC
    1. Manually Check Sequence Context (IGV)
        - Homopolymers, Repeat Regions are hard to sequence



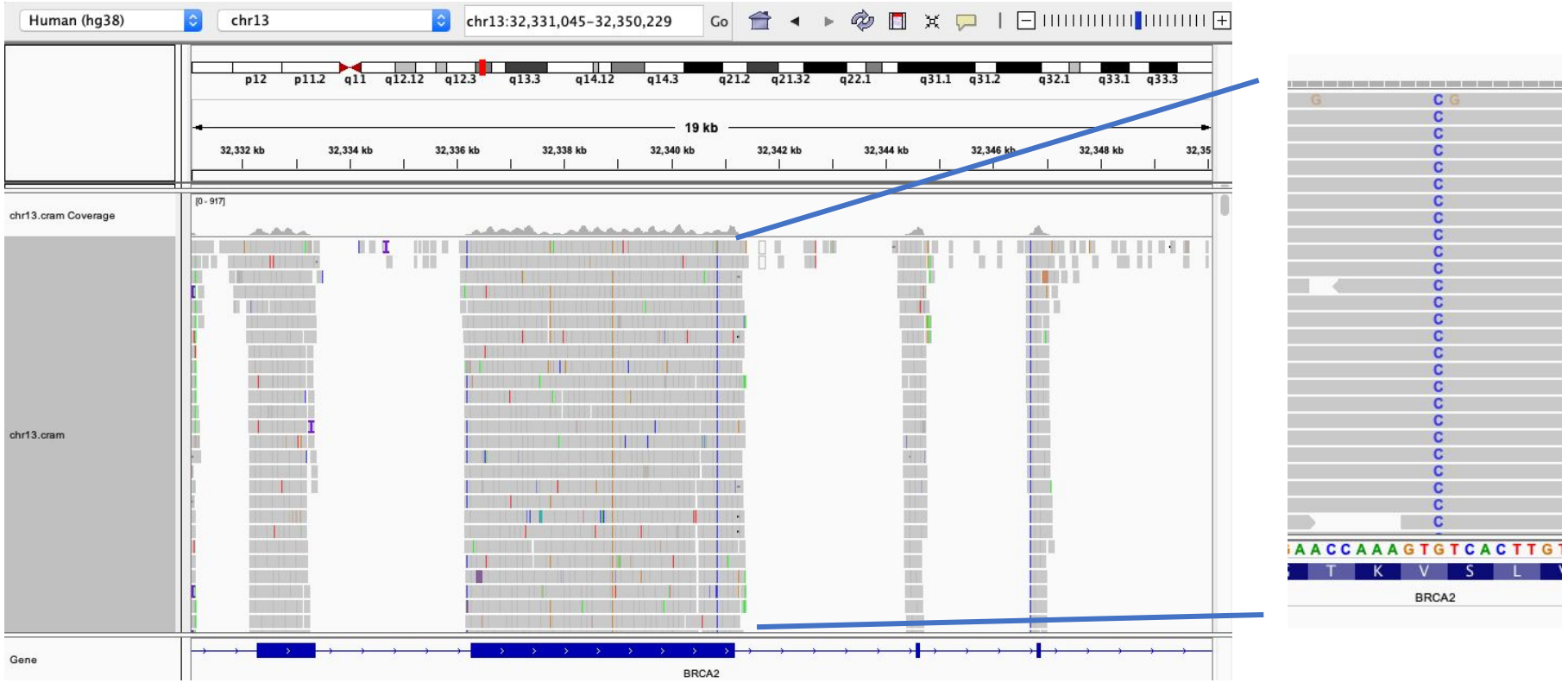
Start with Analysis-Ready Reads



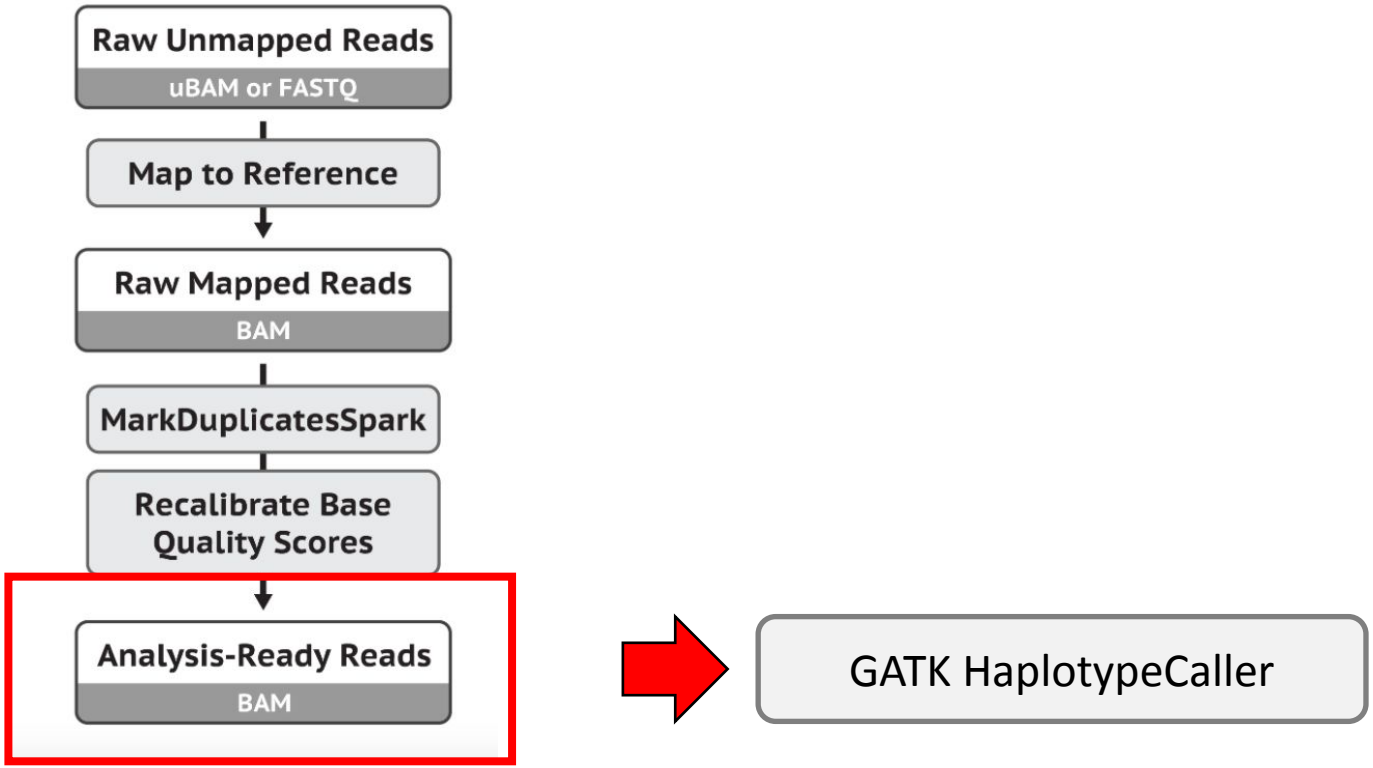
https://gatk.broadinstitute.org/hc/en-us/articles/360035535932-Germline-short-variant-discovery-SNPs-Indels-

Read Alignment Visualization

Exome Sequencing Data from NA12878



Genotype NGS samples with GATK HaplotypeCaller



https://gatk.broadinstitute.org/hc/en-us/articles/360035535932-Germline-short-variant-discovery-SNPs-Indels-

Call Genotypes Using GATK HaplotypeCaller

How HaplotypeCaller works

1. Define active regions

The program determines which regions of the genome it needs to operate on (active regions), based on the presence of evidence for variation.

2. Determine haplotypes by assembly of the active region

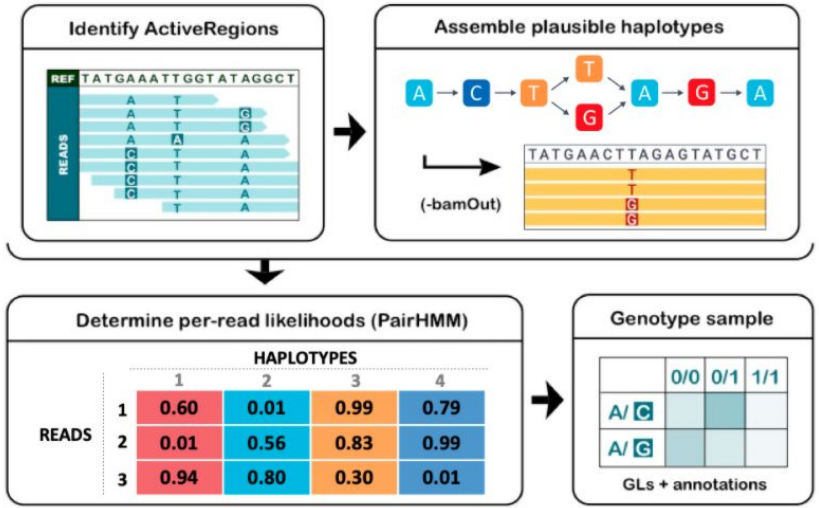
For each active region, the program builds a De Bruijn-like graph to reassemble the active region and identifies what are the possible haplotypes present in the data. The program then realigns each haplotype against the reference haplotype using the Smith-Waterman algorithm in order to identify potentially variant sites.

3. Determine likelihoods of the haplotypes given the read data

For each active region, the program performs a pairwise alignment of each read against each haplotype using the PairHMM algorithm. This produces a matrix of likelihoods of haplotypes given the read data. These likelihoods are then marginalized to obtain the likelihoods of alleles for each potentially variant site given the read data.

4. Assign sample genotypes

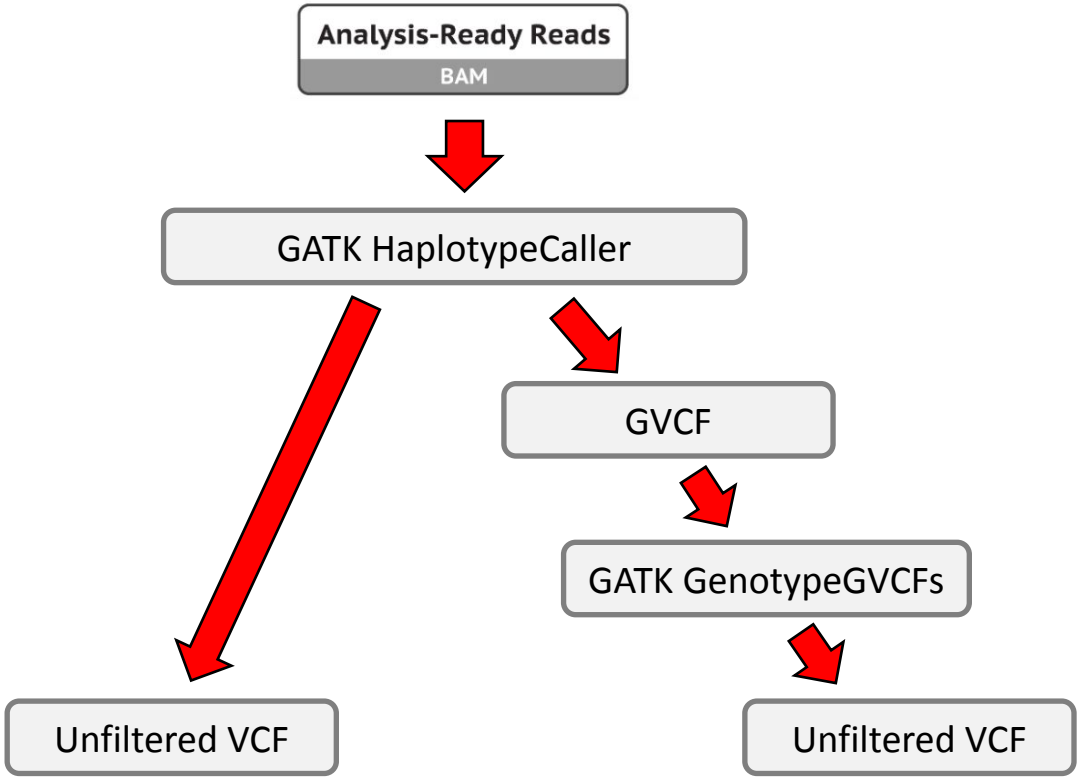
For each potentially variant site, the program applies Bayes' rule, using the likelihoods of alleles given the read data to calculate the likelihoods of each genotype per sample given the read data observed for that sample. The most likely genotype is then assigned to the sample.



https://gatk.broadinstitute.org/hc/en-us/articles/360035531412

https://gatk.broadinstitute.org/hc/en-us/articles/360037225632-HaplotypeCaller

Two Methods for Variant Calling with HaplotypeCaller



Unfiltered VCF Output

header

```
#contig=<ID=HLA-DRB1*01:02:01,length=11229>
#contig=<ID=HLA-DRB1*03:01:01:01,length=13908>
#contig=<ID=HLA-DRB1*03:01:01:02,length=13426>
#contig=<ID=HLA-DRB1*04:03:01,length=15246>
#contig=<ID=HLA-DRB1*07:01:01:01,length=16110>
#contig=<ID=HLA-DRB1*07:01:01:02,length=16120>
#contig=<ID=HLA-DRB1*08:03:02,length=13562>
#contig=<ID=HLA-DRB1*09:21,length=16039>
#contig=<ID=HLA-DRB1*10:01:01,length=13501>
#contig=<ID=HLA-DRB1*11:01:01,length=13921>
#contig=<ID=HLA-DRB1*11:01:02,length=13931>
#contig=<ID=HLA-DRB1*11:04:01,length=13919>
#contig=<ID=HLA-DRB1*12:01:01,length=13404>
#contig=<ID=HLA-DRB1*12:17,length=11260>
#contig=<ID=HLA-DRB1*13:01:01,length=13935>
#contig=<ID=HLA-DRB1*13:02:01,length=13941>
#contig=<ID=HLA-DRB1*14:05:01,length=13933>
#contig=<ID=HLA-DRB1*14:54:01,length=13936>
#contig=<ID=HLA-DRB1*15:01:01:01,length=11080>
#contig=<ID=HLA-DRB1*15:01:01:02,length=11571>
#contig=<ID=HLA-DRB1*15:01:01:03,length=11056>
#contig=<ID=HLA-DRB1*15:01:01:04,length=11056>
#contig=<ID=HLA-DRB1*15:02:01,length=10313>
#contig=<ID=HLA-DRB1*15:03:01:01,length=11567>
#contig=<ID=HLA-DRB1*15:03:01:02,length=11569>
#contig=<ID=HLA-DRB1*16:02:01,length=11005>
#source=HaplotypeCaller
#bcftools_viewVersion=1.10.2-91-g365d117+htslib-1.10.2-109-gdcd4b73
#bcftools_viewCommand=view -r chr13 NA12878-HG001-merged.vcf.gz; Date=Tue Oct 20 08:55:27 2020
```

calls

```
##bcftools_viewCommand=view -r chr13 NA12878-HG001-merged.vcf.gz; Date=Tue Oct 20 08:55:27 2020
#CHROM  POS     ID      REF     ALT     QUAL    FILTER  INFO    FORMAT  NA12878-HG001
chr13   16002338        .       G       A       61.65   .       AC=1;AF=0.5;AN=2;BaseQRankSum=-0.674;DP=4;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=25.51;MQRankSum=0.319;QD=15.41;ReadPosRankSum=-0.319;SOR
chr13   16003734        .       A       G       64.64   .       AC=1;AF=0.5;AN=2;BaseQRankSum=-0.842;DP=6;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=48.15;MQRankSum=0.842;QD=10.77;ReadPosRankSum=-0.842;SOR
chr13   16003747        .       G       T       61.64   .       AC=1;AF=0.5;AN=2;BaseQRankSum=-0.366;DP=7;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=47.07;MQRankSum=1.068;QD=8.81;ReadPosRankSum=0.566;SOR=0
chr13   16003803        .       C       T       113.64  .       AC=1;AF=0.5;AN=2;BaseQRankSum=1.834;DP=6;ExcessHet=3.0103;FS=4.771;MLEAC=1;MLEAF=0.5;MQ=44.96;MQRankSum=-1.834;QD=18.94;ReadPosRankSum=0.842;
chr13   16006318        .       G       A       37.32   .       AC=2;AF=1;AN=2;DP=2;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=22.55;QD=18.66;SOR=0.693    GT:AD:DP:GQ:PL  1/1:0,2:2:6:49,6,0
chr13   16007398        .       G       A       37.32   .       AC=2;AF=1;AN=2;DP=2;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=50;QD=18.66;SOR=0.693     GT:AD:DP:GQ:PL  1/1:0,2:2:6:49,6,0
chr13   16008737        .       T       A       90.84   .       AC=2;AF=1;AN=2;DP=3;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=39.03;QD=30.28;SOR=1.179    GT:AD:DP:GQ:PL  1/1:0,3:3:9:104.9,
chr13   16009860        .       A       G       151.64  .       AC=1;AF=0.5;AN=2;BaseQRankSum=-0.792;DP=7;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=48.97;MQRankSum=0.[illegible]=1.204;SOR=
chr13   16009875        .       A       G,AT    252.06  .       AC=1,1;AF=0.5,0.5;AN=2;DP=7;ExcessHet=3.0103;FS=0;MLEAC=1,1;MLEAF=0.5,0.5;MQ=48.97;QD=25.36;SOR=0.941    GT:AD:DP:GQ:PL  1/2:0,4,3[illegible]7:99:269,10
chr13   16009887        .       CAG     C       61.6    .       AC=1;AF=0.5;AN=2;BaseQRankSum=-0.366;DP=7;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=48.97;MQRankSum=-1.465;QD=8.8;ReadPosRankSum=0;SOR=0.446
chr13   16011213        .       G       A       166.14  .       AC=2;AF=1;AN=2;DP=4;ExcessHet=3.0103;FS=0;MLEAC=2;MLEAF=1;MQ=54.78;QD=28.73;SOR=1.609    GT:AD:DP:GQ:PL  1/1:0,4:4:12:180,12,0
chr13   16011228        .       G       A       166.14  .       AC=2;AF=1;AN=2;DP=4;ExcessHet=3.0103;FS=0;MLEAC=2;MLEAF=1;MQ=54.78;QD=30.97;SOR=1.609    GT:AD:DP:GQ:PL  1/1:0,4:4:12:180,12,0
chr13   16012620        .       C       A       85.14   .       AC=2;AF=1;AN=2;DP=4;ExcessHet=3.0103;FS=0;MLEAC=2;MLEAF=1;MQ=39.56;QD=21.29;SOR=0.693    GT:AD:DP:GQ:PL  1/1:0,4:4:12:99,12,0
chr13   16015532        .       G       GA      67.28   .       AC=2;AF=1;AN=2;DP=3;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=29.58;QD=33.64;SOR=2.303 GT:AD:DP:GQ:PL  1/1:0,2:2:6:79,6,0
chr13   16015554        .       G       A       35.48   .       AC=2;AF=1;AN=2;DP=1;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=40;QD=27.24;SOR=1.609     GT:AD:DP:GQ:PL  1/1:0,1:1:3:45,3,0
chr13   16021251        .       A       G       70.64   .       AC=1;AF=0.5;AN=2;BaseQRankSum=0;DP=4;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=51.66;MQRankSum=-0.674;QD=17.66;ReadPosRankSum=0;SOR=0.69GT:A
chr13   16021268        .       T       A       70.64   .       AC=1;AF=0.5;AN=2;BaseQRankSum=0;DP=5;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=47.54;MQRankSum=-0.674;QD=17.66;ReadPosRankSum=0;SOR=0.69GT:A
chr13   16021728        .       A       G       37.32   .       AC=2;AF=1;AN=2;DP=2;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=27;QD=18.66;SOR=0.693     GT:AD:DP:GQ:PL  1/1:0,2:2:6:49,6,0
chr13   16023773        .       A       T       119.96  .       AC=2;AF=1;AN=2;DP=5;ExcessHet=3.0103;FS=0;MLEAC=2;MLEAF=1;MQ=42.69;QD=23.99;SOR=1.022    GT:AD:DP:GQ:PL  1/1:0,5:5:15:134,15,0
chr13   16042755        .       G       A       70.84   .       AC=2;AF=1;AN=2;DP=3;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=31.93;QD=23.61;SOR=1.179 GT:AD:DP:GQ:PL  1/1:0,3:3:9:84,9,0
chr13   16048306        .       T       A       58.32   .       AC=2;AF=1;AN=2;DP=2;ExcessHet=3.0103;FS=0;MLEAC=1;MLEAF=0.5;MQ=60;QD=29.16;SOR=2.303     GT:AD:DP:GQ:PL  1/1:0,2:2:6:70,6,0
```

Next Step: Filter Raw (Unfiltered) VCF file

- **Problem**: **Too many Variants**
  - Non-related Human's differ by 0.1% --> 3Million SNPs (haploid genome)

- Solution: Filter Low Quality and Common Variants
  - Common Variants > 5%
  - Low Quality
    - GQ < 20, DP < 8, MQ < 40 or GATK VQSR

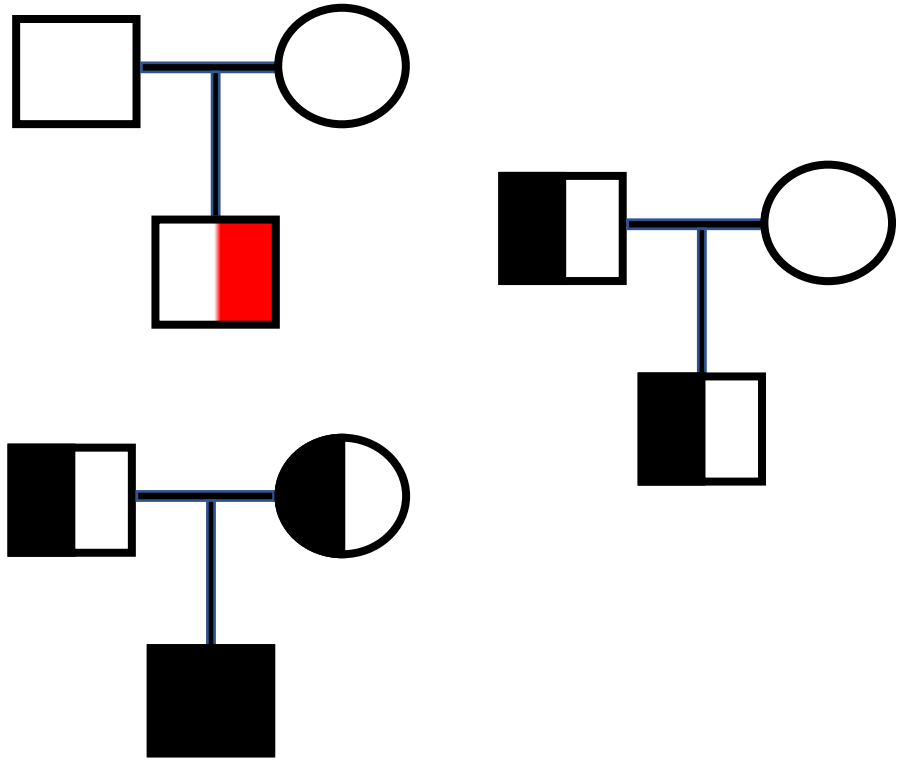
Filtering Criteria

Low Quality	Variant Quality Score Filtering (VQSR) or Quality Annotations	 GQ ≥ 20, DP ≥ 8, MQ ≥ 40
Remove Common Variants	GNOMAD Allele Frequency	GNOMAD_freq < .05
*Select Region	CDS (Whole Exome Sequencing)	chr13

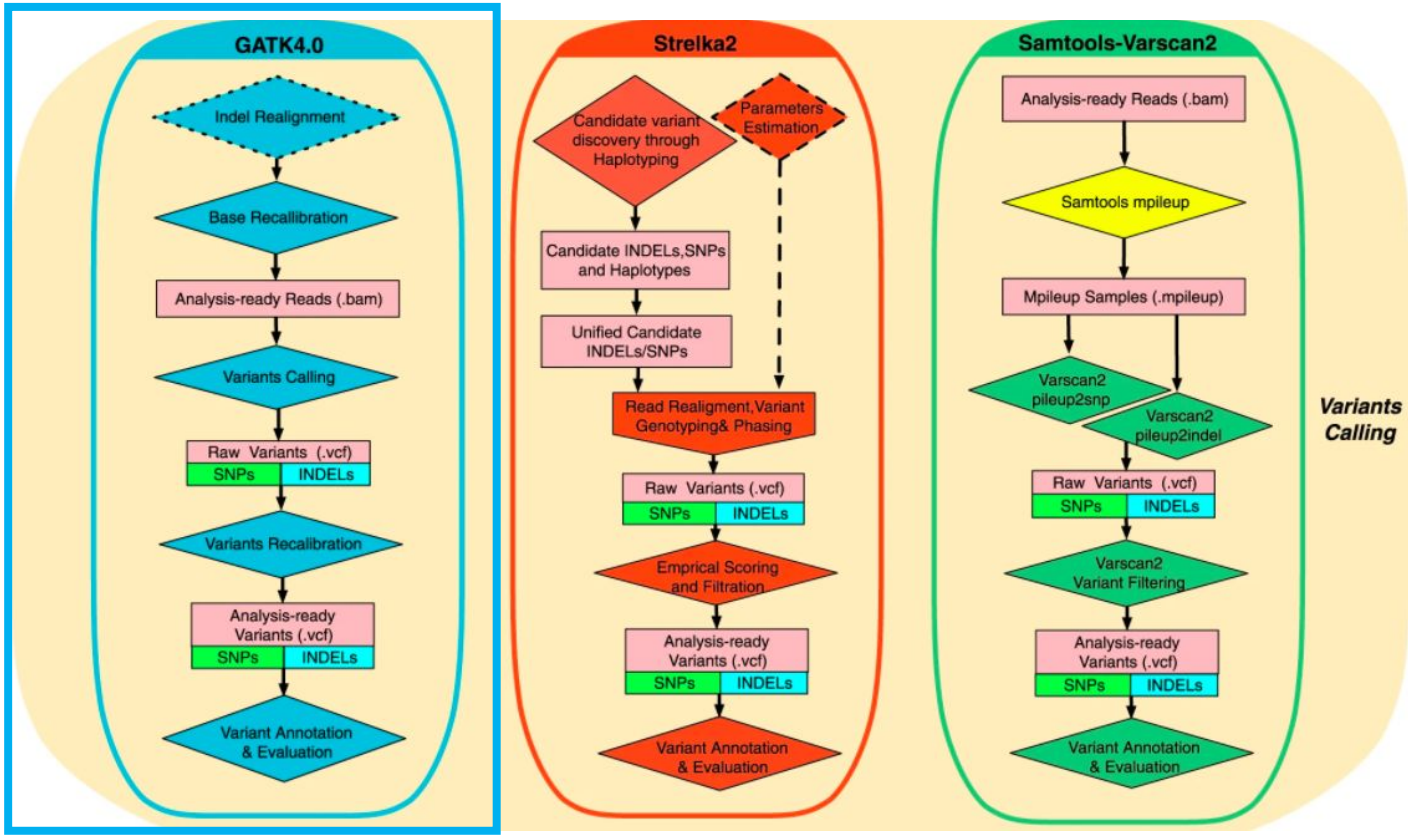
VCF Visualization with IGV

Further Analysis after SNV Calling

- De-novo analysis with trios
- Rare transmitted analysis
- Compound het analysis



Germline SNV Calling Tools



Chen, J., Li, X., Zhong, H. *et al.* Systematic comparison of germline variant calling pipelines cross multiple next-generation sequencers. *Sci Rep* **9**, 9345 (2019). https://doi.org/10.1038/s41598-019-45835-3

GATK4 Docker Image

Use Docker Image with GATK4 Already Installed

- Use GATK4 Docker image: https://hub.docker.com/r/broadinstitute/gatk/
- Advantage over local installation: No need to download multiple programs and worry about incompatibilities
  - Easy to use different versions of GATK4

How to use GATK4

- Use commands in same manner as Linux command line
  - Link for syntax: https://gatk.broadinstitute.org/hc/en-us/articles/360035531892
  - Includes Picard tools

- List all tools: /gatk/gatk --list

Extra Information

GATK4 Installation

Download Files

- See here for guide: https://gatk.broadinstitute.org/hc/en-us/articles/360036194592-Getting-started-with-GATK4
- Download GATK4 jar files: https://github.com/broadinstitute/gatk/releases
- Note: You will need to download a few other programs to run full SNV discovery workflows
  - BWA-MEM
  - SAM-Tools

or

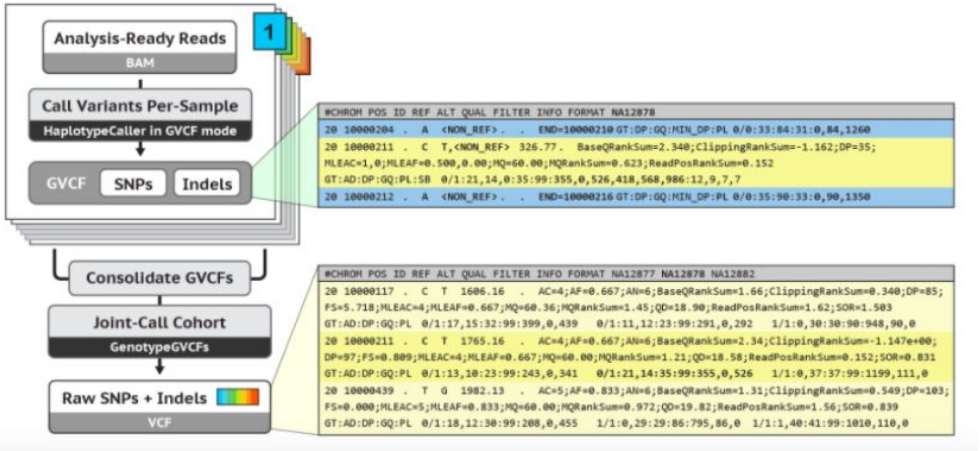
Use Docker Image with GATK4 Already Installed

- Use GATK4 Docker image: https://hub.docker.com/r/broadinstitute/gatk/
- Guide for Local Docker use: https://gatk.broadinstitute.org/hc/en-us/articles/360035889991
  - Compute0 or compute1 will use slightly different syntax
- Advantage over local installation: No need to download multiple programs and worry about incompatibilities

HaplotypeCaller Modes

- HaplotypeCaller Modes
  1. VCF
     - For Single Sample Workflow. No need to run GenotypeGVCFs.
     - No reference confidence call for Genotypes
     - Only produces calls at variant sites
  2. GVCF (Default)
     - Genomic VCF File with condensed non-variant block
     - Produces calls at all sites with compressed non-variant “blocks”
     - Scales well: For joint-analysis and joint-genotyping of large cohorts
     - Best Practices Workflow
  3. GVCF (BP_Resolution)
     - Genomic VCF File with no compression
     - Highest Resolution, but Very large file sizes
- GVCF uses reference model to emit confidence in Genotype call
  - Updates GQ and PL annotation
  - ***Intermediate File**: Must be used with GATK GenotypeGVCFs in order to produce a final vcf

GVCF Format and Information



#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	NA12878-HG001	
chr13	32315314	.	C	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:48:0,48,720
chr13	32315315	.	T	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:48:0,48,720
chr13	32315316	.	C	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:48:0,48,720
chr13	32315317	.	C	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:48:0,48,720
chr13	32315318	.	A	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:48:0,48,720
chr13	32315319	.	G	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:16,1:17:16:0,16,522
chr13	32315320	.	C	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:48:0,48,720
chr13	32315321	.	T	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:48:0,48,720
chr13	32315322	.	C	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:48:0,48,720
chr13	32315323	.	C	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:16,0:16:48:0,48,556
chr13	32315324	.	C	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:16,0:16:48:0,48,556
chr13	32315325	.	G	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:51:0,51,591
chr13	32315326	.	C	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:51:0,51,591
chr13	32315327	.	T	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:51:0,51,591
chr13	32315328	.	T	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:51:0,51,591
chr13	32315329	.	T	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:51:0,51,591
chr13	32315330	.	A	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:51:0,51,591
chr13	32315331	.	T	<NON_REF>	.	.	.	GT:AD:DP:GQ:PL	0/0:17,0:17:51:0,51,591

https://gatk.broadinstitute.org/hc/en-us/articles/360035531812-GVCF-Genomic-Variant-Call-Format

Next Step Annotation

Annotate Variants with known Database Information

Annotation Tools

* Ensembl Variant Effect Predictor (VEP)

ANNOVAR

hail

Databases

gnomAD
genome aggregation database

dbNSFP