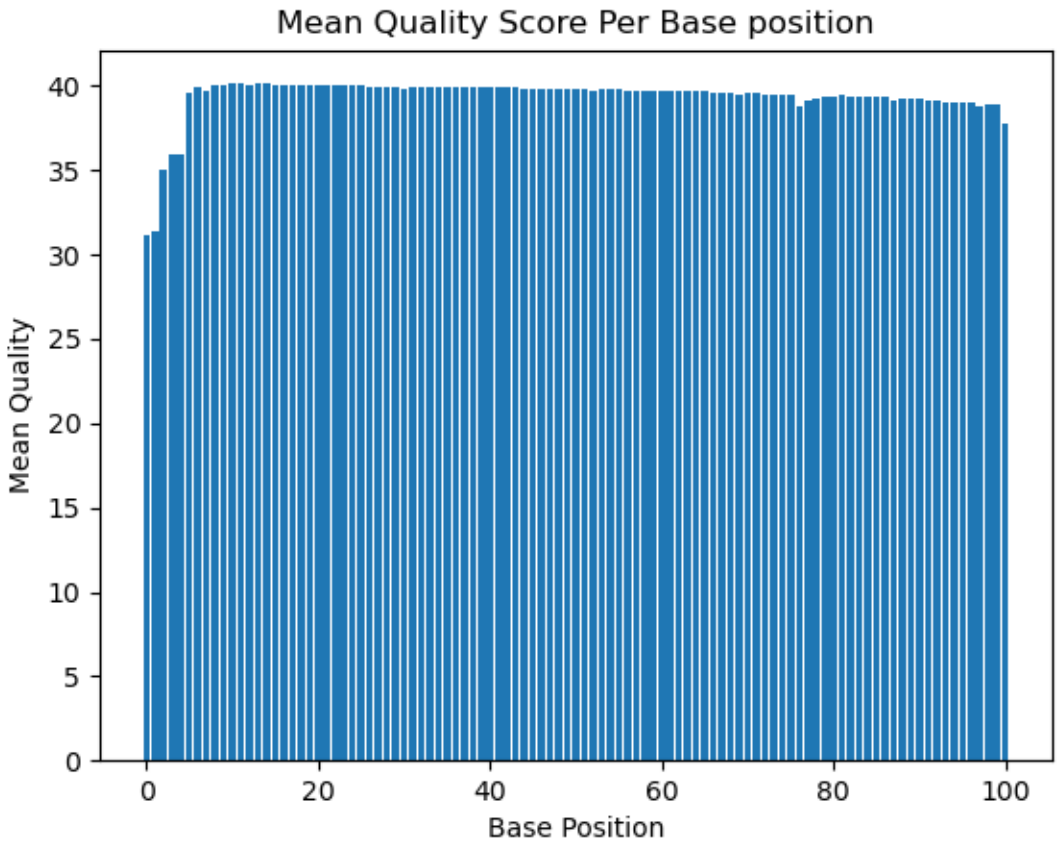
qaa_final

Kobe Ikegami

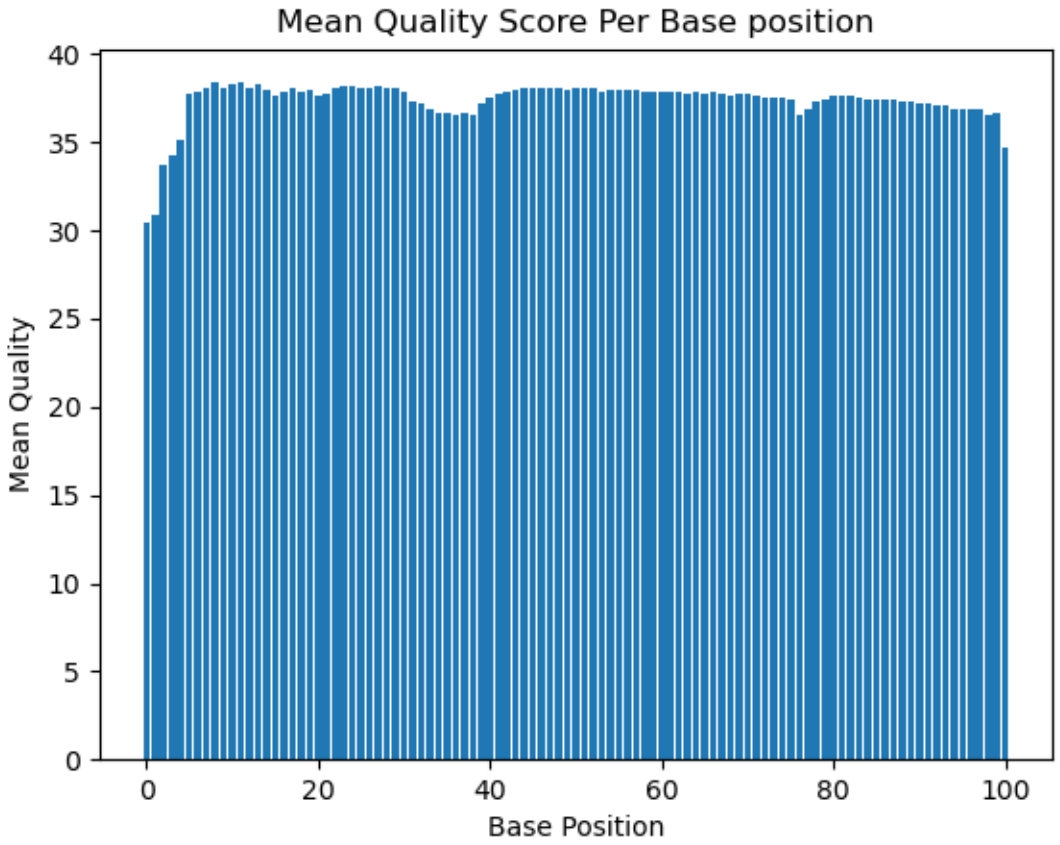
2022-09-07

```
p6r1 = readPNG("/Users/kobeikegami/bioinfo/Bi623/qaa/plots/6_2D_mbnl_S5_L008_R1_001.png")
p6r2 = readPNG("/Users/kobeikegami/bioinfo/Bi623/qaa/plots/6_2D_mbnl_S5_L008_R2_001.png")
p15r1 = readPNG("/Users/kobeikegami/bioinfo/Bi623/qaa/plots/15_3C_mbnl_S11_L008_R1_001.png")
p15r2 = readPNG("/Users/kobeikegami/bioinfo/Bi623/qaa/plots/15_3C_mbnl_S11_L008_R2_001.png")
f6r1 = readPNG("/Users/kobeikegami/bioinfo/Bi623/qaa/plots/6_2D_R1_per_base_quality.png")
f6r2 = readPNG("/Users/kobeikegami/bioinfo/Bi623/qaa/plots/6_2D_R2_per_base_quality.png")
f15r1 = readPNG("/Users/kobeikegami/bioinfo/Bi623/qaa/plots/15_3C_R1_per_base_quality.png")
f15r2 = readPNG("/Users/kobeikegami/bioinfo/Bi623/qaa/plots/15_3C_R2_per_base_quality.png")
```

#6_2D_R1 per base quality (Python Generated)

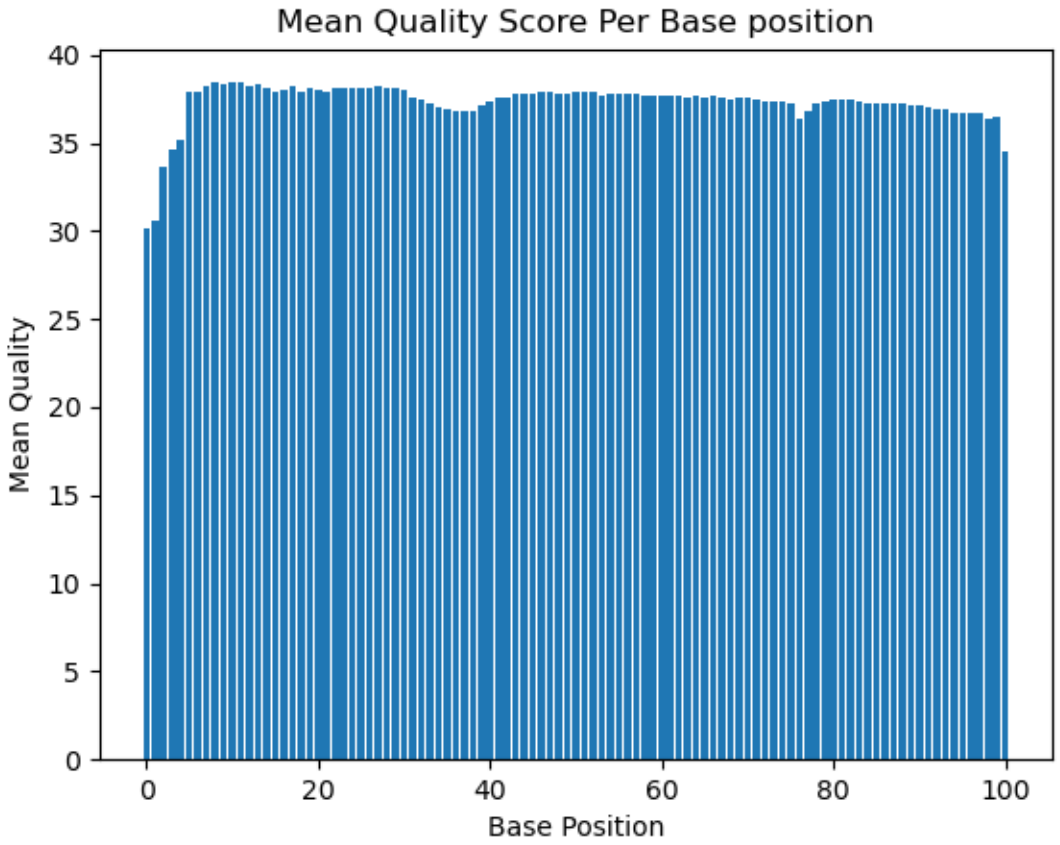


#6_2D_R2 per base quality (Python Generated)

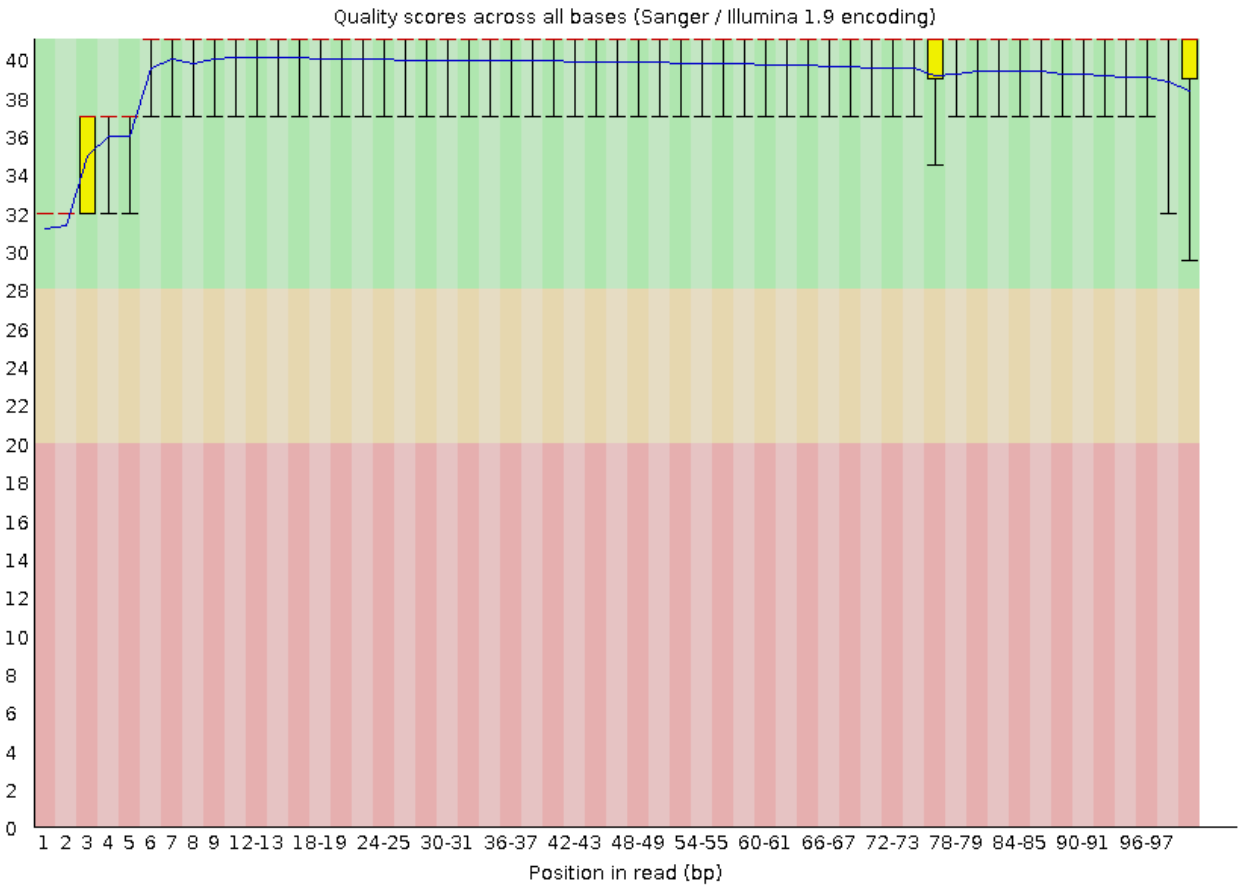


#15_3C_R1 per base quality (Python Generated)

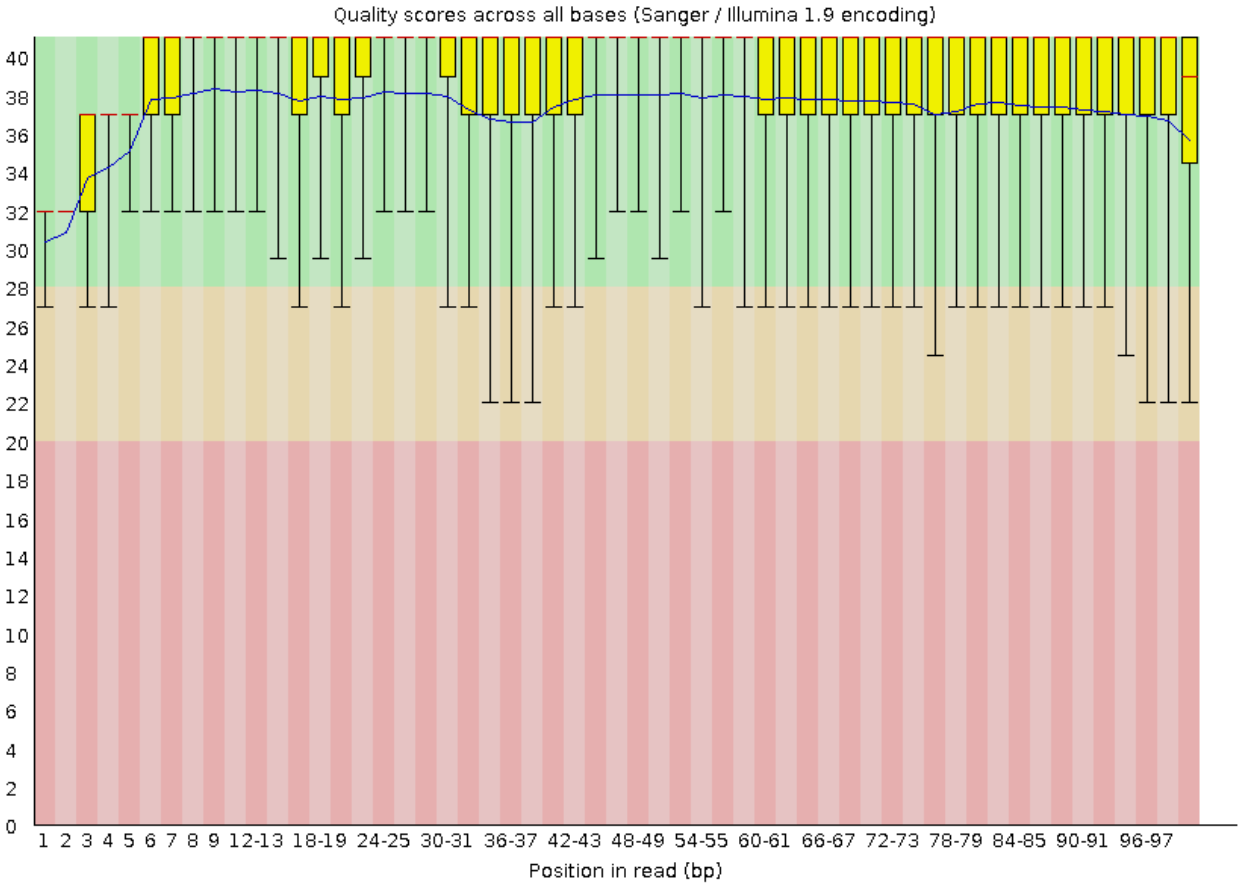
#15_3C_R2 per base quality (Python Generated)



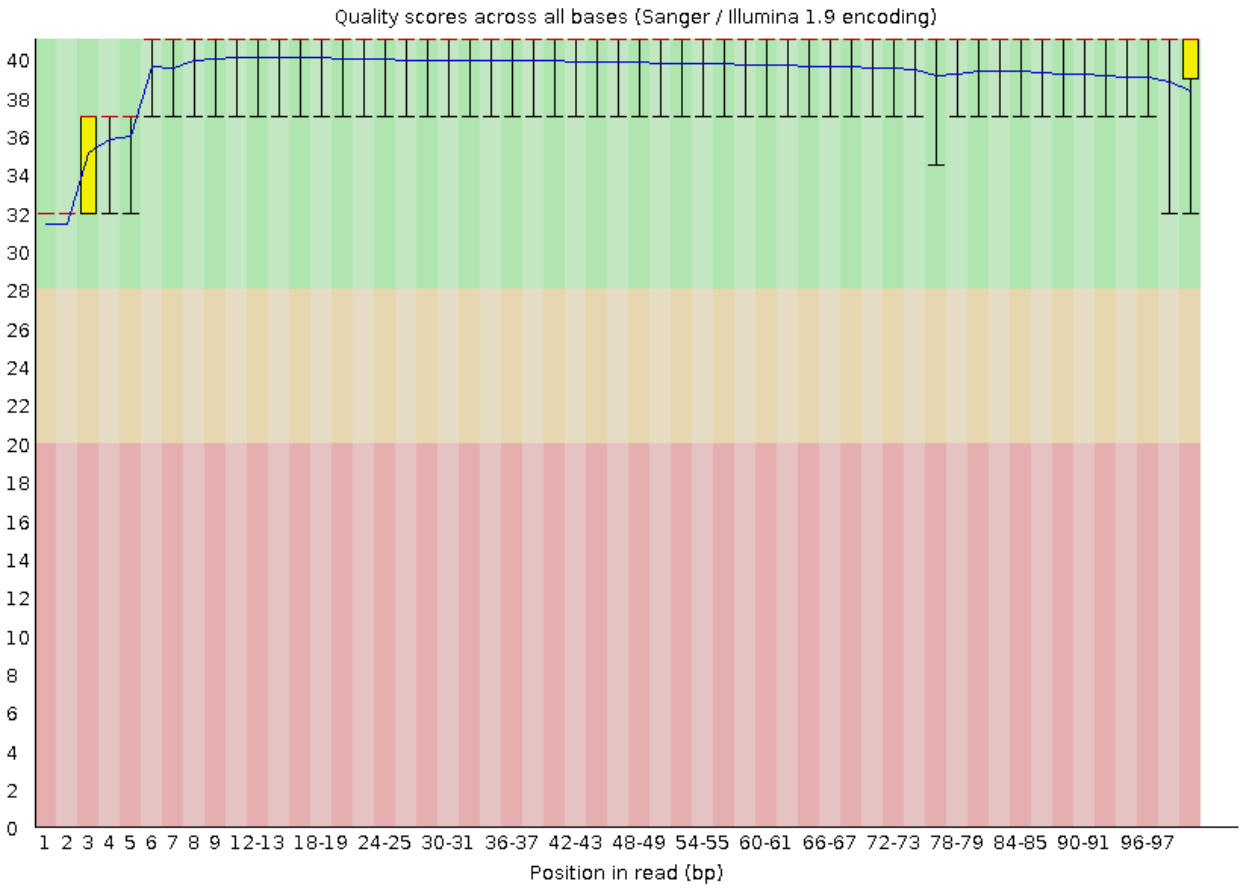
#6_2D_R1 per base quality (FastQC Generated)



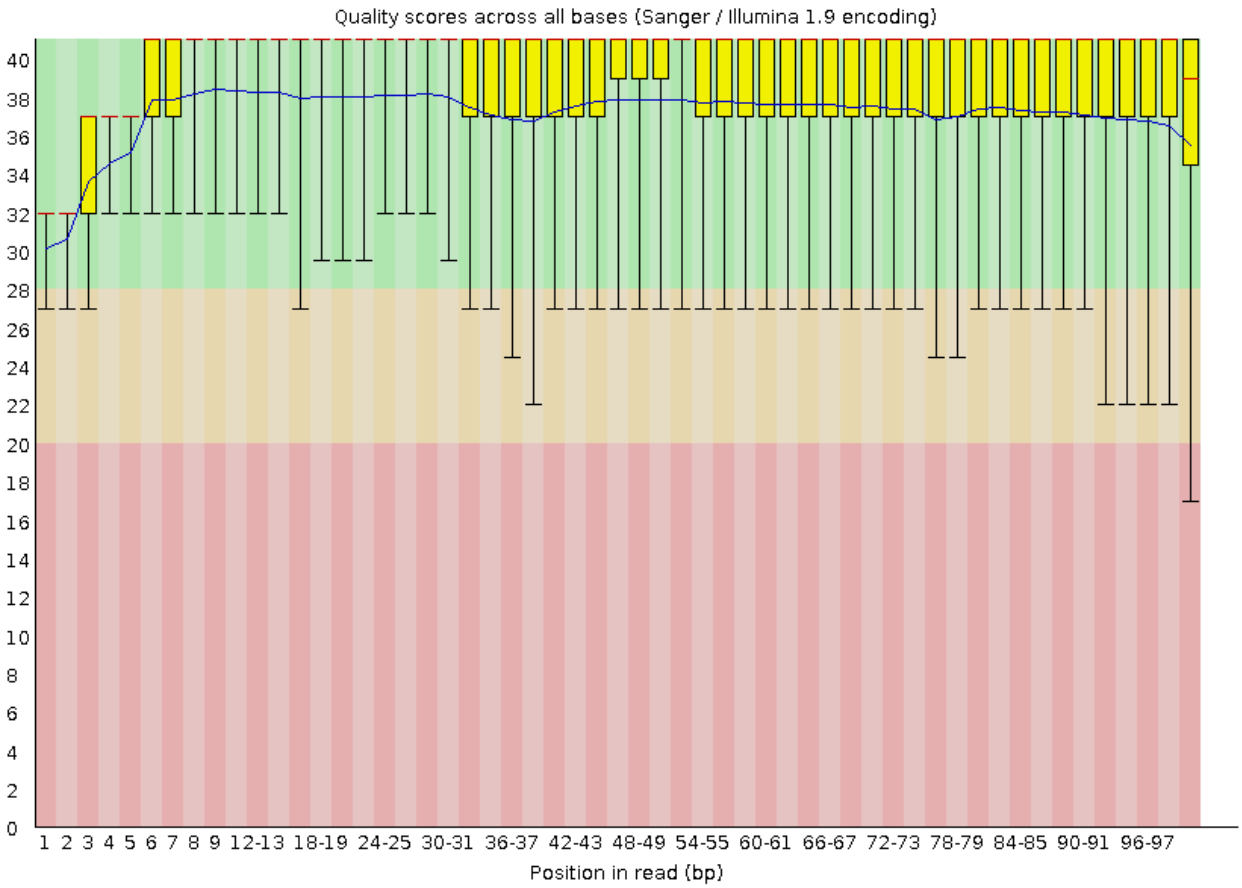
#6_2D_R2 per base quality (FastQC Generated)



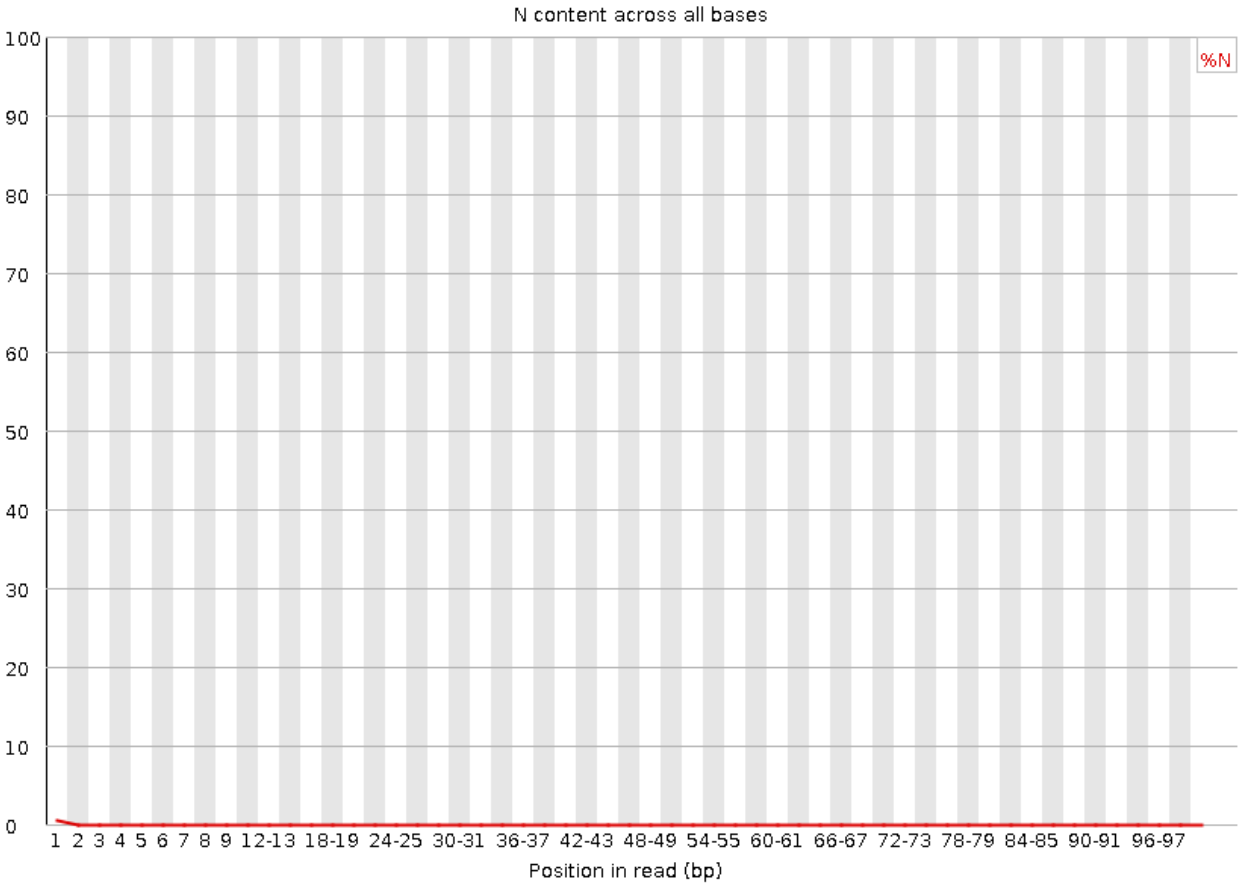
#15_3C_R1 per base quality (FastQC Generated)



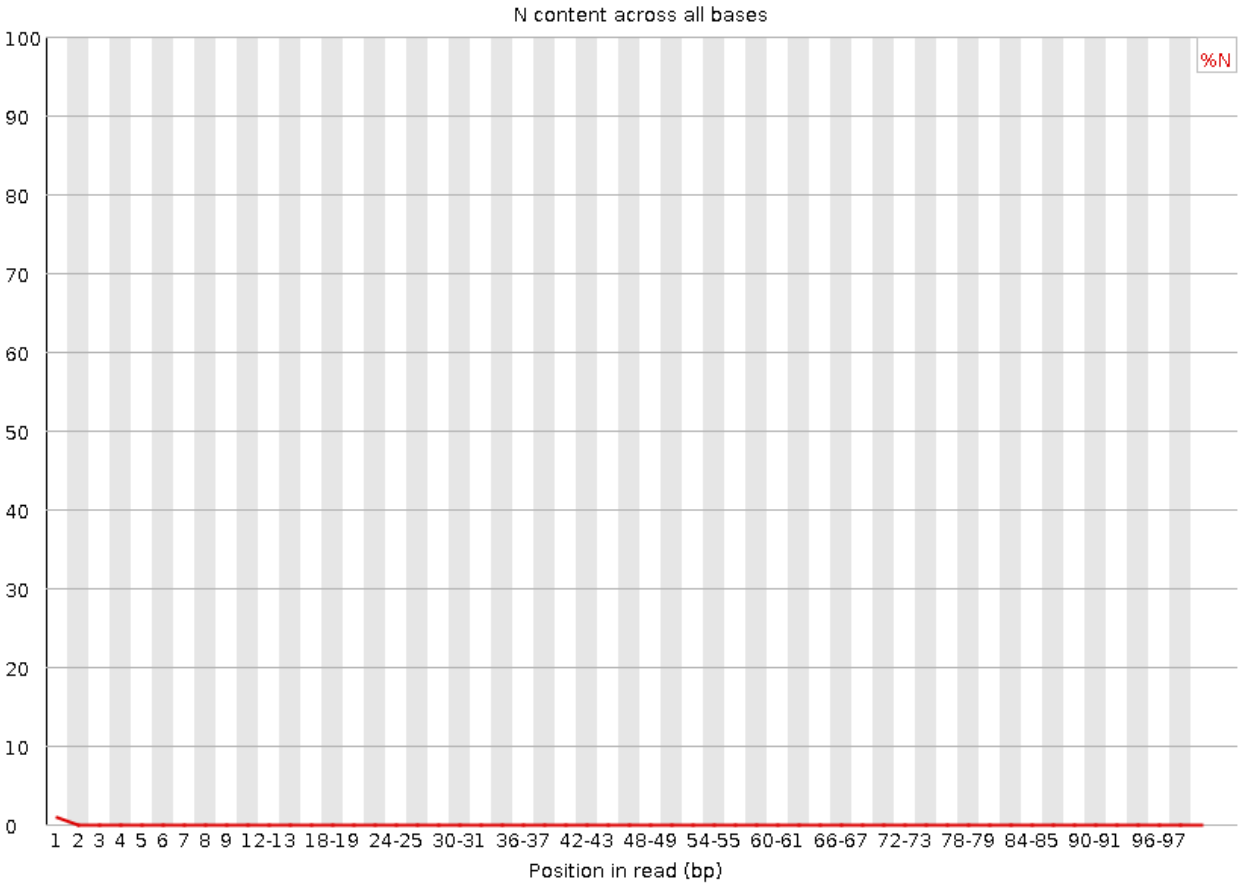
#15_3C_R2 per base quality (FastQC Generated)



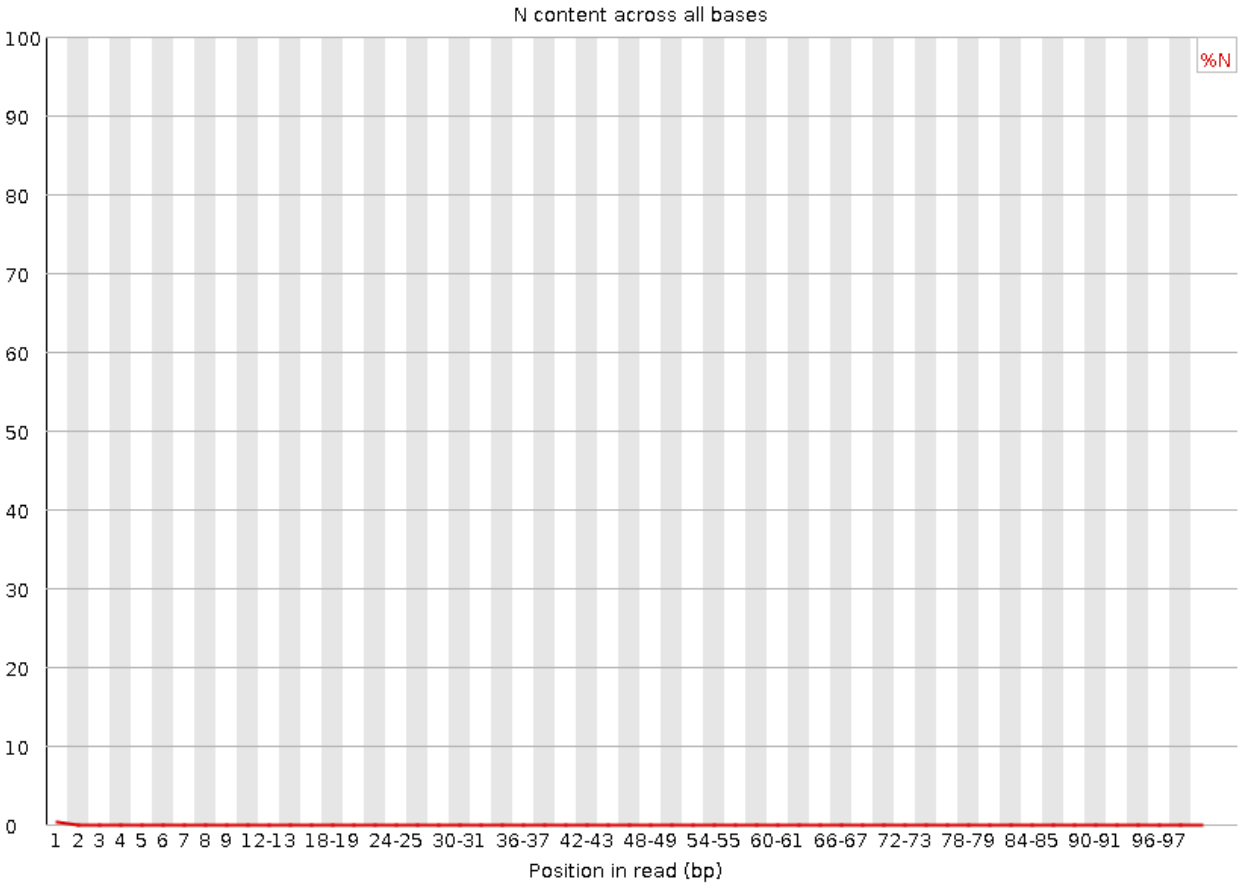
#6_2D_R1 per base n content



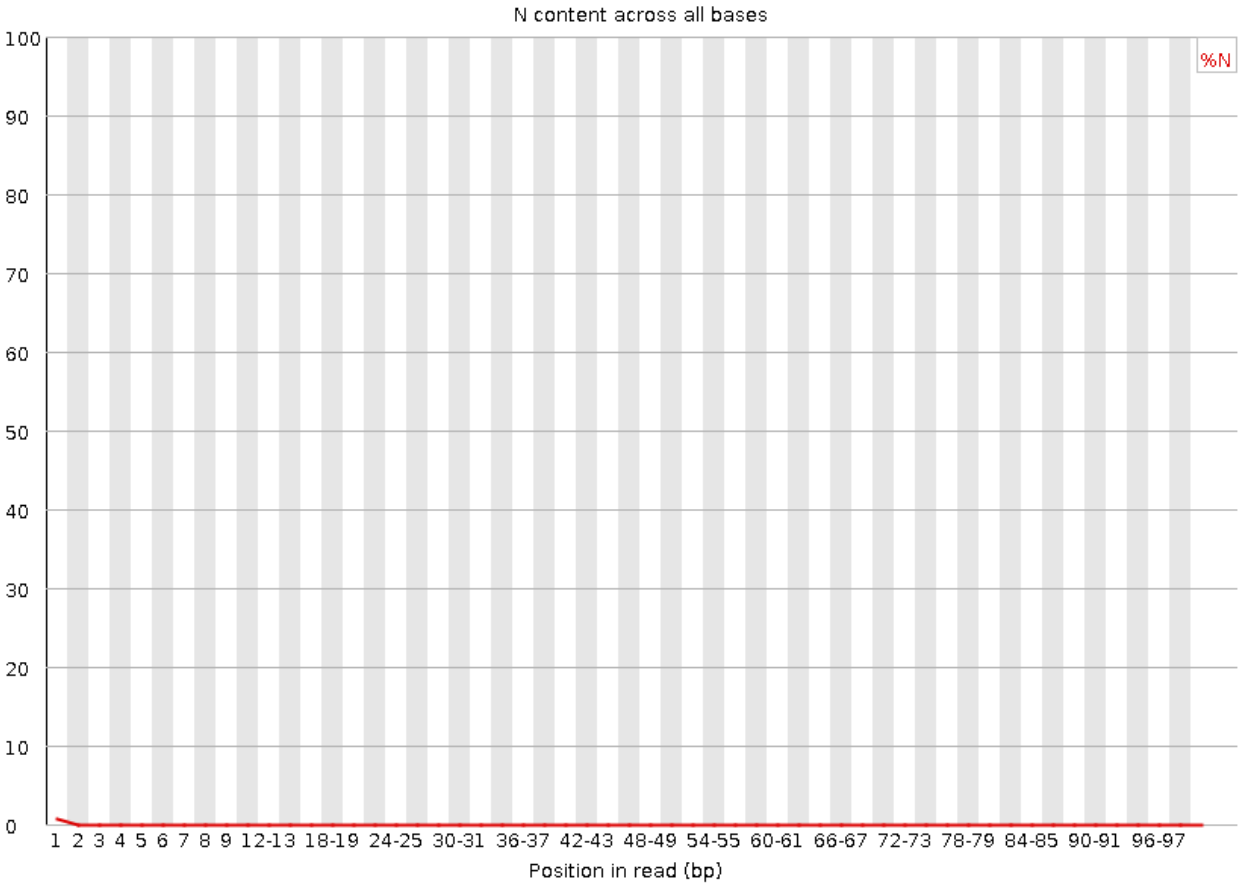
#6_2D_R2 per base n content

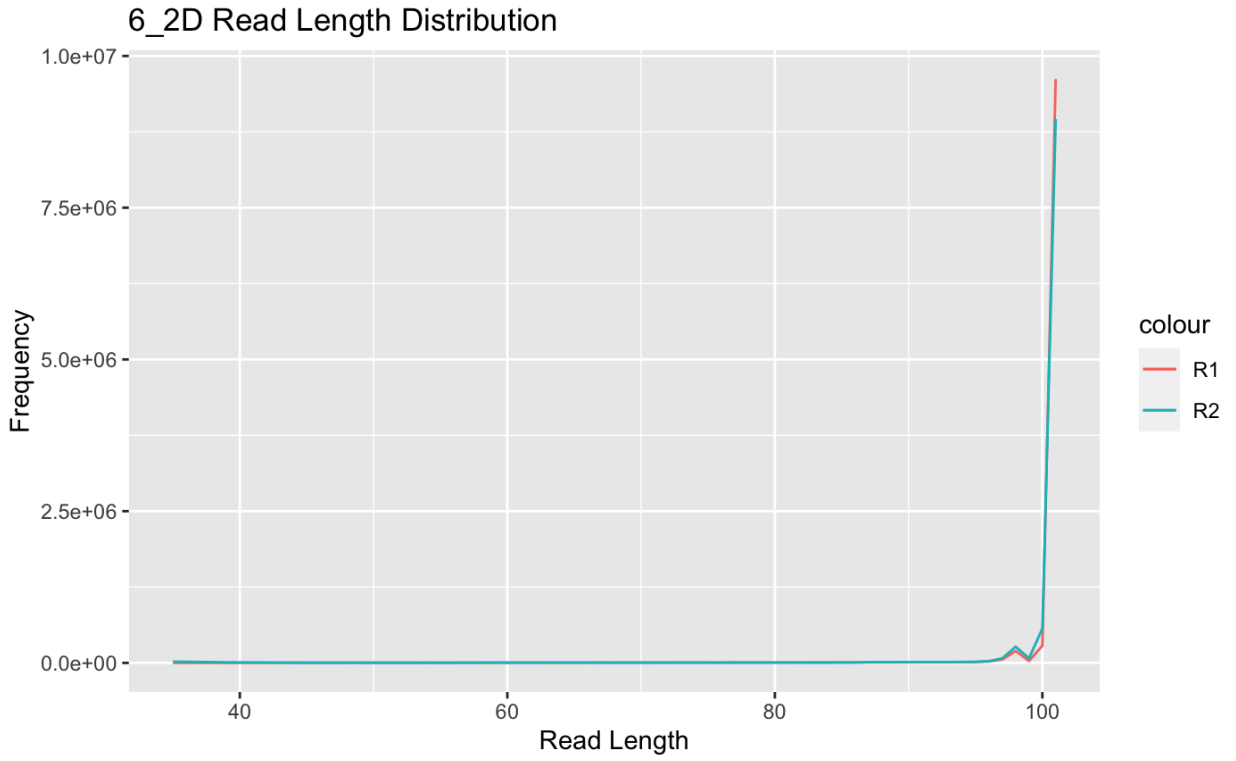


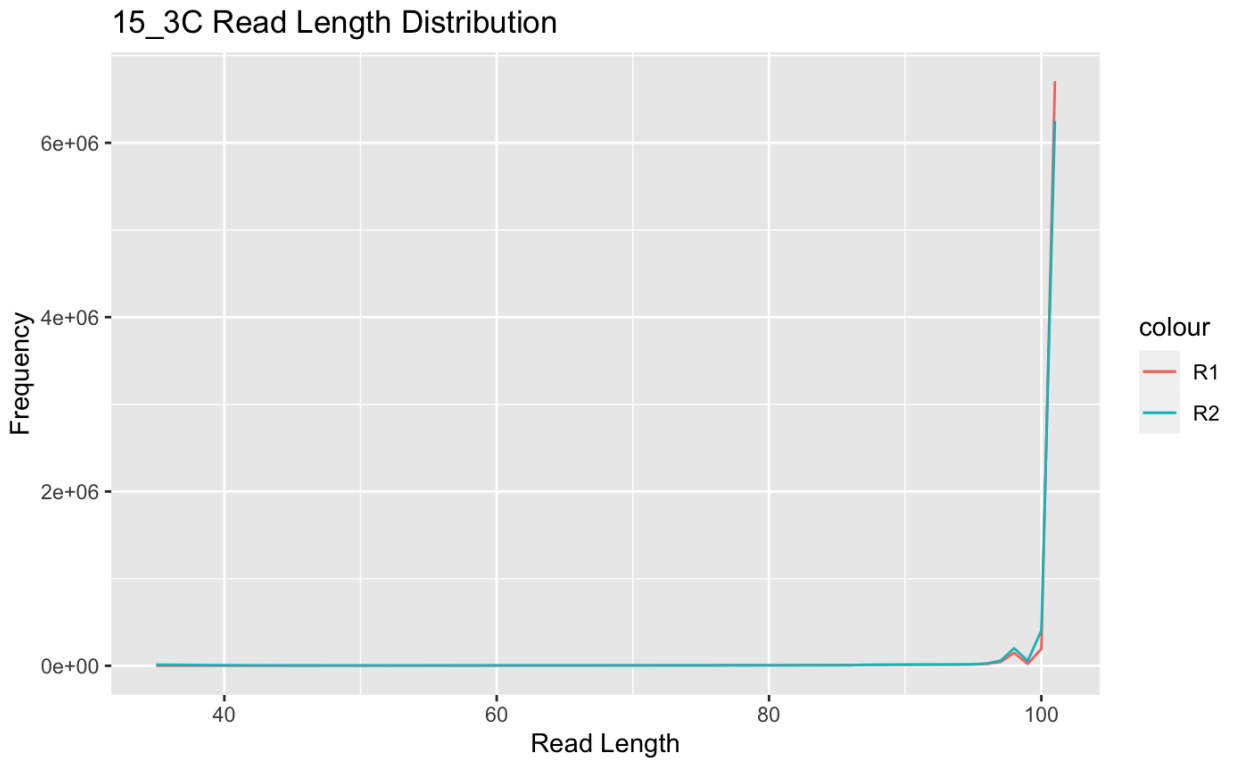
#15_3C_R1 per base n content



#15_3C_R2 per base n content


N content across all bases
%N
100
90
80
70
60
50
40
30
20
10
0
1 2 3 4 5 6 7 8 9 12-13 18-19 24-25 30-31 36-37 42-43 48-49 54-55 60-61 66-67 72-73 78-79 84-85 90-91 96-97
Position in read (bp)


6_2D Read Length Distribution
1.0e+07
7.5e+06
5.0e+06
2.5e+06
0.0e+00
Frequency
40
60
80
100
Read Length
colour
R1
R2



ANSWERS TO QUESTIONS

Answers to QAA Questions

1.2 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 Fast QC plots bear great resemblance to the plots generated with my own code. Both groups show more variance in the distribution of R2 vs R1 for each dataset. The graphs output by my code seem to be a bit more uniform of a distribution but this could be because tyhe FastQC is more detailed, showing error bars ber base position and a trend line along the distribution. Overall, the similarity appears to be strong enough to support my code working as it is supposed to.

Run time differed greatly, FastQC was much faster tham my code, probably because FastQC was developed by a team of experts over a long period of time and my code was written by a single budding bioinformatiacian with little coding experience.

1.3 Comment on the overall data quality of your two libraries. R1 data quality isslightly better, displaying a higher average Q score per base position for their reads. 6_2D and 15_3C don't seem to differ much in data quality. Both have lower quality reads near the beggining of a sequence (30-35) which is to be expected. The majority of the qscores at each position fo reach library is fair, between 35-40

2.5 What proportion of reads (both R1 and R2) were trimmed?

For 6_2D_mbnl_S5_L008: Total read pairs processed: 11,028,244 Read 1 with adapter: 416,045 (3.8%) Read 2 with adapter: 426,679 (3.9%) Pairs written (passing filters): 11,028,244 (100.0%)

For 15_3C_mbnl_S11_L008: Total read pairs processed: 7,806,403 Read 1 with adapter: 417,810 (5.4%) Read 2 with adapter: 362,388 (4.6%) Pairs written (passing filters): 7,806,403 (100.0%)

2.7 Comment on whether you expect R1s and R2s to be adapter-trimmed at different rates. I expect R1 and R2 to be trimmed at the same rate as they both have the same adapter sequence on both ends of each of their molecules. If the areas near adapters have differing quality scores, one may be trimmed more than another.

3.10 report the number of mapped and unmapped reads from each of your 2 sam files

```
6_2D_mbnl_S5_L008:
    Mapped   Unmapped
    14373190    463792

15_3C_mbnl_S11_L008:
    Mapped   Unmapped
    14373190    464878
```

3.11 Count reads that map to features using htseq-count

```
command: $ cat 6_2D_rev_stranded.tsv |grep "^ENS" | awk '{sum+=$2} END{print sum}'

6_2D_yes_stranded.tsv: 392824 (3.75%)
6_2D_rev_stranded.tsv: 8594058 (82%)
15_3C_yes_stranded.tsv: 274199 (3.70%)
15_3C_rev_stranded.tsv: 6140121 (83%)
```

3.12 Demonstrate convincingly whether or not the data are from "strand-specific" RNA-Seq libraries. Include any comands/scripts used. Briefly describe your evidence, using quantitative statements (e.g. "I propose that these data are/are not strand-specific, because X% of the reads are y, as opposed to z.").

```
command: $ cat 6_2D_rev_stranded.tsv |grep "^ENS" | awk '{sum+=$2} END{print sum}'

The command above shows how many reads mapped to a feature when htseq was run with --stranded=yes and --
```

Read Counts Table

Library + Stranded [option]	Count	Percentage
6_2D -s yes	392824	(3.75%)
6_2D -s reverse	8594058	(82%)
15_3C -s yes	274199	(3.70%)
15_3C -s reverse	6140121	(83%)