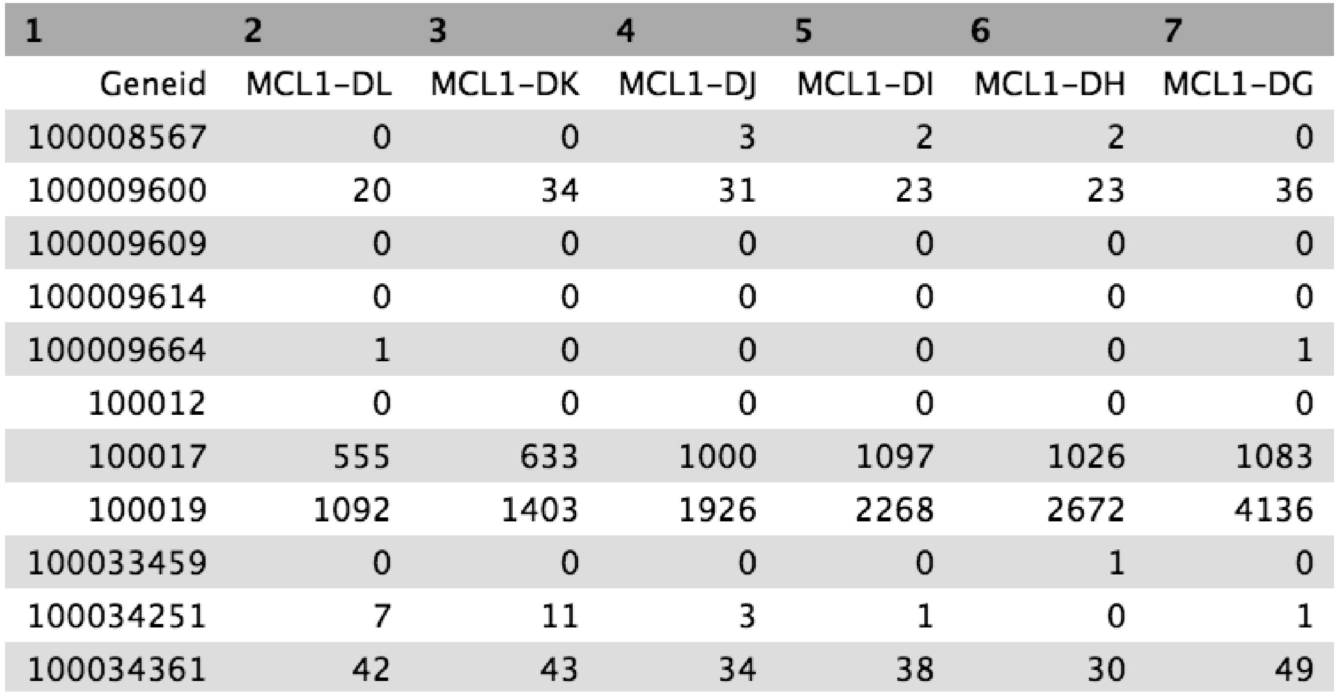
Preprocessing and Library Prep

Center for Health Data Science

Health Data Science Sandbox

The goal

Count matrix

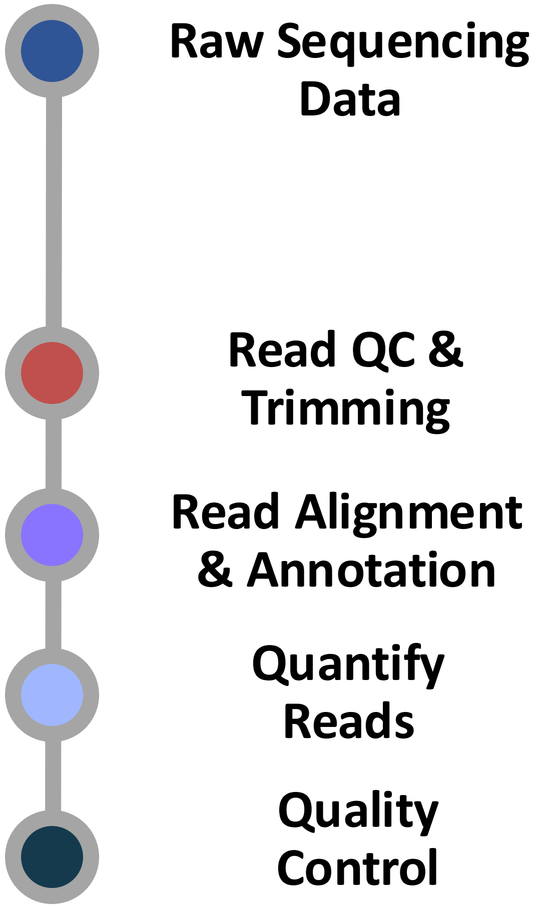


1	2	3	4	5	6	7
Geneid	MCL1-DL	MCL1-DK	MCL1-DJ	MCL1-DI	MCL1-DH	MCL1-DG
100008567	0	0	3	2	2	0
100009600	20	34	31	23	23	36
100009609	0	0	0	0	0	0
100009614	0	0	0	0	0	0
100009664	1	0	0	0	0	1
100012	0	0	0	0	0	0
100017	555	633	1000	1097	1026	1083
100019	1092	1403	1926	2268	2672	4136
100033459	0	0	0	0	1	0
100034251	7	11	3	1	0	1
100034361	42	43	34	38	30	49

genes OR isoforms

Overview

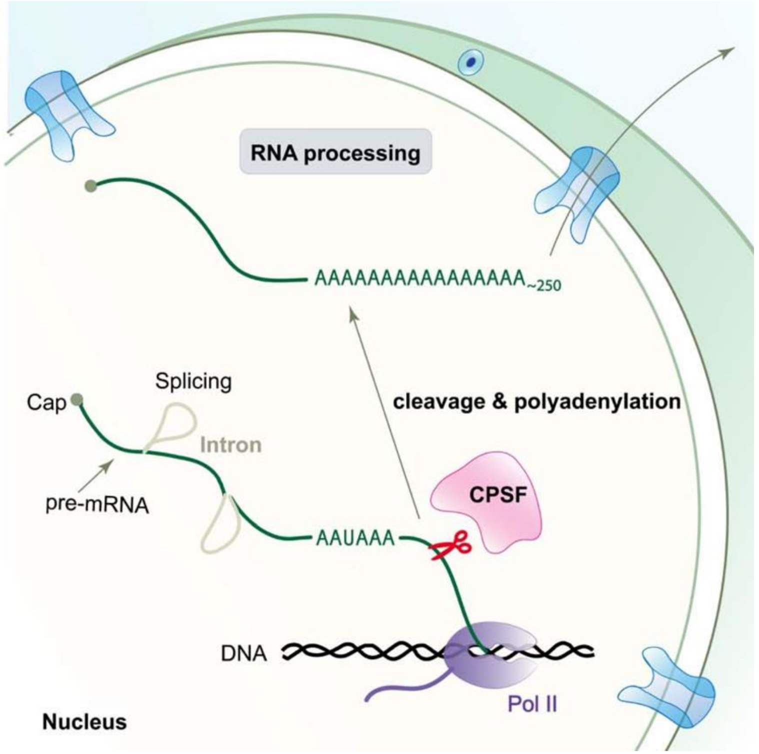
1. Library Prep
2. Sequencing Technology
3. File Formats
4. Quality Control & Trimming
5. Alignment & Annotation
6. Quantification
7. Quality Control



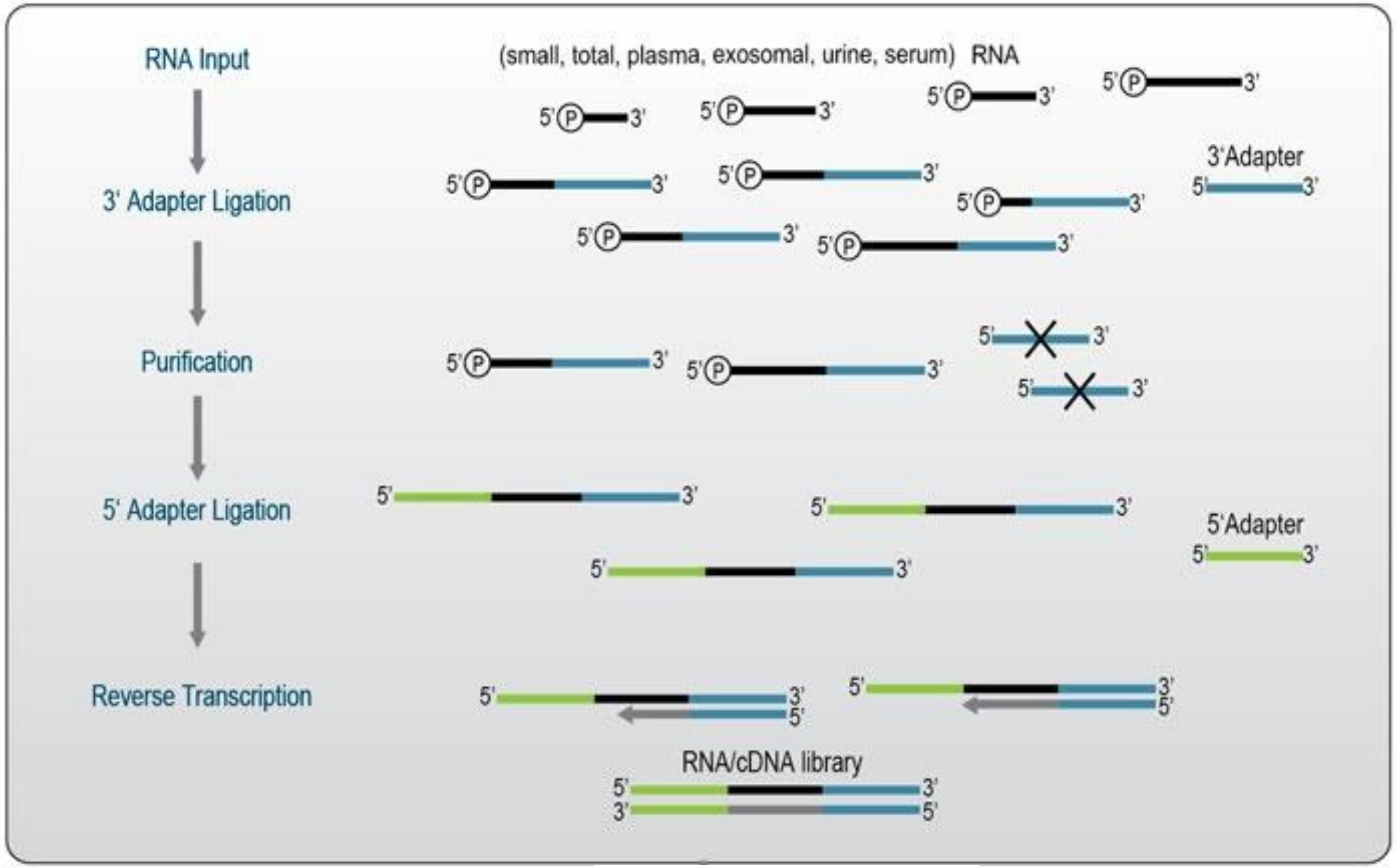
Preprocessing:

- Raw Sequencing Data
- Read QC & Trimming
- Read Alignment & Annotation
- Quantify Reads
- Quality Control

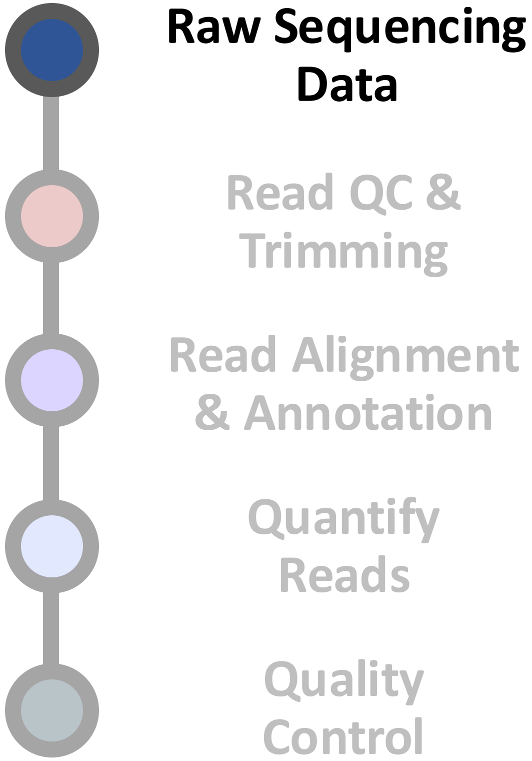
Library Prep



LIBRARY GENERATION



Preprocessing:



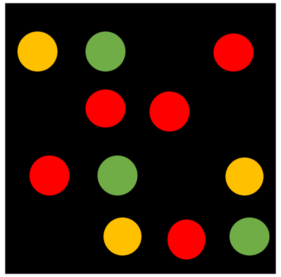
Raw Sequencing Data

Read QC & Trimming

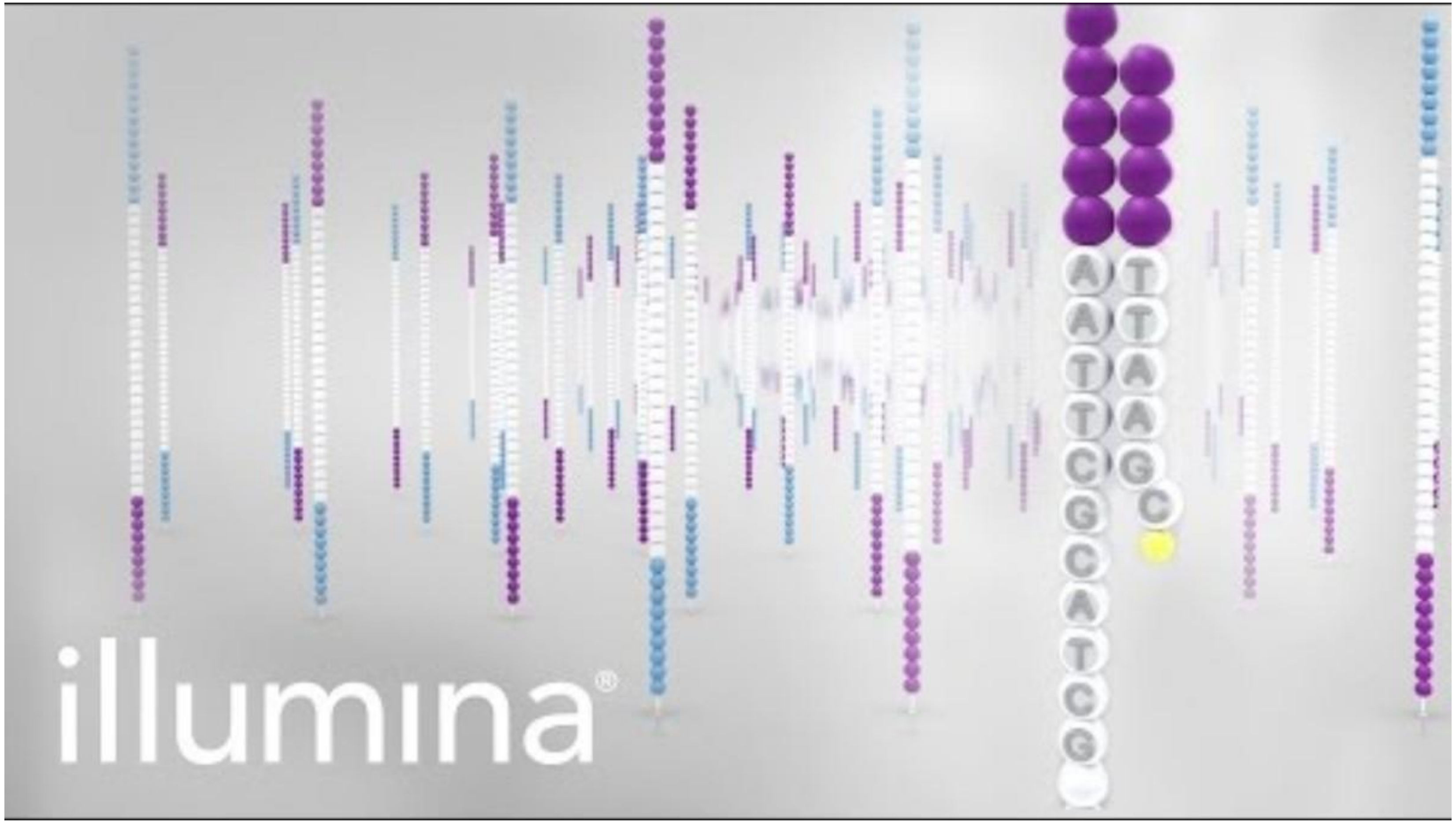
Read Alignment & Annotation

Quantify Reads

Quality Control

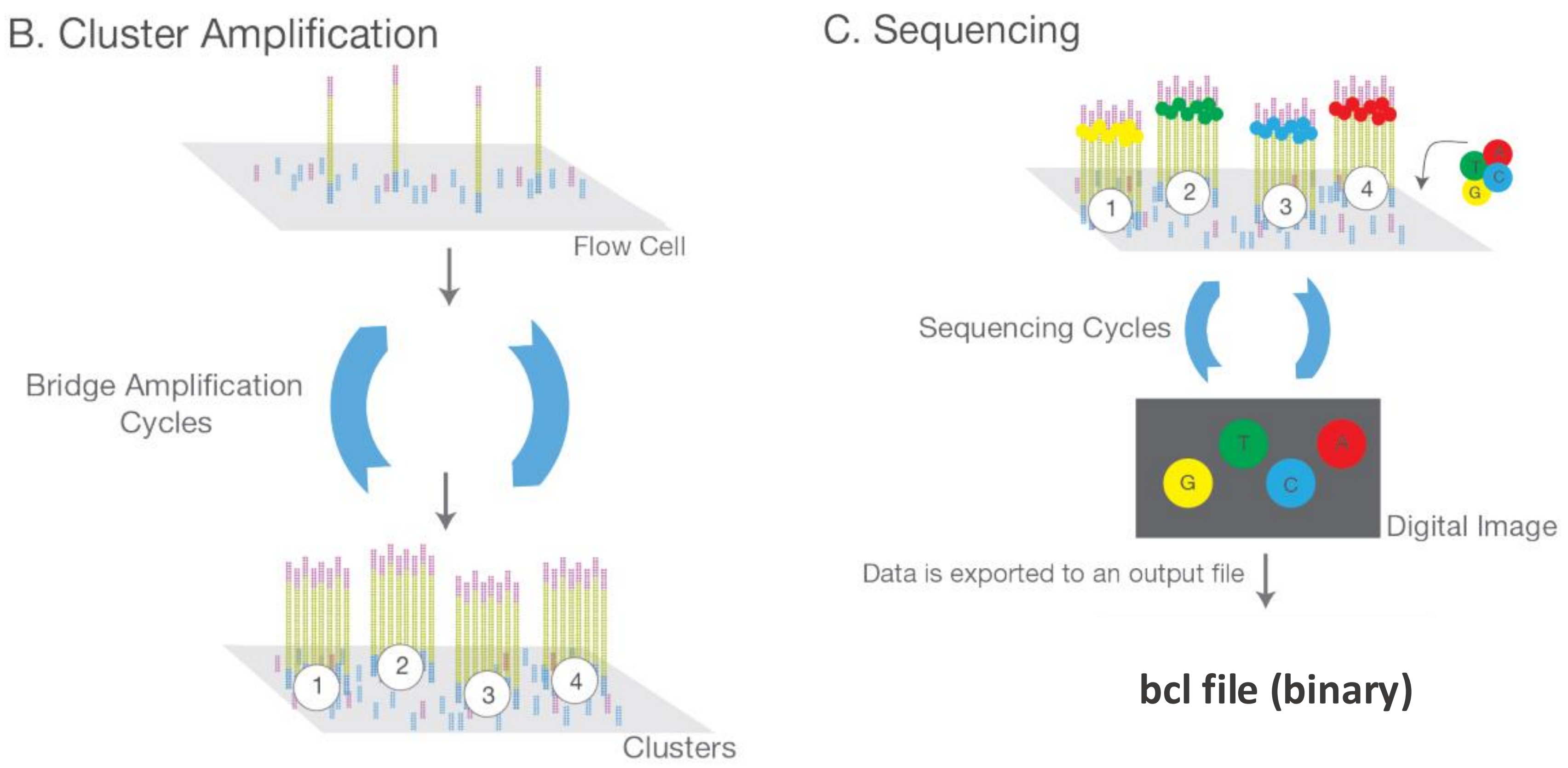


Sequencing by synthesis Real-Time Analysis



https://www.youtube.com/watch?v=fCd6B5HRaZ8

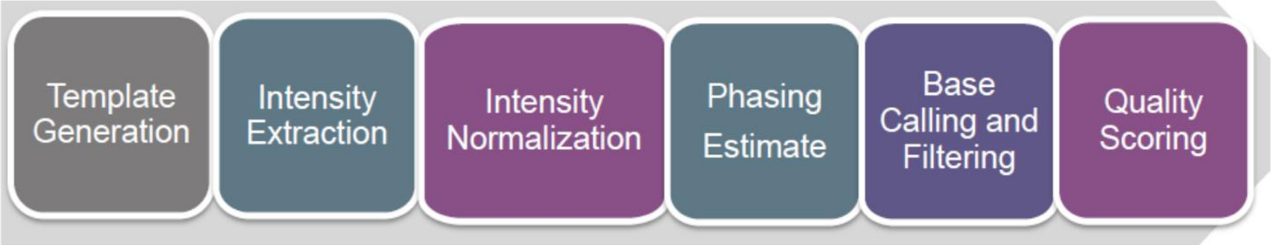
Sequencing by synthesis Real-Time Analysis

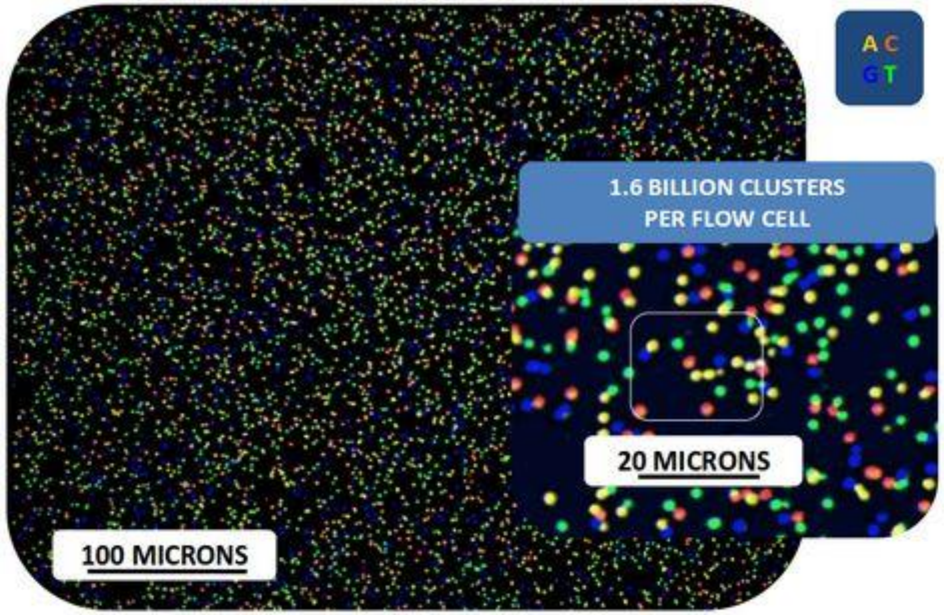


bcl file (binary)

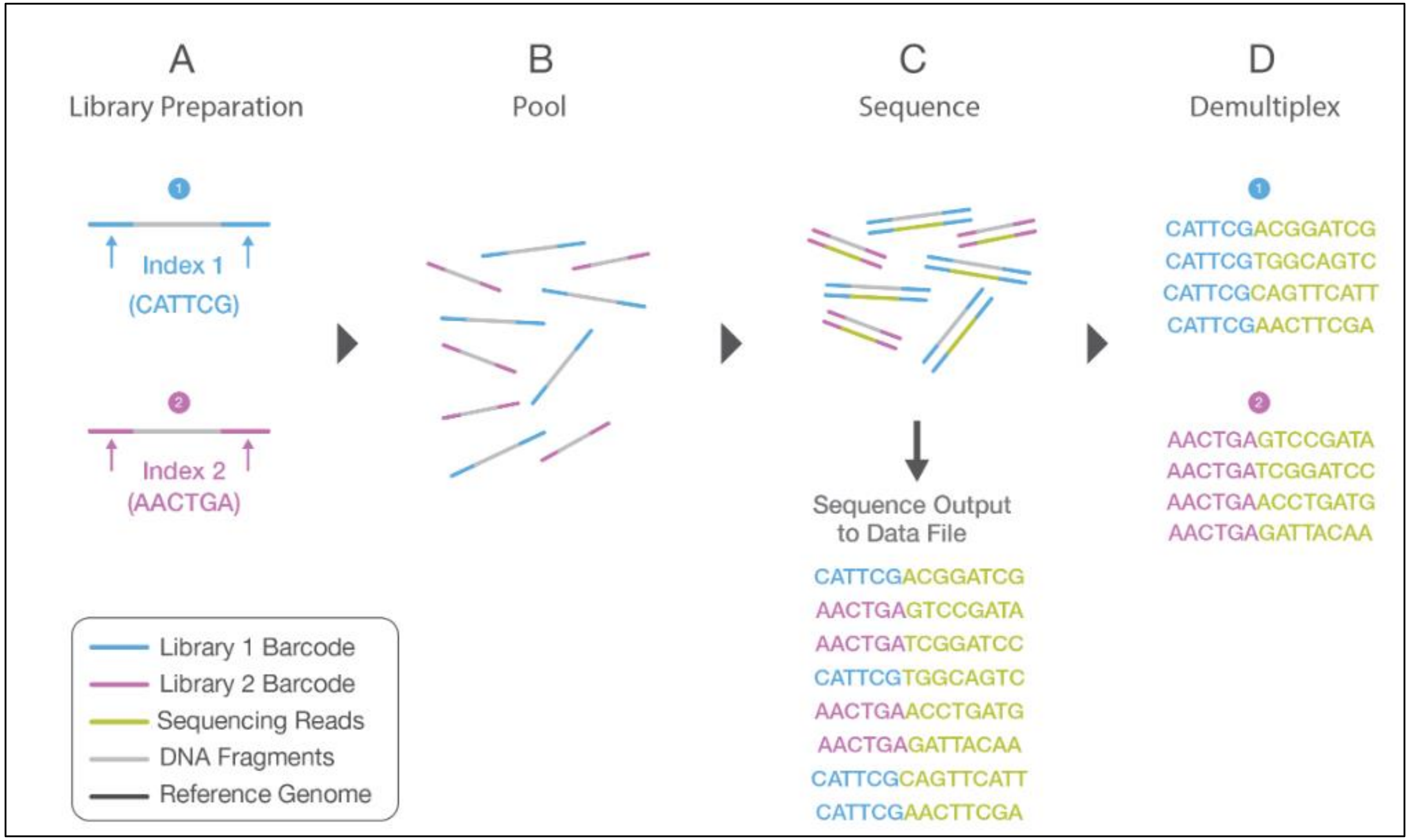
Sequencing by synthesis Real-Time Analysis

RTA stores data as binary base call or BCL files.





Demultiplexing



Convert bcl file to fastq

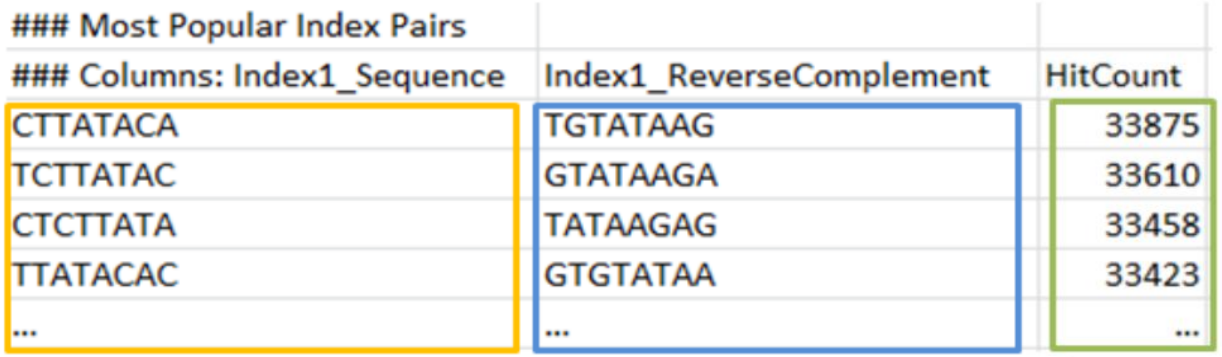
Steps

- Demultiplexes data - sequences are sorted according to their index/barcode sequence(s)
- Converts **BCL** to standard **FASTQ** file
- Adds ASCII Quality scores

Demultiplexing summary

Common causes for poor demultiplexing:

- Index sequences with **wrong** orientation in the sample sheet.
- **Incorrect index sequences** in the sample sheet.
- **Sample mix** ups between lanes.
- **Poor Index Read sequencing quality.**



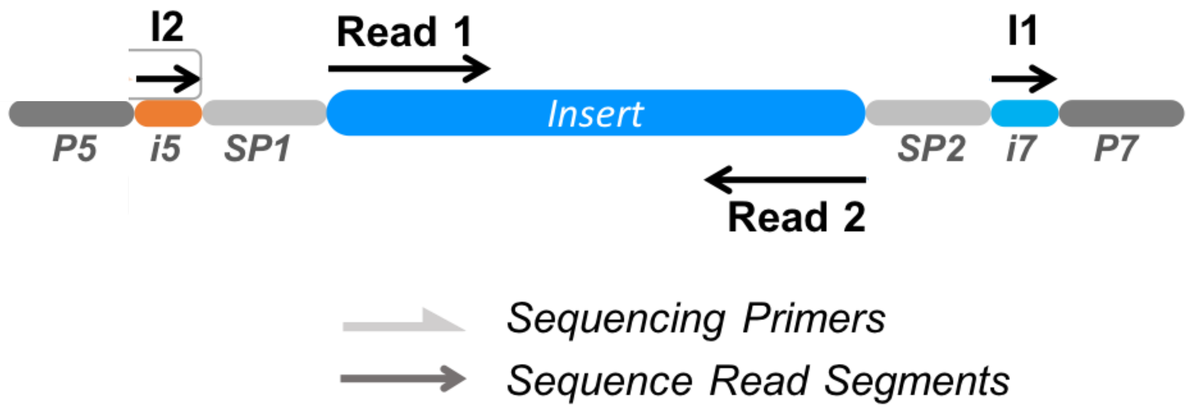
### Most Popular Index Pairs		
### Columns: Index1_Sequence	Index1_ReverseComplement	HitCount
CTTATACA	TGTATAAG	33875
TCTTATAC	GTATAAGA	33610
CTCTTATA	TATAAGAG	33458
TTATACAC	GTGTATAA	33423
...	...	...

↑ Index 1 sequences

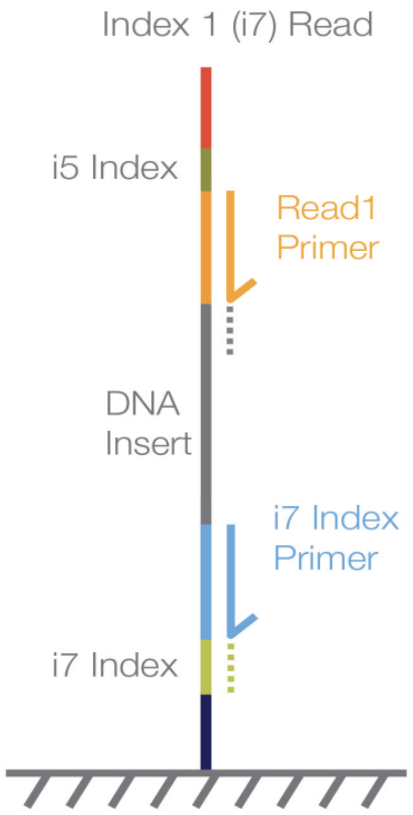
↑ Reverse complement of the index 1 sequences

↑ The number of reads with each pair of index sequences

Anatomy of an Illumina read



Paired-end dual index



P5	– attachment to flow cell oligo
P7	– attachment to the other flow cell oligo
i5	– index 1
i7	– index 2
SP1, SP2	– polymerase attachment sites

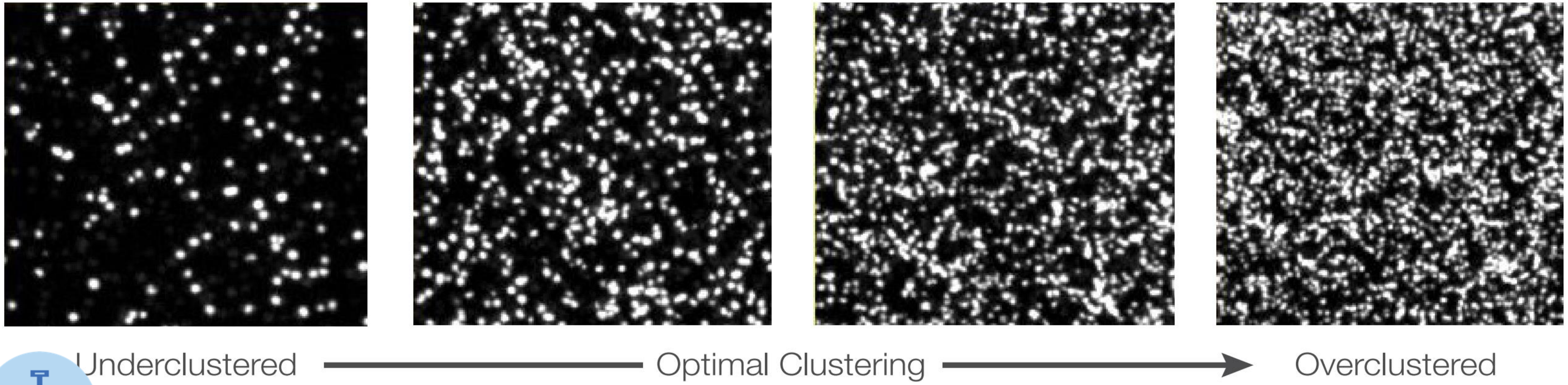
Cluster Density

Under clustering:
Maintains high data quality
Results in lower data output

Over clustering :
Poor run performance
Lower Q30 scores
Introduction of sequencing artifacts
<u>Lower total data output</u>

Underloaded flow cell

Overloaded flow cell,
too much input DNA!



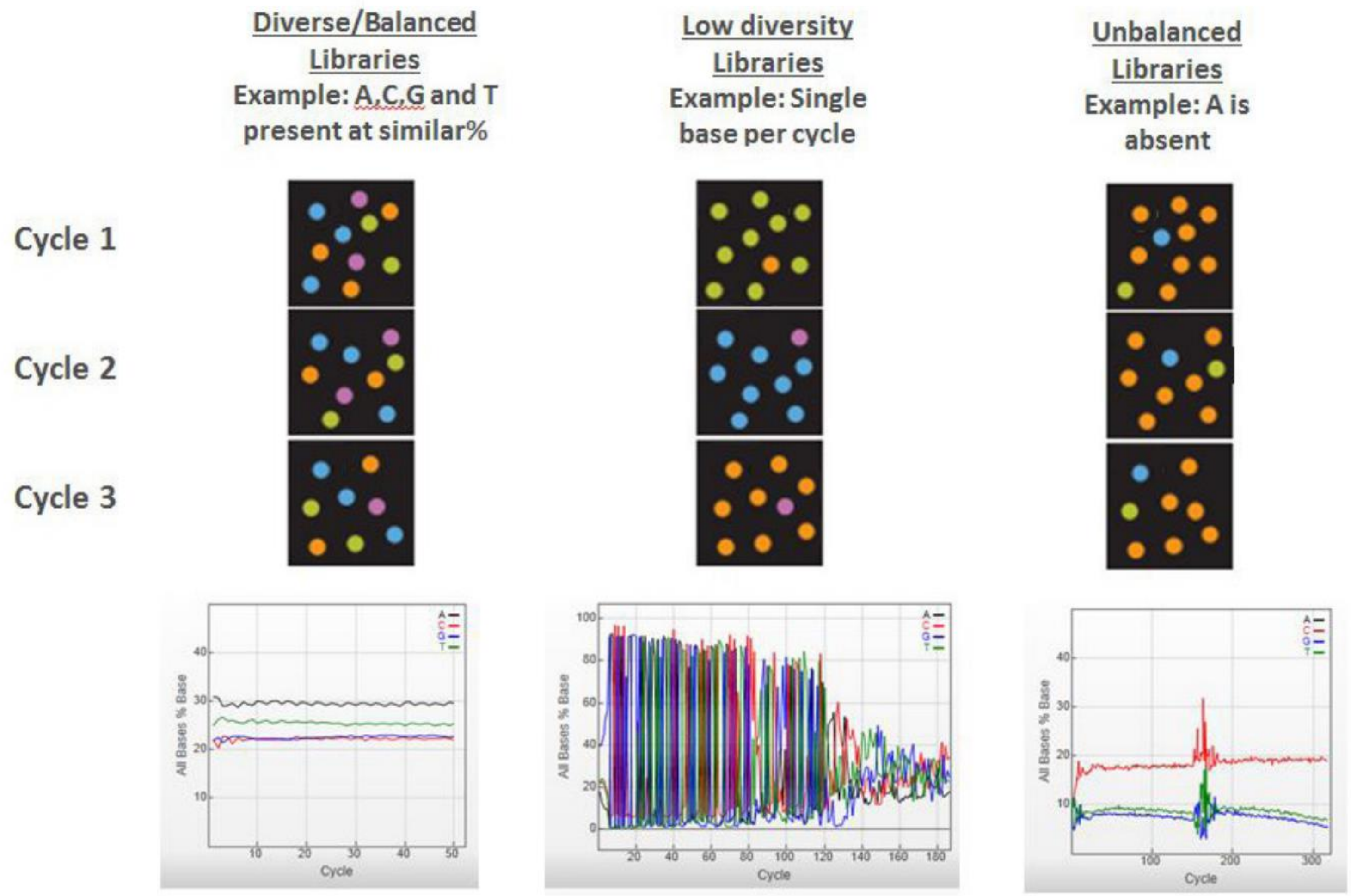
Nucleotide diversity

Nucleotide diversity = proportion of nucleotides A, C, G and T present in every cycle of the run.

Critical for optimal run performance and high-quality data generation:

Cluster identification and quality base calls.

For low diversity or unbalanced libraries we may need barcodes or spike-ins.



Diverse/Balanced Libraries
Example: A,C,G and T present at similar%

Low diversity Libraries
Example: Single base per cycle

Unbalanced Libraries
Example: A is absent

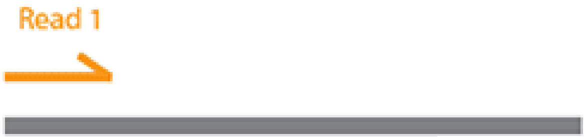
Cycle 1

Cycle 2

Cycle 3

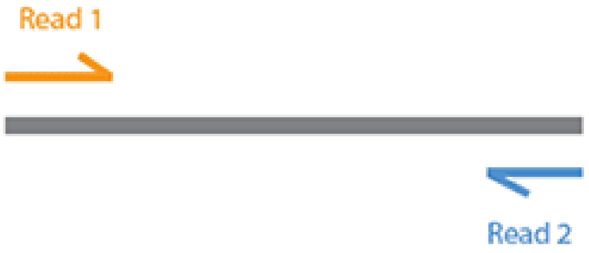
Single end VS paired end

Single end



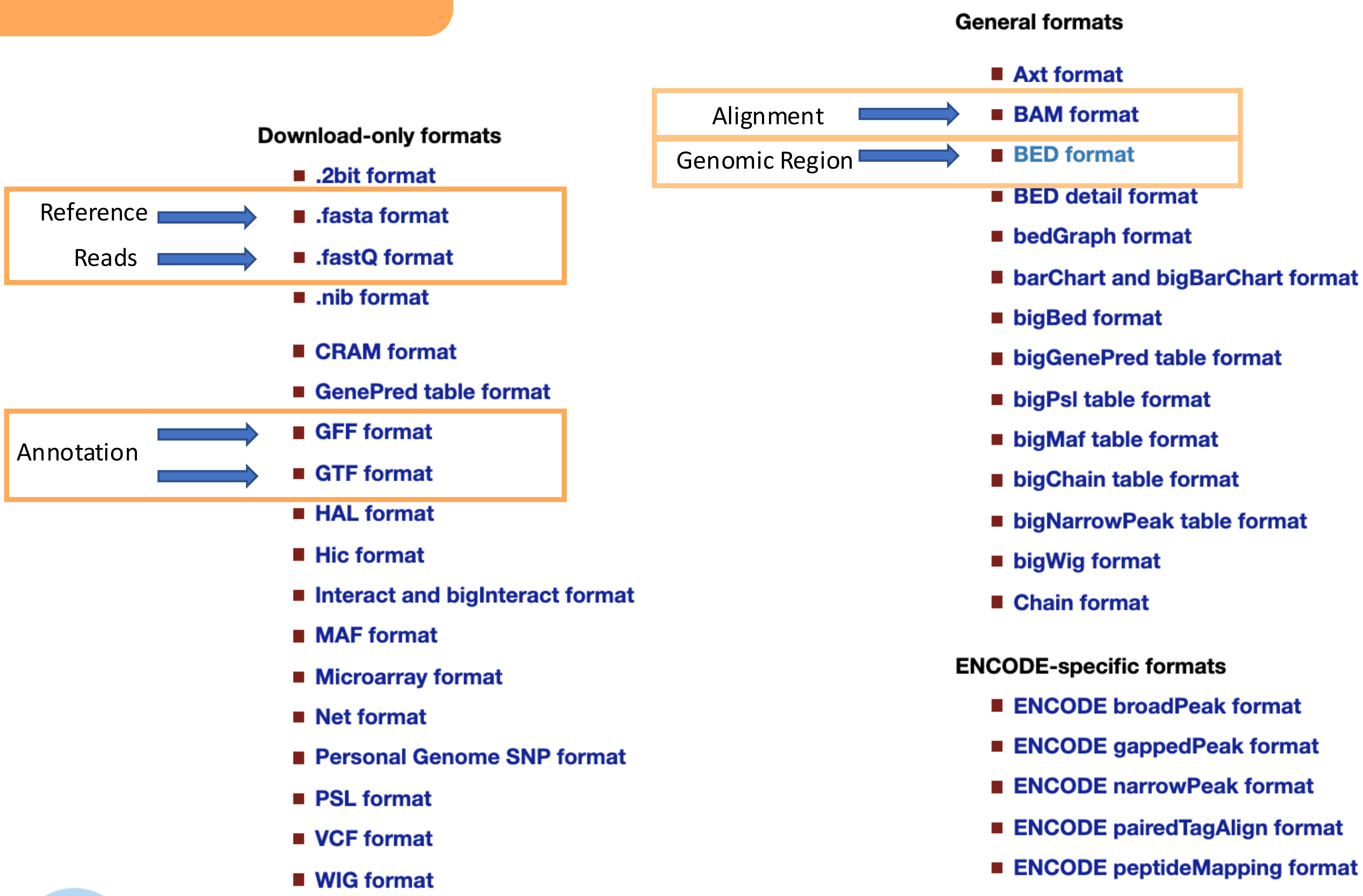
- More DNA molecules inspected (1 read = 1 molecule)

Paired end



- Longer reads (more information on each molecule)
- Higher accuracy if read 1 and read 2 overlap

File formats



Download-only formats

- .2bit format
- .fasta format
- .fastQ format
- .nib format

- CRAM format
- GenePred table format
- GFF format
- GTF format
- HAL format
- Hic format
- Interact and bigInteract format
- MAF format
- Microarray format
- Net format
- Personal Genome SNP format
- PSL format
- VCF format
- WIG format

General formats

- Axt format
- BAM format
- BED format
- BED detail format
- bedGraph format
- barChart and bigBarChart format
- bigBed format
- bigGenePred table format
- bigPsl table format
- bigMaf table format
- bigChain table format
- bigNarrowPeak table format
- bigWig format
- Chain format

ENCODE-specific formats

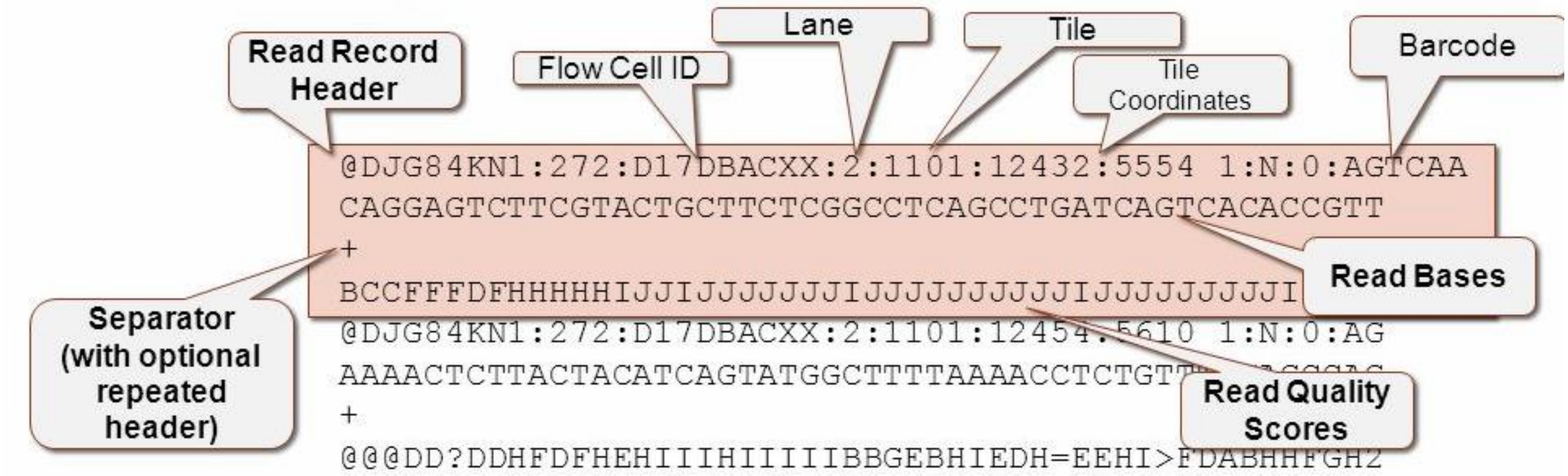
- ENCODE broadPeak format
- ENCODE gappedPeak format
- ENCODE narrowPeak format
- ENCODE pairedTagAlign format
- ENCODE peptideMapping format

FASTQ

Single-end sequencing = one FASTQ files per sample

Paired-end sequencing = two FASTQ files per sample **OR**

one merged FASTQ file per sample



```
@DJG84KN1:272:D17DBACXX:2:1101:12432:5554 1:N:0:AGTCAA
CAGGAGTCTTCGTACTGCTTCTCGGCCTCAGCCTGATCAGTCACACCGTT
+
BCCFFFDFHHHHHIJJIJJJJJJIJJJJJJJJJIJJJJJJJJI
@DJG84KN1:272:D17DBACXX:2:1101:12454:5610 1:N:0:AG
AAAACTCTTACTACATCAGTATGGCTTTTAAAACCTCTGTT
+
@@@DD?DDHFDFHEHIIIHIIIIIBBGEBHIEDH=EEHI>FDABHHFGH2
```

Quality Score

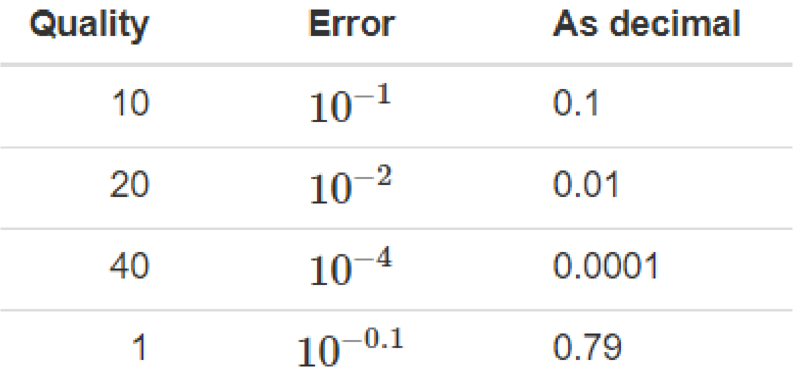
Quality score = the probability of an error in base calling **(probability of base being wrongly called).**

$$P(error) = 0.1$$

$$P(error) = 10^{-1}$$

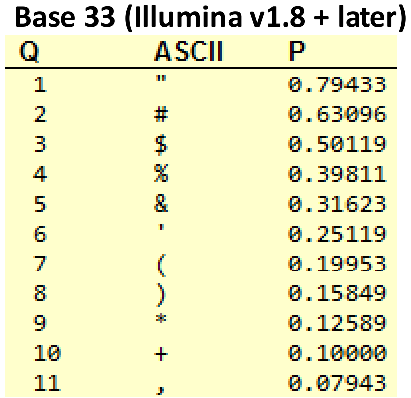
$$P(error) = 10^{-\frac{Q}{10}}$$

→ Use Q to describe the error:



Quality	Error	As decimal
10	10^{-1}	0.1
20	10^{-2}	0.01
40	10^{-4}	0.0001
1	$10^{-0.1}$	0.79

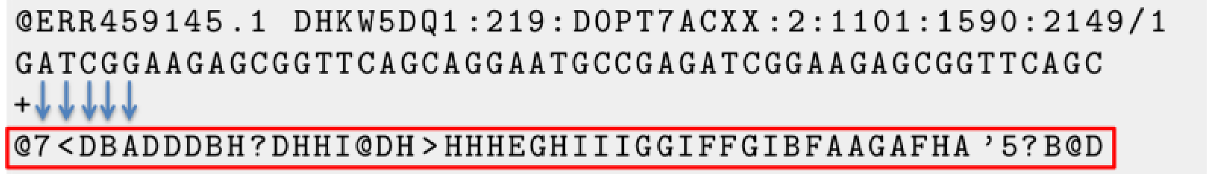
→ Transcribe Q into ASCII:



Base 33 (Illumina v1.8 + later)

Q	ASCII	P
1	"	0.79433
2	#	0.63096
3	$	0.50119
4	%	0.39811
5	&	0.31623
6	'	0.25119
7	(	0.19953
8	)	0.15849
9	*	0.12589
10	+	0.10000
11	,	0.07943

Write in fastq file:



```
@ERR459145.1 DHKW5DQ1:219:D0PT7ACXX:2:1101:1590:2149/1
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGATCGGAAGAGCGGTTCAGC
+
@7<DBADDDBH?DHHI@DH>HHHEGHIIIGGIFFGIBFAAGAFHA'5?B@D
```

We use this score to filter low-quality reads that contain bases with a high probability of being wrong.

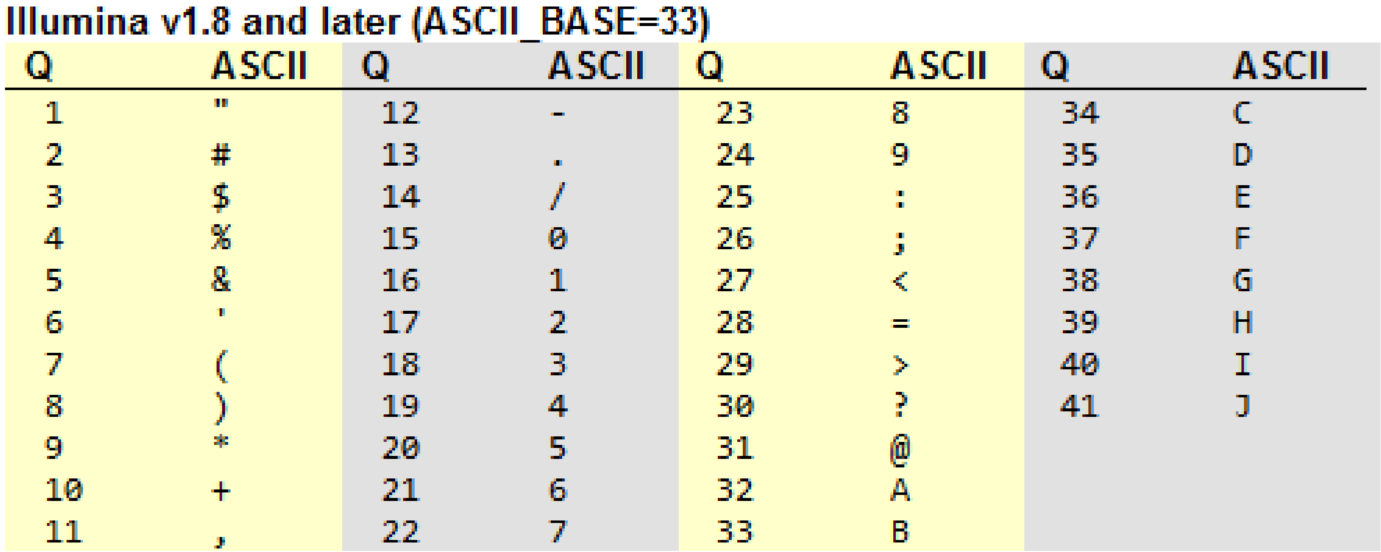
Exercise

Calculate the error probability of the following short read:

$$P(error) = 10^{\frac{-Q}{10}}$$

@SRR4420293.3
ATTCGCAGATC
+
=B@FFHH67<?

As well as the average quality score of the whole read.



Illumina v1.8 and later (ASCII_BASE=33)

Q	ASCII	Q	ASCII	Q	ASCII	Q	ASCII
1	"	12	-	23	8	34	C
2	#	13	.	24	9	35	D
3	$	14	/	25	:	36	E
4	%	15	0	26	;	37	F
5	&	16	1	27	<	38	G
6	'	17	2	28	=	39	H
7	(	18	3	29	>	40	I
8	)	19	4	30	?	41	J
9	*	20	5	31	@		
10	+	21	6	32	A		
11	,	22	7	33	B		

Preprocessing:

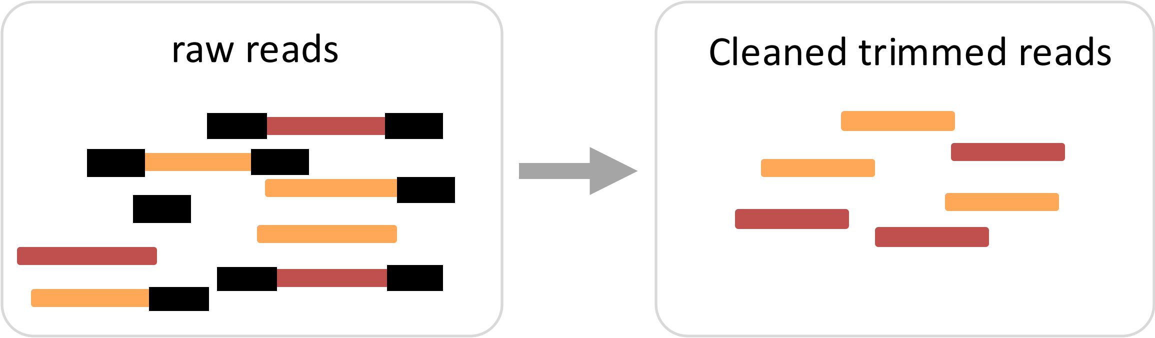
Raw Sequencing Data

Read QC & Trimming

Read Alignment & Annotation

Quantify Reads

Quality Control



Quality Control - FASTQC

What is the point of QC?

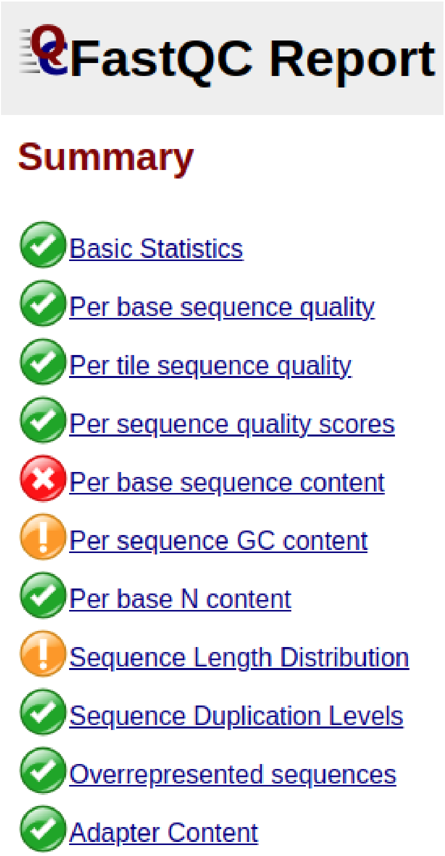
- Technical errors will not cause pipelines to fail
- Technical errors will still generate hits
- Technical hits often look biologically real
- Unexpected (interesting) artefacts are missed

Quality Control checks of raw sequencing data

- An impression of whether your data has any technical problems
- QC saves you time, effort and money!

```
@M04743:199:000000000-CGG4F:1:1101:16145:1655 1:N:0:233
GGTGCCAGCCGCCGCGGTAATACGAAGGTGGCAAGCGTTGTTCGGATTCACTGGGCGTACAGGGAGCGTAGGCGGTTGGGTAAGCC
+
ABCCCFFFCADBGGGGGGGGGGHHGHGGFHGHHHGHGGGAFFHGGGGHHHHHHHHGGGGHHGGGGGGGGHGGEGGGGHHHHH
@M04743:199:000000000-CGG4F:1:1101:18938:1729 1:N:0:233
GGTGCCAGCCGCCGCGGTAATACGTAGGGTGCGAGCGTTAATCGGAATTACTGGGCGTAAAGCGTGCGCAGGCTGTTTTGTAAGTC
+
BBBBBFFFBBBBGGGGGGGGGFHHHHHGGHGGGGGGGGGGGHHGGEGFHHHHHHHHGGGGHFHGGGGGGGGGHHHHHHHHHHHH
@M04743:199:000000000-CGG4F:1:1101:13893:1760 1:N:0:233
GGTGCCAGCAGCCGCGGTACTACGTAGGGTGCGAGCGTTGTCCGGAATTACTGGGCGTAAAGAGTTCGTAGGCGGTTTGTCGCGTC
+
BBBBBFFFB4CCGGGGGGGCFFHGHHHGGHGGGGGGGGGAFGHGG?EFHFEHHHHHGGGGFHFHFGHGGHGG3EEEGGGHHEHGGG
F9FFFFFFFFFFFFEFFBBBBFEB;-@DFB-BBBFFFFEFF/EBBEFFF/BADFFDFFF.;
@M04743:199:000000000-CGG4F:1:1101:14830:1795 1:N:0:233
GGTGCCAGCCGCCGCGGTAATACGTAGGTGGCAAGCGTTGTCCGGATTTATTGGGTTTAAAGGGTGCGTAGGCGGTTCTTTAAGTC
+
ABBABFBFB?AAEE?EGEFCGGHHFFHGEHFFHHGHGGGCFHHGEEGGDFGDHHHGGGFGDGHGGFEGFGGDFGGGGGHHFFFBGF
9BD?99-9/9@-BD.;ADFFFBF///BBF:FFFFFFED?DFDFF?A.
@M04743:199:000000000-CGG4F:1:1101:14968:1984 1:N:0:233
AGTGCCAGCCGCCGCGGTAATACGTAGGTGGCAAGCGTTGTCCGGATTTATTGGGTTTAAAGGGTGCGTAGGCGGTTCTTTAAGTC
+
BBBBBFFFBABBGGGGGGGGGGHHGFHGHHGHHHGHGGGCFHHGGEGGHHHHHHGGGGHHHHHGHGGGGGGGHGGGGGHHHHHHHH
FCHHHGGHHHHHHHHHHHHHHHHHHHHHHHFHHHHGFHHGEGGFHHGHGGGFEGG9FGGAEGGGGAFDGEFFGGFFFBFEFFFFFFFFF
@M04743:199:000000000-CGG4F:1:1101:12706:2099 1:N:0:233
TGTGCCAGCCGCCGCGGTAATACGGAGGGAGCTAGCGTTGTTCGGAATTACTGGGCGTAAAGCGCACGTAGGCGGTTTTTTAAGTC
+
BCCCCFFFCCCCGGGGGGGGGGHHEGGGGDFGGHHHGGGGGHHGGGGFHHGHHHHHGGGGHHHGGGGGHHHGHGGGGGGGGACGHH
BFFFFFFFF9FFFFFFFFFFFFFFF/
@M04743:199:000000000-CGG4F:1:1101:13747:2260 1:N:0:233
CGTGCCAGCCGCCGCGGTAATACGAAGGGGGCTAGCGTTGTTCGGAATTACTGGGCGTAAAGAGTTCGTAGGCGGTTTGTCGCGTC
+
CCCCCCFFCABCGGGGGGGGGGHHFCEGDGGGGHHHGGGEFHHGGGFFHHFHHHHGGGGGHH@GHHHGGHGGHGGGGGGFH</>CF
A@@FFFFFFFFFFFFFBF9C;=CF.@;CDFFFFFBDFFFFFF?BEFFFFFFFFFFFFFFFFFF?
@M04743:199:000000000-CGG4F:1:1101:20151:2263 1:N:0:233
TGTGCCAGCCGCCGCGGTAATACGTAGGGTGCGAGCGTTAATCGGAATTACTGGGCGTAAAGCGTGCGCAGGCTGTTTTGTAAGTC
+
BBBBBFFFBAADGGGGGGGGGGHHHHHGGHGGGGGGGGGGGHHGGDFFHHHHHHHGGGGGHGHGGGGGGGGGHHHHHHHHHHHH
@M04743:199:000000000-CGG4F:1:1101:17232:2363 1:N:0:233
GGTGCCAGCCGCCGCGGTAATACGGAGGGGGCTAGCGTTGTTCGGAATTACTGGGCGTAAAGCGCACGTAGGCGGATCGGAAAGTC
+
BBBBBFFFBBBBGGGGGGGGGGHHGDGGGGGGGHHHGGG0FGHGGEGFHHHHHHHHGGGGHHHGGGGGHHHGHGGGGGGGGGHHH
```

Quality Control - FASTQC



FastQC Report

Summary

- ✓ Basic Statistics
- ✓ Per base sequence quality
- ✓ Per tile 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

- Failed modules are *not always* a problem, it depends on what you have sequenced and your protocol.

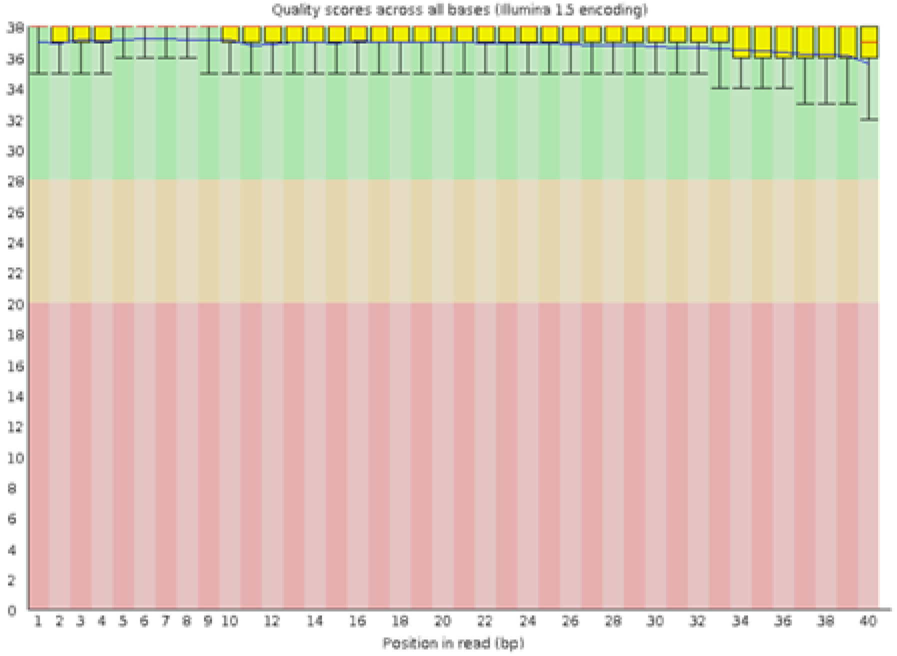
- Documentation has info on
  - what each module checks
  - what will cause it to fail
  - common explanations for fails

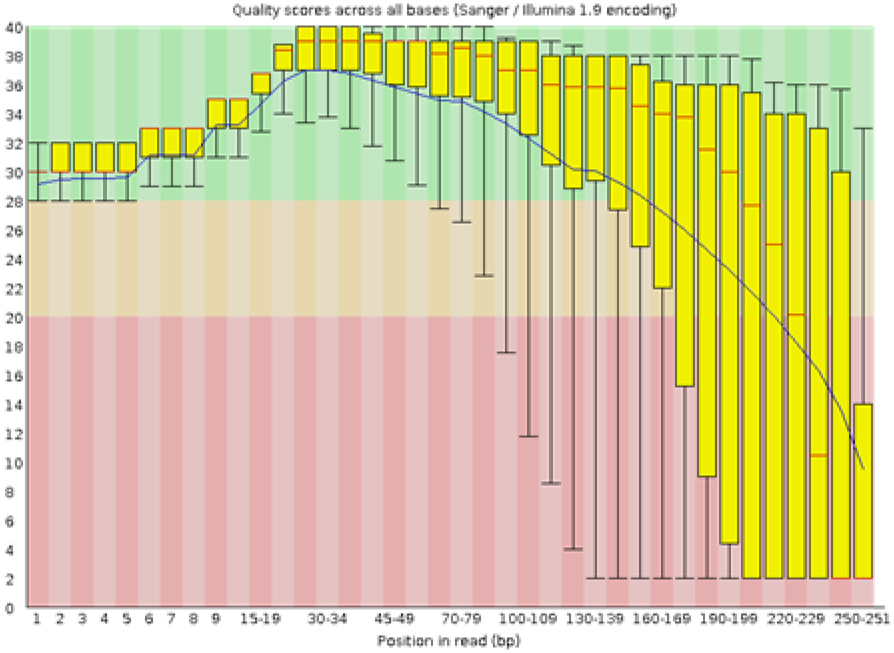
http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

QC - Per Base Quality Score

General degradation of quality over the duration of long runs.

- Sequencing chemistry issues results in accumulation of errors with increasing read length / long runs.
- Short loss of quality mid sequence can be due to bubbles in the flow cell, ect(consider base masking, sub with N).

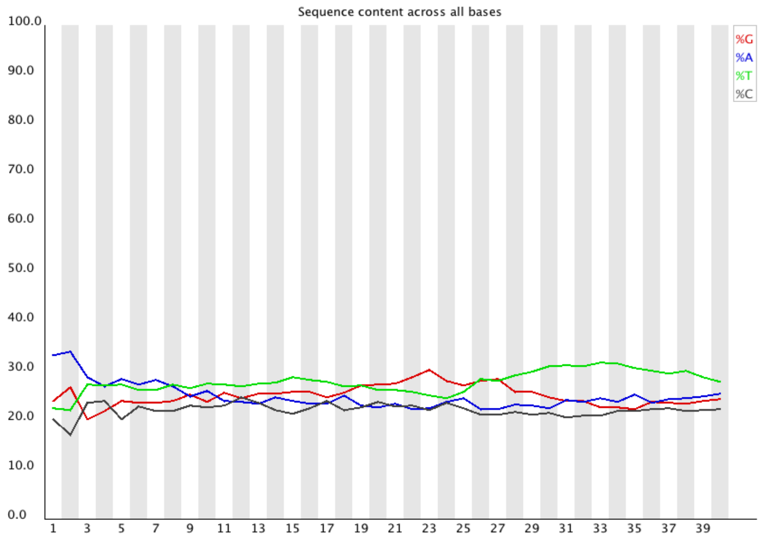


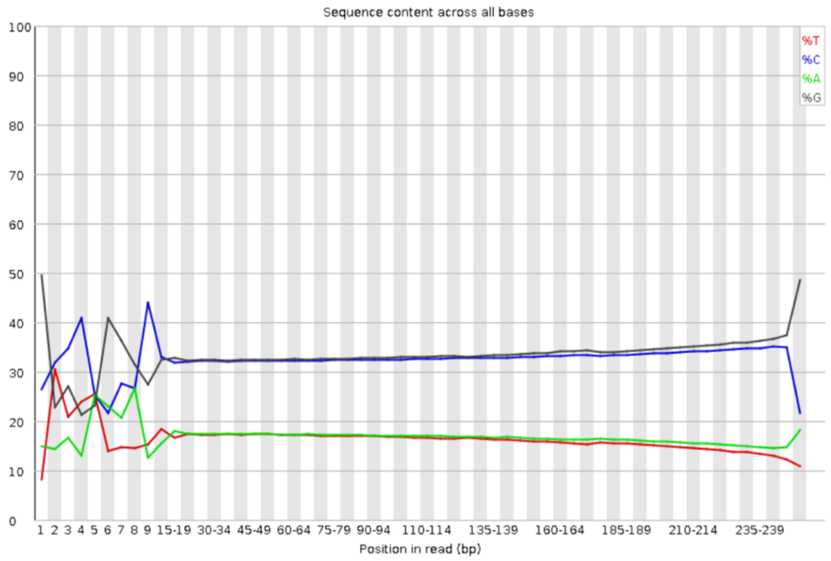


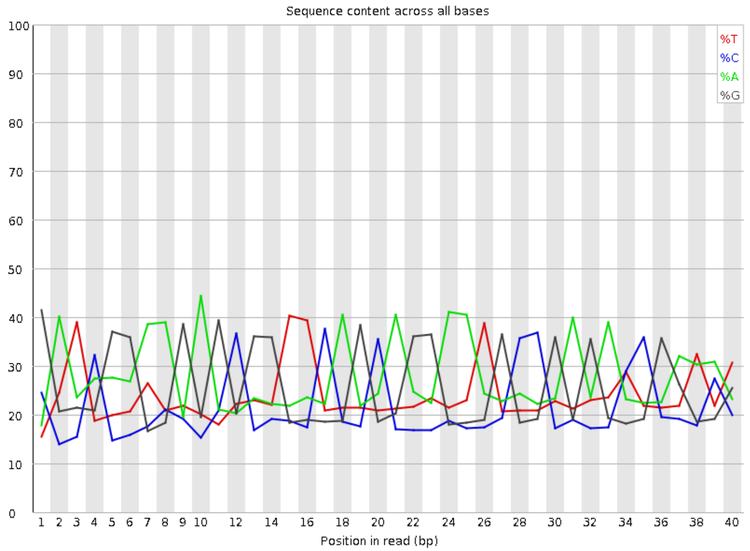
QC - Per Base Sequence Content

Ideally, random library - Some libraries are inherently biased in their sequence composition.

- GC content of the species (GC vs AT)
- Systematic biases per library type (primer, tagmentation)
- 3' RNA Sequence (PolyA)
- Overrepresented sequences (adapter dimers, rRNA)



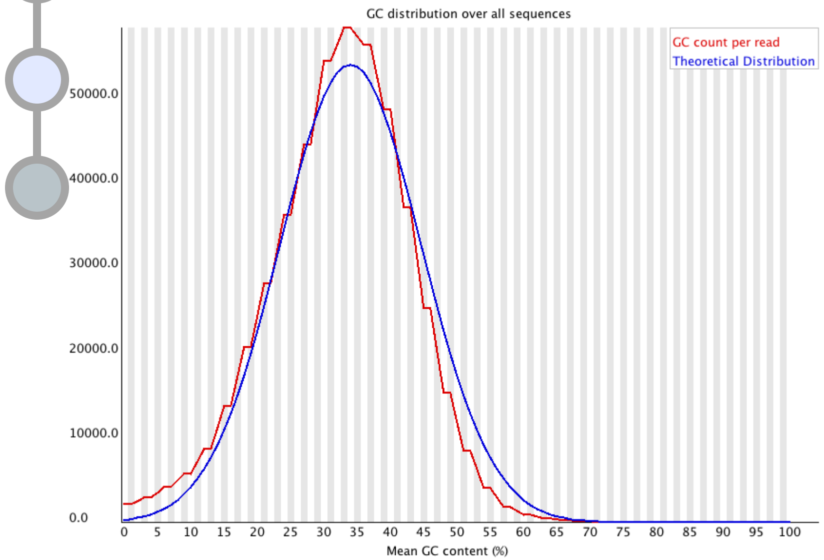




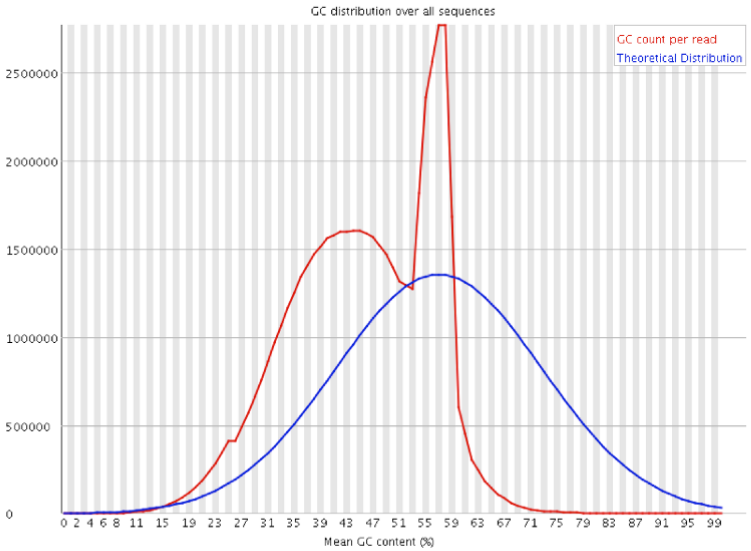
QC - Per Sequence GC Content

An unusual distribution could indicate a contaminated library:

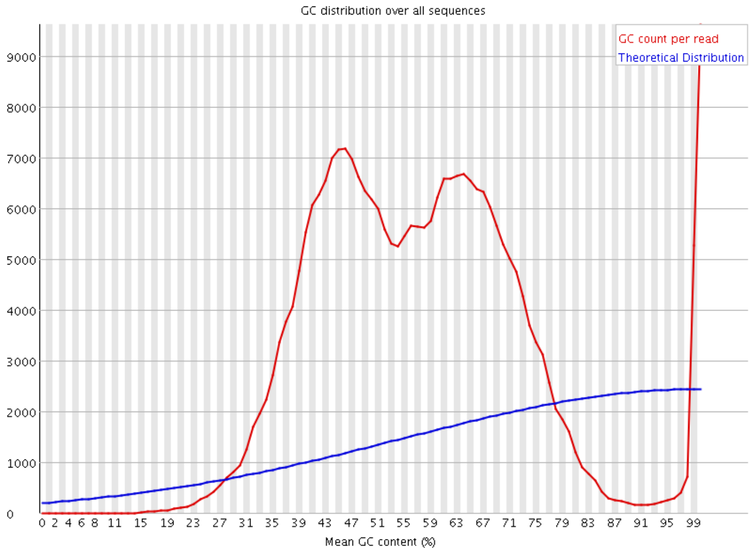
- Sharp peak on a smooth distribution may be due to a specific contaminant (ex. adapter dimers).
- Broader peaks may represent contamination with a different species.



No Contamination



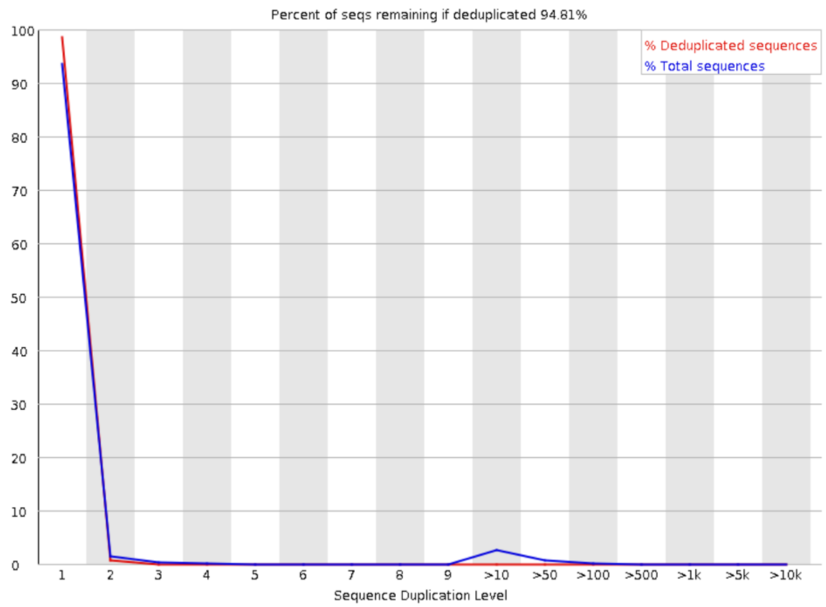
Specific Contamination

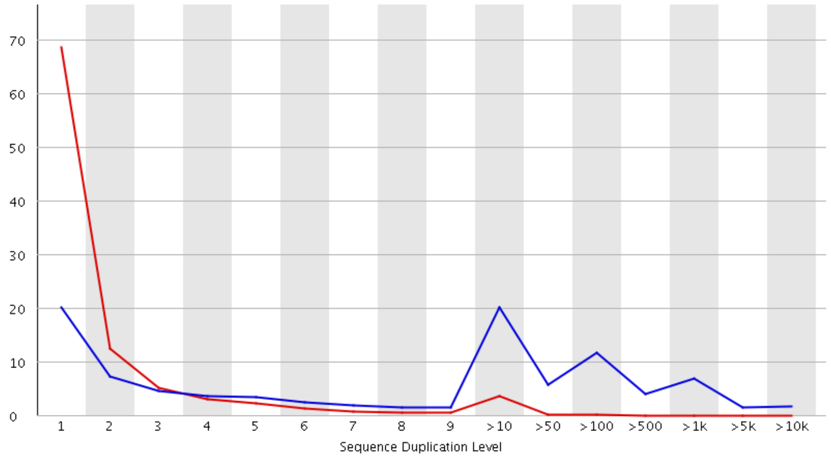


Broad Contamination

QC - Duplicate Sequences

- In a diverse library, most sequences should occur only once in the final set
- **Types** of duplicates in a library:
  - **technical duplicates:** PCR artefacts / overamplification
  - **biological duplicates:** different copies of exactly the same sequence are randomly selected.
  - **optical duplicates**, same DNA cluster erroneously reported as separate clusters
- No way to distinguish between these types



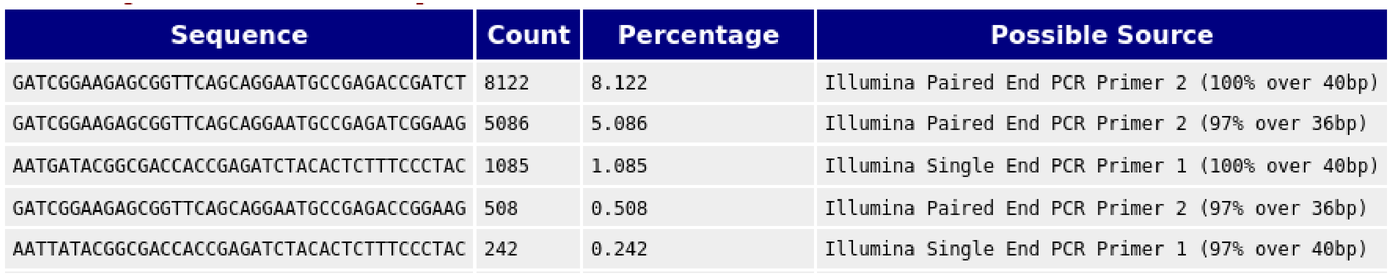


QC - Overrepresented Sequences

This module lists all of the sequences which make up more than 0.1% of the total.

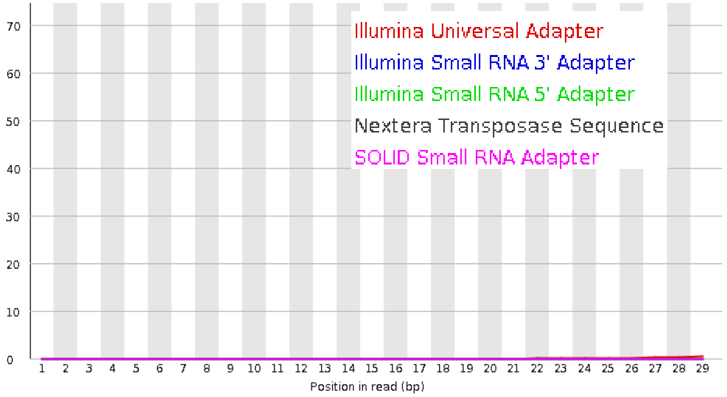
Comes from:

- Primers
- Adapter dimers
- Adapter read-though
- PolyA: Common in RNA-seq
- PolyN: Quality too poor

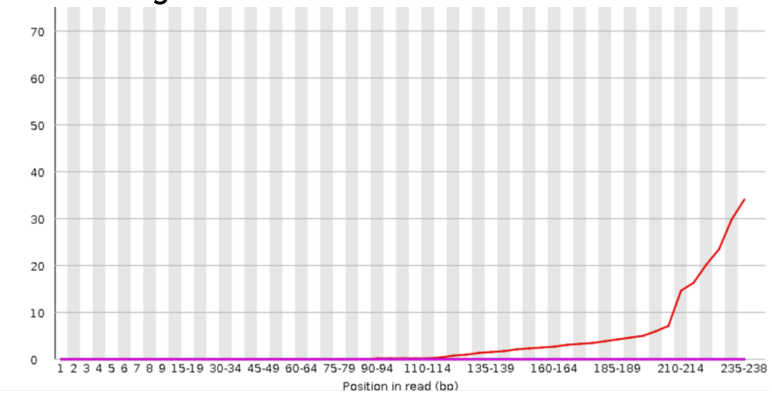


Sequence	Count	Percentage	Possible Source
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGATCT	8122	8.122	Illumina Paired End PCR Primer 2 (100% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGATCGGAAG	5086	5.086	Illumina Paired End PCR Primer 2 (97% over 36bp)
AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTAC	1085	1.085	Illumina Single End PCR Primer 1 (100% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGAAG	508	0.508	Illumina Paired End PCR Primer 2 (97% over 36bp)
AATTATACGGCGACCACCGAGATCTACACTCTTTCCCTAC	242	0.242	Illumina Single End PCR Primer 1 (97% over 40bp)

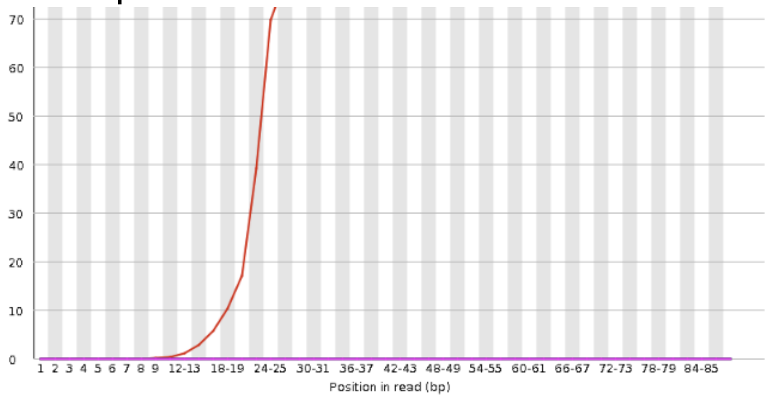
Ideal



Okay

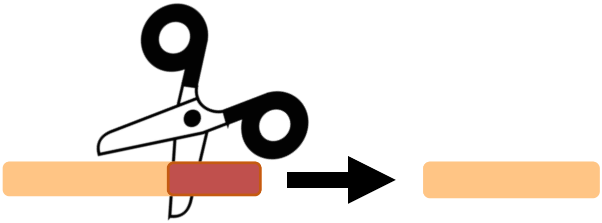


Nope!

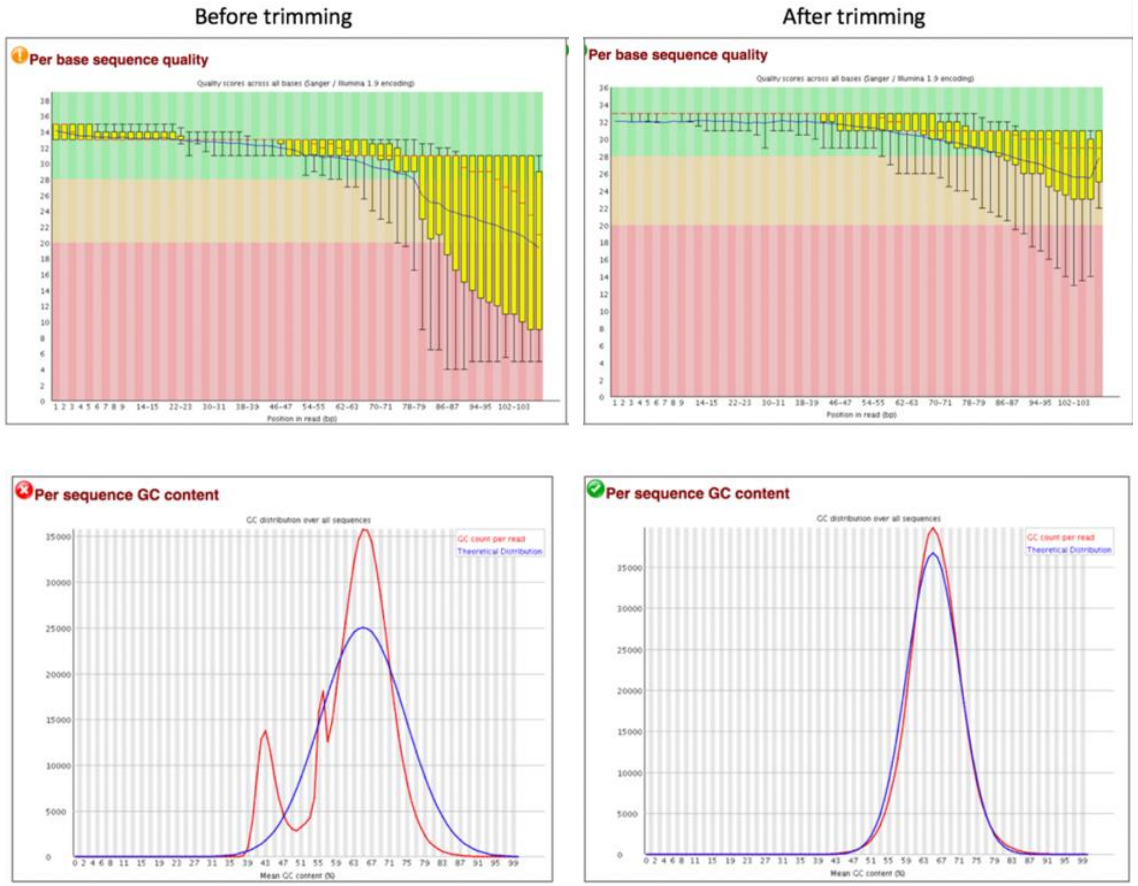


Quality trimming

- Remedy for poor read quality is trimming:
  - Adapter and primer seq. are removed
  - Reads with low-quality bases are truncated based on quality measure
  - End trimming - clip bases from the ends till quality threshold is met
- Based on: sliding window (N bases) moving average
- Short Reads are removed <25/<30 nt



Tools
CutAdapt
TrimGalore
Trimmomatic


Before trimming

After trimming

Exercise

On the course website we have put up two fastQC reports:
https://hds-sandbox.github.io/bulk_RNAseq_course/develop/workshop_RNAseq_nov2024.html
(download QC files button)

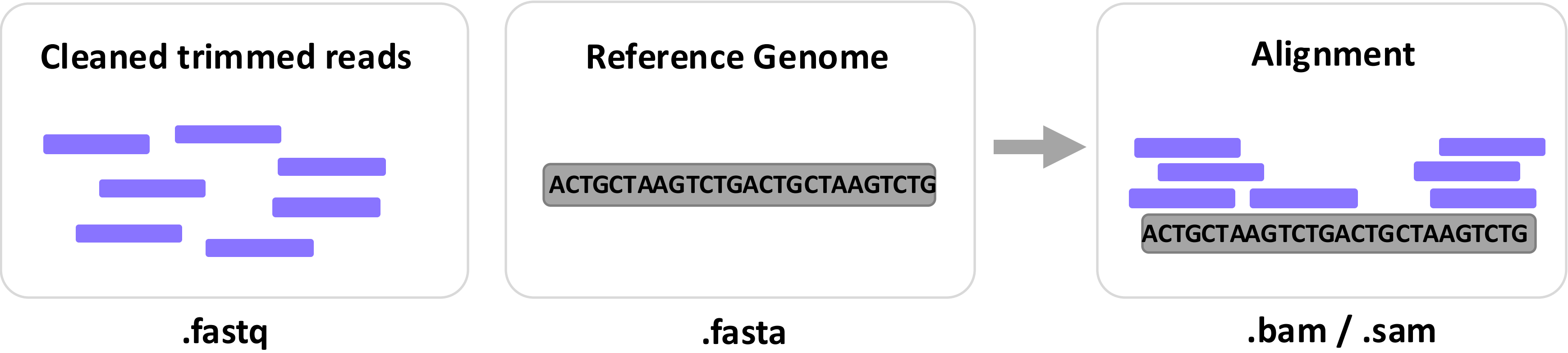
ERR430993_1_fastqc.html
small_rna_fastqc.html

Download them and discuss in the group at your table:

- Do you see any issues with this data?
- What could you do to clean the data?

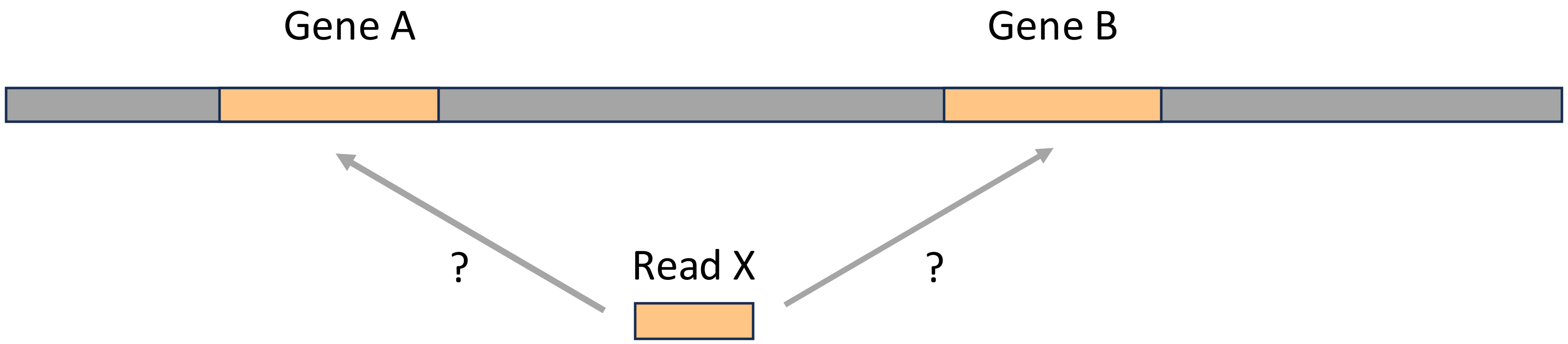
Preprocessing:

- Raw Sequencing Data
- Read QC & Trimming
- **Read Alignment & Annotation**
- Quantify Reads
- Quality Control



Alignment - Traditional mapping

Mapping/alignment is the process of figuring out the most likely origin of a read in a reference genome:



Alignment - Traditional mapping

This is not straightforward:

1. **search space problem**
   - millions/ billions of reads VS large reference (the human genome is 3.2 billion bases)

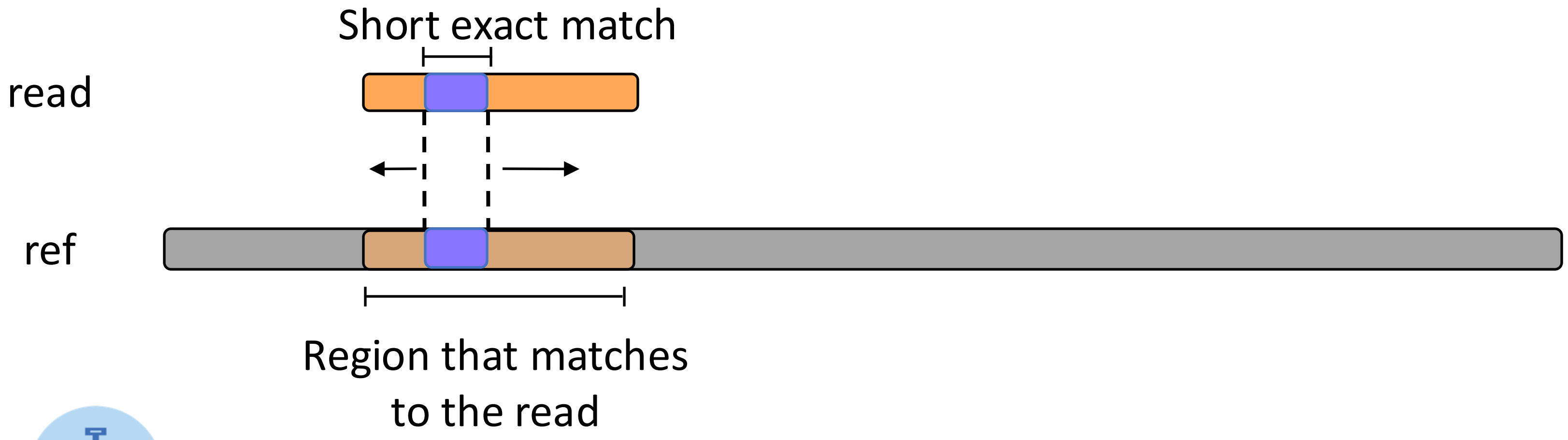
2. **non-exact matching**
   - Many reads will not match the reference perfectly, due to mutations and sequencing errors.

3. **multi mappers**
   - If a read cannot be unambiguously assigned, should the software report all matches, none, or pick one heuristically/randomly?

Alignment - Traditional mapping

To solve the search space problem most alignment tools use the ***seed and extend*** method:

1. Identify short identical sequences between the read and the reference
2. Extend the match on both sides until the alignment becomes poor



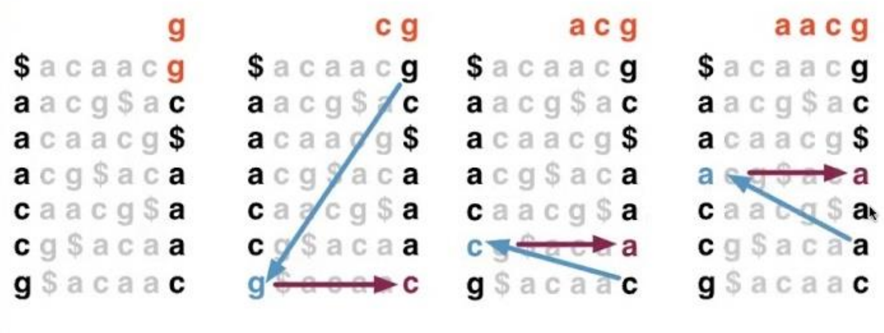
Alignment - Traditional mapping

- To use *the seed and extend* method, we must create a *dictionary* of all short sequences that occur in the reference and their location.

- This dictionary is called the **index**.

- A convenient way of creating the index is the Burrows-Wheeler transform (**BWT**).

- **BWT** allows for fast exact matching with low memory requirements.



```
      g         cg        acg       aacg
$acaacg   $acaacg   $acaacg   $acaacg
aacg$ac   aacg$ac   aacg$ac   aacg$ac
acaacg$   acaacg$   acaacg$   acaacg$
acg$aca   acg$aca   acg$aca   acg$aca
caacg$a   caacg$a   caacg$a   caacg$a
cg$acaa   cg$acaa   cg$acaa   cg$acaa
g$acaac   g$acaac   g$acaac   g$acaac
```

Original publication:
Heng Li, Richard Durbin; Fast and accurate short read alignment with Burrows-Wheeler transform, Bioinformatics, Volume 25, Issue 14, 2009

Reference Genome - fasta

```
>mm10_chr1| Chromosome 1 of Mouse genome version 10
CACACACTTTTCTCACACTATTAGGTAACCCTCTGCCTTATTCCACACTTCCTCCATGAT
GTGCTCCCCGTAGTCACAAAGCCAGCTGGAGACAGACTAGAGCAATGAGCCAAAAGTAGA
CCTTTCCTCCTTTTAACTTGACTGTCTTGAATATTCTGGTACAGTAACAGAAATCTGACT
AGCAGGTCCTTGAGTGAATTCCACCTACATGTGGATATGTGTGAGGATAGAGACCTGTTC
```

- Reference genomes are annotated as .fasta file format

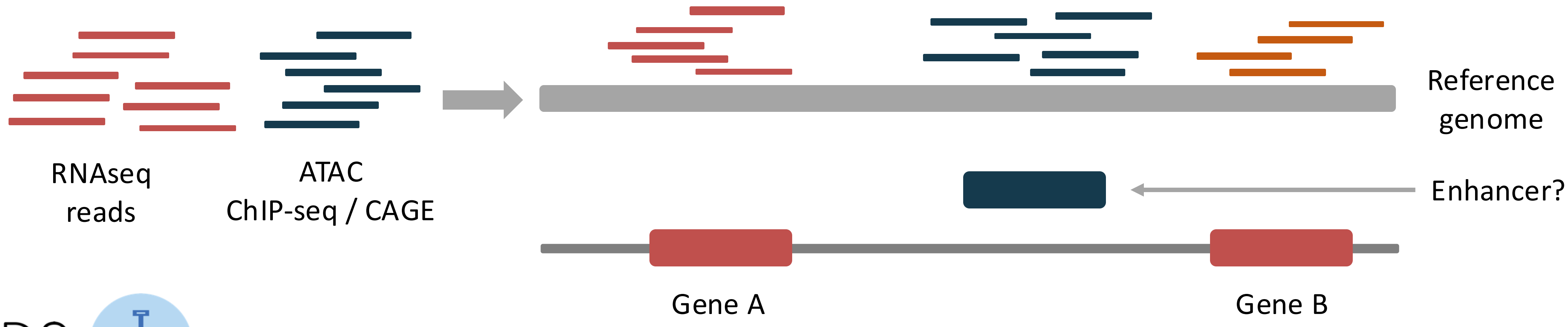
Fasta file:

- Amino acid or nucleotide sequence
- Starts with a single line description ”>”,
  - Reference IDs or descriptions of the sequence, separated by “|”
- Usually used for reference genomes, transcriptomes/exomes

Annotation - Traditional mapping

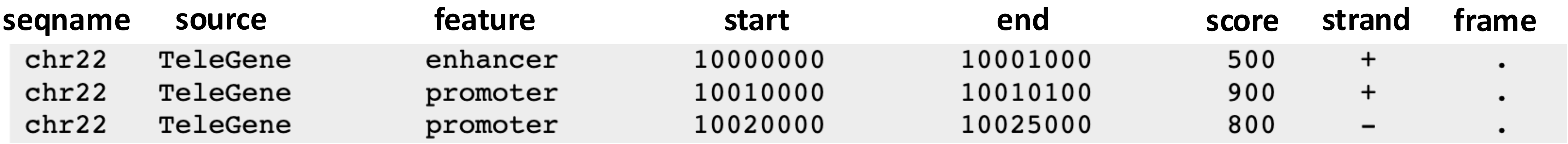
- We are often interested in genes or other annotated regions
- Annotation file is needed = **.GTF / .GFF3** (more on this later)
- **N.B** For some organisms there are multiple versions of the reference genome and annotation files.

Tools:
STAR
Bowtie
BWA
HISAT2



File formats: GFF/GTF files

- General Feature Format (GFF3). Tab separated file



seqname	source	feature	start	end	score	strand	frame
chr22	TeleGene	enhancer	10000000	10001000	500	+	.
chr22	TeleGene	promoter	10010000	10010100	900	+	.
chr22	TeleGene	promoter	10020000	10025000	800	-	.

- Column 9 contains attributes: gene id, gene name, transcript id, etc.

ID=ENSMUSG00000102693.2;gene_id=ENSMUSG00000102693.2;gene_type=TEC;gene_name=4933401J01Rik;
level=2;mgi_id=MGI:1918292;havana_gene=OTTMUSG00000049935.1

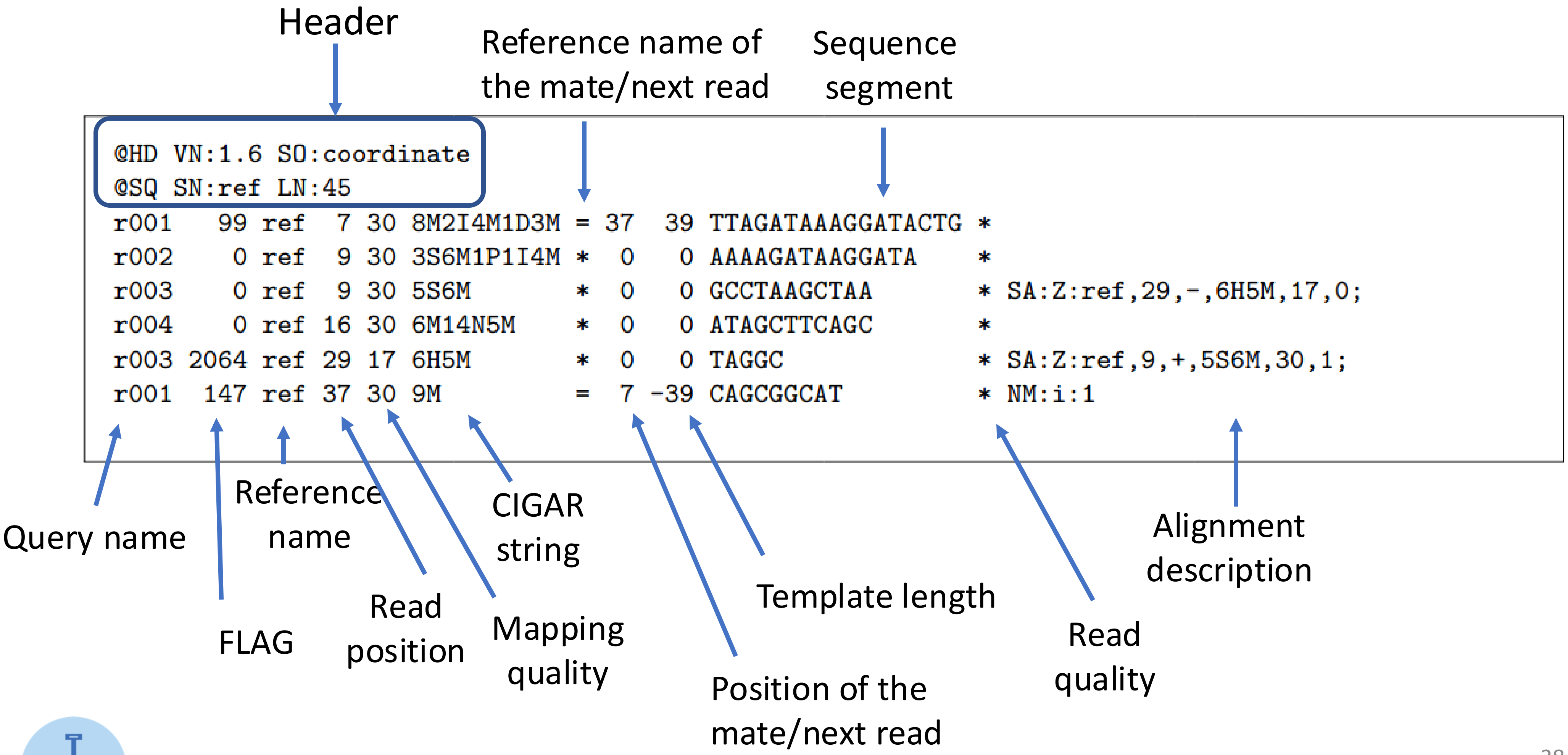
- Gene Transfer Format (GTF). Pretty much the same

File formats: SAM/BAM file

- **Sequence Alignment Map (SAM)**
- SAM format is a generic format for storing large nucleotide sequence alignments
- BAM file is the compressed version → Otherwise the files can be huge!
- You use **samtools** to interact with alignment data

SAMtools

File formats: SAM file



```
@HD VN:1.6 SO:coordinate
@SQ SN:ref LN:45
r001   99 ref  7 30 8M2I4M1D3M = 37  39 TTAGATAAAGGATACTG *
r002    0 ref  9 30 3S6M1P1I4M *  0   0 AAAAGATAAGGATA    *
r003    0 ref  9 30 5S6M       *  0   0 GCCTAAGCTAA       * SA:Z:ref,29,-,6H5M,17,0;
r004    0 ref 16 30 6M14N5M    *  0   0 ATAGCTTCAGC       *
r003 2064 ref 29 17 6H5M       *  0   0 TAGGC             * SA:Z:ref,9,+,5S6M,30,1;
r001  147 ref 37 30 9M         =  7 -39 CAGCGGCAT         * NM:i:1
```

Header

Reference name of the mate/next read

Sequence segment

Query name

FLAG

Reference name

Read position

Mapping quality

CIGAR string

Position of the mate/next read

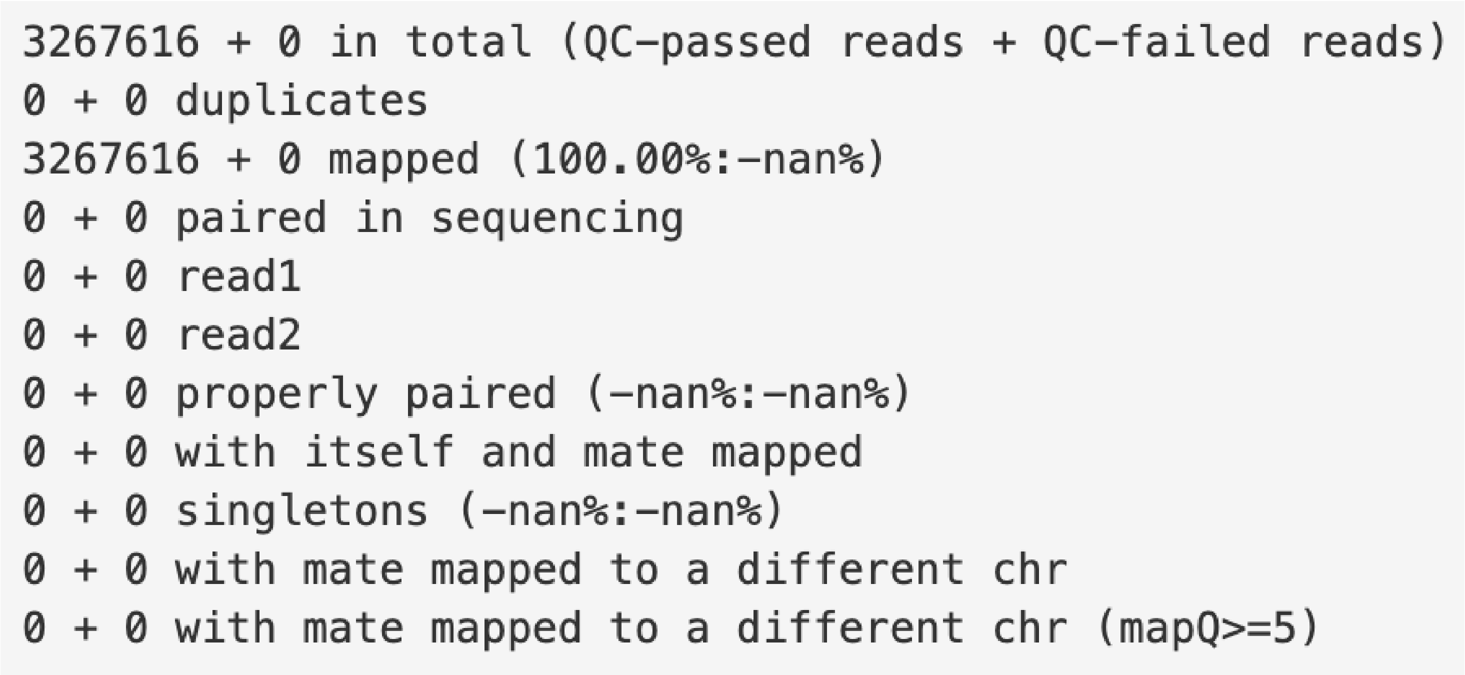
Template length

Read quality

Alignment description

File formats: SAM/BAM file

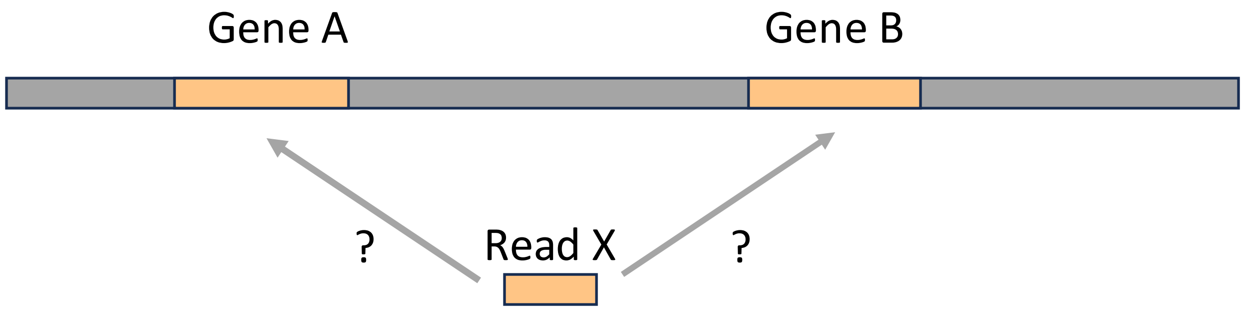
Samtools flagstat will check your sam/bam files and give you general information about the aligned reads



```
3267616 + 0 in total (QC-passed reads + QC-failed reads)
0 + 0 duplicates
3267616 + 0 mapped (100.00%:-nan%)
0 + 0 paired in sequencing
0 + 0 read1
0 + 0 read2
0 + 0 properly paired (-nan%:-nan%)
0 + 0 with itself and mate mapped
0 + 0 singletons (-nan%:-nan%)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
```

Reminder: Alignment

Mapping/alignment is the process of figuring out the most likely origin of a read in a reference genome:



x 20 mio reads!

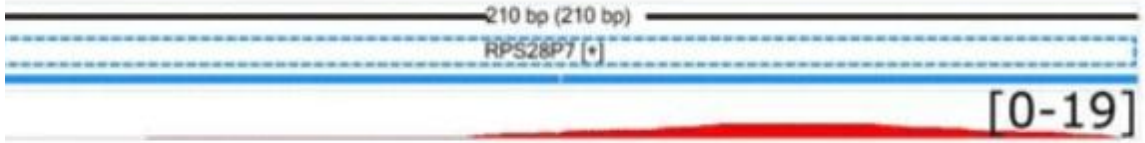
Exercise

Have a look at the mapping results depicted below.

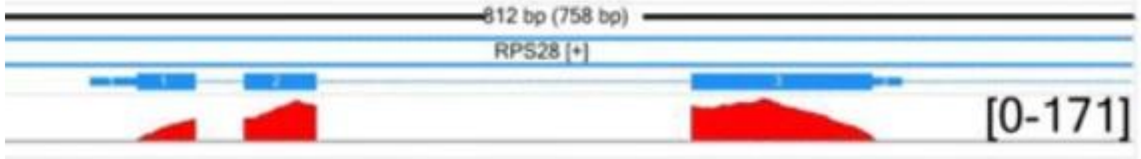
We see reads mapped to two genes, RPS28P7 (Gene A) and RPS28 (Gene B).

The coverage (how many reads have mapped) is shown in red. In blue we see the exon annotation of the two genes.

Gene A



Gene B



Why does the mapping for the two genes look different? What is it we see here? Discuss with your neighbours.

Preprocessing:

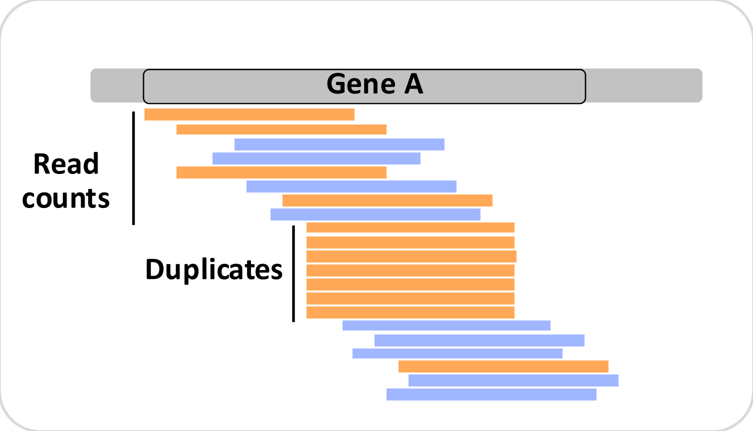
Raw Sequencing Data

Read QC & Trimming

Read Alignment & Annotation

Quantify Reads

Quality Control



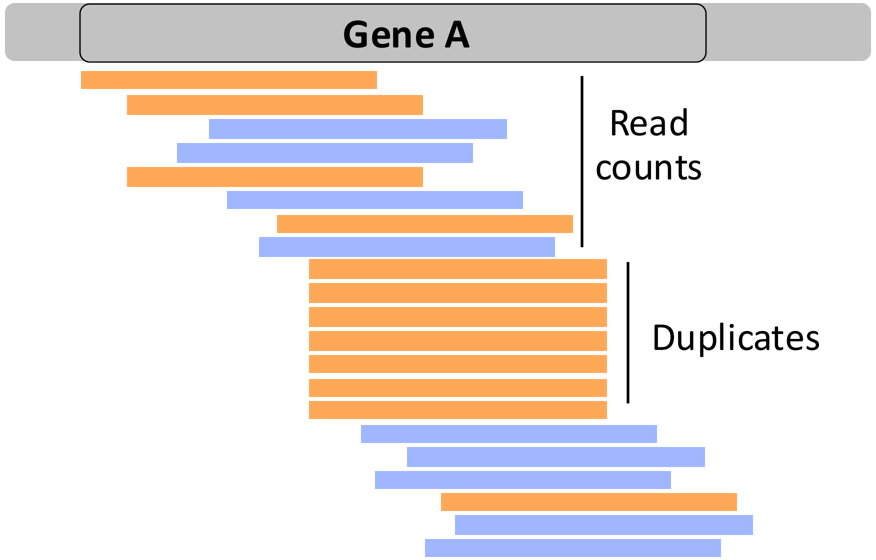
Read Duplicates

Tools:
MarkDuplicates
Samtools

- Biological:
  - Random reads attachment to flow cell
  - Over-sequencing of highly expressed genes

- Technical :
  - PCR enriches smaller and more GC-poor molecules
  - Over-amplification
  - Low library complexity

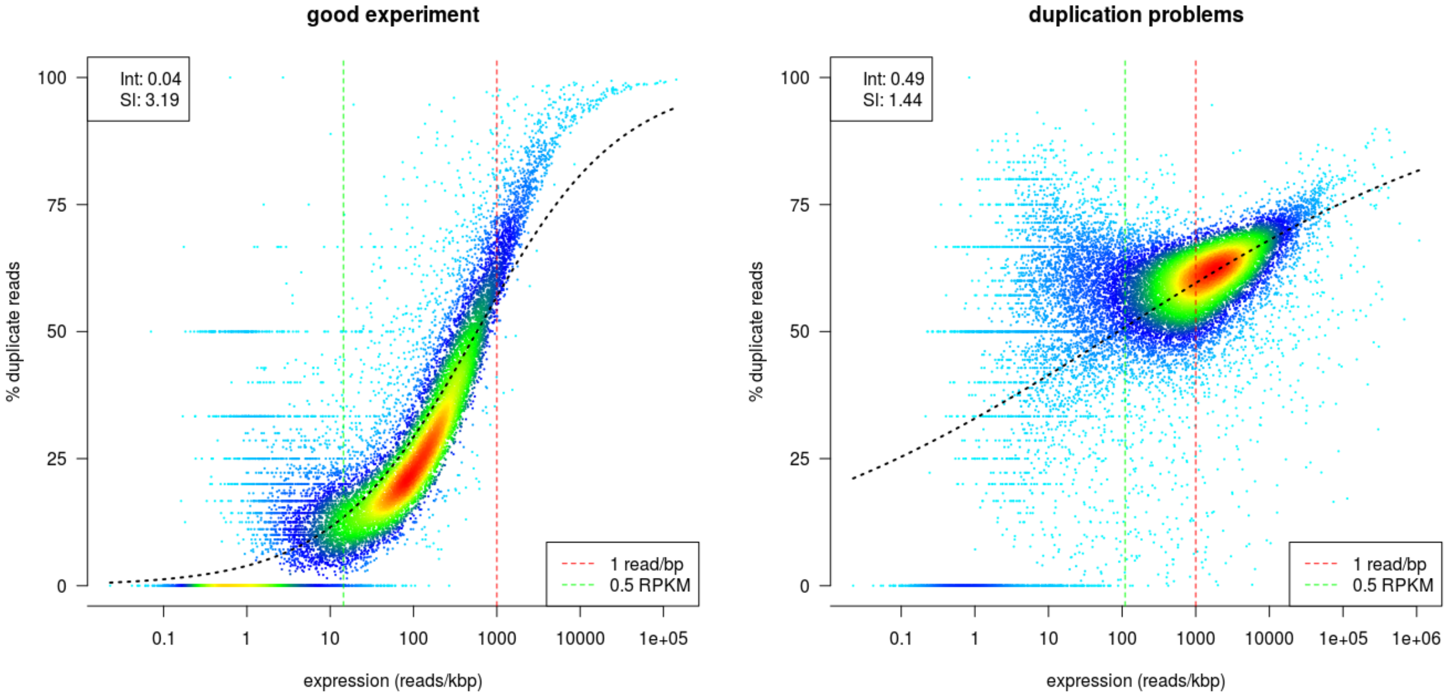
- It is usually not recommended to remove duplicates



Remove Duplicates?

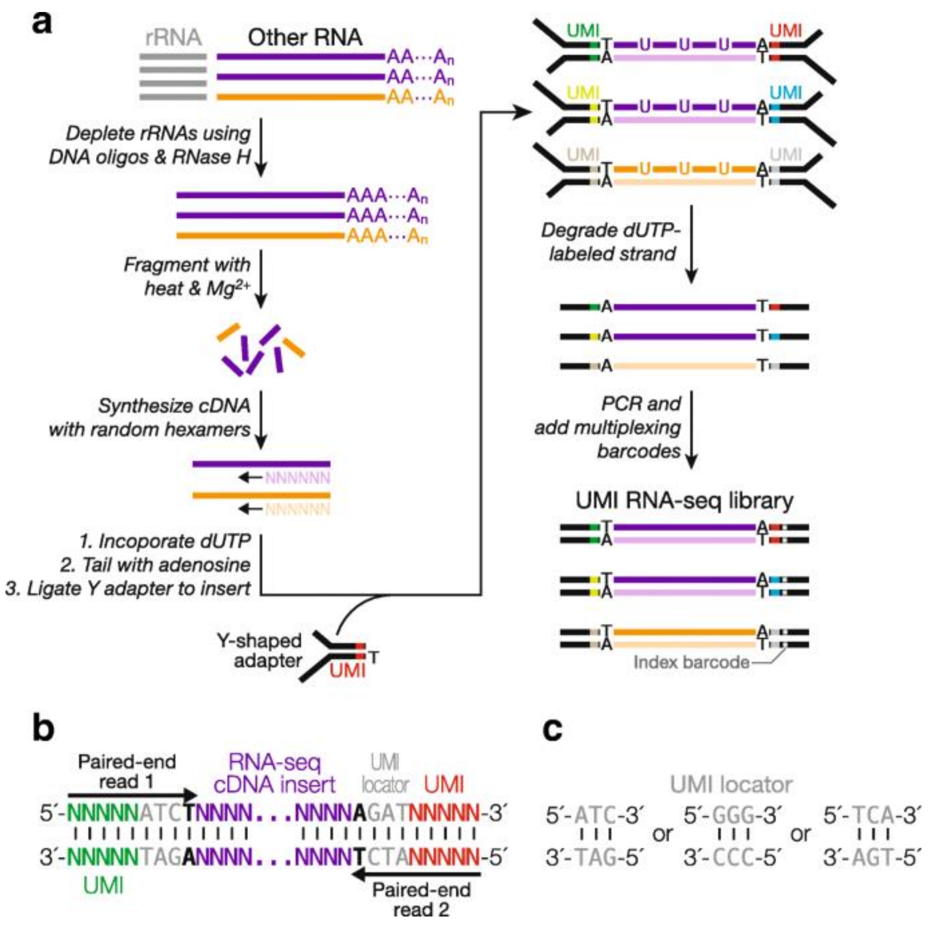
Tools:
dupRadar

- QC metrics and plots
  - duplication vs gene expression
- Technical duplication:
  - Is low: high duplication for highly expressed genes.
  - Is high :duplication for all genes, irrespective of transcription level.



Avoid Duplicates!

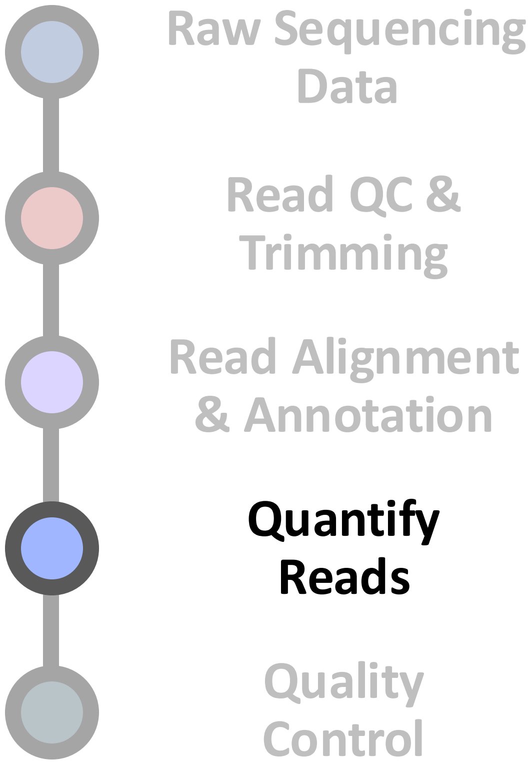
- Avoid duplicates with **Unique Molecular Identifiers** (**UMI**)
- UMI protocol is standard for single cell
- Small sequence is added to library
  - Biological duplicates -> Different UMI
  - Identify PCR duplicates and remove them
- Specially for low-input or deep RNA-seq experiment



Fu et al. 2018

Lunch break

Preprocessing:



Raw Sequencing Data

Read QC & Trimming

Read Alignment & Annotation

Quantify Reads

Quality Control

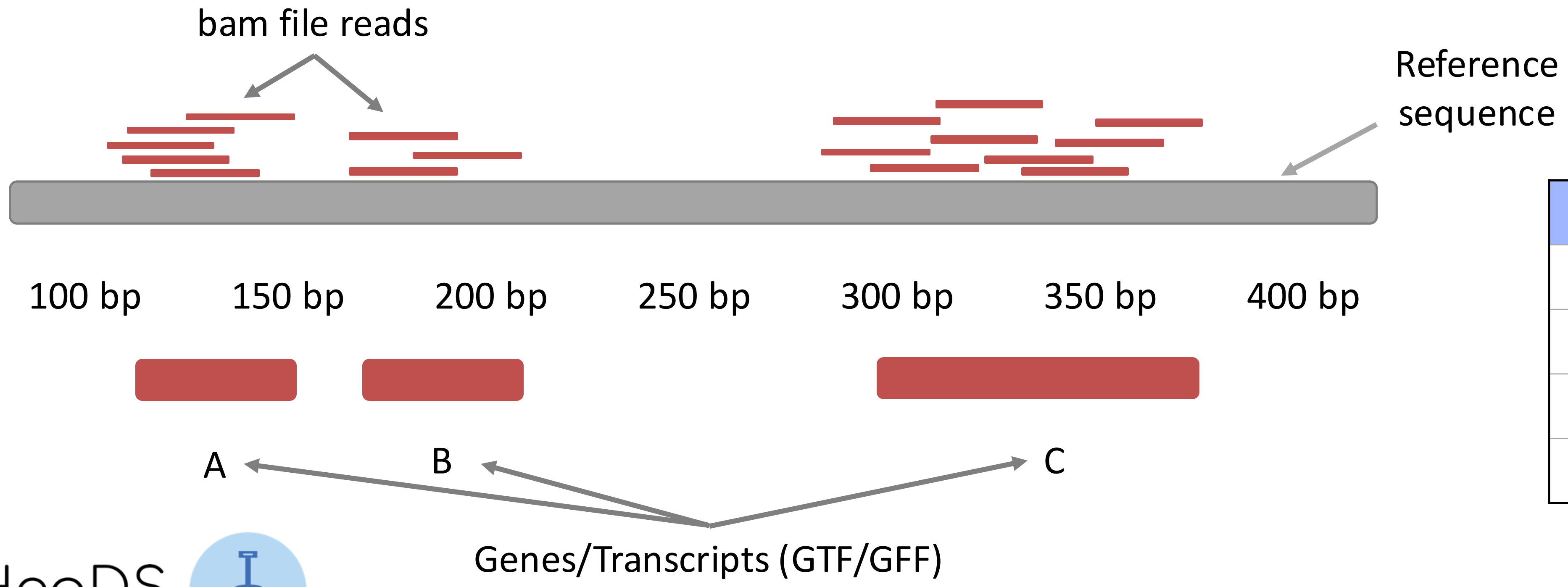
Gene ID	Counts
Gene A	5
Gene B	3
Gene C	9
Gene D	0

Read count tables

Most of times is this the end goal!

Matrix of read counts per gene or per genomic region

Tools:
Salmon
RSEM
(bedtools)



Gene ID	Counts
Gene A	5
Gene B	3
Gene C	9
Gene D	0

Pseudoalignment & Quantification

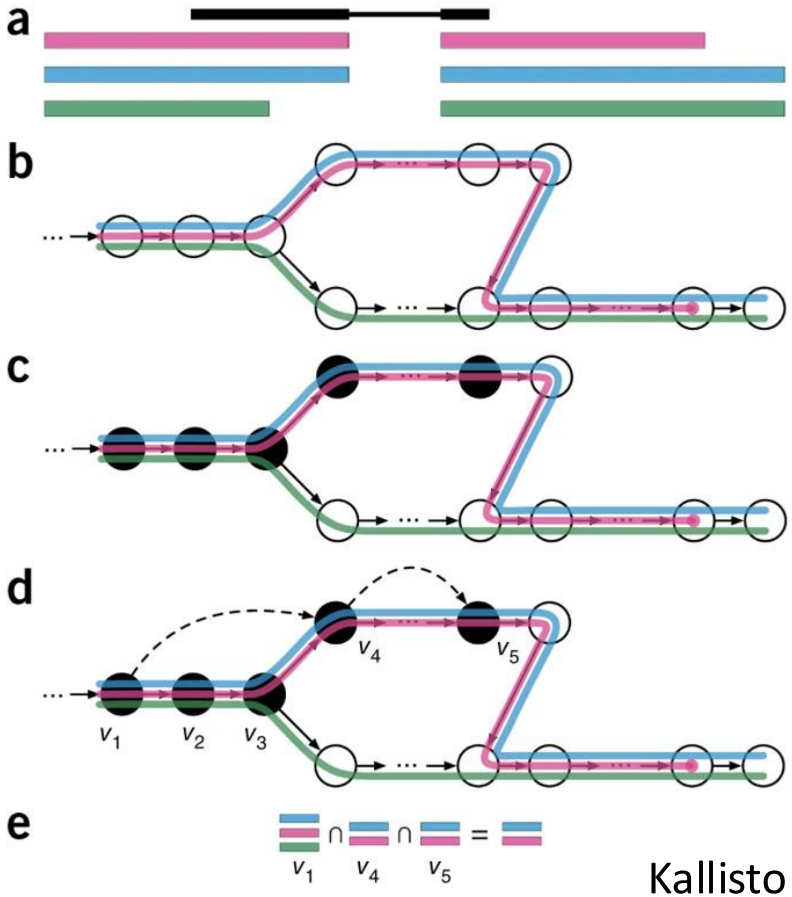
Tools:
Salmon
Kallisto

Alignment and quantification in one step:

- 4 x faster than the **fastest** regular alignment tools
- Memory usage ~ 10 x lower
- Easy to use!
- Precomputed indexes for several species

Bray, Nicolas L., et al. "Near-optimal probabilistic RNA-seq quantification." Nature biotechnology 34.5 (2016): 525-527.

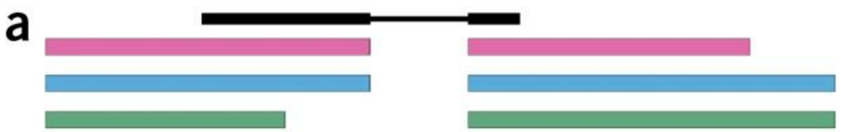
https://tinyheero.github.io/2015/09/02/pseudoalignments-kallisto.html



Kallisto

HeaDS

Pseudoalignment & Quantification



a) A read and three gene transcripts
b) Create a graph of transcripts
    a) Each node = k-mer
    b) Each node = compatible with X transcripts
c) Index nodes and compatibilities
d) Remove redundant information
e) Calculate k-compatibility of read

Pseudoalignment & Quantification

Traditional quantification vs pseudo-quantification

- Traditional quantification assigns one mapped read to a genomic feature (integers)
- Pseudo-quantification estimates and models expected counts to transcripts (continuous)
- Transcript pseudo-counts can be transformed to gene counts
- Pseudo-counts need to be slightly processed for downstream analysis

Traditional quantification

Transcript	Sample A	Sample B	Sample C
A	5	20	98
B	3	0	22
C	9	109	15

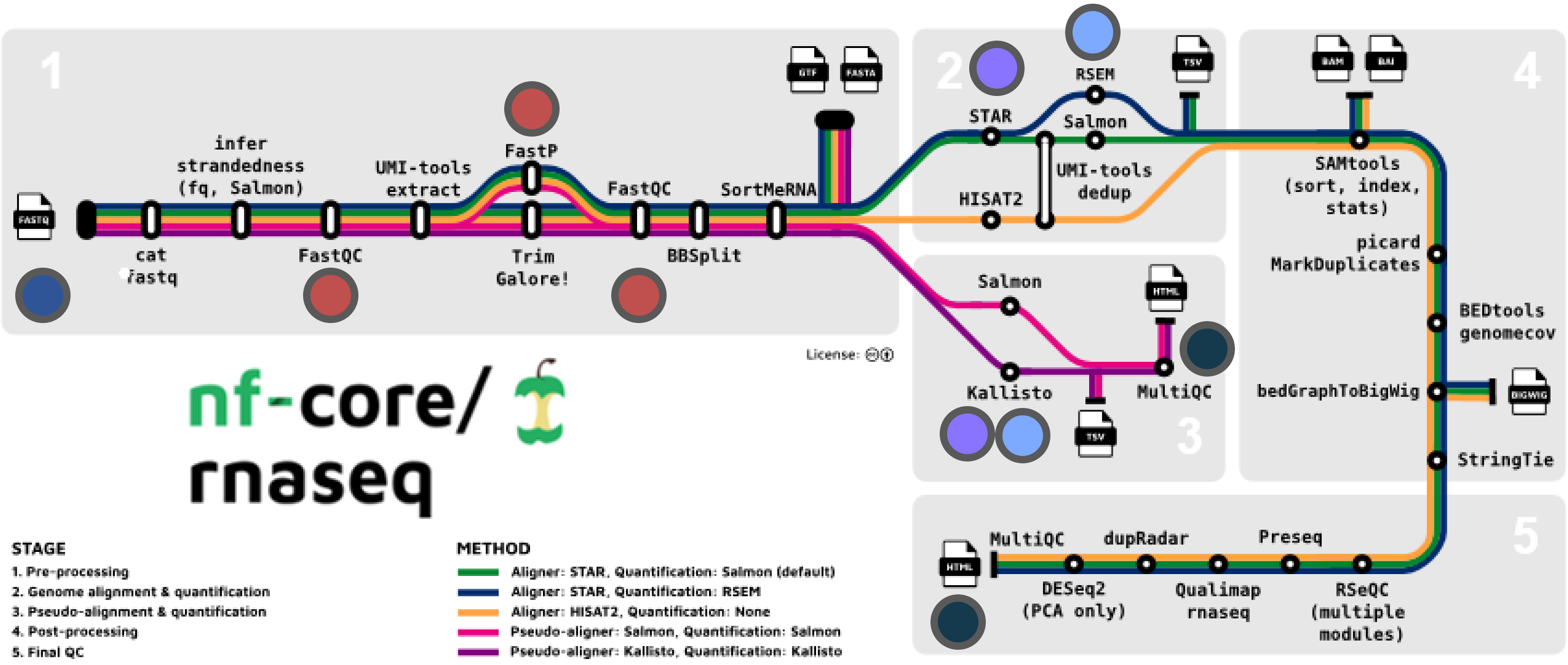
Pseudo-quantification

Transcript	Sample A	Sample B	Sample C
A	10.2	42.11	203.19
B	6.12	0.00	97.43
C	20.35	204.1	64.12

Overview

Preprocessing:

- ● Raw Sequencing Data
- ● Read QC & Trimming
- ● Read Alignment & Annotation
- ● Quantify Reads
- ● Quality Control



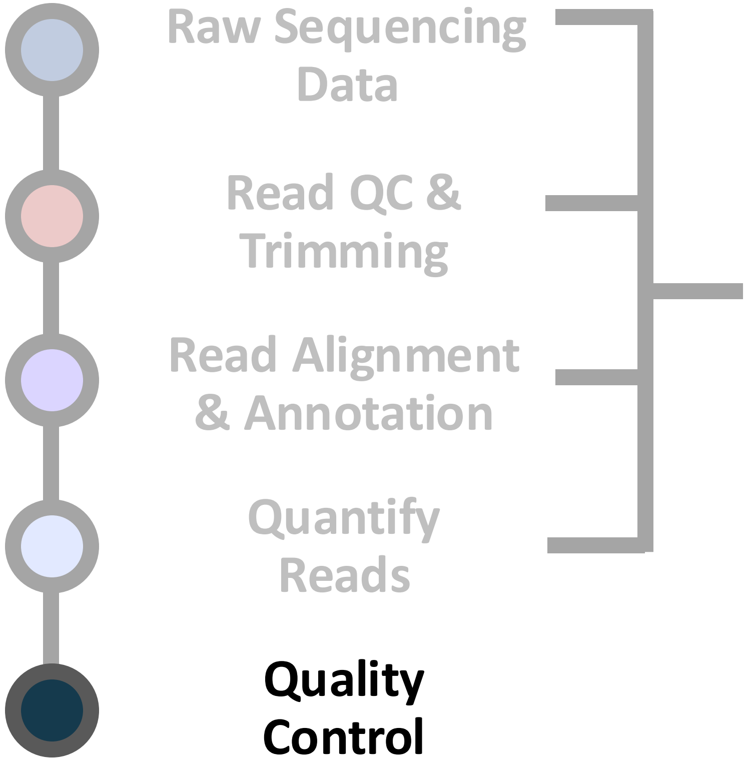
STAGE

1. Pre-processing
2. Genome alignment & quantification
3. Pseudo-alignment & quantification
4. Post-processing
5. Final QC

METHOD

- 🟩 Aligner: STAR, Quantification: Salmon (default)
- 🟦 Aligner: STAR, Quantification: RSEM
- 🟧 Aligner: HISAT2, Quantification: None
- 🟪 Pseudo-aligner: Salmon, Quantification: Salmon
- 🟪 Pseudo-aligner: Kallisto, Quantification: Kallisto

Preprocessing:



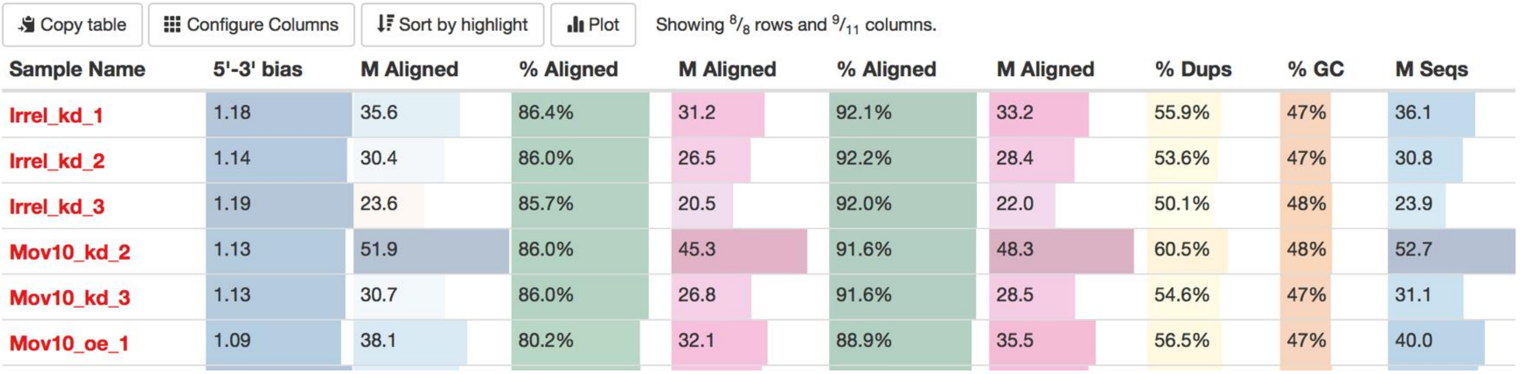
Raw Sequencing Data

Read QC & Trimming

Read Alignment & Annotation

Quantify Reads

Quality Control



Showing $^8/_8$ rows and $^9/_{11}$ columns.

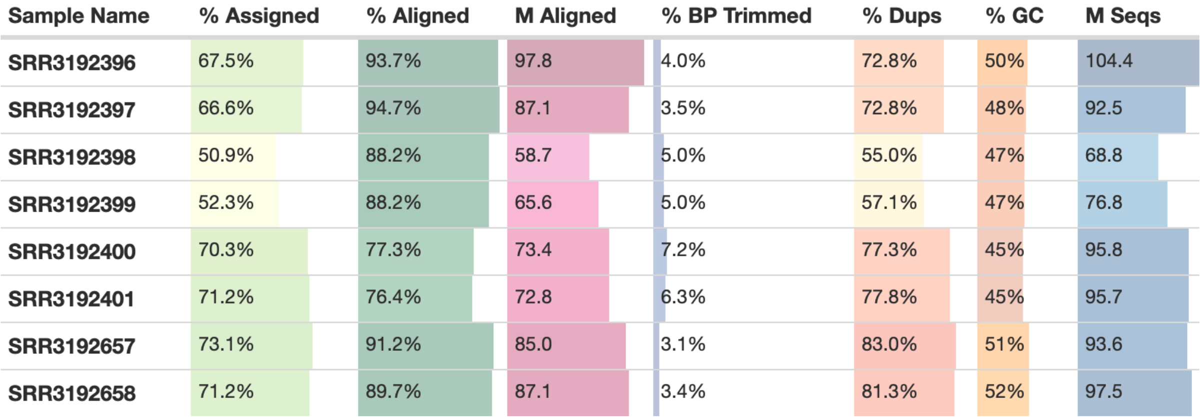
Sample Name	5'-3' bias	M Aligned	% Aligned	M Aligned	% Aligned	M Aligned	% Dups	% GC	M Seqs
Irrel_kd_1	1.18	35.6	86.4%	31.2	92.1%	33.2	55.9%	47%	36.1
Irrel_kd_2	1.14	30.4	86.0%	26.5	92.2%	28.4	53.6%	47%	30.8
Irrel_kd_3	1.19	23.6	85.7%	20.5	92.0%	22.0	50.1%	48%	23.9
Mov10_kd_2	1.13	51.9	86.0%	45.3	91.6%	48.3	60.5%	48%	52.7
Mov10_kd_3	1.13	30.7	86.0%	26.8	91.6%	28.5	54.6%	47%	31.1
Mov10_oe_1	1.09	38.1	80.2%	32.1	88.9%	35.5	56.5%	47%	40.0

MultiQC report

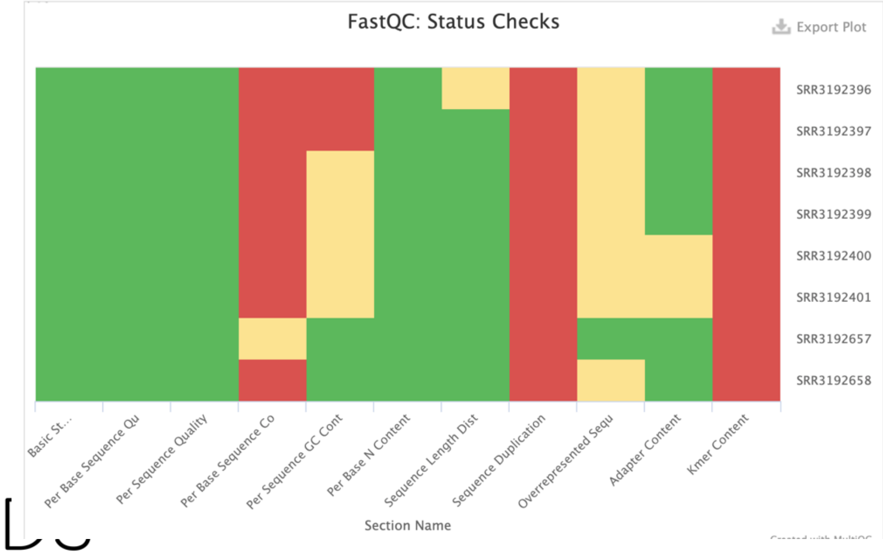
MultiQC report

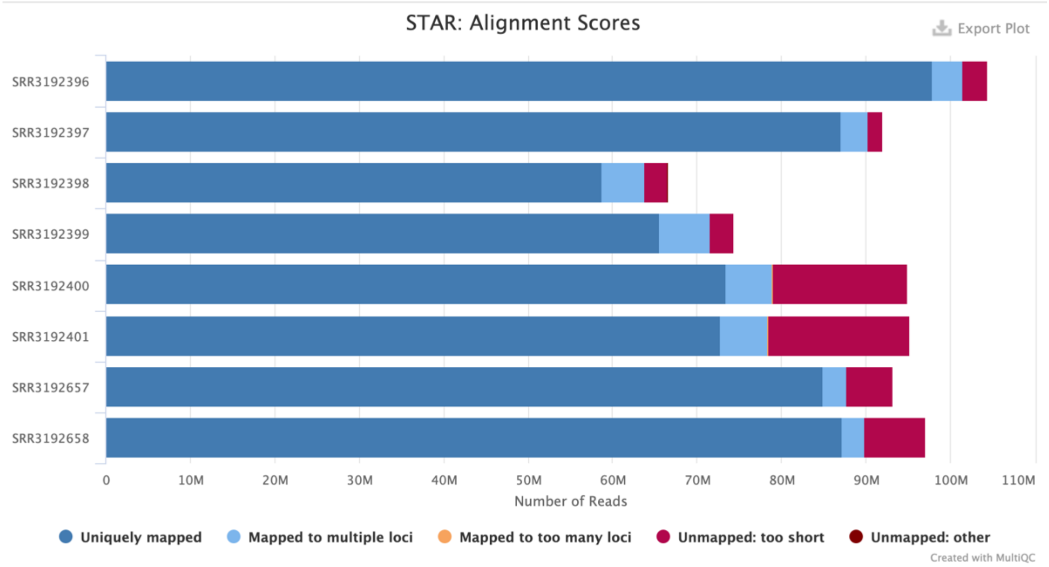
Summarises all kinds of reports:

- FastQC
- Trimming
- Alignment
- Feature counts
- Differential Expression



Sample Name	% Assigned	% Aligned	M Aligned	% BP Trimmed	% Dups	% GC	M Seqs
SRR3192396	67.5%	93.7%	97.8	4.0%	72.8%	50%	104.4
SRR3192397	66.6%	94.7%	87.1	3.5%	72.8%	48%	92.5
SRR3192398	50.9%	88.2%	58.7	5.0%	55.0%	47%	68.8
SRR3192399	52.3%	88.2%	65.6	5.0%	57.1%	47%	76.8
SRR3192400	70.3%	77.3%	73.4	7.2%	77.3%	45%	95.8
SRR3192401	71.2%	76.4%	72.8	6.3%	77.8%	45%	95.7
SRR3192657	73.1%	91.2%	85.0	3.1%	83.0%	51%	93.6
SRR3192658	71.2%	89.7%	87.1	3.4%	81.3%	52%	97.5





Exercise

On the course website we have put up a multiQC report:
https://hds-sandbox.github.io/bulk_RNAseq_course/develop/workshop_RNAseq_nov2024.html
(download QC files button)

multiqc_report_star_rsem_mod.html

The report has been slightly shortened for the purpose of this course. Go through the file with your seat neighbour(s). For each section, pinpoint which preprocessing step it belongs to and explain what is shown in the report. Then answer the following questions:

- What has the trimming changed?
- Is the data quality after trimming acceptable?
- Are duplicates in the data problematic? (check the dupRadar result)
- Was the mapping successful? Why are none of the reads paired?
- Look at the genomic features the reads are mapped to. Is this what you expected?