Trimming and quality control (June 2016)

Alexander Jueterbock, Martin Jakt*

PhD course: High throughput sequencing of non-model organisms

Contents

After a general introduction to the UNIX command line, it is time for you to analyze your own fastq files. The first important step for any kind of sequencing data is to get rid of adapter contamination and bad quality reads. In this tutorial we will use the programs FastQC and TrimGalore! to check the quality of the sequenced libraries before and after trimming.

IMPORTANT NOTE Before you get started: to compare characteristics of your libraries, please keep record of the resulting numbers, like the number of raw reads, reads after quality control, number of mapped reads etc. This helps to identify peculiarities/outliers in your libraries which may either be due to biological peculiarities of your species or unknown technical issues.

Log on (with `ssh`) to the remote computer with the `-X` option (like `ssh -x user@158...`) to be able to use graphical interfaces.

1 Overview of sequence lengths

Next Generation Sequencing data is generally stored in fastq files. Most of the time the data are compressed, either in .zip or in .gz format.

If your file is zip-compressed, you can use the following command to unzip it:

```
unzip FILE.fastq.zip
```

[*] University of Nordland, Norway

If your file iz gz-compressed, use the following command instead:

```
gunzip FILE.fastq.gz
```

NOTE: For paired-end sequencing from Illumina, you have two files. One file with the forward reads (`Sequence_R1.fq`) and one file with the reverse reads (`Sequence_R2.fq`).

To get a quick impression of the minimum and maximum read lengths in your fastq file, you can use the following commands (replace `FILE.fastq` with your own filename):

```
awk '{if(NR%4==2) print length($0)}' FILE.fastq| sort -n | head -n1
awk '{if(NR%4==2) print length($0)}' FILE.fastq| sort -n | tail -n1
```

It reads like this: measure the length of every second line in every group of 4 lines (the sequence line in a fastq file), `sort` it (numerically with `-n`) and print out either the first (smallest) value with `head` or the last (biggest) value with `tail`. `NR` represents the current line number and the `%` sign is the modulus operator, which divides the line number by 4 (`NR%4`) and returns only the remainder. This extracts all the sequences, which are on line 2,6,10,14. . .

The following command allows you to count the sequence lengths:

```
awk '{if(NR%4==2) print length($0)}' FILE.fastq | sort -n | uniq -c > read_length.txt
```

The lines that follow make use of the program R. If you copy and paste the code into the command line, you will get an overview graphic of the sequence length distribution

```
cat >> Rplot.r << 'EOF'
reads<-read.csv(file="read_length.txt", sep="", header=FALSE)

png(filename = "SequenceLengthDistribution.png",
         width = 480, height = 480, units = "px", pointsize = 12,
          bg = "white")
plot(reads$V2,reads$V1,type="l",xlab="read length",ylab="occurences",col="blue")
dev.off()

EOF


R CMD BATCH Rplot.r
```

You can open the created figure with the GNOME image viewer using the following command:

```
eog SequenceLengthDistribution.png
```

Does the sequence length-distribution meet your expectations?

NOTE: While the Ion Torrent sequences will differ in length (as in Figure 1), the Illumina sequences will all have the same read length (301 bp).

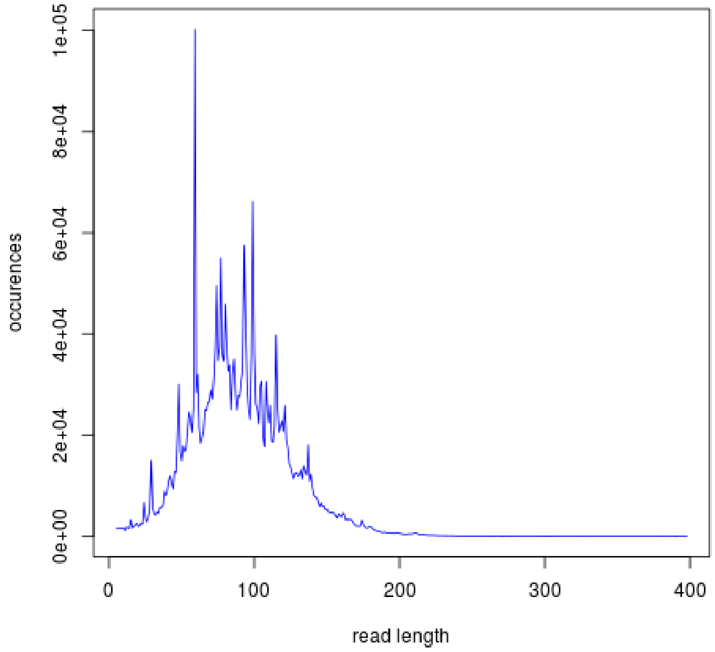


Figure 1: Example graphic of the length distribution in a fastq file

2 Quality control

To inspect the quality of the sequencing data, we use FastQC. In the installation and setup instructions of the program (link), you will find that FastQC can run in an interactive mode or in a command line mode. This tutorial uses the command-line version.

FastQC knows a number of standard adapter sequences used for HTS, including adapters used on Illumina platforms; To get an overview of the contaminants (overrepresented sequences) that FastQC will look for by default, type

```
less /usr/share/fastqc/Contaminants/contaminant_list.txt
```

However, FastQC is not aware of the sequences used by the IonTorrent platform. To inform FastQC of the Ion Torrent adapter sequences call FastQC with the `--contaminants` option to specify a file containing the adapter sequences we have used.

So, to run FastQC on your file, type:

```
fastqc --contaminants adapters.txt FILE.fastq
```

The adapters.txt file contains the adapter sequences in a name[tab]sequence format, like

```
IonTorrentAAdapter      CCATCTCATCCCTGCGTGTCTCCGACTCAG
IontTorrentP1Adapter    CCACTACGCCTCCGCTTTCCTCTCTATGGGCAGTCGGTGAT
```

The output of the FastQC program will be saved in a folder that has the name of your fastq file and ends with fastqc, like `FILE_fastqc`. Use the `cd` command to move into the folder and open the produced `fastqc_report.html` either with `firefox` or `chromium-browser` (one of the two should work).

```
cd FILE_fastqc
firefox fastqc_report.html
chromium-browser fastqc_report.html
```

You can also copy the output folder to your computer with FileZilla. Get familiar with the output of each module.

For example, it is normal that the the per base sequence quality drops towards the end of the read, as seen in Figure 2. In the next chapter we will see how to trim away these low-quality reads.

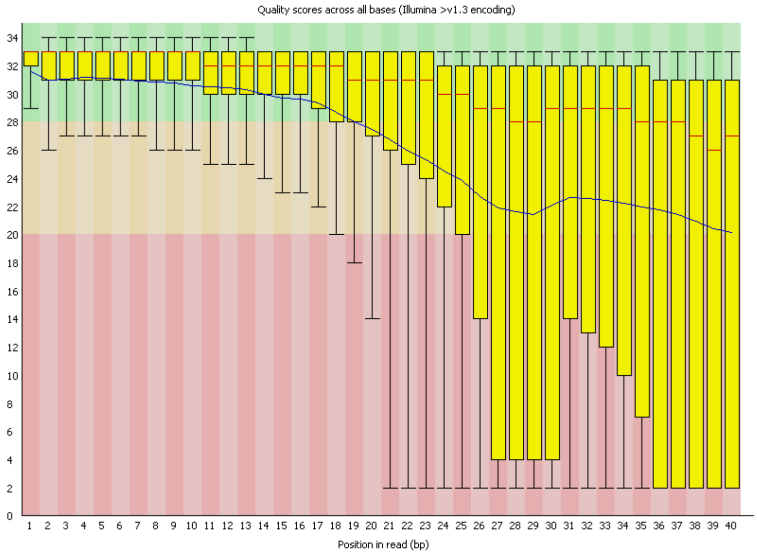


Figure 2: Per base sequence quality (from link)

The figure on duplication levels (Figure 3) informs you about the percentage of duplicate reads in your sequenced library. Duplicates result from primer or PCR bias towards these reads. As they can skew genotype estimates, we will remove duplicate reads later in the week before SNP calling.

You can find guidance on how to interpret the output of each module here

3 Trimming low quality reads and adapters

TrimGalore! is a wrapper script to automate quality and adapter trimming as well as quality control (User Guide).

When the program is installed, it can be used with

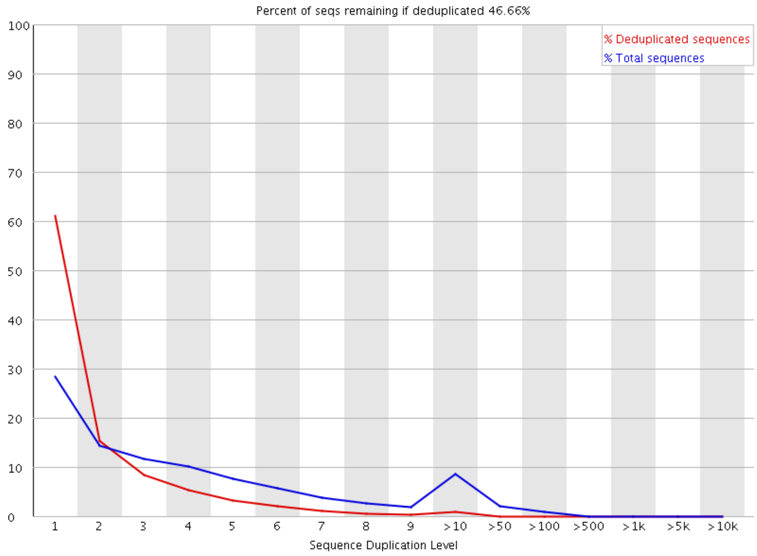


Figure 3: Per base sequence quality (from link)

```
trim_galore [options] <filename(s)>
```

You can get an overview of the options with the `--help` option:

```
trim_galore --help
```

With the default settings, TrimGalore! trims low-quality ends with a Phred quality score threshold of 20 (can be changed with `-q`) and discards reads that become shorter than 20 bp (can be changed with `--length`).

TrimGalore! uses the program Cutadapt to find and remove adapters from the 3' end of the reads (see Fig. 4). The program Cutadapt itself gives you more options for adapter trimming and allows you to remove adapters also from the 5'-end of the sequence (see `http://cutadapt.readthedocs.org/en/latest/guide.html`)

3.1 Trimmiing Ion-Torrent adapters

The Ion-P1- and Ion-A-adapters are supposed to be automatically trimmed off on the Ion Server. So, the fastq files with the raw reads should not contain these adapters anymore. Still, it is good to check if there are any adapters left in your library - they can have negative effects on further analyses.

The adapters used for Ion Torrent sequencing are shown in Fig. 5 and their orientation in the libraries is shown in Fig. 6.

.

To trim off the A-adapter, use TrimGalore! with the command:

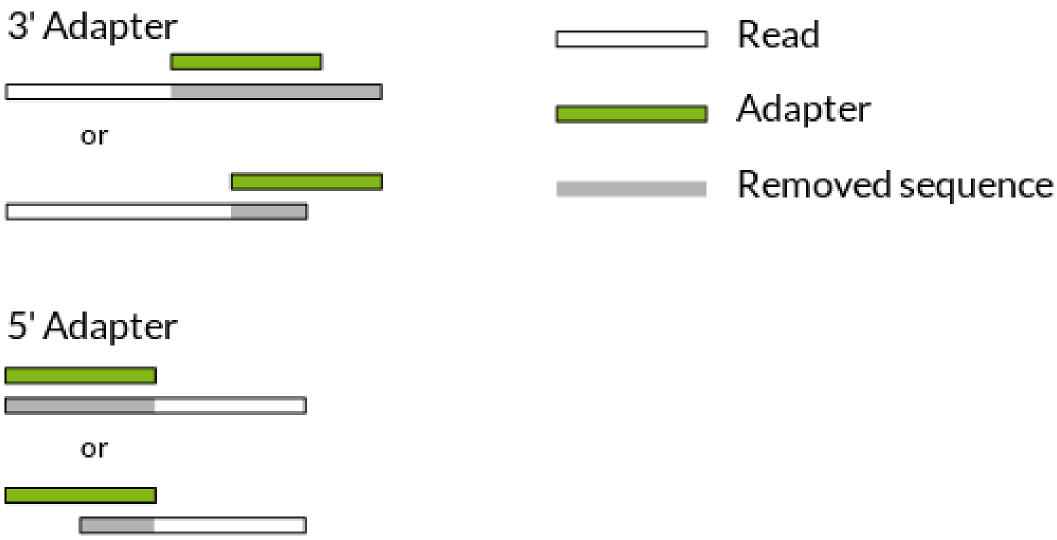
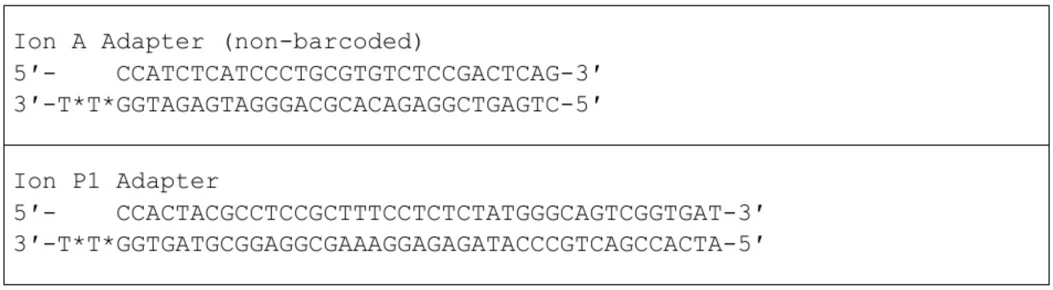


Figure 4: 3'- and 5'-adapter trimming (source)



```
Ion A Adapter (non-barcoded)
5'-    CCATCTCATCCCTGCGTGTCTCCGACTCAG-3'
3'-T*T*GGTAGAGTAGGGACGCACAGAGGCTGAGTC-5'
```

```
Ion P1 Adapter
5'-    CCACTACGCCTCCGCTTTCCTCTCTATGGGCAGTCGGTGAT-3'
3'-T*T*GGTGATGCGGAGGCGAAAGGAGAGATACCCGTCAGCCACTA-5'
```

Figure 5: Non-barcoded Ion-A and -P1 adapter sequences. In each sequence, a "*" indicates a phosphorothioate bond, for protection from nucleases and to preserve the directionality of adapter ligation. This is not relevant for adapter trimming.

```
trim_galore \
-a CCATCTCATCCCTGCGTGTCTCCGACTCAG \
--stringency 3 \
FILE.fastq
```

The \ sign just means that the command continues on the next line. You could type the entire command on a single line.

The option `--stringency 3` means that a >3bp overlap with the adapter sequence will be trimmed off the 3' end. The program writes a file that ends with `trimming_report.txt`, which reports the number of reads that have been trimmed and/or removed.

The output file has the ending `trimmed.fq`. Use this file as input to TrimGalore! to trim off the P1-adapter:

```
trim_galore \
-a CCACTACGCCTCCGCTTTCCTCTCTATGGGCAGTCGGTGAT \
--stringency 3 \
--fastqc FILE_trimmed.fq
```

The `--fastqc` option will automatically run FastQC in the default mode. Compare the FastQC

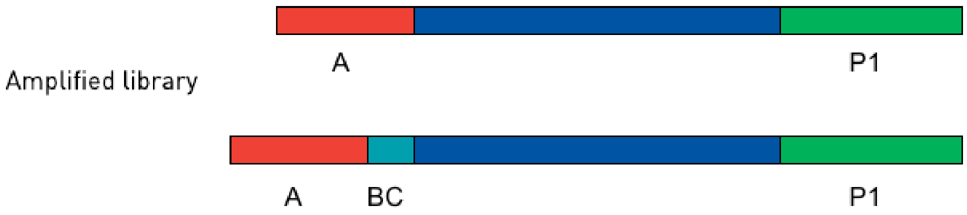


Figure 6: Ion adapters in the amplified library. BC is an optional barcode sequence.

outputs before and after trimming.

3.2 Trimming Illumina adapters

Depending on the settings for Illumina sequencing, the adapters can be automatically removed from the fastq files that you get from the sequencing machine. This, however, has to be defined before sequencing. If you are not sure whether adapters have been trimmed off or not, it is safe to trim the adapters before using the sequences for any further analyses.

The Illumina adapters are as follows:

```
TruSeq Universal Adapter:
5' AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCTTCCGATCT 3'

TruSeq Indexed Adapter
5' P*GATCGGAAGAGCACACGTCTGAACTCCAGTCACNNNNNNATCTCGTATGCCGTCTTCTGCTTG 3'
```

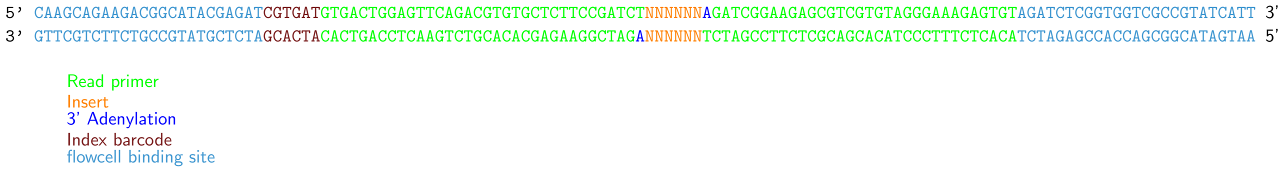
Here, `NNNNNN` represents a barcode of six nucleotides in the indexed adapter.

TrimGalore! can be run with the option `--illumina`. This trims the first 13bp of the Illumina universal adapter `AGATCGGAAGAGC`. This option removes illumina adapters from most standard libraries, including TruSeq adapters.

The location of this sequence in the TruSeq adapter is shown here:

```
TruSeq Universal Adapter:
5' AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCTTCCGATCT 3'
   Reverse                                      CGAGAAGGCTAGA
TruSeq Indexed Adapter
5' P*GATCGGAAGAGCACACGTCTGAACTCCAGTCACNNNNNNATCTCGTATGCCGTCTTCTGCTTG 3'
    AGATCGGAAGAGC
```

The A on the 5'-end of the TruSeq indexed adapter is added during A-tailing of your DNA library fragments. The orientation of the adapters in the illumina library are shown in Fig. 7.



```
5’ CAAGCAGAAGACGGCATACGAGATCGTGATGTGACTGGAGTTCAGACGTGTGCTCTTCCGATCTNNNNNNAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT 3’
3’ GTTCGTCTTCTGCCGTATGCTCTAGCACTACACTGACCTCAAGTCTGCACACGAGAAGGCTAGANNNNNNTCTAGCCTTCTCGCAGCACATCCCTTTCTCACATCTAGAGCCACCAGCGGCATAGTAA 5’
```

Figure 7: Orientation of the illumina adapters around the DNA inserts

TrimGalore! also performs trimming of paired-end libraries, as the Illumina libraries that were prepared in this course. This allows to discard too short read pairs without disturbing the sequence order of FastQ files, which is required by many a ligners. With the option `--paired` TrimGalore! expects two paired fastq input files, like `file1_1.fq` and `file1_2.fq`. Here, both sequences of a sequence pair must have a certain minimum length (specified by the `--length` option in order to be kept. If only one of the two paired end reads became too short, tThe option `--retain_unpaired` can be applied to write the unpaired read that is long enough to eithe `unpaired_1.fq` or `unpaired_2.fq` The length cutoff for unpaired si ngle end reads is governed by the parameters `--length_1` and `--length_2`

To trim our illumina paired-end libraries we can use:

```
trim_galore \
--illumina \
--stringency 3 \
--paired \
--retain_unpaired \
--length_1 20 \
--length_2 20 \
--fastqc \
Illumina_R1.fastq \
Illumina_R2.fastq
```

This command runs automatically FastQC on the trimmed libraries. You can now compare the quality of your raw libraries and your quality-trimmed libraries. What did improve? Are there still any problems with your libraries after trimming?

Emacs 24.3.1 (Org mode 8.3beta)