UNIX command line tools for manipulation and analysis of genomic data

The objective of this practice is learning the usage of some powerful UNIX command line tools, which allow manipulating and extracting information of large files without relying on specific software.

UNIX command line tools are not available by default in Windows-based systems. Although there are several ways of installing them, for practical reasons we will use a portable version of MobaXterm, available in the tools provided for the course.

- First open a terminal and navigate to the directory where you extracted the course data by using the `cd` command.

  In MobaXterm your drives are located in the directory /drives. For example navigating to C: can be done with the command:
  - `cd /drives/c`

  In Windows Linux Subsytem (WLS) file systems are mounted in /mnt
  - `cd /mnt/c`

- Check your current working directory
  - `pwd`

- List the contents
  - `ls`

- List the contents of the fastq directory
  - `ls fastq`

- List the index files of the bam
  - `ls bam/*.bai`

- Count the number of lines of this file
  - `wc -l fastq/1M_SRR9336457.fastq`
  - How many reads does this file contain?

- Create software directory (we will use it later)
  - `mkdir software`

- Inspect the contents of the 1M_SRR9336457.fastq file
  - `less fastq/1M_SRR9336457.fastq`

- Print the first 20 lines of the 1M_SRR9336457.fastq file

  -
    ```
    head -20 fastq/1M_SRR9336457.fastq
    ```

- Concatenate BED files into a

  -
    ```
    cat bed/regions_example.bed bed/regions_example2.bed bed/regions_example3.bed > bed/all_regions.bed
    ```

- Sort BED files

  -
    ```
    sort -k 1,1 -k2,2n bed/all_regions.bed > bed/all_regions.sorted.bed
    ```

- List unique chromosomes in the BED file

  -
    ```
    cut -f 1 bed/all_regions.sorted.bed | uniq
    ```

- List the contents of the directory with the reference genome file

  -
    ```
    ls chromosomes
    ```

- What are the names of the chromosomes in the reference genome?

  -
    ```
    grep '>' chromosomes/Saccharomyces_cerevisiae.R64-1-1.dna.toplevel.fa
    ```

- How many features of each type are in the genome annotation file?

  -
    ```
    cut -f 3 genes/Saccharomyces_cerevisiae.R64-1-1.99.gff3 | sort | uniq -c
    ```

- Remove chromosomes from GFF file (see chromosomes first)

  -
    ```
    grep -P '\tchromosome\t' genes/Saccharomyces_cerevisiae.R64-1-1.99.gff3
    ```

  -
    ```
    grep -v -P '\tchromosome\t' genes/Saccharomyces_cerevisiae.R64-1-1.99.gff3 > genes/Saccharomyces_cerevisiae.R64-1-1.99.noChr.gff3
    ```

- Create BED file containing only genes

  -
    ```
    grep -P '\tgene\t' genes/Saccharomyces_cerevisiae.R64-1-1.99.gff3 | cut -f 1,4,5,9 > bed/genes.bed
    ```

  -
    ```
    less bed/genes.bed
    ```

- Note that GFF files are 1 index and BED files are 0-index, therefore the last conversion is erroneous. See example in IGV

- Extract only sequences from FASTQ file

  -
    ```
    grep -P '^[ACGTN]+\r$' fastq/1M_SRR9336457.fastq > 1M_SRR9336457.txt
    ```

- Count how many unique reads are there in the new 1M_SRR9336457.txt file.

  -
    ```
    sort -u 1M_SRR9336457.txt | wc -l
    ```

- Count how many times each read appears and store the information in an new file named 1M_SRR9336457.txt-counts.tsv.
  - `sort 1M_SRR9336457.txt | uniq -c > 1M_SRR9336457.txt-counts.tsv`

- Reads in 1M_SRR9336457.txt are stored as DNA, however some programs require an input where sequences are stored as RNA. Therefore we would like to change Ts to Us. How can we do that?
  - `tr T U < 1M_SRR9336457.txt > 1M_SRR9336457-RNA.txt`

INSTALLATION

1. Install libraries

- MobaXterm

```
apt-get install make
apt-get install gcc-g++
apt-get install zlib-devel
apt-get install libbz2-devel
apt-get install liblzma-devel
apt-get install libncurses-devel
```

- Ubuntu

```
sudo apt-get install make
sudo apt-get install g++
sudo apt-get install libncurses5-dev
sudo apt-get install zlib1g-dev
sudo apt-get install libbz2-dev
sudo apt-get install liblzma-dev
```

2. Download Samtools source (samtools-1.10.tar.bz2) from http://www.htslib.org/download/ into “software” directory

3. Navigate to downloaded samtools directory

```
cd software
```

4. Unzip and compile Samtools

```
bunzip2 samtools-1.10.tar.bz2
tar -xvf samtools-1.10.tar
cd samtools-1.10
./configure
make
```

5. Download Bedtools source code (zip) into “software” directory https://github.com/jchenpku/bedtools2-cygwin/releases (MobaXterm)

https://github.com/arq5x/bedtools2/archive/v2.29.2.zip (Others)

6. Navigate to downloaded bedtools directory

```
    cd ../../software
```

7. Unzip and compile Bedtools

MobaXterm

```
        unzip bedtools2-cygwin-2.29.2.zip
        cd bedtools2-cygwin-2.29.2
        make static
```

Others

```
        unzip bedtools2-2.29.2.zip
        cd bedtools2-2.29.2
        make static
```

Note: Both Samtools and Bedtools can be installed using package managers for Linux (apt-get) and OS X (brew or macports)

Linux:

```
        apt-get install samtools
        apt-get install bedtools
```

OSX:

```
        /usr/bin/ruby -e "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/master
/install)"
        brew install samtools
        brew install bedtools
```

SAMTOOLS

- Get flag summary of bam alignment
    - `software/samtools-1.10/samtools flagstats bam/1M68_pH5_0.04CO2_R1.bam`

- Extract reads that are properly aligned
    - `software/samtools-1.10/samtools view -f 2 bam/1M68_pH5_0.04CO2_R1.bam | less`