

Quality Control for Raw Data.

Practical session

Exercise 1. **Several samples** from FastQC

1. Open a terminal window

The programs used in this tutorial are called from the command line. In order to do that, the first step is to open a Terminal window. To do this go to:

Applications → **Accessories** → **Terminal** 

A new window will open with a prompt ready for an input. Now change to the directory with the sequence data. Type on the terminal:

```
cd /home/biouser/fastq
```

2. Open FastQC program

To start the FastQC program, you have to type on the terminal window:

```
fastqc &
```

The FastQC application will start in a new window.

You can minimize the terminal window, but **do not close it** while using the FastQC application. Otherwise, FastQC will be also closed.

3. Load a file into FastQC

From the FastQC program, go to:

File → **Open**

And load from the folder **/home/biouser/fastq** the file called

```
mirna.fastq
```

4. Look at the different FastQC result sections and answer the following questions

Questions:

We have a panel of genes to confirm Charcot-Marie-Tooth disease for several people.

A. Sample [panel1_1.fq](#)

1. Do a quality control for the sample **using FastQC**.
2. How much sequences are there?
3. Describe the sequence length: minimum and maximum
4. Did you detect adapters?
5. Any repeated sequences?
6. What are the parameters you consider bad quality indicators?
7. Any comments?

B. Sample [panel1_2.fq](#)

[panel1_1.fq](#) and [panel1_2.fq](#) are two fastq files for the same person. Could you reply the same questions for [panel1_2.fq](#)?

1. Do a quality control for the sample **using FastQC**.
2. How much sequences are there?
3. Describe the sequence length: minimum and maximum
4. Did you detect adapters?
5. Any repeated sequences?
6. What are the parameters you consider bad quality indicators?
7. Any comments?

C. Samples [panel2_1.fq](#) and [panel2_2.fq](#)

- This is the moment for the second panel. Could you evaluate quality control for both fastq files of the panel 2?
- Are there common patterns between [panel1](#) and [panel2](#)?

Exercise 2. Sample **mirna.fastq**

1. Open a terminal window

The programs used in this tutorial are called from the command line. In order to do that, the first step is to open a Terminal window. To do this go to:

Applications → **Accessories** → **Terminal**



A new window will open with a prompt ready for an input. Now change to the directory with the sequence data. Type on the terminal:

cd /home/biouser/fastq

2. Open FastQC program

To start the FastQC program, you have to type on the terminal window:

fastqc &

The FastQC application will start in a new window.

You can minimize the terminal window, but **do not close it** while using the FastQC application. Otherwise, FastQC will be also closed.

3. Load a file into FastQC

From the FastQC program, go to:

File → **Open**

And load from the folder **/home/biouser/fastq** the file called

mirna.fastq

4. Look at the different FastQC result sections and answer the following questions

Questions:

Sample **mirna.fastq**

1. Do a quality control for the sample **using FastQC**.

What are the parameters you consider bad quality indicators?

Write down your conclusions:

2. Trim your sample based on its quality with a **minimum quality threshold of 20**. You should use `Fastx_toolkit`, so type on the terminal:
`module load fastx_toolkit/0.0.13`
`fastq_quality_trimmer -t 20 -i mirna.fastq -o mirna_t20.fastq`

And now you do the quality control for the new file `mirna_t20.fastq` from FastQC.
What are the main changes?
Write down your conclusions:

3. Trim the sample based on its quality with a **minimum quality threshold of 28**. You should use `Fastx_toolkit`, so type on the terminal:
`fastq_quality_trimmer -t 28 -i mirna.fastq -o mirna_t28.fastq`

And now you do the quality control for the new file `mirna_t28.fastq` from FastQC.
Is there any quality improvement over the previous filter?
Write down your conclusions:

4. Trim the sample based on its quality with a **minimum quality threshold of 28**, removing the reads with a **length lower than 30**. You should use `Fastx_toolkit`, so type on the terminal:
`fastq_quality_trimmer -t 28 -l 30 -i mirna.fastq -o mirna_t28l30.fastq`

And now you do the quality control for the new file `mirna_t28l30.fastq` from FastQC.
Is there any quality improvement over the previous step?
How many reads have been removed?
Write down your conclusions:

5. Trim the sample based on its quality with a **minimum quality threshold of 28**, removing the reads with a **length lower than 35**.

You should use `Fastx_toolkit`, so type on the terminal:

```
fastq_quality_trimmer -t 28 -l 35 -i mirna.fastq -o mirna_t28l35.fastq
```

And now you do the quality control for the new file `mirna_t28l35.fastq` from FastQC.

Is there any quality improvement over the previous step?

How many reads have been removed?

Write down your conclusions:

Exercise 3. Sample **solid.fastq**

1. Do a quality control for the sample using **FastQC**.
What are the parameters you consider bad quality indicators?
Write down your conclusions:

2. **Trim** your sample based on its quality with a **minimum quality** threshold of **20**.
You should use `Fastx_toolkit`, so type on the terminal:
fastq_quality_trimmer -t 20 -i solid.fastq -o solid_t20.fastq

And now you do the quality control for the new file *solid_t20.fastq* from FastQC.
What are the main changes?
Do you consider the trimming to be effective?
Write down your conclusions:

3. **Trim** the sample based on its quality with a **minimum quality** threshold of **28**.
You should use `Fastx_toolkit`, so type on the terminal:
fastq_quality_trimmer -t 28 -i solid.fastq -o solid_t28.fastq

And now you do the quality control for the new file *solid_t28.fastq* from FastQC.
Is there any quality improvement over the previous filter?
Write down your conclusions:

4. Trim the sample based on its quality with a **minimum quality** threshold of **28**, removing the reads with a **length lower than 47**.

You should use `Fastx_toolkit`, so type on the terminal:

```
fastq_quality_trimmer -t 28 -l 47 -i solid.fastq -o solid_t28l47.fastq
```

And now you do the quality control for the new file `solid_t28l47.fastq` from FastQC.

Is there any quality improvement over the previous step?

How many reads have been removed?

Write down your conclusions:

5. Remove the reads with **less than a 90%** with **quality above 20**.

You should use `Fastx_toolkit`, so type on the terminal:

```
fastq_quality_filter -q 20 -p 90 -i solid.fastq -o solid_q20p90.fastq
```

And now you do the quality control for the new file `solid_q20p90.fastq` from FastQC.

Has the filter been effective?

How many reads have been removed?

Write down your conclusions:

Annex 1: Fastx_toolkit

fastq_quality_trimmer

usage: fastq_quality_trimmer [-h] [-v] [-t N] [-l N] [-z] [-i INFILE] [-o OUTFILE]
Part of FASTX Toolkit 0.0.13 by A. Gordon (gordon@cshl.edu)

- [-h] = This helpful help screen.
- [-t N] = Quality threshold - nucleotides with lower quality will be trimmed (from the end of the sequence).
- [-l N] = Minimum length - sequences shorter than this (after trimming) will be discarded. Default = 0 = no minimum length.
- [-z] = Compress output with GZIP.
- [-i INFILE] = FASTQ input file. default is STDIN.
- [-o OUTFILE] = FASTQ output file. default is STDOUT.
- [-v] = Verbose - report number of sequences.
If [-o] is specified, report will be printed to STDOUT.
If [-o] is not specified (and output goes to STDOUT), report will be printed to STDERR.

fastq_quality_filter

usage: fastq_quality_filter [-h] [-v] [-q N] [-p N] [-z] [-i INFILE] [-o OUTFILE]
Part of FASTX Toolkit 0.0.13 by A. Gordon (gordon@cshl.edu)

- [-h] = This helpful help screen.
- [-q N] = Minimum quality score to keep.
- [-p N] = Minimum percent of bases that must have [-q] quality.
- [-z] = Compress output with GZIP.
- [-i INFILE] = FASTA/Q input file. default is STDIN.
- [-o OUTFILE] = FASTA/Q output file. default is STDOUT.
- [-v] = Verbose - report number of sequences.
If [-o] is specified, report will be printed to STDOUT.
If [-o] is not specified (and output goes to STDOUT), report will be printed to STDERR.

Examples:

- Trimming of sequences with quality lower than 20:

```
fastq_quality_trimmer -t 20 -i <sample>.fastq -o <sample_out>.fastq
```

- Trimming of sequences with quality lower than 20 and minimum length of 30:

```
fastq_quality_trimmer -t 20 -l 30 -i <sample>.fastq -o <sample_out>.fastq
```

- Trimming of sequences with less than 90% of bases with quality above 20:

```
fastq_quality_filter -q 20 -p 90 -i <sample>.fastq -o <sample_out>.fastq
```