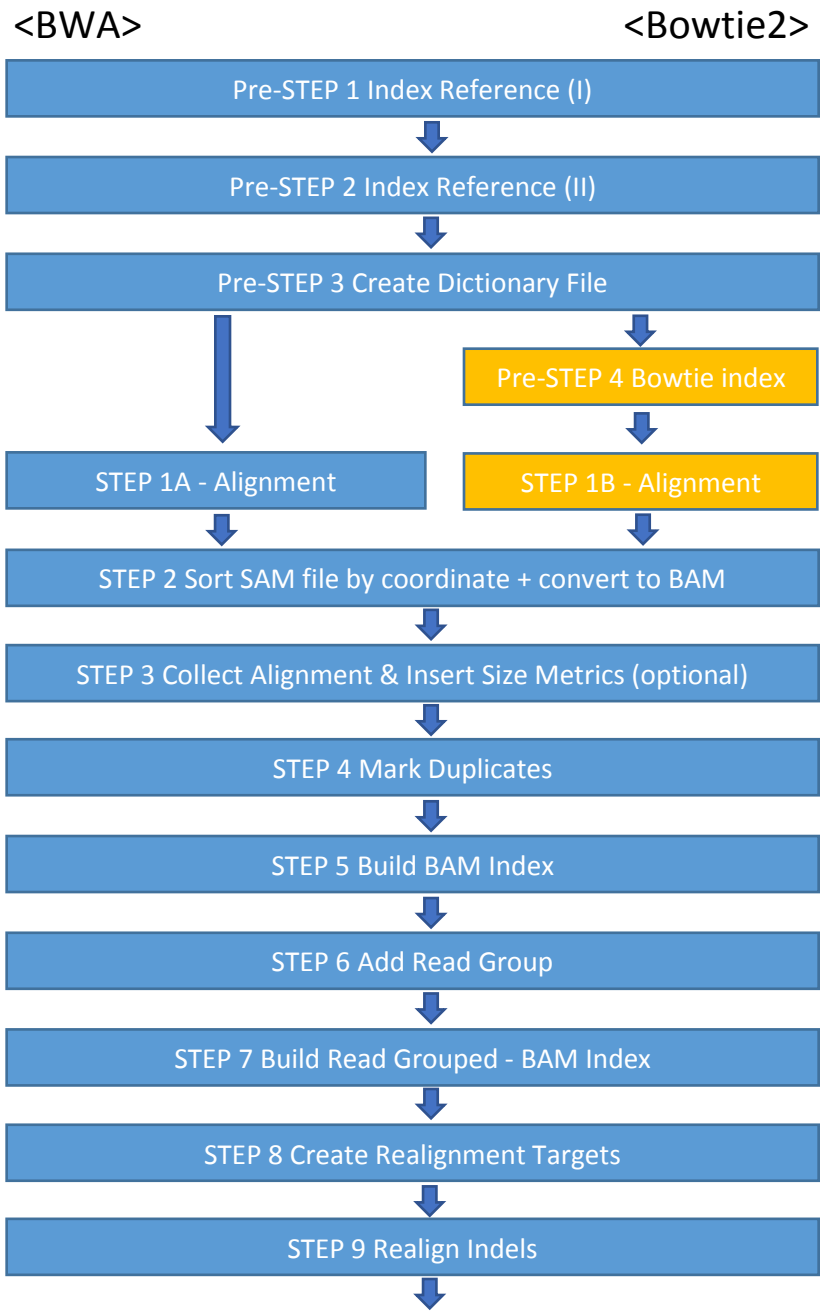
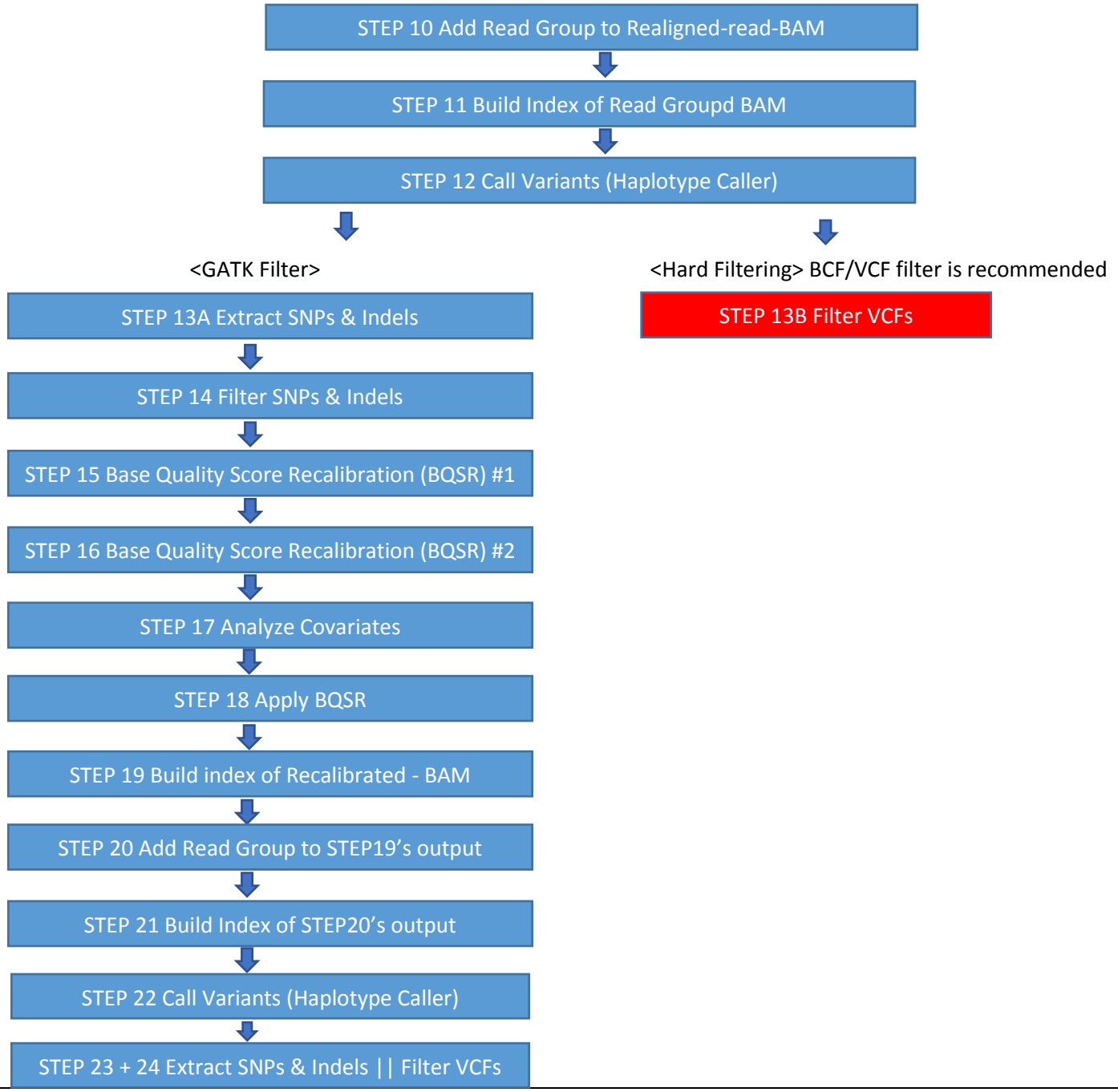

<BWA>
<Bowtie2>
Pre-STEP 1 Index Reference (I)
Pre-STEP 2 Index Reference (II)
Pre-STEP 3 Create Dictionary File
Pre-STEP 4 Bowtie index
STEP 1A - Alignment
STEP 1B - Alignment
STEP 2 Sort SAM file by coordinate + convert to BAM
STEP 3 Collect Alignment & Insert Size Metrics (optional)
STEP 4 Mark Duplicates
STEP 5 Build BAM Index
STEP 6 Add Read Group
STEP 7 Build Read Grouped - BAM Index
STEP 8 Create Realignment Targets
STEP 9 Realign Indels


STEP 10 Add Read Group to Realigned-read-BAM
STEP 11 Build Index of Read Groupd BAM
STEP 12 Call Variants (Haplotype Caller)
<GATK Filter>
<Hard Filtering> BCF/VCF filter is recommended
STEP 13A Extract SNPs & Indels
STEP 13B Filter VCFs
STEP 14 Filter SNPs & Indels
STEP 15 Base Quality Score Recalibration (BQSR) #1
STEP 16 Base Quality Score Recalibration (BQSR) #2
STEP 17 Analyze Covariates
STEP 18 Apply BQSR
STEP 19 Build index of Recalibrated - BAM
STEP 20 Add Read Group to STEP19's output
STEP 21 Build Index of STEP20's output
STEP 22 Call Variants (Haplotype Caller)
STEP 23 + 24 Extract SNPs & Indels || Filter VCFs

**This protocol is optimized for influenza A virus genome sequence data

**Bash Commands

<Job-submission commands>

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=1
#SBATCH --time=5:00:00
#SBATCH --mem=2GB
#SBATCH --job-name=myTest
#SBATCH --mail-type=END
#SBATCH --mail-user=bob.smith@nyu.edu
#SBATCH --output=slurm_%j.out

cd /path/to/the/files
module purge
module load <module name and version>

commands
```

pre-STEP 1	Index Reference (I)
Tool	BWA
Input	reference.fasta
Output	reference.fasta(or fa).ann; reference.fasta.pac; reference.fasta.amb; reference.fasta.bwt; reference.fasta.sa
Command	bwa index <reference.fasta>
example	module load bwa/intel/0.7.15 bwa index H1N1_ref.fa
note	Common in BWA track and Bowtie2 track

pre-STEP 2	Index Reference (II)
Tool	Samtools
Input	reference.fasta
Output	reference.fasta.fai
Command	samtools faidx <reference.fasta>
example	module samtools/intel/1.3.1

	samtools faidx H1N1_ref.fa
note	Common in BWA track and Bowtie2 track

pre-STEP 3	Create Dictionary File
Tool	PICARD
Input	reference.fasta
Output	reference.dict
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar CreateSequenceDictionary R=<reference> O=<reference.dict >
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar CreateSequenceDictionary R=H1N1_ref.fa O=H1N1_ref.dict
note	Common in BWA track and Bowtie2 track

pre-STEP 4	Bowtie2 Index
Tool	Bowtie2
Input	reference.fasta
Output	<base_name>.1.bt2 <base_name>.2.bt2 <base_name>.3.bt2 <base_name>.4.bt2 <base_name>.rev.1.bt2 <base_name>.rev.2.bt2
Command	bowtie2-build [options] <reference_in> <base_name>
example	module load bowtie2/intel/2.3.2 bowtie2-build H1N1_ref.fa H1N1_ref
note	Bowtie2 track only

STEP 1A	Alignment - Map to the reference	STEP 1B	Alignment - Map to the reference
Tool	BWA	Tool	Bowtie2
Input	reference.fasta	Input	reference.fasta
Output	<aligned_reads.sam>	Output	<aligned_reads.sam>
Command	bwa mem -M <reference> <forward.fastq> <reverse.fastq> \ > <output.sam>	Command	bowtie2 -x <reference> -1 <forward.fastq> -2 <reverse.fastq> \ -S <output.sam>
example	bwa mem -M *.fasta *.trimmed.r1.fastq *.trimmed.r2.fastq \ > *.aligned_reads.sam	example	module load bowtie2/intel/2.3.2 bowtie2 -x *.fasta -1 *.r1.fastq -2 *.r2.fastq \ -S *.aligned_reads.sam
note	BWA track only	note	Bowtie2 track only

STEP 2	Sort SAM file by coordinate + convert to BAM
Tool	PICARD
Input	<aligned_reads.sam>
Output	<sorted_reads.bam>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar SortSam INPUT=<aligned_reads_sam> OUTPUT=<sorted_reads.bam> \ SORT_ORDER=coordinate
example	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar SortSam INPUT=$i.aligned_reads.sam OUTPUT=$i.sorted_reads.bam \ SORT_ORDER=coordinate
note	Common in BWA track and Bowtie2 track

STEP 3	Collect Alignment & Insert Size Metrics (optional)
Tool	① PICAR ② R ③ Samtools
Input	<sorted_reads.bam>; reference.fasta
Output	① <alignment_metrics.txt> ② <insert_metrics.txt> ③ <insert_size_histogram.pdf> ④ <depth_out.txt>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar CollectInsertSizeMetrics I=<sorted_reads.bam> \ O=<alignment_metrics.txt> H =<insert_size_histogram_1.pdf> M=0.5 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar CollectInsertSizeMetrics I=<sorted_reads.bam> \ O=<insert_metrics.txt> H=<insert_size_histogram_2.pdf> M=0.5 samtools depth -a <sorted_reads.bam> > <depth_out.txt>
example	① load module module load picard/2.8.2 module load r/intel/3.4.2 module load samtools/intel/1.3.1 ② Collecting Alignment Summary Metrics java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar CollectInsertSizeMetrics I=*.sorted_reads.bam \ O=*.alignment_metrics.txt H =*.insert_size_histogram_1.pdf M=0.5 ③ Collect Insert Size Metrics java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar CollectInsertSizeMetrics INPUT=*.sorted_reads.bam \ O=*.insert_metrics.txt H=*.insert_size_histogram_2.pdf M=0.5 ④ Depth size samtools depth -a *.sorted_reads.bam > *.depth_out.txt

note	① Before executing those command, all the Pre-STEPs should be taken for the reference sequence. ② Common in BWA track and Bowtie2 track

STEP 4	Mark Duplicates
Tool	PICARD
Input	<sorted_reads.bam>
Output	① <dedup_reads.bam> ② <metrics.txt>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar MarkDuplicates \ INPUT=<sorted_reads.bam>OUTPUT=<dedup_reads.bam> \ METRICS_FILE=<metrics.txt>
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar MarkDuplicates \ INPUT=*.sorted_reads.bam OUTPUT=*.dedup_reads.bam \ METRICS_FILE=*.metrics.txt
note	Common in BWA track and Bowtie2 track

STEP 5	Build BAM Index
Tool	PICARD
Input	<dedup_reads.bam>
Output	<dedup_reads.bai>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=<dedup_reads.bam>
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=*.dedup_reads.bam;
note	Common in BWA track and Bowtie2 track

STEP 6	Add Read Group
Tool	PICARD
Input	<dedup_reads.bam>
Output	<RDGR.dedup_reads.bam>
Command	① load module module load picard/2.8.2 ② Add or Replace Read Group java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar AddOrReplaceReadGroups \

	I=<dedup_reads.bam> O=<RDGR.dedup_reads.bam> RGID=4 RGLB=lib1 RGPL=illumina RGPU=unit1 RGSM=20 \
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar AddOrReplaceReadGroups \ I=*.dedup_reads.bam O=*.RDGR.dedup_reads.bam RGID=4 RGLB=lib1 RGPL=illumina RGPU=unit1 RGSM=20 \
note	①Documentation link http://broadinstitute.github.io/picard/command-line-overview.html#AddOrReplaceReadGroups ②Common in BWA track and Bowtie2 track

STEP 7	Build Read-Grouped-BAM index
Tool	PICARD
Input	<RDGR.dedup_reads.bam>
Output	<RDGR.dedup_reads.bai>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=<RDGR.dedup_reads.bam>
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=*.RDGR.bam
note	Common in BWA track and Bowtie2 track

STEP 8	Creat Realignment Targets
Tool	GATK
Input	<RDGR.dedup_reads.bam>; reference.fasta
Output	① <realignment_targets.list>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAanlysisTK.jar -T RealignerTargetCreator -R <reference.fasta> -I <RDGR.dedup_reads.bam> \ -o <realignment_targets.list>
example	module load gatk/3.8-0 java -jar /share/apps/gatk/3.8-0/GenomeAanlysisTK.jar -T RealignerTargetCreator -R reference.fasta -I *.RDGR.dedup_reads.bam \ -o *.realignment_targets.list
note	Common in BWA track and Bowtie2 track

STEP 9	Realign Indels
Tool	GATK
Input	<realignment_target.list>; <RDGR.dedup_reads.bam>; reference.fasta
Output	<realigned_reads.bam>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T IndelRealigner -R reference.fasta -I <RDGR.dedup_reads.bam> \ -targetIntervals <realignment_target.list> -o <realigned_reads.bam>

example	module load gatk/3.8-0 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T IndelRealigner -R reference.fasta -I *.RDGR.dedup_reads.bam \ -targetIntervals *.realignment_targets.list -o *.realigned_reads.bam
note	Common in BWA track and Bowtie2 track

STEP 10	Add Read Group to <realigned_reads.bam>
Tool	PICARD
Input	<realigned_reads.bam>
Output	<RDGR_realinged_reads.bam>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar AddOrReplaceReadGroups \ I=<realigned_reads.bam> O=<RDGR_realinged_reads.bam> RGID=4 RGLB=lib1 RGPL=illumina RGPU=unit1 RGSM=20
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar AddOrReplaceReadGroups\ I=*.realigned_reads.bam O=*.RDGR_realigned_reads.bam RGID=4 RGLB=lib1 RGPL=illumina RGPU=unit1 RGSM=20
note	Common in BWA track and Bowtie2 track

STEP 11	Build Index of Read-Grouped-realigned_reads-BAM
Tool	PICARD
Input	<RDGR_realigned_reads.bam>
Output	<RDGR_realigned_reads.bai>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=<RDGR_realigned_reads.bam>
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=*.RDGR_realigned_reads.bam
note	Common in BWA track and Bowtie2 track

STEP 12	Call Variants (Haplotype Caller)
Tool	GATK
Input	<RDGR_realigned_reads.bam>; reference.fasta
Output	<GATK.raw.vcf>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T HaplotypeCaller -R reference.fasta \ -I <RDGR_realigned_reads.bam> -o <GATK.raw.vcf>
example	module load gatk/3.8-0 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T HaplotypeCaller -R reference.fasta \ -I *.RDGR_realigned_reads.bam -o *.GATK.raw.vcf
note	Common in BWA track and Bowtie2 track

STEP 13A	Extract SNPs & Indels	STEP 13B (opt 1.)	Filter VCFs
Tool	GATK	Tool	BCFtools
Input	<GATK.raw.vcf>; reference.fasta	Input	<GATK.raw.vcf>; reference.fasta
Output	<GATK.raw.indel.vcf> ;<GATK.raw.snp.vcf>	Output	<GATK-BCF.flt.vcf>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar \ -T SelectVariants -R reference.fasta \ -V <GATK.raw.vcf> \ -selectType SNP -o <$i.GATK.raw.snvs.vcf> java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar \ -T SelectVariants -R reference.fasta \ -V <GATK.raw.vcf> \ -selectType INDEL -o <GATK.raw.indel.vcf>	Command	bcftools filter -s PASS -e '%QUAL > (threshold#) || DP > (threshold#)' <GATK.raw.vcf> > <GATK-BCF.flt.vcf>
example	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar \ -T SelectVariants -R reference.fasta \ -V *.GATK.raw.vcf -selectType SNP -o *.GATK.raw.snp.vcf; java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar \ -T SelectVariants -R reference.fasta \ -V *.GATK.raw.vcf -selectType INDEL -o *.GATK.raw.indel.vcf	example	module bcftools/intel/1.3.1 bcftools filter -s PASS -i '%QUAL>20 || DP >200' \ *.GATK.raw.vcf > *.GATK-BCF.flt.vcf
note	①For applying GATK filter, do not apply STEP13B. After completing STEP 13A, go on to STEP14	note	① If you are going to use BCF filter, this is the last step. ② If you want to extract SNPs and Indels independently, you can apply STEP13A after completing STEP13B. ③ either BCFtools or vcflib(vcffilter) is recommended. From my experience, vcflib's vcffilter works more accurately.

STEP 13B (opt. 2)	Filter VCFs
Tool	vcflib
Input	<GATK.raw.vcf>; reference.fasta
Output	<GATK-BCF.flt.vcf>
Command	vcffilter -f "QUAL > (threshold#) & DP > (threshold#)" <GATK.raw.vcf> > <GATK-BCF.flt.vcf>
example	module load vcflib/intel/20170223 vcffilter -f "QUAL > 20 & DP > 200" input.vcf > output.vcf
note	vcflib performs better than BCFtools '||':or, '&':and

STEP 14	Filter SNPs and Indels
Tool	GATK
Input	<GATK.raw.indel.vcf> ;<GATK.raw.snp.vcf>;reference.fasta
Output	<GATK.flt.indel.vcf> ;<GATK.flt.snp.vcf>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T VariantFiltration -R reference.fasta -V <GATK.raw.indel.vcf> \ --filterExpression "QD < 2.0 || FS > 200.0 || ReadPosRankSum < -20.0" --filterName "PASS" -o <GATK.flt.indel.vcf>; java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T VariantFiltration -R reference.fasta -V <GATK.raw.snp.vcf> \ --filterExpression "QD < 2.0 || FS > 60.0 || MQ < 40.0 || MQRankSum < -12.5 || ReadPosRankSum < -8.0" \ --filterName "PASS" -o <GATK.flt.snp.vcf>
example	module load gatk/3.8-0 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T VariantFiltration -R reference.fasta -V *.GATK.raw.indel.vcf \ --filterExpression "QD < 2.0 || FS > 200.0 || ReadPosRankSum < -20.0" --filterName "PASS" -o *.GATK.flt.indel.vcf; java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T VariantFiltration -R reference.fasta -V *.GATK.raw.snp.vcf \ --filterExpression "QD < 2.0 || FS > 60.0 || MQ < 40.0 || MQRankSum < -12.5 || ReadPosRankSum < -8.0" \ --filterName "PASS" -o *.GATK.flt.snp.vcf
note	

STEP 15	Base Quality Score Recalibration (BQSR) #1
Tool	GATK
Input	① <RDGR_realigned_reads.bam> ② <GATK.flt.indel.vcf> ③ <GATK.flt.snp.vcf> ④ reference.fasta
Output	<GATK_recal_data.table>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T BaseRecalibrator -R reference.fasta \ -I <RDGR_realigned_reads.bam> -knownSites <GATK.flt.snp.vcf> -knownSites <GATK.flt.indel.vcf> -o <GATK_recal_data.table>
example	module load picard/2.8.2 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T BaseRecalibrator -R reference.fasta \ -I *.RDGR_realigned_reads.bam -knownSites *.GATK.flt.snp.vcf -knownSites *.GATK.flt.indel.vcf -o *.GATK_recal_data.table
note	

STEP 16	Base Quality Score Recalibration (BQSR) #2

Tool	GATK
Input	① <GATK_recal_data.table> ② <RDGR_realigned_reads.bam> ③ <GATK.flt.indel.vcf> ④ <GATK.flt.snp.vcf> ⑤ reference.fasta
Output	<GATK.post_recal_data.table>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T BaseRecalibrator -R reference.fasta -I <RDGR_realigned_reads.bam> \ -knownSites <GATK.flt.snp.vcf> -knownSites <GATK.flt.indel.vcf> -BQSR <GATK_recal_data.table> -o <GATK.post_recal_data.table>
example	module load picard/2.8.2 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T BaseRecalibrator -R reference.fasta -I *.RDGR_realigned_reads.bam \ -knownSites *.GATK.flt.snp.vcf -knownSites *GATK.flt.indels.vcf -BQSR *.GATK_recal_data.table -o *.GATK.post_recal_data.table
note	

STEP 17	Analyze Covariates
Tool	GATK
Input	① <GATK_recal_data.table> ② <GATK.post_recal_data.table> ③ reference.fasta
Output	<GATK.recalibration_plots.pdf>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T AnalyzeCovariates -R reference.fasta \ -before <GATK_recal_data.table> -after <GATK.post_recal_data.table> -plots <GATK.recalibration_plots.pdf>
example	module load picard/2.8.2 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T AnalyzeCovariates -R reference.fasta \ -before *.GATK_recal_data.table -after *.GATK.post_recal_data.table -plots *.GATK.recalibration_plots.pdf
note	

STEP 18	Apply BQSR
Tool	GATK
Input	① <GATK_recal_data.table> ② <RDGR_realigned_reads.bam> ③ reference.fasta
Output	<GATK.recal_reads.bam>

Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T PrintReads -R reference.fasta -I <RDGR_realigned_reads.bam> \ -BQSR <GATK.recal_data.table> -o <GATK.recal_reads.bam>
example	module load picard/2.8.2 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T PrintReads -R reference.fasta -I *.RDGR_realigned_reads.bam \ -BQSR *.GATK.recal_data.table -o *.GATK.recal_reads.bam
note	

STEP 19	Build index of Recalibrated-BAM
Tool	PICARD
Input	<GATK.recal_reads.bam>
Output	<GATK.recal_reads.bai>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=<GATK.recal_reads.bam>
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=*.GATK.recal_reads.bam
note	

STEP 20	Add Read Group to Recalibrated-BAM
Tool	PICARD
Input	<GATK.recal_reads.bam>
Output	<RDGR.GATK.recal_reads.bam>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar AddOrReplaceReadGroups I=<GATK.recal_reads.bam> \ O=<RDGR.GATK.recal_reads.bam> RGID=4 RGLB=lib1 RGPL=illumina RGPU=unit1 RGSM=20
example	module load picard/2.8.2 java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar AddOrReplaceReadGroups I=*.GATK.recal_reads.bam \ O=*.RDGR.GATK.recal_reads.bam RGID=4 RGLB=lib1 RGPL=illumina RGPU=unit1 RGSM=20
note	

STEP 21	Build Index of Read-Group-Recalibrated-BAM
Tool	PICARD
Input	<RDGR.GATK.recal_reads.bam>
Output	<RDGR.GATK.recal_reads.bai>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=<RDGR.GATK.recal_reads.bam>
example	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar BuildBamIndex INPUT=*.RDGR.GATK.recal_reads.bam
note	

STEP 22	Call Variants (Haplotype Caller)
Tool	GATK
Input	<RDGR.GATK.recal_reads.bam>; reference.fasta
Output	<GATK.recal.raw.vcf>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T HaplotypeCaller -R reference.fasta -I <RDGR.GATK.recal_reads.bam> \ -o <GATK.recal.raw.vcf>
example	module load gatk/3.8-0 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T HaplotypeCaller -R reference.fasta -I *.RDGR.GATK.recal_reads.bam \ -o *.GATK.recal.raw.vcf
note	

STEP 23	Extract SNPs & Indels
Tool	GATK
Input	<GATK.recal.raw.vcf>;reference.fasta
Output	①<GATK.recal.raw.snp.vcf> ②<GATK.recal.raw.indels.vcf>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T SelectVariants -R reference.fasta -V <GATK.recal.raw.vcf> \ -selectType SNP -o <GATK.recal.raw.snvs.vcf>; java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T SelectVariants -R reference.fasta -V <GATK.recal.raw.vcf> \ -selectType INDEL -o <GATK.recal.raw.indels.vcf>
example	module load gatk/3.8-0 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T SelectVariants -R reference.fasta -V *.GATK.recal.raw.vcf \ -selectType SNP -o *.GATK.recal.raw.snp.vcf; java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T SelectVariants -R reference.fasta -V $i.GATK.recal.raw.vcf \ -selectType INDEL -o *.GATK.recal.raw.indel.vcf
note	

STEP 24	Filter VCFs
Tool	GATK
Input	①<GATK.recal.raw.snp.vcf> ②<GATK.recal.raw.indels.vcf> ③reference.fasta
Output	①<GATK.recal.flt.indels.vcf> ②<GATK.recal.flt.indels.vcf>
Command	java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T VariantFiltration -R reference.fasta -V <GATK.recal.raw.indel.vcf> \

	--filterExpression "QD < 2.0 || FS > 200.0 || ReadPosRankSum < -20.0" --filterName "PASS" -o <GATK.recal.flt.indel.vcf>; java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T VariantFiltration -R reference.fasta -V <GATK.recal.raw.snp.vcf> \ --filterExpression "QD < 2.0 || FS > 60.0 || MQ < 40.0 || MQRankSum < -12.5 || ReadPosRankSum < -8.0" --filterName "PASS" \ -o <GATK.recal.flt.snp.vcf>
example	module load gatk/3.8-0 java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T VariantFiltration -R reference.fasta -V *.GATK.recal.raw.indel.vcf \ --filterExpression "QD < 2.0 || FS > 200.0 || ReadPosRankSum < -20.0" --filterName "PASS" -o *.GATK.recal.flt.indel.vcf; java -jar /share/apps/gatk/3.8-0/GenomeAnalysisTK.jar -T VariantFiltration -R reference.fasta -V *.GATK.recal.raw.snp.vcf \ --filterExpression "QD < 2.0 || FS > 60.0 || MQ < 40.0 || MQRankSum < -12.5 || ReadPosRankSum < -8.0" --filterName "PASS" \ -o *.GATK.recal.flt.snp.vcf
note	

Reference

1. Khalfan M., Variant Calling Pipeline: FastQ to Annotated SNPs in Hours

(The Genomics Core Facility @ NYU CGSBSkip to content -https://gencore.bio.nyu.edu/variant-calling-pipeline/)

2. Genome Analysis ToolKit documentations (Broad Institute: https://software.broadinstitute.org/gatk/documentation/)