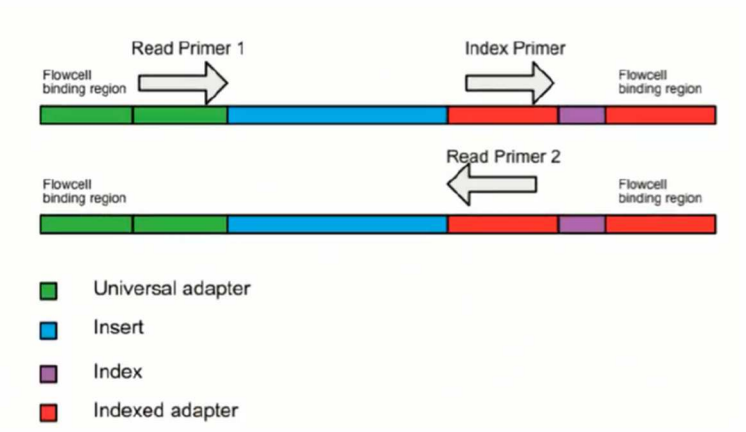
DATA ANALYSIS OF RNA-SEQUENCING

Bulk analysis = simultaneous sequencing of a mixture of cells. **Single cell analysis** = each cell is separated from the bulk prior to the analysis.

RNA-seq requires generation of cDNA → cDNA is blunted at both ends, then repaired, then a 3'A tail is added to both ends and adapters are ligated. Once adapters are attached, primers can be added. Now cDNA fragments are amplified through PCR to generate a library, and sequencing can be performed. **RNA is sequenced after generation of short reads** (the longest cDNA is about 150-500 nucleotides long: no way to sequence a whole transcript) → second generation sequencing.

The presence of **Read Primer 1** and **Read Primer 2** entails that it is possible to perform either single or double end/paired-end sequencing. Nowadays, single-end sequencing is enough to perform this kind of analysis (75 to 150 nucleotides per side can be sequenced now, and this allows to have already a unique identification of the transcript sequence).

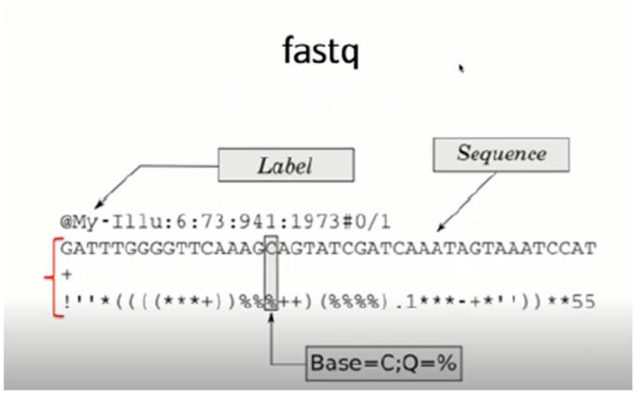
There is also an **index primer** that allows to sequence more than one sample and distinguish among the different samples (since all sequences are analysed together).



Paired-end sequencing scheme

```
>NM_000546.6 Homo sapiens tumor protein p53 (TP53), transcript variant 1, mRNA
CTCAAAAGTCTAGAGCCACCGTCCAGGGAGCAGGTAGCTGCTGGGCTCCGGGGACACTTTGCGTTCGGGC
TGGGAGCGTGCTTTCCACGACGGTGACACGCTTCCCTGGATTGGCAGCCAGACTGCCTTCCGGGTCACTG
CCATGGAGGAGCCGCAGTCAGATCCTAGCGTCGAGCCCCCTCTGAGTCAGGAAACATTTTCAGACCTATG
GAAACTACTTCCTGAAAACAACGTTCTGTCCCCCTTGCCGTCCCAAGCAATGGATGATTTGATGCTGTCC
CCGGACGATATTGAACAATGGTTCACTGAAGACCCAGGTCCAGATGAAGCTCCCAGAATGCCAGAGGCTG
CTCCCCCCGTGGCCCCTGCACCAGCAGCTCCTACACCGGCGGCCCCTGCACCAGCCCCCTCCTGGCCCCT
GTCATCTTCTGTCCCTTCCCAGAAAACCTACCAGGGCAGCTACGGTTTCCGTCTGGGCTTCTTGCATTCT
GGGACAGCCAAGTCTGTGACTTGCACGTACTCCCCTGCCCTCAACAAGATGTTTTGCCAACTGGCCAAGA
CCTGCCCTGTGCAGCTGTGGGTTGATTCCACACCCCCGCCCGGCACCCGCGTCCGCGCCATGGCCATCTA
CAAGCAGTCACAGCACATGACGGAGGTTGTGAGGCGCTGCCCCCACCATGAGCGCTGCTCAGATAGCGAT
GGTCTGGCCCCTCCTCAGCATCTTATCCGAGTGGAAGGAAATTTGCGTGTGGAGTATTTGGATGACAGAA
ACACTTTTCGACATAGTGTGGTGGTGCCCTATGAGCCGCCTGAGGTTGGCTCTGACTGTACCACCATCCA
CTACAACTACATGTGTAACAGTTCCTGCATGGGCGGCATGAACCGGAGGCCCATCCTCACCATCATCACA
CTGGAAGACTCCAGTGGTAATCTACTGGGACGGAACAGCTTTGAGGTGCGTGTTTGTGCCTGTCCTGGGA
GAGACCGGCGCACAGAGGAAGAGAATCTCCGCAAGAAAGGGGAGCCTCACCACGAGCTGCCCCCAGGGAG
CACTAAGCGAGCACTGCCCAACAACACCAGCTCCTCTCCCCAGCCAAAGAAGAAACCACTGGATGGAGAA
TATTTCACCCTTCAGATCCGTGGGCGTGAGCGCTTCGAGATGTTCCGAGAGCTGAATGAGGCCTTGGAAC
TCAAGGATGCCCAGGCTGGGAAGGAGCCAGGGGGGAGCAGGGCTCACTCCAGCCACCTGAAGTCCAAAAA
GGGTCAGTCTACCTCCCGCCATAAAAAACTCATGTTCAAGACAGAAGGGCCTGACTCAGACTGACATTCT
```

FASTA files contain a header followed by the sequence



Sequences resulting from this analysis are in a FASTQ format → modification of FASTA format.

FASTQ files store sequencing data: they have a header (starting with **@**), the sequence, a separator (normally it's **+**) and then some quality info to tell whether nucleotides reliable or not to the aligner software (**Q**, quality score), this way it will be possible to generate a score telling us if the output is reliable or not (smaller/higher error probability). Quality score is encoded in ASCII, which is read by the alignment software (see later why).

The quality score Q is also called **Phred score** and it can be calculated as it follows:

$$Q = -10log_{10}p$$

Where p is the probability the called base is wrong (p-value). We want p to be low. The logarithmic representation enables to reduce the number of 0 and have a better visualization of results. However, when the Q is higher than 9 (thus when it requires 2

digits instead of 1), it is more difficult to align the score with the corresponding nucleotide. That is why Q scores are represented in ASCII code:

ASCII Table

Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	`
1	1	1		33	21	41	!	65	41	101	A	97	61	141	a
2	2	2		34	22	42	"	66	42	102	B	98	62	142	b
3	3	3		35	23	43	#	67	43	103	C	99	63	143	c
4	4	4		36	24	44	$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47	'	71	47	107	G	103	67	147	g
8	8	10		40	28	50	(	72	48	110	H	104	68	150	h
9	9	11		41	29	51	)	73	49	111	I	105	69	151	i
10	A	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	B	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	C	14		44	2C	54	,	76	4C	114	L	108	6C	154	l
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	E	16		46	2E	56	.	78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	O	111	6F	157	o
16	10	20		48	30	60	0	80	50	120	P	112	70	160	p
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	s
20	14	24		52	34	64	4	84	54	124	T	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	v
23	17	27		55	37	67	7	87	57	127	W	119	77	167	w
24	18	30		56	38	70	8	88	58	130	X	120	78	170	x
25	19	31		57	39	71	9	89	59	131	Y	121	79	171	y
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	z
27	1B	33		59	3B	73	;	91	5B	133	[	123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	|
29	1D	35		61	3D	75	=	93	5D	135	]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137	_	127	7F	177	

In very old ASCII tables, as this one, characters 0-32 are present but they're not typeable. To have a single character representing your datum (i.e. the nucleotide), Q has to be more than 33: for this reason, characters from 33 to 73 represent the actual quality scores from 1 to 40, which are those normally assigned to nucleotides. **Quality score starts decreasing starting from the 100th nucleotide** (thus from there the sequencing becomes less reliable).

Table 5.1. Base Quality and Accuracy.

Q_{Phred}	p	Accuracy
0	1	0%
10	10^{-1}	90%
20	10^{-2}	99%
30	10^{-3}	99.9%
40	10^{-4}	99.99%
50	10^{-5}	99.999%
60	10^{-6}	99.9999%
70	10^{-7}	99.99999%
80	10^{-8}	99.999999%
90	10^{-9}	99.9999999%
93	$10^{-9.3}$	99.99999995%

The highest Q score obtainable is 40 nowadays, but everything above 30 is considered a good quality score.

The Q score is used to associate the reads to specific positions of the genome during the alignment.

FASTQ files are generated via the **bcl2fastq** (standing for bcl-to-fastq) software, where **BCL** are the image files generated by the sequencer. These files are first divided considering the index (divided by sample), and then for each sample associated files are created.

Sequencing generates 20 to 100 million reads per sample. In bulk RNA-seq at least 20 million reads per samples are done to detect genomic locations of transcripts (but it's not possible to identify isoforms). For noncoding genes, 80-100 million reads are necessary instead. The amount of reads needed is proportional to the amount of info to be retrieved. A **minimal coverage** is necessary to have a significant result. Also, absolute evaluations with RNA-seq cannot be done, but rather comparative evaluations are possible: this is because we rely upon retrotranscription into cDNA sequences (thus on the RT enzyme efficiency and the quality of the RNA fragment) → meaning that, for instance, we cannot say that GAPDH is more expressed

than p53 in the same sample. However, we can say that the expression of the same gene changes from sample 1 to sample 2 (same gene, same fragment, same enzyme, but different experiments).

SEQUENCER OUTPUT

```
<sample_name>_<barcode_sequence>_L<lane>\
_R<read_number>_<set_number>.fastq.gz

NIST7035_TAAGGCGA_L001_R1_001.fastq.gz
```

1. **<sample_name>** → NIST7035
2. **<barcode_sequence>** → TAAGGCA (the index associated to the sequence).
3. **L<XXX>** → L001 (the lane in the flowcell in the sequencer where the sequence comes from)
4. **R<X>** → R1 (to know whether the read comes from Read Primer 1 or 2: R1 is read from 5' to 3', forward).
5. **<set_number>** → 001
6. **Extension** → .fastq.gz

The read (piece of sequence generated by the sequencer) is characterized by a header/label starting with @ and ending with 1 or 2 (for Read Primer 1 or 2). Thus, Read Primer 1 and 2 of the same sequence share the same length and name, but differ for the number the label ends with. This way reads of the same sequence can be paired.

QUALITY CONTROL (QC) OF RAW DATA

Bioinformaticians should be involved in the experiment design in order to better know what kinds of data is worth to work with and avoid producing data that are not useful for data analysis. It's always important to have a **biological question** in mind to produce useful data and thus avoid so called **fishing expeditions** (no biological question, lot of data are produced: bunch of patients taken and sequencing of their genomes), which are a waste of time and money. The way you organize data is fundamental as well. Moreover, **homogeneity** is paramount: number of control and treated should be similar to avoid biased analyses.

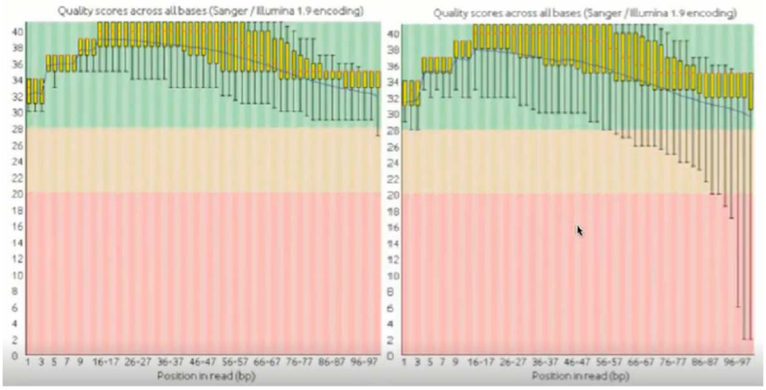
Replicates in an experiment are another important aspect to address. The number of replicates changes depending on the model used (cells, animals, humans) and the biological question (a statistical study on autism will require way more replicates than the analysis of an anti-tumoral drug). Importantly, control and treated samples should be treated in the very same way and timeframes.

All this is necessary to ensure that the data produced are not confused/covered by the background noise.

FastQC is a software that evaluate the quality of the sequence in its FASTQ format. It's a Java software used on RNA and DNA sequences. FastQC evaluates quality of sequences by taking into account multiple parameters. Of course, the interpretation of the results of a FastQC analysis varies according to the considered experiment. Parameters analysed are:

1. **Per base sequence quality (quality score of the reads)** → it gives a visually easy to read report of the quality score Q for every base of all of the reads (of all samples). The green region of the graph is the best one (score above 28). If the mean stays above 28 it's good, no matter the whiskers. If the output of the analysis is bad, i.e. the quality scores are poor, it is possible to set a quality cut-off: some sequences will be lost but the analysis is cleaned up. Otherwise, it is possible to

it makes a mean of Q for every base (a mean considering that specific base in all the reads)



Left graph refers to Read Primer 1 (R1), right one to Read Primer 2 (R2)

analyse only a piece of each sequence (e.g., up to nucleotide 60): this way the number of sequences that are analysed does not vary, but their length does. Either one strategy or the other can be adopted depending on the experiment and the question. If length isn't a problem (for instance when the aim is to detect genes (thus for **gene analysis**), where 50 nucleotides are enough), only R1 sequences can be kept (since R2 are generally of lower quality). Instead, both R1 and R2 are necessary for instance to analyse splicing.

2. **Per tile sequence quality** → this graph will only appear if you're using an Illumina library which retains its original sequence identifiers. In these identifiers it is encoded the flowcell tile from which reads came. The graph allows you to look at the quality scores from each tile across all of your bases to see if there was a loss in quality associated with only one part of the flowcell (i.e., these plots represent the change of quality for each sample compared to the mean quality of the experiment. A good quality experiment should have a blue plot → low variation from the mean. Red bands indicate that some lanes of the sequencer weren't reliable, showing a technical issue rather than a problem linked to the RNA sample).

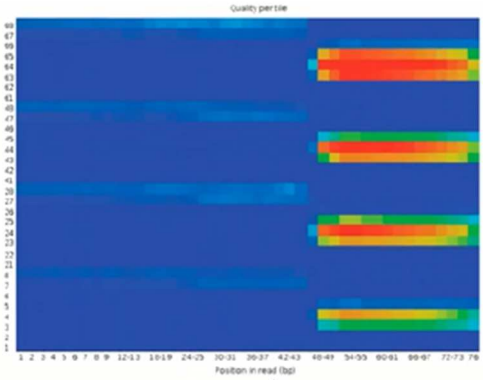
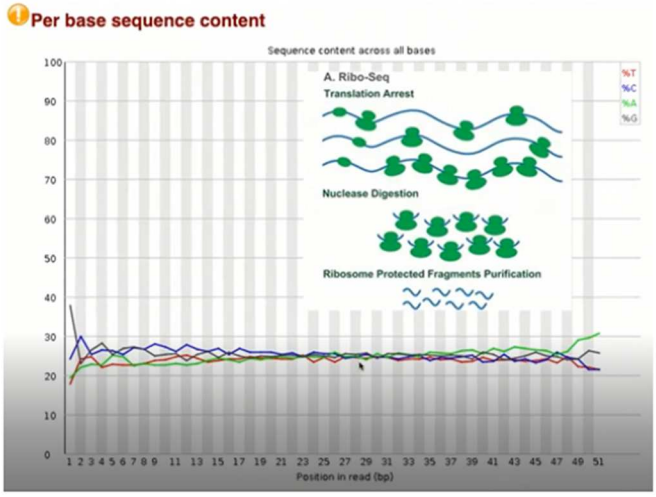
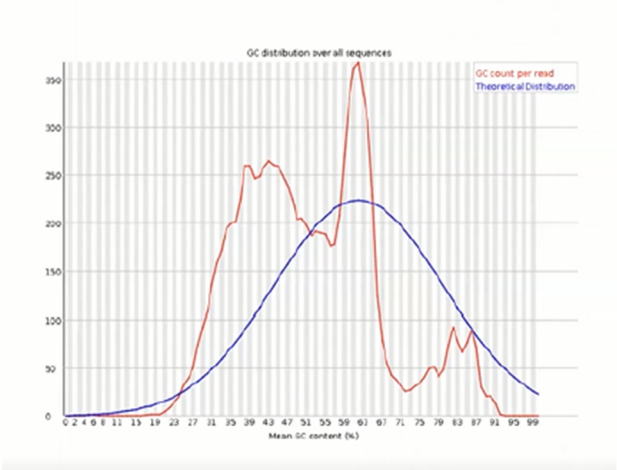


Figure 6.10. Per tile quality. FastQC `Per tile sequence quality` plot for the reverse read of sample SRR098359. The plot shows the deviation from the average quality for each tile, with hotter colors showing that a tile had worse qualities than other tiles for that base. A good plot should be blue all over. In this example, it is apparent that some tiles display poor overall quality.



3. **Per base sequence content** → in theory we expect that each nucleotide has the same distribution: 25% (as in the human genome each nucleotide is equally represented). When we get far from 25%, some technical problems should be addressed, specially if the incoherence appears in the first portion of the sequences. ??? When analysing sequences from kit that investigate only the 3' end (mainly used in scRNA-seq), a different distribution of bases is to be considered normal: the 3' end of transcripts is often bearing noncoding exons with different base distribution than coding regions.

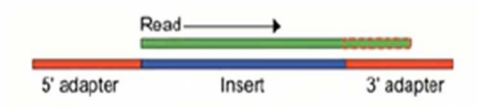
4. **Distribution of GC** → not really interesting in RNA-seq since this is calculated on genomic DNA (in blue). In RNA it is expected to find a difference from the expected GC content (the latter is the blue curve). It's not that important the shape of the curve but rather the fact that the shape is the roughly the same among all reads.



5. **Duplication level** → related to the fact that PCR steps are need to get enough starting material. 12-15 cycles are generally done (to stay in the linear amplification range and have homogeneity

between amplified and non-amplified sequences). The problem is that some sequences can be artifactually overrepresented during PCR. This is an issue especially in scRNA-seq, while in bulk RNA-seq the distribution is the same across all samples and the technical error is equally spread. Some issues in representation can also be due to non-uniform fragmentation (not a problem for scRNA-seq). When duplication artefacts are removed in the bulk analysis → increase (worsening) in the False Discovery Rate (FDR), which indicates the number of false positives.

During fragmentation, fragments smaller than 150 bp can be produced: when sequenced, the resulting reads may end with the 3' adapter sequence (too short fragment, sequencing reaches the adapter?), but we don't want that:



How to remove adaptor sequences present in the reads? By using **trimming tools**. These remove sequences with a quality score lower than a set threshold (also other low-quality sequences than adapters can be removed this way). Examples of such tools are **Skewer** (for both single-end and paired-end data trimming).

In bulk libraries, the unknown sequence is what is in between the p5 and p7 adapters. Within the adapters, there is an index (~6 bp) to discriminate the sample of origin of the sequence (index primer). It goes differently for scRNA-seq libraries; the main platform in this case is **10x Genomics**. The 10x barcode distinguishes the cell from which the sequences come from, while the UMI is a 6 bp random sequence unique for each starting transcript (enables to count the number of different transcripts within the same single cell).