Final Assignment – Margarida Pinheiro (up201805012)

The code used to obtain the different output files was sent under the name "Command_lines_up201805012.txt".

SAM/BAM files

Using the commands samtools view -c ERR*.sam and samtools view -c ERR*.bam it was possible to compare the number of reads before and after sorting and removing duplicates.

The results are shown in the following table:

SRA Number	Before	After
ERR748562	24232007	12815835
ERR753053	22025056	11656658
ERR753072	19391619	10086465
ERR748552	17864429	9265585
ERR753091	19118898	9593237
ERR748543	16563364	8660730

VCF files

In order to count the number of variants in the .vcf files (both the filtered and the non-filtered), the following command was used:

```
[up201805012@mbge Assignment4]$ cat *.vcf | grep -v '^#' | wc -l
7029809
```

```
[up201805012@mbge Assignment4]$ cat final.vcf | grep -v '^#' | wc -l
2485860
```

We have 164851 variants in total and 82427 in the filtered VCF file.

To count the number of indels (insertions or deletion of bases) in the genome the following command was used:

```
[up201805012@mbge Assignment4]$ grep -c INDEL final.vcf
206837
```

In order to return the number of variants in the chromosome selected the following command was used for the chromosomes that were in my assigned BED file (chr2, ch3, chr4, chr5, chr6, chr7, chr8, chr15, chr16 chr20 and chr21):

```
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr2
574279
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr3
315794
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr4
277661
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr5
274002
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr6
262524
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr7
255489
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr8
225946
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr15
144738
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr16
155586
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr20
115454
[up201805012@mbge Assignment4]$ gawk '{print "chr"$0}' final.vcf | grep -c chr21
61091
```

The values of variants in each chromosome ranged between 574279 and 61091, where chromosome 2 was the one with the highest value of variants and the chromosome 21 the one with the lowest.

Regarding the VCF files, one important thing that we can also observe is the number of variants that have a quality score larger than 100. Using the following command, we know thar in this case, 469095 variants had a quality score larger than 100.

```
[up201805012@mbge Assignment4]$ gawk '{if($6>100) print $0}' final.vcf | wc -l
469095
```

bcftools

To filter a VCF file and output a new VCF file with only variants that pass the filter, I used the following command:

```
bcftools view -f PASS final.vcf > final.PASS.vcf
```

The `view` command in bcftools is used to select and filter variant records from VCF files. The `-f PASS` option specifies that only records with a PASS value in the FILTER field should be output. The input file is final.vcf and the output file is final.PASS.vcf.

After that, I ran the following command to discover that 848527 variants passed the filter.

```
[up201805012@mbge Assignment4]$ grep -v "#" -c final.PASS.vcf
848527
```

The following commands allowed to discover the number of multiallelic sites (`-m3`), the number of biallelic sites (`-m2`), the heterozygous variants (`-g het`) and the homozygous variants (`-get hom`).

```
[up201805012@mbge Assignment4]$ bcftools view -m3 final.vcf | grep -v "#" -c
6315
[up201805012@mbge Assignment4]$ bcftools view -m2 final.vcf | grep -v "#" -c
2485860
[up201805012@mbge Assignment4]$ bcftools view -g het final.vcf | grep -v "#" -c
562239
[up201805012@mbge Assignment4]$ bcftools view -g hom final.vcf | grep -v "#" -c
2395635
```

Finally, the following command was used to output the file with different statistics about the corresponding VCF file:

```
bcftools stats final.vcf > final.stats
```

In this file we can find information about the SNPs, indels, the number of multiallelic sites, the number of biallelic sites, the heterozygous variants, the homozygous variants and others.

Example of the information provided by the final.stats file:

```
# Definition of sets:
# ID  [2]id [3]tab-separated file names
ID    0     final.vcf
# SN, Summary numbers:
#   number of records   .. number of data rows in the VCF
#   number of no-ALTs   .. reference-only sites, ALT is either "." or identical to REF
#   number of SNPs      .. number of rows with a SNP
#   number of MNPs      .. number of rows with a MNP, such as CC>TT
#   number of indels    .. number of rows with an indel
#   number of others    .. number of rows with other type, for example a symbolic allele or
#                          a complex substitution, such as ACT>TCGA
#   number of multiallelic sites     .. number of rows with multiple alternate alleles
#   number of multiallelic SNP sites .. number of rows with multiple alternate alleles, all SNPs
#
#   Note that rows containing multiple types will be counted multiple times, in each
#   counter. For example, a row with a SNP and an indel increments both the SNP and
#   the indel counter.
#
# SN  [2]id [3]key      [4]value
SN    0     number of samples:      6
SN    0     number of records:      2485860
SN    0     number of no-ALTs:      0
SN    0     number of SNPs:   2279024
SN    0     number of MNPs:   0
SN    0     number of indels: 206836
SN    0     number of others: 0
SN    0     number of multiallelic sites: 6315
SN    0     number of multiallelic SNP sites:   2274
```

MAF (Minor Allele Frequency)

After applying the maf filter to create a new VCF file containing only the variants that have a minor allele frequency of at least 0.05, we obtain the analysis_maf.log file that give information about process.

```
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 6 out of 6 Individuals
Outputting VCF file...
After filtering, kept 604781 out of a possible 848527 Sites
Run Time = 21.00 seconds
```

Principal Component Analysis (PCA)

PCA is a statistical technique used to reduce the dimensionality of a dataset. It does this by identifying the directions in which the data varies the most, and transforming the data so that these directions are aligned with the axes of the new coordinate system. The transformed data can then be represented using fewer dimensions, while still retaining as much of the original information as possible. PCA is often used to visualize high-dimensional data, or to identify patterns and relationships in the data that might not be apparent when the data is examined in its original form.

After running the commands in the "Command_lines_up201805012.txt" file, the following plots were obtained:

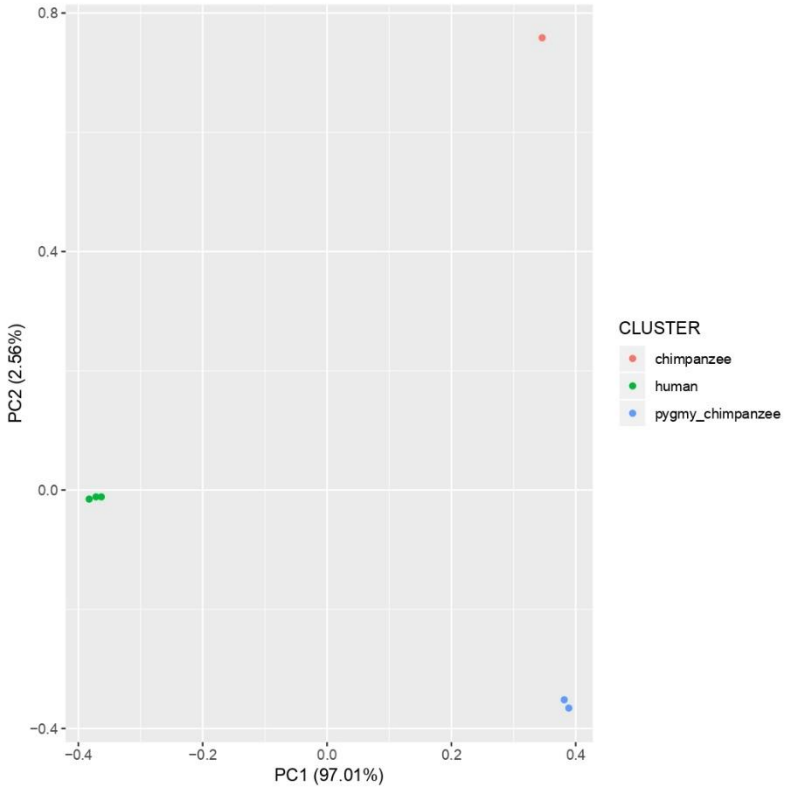

Figure 1. PCA plot for the SNPs.

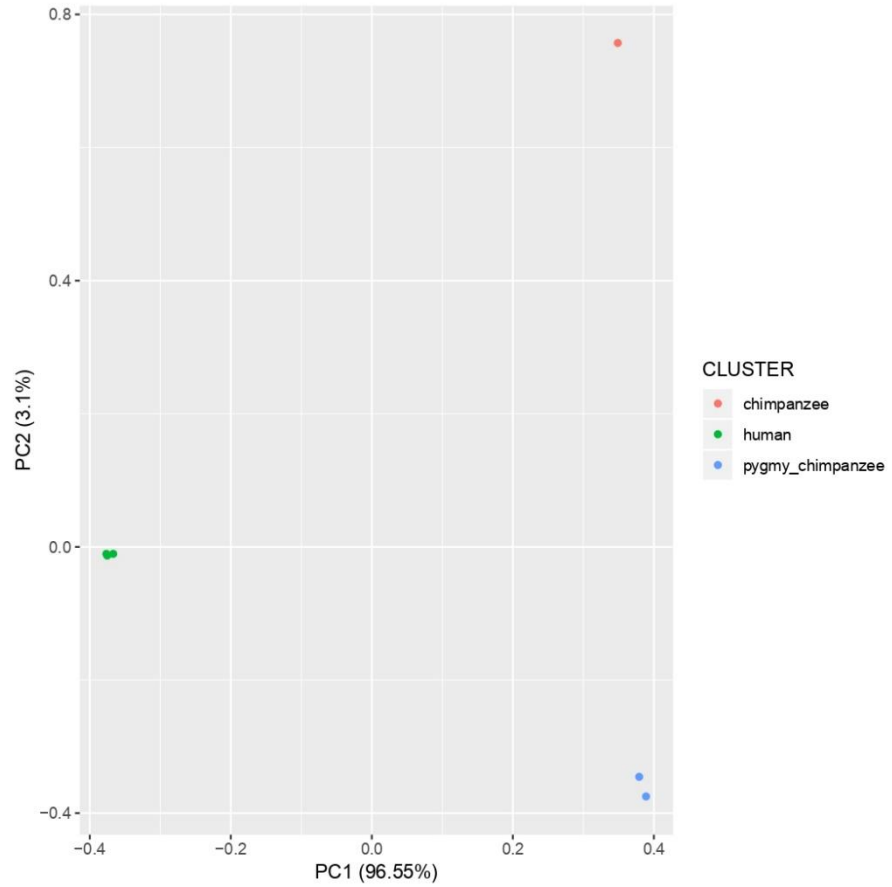

Figure 2. PCA plot for the INDELS.

Usually PCA plots are used for a large amount of samples, which is not the case here.

In a PCA plot, each data point is represented by a point in the graph, and the position of the point reflects the values of the variables for that data point. They can help to identify clusters or groups of similar data points, and to understand the relationships between the variables in the data. The first principal component is usually plotted on the x-axis, and the second principal component is plotted on the y-axis.

In both plots the 6 samples that correspond to chimpanzee, human and pygmy chimpanzee are represented.

It is possible to observe a larger overlap of the points corresponding to human in the INDELS plot. On the other hand, in this plot the points corresponding to pygmy chimpanzee are more separated than in the SNPs plot.

The value on the x-axis of the point representing the chimpanzee is close to the value of the pygmy chimpanzee points, which may reveal some significant relationships between the two. On the other hand, the point representing the chimpanzee is very distant from the points representing the human.

The value on the y-axis of the points representing the human is somewhat similar with the points representing the pygmy chimpanzee, which suggests the presence of a relationship between the two.