HW1 For BICB 8510

Yangyang Li
Department of Computer Science
University of Minnesota
`li002252@edu.com`

1 Question 1

the first line of the E.coli reference genome shows information such as the chromosome ID and the organism.

2 Question 2

For each sample, there will be two files as it is sequenced by using paired-end techniques. As we have two samples, we get four files finally.

One is the forward sequencing read and another is the backward sequencing read.

3 Question 3

we give different values for -R for the data from SRR1770413 and SRR341549 as they are processed by different steps and methods such as Library or flowcell, etc.

Hence, we need to identify them in terms of the R-value (read group). What is more, it indicates that they are from different strains.

If the two samples come from identical strain, which indicates they are sequenced through the same process, they will have same information or R-value.

4 Question 4

the **DP** of **INFO** filed states the total sequencing depth across all the samples at a specific variant. In addition, the **DP** of **FORMAT** filed shows the sequencing depth for each sample at a specific variant.

5 Question 5

`0` means the sample has the reference allele, and `1` means the sample has the variant allele. Hence, we can choose the variant:

```
NC_000913.3 759 C T GT:DP:AD:RO:QR:AO:QA:GL 1:110:1,109:1:10:109:3579:-321.254,0
0:60:58,1:58:1824:1:2:0,-164.18
```

Note:I pick up the column including important information to show.

6 Question 6

I use the tool snakemake to build the workflow, and it works well for the whole process. In addition, the code is attached in the submit, and the concrete code is shown below as well:

Listing 1: Workflow for variant calling

```
SAMPLES = ["SRR341549", "SRR1770413"]
RG = {"SRR341549": "O104_H4", "SRR1770413":"K12"}

rule all:
    input:
        "e_colis.vcf"

rule bwa_map:
    input:
        read1 = "fq/{sample}_1.fastq",
        read2 = "fq/{sample}_2.fastq",
        ref = "E.coli_K12_MG1655.fa",
    params:
        rg = lambda wildcards: RG[wildcards.sample]
    output:
        "mapped_reads/{sample}.raw.sam"
    shell:
        r"bwa mem -R '@RG\tID:{params.rg}\tSM:{params.rg}' {input.ref}"
        r"{input.read1} {input.read2} -o {output}"

rule sam2bam:
    input:
        "mapped_reads/{sample}.raw.sam"
    output:
        temp("mapped_reads/{sample}.raw.bam")
    shell:
        "samtools view -b {input} > {output}"

rule bam_sort:
    input:
        "mapped_reads/{sample}.raw.bam"
    output:
        "sorted_reads/{sample}.raw.sorted.bam"
    shell:
        "sambamba sort {input} -o {output}"

rule makeup:
    input:
        "sorted_reads/{sample}.raw.sorted.bam"
    output:
        "sorted_reads/{sample}.bam"
    shell:
        "sambamba markdup {input} {output}"

rule index:
    input:
        "sorted_reads/{sample}.bam"
    output:
        "sorted_reads/{sample}.bam.bai"
    shell:
        "samtools index {input}"
```

```
rule call:
    input:
        ref = "E.coli_K12_MG1655.fa",
        bam = expand("sorted_reads/{sample}.bam", sample=SAMPLES),
        bai = expand("sorted_reads/{sample}.bam.bai", sample=SAMPLES)
    output:
        "e_colis.vcf"
    shell:
      "freebayes -f {input.ref} --ploidy 1  {input.bam} > {output}"
```

List of source codes