Quality Control

Luigi Grassi < *lg490@medschl.cam.ac.uk* >
Romina Petersen < *rp520@medschl.cam.ac.uk* >

TRAIN MALTA SUMMER SCHOOL 2016
September 2016

General information

The following standard icons are used in the hands-on exercises to help you locating:

- i Important Information
- 📄 General information / notes
- 👣 Follow the following steps
- Q Questions to be answered
- ! Warning – PLEASE take care and read carefully
- ✸ Optional Bonus exercise
- ✸ Optional Bonus exercise for a champion

Resources used

FastQC: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

fastx-toolkit: http://hannonlab.cshl.edu/fastx_toolkit/

Going on a blind date with your read set? For a better understanding of the consequences please check the data quality!

For the purpose of this tutorial we are focusing only on the Illumina sequencing which uses ‘sequence by synthesis’ technology in a highly parallel fashion. Although Illumina high throughput sequencing provides highly accurate sequence data, several sequence artefacts, including base calling errors and small insertions/deletions, poor quality reads and primer/adapter contamination are quite common in the high throughput sequencing data. The primary errors are substitution errors. The error rates can vary from 0.5-2.0% with errors mainly rising in frequency at the 3’ ends of reads. One way to investigate sequence data quality is to visualise the quality scores and other metrics in a compact manner to get an idea about the quality of a read data set. Read data sets can be improved by post processing in different ways like trimming off low quality bases, cleaning up the sequencing adapters if any, removing PCR duplicates if required. We can also look at other statistics such as, sequence length distribution, base composition, sequence complexity, presence of ambiguous bases etc. to assess the overall quality of the data set. Highly redundant coverage (>15X) of the genome can be used to correct sequencing errors in the reads before assembly and errors. Various k-mer based error correction methods exist but are beyond the scope of this tutorial. To investigate sequence data quality we would demonstrate tools called FastQC (http://www.bioinformatics.babraham.ac.uk/projects/fastqc/) and fastx-toolkit (http://hannonlab.cshl.edu/fastx_toolkit/). FastQC will process and present the reports in visual manner. Based on the results the sequence data can be processed using the fastx-toolkit.

Prepare the environmnet

Open the Terminal.

First, go to the right folder, where the data are stored.

```
cd /data/day2/QC/practical
```

Understand the quality encoding of your data

Before any bioinformatic analysis it is essential to know which quality encoding the fastq formatted reads are in.

Questions

1. Can you tell which quality encoding our good_example.fastq formatted reads are in? ____

   ____

   Hint: Look at the first few reads of the file `good_example.fastq` by typing:

```
head -n 20 good_example.fastq
```

2. Compare the quality strings with the table found at http://en.wikipedia.org/wiki/FASTQ_format#Encoding

Running FastQC

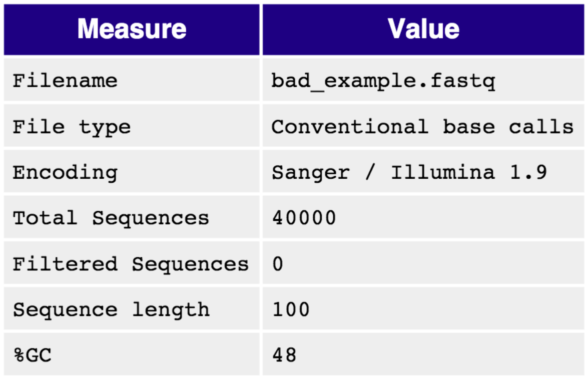
FastQC is installed on the server and can be launched from the command line. We are going to run it on the two fastq files we have (bad_example.fasq and good_example.fastq).

```
cd ~/scratch/day2
fastqc /data/day2/QC/practical/good_example.fastq -o .
fastqc /data/day2/QC/practical/bad_example.fastq -o .
```

Results are saved in the directory you specified as output directory (~/scratch/day2). You can visualise them using the Apache server link on the course web site.

Quality visualisation

On running FastQC, it will generate a QC report, containing several items. For example, the report file will have a **Basic Statistics** table and various graphs and tables for different quality statistics.



Measure	Value
Filename	bad_example.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	40000
Filtered Sequences	0
Sequence length	100
%GC	48

Figure 1: FastQC Basic Statistics

In addition, FastQC reports information about the quality scores of the reads.

Q-scores

A quality score (or Q-score) expresses an error probability. In particular, it serves as a convenient and compact way to communicate very small error probabilities. Given an assertion, A, the probability that A is not true, P(~A), is expressed by a quality score,Q(A), according to the relationship:

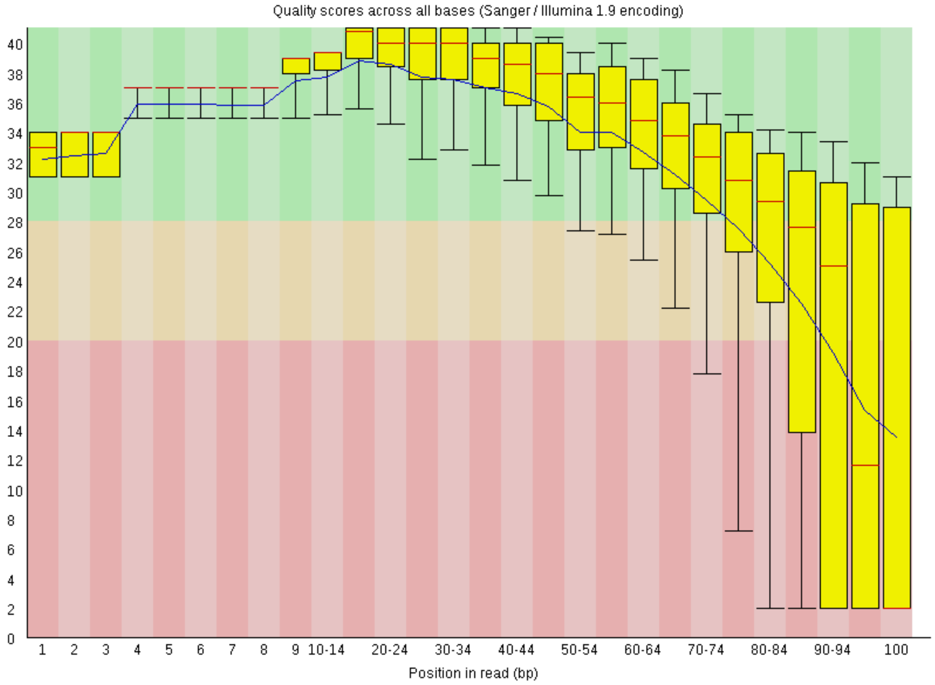


Figure 2: Per base sequence quality plot: visual output from FastQC. Base positions in the reads are shown on x-axis and quality score (Q Score) are shown on the Y-axis.

$$Q(A) = -10log10(P(\ A)) \tag{1}$$

where P(~A) is the estimated probability of an assertion A being wrong. The relationship between the quality score and error probability is demonstrated with the following table:

Quality score, Q(A)	Error probability, P(~A)
10	0.1
20	0.01
30	0.001
40	0.0001

Questions

1. How many sequences were there in the `bad_example.fastq`? What is the read length? ______________________________

____

   1. Does the quality score value vary throughout the read length? (hint: look at the 'per base sequence quality plot') ______________________
   2. What is the quality score range you see? ______________________
2. At around which position do the scores start falling below Q20? __________
3. Why does the quality deteriorate at the end of the read? ______________
4. How can we trim the reads to filter out the low quality data? __________

   ____

Sequencing errors can complicate the downstream analysis, which normally requires that reads be aligned to each other (for genome assembly) or to a reference genome (for detection of mutations). Sequence reads containing errors may lead to ambiguous paths in the assembly or improper gaps. In variant analysis projects sequence reads are aligned against the reference genome. The errors in the reads may lead to more number of mismatches than expected due to mutations alone. But, if these errors can be removed or corrected, the reads alignment and hence the variant detection will improve. The assemblies will also improve after pre-processing the reads with errors.

Read Trimming

The read trimming can be done in a variety of different ways. Choose a method which best suits your data. Here we are giving examples of fixed-base trimming and quality score-based trimming.

Fixed Length Trimming

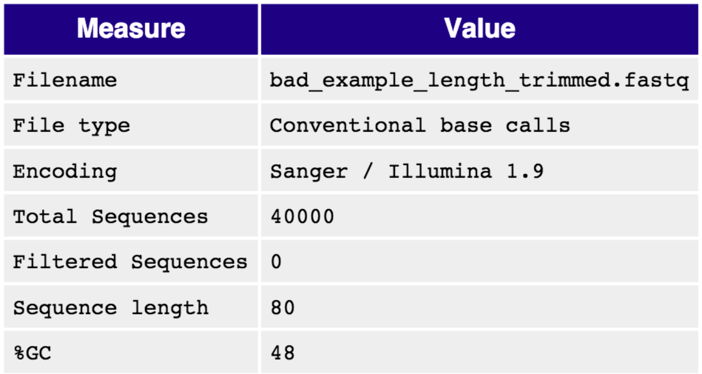
Low quality read ends can be trimmed using a fixed length timmer. We will use the **fastx_trimmer** from the fastx-toolkit. Load the corresponding module

```
module load fastx_toolkit/0.0.13.2
```

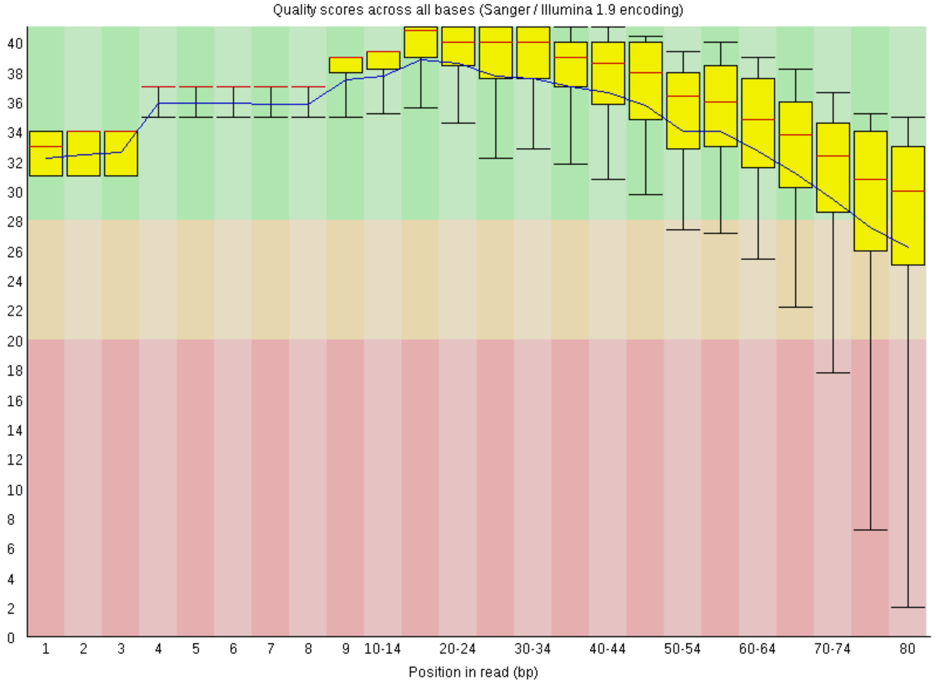
Type `fastx_trimmer -h` at anytime to display the various options you can use with this tool. In order to do fixed trimming with the fastq file `bad_example.fastq` use the following command. The output will be stored in `bad_example_length_trimmed.fastq`.

```
cd ~/scratch/day2/
fastx_trimmer -h
fastx_trimmer -f 1 -l 80 -Q 33 -i
   /data/day2/QC/practical/bad_example.fastq -o
   bad_example_length_trimmed.fastq
```

Run FastQC on the resulting file.



Measure	Value
Filename	bad_example_length_trimmed.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	40000
Filtered Sequences	0
Sequence length	80
%GC	48

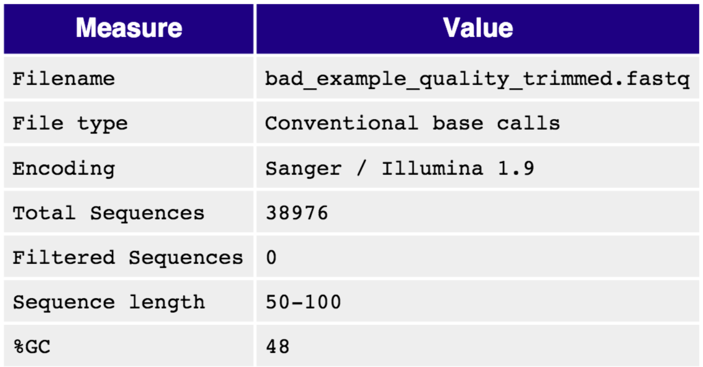


Quality Based Trimming

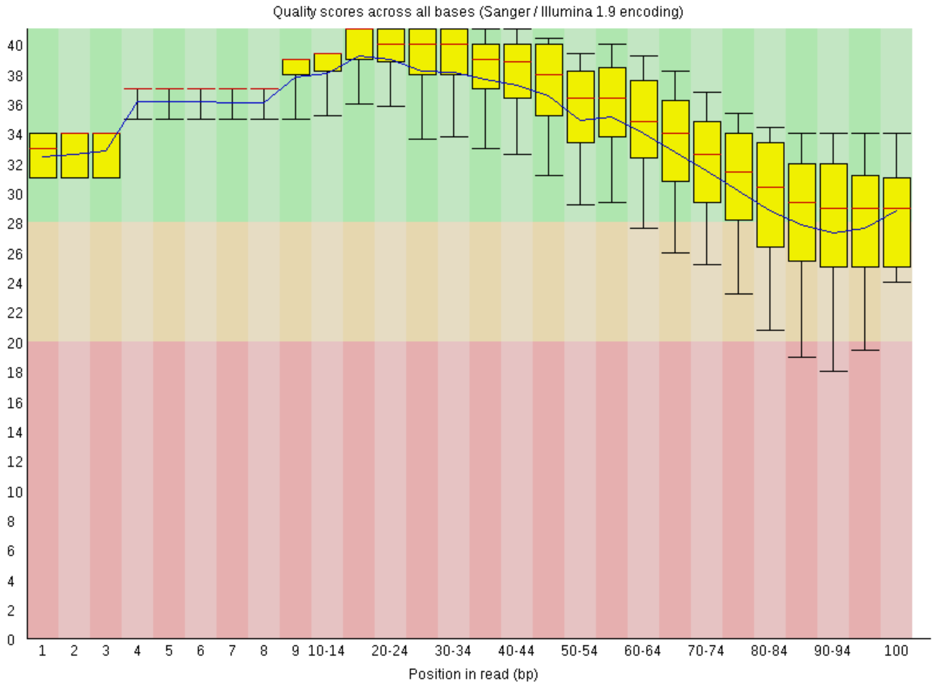
Base call quality scores can also be used for trimming sequence end. A quality score threshold and minimum read length after trimming can be used to remove low quality data:

```
cd ~/scratch/day2/
fastq_quality_trimmer -h
fastq_quality_trimmer -Q 33 -t 20 -l 50 -i
   /data/day2/QC/practical/bad_example.fastq -o
   bad_example_quality_trimmed.fastq
```

Again, run FastQC on the resulting file.



Measure	Value
Filename	bad_example_quality_trimmed.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	38976
Filtered Sequences	0
Sequence length	50-100
%GC	48



Questions

1. How did the quality score range change with the two types of trimming? ________
   ____
2. Did the number of total reads change after the two types of trimming? ________
   ____
3. What read lengths were obtained after quality based trimming? ________________
4. Did you observe adapter sequences in the data? ________________________________
   ____

Advanced Options:

Adapter Clipping

Sometime sequence reads may end up getting the leftover of adapters and primers used for the sequencing process. It's a good practice to screen your data for these possible contaminations for more sensitive alignment and assembly based analysis. This is usually a necessary steps in sequencing projects where read lengths are longer than the molecule sequenced, for example when sequencing miRNAs. Various QC tools are available to screen and/or clip these adapter/primer sequences from your data. (e.g. FastQC, fastx, cutadapt) Here we are demonstrating `fastx_clipper` to trim a given adapter sequence. Use `fastx_clipper -h` to display help at anytime.

```
cd ~/scratch/day2/
fastx_clipper -h
fastx_clipper -v -Q 33 -l 20 -M 15 -a
   GATCGGAAGAGCACACGTCTGAACTCCAGTCACCGATGTATCTCGTATGC -i
   /data/day2/QC/practical/adapter_contamination.fastq -o
   adapter_contamination_clipped.fastq
```

An alternative tool, not installed on this system, for adapter clipping is `fastq-mcf`. A list of adapters is provided as a list in a text file. For more information, see: http://code.google.com/p/ea-utils/wiki/FastqMcf