Introduction to Short Variant Discovery

Presented by Wes Brashear

14 April 2023

High Performance
Research Computing
DIVISION OF RESEARCH

Types of Variants

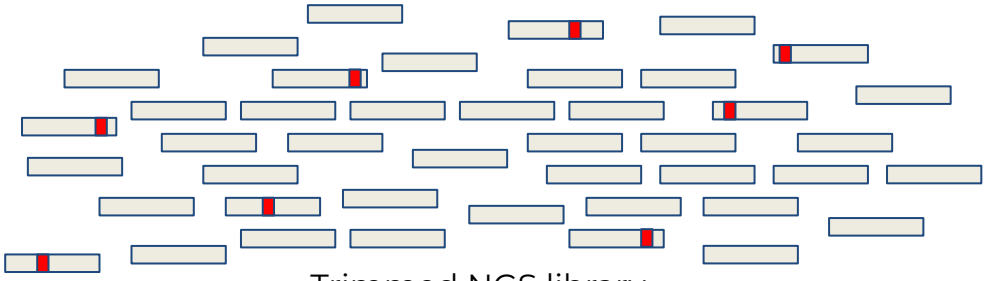
- **Germline** - variants in egg or sperm cells, passed from parent to offspring
- **Somatic** - variants found in specific cells in the body, not hereditary
- **Single nucleotide variant (SNV)**- substitution of one nucleotide for another
- **Single nucleotide polymorphism (SNP)** - SNV present in at least 1% of the population
- **Indel** - small insertion or deletion (<50 bps)

```
Reference:  ACGTGCCAGGACATAATGACA
SNV/SNP:    ACGTGCCAGGTCATAATGACA
Indel:      ACGTGCC-------AATGACA
```

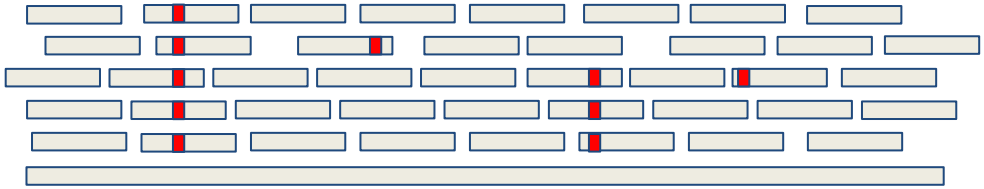
Typical short variant discovery pipeline

1. Trim NGS libraries for adapters and low-quality bases
2. Map reads to a reference genome
3. Process alignment files (e.g. sort, add read groups)
4. Variant calling
5. Downstream analyses (e.g. annotation, population analysis)

https://support.illumina.com/bulletins/2020/12/how-short-inserts-affect-sequencing-performance.html

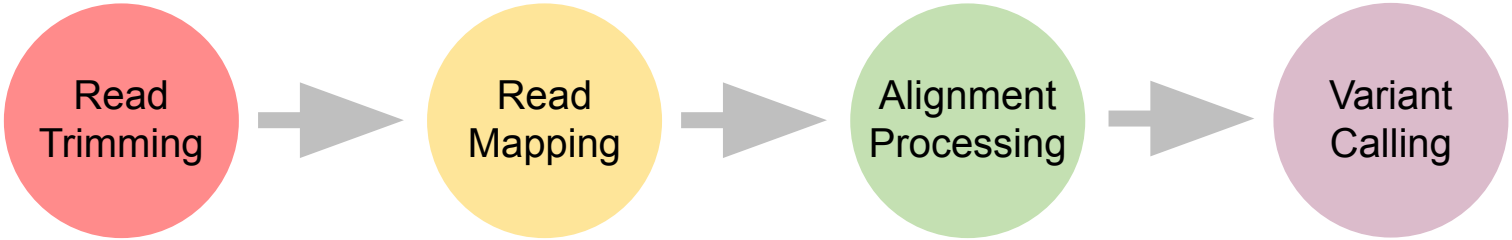


Trimmed NGS library



Mapped to reference genome

Commonly Used Software



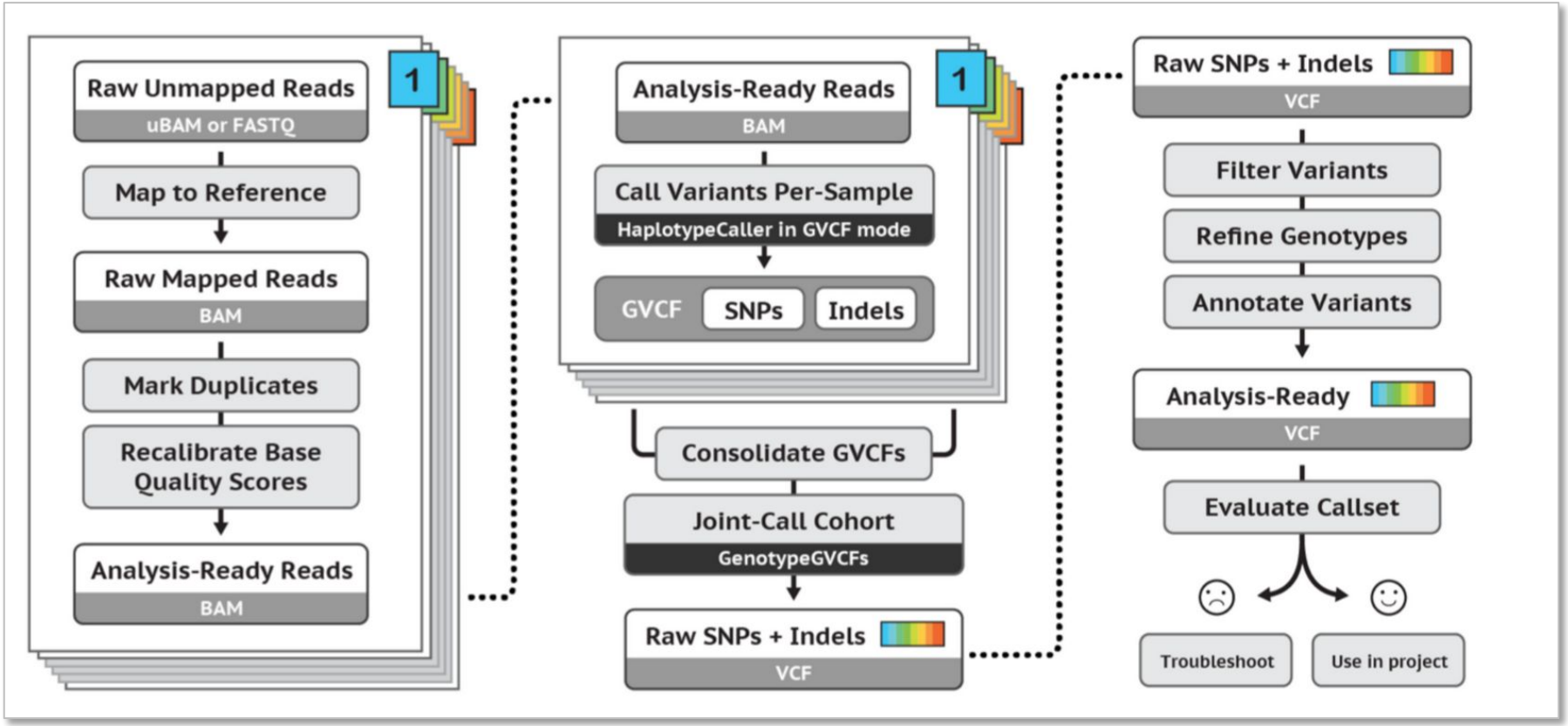
- FastQC
- TrimGalore!
- Cutadapt
- Trimmomatic
- BBDuk

- bwa
- Bowtie2
- BBMap
- bwa-mem2

- SAMtools
- Picard Tools
- BEDtools

- GATK
- FreeBayes
- BCFtools
- Platypus
- VarScan

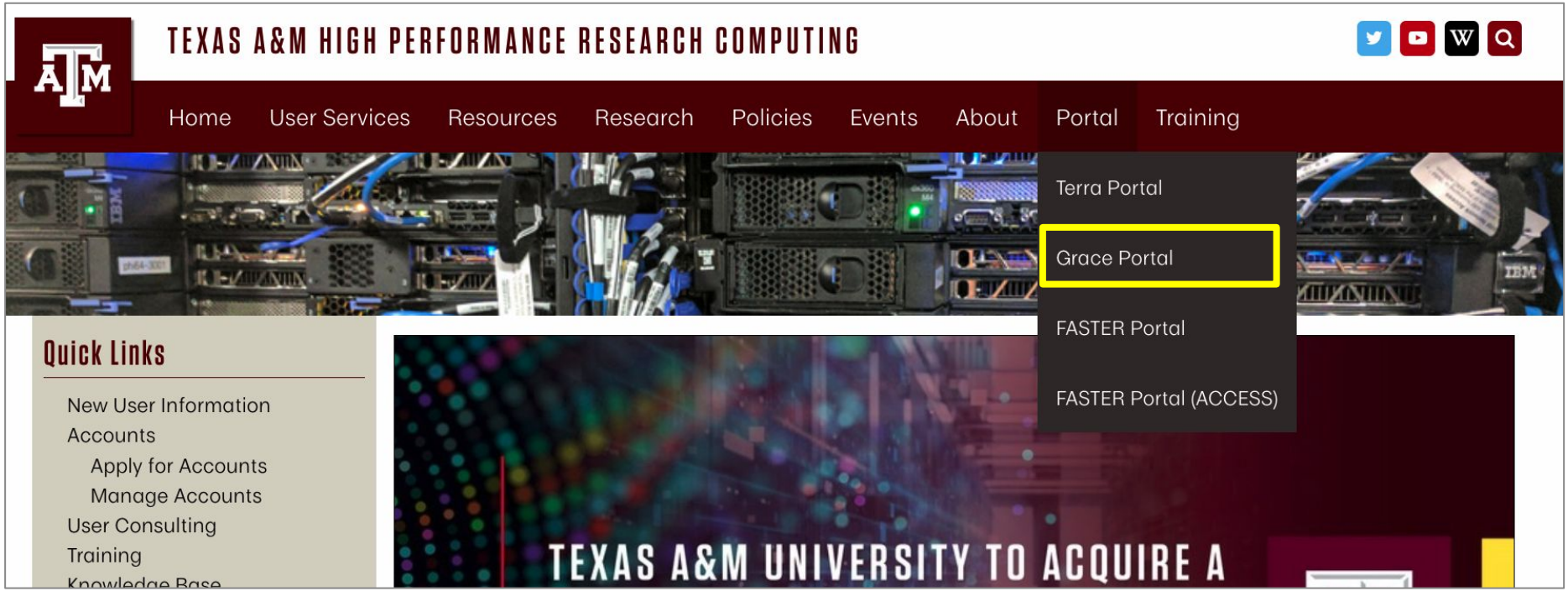
GATK Best Practices



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

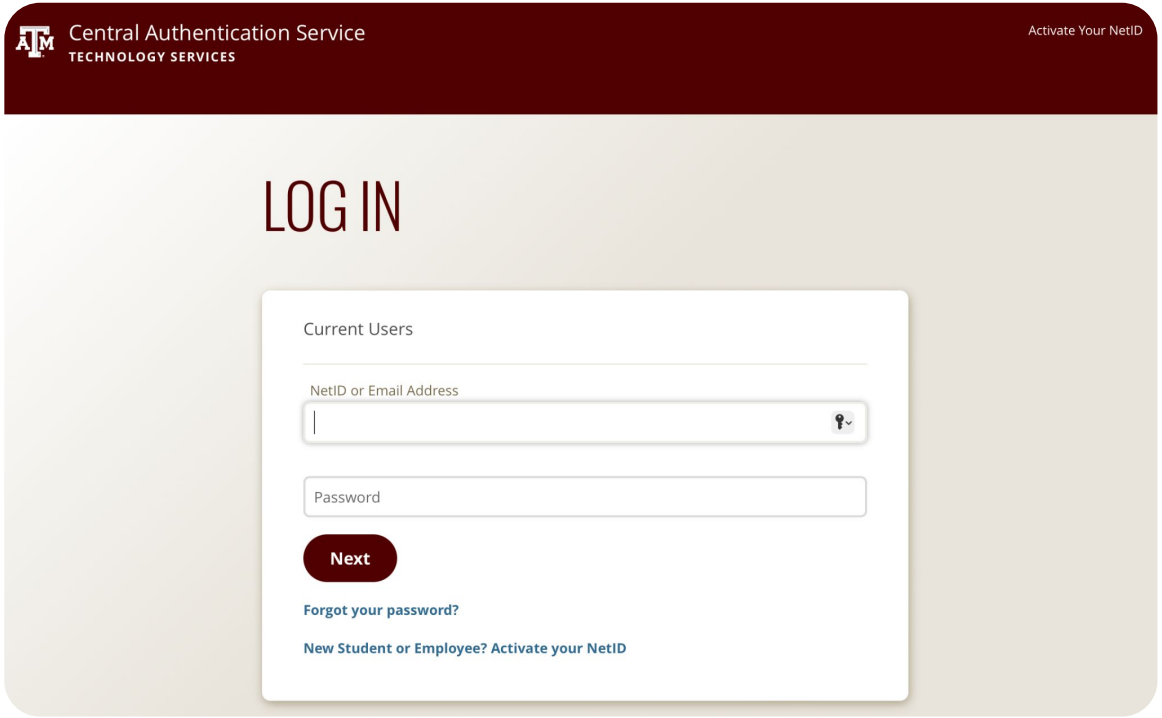
Accessing the HPRC Portal

- HPRC webpage: https://hprc.tamu.edu/



Accessing the HPRC Portal

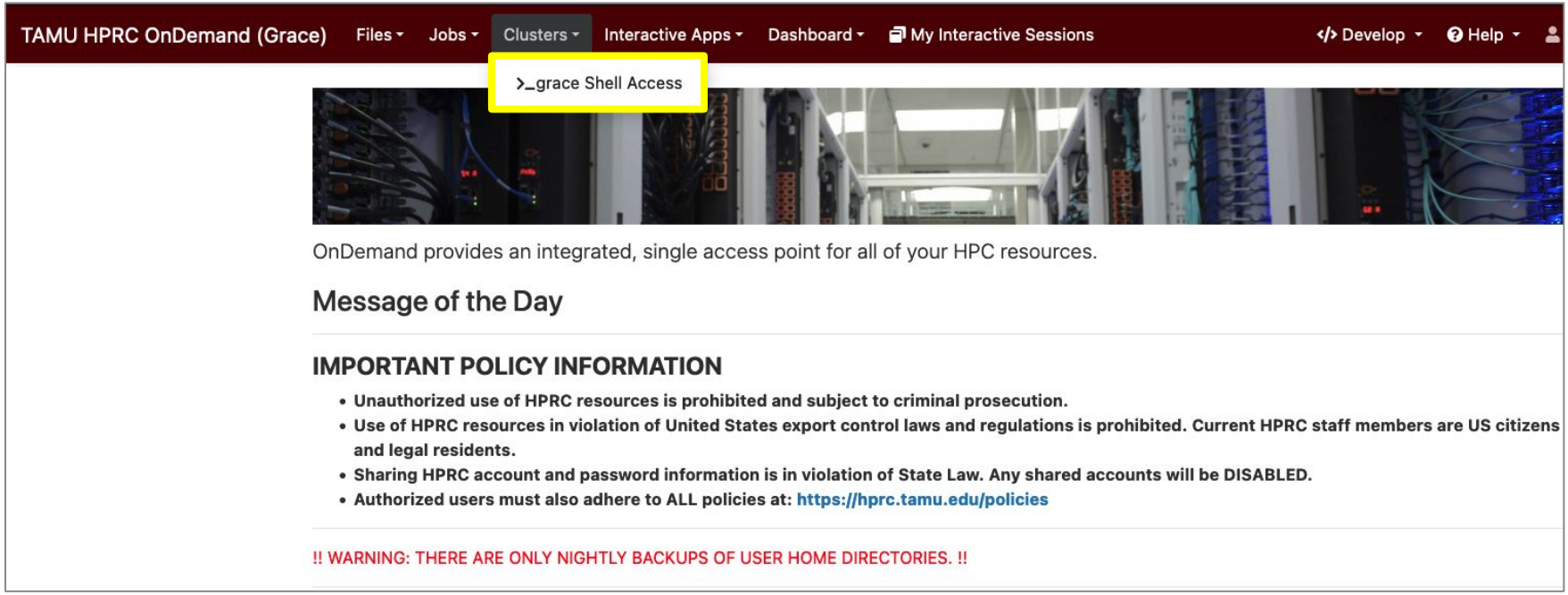
Log-in using your TAMU NetID credentials.



Shell access via the HPRC Portal

Access through (most) web browsers

- Top banner menu: “Clusters” ⇨ “_grace Shell Access”



Finding Bioinformatics Software

- TAMU HPRC Documentation (https://hprc.tamu.edu/wiki/Bioinformatics)
- The following Unix commands allow you to search for software:

```
$ module avail
```

```
$ module spider tool name
```

```
$ module key search phrase
```

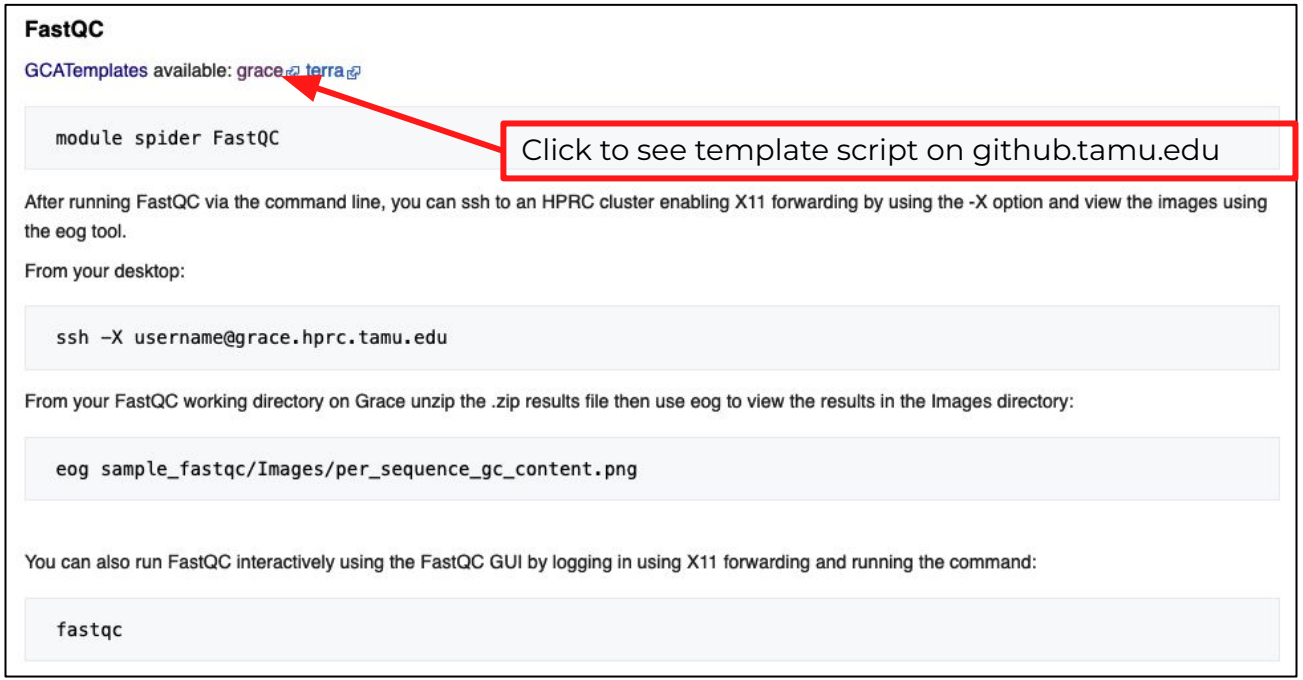
- If you would like a program installed on Grace, send an email with the URL link to help@hprc.tamu.edu

Submitting Jobs on Grace

- Login nodes are not used for large computational jobs
  - File manipulation
  - Job script preparation
  - Short computational jobs (< 60 minutes, 8 cores max)
- Most tasks on HPC require job submission and utilization of compute nodes
- Template job scripts available from HPRC (GCATemplates)

Template Job Scripts

- Available on HPRC wiki page



Template Job Scripts

Latest commit 844f92c on Nov 15, 2021 **History**

0 contributors

Executable File | 44 lines (34 sloc) | 1.9 KB — Raw | Blame

```
#!/bin/bash
#SBATCH --export=NONE               # do not export current env to the job
#SBATCH --job-name=fastqc           # job name
#SBATCH --time=01:00:00             # max job run time dd-hh:mm:ss
#SBATCH --ntasks-per-node=1         # tasks (commands) per compute node
#SBATCH --cpus-per-task=2           # CPUs (threads) per command
#SBATCH --mem=14G                   # total memory per node
#SBATCH --output=stdout.%x.%j       # save stdout to file
#SBATCH --error=stderr.%x.%j        # save stderr to file

module load FastQC/0.11.9-Java-11

<<README
    - FASTQC homepage: http://www.bioinformatics.babraham.ac.uk/projects/fastqc
    - FASTQC manual: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help
README
```

Click “Raw” if you want to copy and paste from your web browser

Sample GCATemplate Job Script (Grace)

```
#!/bin/bash
#SBATCH --export=NONE                # do not export current env to the job
#SBATCH --job-name=fastqc            # job name
#SBATCH --time=01:00:00              # max job run time dd-hh:mm:ss
#SBATCH --ntasks-per-node=1          # tasks (commands) per compute node
#SBATCH --cpus-per-task=2            # CPUs (threads) per command
#SBATCH --mem=14G                    # total memory per node
#SBATCH --output=stdout.%x.%j        # save stdout to file
#SBATCH --error=stderr.%x.%j         # save stderr to file

module load FastQC/0.11.9-Java-11

<<README
    - FASTQC homepage: http://www.bioinformatics.babraham.ac.uk/projects/fastqc
    - FASTQC manual: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help
README

################################### VARIABLES ###################################
# TODO Edit these variables as needed:

########## INPUTS ##########
pe1_1='/scratch/data/bio/GCATemplates/data/miseq/c_dubliniensis/DR34_R1.fastq.gz'
pe1_2='/scratch/data/bio/GCATemplates/data/miseq/c_dubliniensis/DR34_R2.fastq.gz'

######## PARAMETERS ########
threads=$SLURM_CPUS_PER_TASK

########## OUTPUTS #########
output_dir='./'

################################### COMMANDS ####################################
# use -o <directory> to save results to <directory> instead of directory where reads are located
#   <directory> must already exist before using -o <directory> option
# --nogroup will calculate average at each base instead of bins after the first 50 bp
# fastqc runs one thread per file; using 20 threads for 2 files does not speed up the processing

fastqc -t $threads -o $output_dir $pe1_1 $pe1_2
```

```
#!/bin/bash
#SBATCH --export=NONE                # do not export current env to the job
#SBATCH --job-name=fastqc            # job name
#SBATCH --time=01:00:00              # max job run time dd-hh:mm:ss
#SBATCH --ntasks-per-node=1          # tasks (commands) per compute node
#SBATCH --cpus-per-task=2            # CPUs (threads) per command
#SBATCH --mem=14G                    # total memory per node
#SBATCH --output=stdout.%x.%j        # save stdout to file
#SBATCH --error=stderr.%x.%j         # save stderr to file

module load FastQC/0.11.9-Java-11

<<README
    - FASTQC homepage: http://www.bioinformatics.babraham.ac.uk/projects/fastqc
    - FASTQC manual: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help
README

################################### VARIABLES ###################################
# TODO Edit these variables as needed:

########## INPUTS ##########
pe1_1='/scratch/data/bio/GCATemplates/data/miseq/c_dubliniensis/DR34_R1.fastq.gz'
pe1_2='/scratch/data/bio/GCATemplates/data/miseq/c_dubliniensis/DR34_R2.fastq.gz'

######## PARAMETERS ########
threads=$SLURM_CPUS_PER_TASK
```

These parameters are read by the job scheduler

Load the required module(s)

Comment section

Section to edit

```
########## OUTPUTS #########
output_dir='./'
```

Can be edited

```
################################### COMMANDS ###################################
# use -o <directory> to save results to <directory> instead of directory where reads are
located
#   <directory> must already exist before using -o <directory> option
# --nogroup will calculate average at each base instead of bins after the first 50 bp
# fastqc runs one thread per file; using 20 threads for 2 files does not speed up the
processing

fastqc -t $threads -o $output_dir $pe1_1 $pe1_2
```

Command to run the application

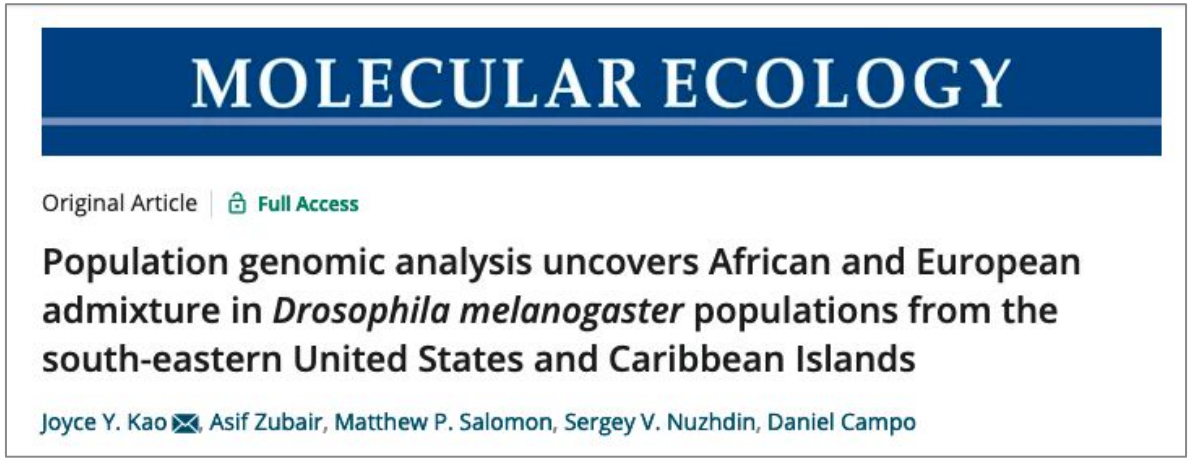
```
################################################################################
<<CITATION
    - Acknowledge TAMU HPRC: https://hprc.tamu.edu/research/citations.html

    - FastQC: http://www.bioinformatics.babraham.ac.uk/projects/fastqc
CITATION
```

GCATemplates - Genomic Computational Analysis Templates

```
$ gcatemplates
```

Example Data

- We will be using example data to complete each step of the full pipeline for short variant discovery. The data comes from a previously published study that examined populations of *Drosophila melanogaster*:



https://www.ncbi.nlm.nih.gov/bioproject/PRJNA274815/

Downloading the data

- Make a directory in your scratch folder and copy the data there:

```
$ mkdir $SCRATCH/ShortVariantCourse
```

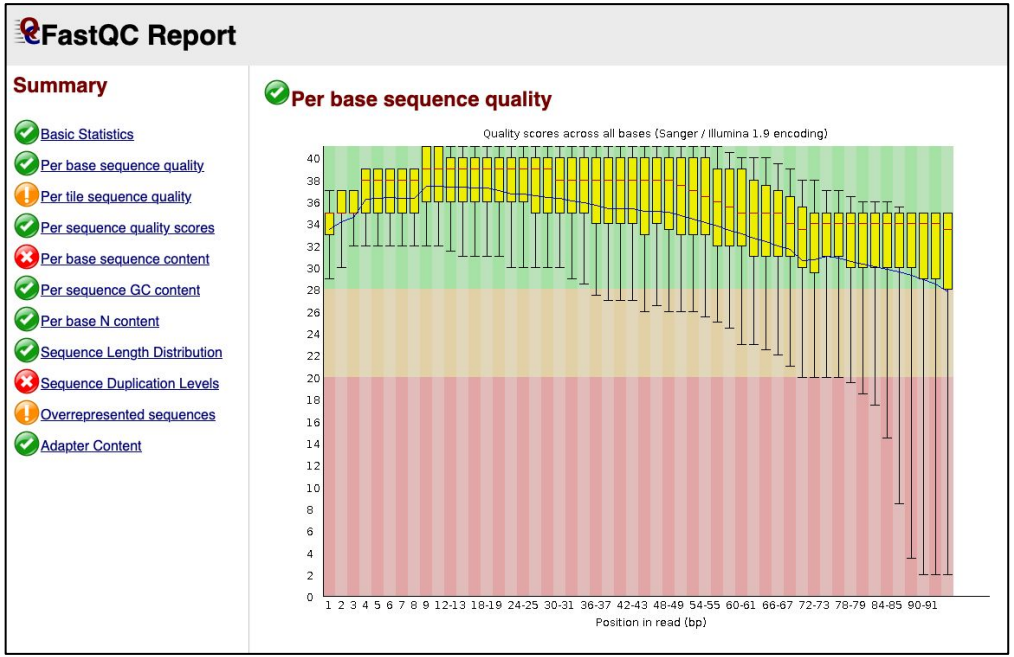
```
$ cd $SCRATCH/ShortVariantCourse
```

```
$ cp -r /scratch/training/bio/svcourse/* .
```

```
$ ls
```

Quality Control

- NGS libraries should be assessed for quality before downstream analyses
  - Adapter content
  - Contamination
  - Run quality
- FastQC allows us to check multiple attributes of NGS libraries

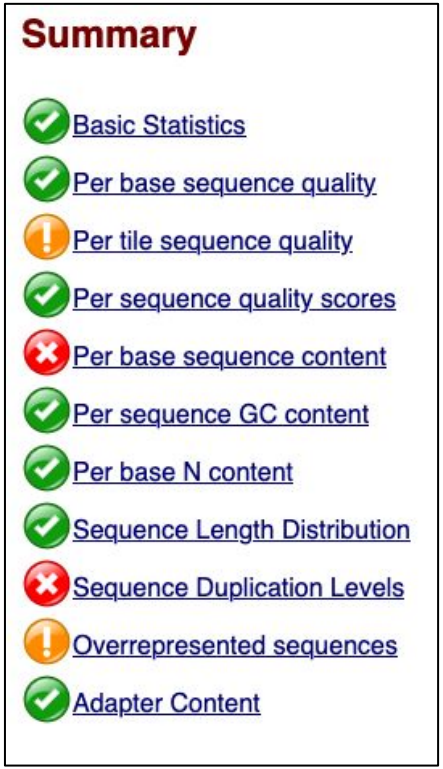


Quality Control

- We will use the program FastQC to evaluate the quality of our libraries
- Use GCATemplates to copy a template job script to your directory (use the menu to find the template for FastQC)
- Modify the script to work with the SRR1820281_subset paired-end reads
- Submit the job script to the compute nodes:

```
$ sbatch run_fastqc_0.11.9_grace.sh
```

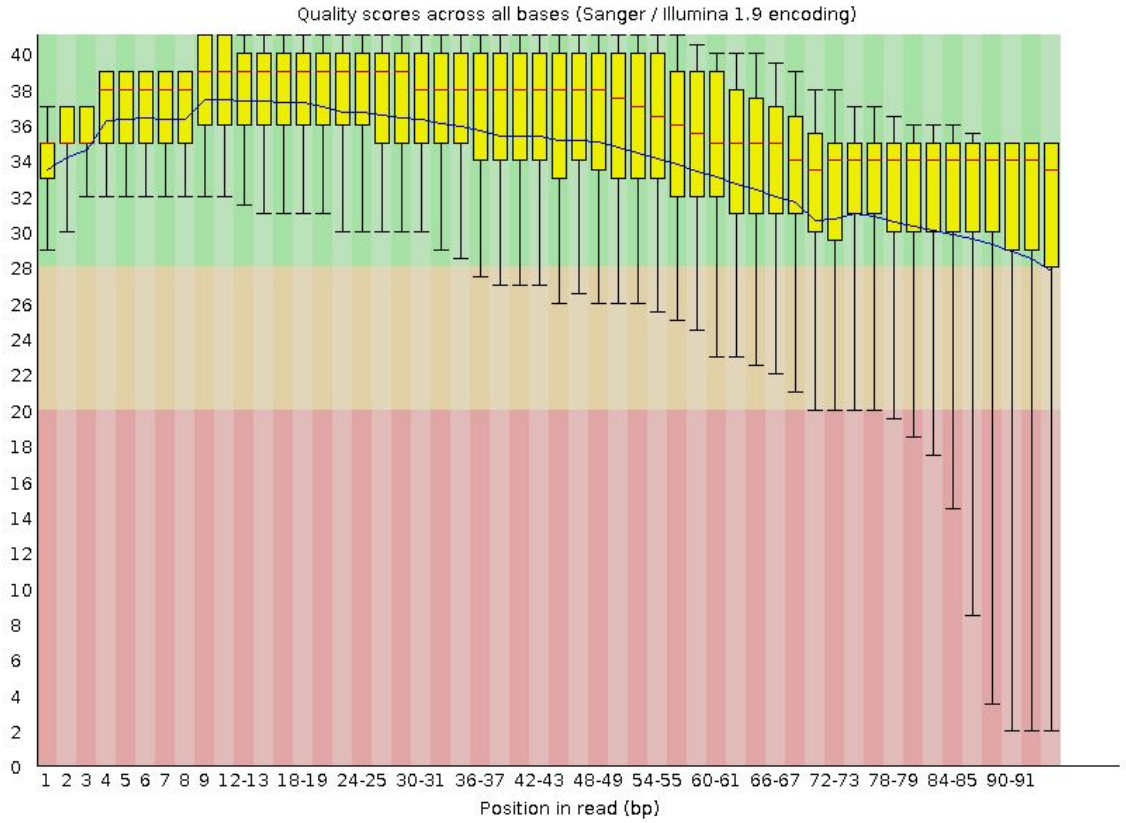
Quality Control

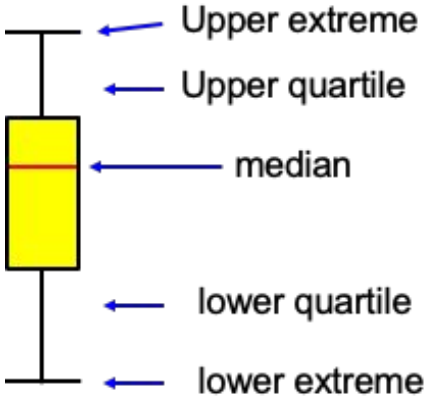


Basic Statistics

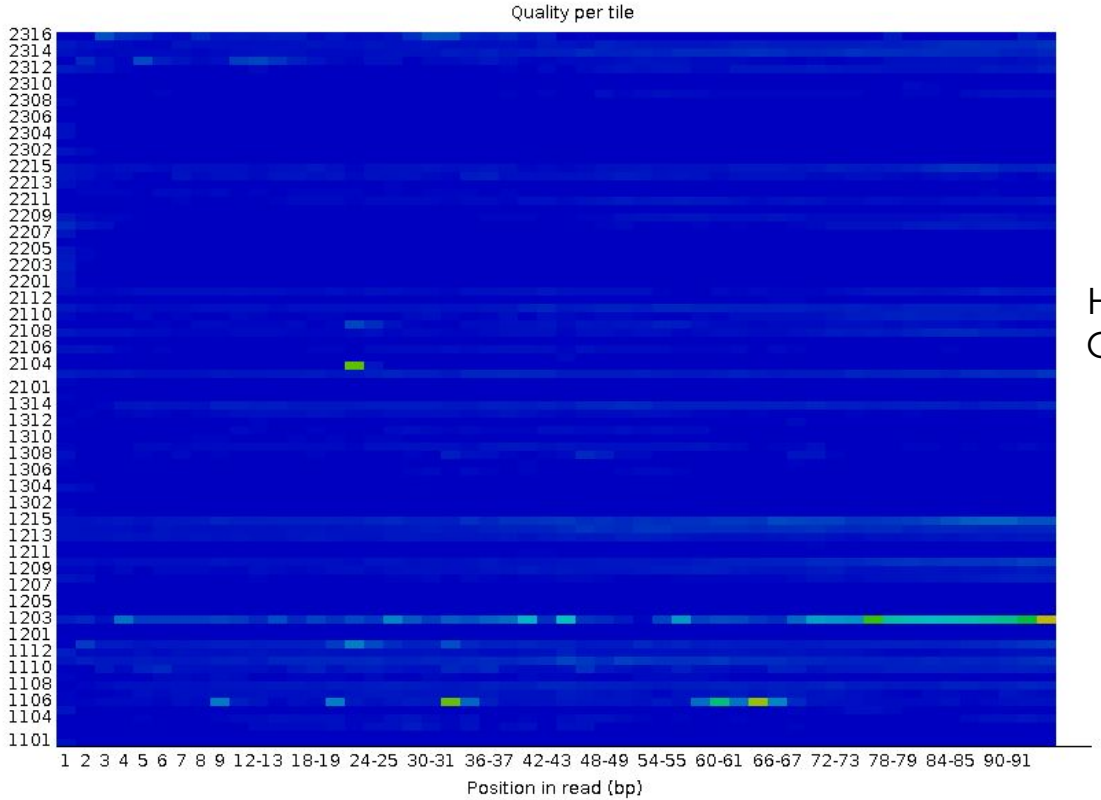
Measure	Value
Filename	SRR1820281_1.fastq.gz
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	13262984
Sequences flagged as poor quality	0
Sequence length	95
%GC	41

Per base sequence quality





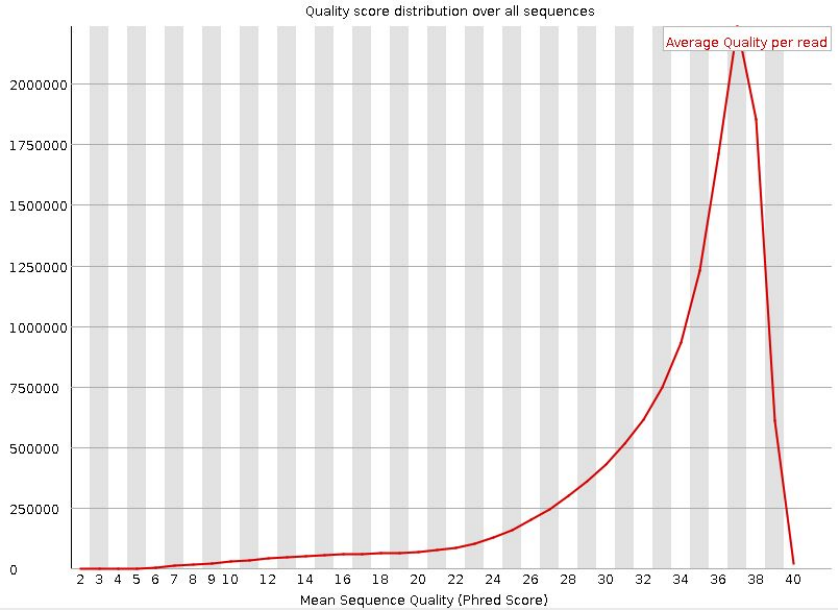
Per tile sequence quality



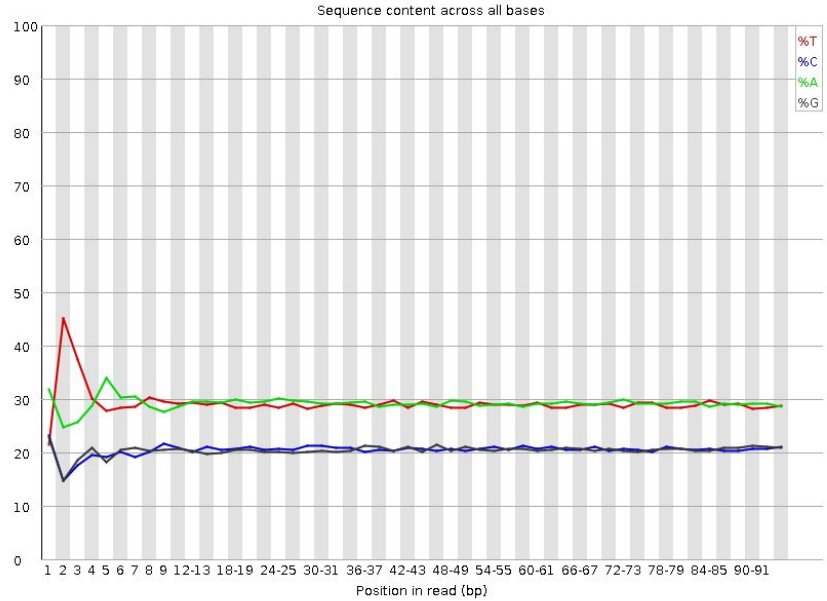
High Quality

Low Quality

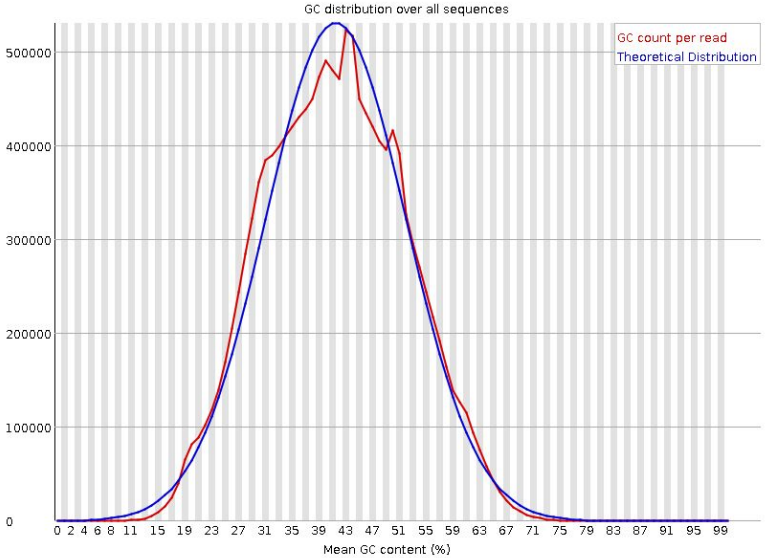
Per sequence quality scores



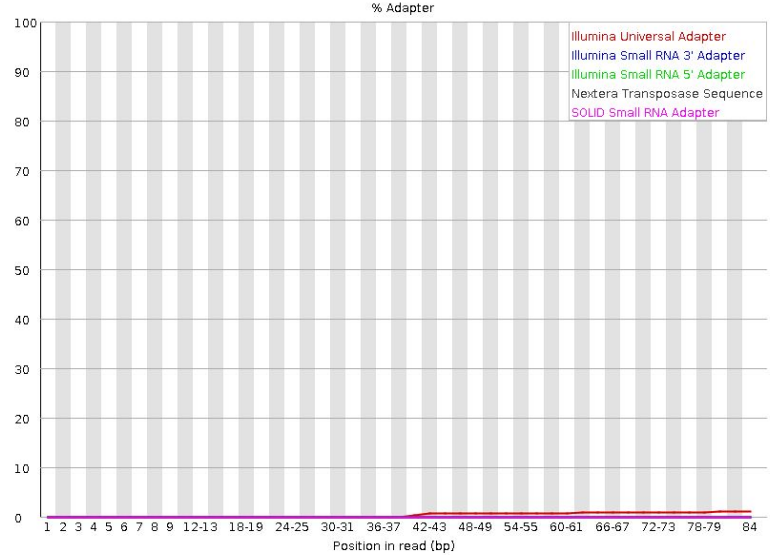
Per base sequence content



Per sequence GC content



Adapter Content



Overrepresented sequences

Sequence	Count	Percentage	Possible Source
GGGGCTTGCTGAGTATTCAGCATACATACCAATATGGGTTTCGTATGCAG	116222	0.8762884732425221	No Hit
TTTGAGATGCTTTCCACGACACAAAACACTCCCCCAAGCTAGATCGGAAG	83511	0.6296546840439528	No Hit
TTTGAGATGCTTTCCACGATTTCTGTCATTGTATATACTTTCGTTTATGA	15746	0.11872139783927961	No Hit

Library Trimming

- We will trim our libraries for adapter sequences and low-quality bases using the TrimGalore! Package
- We will run TrimGalore! interactively on the login nodes

```
$ module purge
```

```
$ module spider trim_galore
```

```
$ module spider Trim_Galore/0.6.7
```

```
$ module load GCCcore/11.2.0 Trim_Galore/0.6.7
```

```
$ trim_galore --paired --fastqc \
   SRR1820281_subset_1.fastq.gz SRR1820281_subset_2.fastq.gz
```

Aligning reads to a reference genome

- Popular NGS alignment software:
  - bwa
  - Bowtie2
  - BBMap
  - bwa-mem2
- Reference genome must be indexed before mapping
  - Only needs to be done once
  - Some genomes already indexed on Grace

```
/scratch/data/bio/genome_indexes/
```

If you would like a genome index added to any of our clusters, send an email to help@hprc.tamu.edu

Aligning reads to a reference genome

We will use bwa-mem2 to align our reads to the genome

```
$ module purge
```

```
$ module load bwa-mem2/2.2.1-Linux64
```

```
$ bwa-mem2 index GCF_000001215.4_genomic.fa
```

```
$ bwa-mem2 mem -t 8 GCF_000001215.4_genomic.fa \
    -o SRR1820281_subset.sam \
    SRR1820281_subset_1_val_1.fq.gz \
    SRR1820281_subset_2_val_2.fq.gz
```

Working with alignment files

We can use SAMtools to look at our mapping statistics and format our alignment file

```
$ module purge
```

```
$ module load GCC/11.3.0 SAMtools/1.16.1
```

```
$ samtools flagstat SRR1820281_subset.sam
```

- We need to sort the alignment and convert it to a binary format

```
$ samtools sort SRR1820281_subset.sam \
   -o SRR1820281_subset_sorted.bam
```

Working with alignment files

We can use picard tools to add read groups and mark duplicates:

```
$ module purge
```

```
$ module load picard/2.25.1-Java-11
```

```
$ java -jar $EBROOTPICARD/picard.jar AddOrReplaceReadGroups \
    -I SRR1820281_subset_sorted.bam \
    -O SRR1820281_subset_sorted_RG.bam \
    -LB SRR1820281 -PL Illumina -PU HWI-ST550_0199.3 -SM SRR1820281
```

```
$ java -jar $EBROOTPICARD/picard.jar MarkDuplicates \
    -I SRR1820281_subset_sorted_RG.bam \
    -O SRR1820281_subset_sorted_RG_DM.bam \
    -M SRR1820281_subset_metrics.txt
```

Creating additional files for GATK

We also need to prepare index files for our alignments and dictionary files for our reference genome:

```
$ module purge
```

```
$ module load GCCcore/11.2.0 GCC/11.2.0 GATK/4.2.6.1-Java-11 SAMtools/1.14
```

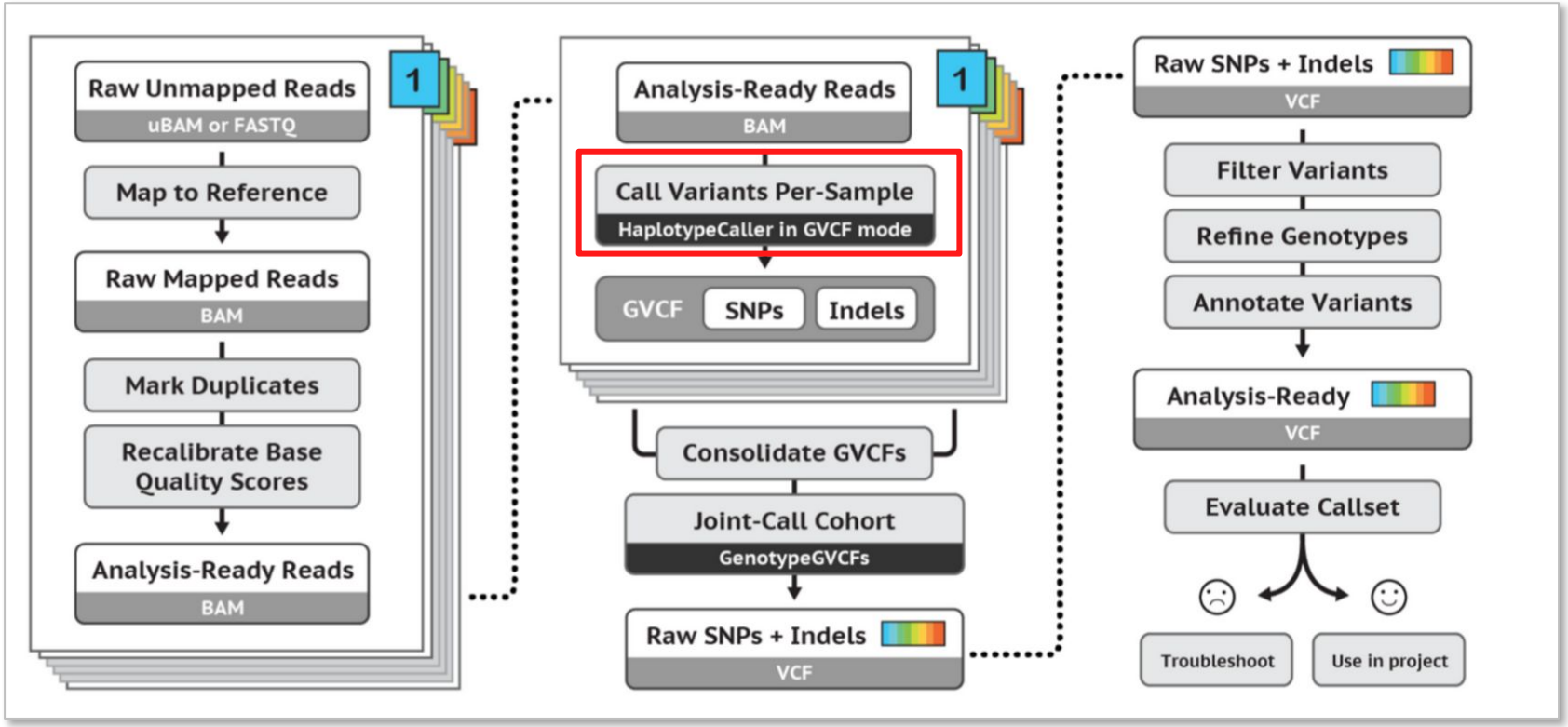
```
$ gatk CreateSequenceDictionary -R GCF_000001215.4_genomic.fa
```

- Use SAMtools to index the genome and bam files:

```
$ samtools faidx GCF_000001215.4_genomic.fa
```

```
$ samtools index SRR1820281_subset_sorted_RG_DM.bam
```

Running GATK



Running HaplotypeCaller (in GVCF mode)

```
$ gatk HaplotypeCaller -I SRR1820281_subset_sorted_RG_DM.bam \
    -R GCF_000001215.4_genomic.fa \
    -O SRR1820281_subset.g.vcf.gz \
    -ERC GVCF
```

- Running in parallel requires “L” argument (separate jobs by genomic intervals)
- This will take ~7 minutes

Consolidating GVCF Files

- GenomicsDBImport replaced CombineGVCFs
- Merges GVCFs from multiple samples

```
$ mkdir ConsolGVCFs && cd ConsolGVCFs
```

```
$ cp ../support/*vcf.gz* .
```

```
$ cp ../GCF_000001215.4_genomic.interval_list .
```

Consolidating GVCF Files

- Create the script shown on the right
- Save it as Consol.sh
- Submit the job to the compute nodes:

```
$ sbatch Consol.sh
```

- This will take about an hour to run, so we will move on with previously generated data

```
#!/bin/bash
#SBATCH --job-name=ConsolidateGVCFs
#SBATCH --time=02:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=2
#SBATCH --mem=4G
#SBATCH --output=%x%j.output
#SBATCH --error=%x%j.error

module purge
module load GCCcore/11.2.0 GATK/4.2.6.1-Java-11

gatk GenomicsDBImport -V SRR1820281.g.vcf.gz \
     -V SRR1820288.g.vcf.gz \
     -V SRR1820302.g.vcf.gz \
     --genomicsdb-workspace-path SVC_db \
     -L GCF_000001215.4_genomic.interval_list
```

Joint Genotyping with GATK

```
$ mkdir ../GenotypingGVCFs
```

```
$ cd ../GenotypingGVCFs
```

- Create the script shown on the right
- Save it as Genotype.sh
- Submit the job to the compute nodes

```
#!/bin/bash
#SBATCH --job-name=GenotypeGVCFs
#SBATCH --time=04:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=2
#SBATCH --mem=64G
#SBATCH --output=%x%j.output
#SBATCH --error=%x%j.error

module purge
module load GCCcore/11.2.0 GATK/4.2.6.1-Java-11

gatk GenotypeGVCFs -R ../GCF_000001215.4_genomic.fa \
     -V gendb:///scratch/training/bio/SVC_db \
     -O SVC.vcf.gz
```

The VCF Format

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

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                                  …
2          4370        rs6057      G     A           29    .          NS=2;DP=13;AF=0.5;DB;H2
2          7330        .           T     A           3     q10        NS=5;DP=12;AF=0.017
2          110696      rs6055      A     G,T         67    PASS       NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2          130237      .           T     .           47    .          NS=2;DP=16;AA=T
```

Header

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                              …
2       4370    rs6057    G    A      29    .       NS=2;DP=13;AF=0.5;DB;H2
2       7330    .         T    A      3     q10     NS=5;DP=12;AF=0.017
2       110696  rs6055    A    G,T    67    PASS    NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2       130237  .         T    .      47    .       NS=2;DP=16;AA=T
```

Chromosome

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                              ...
2       4370    rs6057    G    A      29    .       NS=2;DP=13;AF=0.5;DB;H2
2       7330    .         T    A      3     q10     NS=5;DP=12;AF=0.017
2       110696  rs6055    A    G,T    67    PASS    NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2       130237  .         T    .      47    .       NS=2;DP=16;AA=T
```

Nucleotide Position

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                              ...
2       4370    rs6057    G    A      29    .       NS=2;DP=13;AF=0.5;DB;H2
2       7330    .         T    A      3     q10     NS=5;DP=12;AF=0.017
2       110696  rs6055    A    G,T    67    PASS    NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2       130237  .         T    .      47    .       NS=2;DP=16;AA=T
```

SNP ID

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                                  …
2          4370       rs6057    G     A          29    .          NS=2;DP=13;AF=0.5;DB;H2
2          7330       .         T     A          3     q10        NS=5;DP=12;AF=0.017
2          110696     rs6055    A     G,T        67    PASS       NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2          130237     .         T     .          47    .          NS=2;DP=16;AA=T
```

Reference Allele

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                              …
2       4370    rs6057    G    A      29    .       NS=2;DP=13;AF=0.5;DB;H2
2       7330    .         T    A      3     q10     NS=5;DP=12;AF=0.017
2       110696  rs6055    A    G,T    67    PASS    NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2       130237  .         T    .      47    .       NS=2;DP=16;AA=T
```

Alternative Allele

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                              ...
2         4370      rs6057    G     A       29    .         NS=2;DP=13;AF=0.5;DB;H2
2         7330      .         T     A       3     q10       NS=5;DP=12;AF=0.017
2         110696    rs6055    A     G,T     67    PASS      NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2         130237    .         T     .       47    .         NS=2;DP=16;AA=T
```

Phred-scaled quality score

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                                  …
2          4370       rs6057     G     A          29     .         NS=2;DP=13;AF=0.5;DB;H2
2          7330       .          T     A          3      q10       NS=5;DP=12;AF=0.017
2          110696     rs6055     A     G,T        67     PASS      NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2          130237     .          T     .          47     .         NS=2;DP=16;AA=T
```

Filter or Pass (“.” means no filter applied)

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##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                              ...
2       4370    rs6057    G    A      29    .       NS=2;DP=13;AF=0.5;DB;H2
2       7330    .         T    A      3     q10     NS=5;DP=12;AF=0.017
2       110696  rs6055    A    G,T    67    PASS    NS=2;DP=10;AF=0.333,0.667;AA=T;DB
2       130237  .         T    .      47    .       NS=2;DP=16;AA=T
```

The VCF Format

```
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##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=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
```

FORMAT	Sample1	Sample2
GT:GQ:DP:HQ	0|0:48:1:52,51	1|0:48:8:51,51
GT:GQ:DP:HQ	0|0:46:3:58,50	0|1:3:5:65,3
GT:GQ:DP:HQ	1|2:21:6:23,27	2|1:2:0:18,2
GT:GQ:DP:HQ	0|0:54:7:23,60	0|0:48:4:56,51

Filtering Variants

- Ensure that you are in the “ShortVariantCourse” directory in your scratch space and set up a new directory for filtering

```
$ cd $SCRATCH/ShortVariantCourse
```

```
$ mkdir FilteringVariants && cd FilteringVariants
```

```
$ cp ../support/SVC.vcf.gz* .
```

```
$ module purge
```

```
$ module load GCCcore/11.2.0 GATK/4.2.6.1-Java-11
```

Filtering Variants

- Split the variants into SNPs and Indels:

```
$ gatk SelectVariants -V SVC.vcf.gz -select-type SNP \
    -O SVC.snps.vcf.gz
```

```
$ gatk SelectVariants -V SVC.vcf.gz -select-type INDEL \
    -O SVC.indels.vcf.gz
```

Filtering Variants

- Filter the variants:

```
$ gatk VariantFiltration -V SVC.snps.vcf.gz \
    -filter "QUAL < 30.0" --filter-name "QUAL30" \
    -O SVC.snps_filtered.vcf.gz
```

```
$ gatk VariantFiltration -V SVC.indels.vcf.gz \
    -filter "QUAL < 30.0" --filter-name "QUAL30" \
    -O SVC.indels_filtered.vcf.gz
```

Filtering Variants

- Sort the filtered VCFs:

```
$ gatk SortVcf -I SVC.snps_filtered.vcf.gz \
    -O SVC.snps_filtered.sorted.vcf.gz
```

```
$ gatk SortVcf -I SVC.indels_filtered.vcf.gz \
    -O SVC.indels_filtered.sorted.vcf.gz
```

- Merge the filtered, sorted VCF files:

```
$ gatk MergeVcfs -I SVC.snps_filtered.sorted.vcf.gz \
    -I SVC.indels_filtered.sorted.vcf.gz \
    -O SVC.filtered.sorted.vcf.gz
```

Final Notes

- The filtered variant files can be used to create a new “high-quality” SNP database
- The new database can be used to run through the pipeline again (this time re-calibrating base quality scores)