1. **Quality control**

***(adapted from Alex Bossers, Wageningen University & Research; alex.bossers@wur.nl)***

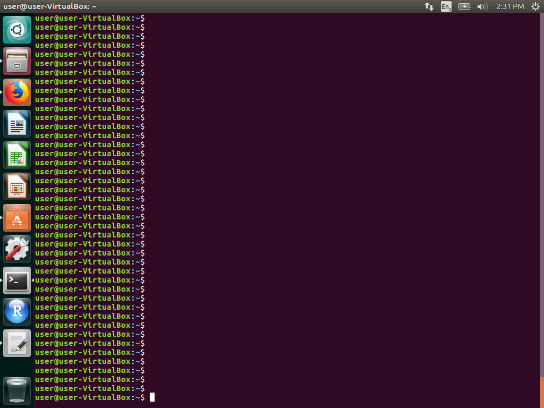
*The objectives of this section are to inspect raw sequencing files, perform some basic QC steps and look for contaminant sequences*

Before we can analyse metagenomics data (and deep-sequencing data in general) we need to make sure it comes from a proper experimental design (sampling), that we took all efforts to preserve sample diversity (sample handling) and that we isolated high quality DNA representing the original bacterial community as close as possible, by enriching for bacterial DNA contrasting to host genome-derived DNA.

As soon as high quality DNA was obtained from the collected samples, a sequencing strategy was chosen and libraries were generated, finally samples were deep sequenced on our platform of choice - Illumina short-read sequencing in paired-end modus 150bp each. Raw sequencing data was demultiplexed after the sequencing at the core facility, therefore the data provided is ready for inspection. The first analysis step is to perform quality control (QC) and some basic data descriptions. Follow this tutorial in order to perform the QC analysis and reply to the section II quizz.

1. **Discover command line**

After starting the (bash) shell (see red arrow in figure below) you can enter commands (**password: 1234**). Only enter blue parts of the command lines in this tutorial. Copy/paste is done by mouse or shortcut paste in the terminal: *ctrl-shift-v*



Which files are in the working directory? The parameter -l means show in long format (rights and size). This is not obligatory but usually quite handy.

ls -l

The command below will show the current working directory (path)

pwd

You change to another directory relative to your working directory using cd pathToWhereYouWantTOGo. Preceding it with a / means the absolute path relative to the top directory (root) for instance /home/user/Vboxshare

cd /home/user/Vboxshare

The ***TAB key*** can be very handy for auto completing commands and file/folder names! Repeating the last command(s) is done by the ***UP key***. Your lab journal can be retrieved by the command history. Try it! Terminating a running command can be done by ***ctrl-c***.

Moving or renaming files is done using mv. Note that the file system is CaseSenSiTive. Better NOT to use special characters and spaces in file or directory names. Doing so means you have to use quotes around them every time since the file system will see the space as a parameter separator! Use \_ instead.

File ReadMe.txt is different from file readme.txt on a unix file system.

mv oldfile newfile

mv oldfile otherdirectory/ !Note the trailing slash!

mv file /dev/null/ Complex way for deleting files (=del command:

del filename)

**!!! Files are immediately deleted and NO trash can !!!**

You can chain (pipeline) several commands where the output of the first is ‘piped’ to the second. For instance if we only want to show the top 3 lines of the output of some command we can use the head command which comes in very handy for quickly looking at LARGE text files! You do not want to open text files of several hundred MBs in a text editor.

ls -l | head -n 3

Some other basic command we will be using for inspecting and viewing files are:

cat <filename> : display text file contents. zcat does the same on zipped/gzipped text files.

less <filename> : displays a text file page by page and allows scrolling (similar to more).

grep .... : powerful command to i.e. search for specific occurrences in the text and count.

nano <filename> : a very simple command line text editor. nano filename. *Ctrl-x* to save and quit.

man <command> : request how a command should/can be used. Press Q to exit.

1. **Inspect raw sequencig files**

First, make sure the sample fastq files that you need for the exercise are placed in the right folder. You can do this in one of two ways:

1. download the zip. file Samples.R1.R2 and extract the two fastq files Sample.R1.fastq.gz and Sample.R2.fastq.gz into the shared folder Vboxshare, **or**
2. start the Virtual Machine, open the terminal and move the sample files from an already existing folder in the Virtual Machine into the shared folder Vboxshare, by running the following command:

mv /home/user/WorkshopData/\*gz /home/user/Vboxshare

Then, check if the sequencing files are in the proper location:

/home/user/Vboxshare

Sample.R1.fastq.gz

Sample.R2.fastq.gz

Now you are ready to start the quality control exercise.

1. Raw fastq sequencing files

We know our experiment is from a paired-end approach so by default we should have gotten at least two fastq raw sequencing files from the sequence provider for each sample. Corresponding file pairs usually have the .R1. .R2. identifier in the filename. Filenames can be renamed for convenience as we did for this tutorial:

Sample.R1.fastq.gz

Sample.R2.fastq.gz

But usually there will be a name that corresponds to the machine/project identifier-sample number, barcode sequence used in multiplexing approaches (almost always the case) and lane within the flow cell used (8 lanes per flow cell). R1 and R2 referring to the paired-set. Such names could look something like the example below:

HG22MBBXX\_102905-001-049\_CTGAAGCT-AGGCTATA\_L001\_R1.fastq.gz

HG22MBBXX\_102905-001-049\_CTGAAGCT-AGGCTATA\_L001\_R2.fastq.gz

HG22MBBXX\_102905-001-049\_CTGAAGCT-AGGCTATA\_L002\_R1.fastq.gz

HG22MBBXX\_102905-001-049\_CTGAAGCT-AGGCTATA\_L002\_R2.fastq.gz

HG22MBBXX\_102905-001-049\_CTGAAGCT-AGGCTATA\_L003\_R1.fastq.gz

HG22MBBXX\_102905-001-049\_CTGAAGCT-AGGCTATA\_L003\_R2.fastq.gz

...

...

HG22MBBXX\_102905-001-050\_CTGAAGCT-GCCTCTAT\_L001\_R1.fastq.gz

HG22MBBXX\_102905-001-050\_CTGAAGCT-GCCTCTAT\_L001\_R2.fastq.gz

HG22MBBXX\_102905-001-050\_CTGAAGCT-GCCTCTAT\_L002\_R1.fastq.gz

Since these are TEXT files they can be compressed quite easily with zip or gzip. Our demo dataset is compressed by gzip (.gz extension). Most tools are nowadays capable of directly using gzipped files as input without the need to deflate which takes up lots of disk space.

Note that for viewing these files we need either the commands for regular text files (cat, grep) or those designed for zipped files (zcat, zgrep).

1. Inspecting the format

Fastq sequencing files have a very specific make up where all read info of a single read is described in exact 4 lines.

1 – header of that read

2 – the sequence itself

3 – a + or sometimes the header again to denote that the following line are the quality scores

4 – quality score per base (PHRED scoring)

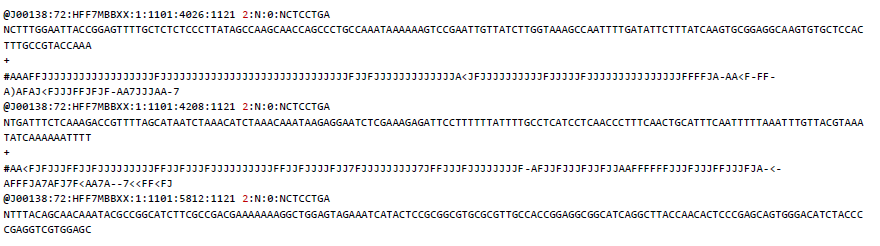
Let us inspect the first 10 line of the sequencing file. Depending on your console width they will fit nicely on 10 lines or they will be wrapped. Enlarge your console window and/or lower your font size.

zcat Vboxshare/Sample.R1.fastq.gz | head Use head –n 20 to show 20 lines...



Now see if the read file for the corresponding R2 is the same...

zcat Vboxshare/Sample.R2.fastq.gz | head



**Note 1:** the header is identical except for the read-pair identifier 1 or 2 (see above). Of course the sequence and quality score is different.

**Note 2:** the order of the reads is the same! This is important when working with PE data that the sequence files are kept and processed together so that no orphan reads will occur in any of the files. These reads disconnect the link between the reads since most tools **expect the files to be in the same order**. Therefore we will be QC quality curating both read files simultaneously and use both in mapping (in section III).

1. Counting the number of reads

So by requesting the directory list using ls -l we already see the file size. More informative would be to know the number of reads in each file:

zgrep -c '@J00138:' Vboxshare/Sample.R1.fastq.gz

Zgrep (or grep on non-zipped files) let you find text patterns in text files. In the example above the pattern is the start off the sequence identifier @J000138: but we could as well take another unique identifier that occurs only once in each read-set. The command line switch -c will only return the count instead of listing every hit. Try it when you leave it out!

We want to check the number of reads in the R1 AND R2 files, which should be equal, if nothing went wrong somewhere (in copy, in clean-up,....). In this case, we can use wildcards (\*) like below:

zgrep -c '@J00138:' Vboxshare/Sample.R\*.fastq.gz

QUESTION 1:

How many reads does this dataset have?

1. **Perform some basic QC**

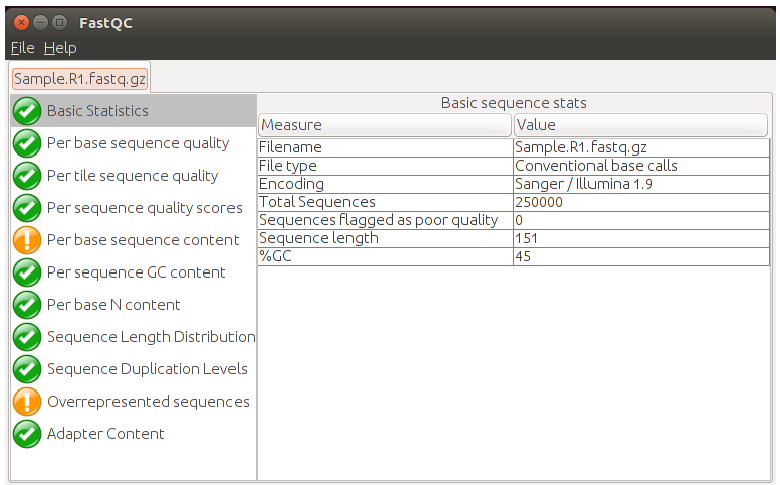
More details can be found here: https://www.bioinformatics.babraham.ac.uk/projects/fastqc/

We can both work with fastqc from the command line as well as using a convenient GUI (graphical application). The results should be equal. The command line is convenient when you have to process many files and only want some of the output data to be present in other reports. You don’t want to open several hundreds of files for manual inspection. In this exercise, you will work with the GUI.

In the terminal window, go to your home directory if you aren’t there yet: cd ~

Start the tool without any arguments so it will invoke the graphical version: fastqc

The tool should start, you can usually ignore any output to the console. Use the file -> open to open one of our sequencing files in /home/user/Vboxshare. It takes a short time for the tool to get the QC data and it will show an overview of the analyses:



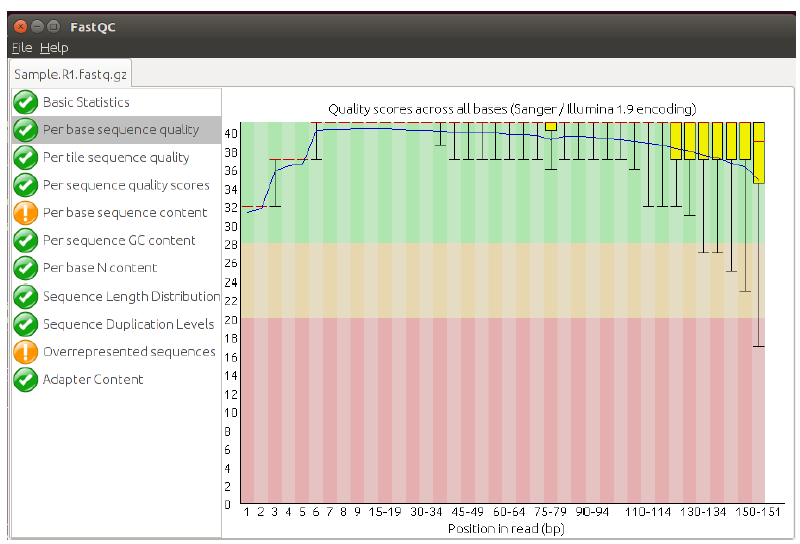
QUESTION 2:

It will show some basic descriptive statistics of the type of sequencing file and reads.

Does the read number shown in the fastQC report match the number of reads shown with the command line zgrep ?

It also shows the sequence **length** and the **encoding** of the fastq file to be of the later type having the PHRED quality scores and not an older *Illumina-Solexa* scoring system. If all topics have a green checkmark this is usually a good sign. But we still want to do some additional quality curation because we are picky... We also see 2 topics that are flagged which we need to inspect. Not necessarily meaning a bad thing. Metagenomics data or very deep bacterial whole genome data can flag overrepresented sequences quite easily due to the extreme depth of sequencing...

Most important to check is the “Per base sequence quality” analysis:



Here the boxplots show the overall quality scores per position in the sequence reads. You can ignore the beginning starting at 32 and climbing to 40. More important is that we see that the overall sequence quality is dropping the longer the read is. This is normal for sequencing by synthesis technology like Illumina is using. Illumina can produce reads up to 300 bases and there the effect is even stronger.

In our example we see that at the end sequences exist that have a quality score below 28 or even 20. In many applications, a cut-off of 20 is used. In section III, you will trim (shorten) the sequences by removing bases below a certain quality score.

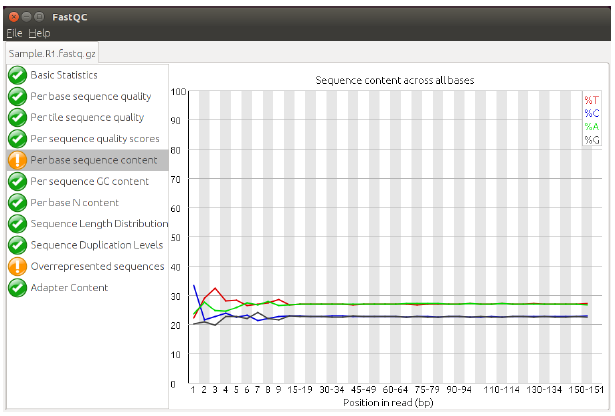
QUESTION 3:

Inspect the R2 (the reversed) file of this read set. Is the quality distribution similar or better/worse?

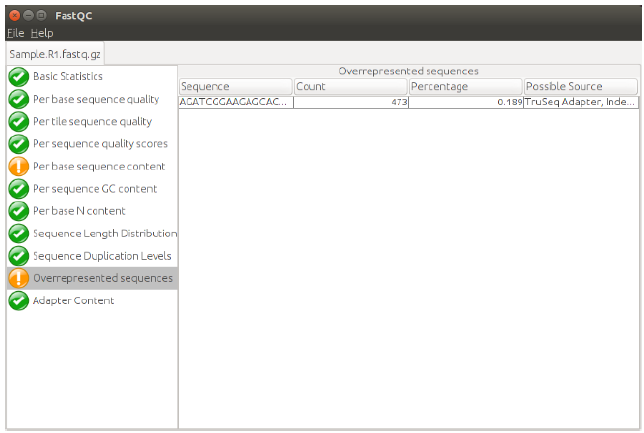
Let us continue and inspect the first orange exclamation mark. It is the “per base sequence content”. Overall, we see a certain balance G/C and A/T percentage. But at the start of the reads we see a disturbed distribution indicating that from the start until base 10-15 it is not entirely random.

Even though the steps to shear the DNA in library preparation is by sonication and expected to be random, it is not entirely and we see this preferential sequence enrichment. This should normally be zero or very low. If a sequence has a high % of bases that can be read, this disturbed distribution is usually an adapter or primer clipping problem, which can be fixed most of the time during the trimming step.

Another method to generate a library is Nextera, where instead of sonication, transposon-insertion is used to shear/tag DNA (tag-mentation). This is at TA sites in the genomes. Depending on the subject(s) to be sequenced, this can give a strong bias in the first 10 bases.



The final two sets focus on technical contaminants by adapters or other somehow overrepresented sequences (for instance primers used in amplicon based strategies). We usually fix any issues in the trimming steps.



1. **Looking for contaminants**

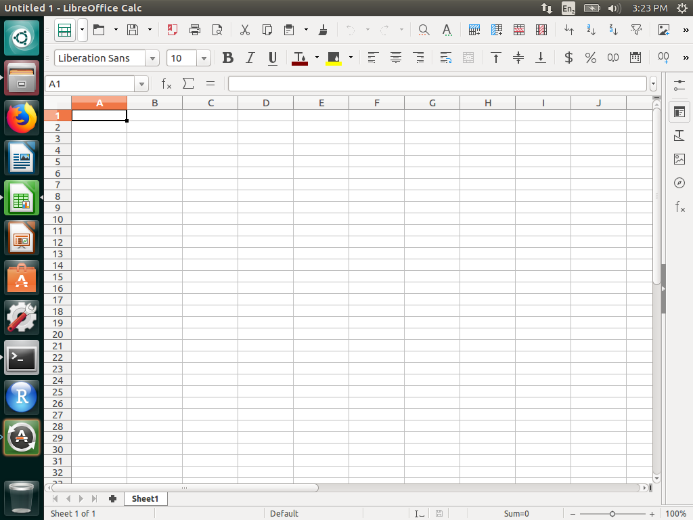
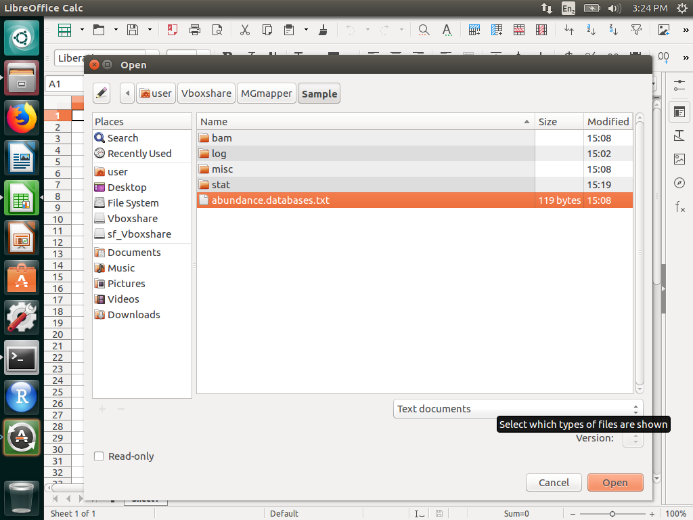
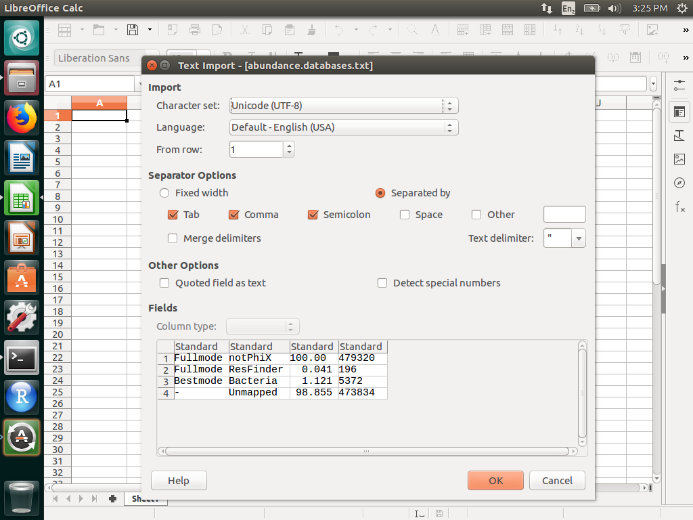
Deep-sequencing datasets usually have some common ‘technical’ contaminants depending on the technology used. Such technological contaminants for the Illumina platform are for instance the spike-in control PhiX DNA that is needed for run-performance monitoring, or residual adapters, primers and barcodes in multiplexed sequencing. Another form of contamination in especially metagenomic deep-sequencing is the contamination with host genome-derived DNA which in bad cases can easily consume 90%+ of sequencing capacity (which is a waste). The host-genome-derived contamination is one of the things where a good experimental design and nucleic acid isolation procedure is going to make you happy.

A common way to look for and filter contaminants is by mapping reads to contaminant and reference genome databases - an approach we will be using. In the previous fastqc steps, we might already have detected some contaminant adapter sequences or sequences coming from PhiX.

Here we will map our reads to the contaminant database PhiX using a special short-read mapper (similar to BLAST but highly optimised for sort read mapping). The mapper is called bbmap and comes from the BBMAP suite of Brian Bushnell. **The command below should be pasted/typed all in one line!**

tools/bbmap/bbmap.sh semiperfectmode=t in=Vboxshare/Sample.R1.fastq.gz ref=tools/MGmapper/db/phiX174/phiX174 scafstats=Vboxshare/phiX\_mapping\_before.txt out=/dev/null nodisk -Xmx800m

The output should be in the table phiX\_mapping\_before.txt in the scafstats output file, which you will find in the Vboxshare folder. You can open it using the command less Vboxshare/phiX\_mapping\_before.txt (use *q* to quit, *w* to scroll up and *space* tab to scroll down) or using the LibreOffice Calc (Ubuntu equivalent to Microsoft Excel), and selecting .txt file type – see figure below.



QUESTION 4:

How many reads map to the phiX genome?

1. **Clean-up raw sequencing data for further processing (trimming)**

Before we are going to use the sample dataset we are going to clean it up:

- We clip any contaminating adapters and barcodes

- We clip any bases from the ends of the reads having a quality lower than Q20

- We trash sequence reads shorter than 30 bases

- We also filter PhiX

There are many tools available for doing this job including cutadapt and bbduk. We will use the latter from the BBMAP suite to do all above steps in one run

Note that it is important to process R1 and R2 in paired-end mode!

**NB: remember to paste/type the command below in a single line!**

Now run the clean-up:

tools/bbmap/bbduk.sh in1=Vboxshare/Sample.R1.fastq.gz in2=Vboxshare/Sample.R2.fastq.gz out1=Vboxshare/Sample.R1.trim.fastq.gz out2=Vboxshare/Sample.R2.trim.fastq.gz ref=adapters,phix k=21 mink=6 ktrim=r ftm=5 qtrim=rl trimq=20 minlen=30 overwrite=true -Xmx800m

Now we generated the trimmed dataset to be used for the further exercises. The output files of this step are called Sample.R1.trim.fastq.gz and Sample.R2.trim.fastq.gz and they are located in the Vboxshare folder. Please use these names for the output files, since you will need them in the exercise of the next section!

Open the Vboxshare/Sample.R1.trim.fastq.gz in fastqc and check the read quality.

QUESTION 5: *Did quality improve and were adapters removed?*

Now also check if PhiX was successfully removed by repeating step 4 (looking for contaminants) but using the trimmed dataset:

tools/bbmap/bbmap.sh semiperfectmode=t in=Vboxshare/Sample.R1.trim.fastq.gz ref=tools/MGmapper/db/phiX174/phiX174 scafstats=Vboxshare/phiX\_mapping\_trimmed.txt out=/dev/null nodisk -Xmx800m

The output should be in the table phiX\_mapping\_trimmed.txt in the scafstats output file, which you will find in the Vboxshare folder. You can open it using the command less Vboxshare/phiX\_mapping\_trimmed.txt or using the LibreOffice Calc (Ubuntu equivalent to Microsoft Excel), and selecting .txt file type.

QUESTION 6 How many reads map to the phiX genome now, after trimming ?.

QUESTION 7: *What is the read number of the trimmed sample* Sample.R1.trim.fastq.gz?

All done! Your data is ready for mapping.