Final Assignment

Next Generation Sequencing

REPORT

João Cunha nº202103227

Mestrado Bioinformática e Biologia Computacional

Note: Since the command lines and scripts used are included in the ".txt" file, they were not added to this report. I will only include the relevant information and comment on the output of each step.

Execute a SNP calling using the following SRA numbers (SRR2125267, SRR2125268, SRR2125272, SRR2125297)

First, I had to download some files, with the reference genome, the bed file, and the SRA sequences, 3 files per each different SRA (with the help of "fastq-dump" command).

After a failed attempt to create a script to automatically download the necessary sequences I decided to do this download manually, generating the following 12 new files:

```
[up202103227@mbge assigF]$ ls
file_4.bed              SRR2125267.fastq.gz    SRR2125272_1.fastq.gz  SRR2125297_2.fastq.gz
hg38.fa.gz              SRR2125268_1.fastq.gz  SRR2125272_2.fastq.gz  SRR2125297.fastq.gz
SRR2125267_1.fastq.gz   SRR2125268_2.fastq.gz  SRR2125272.fastq.gz
SRR2125267_2.fastq.gz   SRR2125268.fastq.gz    SRR2125297_1.fastq.gz
```

After that, the reference genome was unzipped and indexed (with 710 iterations).

Ending output of the code used:

```
[BWTIncConstructFromPacked] 680 iterations done. 6347488418 characters processed.
[BWTIncConstructFromPacked] 690 iterations done. 6374612482 characters processed.
[BWTIncConstructFromPacked] 700 iterations done. 6398716386 characters processed.
[BWTIncConstructFromPacked] 710 iterations done. 6418572210 characters processed.
[bwt_gen] Finished constructing BWT in 710 iterations.
[bwa_index] 3402.61 seconds elapse.
[bwa_index] Update BWT... 25.48 sec
[bwa_index] Pack forward-only FASTA... 23.45 sec
[bwa_index] Construct SA from BWT and Occ... 1482.61 sec
[main] Version: 0.7.17-r1188
[main] CMD: bwa index hg38.fa
[main] Real time: 4979.690 sec; CPU: 4967.596 sec
```

Create and Preparate the SAM files

Then, create SAM files from the files provided, one to one again.

After that, I got a SAM file for each of the SRA numbers, meaning 4 new files were created the ".sam" files.

Endings outputs of each code used:

```
[M::mem_process_seqs] Processed 305994 reads in 22.491 CPU sec, 7.312 real sec
[main] Version: 0.7.17-r1188
[main] CMD: bwa mem -t3 hg38.fa SRR2125267_1.fastq.gz SRR2125267_2.fastq.gz
[main] Real time: 569.990 sec; CPU: 1771.537 sec
[M::mem_process_seqs] Processed 220544 reads in 17.131 CPU sec, 5.459 real sec
[main] Version: 0.7.17-r1188
[main] CMD: bwa mem -t3 hg38.fa SRR2125268_1.fastq.gz SRR2125268_2.fastq.gz
[main] Real time: 586.271 sec; CPU: 1820.413 sec
[M::mem_process_seqs] Processed 7444 reads in 0.907 CPU sec, 0.254 real sec
[main] Version: 0.7.17-r1188
[main] CMD: bwa mem -t3 hg38.fa SRR2125272_1.fastq.gz SRR2125272_2.fastq.gz
[main] Real time: 564.873 sec; CPU: 1760.562 sec
[M::mem_process_seqs] Processed 158236 reads in 13.770 CPU sec, 4.390 real sec
[main] Version: 0.7.17-r1188
[main] CMD: bwa mem -t3 hg38.fa SRR2125297_1.fastq.gz SRR2125297_2.fastq.gz
[main] Real time: 562.164 sec; CPU: 1748.894 sec
```

After having the SAM files created, I developed a script for each SRR sequence (file_67.sh, file_68.sh, file_72.sh, file_97.sh). These scripts transform them into BAM files, also remove duplicated ones.

For each SAM file created:

- I used the bed file to call only the variants included in the file;
- Transformed them into BAM;
- Fill in mate coordinates;
- Obtained the fixmate information;
- Marked the duplicates and removed them;
- Used a thread value of 3, for all files.

Note: After creating each script I had to make them executable before executing each one.

Outputs of each code used:

```
[up202103227@mbge assigF]$ ./file_67.sh
[bam_sort_core] merging from 3 files and 3 in-memory blocks...
[bam_sort_core] merging from 3 files and 3 in-memory blocks...
READ 10741107 WRITTEN 10740044
EXCLUDED 2717 EXAMINED 10738390
PAIRED 10737528 SINGLE 862
DULPICATE PAIR 888 DUPLICATE SINGLE 175
DUPLICATE TOTAL 1063
[up202103227@mbge assigF]$ ./file_68.sh
[bam_sort_core] merging from 3 files and 3 in-memory blocks...
[bam_sort_core] merging from 3 files and 3 in-memory blocks...
READ 10283063 WRITTEN 10280603
EXCLUDED 3740 EXAMINED 10279323
PAIRED 10277896 SINGLE 1427
DULPICATE PAIR 2136 DUPLICATE SINGLE 324
DUPLICATE TOTAL 2460
[up202103227@mbge assigF]$ ./file_72.sh
[bam_sort_core] merging from 3 files and 3 in-memory blocks...
[bam_sort_core] merging from 3 files and 3 in-memory blocks...
READ 10668834 WRITTEN 10667976
EXCLUDED 2153 EXAMINED 10666681
PAIRED 10665964 SINGLE 717
DULPICATE PAIR 698 DUPLICATE SINGLE 160
DUPLICATE TOTAL 858
```

```
[up202103227@mbge assigF]$ ./file_97.sh
[bam_sort_core] merging from 3 files and 3 in-memory blocks...
[bam_sort_core] merging from 3 files and 3 in-memory blocks...
READ 10573430 WRITTEN 10571788
EXCLUDED 2354 EXAMINED 10571076
PAIRED 10570388 SINGLE 688
DULPICATE PAIR 1484 DUPLICATE SINGLE 158
DUPLICATE TOTAL 1642
```

After generating several files and, I will mention the most important ones, 4 files "*_final.bam":

- SRR2125267_final.bam;
- SRR2125268_final.bam;
- SRR2125272_final.bam;
- SRR2125297_final.bam.

From these bam files we can see some information, like:

- Count all the reads before removing duplicates;
- Count all the reads after removing duplicates;
- Count all the reads mapped before removing duplicates;
- Count all the reads mapped after removing duplicates.

SNP Calling

After having the bam files, I had all the alignments and base calls ready to start the SNP calling. This step generates a "final.vcf" file.

Output of each codes used:

```
[up202103227@mbge assigF]$ samtools mpileup -uf hg38.fa *_final.bam | bcftools call -mv  > befor
efilt.vcf
Note: none of --samples-file, --ploidy or --ploidy-file given, assuming all sites are diploid
[mpileup] 4 samples in 4 input files
<mpileup> Set max per-file depth to 2000
[up202103227@mbge assigF]$ bcftools filter -s LowQual -e '%QUAL<20 || DP>100' beforefilt.vcf > f
inal.vcf
```

MAF filter of 0.05

After the SNP calling, I applied the MAF filter of 0.05. This step only included sites with a Minor Allele Frequency greater than or equal to 0.05 (avoiding alleles that are in low frequency).

Also generated a final file: "analysis_maf.record.vcf."

Output of each code used:

```
[up202103227@mbge assigF]$ vcftools --vcf final.vcf --maf 0.05 --out analysis_maf --recode

VCFtools - 0.1.15
(C) Adam Auton and Anthony Marcketta 2009

Parameters as interpreted:
        --vcf final.vcf
        --maf 0.05
        --out analysis_maf
        --recode

After filtering, kept 4 out of 4 Individuals
Outputting VCF file...
After filtering, kept 75372 out of a possible 79898 Sites
Run Time = 2.00 seconds
```

Remove SAM Files

Finally I remove all the files that we are not interested in (SAM files).