QAA_Jessica_Salguero

2022-09-07

Part 1 – Read quality score distributions

1. Using FastQC produce plots of quality score distributions for R1 and R2 reads. Also, produce plots of the per-base N content, and comment on whether or not they are consistent with the quality score plots.

6.2D.mbnl.S5.L008

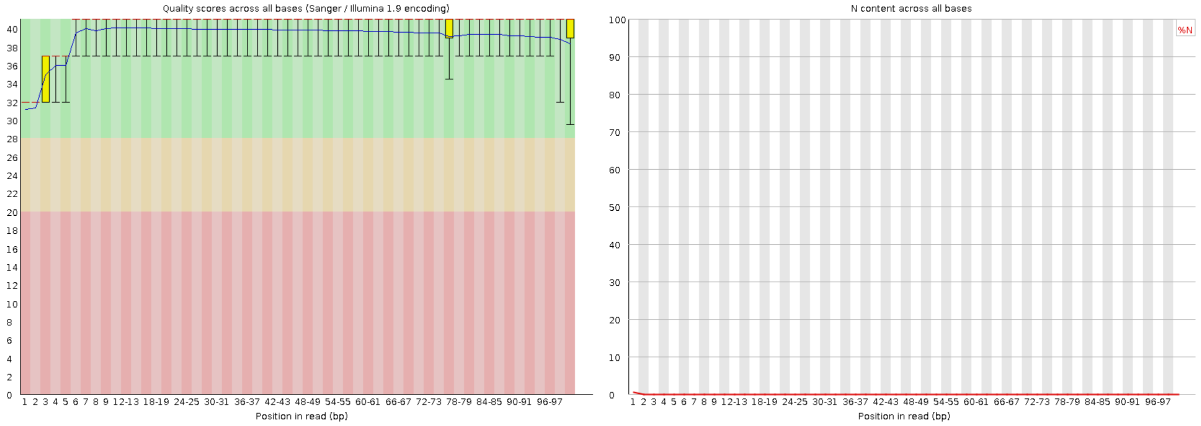


Figure 1: 6.2D.mbnl.S5.L008 – Read 1 Per Base Quality Score (left) and Per Base N Content (right)

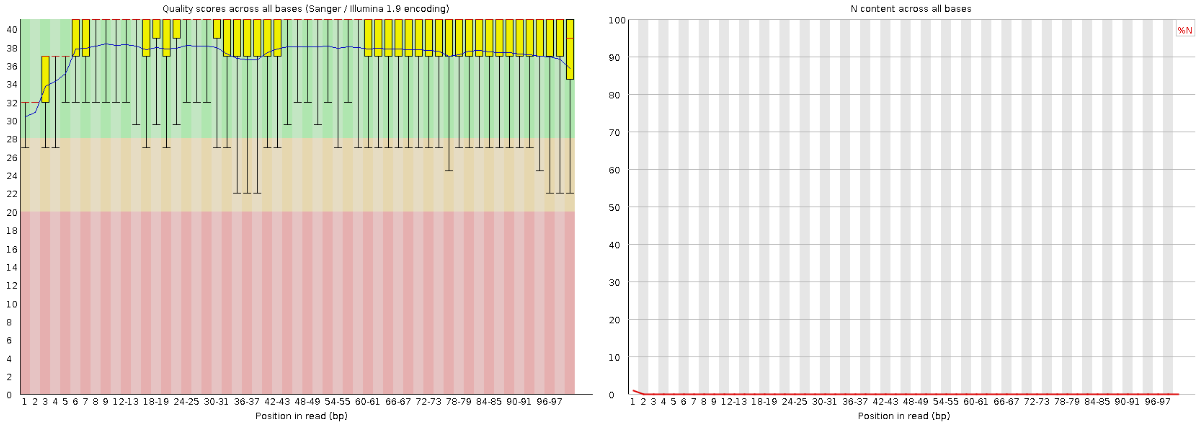


Figure 2: 6.2D.mbnl.S5.L008 – Read 2 Per Base Quality Score (left) and Per Base N Content (right)

- On the read #1 Qscore plots, the first 5 bases had lower quality scores (32-36), which means that there were some significant errors during sequencing in those positions. However, the remaining 96 base positions primarily had a quality score of 39-40 with small and relatively consistent std deviations, which is deemed to be overall very high quality data. These findings are consistent with the per-base N content, as there is a very small percentage of N's in the first few base positions, but the rest of the sequence has a consistent N percentage of almost 0%.

- On the read #2 Qscore plot, it shows similar results as read #1 where the first few positions have a lower quality score than the rest/majority of the sequence. However, the range of values for the quality score have changed, with 96 base positions having a Qscore of 37-38. This decrease in quality scores is likely due to degradation from read #2 being on the instrument for a longer period of time.These findings are consistent with the per-base N content, as there is a very small percentage of N's in the first few base positions, but the rest of the sequence has a consistent N percentage of almost 0%.

15.3C.mbnl.S11.L008

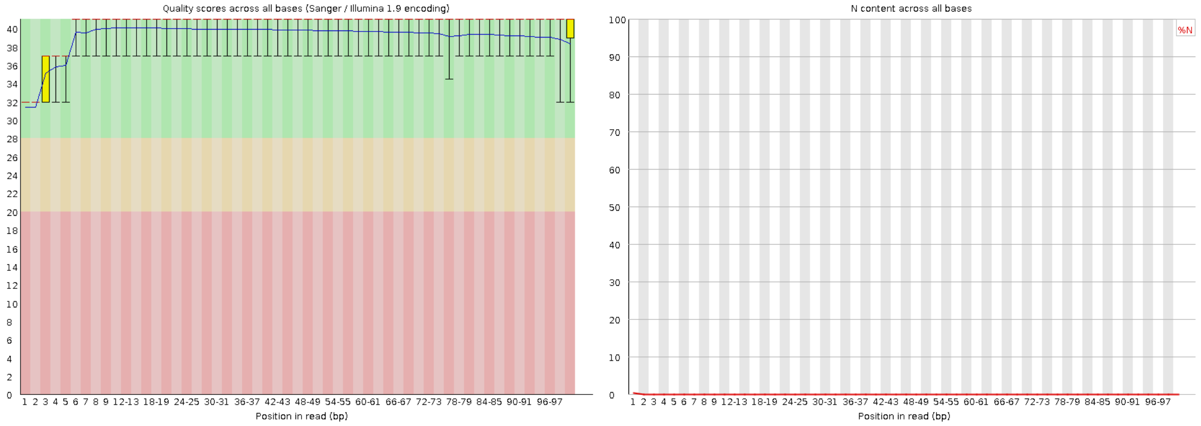


Figure 3: 15.3C.mbnl.S11.L008 – Read 1 Per Base Quality Score (left) and Per Base N Content (right)

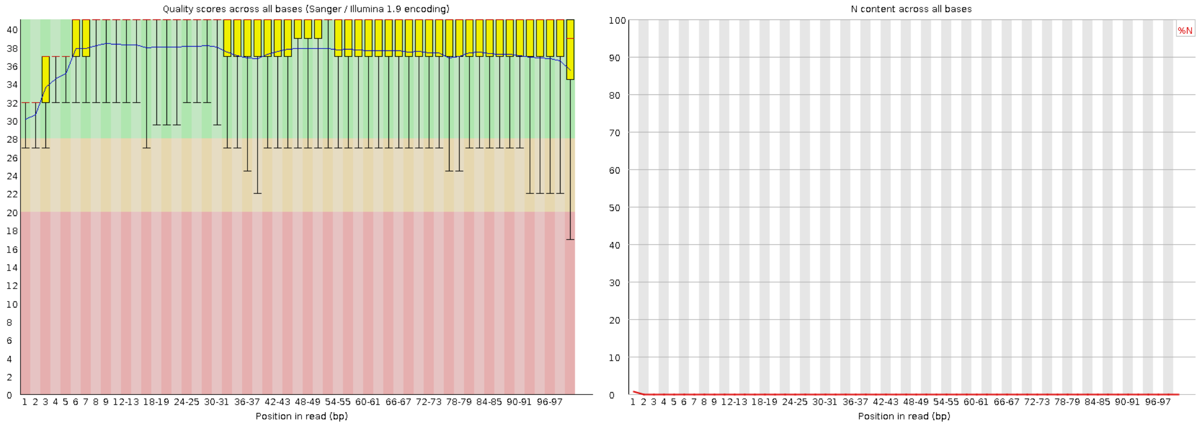


Figure 4: 15.3C.mbnl.S11.L008 – Read 2 Per Base Quality Score (left) and Per Base N Content (right)

- The read #1 Qscore plot is similar to the other sample group, where the first 5 bases had lower quality scores of 32-36, whereas the remaining 96 base positions primarily had a quality score of 38-40 with small and relatively consistent std deviations. Read #1 had overall very high quality data. These findings are consistent with the per-base N content, as there is a very small percentage of N's in the first few base positions, but the rest of the sequence has a consistent N percentage of nearly 0%

- On the read #2 Qscore plot, again the first few positions have a lower quality score than the rest/majority of the sequence. However, the range of values for the quality score have changed, with 96 base positions having a Qscore of 36-38, likely due to degradation caused from being on the instrument for a longer period of time. There were also much larger standard deviations, also likely caused by sample degradation. These findings are consistent with the per-base N content, as there is a very small percentage of N's in the first few base positions, but the rest of the sequence has a consistent N percentage of 0%.

2. Run your quality score plotting script from your Demultiplexing assignment. Describe how the FastQC quality score distribution plots compare to your own. If different, propose an explanation. Also, does the runtime differ? If so, why?

- The FastQC quality score distribution plots look relatively close to the shape of the ones created with my code.The main difference is that they included standard deviations in their graphs. The runtime of fastqc was much faster at producing lots of different graphs, whereas my code took much longer to produce only one histogram. These differences are likely due to using different code; I wrote my code in about a week, whereas their code has been continuously optimized over many years/updates.

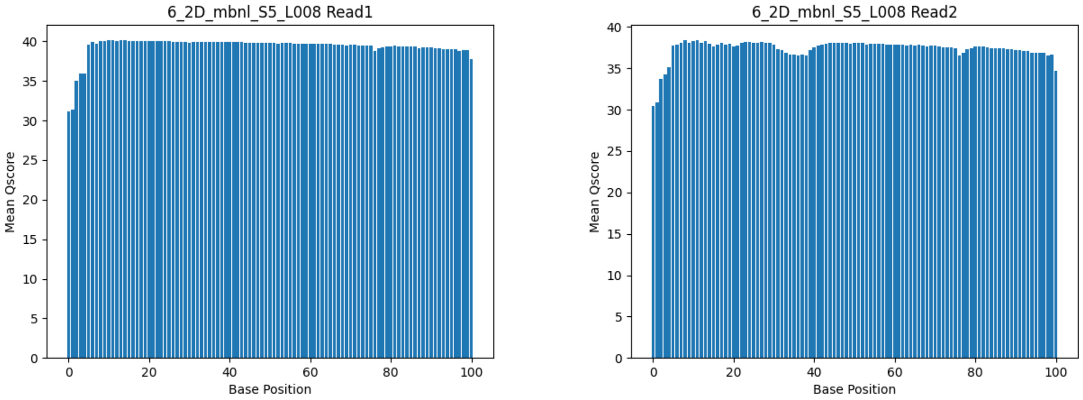


Figure 5: 6.2D.mbnl.S5.L008 – Read 1 (left) and Read 2 (right) Demultiplexing Per Base Quality Score

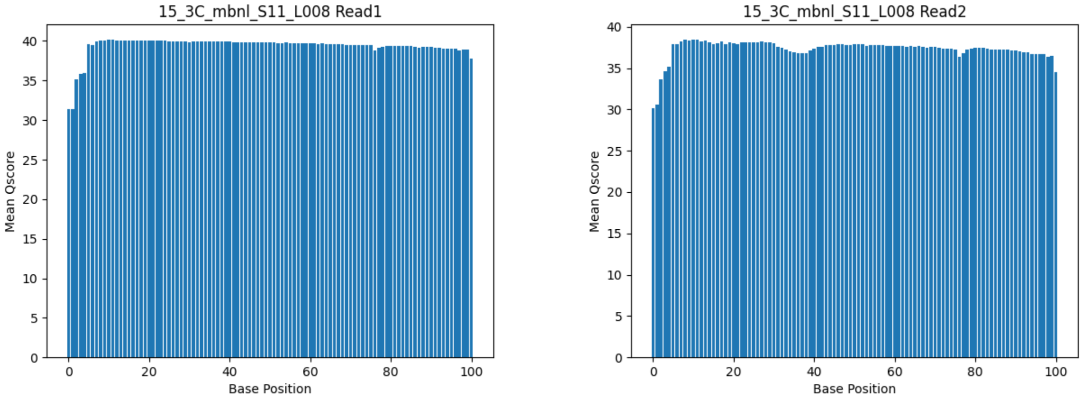


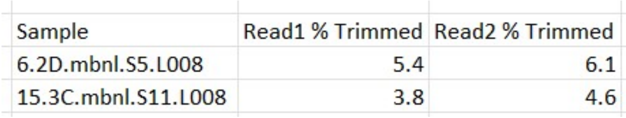
Figure 6: 15.3C.mbnl.S11.L008 – Read 1 (left) and Read 2 (right) Demultiplexing Per Base Quality Score

3. Comment on the overall data quality of your two libraries.

- For both sample datasets, the mean Qscores at every base position reached a minimum of at least 30. Due to this, I believe the overall data quality of these two libraries is very high. This qualifies as high quality data with an error rate of 0.1% or lower; this same Qscore cut-off of 30 is also used by various instruments/companies, including Illumina.

Part 2 – Adaptor trimming comparison

5. Using cutadapt, properly trim adapter sequences from your assigned files. What proportion of reads (both R1 and R2) were trimmed?



Sample	Read1 % Trimmed	Read2 % Trimmed
6.2D.mbnl.S5.L008	5.4	6.1
15.3C.mbnl.S11.L008	3.8	4.6

Figure 7: Cutadapt Percent of Trimmed Reads

7. Plot the trimmed read length distributions for both R1 and R2 reads (on the same plot). Comment on whether you expect R1s and R2s to be adapter-trimmed at different rates.

- For both samples, R2s are trimmed more extensively than R1s. It is expected that R1 and R2 would be adapter-trimmed at different rates, with R2s expected to be trimmed at higher rates because the R2 RNA/DNA sample has been sitting on the sequencer for a longer period of time and has begun to degrade.

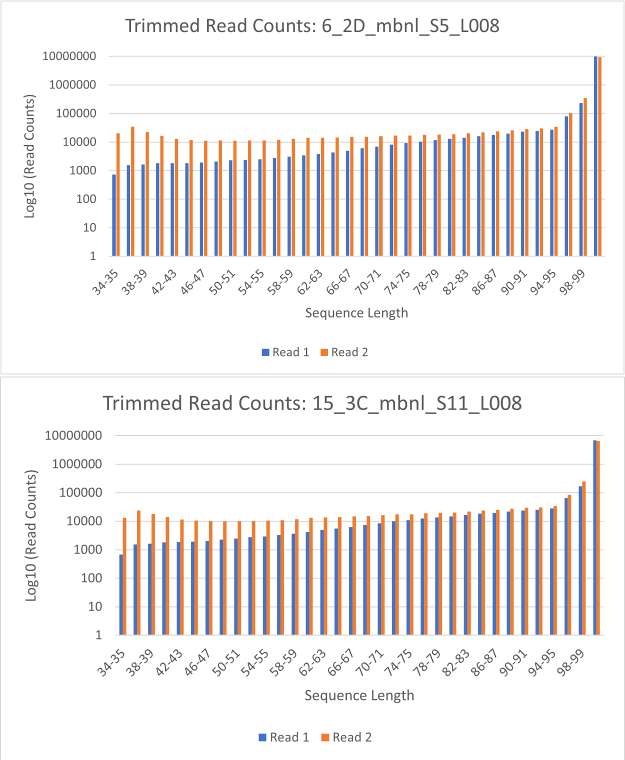
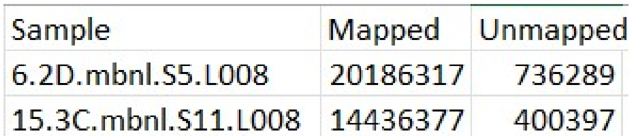


Figure 8: Read 1 and Read 2 Trimmed Length Distribution

Part 3 – Alignment and strand-specificity

10. Report the number of mapped and unmapped reads from each of your 2 sam files.

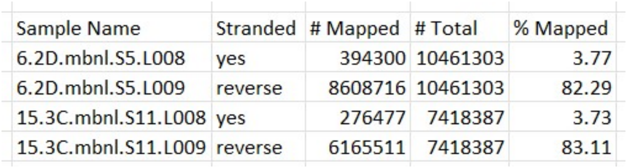


Sample	Mapped	Unmapped
6.2D.mbnl.S5.L008	20186317	736289
15.3C.mbnl.S11.L008	14436377	400397

Figure 9: Mapped and Unmapped Sam Read Counts

12. Count reads that map to features using htseq-count twice: –stranded=yes, –stranded=reverse. Is the data from "strand-specific" RNA-Seq libraries? Include any comands/scripts used. Briefly describe your evidence, using quantitative statements.

- If the data was not strand-specific, then running htseq-count with these two different parameter settings would result in similar percentages of mapped reads. However, when analyzing these datasets, there was a much higher percentage of mapped reads when using stranded=reverse. I propose that these data are strand-specific because 82-83% of the reads are mapped with reverse, as opposed to only 3.7% with stranded=yes.



Sample Name	Stranded	# Mapped	# Total	% Mapped
6.2D.mbnl.S5.L008	yes	394300	10461303	3.77
6.2D.mbnl.S5.L009	reverse	8608716	10461303	82.29
15.3C.mbnl.S11.L008	yes	276477	7418387	3.73
15.3C.mbnl.S11.L009	reverse	6165511	7418387	83.11

Figure 10: Mapped and Unmapped Stranded Read Counts