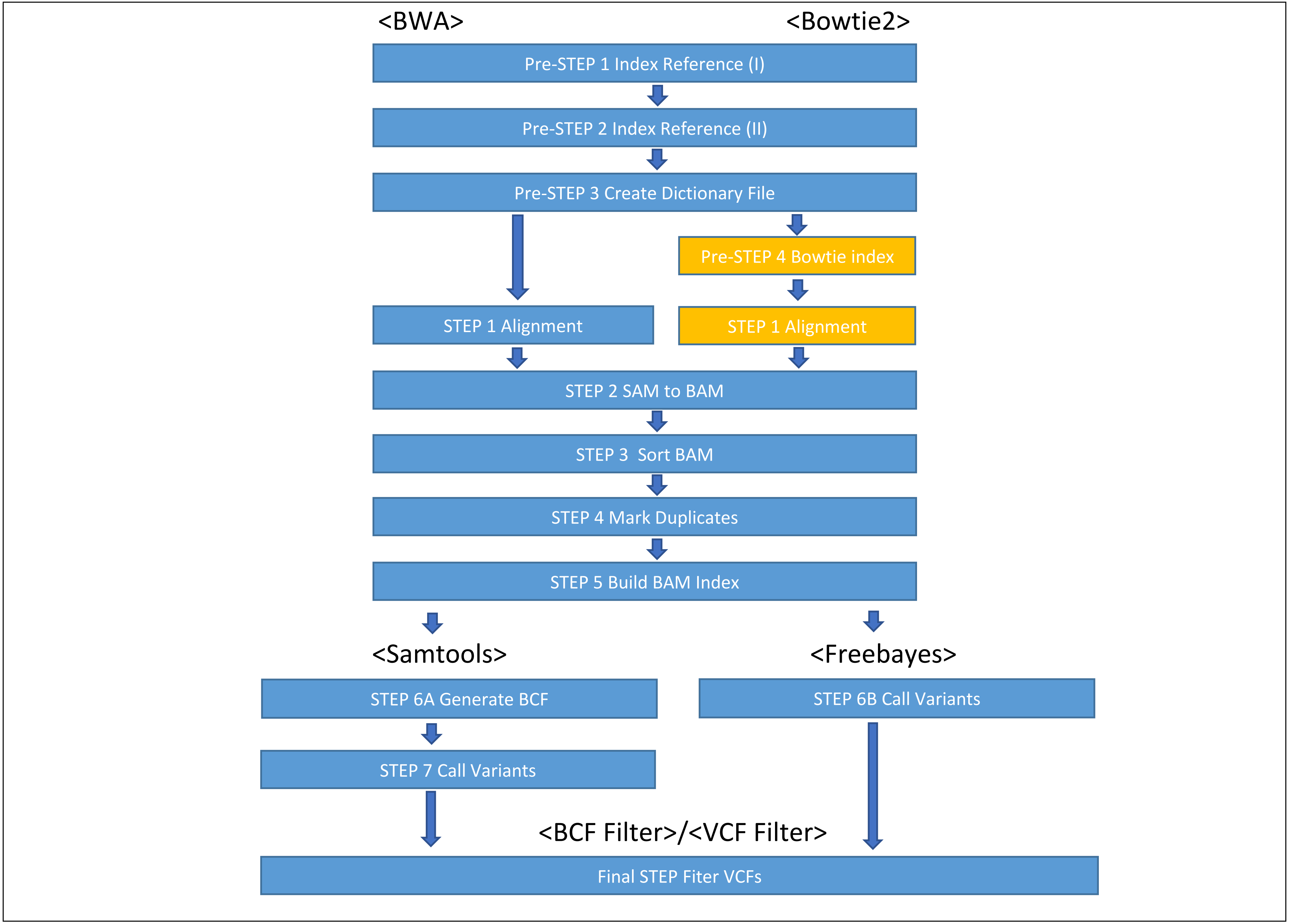
Samtools & Frebayes Varicant Calling pipelines

Minju Kim



<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 1 Alignment

STEP 1 Alignment

STEP 2 SAM to BAM

STEP 3 Sort BAM

STEP 4 Mark Duplicates

STEP 5 Build BAM Index

<Samtools>

<Freebayes>

STEP 6A Generate BCF

STEP 6B Call Variants

STEP 7 Call Variants

<BCF Filter>/<VCF Filter>

Final STEP Fiter VCFs

Samtools & Frebayes Varicant Calling pipelines Minju Kim

**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

Samtools & Frebayes Varicant Calling pipelines Minju Kim

STEP 2	SAM to BAM
Tool	Samtools
Input	<aligned_reads.sam>; reference.fasta
Output	<bam>
Command	samtools view -S -b [-h] <aligned_reads.sam> > <output_bam>
example	module load samtools/intel/1.3.1 samtools view -S -b -h *.aligned_reads.sam > *.bam
note	Common in BWA track and Bowtie2 track

STEP 3	Sort BAM
Tool	Samtools
Input	<input_bam>
Output	<sorted_bam>
Command	samtools sort -T /tmp/sorted -o <sorted.bam> <input_bam>
example	module samtools/intel/1.3.1 samtools sort -T /tmp/sorted -o *.sorted.bam *.bam
note	Common in BWA track and Bowtie2 track

STEP 4	Mark Duplicate
Tool	PICARD
Input	<sorted.bam>
Output	<dedup_reads.bam>; <metrics.txt>
Command	java -jar /share/apps/picard/2.8.2/picard-2.8.2.jar MarkDuplicates INPUT=<sorted.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.bam OUTPUT=*.dedup_reads.bam \ METRICS_FILE=*.metrics.txt
note	Common in BWA track and Bowtie2 track

STEP 5	Build BAM Index
Tool	Samtools
Input	<dedup_reads.bam>
Output	<dedup_read.bai>
Command	samtools index <dedup_reads.bam>
example	module samtools/intel/1.3.1 samtools index *.dedup_reads.bam

note	Common in BWA track and Bowtie2 track

STEP 6A	Generate BCF file	STEP 6B	Call Variants (Freebayes)
Tool	Samtools	Tool	Freebayes
Input	<dedup_reads.bam>; reference.fasta	Input	<dedup_reads.bam>; reference.fasta
Output	<SAM-BCF.bcf>	Output	<FB.raw.vcf>
Command	samtools mpileup -g -f reference.fasta \ <dedup_reads.bam> > <SAM-BCF.bcf>	Command	/share/apps/freebayes/1.1.0/intel/bin/freebayes \ -f reference.fasta <dedup_reads.bam> > <FB.raw.vcf>
example	module load samtools/intel/1.3.1 samtools mpileup -g -f reference.fasta \ *.dedup_reads.bam > *.SAM-BCF.bcf	example	module freebayes/intel/1.1.0 /share/apps/freebayes/1.1.0/intel/bin/freebayes \ -f reference.fasta *.dedup_reads.bam > *.FB.raw.vcf
note	①Samtools track only ②Go on to STEP 7	note	①Freebayes Track only ②Go on to the Final STEP

STEP 7	Call Variants (Samtools)
Tool	BCFtools
Input	<SAM-BCF.bcf>
Output	<SAM-BCF.raw.vcf>
Command	bcftools call -c -v <SAM-BCF.bcf> > <SAM-BCF.raw.vcf>
example	module load bcftools/intel/1.3.1 bcftools call -c -v *.SAM-BCF.bcf > *.SAM-BCF.raw.vcf
note	Samtools track only

Final STEP (opt 1.)	Filter VCFs
Tool	BCFtools
Input	<SAM-BCF.raw.vcf> / <FB.raw.vcf>
Output	<SAM-BCF.flt.vcf> / <FB.flt.vcf>
Command	bcftools filter -s PASS -i '%QUAL > (threshold#) || DP > (threshold#)' <input_raw.vcf> > <output_flt.vcf>
example	module bcftools/intel/1.3.1 bcftools filter -s PASS -i '%QUAL>20 || DP >200' \ *.FB.raw.vcf > *.FB.flt.vcf
note	Common in Samtools track and Freebayes track the flag ‘-i’ means ‘include’ ; the flag ‘-e’ means ‘exclude’

Final STEP (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

Reference

1. SAMtools: http://samtools.sourceforge.net/mpileup.shtml

2. VCFlib: https://github.com/vcflib/vcflib

3. FreeBayes: https://github.com/ekg/freebayes