Pre-Processing SRA/raw reads

Processing reference genome

1. Download the reference genome in case of humans either - GRCH7/hg19 or GRCH38/hg38, in case of mouse - GRCm38/mm10, and drosophila - dm6.
   For human:
   Download reference command:
   wget [link to download]
   ls #to check the file is there.

Hg19 link: click to download

2. Unzip the reference genome.
   gunzip hg38.fa.gz

3. Index the reference genome. (two approaches are possible)
   Using bowtie2
   probability of the occurrence of the read sequence given an alignment location
   bowtie2-build hg38.fa hg38_index

   Output files in working directory: hg38_index.1.bt2, hg38_index.2.bt2, hg38_index.3.bt2, hg38_index.4.bt2, hg38_index.rev.1.bt2, andhg38_index.rev.2.bt2:

Processing sequencing read files

1. **Download the .sra(sequence read archive) from ncbi or GEO if the accession number provided.** (run selector: link, ENA: link)
   Command:
   prefetch [accession number1], [accession number2], etc
   For example:
   prefetch ERR2675347 ERR2675342 ..

   P.S. sra accession numbers either start with ERR or SRR followed by numeric digits, with prefetch command we can order one or multiple sra files in one go.
   If in case we have two sra from one sample download both.

2. **Convert .sra to fastq format. (for converting all .sra files to .fastq paired end reads.)**
   For single end reads:
   fastq-dump filename or fastq-dump filename.sra

Using the "parallel" tool for automating downloads

```
parallel fastq-dump --split-files {} ::: ERR2675347 ERR2675342
```

- parallel runs the fastq-dump --split-files command on multiple .sra files concurrently.
- {} represents the placeholder for each .sra file.
- ::: ERR2675347 ERR2675342 lists the .sra files to be processed

Quality control

```
fastqc merged_1.fastq merged_2.fastq
```

3. Alignment

Align the FASTQ file to the reference genome since the reads are paired-end reads

Using bowtie2

Single end reads

```
bowtie2 --very-sensitive -x  hg38_index -U sample.fastq -S sample.sam
```

Example:

```
bowtie2 --very-sensitive -x  hg38_index -U ERR2675342.fastq -S sample.sam
```

By using multithreading:

```
bowtie2 --very-sensitive -p 10 -x hg38_index -1 ERR6131780_1.fastq -2 ERR6131780_2.fastq

-S ERR6131765.sam
```

-p 10 : specifying the number of threads

4. Convert SAM to BAM and remove SAM file

```
samtools view -bS sample.sam > sample.bam
```

Example:

```
samtools view -bS ERR2675342.sam > ERR2675342.bam
```

By using multithreading:

```
samtools view -@ 10 -bS ERR2675342.sam > ERR2675342.bam
```

-@ 10 : specifying the number of threads

5. Sort BAM by coordinate

```
samtools sort input.bam -o sorted_output.bam
```

Example:

```
Samtools sort ERR2675342.bam -o ERR2675342_sorted.bam
```

6. Index sorted BAM

```
samtools index sorted_output.bam
```

This will create an index file with the .bai extension (e.g., sorted_output.bam.bai).

7. **BAM to bed**

```
bedtools bamtobed -i sorted_output.bam > output.bed
```

Output bed file should include:

- Chromosome (reference name)
- Start (0-based start position of the alignment)
- End (1-based end position of the alignment)
- Name (optional, usually the read name)
- Score (optional)
- Strand (optional, + or - for strand direction)