NGS Variant Calling Pipeline for SRA Accession SRR32313970

This document outlines the complete variant calling pipeline used for processing SRA accession **SRR32313970**. Each section describes the tool, the purpose of the step, and the exact commands (with comments for clarity and reproducibility).

---

1. Data Acquisition and Conversion

Tools:
- **SRA Toolkit** (`prefetch`, `fastq-dump`)

Purpose:
Download the SRA file and convert it to paired-end FASTQ format.

```
# Download the SRA file for SRR32313970
prefetch SRR32313970 --progress
```

Convert the SRA file to paired-end FASTQ files

```
fastq-dump --split-files SRR32313970/
```

Run FastQC on raw FASTQ files to generate quality reports

```
fastqc SRR32313970_1.fastq SRR32313970_2.fastq
```

Create a directory for trimmed reads and switch into it

```
mkdir -p Trimmed
cd Trimmed
```

Run Trimmomatic in paired-end mode with sliding window trimming and minimum length filtering

```
trimmomatic PE ../SRR32313970_1.fastq ../SRR32313970_2.fastq \
    SRR32313970_1_paired.fastq SRR32313970_1_unpaired.fastq \
    SRR32313970_2_paired.fastq SRR32313970_2_unpaired.fastq \
    SLIDINGWINDOW:4:20 MINLEN:50
```

Run FastQC on the trimmed reads for quality control

```
fastqc SRR32313970_1_paired.fastq SRR32313970_2_paired.fastq
SRR32313970_1_unpaired.fastq SRR32313970_2_unpaired.fastq
```

Return to the parent directory

```
cd ../

# Switch to the Trimmed directory
cd Trimmed

# Build the Bowtie2 index for the reference genome
bowtie2-build GRCh38.primary_assembly.genome.fa index

# Align paired-end reads using the built index
bowtie2 --no-unal -p 2 -x
/home/madhuram9011/Downloads/variant_call_pipe/trimmed_reads/index \
  -1 SRR32313970_1_paired.fastq -2 SRR32313970_2_paired.fastq -S
alignment.sam
```

Move the SAM file to the parent directory for downstream processing

```
mv alignment.sam ../
cd ../

# Convert the SAM file to BAM format
samtools view -Sb -o alignment.bam alignment.sam

# Sort the BAM file by coordinate
samtools sort -O bam -o sorted.bam alignment.bam

# Create an index for the sorted BAM file
samtools index sorted.bam
```

6. Adding/Updating Read Groups

Tool:
- GATK (Genome Analysis Toolkit)

Purpose:
Add or update read groups in the sorted BAM file for proper downstream processing.

```
gatk AddOrReplaceReadGroups \
  -I sorted.bam \
  -O sorted_rg.bam \
  --RGID 1 \
  --RGLB lib1 \
  --RGPL ILLUMINA \
  --RGPU unit1 \
  --RGSM SampleName
```

7. Marking Duplicates

Tool:
- **GATK MarkDuplicates**

Purpose:
Identify and mark duplicate reads in the BAM file.

```
gatk MarkDuplicates \
  -I sorted_rg.bam \
  -R GRCh38.primary_assembly.genome.fa \
  -M metrics.txt \
  -O unique_reads.bam
```

8. Base Quality Score Recalibration (BQSR)

Tool:
- **GATK BaseRecalibrator** and **ApplyBQSR**

Purpose:
Generate and apply a recalibration table to adjust base quality scores.

Generate a recalibration table using known variant sites

```
gatk BaseRecalibrator \
  -R GRCh38.primary_assembly.genome.fa \
  -I unique_reads.bam \
  --known-sites Mills_and_1000G_gold_standard.indels.hg38.renamed.vcf.gz \
  -O recal_data.table

# Apply the recalibration to adjust base quality scores

gatk ApplyBQSR \
  -R GRCh38.primary_assembly.genome.fa  \
  -I unique_reads.bam \
  --bqsr-recal-file recal_data.table \
  -O recalibrated.bam
```

9. Variant Calling

Tool:
- **GATK HaplotypeCaller**

Purpose:
Call variants (SNPs and Indels) from the recalibrated BAM file.

```
gatk HaplotypeCaller \
  -R GRCh38.primary_assembly.genome.fa  \
  -I recalibrated.bam \
  -O output.vcf
```

10. Variant Filtering

Tool:
- **GATK VariantFiltration**

Purpose:
Apply custom filters to flag or remove low-confidence variants.

```
gatk VariantFiltration \
  -R GRCh38.primary_assembly.genome.fa  \
  -V output.vcf \
  -O filtered_output.vcf \
  --filter-expression "QD < 2.0" \
  --filter-name "LowQD" \
  --filter-expression "FS > 60.0" \
  --filter-name "HighFS"
```

11 Variant Annotation with GATK VariantAnnotator

Tool:
- **GATK VariantAnnotator**

Purpose:
Enhance the raw variant calls by adding informative annotations (such as coverage metrics, quality by depth, and mapping quality rank sum) to your VCF file. These annotations support further filtering and prioritization of variants.

```
gatk VariantAnnotator \
  -R GRCh38.primary_assembly.genome.fa  \
  -V output.vcf \
  -I recalibrated.bam \
  -O output.annotated.vcf \
  -A Coverage \
  -A QualByDepth \
  -A MappingQualityRankSumTest
```

12. Visualization

Tool:
- **IGV (Integrative Genomics Viewer)**

Purpose:
Load the `sorted.bam` file in IGV to visually inspect alignments and verify variant calls (SNPs and Indels).

This file provides a step-by-step, reproducible pipeline for processing NGS data from raw SRA files to final variant calls. Each section is clearly demarcated with bold and large headings, ensuring clarity and ease of navigation. Enjoy your reproducible workflow! ```