Chapter 9

Raw Sequence Data and Quality Control

Giovanni Bacci

Abstract

Next-generation sequencing technologies are extensively used in many fields of biology. One of the problems, related to the utilization of this kind of data, is the analysis of raw sequence quality and removal (trimming) of low-quality segments while retaining sufficient information for subsequent analyses. Here, we present a series of methods useful for converting and for refinishing one or more sequence files. One of the methods proposed, based on dynamic trimming, as implemented in the software StreamingTrim allows a fast and accurate trimming of sequence files, with low memory requirement.

Key words Next-generation sequencing, DNA sequence, Trimming, FASTQ, FASTA, QUAL, Base-calling

1 Introduction

DNA sequencing is the process of determining the order of the nucleotides that composed a DNA molecule. Knowledge of DNA sequences is becoming indispensable for a great number of biological fields such as diagnostic, biotechnology, forensic biology, systems biology, and evolutionary biology [1]. The increasing speed of sequencing reached with modern DNA sequencing technology has been crucial in the sequencing of longer and longer complete DNA sequences. In recent years this process has led to the sequencing of entire genomes of numerous types and species of life such as human genome [2], plant genomes, and complete genomes of several microbial species.

When we speak about DNA sequences, normally we refer to "already processed" sequences present in a dedicated database such as NCBI or EMBL. However, we have to know that the first type of sequence produced by "next-generation sequencing" machine is the so-called flowgram or chromatogram. These sequence types are represented by a series of peaks along time where each peak is the signal intensity and the time is the order of

Alessio Mengoni et al. (eds.), *Bacterial Pangenomics: Methods and Protocols*, Methods in Molecular Biology, vol. 1231, DOI 10.1007/978-1-4939-1720-4_9,

the bases within the DNA sequence. As a consequence, if we want to transform a chromatogram or a flowgram into a simple DNA sequence (in other words a series of bases) there are several steps that we have to perform.

First of all, we have to use a "base calling algorithm" in order to assign a nucleotide to each peak present in the raw file. The most common "base calling algorithm" is Phred [3]; in fact the quality of each nucleotide inside a DNA sequence is commonly expressed as "Phred quality score". Phred's algorithm uses a probabilistic based quality score estimated using the per-base error probabilities. The quality score, Q, assigned to a base is proportional to its error probability, P, and is calculated using this formula:

$$Q = -10 \log_{10} P$$

Accordingly, a Phred quality score of 30 corresponds to an error probability of 0.1 %. There are also other base caller algorithms as TraceTuner (http://sourceforge.net/projects/tracetuner/) or LifeTrace [4] but, for the purpose of this chapter, their differences are very small and we have no specific recommendations from the ones here described.

After the base calling step, two different files are generated: one file containing the sequence data (the nucleotide sequence, normally in FASTA format) and the other file containing a series of quality scores separated by a white space. This file format is called QUAL file and is one of the standard file formats used by bioinformaticians [5]. However, this is not the only file format used for storing nucleotide data and quality data. In fact, a different file format able to store a numeric quality score associated with each nucleotide in a sequence is commonly used and is becoming the de facto standard for storing the output of high-throughput sequencing instruments. This format is called the FASTQ format; no doubt because of its simplicity, the FASTQ format has become widely used as a simple interchange file format. Unfortunately the FASTQ format suffers from the absence of a clear definition bringing to light some incompatibilities between its different encodings.

Normally, a FASTQ file uses four different lines to store a DNA sequence with its quality. The first line contains the id of the sequence and is preceded by a "@" character followed by the sequence identifier. The second line contains the DNA sequence itself as a repetition of four characters, one per each nucleotide ("A" for adenine, "C" for cytosine, "T" for thymine, and "G" for guanine). The third line starts with a "+" character that may be followed by a repetition of the sequence id (the same contained in the first line) or not. Finally, the fourth line contains the quality values, and must contain the same number of symbols as letters in the sequence. Here is an example of a FASTQ sequence as reported in [5]:

```
@SRR014849.1 EIXKN4201CFU84 length=93
GGGGGGGGGGGGGGGGCTTTTTTTGTTTGGAACCGAA
   AGGGTTTTGAATTTCAAACCCTTTTCGGTTTCCAAC
   CTTCCAAAGCAATGCCAATA
+SRR014849.1 EIXKN4201CFU84 length=93
3+&$#""""""""""""7F@71,'";C?,B;?6B;:EA1EA1EA5'9
   B:?:#9EA0D@2EA5':>5?:%A;A8A;?9B;D@/=<?7=9
   <2A8==
```

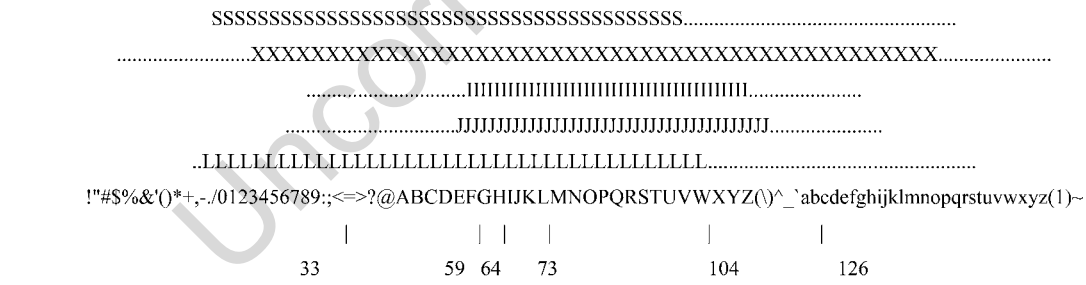
@title and optional description

sequence line(s)

+optional repeat of title line

quality line(s)

The quality line is encoded using a simple ASCII character for each base of the sequence. In fact, each ASCII character can be represented as an integer ranging from 0 to 128 [6]. However, the quality score of a nucleotide can range from 0 to 40 Phred (41 for the most recent Illumina machines). In order to scale ASCII values according to Phred quality scores we have to subtract an offset from each ASCII value and use only a portion of the ASCII scale. Unfortunately, there are more than one different encoding for FASTQ quality format but the conversion between these different standards is very straightforward. Thus, all we have to do is to subtract to the ASCII value an offset specified by the FASTQ encoding. The offsets most commonly used are 33 and 64 as reported in the Wikipedia FASTQ file format page [7]:



S - Sanger Phred+33, raw reads typically (0, 40)

X - Solexa Solexa+64, raw reads typically (-5, 40)

I - Illumina 1.3+ Phred+64, raw reads typically (0, 40)

J - Illumina 1.5+ Phred+64, raw reads typically (3, 40)

L - Illumina 1.8+ Phred+33, raw reads typically (0, 41)

Despite all this different encoding formats and the absence of a clear definition, the FASTQ file format has a great advantage in respect to the FASTA + QUAL file format: it uses only 1 character

for the encoding of a nucleotide quality instead of 2 or 3 character (1 or 2 for the quality and 1 for the withe space) used by the QUAL file. In fact, if we consider that a simple character uses 1 byte to store its value, a FASTQ sequence of 1,000 nucleotides will use about 2,000 bytes of space while a FASTA + QUAL sequence of the same length will use from 3,000 to 4,000 bytes. In addition, if we consider that DNA sequencing cost is decreasing year by year at the same speed that DNA sequencing data is increasing in size, using a "more compressed" file format to store DNA sequences and their quality values is certainly a better choice.

When all the steps described above have been completed, it is time for the central steps of this chapter: the quality control step. One of the most important problems related to the production and utilization of DNA sequence reads is the analysis of base quality and removal (trimming) of low-quality segments while retaining sufficient information for subsequent analyses [8]. Several trimming algorithms and software programs have been developed to cope with the cleanup of DNA sequence reads, e.g., SolexaQA DynamicTrim [9], FASTX-ToolKit (http://hannonlab.cshl.edu/fastx_toolkit), ConDeTri [10], and NGS QC Toolkit [11]. However, all these software were developed in order to be used by expert bioinformaticians; in fact they have not been equipped with a graphical user interface and the setting of their parameters has to be hand made by the user.

To overcome this limitation imposed by the existing trimming software programs, we have developed StreamingTrim [12] using standard Java language and BioJava libraries [13] (included in the package). This software uses a very flexible "dynamic window" algorithm to remove low-quality segments of DNA sequences, beginning from the end of each read in a sequence file. This approach is very useful because it allows users to set a more stringent quality cutoff, which increases the read quality and reduces the risk of losing too much information. In addition, due to its graphical user interface, StreamingTrim can be simply installed and launched, allowing the software to be used even by inexperienced bioinformaticians, easily permitting "wet lab" molecular ecologists to analyze their data.

In Fig. 1 we report a comparison of StreamingTrim and other four commonly used trimming software (SolexaQA DynamicTrim, ConDeTri, NGS QC Toolkit, and Mothur [14]). In order to compare the number of removed bases and the quality increment in two sample datasets using a single metric, we introduced a trimming performance estimator, called *Z*-score. This estimator is proportional to the ratio between the increase in quality and the decrease in the number of bases for each dataset. The *Z*-score was calculated as follows:

$$Z_{\text{score}} = \log_{10}\left(\frac{Q_{\text{diff}}}{|L_{\text{diff}}|}\right)$$

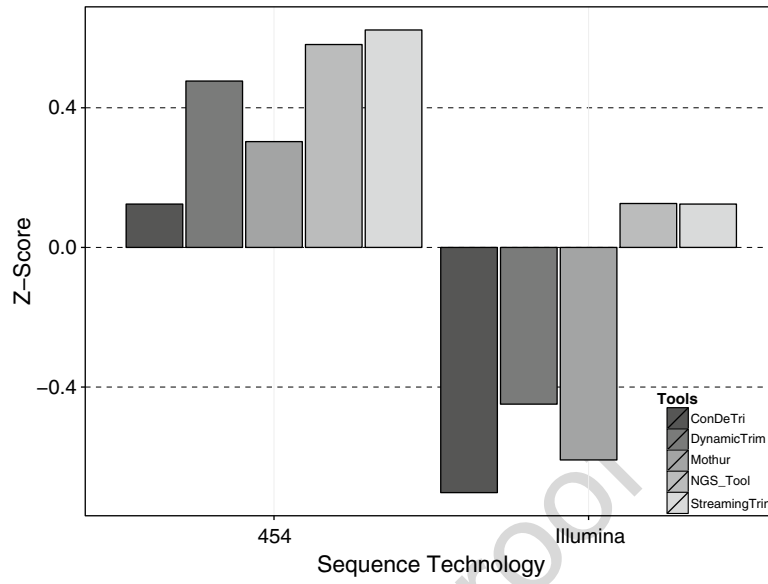


Fig. 1 *Z*-score of different trimming software programs. Bar charts of the *Z*-score after executing the trimming on two datasets (Illumina and 454) are shown. *Negative values* of the *Z*-score indicate that the percentage of bases lost during the trimming process is higher than the percentage of increase in quality. *Positive values* of the *Z*-score indicate that the quality increase is higher than the percentage of bases lost

where:

$$Q_{\text{diff}} = \frac{\left(Q_f - Q_i\right)}{\left(Q_{max} - Q_i\right)} \quad \text{and} \quad L_{\text{diff}} = \frac{\left(L_f - L_i\right)}{\left(L_{min} - L_i\right)}$$

with:

Q_i = initial average quality; L_i = initial number of bases

Q_f = final average quality; L_f = final number of bases

L_{min} = minimum final number of bases
(if users do not specify the minimum length parameter, this value is set to 0)

Q_{max} = maximum final quality
(for Phred score this parameter is set to 40)

The results obtained with all tested trimming tools considered on the 454 and Illumina datasets showed that StreamingTrim had the highest *Z*-score values (Fig. 1), indicating the presence of a good compromise between base conservation and increase in read quality.

1.1 Note to This Chapter

As you may have noticed in this manual we use some type-setting conventions. We use:

this format

in order to refer to command line input or output, but also to refer to external text (for example a DNA sequence contained in a sequence file); when we want to indicate a program menu or function we use <this format>. If you see something like <File → Open File> it means that we refer to the Open File item in the File menu.

2 Materials

All software used in this chapter can be downloaded for free. StreamingTrim is distributed under the BSD-2-Clause license; if you want to learn more about this kind of license visit the page http://opensource.org/licenses/BSD-2-Clause. Since StreamingTrim keeps in memory only one sequence at a time, it can be used even with a standard desktop PC or a laptop. However we recommend having at least 1 or 2 Gigabytes free for each 500 Megabytes of raw data. In this chapter we assume that you have your sequences in FASTQ file format; however, if it is not your case, here we report a two-step procedure in order to convert your chromatogram files into FASTQ file. If you have your sequences already in FASTQ file format you can ignore the two subheadings described below.

2.1 Obtaining Sequence Data from Chromatograms

In order to generate a sequence file you have to perform at least one base calling step as described in Subheading 1.

1. Download and install Phred from http://www.phrap.org/phredphrapconsed.html.
2. Run Phred on your raw sequence file. Here is an example using the standard Phred analysis:

```
phred -id chromat_dir -sa seqs_fasta -qa seqs_fasta.qual
```

Running this line will convert all chromatogram files present in the chromate_dir directory into two files: a FASTA file called seqs_fasta and a QUAL file called seqs_fasta.qual.

2.2 Converting the FASTA + QUAL Files into One FASTQ File

There are many tools able to encode a FASTQ file starting from a FASTA file and a QUAL file; here we report only one script developed by the Bio-Linux community [15] (http://nebc.nerc.ac.uk/) in order to be as simple as possible.

1. Download and install Phyton from http://www.python.org/download/.
2. Download and install Biophyton from http://biopython.org/wiki/Download.

3. Download the script called fasta_to_fastq.py from the Bio-Linux community: http://nebc.nerc.ac.uk/tools/code-corner/scripts/sequence-formatting-and-other-text-manipulation.
4. Run the script as described below:

   fasta_to_fastq.py input.fna

   The script does not care if you use a different FASTA extension but there must be a file named input.qual containing the phred quality scores; otherwise the FASTQ file will not be generated.

2.3 Downloading StreamingTrim

StreamingTrim is a software built using Java 1.7, so you have to ensure that you have at least Java 1.7.0 version installed on your system. In order to do this you have to open your command windows (cmd.exe in Windows systems and terminal in OS systems) and type this:

java –version

If you receive an error message it means that you do not have Java installed on your system. Otherwise, if you receive a message like this one:

java version "1.7.0_09"
OpenJDK Runtime Environment (IcedTea7 2.3.4)
OpenJDK 64-Bit Server VM (build 23.2-b09, mixed mode)

If the number between brackets is smaller than 1.7.0 it means that you have Java installed on your system but you have an old version of the software. In both cases you have to install an up-to-date *Java Runtime Environment*; you can download it from the oracle website: http://www.java.com/en/download/ (if you have an old version of Java it is recommended that you uninstall it before installing the new version). Otherwise, if your Java version is up to date you can proceed to download the software from the GitHub repository at https://github.com/GiBacci/StreamingTrim and save it in a folder of your choice.

2.4 Running StreamingTrim for the First Time

Once you have downloaded the software you can launch it by double clicking one of the two launchers present in the software's folder. If you have a Microsoft Windows-based system you have to use the windowsLauncher.bat file, while if you have a Linux-based system or a Mac OS-based system, you can launch it with the unixLauncher.sh file (remember to allow executing file as an application). If everything has gone well you would be able to see the main window of StreamingTrim software. Now you are able to analyze your FASTQ files and trim them using this trimmer.

2.5 StreamingTrim Workflows

StreamingTrim algorithm workflows and example steps are reported in Fig. 2. Given a DNA sequence of length N, the algorithm starts

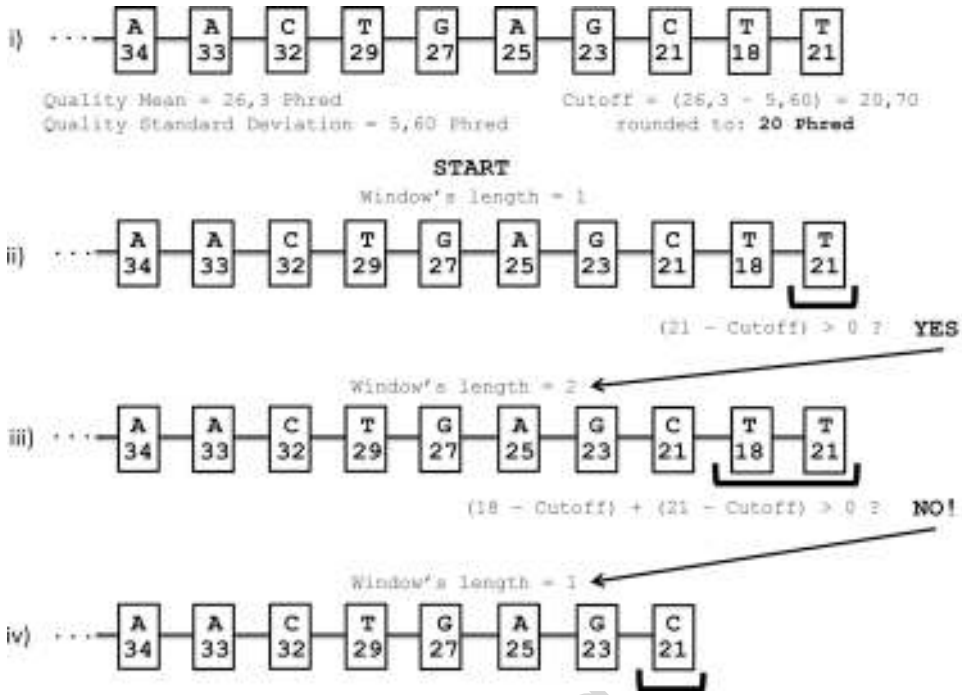


Fig. 2 Workflow of the StreamingTrim algorithm. First (1), a sample sequence is selected from a sequence file with a mean quality of 26,30 Phred and a quality standard deviation (SD) of 5,60. Then (1), a quality cutoff is calculated by subtracting one SD from the quality mean. Next (2), the last base of the sequence is analyzed by subtracting the previously obtained cutoff from its quality value. If this result is bigger than 0, the base is maintained and (3) the analysis window is increased by one. Now, the quality of each base is analyzed as in step (2) and the results are summed up. In the displayed example, the result is less than 0 and, consequently (4), the two bases are removed from the sequence and the size of the analysis window is set again to 1. All these steps are repeated until the sequence has been entirely analyzed

from the last nucleotide (the n^{th} nucleotide), using a window length (W) of 1 and checks if:

$$\left(\text{Quality}_{n^{th}} - \text{cutoff}\right) \geq 0$$

If this is true, the algorithm will proceed by enlarging the window length by 1 (in this case putting $W=2$); otherwise the n^{th} nucleotide is removed. N is then decreased by the number of removed nucleotides (in this case 1) and W is set to 1. This process is repeated until the algorithm reaches the first nucleotide of the DNA sequence ($N=1$), or if the trimmed sequence length goes below a minimum value previously chosen by the user (default 1). A formal description of the algorithm is shown here:

$$N = \text{sequence length};\ W = \text{window length};\ M = \left(N - W\right)$$

$$T = \sum_{M<k\leq N} \left(\text{Nucl}_k - \text{cutoff}\right)$$

$$\text{If } T \geq 0 \rightarrow (W+1); \text{If } T < 0 \rightarrow N = (N-W) \,\&\, W = 1$$

Continue with the test T until $(N - W) \leq 0$ or $N <$ minimum length.

The above reported algorithm has been developed in order to be as conservative as possible. In fact, a DNA segment is deleted only if all its nucleotides are considered to be of low quality. If there are only a few low-quality bases in a sequence, the segment is maintained in order to prevent loss of information.

3 Operating Procedure

Here we describe the crucial steps to perform in order to check the quality of a sequence file.

3.1 Analyzing the Reads

In order to prepare the trimmer for the quality refinement, it is better to perform at least one quality control step.

3.1.1 Open a FASTQ File

To open a sequences file in the program the user can click on <File → Open File> in the main window of the program or type the "Ctrl + o" shortcut on his or her keyboard. After that, the file open windows will appear on the screen and the user can select the file to open. Unfortunately, the FASTQ file format does not have a well-defined set of extensions; .fastq, .fq, and .txt are the most used. If the user has a FASTQ file with another extension he or she must select the "All file" option in the extension menu in the <File Open> windows and then select the right file to open; otherwise he or she will not be able to see and select his or her file. After delecting the file and pressing the <Open File> button the <Input File> section in the main windows will fill with the path to the selected file.

3.1.2 Analyzing the File

After a sequence file is successfully opened the user can analyze it in order to see the quality and length distribution of the DNA sequences present in the file. If the user has not opened a file yet, when he or she presses the <Analyze> button, an <Open File> window will appear and he or she can select the interested file from here.

In order to analyze the file the user has to press the <Analyze> button in the <Controls> section of the software main window. When the user presses the button, the <Progress Bar> will begin to move and the file will be analyzed. After that, the <Reads Properties> window will display all the statistics related to the file. If the user wants a more accurate description of quality and length distribution, he or she can press the <Plot> button in the <Controls> section of the main window; <Plot Window> opens and the software begins to deeply analyze all the sequences in the file. When the program has finished analyzing data a plot will appear in the <Raw data> section of the <Plot Window>.

Two different kinds of plot can be displayed in the <Plot Window>:

1. <Deviation Plot> is a representation of the DNA base quality distribution along each sequence. In the *x*-axis the length of the sequences is reported. If there are sequences with different lengths, then the length of this axis is the length of the longest sequence. In the *y*-axis the quality values from 0 to 40 are reported. The mean quality is represented as a bold line while the range between maximum quality value and minimum quality value is represented as a blue surface. In this way the user can see the distribution of every base quality, and not only the mean or the standard deviation.
2. <Box Plot>: This is a standard box plot representation of the quality distribution for each sequence in the sequence file. If you have reads longer than 200 nucleotides, this type of visualization can be very difficult to read; otherwise if you have short reads (about 100–150 nt) this plot can be very useful since also the median and the first and third quartile (as a normal boxplot) are reported.

There is also another kind of plot that can be displayed in the <Plot Window>, the so-called length plot. This plot gives the user a bar chart representation of the read length distribution. Here, only one type of plot is possible, where in the *x*-axis the sequence length values (they can change by changing the input file) are reported and in the *y*-axis the number of reads in the file that has the corresponding length value is shown.

The user can zoom anywhere in the plot, by simple clicking and dragging with the mouse the part of the plot that he or she wants to zoom. In the bottom of the plot there is the number of reads that are found in the plotted file.

The user can now save the chosen plots by simply right clicking them and choosing the "Save as" option in the pop-up menu.

3.2 Parameter Settings

In the <Advanced Option> window (accessed through <Window → Show advanced option>) the user can specify some trimming parameters in order to adjust the trimming process to his or her will. Here, all the advanced options are described in order to understand the complete StreamingTrim functionality.

3.2.1 Cutoff

This parameter represents the quality cutoff to be used by the software during the trimming process. Typically, the quality range of a FASTQ sequence file goes from 0 to 40, representing hypothetical error probabilities of 100 % and 0.01 %, respectively. If this parameter is not selected, the trimmer chooses a cutoff automatically based on the mean quality and the standard deviation of the reads in the given file (e.g., if we have a file with a mean quality of 31.46 and a standard deviation of 6.54, the quality cutoff is set to 31.46–6.54 = 24.92 and approximated up to 25).

The user can change this parameter in order to perform a more or less stringent quality refinement by using higher or lower cutoff values, respectively.

3.2.2 Offset

This parameter indicates the number of bases to eliminate at the beginning of every reads. Setting a value higher than 0 is useful when the presence of adapters or some unwanted region at the beginning of each sequence is known. Otherwise it is recommended to leave this parameter unchecked.

3.2.3 Minimum Length

With this parameter the user can specify a length cutoff (in bases). Sequences that, after the trimming process, have a length lower than this parameter are not saved in the output file. This parameter is very useful in amplicon-based analysis, where reads that result too short after trimming are useless for the following analyses (e.g., taxonomic identification).

3.2.4 General Considerations

It is recommended to choose this set of parameter based on the previously done analysis of the sequence quality. In fact, for example, choosing a cutoff parameter too small in a very-poor-quality sequence file could lead to inconclusive results. On the other hand, choosing a too high value of cutoff for a very-poor-quality FASTQ file could generate a file with too few sequences. If the user is not sure about the setting of these parameters, the better choice is to let everything unchecked.

3.3 Trimming

The principal function of StreamingTrim is to cut low-quality bases from each sequence in a DNA sequence file. First of all, in order to start the trimming process, the user has to open a valid input file as described in Subheading 3.1.1. Then, the user can proceed to start the analysis clicking on the <Trim> button in the main window of the StreamingTrim interface. When the <Trim> button is pressed a <Save File> window appears and the user can choose the destination and the name of the file containing the trimmed reads. After that the <Progress Bar> begins to move and the trimming process starts using the default trimming parameters or the user-defined parameters (if previously specified, *see* Subheading 3.2).

When the trimming process reaches the end an output file will be saved as previously specified by the user. The output file will be in the same format as the input file and will use the same FASTQ offset (*see* Subheading 1).

3.3.1 The <Trim to FASTA> Function

StreamingTrim can convert a trimmed file into FASTA format while the trimming process goes on. If the checkbox <Trim to FASTA> in the main window is selected, when the user starts the trimming process the software simultaneously converts the output file to FASTA format. When the checkbox is selected from the user, a <Save FASTA file> window opens and the user can choose the directory and the file name he or she prefers.

This function is very useful if there is a need to trim more than one file with the same parameters, without analyzing them each time. In this way the trimming and conversion processes are speeded up.

3.3.2 Controlling Results

Results obtained after the trimming process can be analyzed as described in Subheading 3.1. In the plot window the user can compare the two graphic representations of the sequence file before and after the trimming process. This can be useful in order to check if the result obtained with the set of parameters chosen is satisfactory or not.

If the average quality of the trimmed reads is still too low, the user can repeat the trimming process specifying a more stringent cutoff value. It is recommended to trim the original file again in order to be as much reproducible as possible. If the user attempts to trim an already trimmed file he or she will not be able to repeat the same analysis unless he or she does not perform again the two trimming processes with exactly the same parameters. On the other hand, if the user chooses to trim the original file he or she will be able to reach the same results with only one step.

3.4 Converting Raw Sequencing Data

When the quality refinement step has reached a satisfactory conclusion, it is recommended to convert the raw sequence file (in this case in FASTQ format) into a more suitable sequence format. The most used file format for DNA sequences is the FASTA file format. StreamingTrim can convert FASTQ file into FASTA after the end of the trimming process or even in the same time (as seen in Subheading 3.3.1). If the user wants to convert the refined FASTQ file all he or she has to do is to click the <FASTA> button in the main window of the program. Doing this will cause the <Progress Bar> to start moving and a FASTA file will be created.

References

1. Pettersson E, Lundeberg J, Ahmadian A (2009) Generations of sequencing technologies. Genomics 93:105–111
2. Sawicki MP, Samara G, Hurwitz M, Passaro E (1993) Human genome project. Am J Surg 165:258–264
3. Ewing B, Hillier L, Wendl MC, Green P (1998) Base-calling of automated sequencer traces using Phred. I. Accuracy assessment. Genome Res 8:175–185
4. Walther D, Bartha G, Morris M (2001) Base calling with lifetrace. Genome Res 11: 875–888
5. Cock PJ, Fields CJ, Goto N, Heuer ML, Rice PM (2010) The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. Nucleic Acids Res 38:1767–1771
6. Wikipedia (2014) ASCII. Wikipedia, the free encyclopedia
7. Wikipedia (2014) FASTQ format. Wikipedia, the free encyclopedia
8. Kunin V, Copeland A, Lapidus A, Mavromatis K, Hugenholtz P (2008) A bioinformatician's guide to metagenomics. Microbiol Mol Biol Rev 72:557–578
9. Cox MP, Peterson DA, Biggs PJ (2010) SolexaQA: at-a-glance quality assessment of Illumina second-generation sequencing data. BMC Bioinformatics 11:485

10. Smeds L, Künstner A (2011) ConDeTri-a content dependent read trimmer for Illumina data. PLoS One 6:e26314
11. Patel RK, Jain M (2012) NGS QC Toolkit: a toolkit for quality control of next generation sequencing data. PLoS One 7:e30619
12. Bacci G, Bazzicalupo M, Benedetti A, Mengoni A (2014) StreamingTrim 1.0: a Java software for dynamic trimming of 16S rRNA sequence data from metagenetic studies. Mol Ecol Resour 14:426–434
13. Holland RC, Down TA, Pocock M, Prlić A, Huen D, James K, Foisy S, Dräger A, Yates A, Heuer M (2008) BioJava: an open-source framework for bioinformatics. Bioinformatics 24:2096–2097
14. Schloss PD, Westcott SL, Ryabin T, Hall JR, Hartmann M, Hollister EB, Lesniewski RA, Oakley BB, Parks DH, Robinson CJ (2009) Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. Appl Environ Microbiol 75: 7537–7541
15. Field D, Tiwari B, Booth T, Houten S, Swan D, Bertrand N, Thurston M (2006) Open software for biologists: from famine to feast. Nat Biotechnol 24:801–804