

FastQC Report

❖ FastQC tool –Introduction

Introduction:- Modern high throughput sequencers can generate tens of millions of sequences in a single run. Before analysing this sequence to draw biological conclusions we should always perform some simple quality control checks to ensure that the raw data looks good and there are no problems or biases in our data.

Most sequencers will generate a QC report as part of their analysis pipeline, but this is usually only focused on identifying problems which were generated by the sequencer itself.

Input :- FastQC supports files in the following formats FastQ (all quality encoding variants)

- Casava FastQ files*
- Colospace FastQ
- GZip compressed FastQ
- SAM
- BAM
- SAM/BAM Mapped only (normally used for colospace data)

Evaluating results:- After several analysis modules analysis of FastQC file is generated. An interactive [HTML output file](#) is generated. In a sequence several parameters are applied on the given input sequence file.

Output file content:-

The below mentioned modules are given as output, each plays an important role for further processing of data in future.

- Basic Statistics
- Per base sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content

Purpose:- FastQC aims to provide a QC report which can spot problems which originate either in the sequencer or in the source of organism.

Bash Script Assignment: FastQC Analysis for SRA Data

❖ Methodology

1) Installation of FastQC

- First check whether system having Java Jdk install as FastQC is a java based tool.

```
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop
```

```
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop$ java --version
java 17.0.9 2023-10-17 LTS
Java(TM) SE Runtime Environment (build 17.0.9+11-LTS-201)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.9+11-LTS-201, mixed mode, sharing)
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop$
```

- Download the FastQC tool from Babraham Bioinformatics site in zip format.
- Using ubuntu unzip it by following below command:

```
unzip FastQC
```

- Using below command check the version of FastQC to ensure proper installation of tool:

```
FastQC --version
```

```
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop
```

```
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/sra_data$ cd ..
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop$ FastQC --version
FastQC v0.12.1
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop$
```

2) Download FastQ Data from SRA [sratoolkit/ wget ftp link]

- We need to do first SRA toolkit installation
- Created a new directory in which sratoolkit is installed.

```
mkdir test
```

- From the below image we can see the steps:

- Using following command we install sratoolkit.

- ```
wget --output-document sratoolkit.tar.gz https://ftp-trace.ncbi.nlm.nih.gov/sra/sdk/current/sratoolkit.current-ubuntu64.tar.gz
```

- Then check for the file by ls
- Now extract the content of tar file by following command:

```
tar -vxzf sratoolkit.tar.gz
```

```
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test
kalpande@LAPTOP-0SUL32IP:~$ cd mnt
-bash: cd: mnt: No such file or directory
kalpande@LAPTOP-0SUL32IP:~$ cd /mnt
kalpande@LAPTOP-0SUL32IP:/mnt$ cd c
kalpande@LAPTOP-0SUL32IP:/mnt/c$ cd Users/kkalp/Desktop/
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop$ mkdir test
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop$ cd test
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$ ls
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$ ^V
B: command not found
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$ wget --output-document sratoolkit.tar.gz https://ftp-trace.ncbi.nlm.nih.gov/sra/sdk/current/sratoolkit.current-ubuntu64.tar.gz
2023-11-04 16:22:21-- https://ftp-trace.ncbi.nlm.nih.gov/sra/sdk/current/sratoolkit.current-ubuntu64.tar.gz
Resolving ftp-trace.ncbi.nlm.nih.gov (ftp-trace.ncbi.nlm.nih.gov)... 2607:f220:41e:250::11, 2607:f220:41e:250::12, 130.14.250.11, ...
Connecting to ftp-trace.ncbi.nlm.nih.gov (ftp-trace.ncbi.nlm.nih.gov)|2607:f220:41e:250::11|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 93175723 (89M) [application/x-gzip]
Saving to: 'sratoolkit.tar.gz'

sratoolkit.tar.gz 100%[=====] 88.86M 1.29MB/s in 61s

2023-11-04 16:23:24 (1.45 MB/s) - 'sratoolkit.tar.gz' saved [93175723/93175723]

kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$ ls
sratoolkit.tar.gz
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$ tar -vxzf sratoolkit.tar.gz
sratoolkit.3.0.7-ubuntu64/
sratoolkit.3.0.7-ubuntu64/README.md
sratoolkit.3.0.7-ubuntu64/README-vdb-config
sratoolkit.3.0.7-ubuntu64/schema/
sratoolkit.3.0.7-ubuntu64/schema/vdb/
sratoolkit.3.0.7-ubuntu64/schema/vdb/vdb.vschema
sratoolkit.3.0.7-ubuntu64/schema/vdb/built-in.vschema
sratoolkit.3.0.7-ubuntu64/schema/insdc/
sratoolkit.3.0.7-ubuntu64/schema/insdc/insdc.vschema
sratoolkit.3.0.7-ubuntu64/schema/insdc/sra.vschema
sratoolkit.3.0.7-ubuntu64/schema/insdc/seq.vschema
sratoolkit.3.0.7-ubuntu64/schema/sra/
```

- Check for the extracted files .
- Now set path of folder by following command:

```
export PATH=$PATH:$PWD/sratoolkit.3.0.7-ubuntu64/bin
```

- Then to again verify do

```
which fastq -dump
```

```

kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test
sratoolkit.3.0.7-ubuntu64/example/perl/base-stats.pl
sratoolkit.3.0.7-ubuntu64/example/perl/mismatch-stats.pl
sratoolkit.3.0.7-ubuntu64/example/perl/gene-lookup.pl
sratoolkit.3.0.7-ubuntu64/example/perl/simplefastq.pl
sratoolkit.3.0.7-ubuntu64/example/perl/splitfastq.pl
sratoolkit.3.0.7-ubuntu64/example/perl/dump-reference.pl
sratoolkit.3.0.7-ubuntu64/example/perl/quality-stats.pl
sratoolkit.3.0.7-ubuntu64/CHANGES
sratoolkit.3.0.7-ubuntu64/README-blastn
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test$ ls
sratoolkit.3.0.7-ubuntu64 sratoolkit.tar.gz
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test$ export PATH=$PATH:$PWD/sratoolkit.3.0.7-ubuntu64/bin
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test$ which fastq-dump
/mnt/c/Users/kkalp/Desktop/test/sratoolkit.3.0.7-ubuntu64/bin/fastq-dump
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test$ ls
sratoolkit.3.0.7-ubuntu64 sratoolkit.tar.gz
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test$

```

- Go for configuration using

```
vdb-config -i
```

```

kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test

[X] enable local file-caching

location of user-repository:
[choose]
[clear]

process-local location:
[choose] /mnt/c/Users/kkalp/Desktop/test/test1
[clear]

RAM used: + 1 - MB

exit vdb-config tool
kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/test$

```

- To test whether tool is functional or not we use following command for our given id:

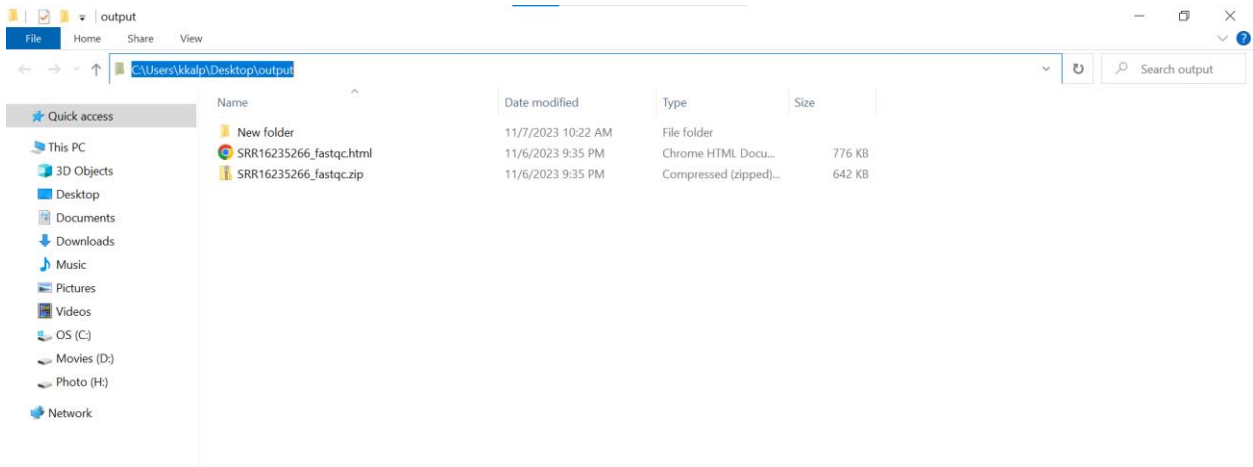
```
fastq-dump --stdout -X 2 SRR162352266
```



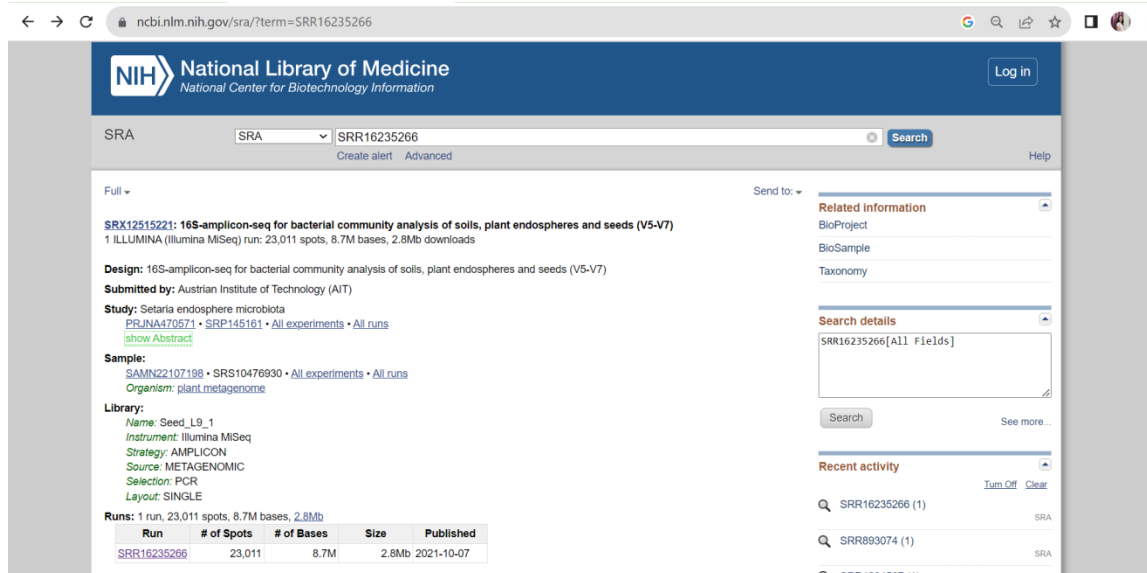
```
spots read : 23,011
reads read : 46,022
reads written : 23,011
reads 0-length : 23,011
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$ ls
SRR16235266.fastq SRR8185279.fasta sratoolkit.3.0.7-ubuntu64 sratoolkit.tar.gz test1
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/test$
```

```
Select kalpande@LAPTOP-0SUL32IP: /mnt/c/Users/kkalp/Desktop/output
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop$ cd apps/FastQC/
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/apps/FastQC$./fastqc /mnt/c/Users/kkalp/Desktop/test/SRR16235266.fastq --outdir /mnt/c/Users/kkalp/Desktop/output
null
Started analysis of SRR16235266.fastq
Approx 5% complete for SRR16235266.fastq
Approx 10% complete for SRR16235266.fastq
Approx 15% complete for SRR16235266.fastq
Approx 20% complete for SRR16235266.fastq
Approx 25% complete for SRR16235266.fastq
Approx 30% complete for SRR16235266.fastq
Approx 35% complete for SRR16235266.fastq
Approx 40% complete for SRR16235266.fastq
Approx 45% complete for SRR16235266.fastq
Approx 50% complete for SRR16235266.fastq
Approx 55% complete for SRR16235266.fastq
Approx 60% complete for SRR16235266.fastq
Approx 65% complete for SRR16235266.fastq
Approx 70% complete for SRR16235266.fastq
Approx 75% complete for SRR16235266.fastq
Approx 80% complete for SRR16235266.fastq
Approx 85% complete for SRR16235266.fastq
Approx 90% complete for SRR16235266.fastq
Approx 95% complete for SRR16235266.fastq
Analysis complete for SRR16235266.fastq
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/apps/FastQC$ cd ..
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/apps$ cd ..
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop$ cd output/
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/output$ ls
SRR15616379_1_fastqc.html SRR15616379_1_fastqc.zip SRR16235266_fastqc.html SRR16235266_fastqc.zip SRR8185279_1_fastqc.html SRR8185279_1_fastqc.zip
kalpande@LAPTOP-0SUL32IP:/mnt/c/Users/kkalp/Desktop/output$
```

.html file is created



Below is the detail regarding used Accession Id-



NIH National Library of Medicine  
National Center for Biotechnology Information

SRA SRR16235266 Search

Full ▾ Send to: ▾

**SRX12515221: 16S-amplicon-seq for bacterial community analysis of soils, plant endospheres and seeds (V5-V7)**  
1 ILLUMINA (Illumina MiSeq) run: 23,011 spots, 8.7M bases, 2.8Mb downloads

**Design:** 16S-amplicon-seq for bacterial community analysis of soils, plant endospheres and seeds (V5-V7)  
**Submitted by:** Austrian Institute of Technology (AIT)  
**Study:** Setaria endosphere microbiota  
PRJNA470571 • SRP145161 • All experiments • All runs  
[Show Abstract](#)

**Sample:** SAMN22107198 • SRS10476930 • All experiments • All runs  
**Organism:** plant metagenome

**Library:**  
Name: Seed\_L9\_1  
Instrument: Illumina MiSeq  
Strategy: AMPLICON  
Source: METAGENOMIC  
Selection: PCR  
Layout: SINGLE

**Runs:** 1 run, 23,011 spots, 8.7M bases, 2.8Mb

| Run                         | # of Spots | # of Bases | Size  | Published  |
|-----------------------------|------------|------------|-------|------------|
| <a href="#">SRR16235266</a> | 23,011     | 8.7M       | 2.8Mb | 2021-10-07 |

**Related information**  
BioProject  
BioSample  
Taxonomy

**Search details**  
SRR16235266[All Fields]  
Search See more...

**Recent activity**  
Turn Off Clear  
SRR16235266 (1) SRA  
SRR893074 (1) SRA  
SRR1234567 (1)

## ❖ Results

### 3) Run FastQC Analysis

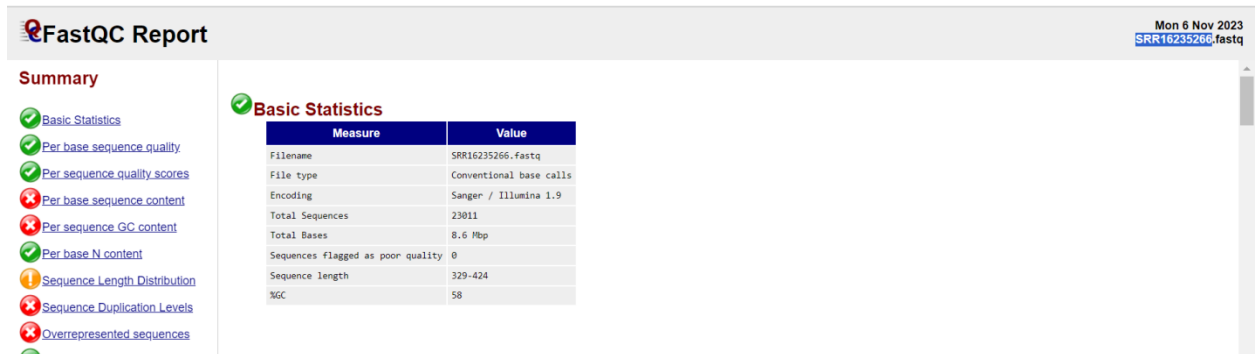
A [.html](#) file is generated for the given fastq file.

Several modules are checked and an interactive graphical file id generated.

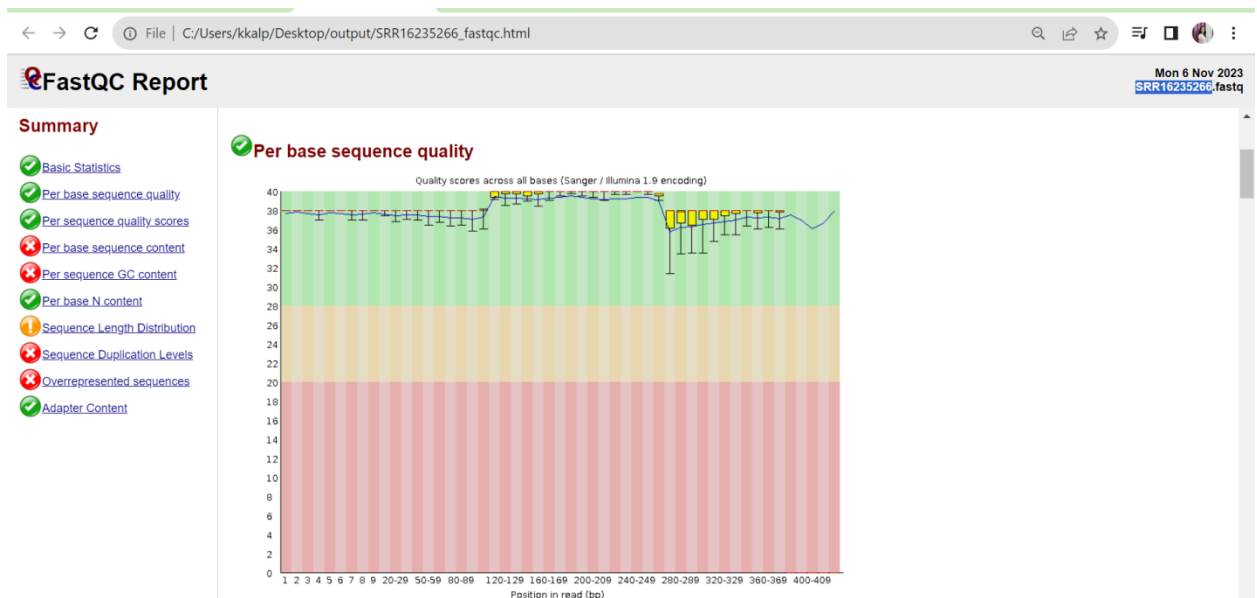
Link of output file is: [click](#)

- The **green tick** represents that given sequence is within the expected or acceptable range, suggesting good data quality.
- The **orange tick** is used to highlight areas of moderate concern. It suggests that there may be some issues or deviations from ideal quality, but they are not severe enough to be an immediate cause for alarm.
- The **red tick** is used to flag critical issues or areas of significant concern. If a metric or result is marked in red, it indicates a potential problem that may require attention or further investigation.

- i. **Basic Statistics** - The Basic Statistics module generates some simple composition statistics for the file analysed.

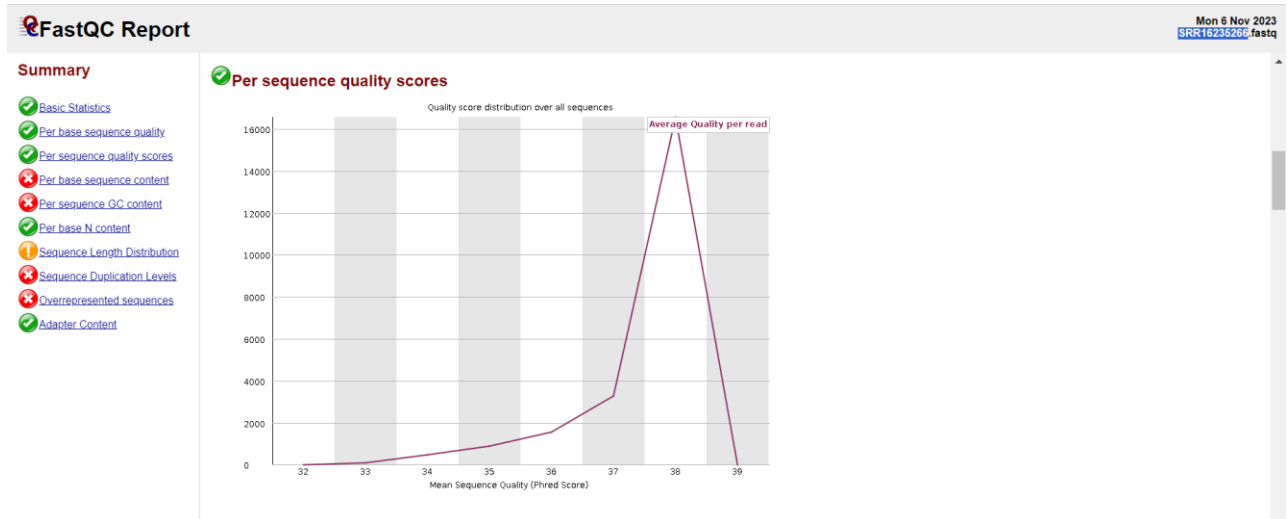


- ii. **Per base sequence quality**- This view shows an overview of the range of quality values across all bases at each position in the FastQ file. The good quality reads are checked by Phred parameter whose **good value should be above 28**. Whereas **poor quality ranges below 20**.



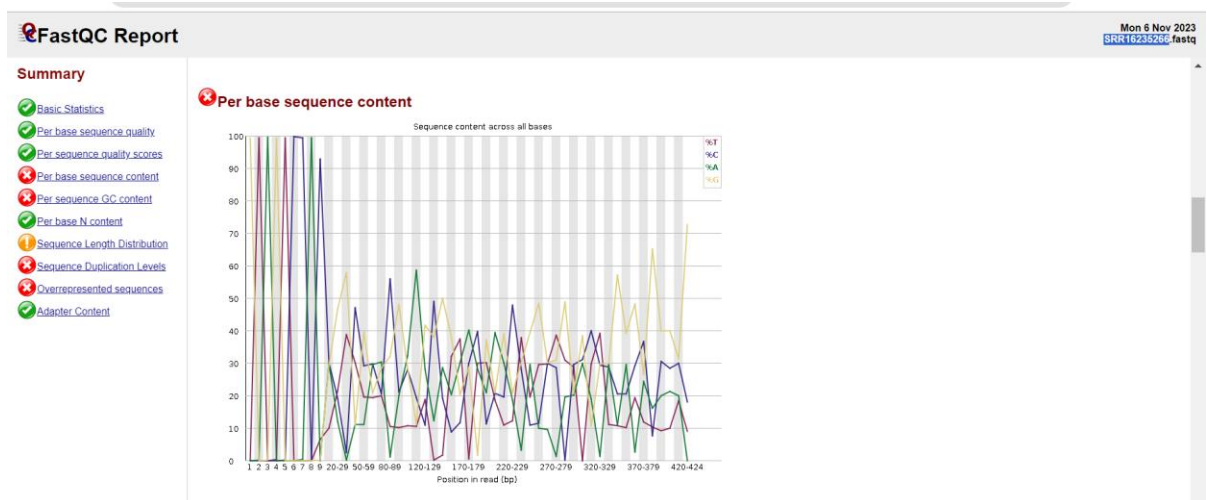
- iii. **Per sequence quality scores**-The per sequence quality score report allows you to see if a **subset of your sequences have universally low quality values**. It is often the case that a subset of sequences will have universally poor quality, often because they are poorly imaged , however these should represent only a small percentage of the total sequences.



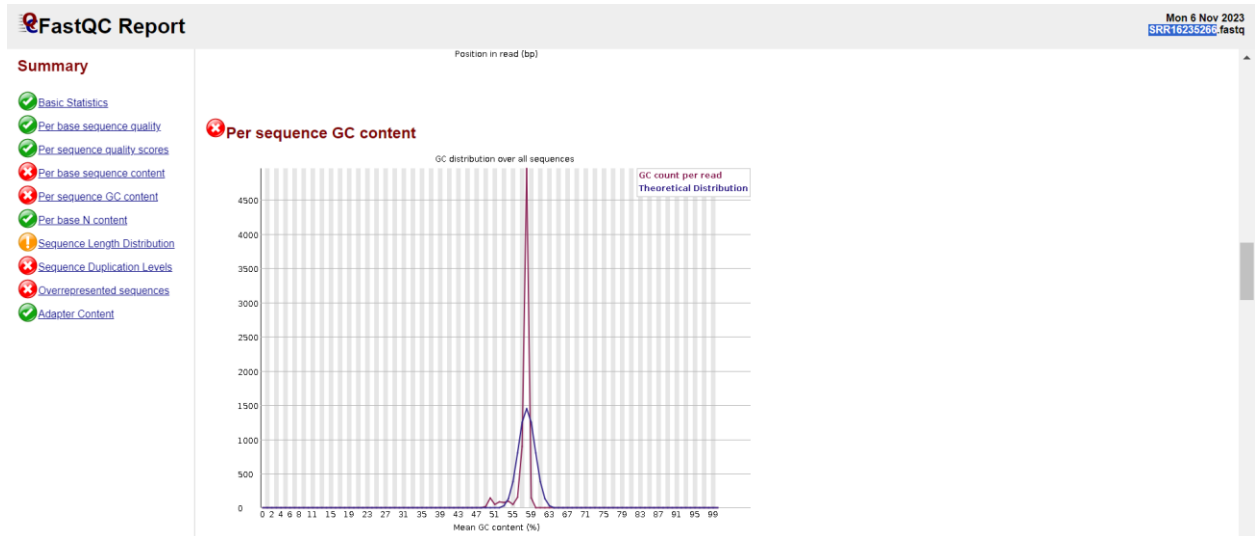


- iv. **Per base sequence content** -Per Base Sequence Content plots out the proportion of each base position in a file for which each of the four normal DNA bases has been called. In a random library you would **expect that there would be little to no difference between the different bases of a sequence run, so the lines in this plot should run parallel with each other**. The relative amount of each base should reflect the overall amount of these bases in your genome, but in any case they should not be hugely imbalanced from each other.

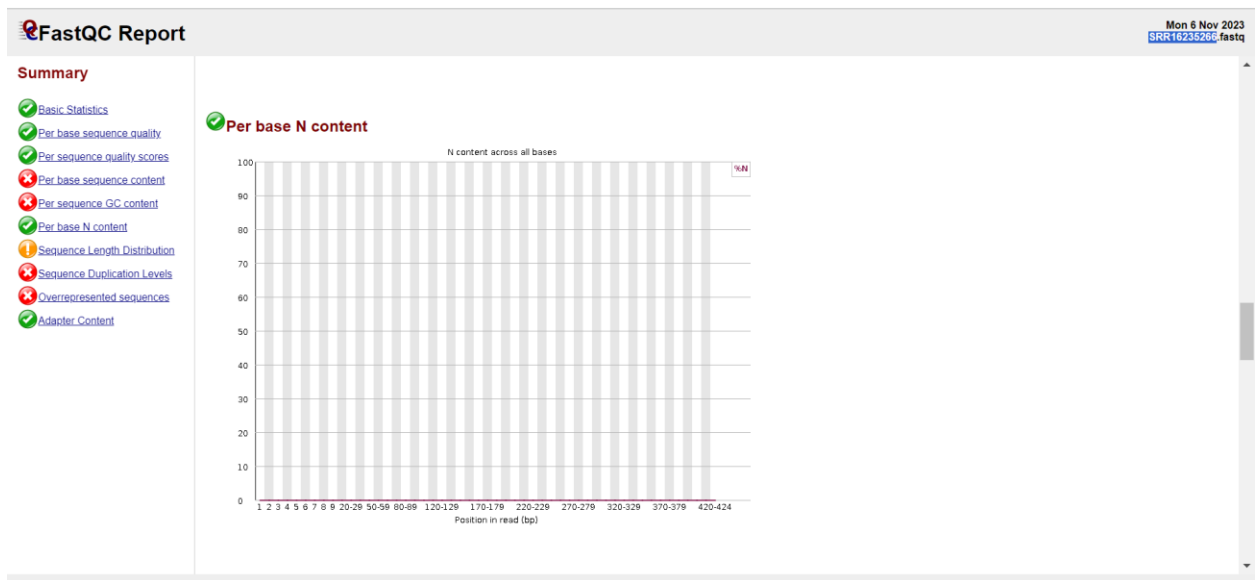
In our case, the bases are not in parallel with each other.



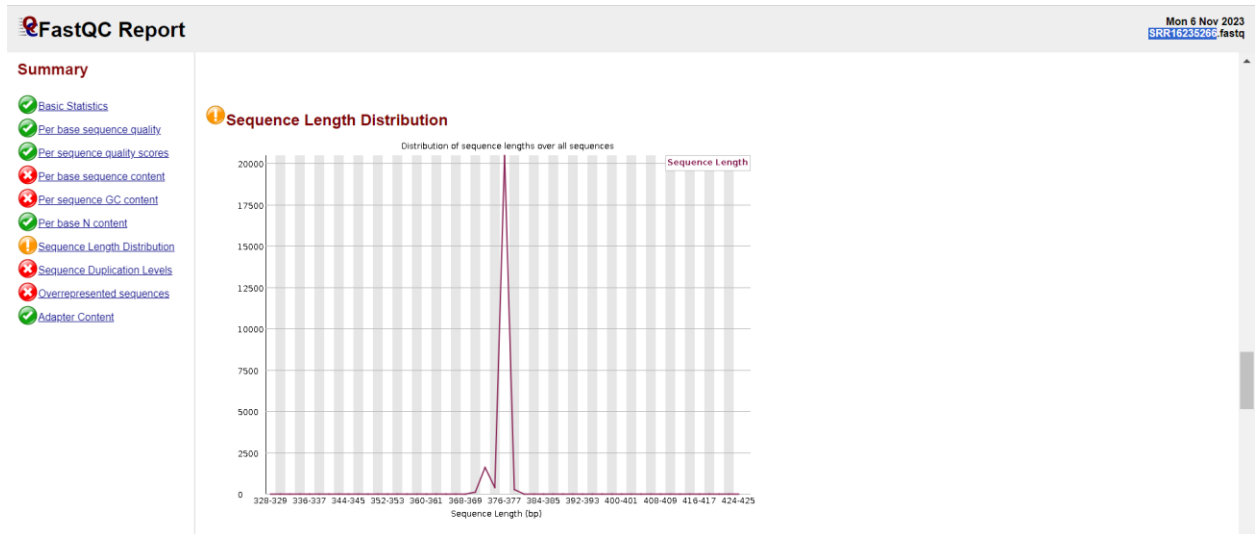
- v. **Per sequence GC content**-This module measures the GC content across the whole length of each sequence in a file and compares it to a modelled normal distribution of GC content. In a normal random library you would expect to see a roughly normal distribution of GC content **where the central peak corresponds to the overall GC content of the underlying genome**. Since we don't know the the GC content of the genome the modal GC content is calculated from the observed data and used to build a reference distribution.



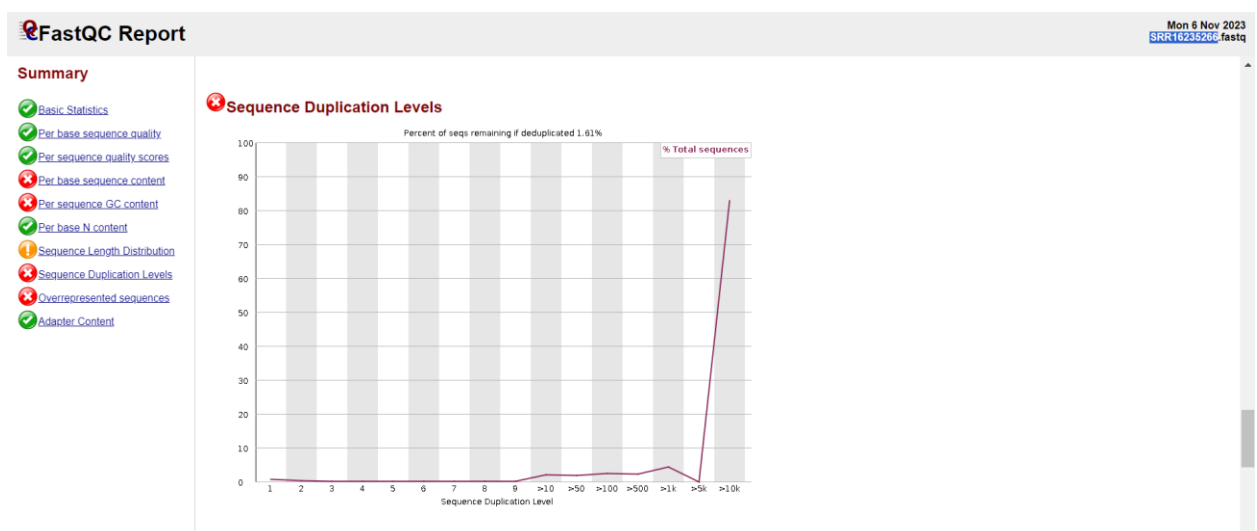
- vi. **Per base N content**- If a sequencer is unable to make a base call with sufficient confidence then **it will normally substitute an N rather than a conventional base call**. This module plots out the percentage of base calls at each position for which an N was called.



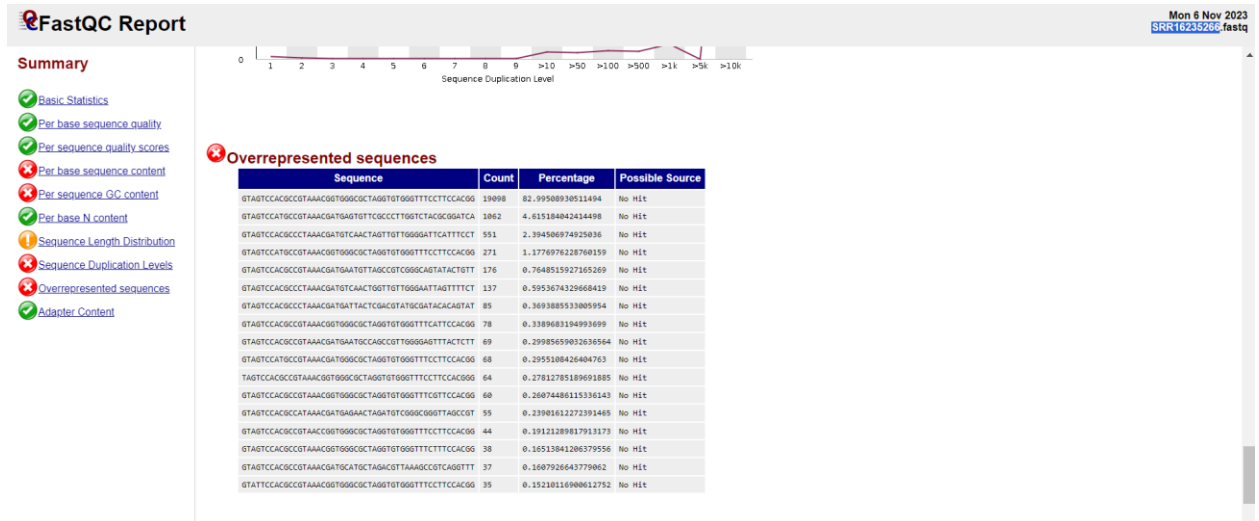
- vii. **Sequence Length Distribution**-Some high throughput sequencers generate sequence fragments of uniform length, but others can contain reads of wildly varying lengths. Even within uniform length libraries some pipelines will trim sequences to remove poor quality base calls from the end.



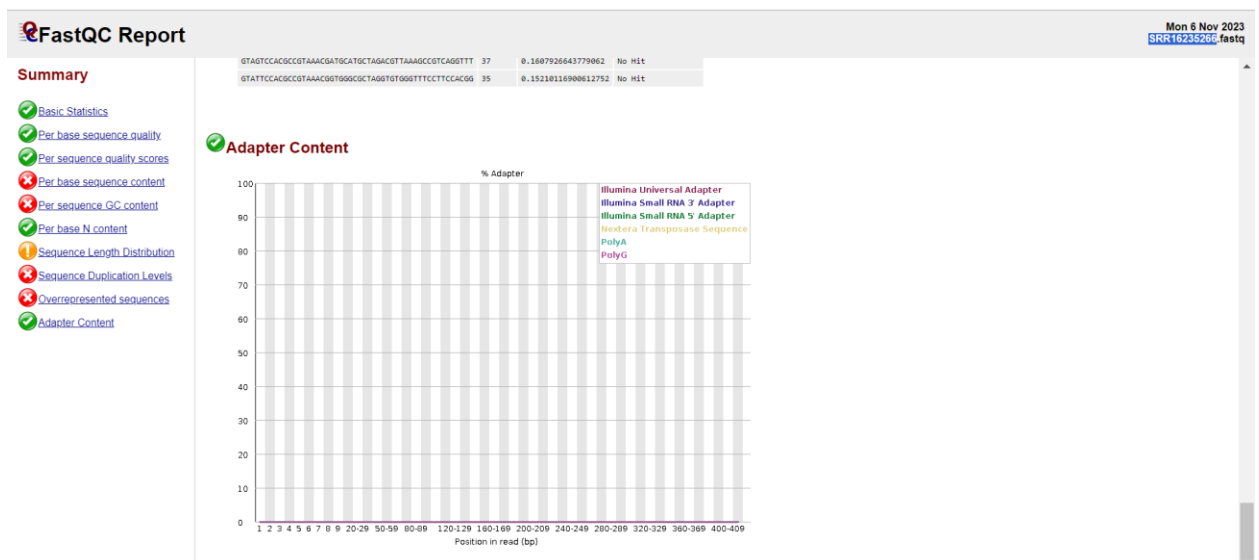
- viii. **Sequence Duplication Levels**-In a diverse library most sequences will occur only once in the final set. A low level of duplication may indicate a very high level of coverage of the target sequence, but a high level of duplication is more likely to indicate some kind of enrichment bias. This module **counts the degree of duplication for every sequence in the set** and creates a plot showing the relative number of sequences with different degrees of duplication.



- ix. **Overrepresented sequences**-A normal high-throughput library will contain a diverse set of sequences, with no individual sequence making up a tiny fraction of the whole. Finding that a **single sequence is very overrepresented in the set either means that it is highly biologically significant, or indicates that the library is contaminated, or not as diverse as you expected.**



- x. **Adapter Content**- Adapters are specific module or section that assesses the presence of adapter sequences in sequencing data. The adapter content analysis in FASTQC checks for the **presence of these adapter sequences in your sequencing data and evaluates how much of your data might be affected by adapter contamination**. This information is important for quality control and preprocessing steps in NGS data analysis, as adapter contamination can lead to issues such as reduced mapping efficiency, erroneous variant calls.



## ❖ Discussion

Explain errors

\*\*\*\*\*LENOVO LAPTOP\*\*\*\*\*

Fastqc file generate

- 1) Navigate to fastqc folder

- 2) Use prefetch to download .sra data from SRR ID
- 3) Then convert sra to fastq format by command  
**fasterq-dump SRR27218538/(can specify location of output)**
- 4) Fastq files will be generated for paired(2),single(1file)
- 5) We use paired end so command is  
**fastqc -t 2 SRR27218538\_1.fastq SRR27218538\_2.fastq -o output\_fastqc/**
  - -t specify paired end
  - -o output directory
  - 2- 2 files of fastq to generate only one report

**ERROR- 2 SEPARATE FASTQC REPORT ARE GENERATED**