**MCB 525, Fall 2016**

**2.1: Read filtering and trimming**

After completing Section 2.0, you should have two files read files that end in.fastq.gz, in separate directories. In this section, we will perform trimming and quality filtering of these reads to exclude low quality sequences.

**Objectives:**

* Gain familiarity with the fastq file format
* Interpret read quality scores
* Process and trim reads to remove low quality sequence

**Protocol:**

1. Navigate to your CC7 directory. The files you have copied there are in a compressed format, indicated by the .gz file extenstion. Uncompress the read file for CC7:

$ SGE\_Batch –c ‘gunzip CC7\_reads.fastq.gz’ –r emilyb

1. Let’s take a look at the unzipped file. We can do this using the head command. The –n option indicates the number of lines we would like to see:

$ head –n 12 CC7\_reads.fastq

Notice the structure of the fastq format. Each ‘read’ is associated with 4 lines in the fastq file:

@NS500451:199:HF7H7BGXY:1:11101:23824:1094 1:N:0:CATTTT+GTTCC

GGAGCAAGGCTTCGAAGTTCATGCAATCTTGAATAA

+

AAAAAEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE

#The first line identifies the read, the second is the sequence, the third is a ‘+’, and the last line gives the Phred quality score for each base

*Based on Illumina 1.8+ encoding, what is the probability of calling a wrong base given a quality score of ‘A’ in the fastq file?*

*Hint: To convert the score to a number, there are actually pretty good wiki pages for ‘FastQ format’ and ‘Phred Quality Score’, or see* [*https://sites.duke.edu/sequencingatduke/files/2013/08/fastq\_quality\_encloding1.png*](https://sites.duke.edu/sequencingatduke/files/2013/08/fastq_quality_encloding1.png)

1. How many reads are in this file? We can use grep to search for the string @NS500451. This is present in the header of all the reads from this Illumina run.

$ grep –c ‘@NS500451’ CC7\_reads.fastq

#The –c option tells us to count the number of occurrences. Without the –c, we would print all >1,000,000 lines that matched the string!

#Anytime this accidentally happens, type CTRL+Z to stop printing…

1. How many lines are in this file?

$ wc –l CC7\_reads.fastq

#the –l option asks to report the number of lines in the file. Quick sanity check: this should be 4x the answer to Step 2.1c!

1. Now lets do a comprehensive quality check of the reads. For this, we will use a program called FastQC.

* First, download the read files to your computer using the SSH File Transfer Client.
* Open the FastQC program by double-clicking the icon on your desktop.
* Open the reads file (File > Open).
* You should see a variety of statistics come up on the left panel. For which categories have our sequences failed?
* Click on these categories to investigate further. Illumina sequencing often does not perform well with invariant bases, in which there are only one or two bases at a particular site.

*Given this information, and the way these sequencing libraries were prepared, can you explain the results of the FastQC analysis?*

1. By now, you may have noticed that the total length of the reads in your data file is 38 bp, but the 2bRAD libraries we constructed were only 36 bp. The extra 2 bases are from the adaptors, where the sequencer continued on past the part of the library fragment derived from genomic DNA to sequence the adaptor (these bases likely also have low quality, as you may have seen in the FastQC report). Trim off these last 2 bases using a tool from the FASTX-Toolkit (<http://hannonlab.cshl.edu/fastx_toolkit/>):

$ SGE\_Batch –c ‘fastx\_trimmer -f 1 -l 36 -i CC7\_reads.fastq -o CC7\_reads\_36.fastq -Q 33’ –r george

#the –f indicates the first position to keep, -l the last position to keep, -i the name of the input file,–o the name of the output file, and -Q specifies that our reads use the Phred+33 quality format.

1. Check that the output file still has the same # of reads as the input file, using grep –c as in Step 2.1c.
2. Finally, let’s remove low quality reads from our dataset. We’ll use another tool from the FASTX-Toolkit, fastq\_quality\_filter:

$ SGE\_Batch –c ‘fastq\_quality\_filter -q 26 -p 83 -i CC7\_reads\_36.fastq -o CC7\_cleanreads\_q26\_p83.fastq -Q 33’ –r elaine

#-q is the minimum quality score to keep, -p is the minimum percent of the read that must have –q quality.

Try out the fastq\_quality\_filter tool with a few different values of – q and –p, saving each output with a unique filename.

*How many reads remain compared to the input file for each parameter combination?*

*Which do you think are the best parameters to use?*

1. Repeat the read trimming and filtering steps with the read file for your other sample. Use the same final filtering parameters for both files (-q 26 and –p 85) and remove the extra output files with the rm command:

$ rm CC7\_cleanreads\_q28\_p95.fastq

#to remove test files. Make sure to keep a file for each sample with –q 26 and –p 83!!