In this tutorial, we will first download some data from the NCBI. With the NGS advent, data are stored in NCBI in SRA format. We need to convert them to the fastq format to further analyze them.

Since sequencing data are produced by a complex pipeline that comprises PCR amplification, DNA elongation and DNA reading, often error have been accumulated in this file. We use some measures to evaluate the quality of the data. This process is called Quality Control (QC).

Then data are mapped to a reference genome. This means that all of the millions short or long reads are processed using computer software and they are matched to a reference genome. This process is called mapping.

In this brief tutorial we will see (not in much detail) the processes followed in every step of the pipeline.

Download data from NCBI

We will use the ‘sra-toolkit’ to obtain data from NCBI.

1. Install sra-toolkit   
   In several linux distributions, sra-toolkit can be downloaded using the command ‘sudo apt install sra-toolkit’. You can do this, but here we will use the github version (it might be a newer version). They offer a precompiled version for Ubuntu, which should be suitable for debian as well. To get it use:  
     
   wget ‘<https://ftp-trace.ncbi.nlm.nih.gov/sra/sdk/3.0.1/sratoolkit.3.0.1-ubuntu64.tar.gz>’  
   tar xvfz sratoolkit.3.0.1-ubuntu64.tar.gz  
   cd sratoolkit.3.0.1-ubuntu64/  
     
   There, there is a ‘bin’ directory with all the executables. To use them is better to modify the .bashrc file in the home folder and put this directory in the PATH. For example:  
     
   emacs -nw .bashrc  
     
   at the end of the file just add  
     
   PATH=<YOUR PATH>/sratoolkit.3.0.1-ubuntu64/bin/:$PATH  
   export PATH  
     
   Then, in the command line, type:   
   source ~/.bashrc   
     
   To confirm that you run the correct executables (it might be that the computer has already older versions of the software) just type:  
   which -a fasterq-dump  
     
   You should get back the fasterq-dump located in the bin directory of the sratoolkit you just have downloaded.
2. Find some data to download:  
   Perhaps this is the most difficult part of the whole analysis. Which data to use for your analysis. If you are going to use public data, then this step includes a manual work through the SRA (or other databases, such as the ENA) to get the IDs of the data that you want to use. I still haven’t found a good solution for this (how to get the data IDs automatically). Here, we will use *Arabidopsis thaliana* data. In the ENA directory (<https://www.ebi.ac.uk/ena/browser/view/PRJNA273563>) they do provide a file for the accession names of the data. You can download this file as:  
     
   curl -o arabidopsis\_accessions.tsv ‘<https://www.ebi.ac.uk/ena/portal/api/filereport?accession=PRJNA273563&result=read_run&fields=study_accession,sample_accession,experiment_accession,run_accession,tax_id,scientific_name,fastq_ftp,submitted_ftp,sra_ftp&format=tsv&download=true&limit=0>’  
     
     
   The file looks like this:  
     
   study\_accession sample\_accession experiment\_accession run\_accession tax\_id scientific\_name fastq\_ftp submitted\_ftp sra\_ftp

PRJNA273563 SAMN03326279 SRX972118 SRR1945435 3702 Arabidopsis thaliana ftp.sra.ebi.ac.uk/vol1/fastq/SRR194/005/SRR1945435/SRR1945435\_1.fastq.gz;ftp.sra.ebi.ac.uk/vol1/fastq/SRR194/005/SRR1945435/SRR1945435\_2.fastq.gz ftp.sra.ebi.ac.uk/vol1/srr/SRR194/005/SRR1945435

PRJNA273563 SAMN03326280 SRX972119 SRR1945436 3702 Arabidopsis thaliana ftp.sra.ebi.ac.uk/vol1/fastq/SRR194/006/SRR1945436/SRR1945436\_1.fastq.gz;ftp.sra.ebi.ac.uk/vol1/fastq/SRR194/006/SRR1945436/SRR1945436\_2.fastq.gz ftp.sra.ebi.ac.uk/vol1/srr/SRR194/006/SRR1945436

PRJNA273563 SAMN03326281 SRX972120 SRR1945437 3702 Arabidopsis thaliana ftp.sra.ebi.ac.uk/vol1/fastq/SRR194/007/SRR1945437/SRR1945437\_1.fastq.gz;ftp.sra.ebi.ac.uk/vol1/fastq/SRR194/007/SRR1945437/SRR1945437\_2.fastq.gz ftp.sra.ebi.ac.uk/vol1/srr/SRR194/007/SRR1945437

Now, of course there is a link (column 7) that we can directly use to get the data from the EBI. However, we will use the sra-toolkit to demonstrate its usage.

Also, this will require several hours and large amounts of space in the HDD. Thus, we will just get a small portion of the data (e.g. the first 3 samples).

head -4 arabidopsis\_accessions.tsv | awk '{print $4}' | tail +2 > accessions\_to\_download.txt  
  
We wil use the prefetch command to get the SRA data  
  
prefetch --option-file accessions\_to\_download.txt  
  
“Both fastq-dump and fasterq-dump are faster when following prefetch, and fasterq-dump paired with prefetch is the fastest way to pull the files from the SRA. There are alternative options, which we will mention later.”

Next, we use fasterq-dump to convert the data in fastq format

for file in `find -iname '\*.sra' | xargs ls`; do echo $file; fasterq-dump $file; done  
  
This command first finds the sra files and then it uses the fasterq-dump to convert the data to fastq format.

1. Download the reference genome.   
   The reference genome can be downloaded from the NCBI, under the ‘Genome’ database  
     
     
     
   Use the ‘genome’ link to get the genome itself.
2. Now, we have the reads from three plants and a reference genome. You can use the bwa software to map the reads to the reference genome.   
     
   bwa comes with difference mapping tools (bwa mem and bwa aln).   
     
   \*\* Useful read: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8208613/>
3. Indexing the reference genome (to speed-up mapping)  
   Indexing
4. After indexing, we proceed with the mapping

REF: the reference file

SAMPLE: the RUN sample from the NCBI

bwa mem -t 20 $REF $SAMPLE\_1.fastq.gz $SAMPLE\_2.fastq.gz -o ${SAMPLE}.sam

## the previous command will initiate 20 threads to speed up calculations

## If reads come in paired files then we need to provide both the \_1 and the \_2 file (these files will be already in the folder if everything has been correctly downloaded). Of course, it might be the case that you don’t have paired end files, so you don’t expect to have \_1 \_2 files (each pair) in your directory.

1. Then, we proceed with the generation of the file (gvcf) containing what we have read in the reference and the sample in each location  
     
   SAMPLE: the sample file

REF: The reference file

## The next command fixes some mapping flags of the sam file

samtools fixmate -O bam ${SAMPLE}.sam ${SAMPLE}.bam

samtools sort -o ${SAMPLE}.sort.bam ${SAMPLE}.bam

bcftools mpileup -g 10 -Oz -o ${SAMPLE}.gvcf.gz -f ${REF} ${SAMPLE}.sort.bam

bcftools index ${SAMPLE}.gvcf.gz

1. bcftools merge -Oz --gvcf REF --merge all -o merged.vcf.gz \*.gvcf.gz

bcftools call -mv merged.vcf.gz -o merged.vcf.gz.mv.vcf