QAA_Report

Sophia Soriano

2022-09-07

Part 1 - Read Quality Score Distributions

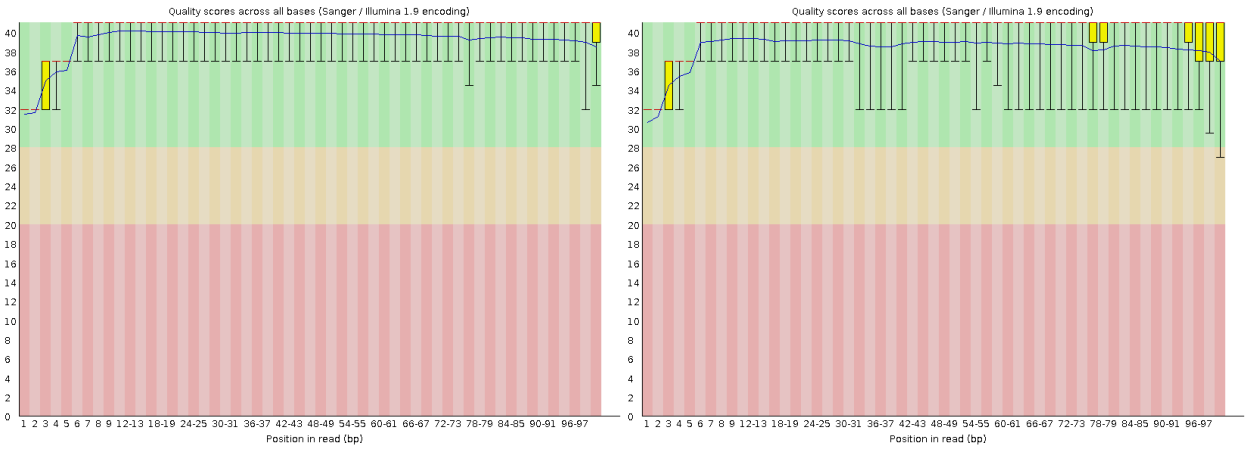

Fig. 1: Library 1 (11_2H_both_S9_L008) FASTQC QScore Distributions for Read 1 (left) and Read 2 (right).

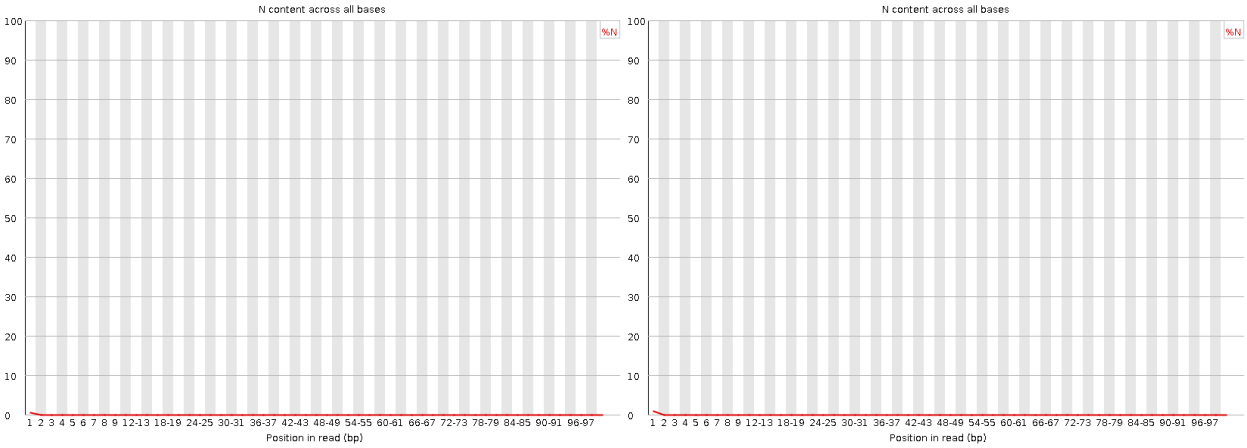

Fig. 2: Library 1 (11_2H_both_S9_L008) FASTQC N-Content Distributions for Read 1 (left) and Read 2 (right).

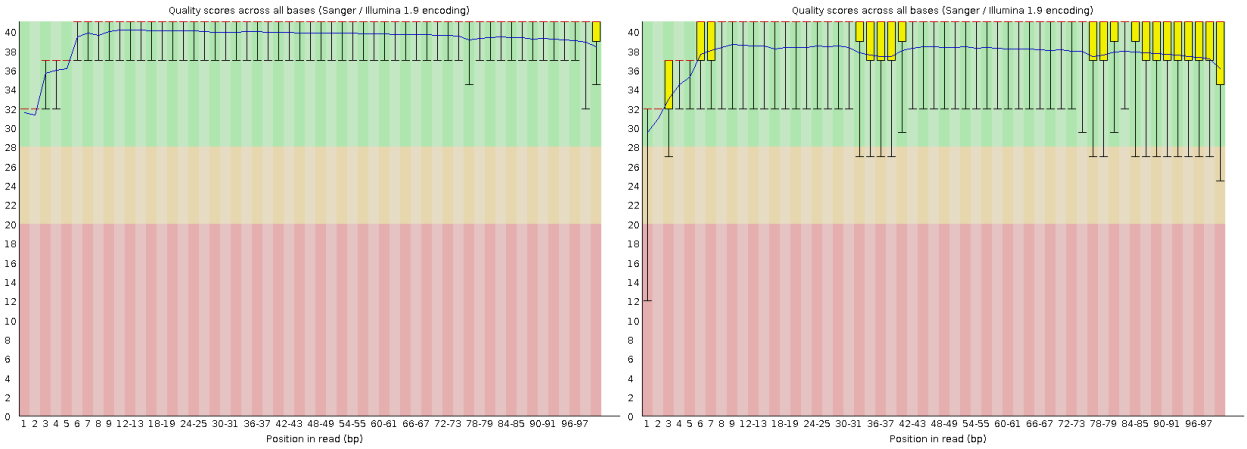

Fig. 3: Library 2 (14_3B_control_S10_L008) FASTQC QScore Distributions for Read 1 (left) and Read 2 (right).

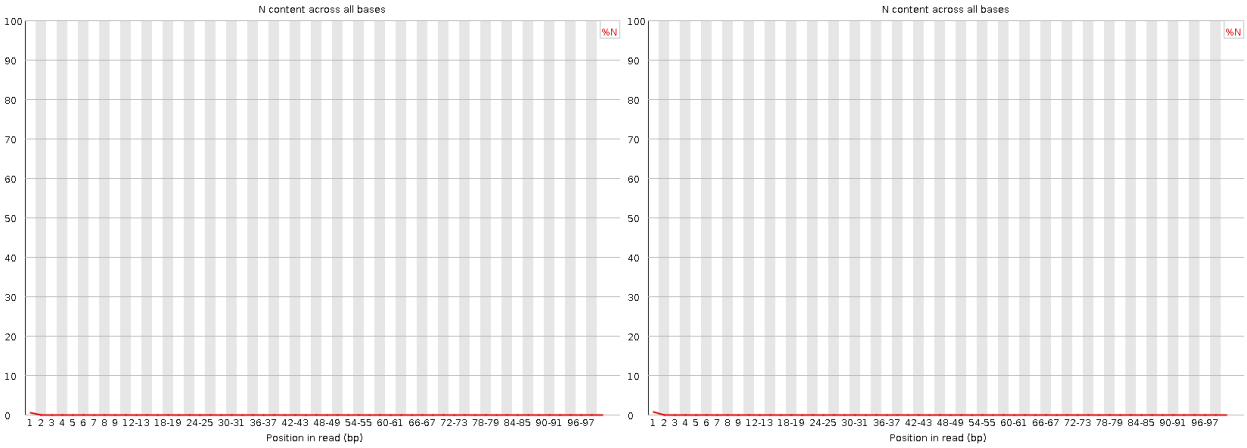

Fig. 4: Library 2 (14_3B_control_S10_L008) FASTQC N-Content Distributions for Read 1 (left) and Read 2 (right).

Summary: The plots of per-base N content are consistent with the quality score plots for each library. The N content is extremely low in both reads for both libraries, and that is reflected in the high (green) values shown in all of the QScore distributions for every data point.

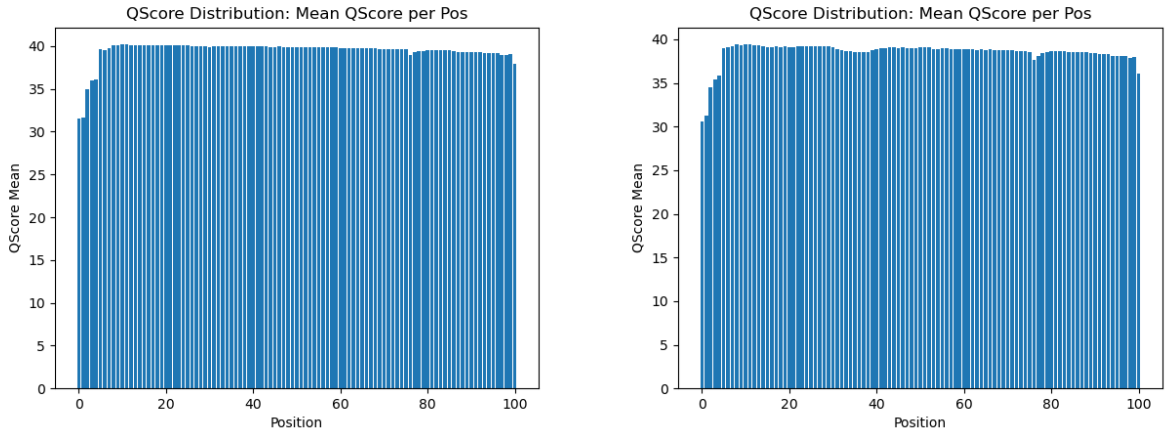

Fig. 5: Library 1 (11_2H_both_S9_L008) QScore_Dist.py QScore Distributions for Read 1 (left) and Read 2 (right).

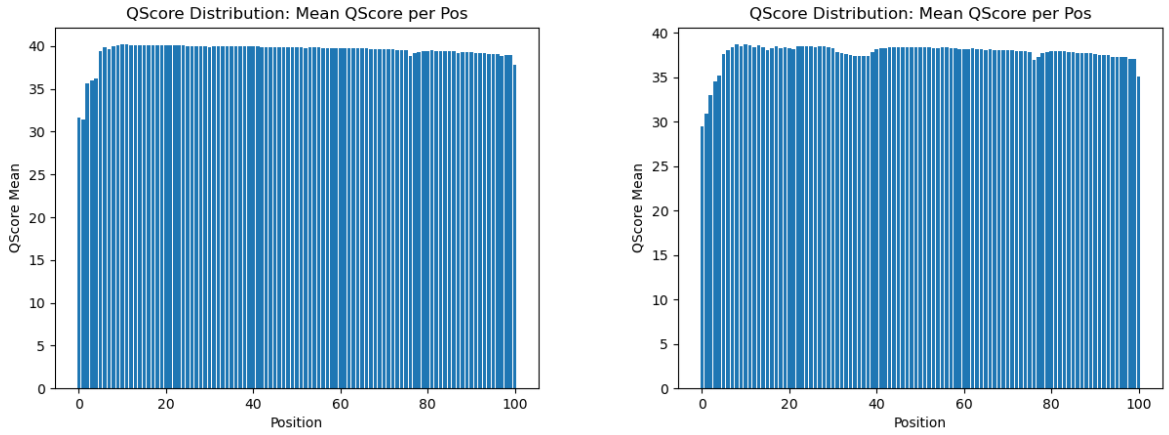

Fig. 6: Library 2 (14_3B_control_S10_L008) QScore_Dist.py QScore Distributions for Read 1 (left) and Read 2 (right).

Summary: The QScore distributions generated from both FASTQC and my QScore_Dist.py program look virtually identical for reads 1 and 2 from both libraries (with the exception of more colorful formatting from FASTQC). The run-times for library 1 (11_2H_both_S9_L008) were about five times longer than the run times for library 2 (14_3B_control_S10_L008), and the FASTQC commands - running two files at once - ran much quicker than my QScore_Dist.py code running one file at a time. See run time results listed below. From both the FASTQC and QScore_Dist.py data for both libraries, one can see that the quality scores for all bases in both reads are very high, generally >35. The FASTQC graphs - which have a little more detail (error bars, etc.) than my QScore_Dist.py graphs - show that read two for both libraries has a little more variation in base quality score, occasionally dipping into the yellow (<29) base quality region for library 2. However, this is expected as the sample has been in the sequencer for a longer amount of time by the time read 2 is in process, and has possibly degraded. Also expected is the section of slightly lower quality scores at the beginning of each read (for both libraries), as the first several bases analyzed by the sequencer are typically error-prone.

Library 1 (11_2H_both_S9_L008): 2:32.58 m:s (FASTQC - both reads), 14:15.86 m:s (QScore_Dist.py, R1) and 14:02.15 m:s (QScore_Dist.py, R2)
Library 2 (14_3B_control_S10_L008): 0:39.34 m:s (FASTQC - both reads), 3:37.55 m:s (QScore_Dist.py, R1) and 3:38.74 m:s (QScore_Dist.py, R2)

Part 2: Adapter trimming comparison

Summary: Used the following bash commands to confirm adapter sequences in each file. "Grep" returns lines that contain the specified sequence:

Lib1 R1: $ zcat /projects/bgmp/shared/2017_sequencing/demultiplexed/11_2H_both_S9_L008_R1_001.fastq.gz | grep "AGATCGGAAGAGCACACGTCTGAACTCCAGTCA"
Lib1 R2: $ zcat /projects/bgmp/shared/2017_sequencing/demultiplexed/11_2H_both_S9_L008_R2_001.fastq.gz | grep "AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT"
Lib2 R1: $ zcat /projects/bgmp/shared/2017_sequencing/demultiplexed/14_3B_control_S10_L008_R1_001.fastq.gz | grep "AGATCGGAAGAGCACACGTCTGAACTCCAGTCA"
Lib2 R2: $ zcat /projects/bgmp/shared/2017_sequencing/demultiplexed/14_3B_control_S10_L008_R2_001.fastq.gz

| grep “AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT”

Adapter sequences expected to be found in each read were confirmed on Illumina website/pdf for TruSeq DNA/RNA preps (https://support-docs.illumina.com/SHARE/AdapterSeq/illumina-adapter-sequences.pdf). The expected sequences - “AGATCGGAAGAGCACACGTCTGAACTCCAGTCA” in R1 and “AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT” in R2 - are found at or near the end of many sequences in each file. These sequences are reverse complements of the adapters containing the read 1/2 PBS where sequencing by synthesis begins (i.e. “AGATCGGAAGAGCACACGTCTGAACTCCAGTCA” is found near the end of R1 sequences, and its reverse complement marks the start of R2 sequence by synthesis). Adapter sequences are not observed at the beginning of sequences because the read PBS (primer binding sites), where the sequencing primers bind to initiate synthesis of the read, are located at the end of the adapter sequence.

For Library 1 (11_2H_both_S9_L008), 4.9% of Read 1 (874,706 reads) and 5.7% (1,016,991 reads) of Read 2 were trimmed (total read pairs = 17,919,193). For Library 2 (14_3B_control_S10_L008), 6.0% of Read 1 (264,208 reads) and 6.7% of Read 2 (299,716 reads) were trimmed (total read pairs = 4,440,378).

As shown in the bar plots below, R2 reads (blue) are significantly more trimmed than R1 reads (red). As previously mentioned, Read 2 is commonly lower quality per base than Read 1 since it occurs after the library sample has likely been in the sequencer for several hours and some degradation has occurred. When we apply our quality filters - especially with Trimmomatic using SLIDINGWINDOW 5(window):15(quality min) and MINLEN (35bp) - there are fewer long reads remaining in Read 2 that meet those cutoffs compared to the higher quality Read 1 sequences.

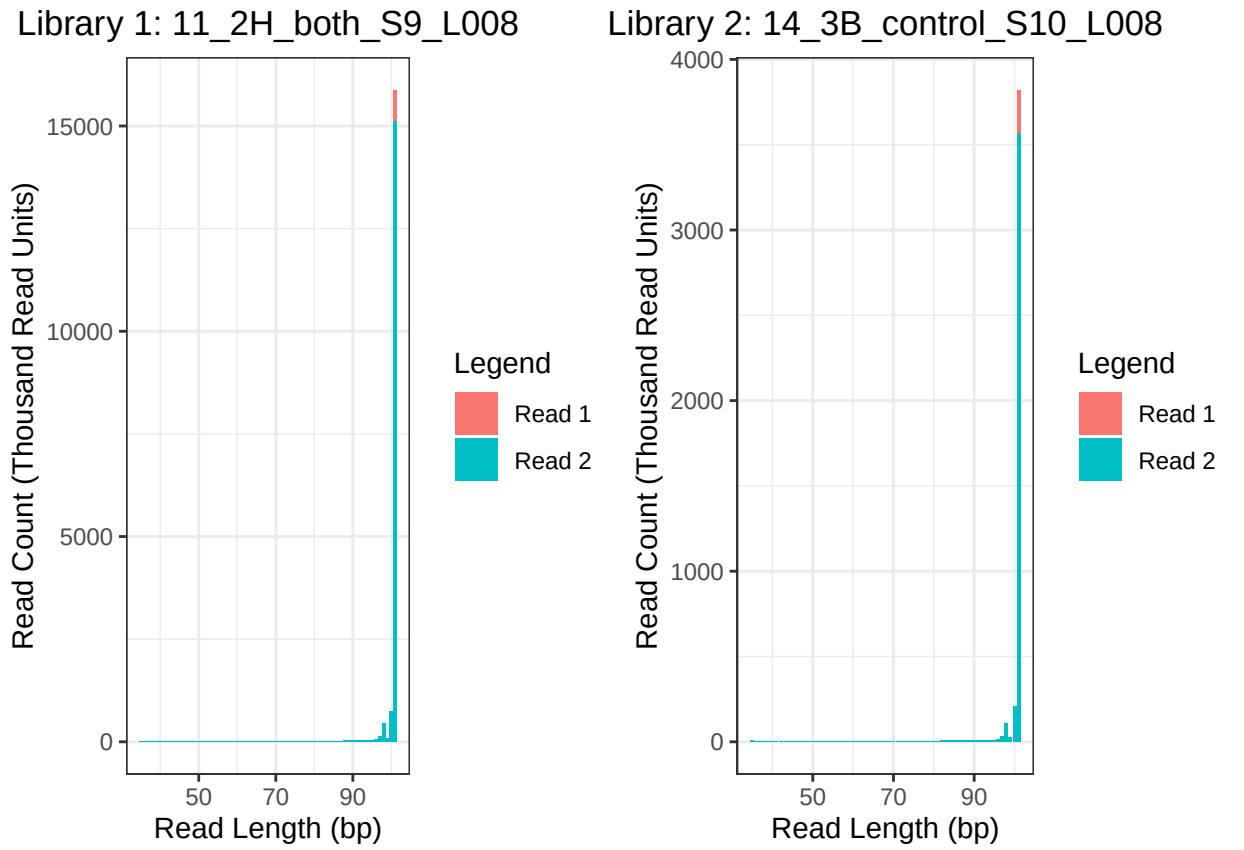

Fig. 7: Read length distributions for Read 1 and Read 2 files for each library (11_2H_both_S9_L008 and 14_3B_control_S10_L008). Read 2 files have a higher percentage of shorter reads compared to Read 1

files in both libraries.

Part 3: Alignment and strand-specificity

Using the python script from PS8, in an updated version named "QAA.py," mapped/unmapped counts were tallied providing results in Table 1.

Table 1: Calculated mapping counts for Library 1 (11_2H_both_S9_L008) and Library 2 (14_3B_control_S10_L008) after STAR Alignment.

Count Category	11_2H_both_S9_L008	14_3B_control_S10_L008
Unmapped Count	1381012	251333
Mapped Count	16084601	3995319
Total Read Count	17465613	4246652

Htseq-count was also used to determine reads mapped to features in each STAR-output SAM file. For each library SAM file, htseq-count was run once with "stranded=yes" (read 1 is on the same strand as the gene/feature, read 2 is on the reverse strand) and once with "stranded=reverse" (read 2 is on the same strand as the gene/feature, read 1 is on the reverse strand). If the generated data is strand-specific, then either the "stranded=yes" or "stranded=reverse" mapped count percentages should be higher than the other. If the data is not strand-specific, then "stranded=yes" and "stranded=reverse" mapped count percentages should be roughly equal. Since the percentage of reads mapped to features is significantly higher in the "stranded=reverse" htseq-count files for both library 1 (11_2H_both_S9_L008, 79.12%) and library 2 (14_3B_control_S10_L008, 86.35%) compared to "stranded=yes" htseq-count files for library 1 (3.45%) and library 2 (3.95%), I propose that both libraries are strand-specific. See "labbook_QAA.txt" for bash commands to calculate percentages.