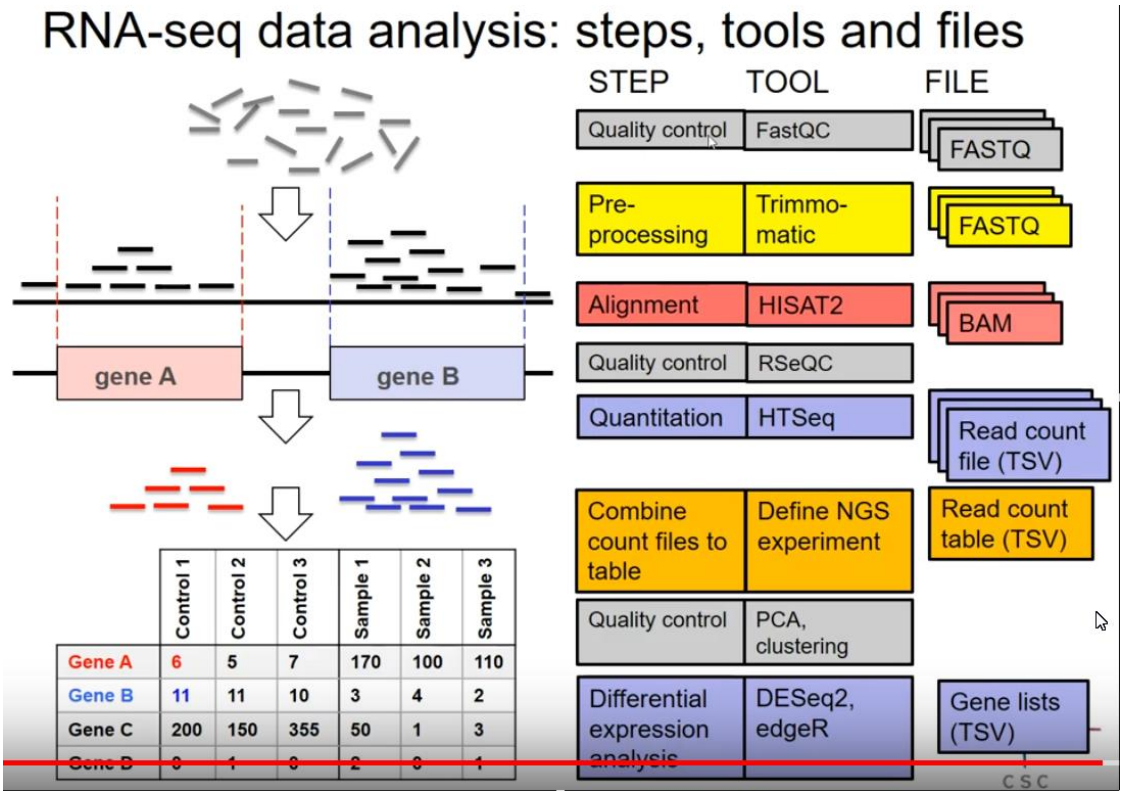
Most of the slides are taken from tutorial videos by Chipster available at https://www.youtube.com/playlist?list=PLjiXAZO27elBj3KYi7ACscgOxlNkNOxPc and from a book P.N. Robinson, R.M. Piro, M. Jäger: Computational Exome and Genome Analysis, CRC Press, 2019.

What can I investigate with RNA-seq?

Differential expression
Isoform switching
New genes and transcripts
New transcriptomes
Variants
Allele-specific expression
Etc etc

Exercise 1. RNA-seq hands-on tutorial using Chipster: ENCODE dataset

In this tutorial you start with raw reads (in fastq files), and learn how to check the read quality, trim bad quality bases, check the strandedness of the data, align reads to genome, and count reads per genes. Then you combine count files for all the samples in one table, and describe your experimental setup using the phenodata file. You also learn how to check coverage uniformity, and whether novel splice junctions were found. Finally, you detect differentially expressed genes and learn how to visualize reads in the genomic context using a genome browser.



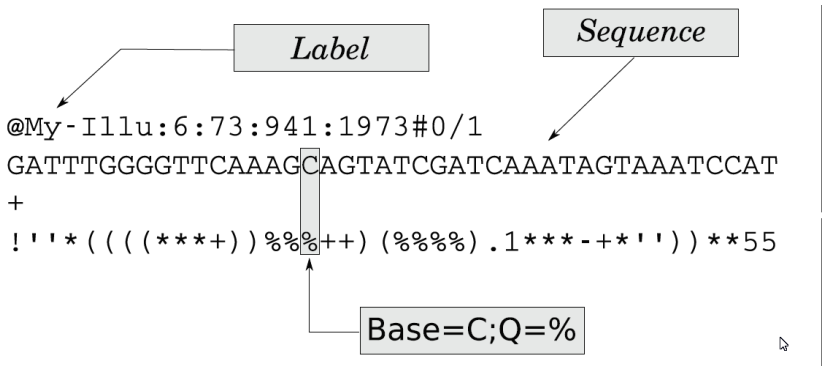
The data is a small subset of single end RNA-seq reads from two human cell lines, h1-hESC and GM12878. Note that when analyzing differential expression you should always have at least 3 biological replicates! We use this small dataset for the first steps of the analysis for the interest of time, but we'll move to another dataset with replicates for the actual differential expression analysis later.

RNA-seq data analysis workflow

I. Quality control (QC) of raw reads
II. Preprocessing if needed
III. Alignment (= mapping) to reference genome
IV. Alignment level QC
V. Quantitation
VI. Describing the experiment with phenodata
VII. Experiment level QC
VIII. Differential expression analysis
IX. Visualization of reads and results in genomic context

1. Launch Chipster. Select **Open example session** and **course_RNAseq_ENCODE**. This session has two fastq files. Note that normally fastq files are zipped and Chipster can use them like that.

Let us look at a format of **hESC.fastq**, e.g., in Notepad++.



5.1. FASTQ. The FASTQ format has four lines per read.

By definition, $\log_a c = \mathrm{b} \Leftrightarrow a^b = c$. The **Phred quality score** is defined as

$$Q = -10 \log_{10} p \Leftrightarrow p = 10^{\frac{-Q}{10}}$$

where ***p*** is the probability that the corresponding base call is **wrong** and ***Q*** is the Phred score (rounded to the closest integer value). E.g., ASCII (%) = ASCII (Q +33) ⇒ 37 = Q + 33 ⇒ Q = 37 – 33 = 4 ⇒ $p = 10^{-4/10} = 10^{-0.4} = 0{,}398$ according to formulae on the next page:

Table 5.1. Base Quality and Accuracy

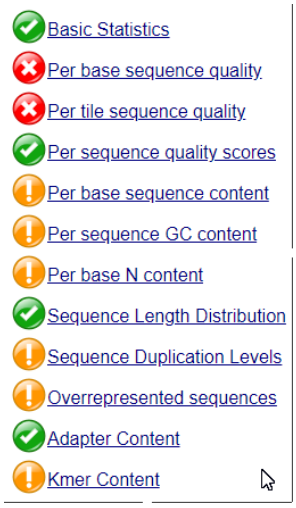
Q_{Phred}	p	Accuracy
0	1	0%
10	10^{-1}	90%
20	10^{-2}	99%
30	10^{-3}	99.9%
40	10^{-4}	99.99%
50	10^{-5}	99.999%
60	10^{-6}	99.9999%
70	10^{-7}	99.99999%
80	10^{-8}	99.999999%
90	10^{-9}	99.9999999%
93	$10^{-9.3}$	99.99999995%

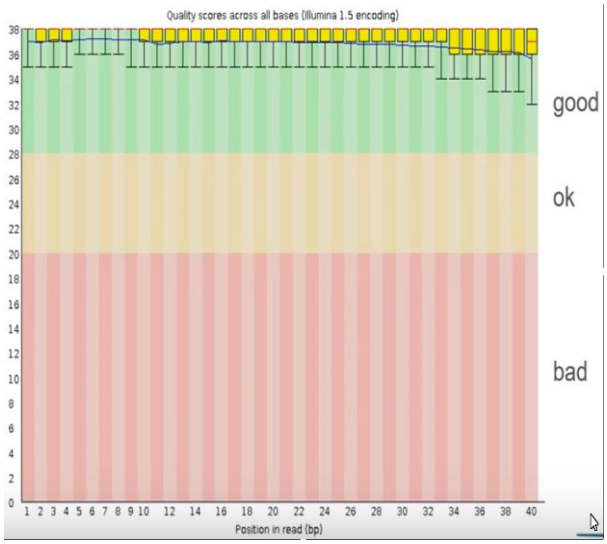
ASCII character	Decimal value	Phred score
!	33	0
"	34	1
#	35	2
$	36	3
⋮	⋮	⋮
A	65	22
B	66	23
⋮	⋮	⋮
x	120	87
y	121	88
z	122	89
{	123	90
|	124	91
}	125	92
~	126	93

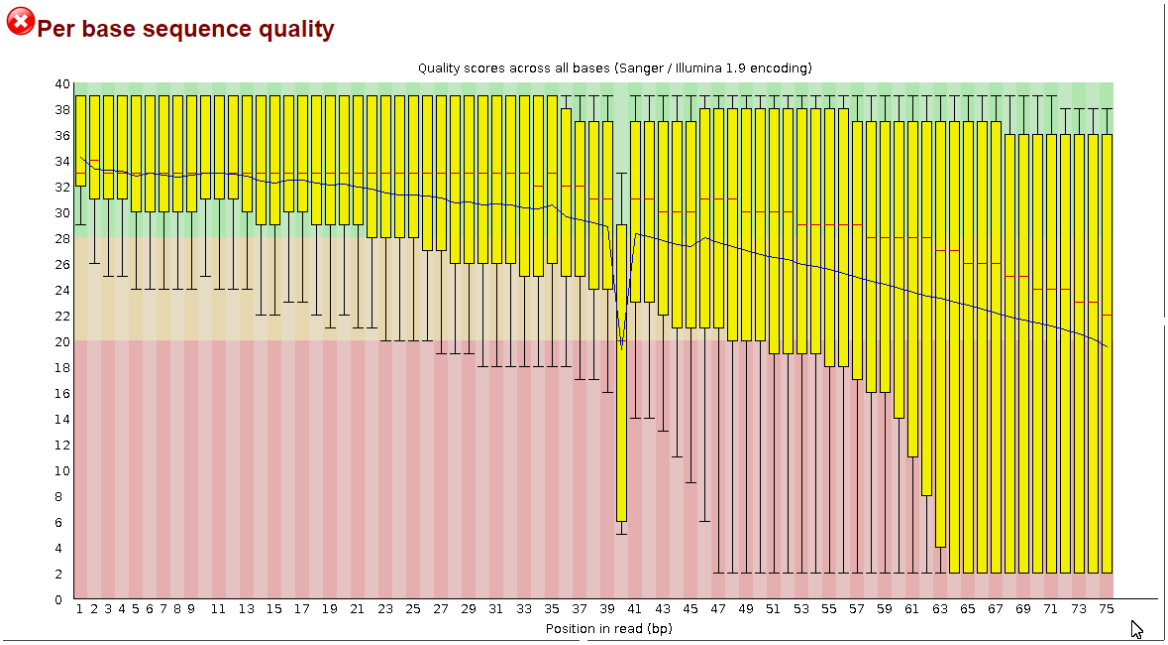
Dec	Hex	Znak	Dec	Hex	Znak	Dec	Hex	Znak
32	20	SPC	64	40	@	96	60	`
33	21	!	65	41	A	97	61	a
34	22	"	66	42	B	98	62	b
35	23	#	67	43	C	99	63	c
36	24	$	68	44	D	100	64	d
37	25	%	69	45	E	101	65	e
38	26	&	70	46	F	102	66	f
39	27	'	71	47	G	103	67	g
40	28	(	72	48	H	104	68	h
41	29	)	73	49	I	105	69	i
42	2a	*	74	4a	J	106	6a	j
43	2b	+	75	4b	K	107	6b	k
44	2c	,	76	4c	L	108	6c	l
45	2d	-	77	4d	M	109	6d	m
46	2e	.	78	4e	N	110	6e	n
47	2f	/	79	4f	O	111	6f	o
48	30	0	80	50	P	112	70	p
49	31	1	81	51	Q	113	71	q
50	32	2	82	52	R	114	72	r
51	33	3	83	53	S	115	73	s
52	34	4	84	54	T	116	74	t
53	35	5	85	55	U	117	75	u
54	36	6	86	56	V	118	76	v
55	37	7	87	57	W	119	77	w
56	38	8	88	58	X	120	78	x
57	39	9	89	59	Y	121	79	y
58	3a	:	90	5a	Z	122	7a	z
59	3b	;	91	5b	[	123	7b	{
60	3c	<	92	5c	\	124	7c	|
61	3d	=	93	5d	]	125	7d	}
62	3e	>	94	5e	^	126	7e	~
63	3f	?	95	5f	_	127	7f	DEL

I. Quality control of raw reads

2. Quality control of reads with FastQC. Select **hESC.fastq**, the tool **Quality control / Read quality with FastQC** and click **Run** (please do not analyze GM12878.fastq yet, we will use it later on for a workflow). Select the result file and viewing option **Open in external web browser**.
- How many reads are there and how long are they?
- Is the base quality good all along the reads?



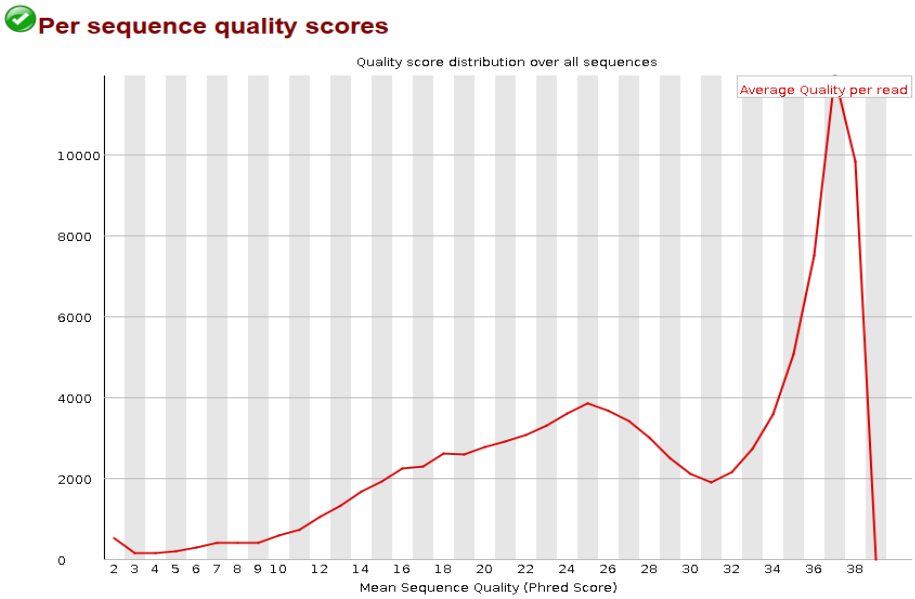




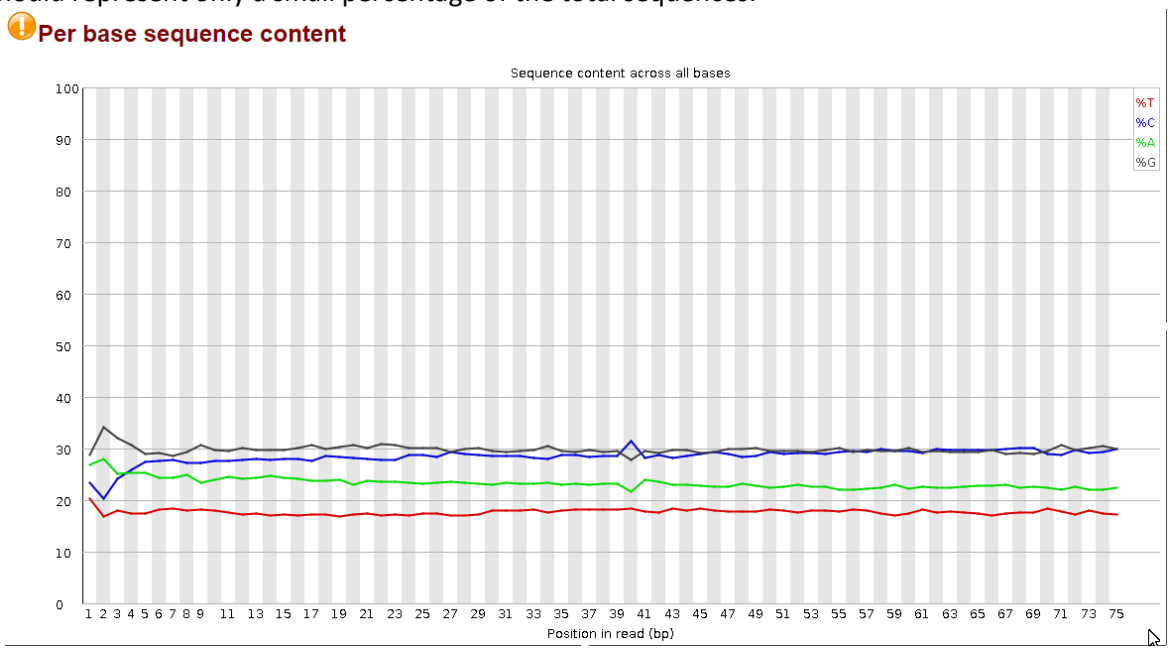
Base quality. The central red line of each box of the boxplot represents the median base quality value at a given read position. The yellow box represents the inter-quartile range (25-75%), and the upper and lower whiskers represent the 10% and 90% points. The blue line shows the mean quality. In this case, the overall base quality of the data cannot be regarded as good.

Comparison of common averages of values { 1, 2, 2, 3, 4, 7, 9 }

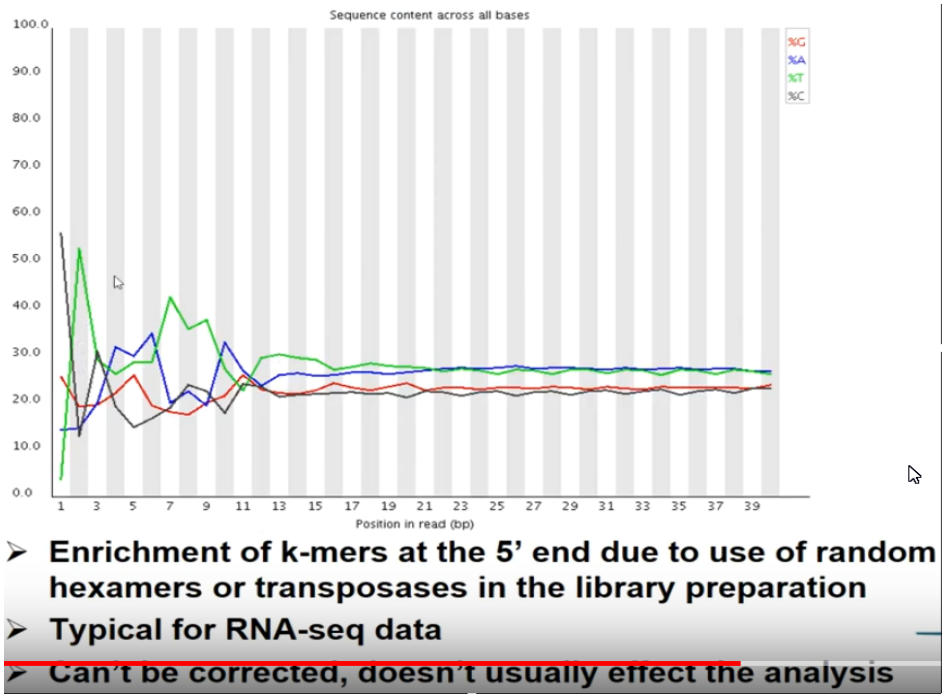
Type	Description	Example	Result
Arithmetic mean	Sum of values of a data set divided by number of values: $\bar{x}=\frac{1}{n}\sum_{i=1}^{n} x_i$	(1 + 2 + 2 + 3 + 4 + 7 + 9) / 7	**4**
Median	Middle value separating the greater and lesser halves of a data set	1, 2, 2, **3**, 4, 7, 9	**3**
Mode	Most frequent value in a data set	1, **2**, **2**, 3, 4, 7, 9	**2**



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 (on the edge of the field of view etc.), however these should represent only a small percentage of the total sequences.



Some types of library preparation produce a biased sequence composition, normally at the start of the read. For instance, cDNA synthesis using random hexamer priming induces biases in the nucleotide composition at the beginning of transcriptome sequencing reads. This is not necessarily the case with WES or WGS data, but libraries fragmented using transposases inherit an intrinsic bias in the positions at which reads start. Other reasons for deviations from uniformity of per base sequence content include overrepresented sequences such as adapters.



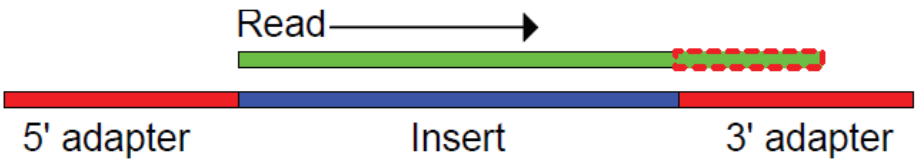
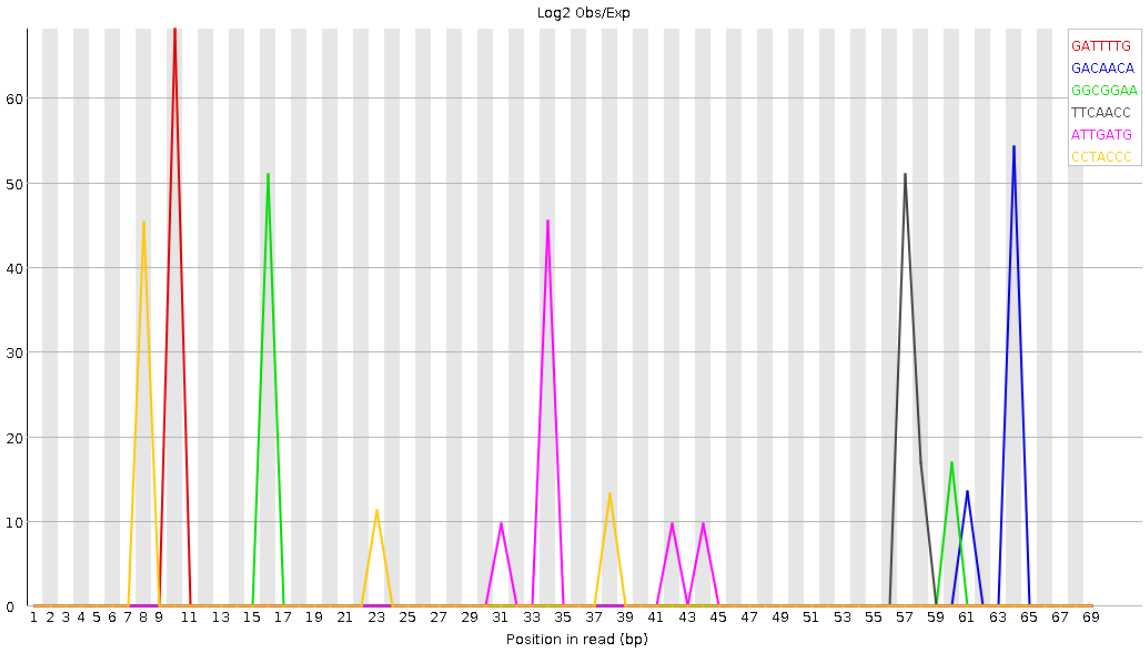


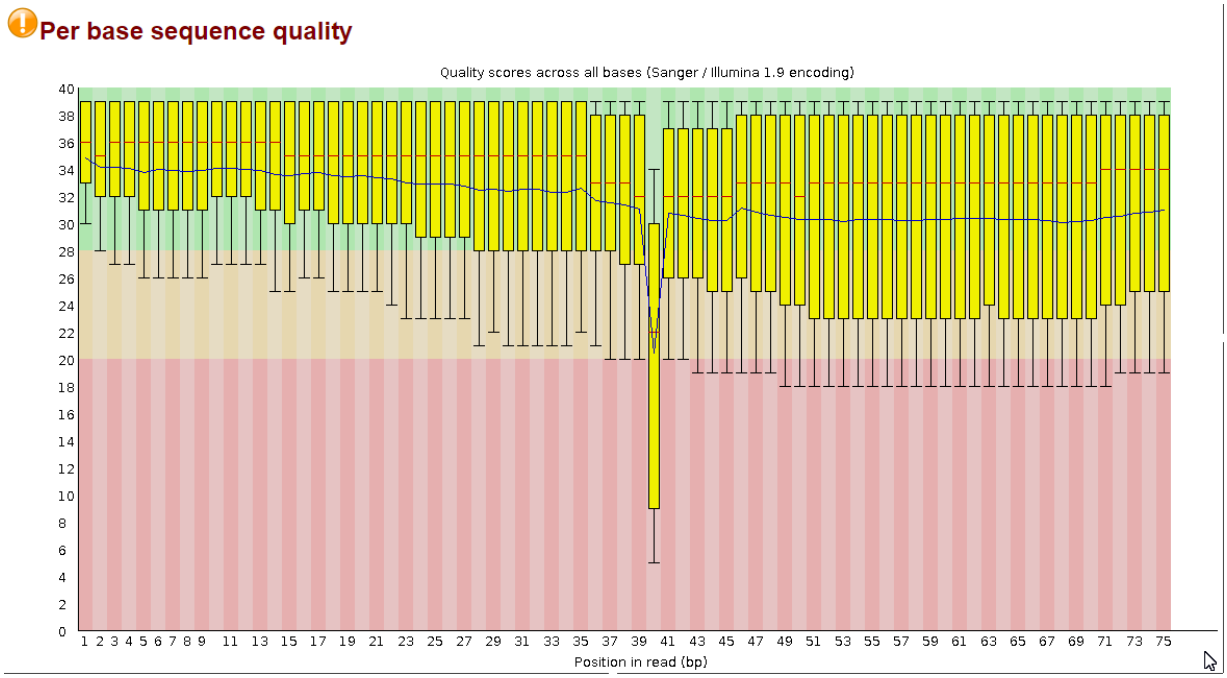
Figure 6.8. Adapter contamination. Illumina read running into the 3' adapter. If the length of the read is longer than the length of the specific insert, then the 3' end of the read will originate from the adapter sequence on the other side (for example, the read length is 100 and the insert was only 80 nucleotides long, then the last 20 base pairs, as indicated here with a dashed red line, will be adapter contamination).

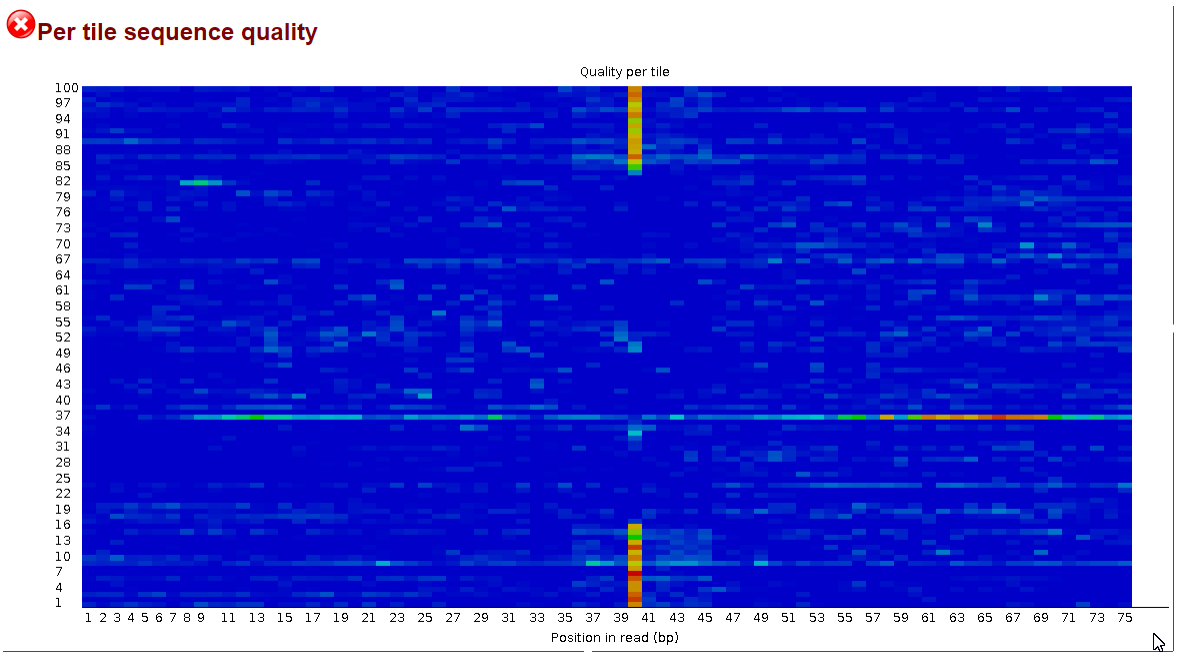
Kmer Content

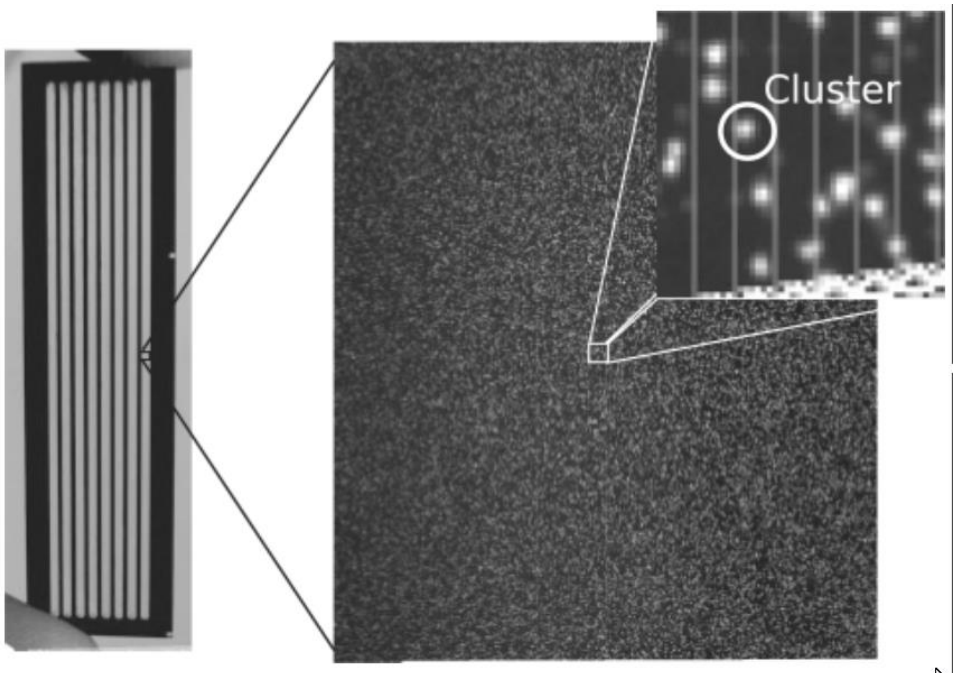


II. Preprocessing (trimming / filtering) if needed

3. Trim reads based on base quality with Trimmomatic (note that this step is optional in real life). Select **hESC.fastq** and the tool **Preprocessing / Trim reads with Trimmomatic** and set the parameters: **Minimum quality to keep a trailing base** = **5** and **Minimum length of reads to keep = 50**.
- How many reads get discarded (check the file hESC-trimlog.txt)? Select **hESC_trimmed.fq.gz** containing trimmed reads and run the tool **Read quality with FastQC** as before.
- Does the base quality look better now?







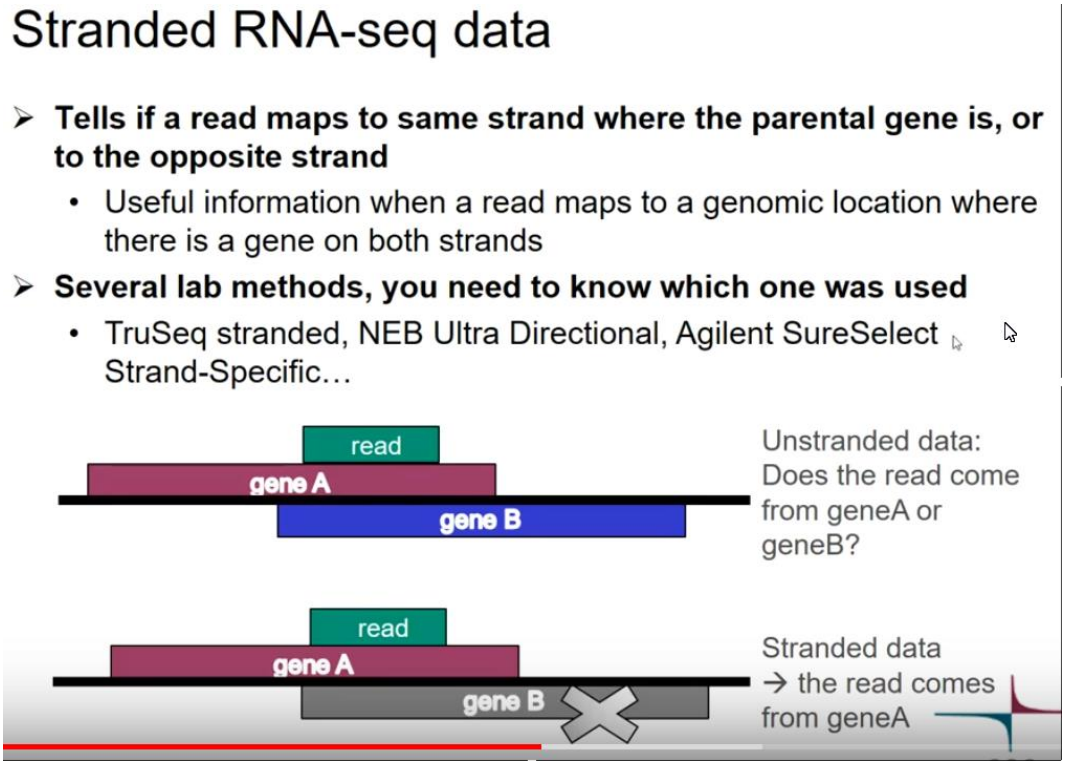
A Genome Analyzer flowcell (left) and imaging region or 'tile' (right), with a magnified section showing a cluster. Images have been normalized, to span the full grey-scale range, for illustration purposes.

6.7 PER TILE SEQUENCE QUALITY

Beside the sample quality itself also the sequencing process may have impact on the quality of the reads. Issues such as bubbles passing the flow cell or damaged (smudged or scratched) flow cells may result in loss of information or a drop in quality. This can be seen in the heatmap

Filtering vs trimming

- **Filtering removes the entire read**
- **Trimming removes only the bad quality bases**
  - It can remove the entire read, if all bases are bad
- **Trimming makes reads shorter**
  - This might not be optimal for some applications
- **Paired end data: the matching order of the reads in the two files has to be preserved**
  - If a read is removed, its pair has to removed as well



4. Check the strandedness of the reads. Select **hESC_trimmed.fq.gz** and run the tool **Quality control / RNA-seq strandedness inference and inner distance estimation using RseQC** (check that you have correct genome selected). Open the resulting **experiment_data.txt**.
- Is the data stranded? From which strand are the reads from? Mark down the parameters for Tophat and HTSeq.

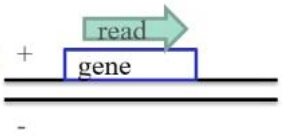
experiment_data.txt:
This is SingleEnd Data
Fraction of reads failed to determine: 0.0428
Fraction of reads explained by "++,--": 0.9505
Fraction of reads explained by "+-,-+": 0.0068

It seems the data is **stranded**. Read is always on the same strand as the gene.

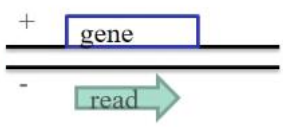
Corresponding parameters are:
TopHat, HISAT2, Cufflinks and Cuffdiff: **library-type fr-secondstrand**
HTSeq: **stranded -- yes**
RSeQC: **++,--**

Single end:

++,--
read mapped to '+' strand indicates parental gene on '+' strand
read mapped to '-' strand indicates parental gene on '-' strand



+-,-+
read mapped to '+' strand indicates parental gene on '-' strand
read mapped to '-' strand indicates parental gene on '+' strand



Paired end:

1++,1–,2+-,2-+

read1 mapped to '+' strand indicates parental gene on '+' strand
read1 mapped to '-' strand indicates parental gene on '-' strand
read2 mapped to '+' strand indicates parental gene on '-' strand
read2 mapped to '-' strand indicates parental gene on '+' strand

1+-,1-+,2++,2--

read1 mapped to '+' strand indicates parental gene on '-' strand
read1 mapped to '-' strand indicates parental gene on '+' strand
read2 mapped to '+' strand indicates parental gene on '+' strand
read2 mapped to '-' strand indicates parental gene on '-' strand

III. Alignment (= mapping) to reference genome

5. Align reads to reference genome using HISAT2. Select **hESC_trimmed.fq.gz** and run the tool **Alignment / HISAT2 for single end reads** setting **genome = Homo_sapiens.GRCh38.92** and **Library type = fr-secondstrand**. Running takes about 15 min.

- What was the overall alignment rate and how many reads have multiple alignments (hisat.log)?
- Inspect the BAM file in the **BAM viewer**. What is the mapping quality of the fourth read? How many alignments does it have (check the NH tag)? Is it a spliced read (check the CIGAR field for Ns)?

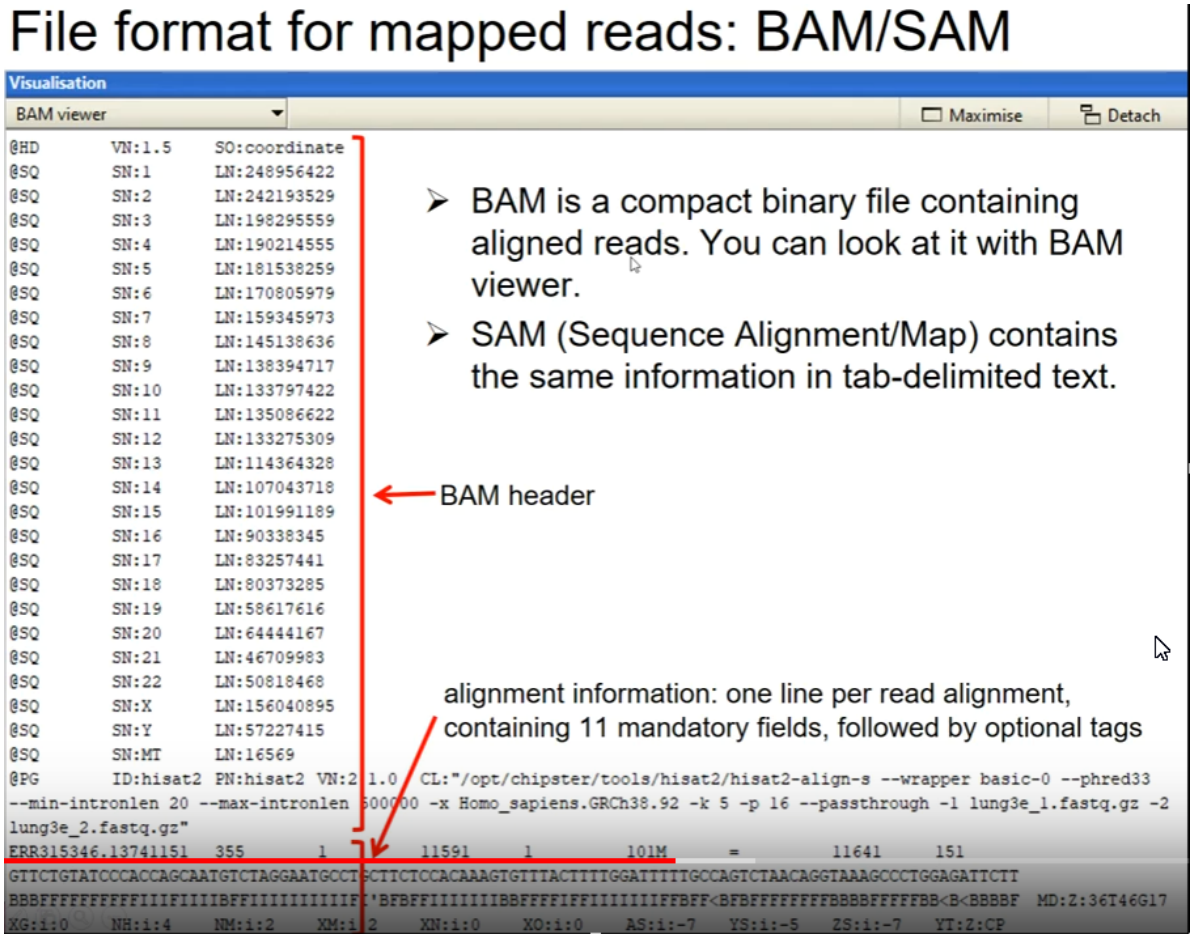


Table 9.1. Mandatory fields of the SAM Format.

Col	Field	Description	Example
1	QNAME	Query template NAME	read_1
2	FLAG	Bitwise FLAG	0
3	RNAME	Reference sequence NAME	chrE
4	POS	Left-most mapping POSition (1-based)	11
5	MAPQ	MAPping Quality	37
6	CIGAR	CIGAR string	10M
7	RNEXT	Ref. name of the mate or NEXT read	*
8	PNEXT	Position of the mate or NEXT read	0
9	TLEN	Observed Template LENgth	0
10	SEQ	Segment SEQuence	ACGCATACTG
11	QUAL	Base QUALity string	DIGAFHHBCA

Note: Each line in the alignment section of a SAM file comprises 11 mandatory fields.

http://broadinstitute.github.io/picard/explain-flags.html

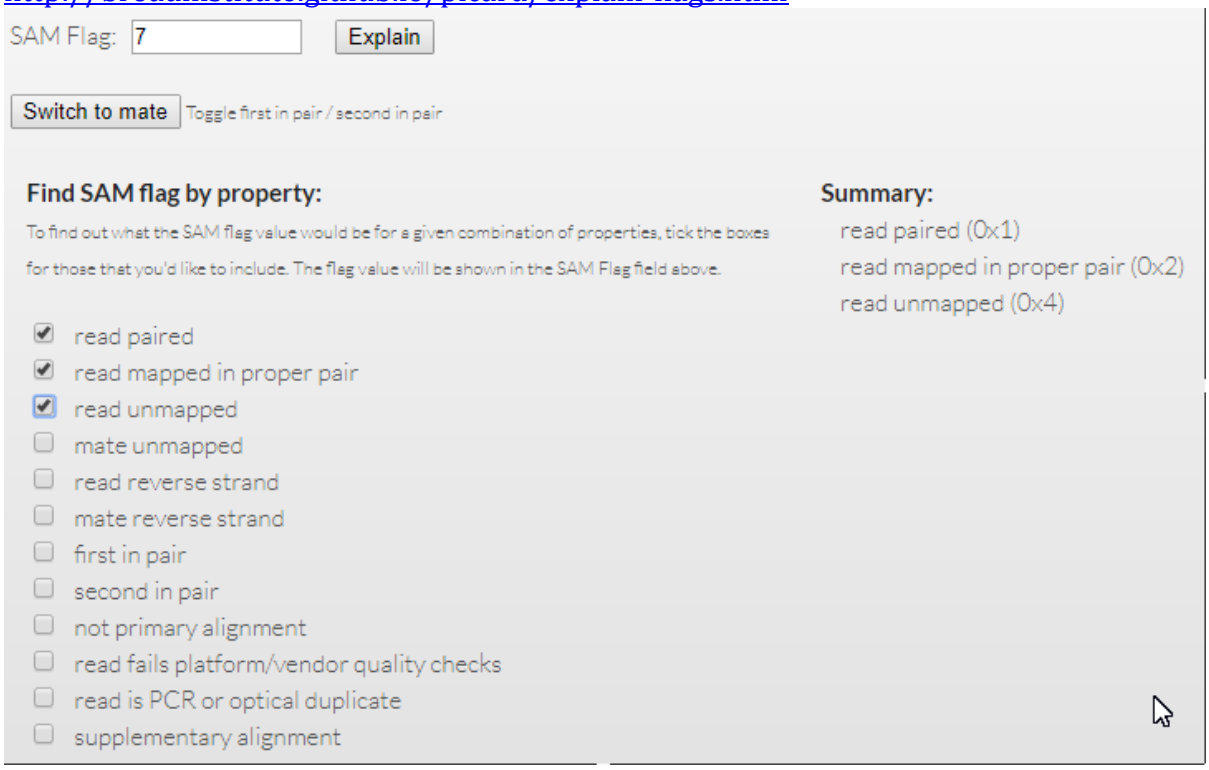
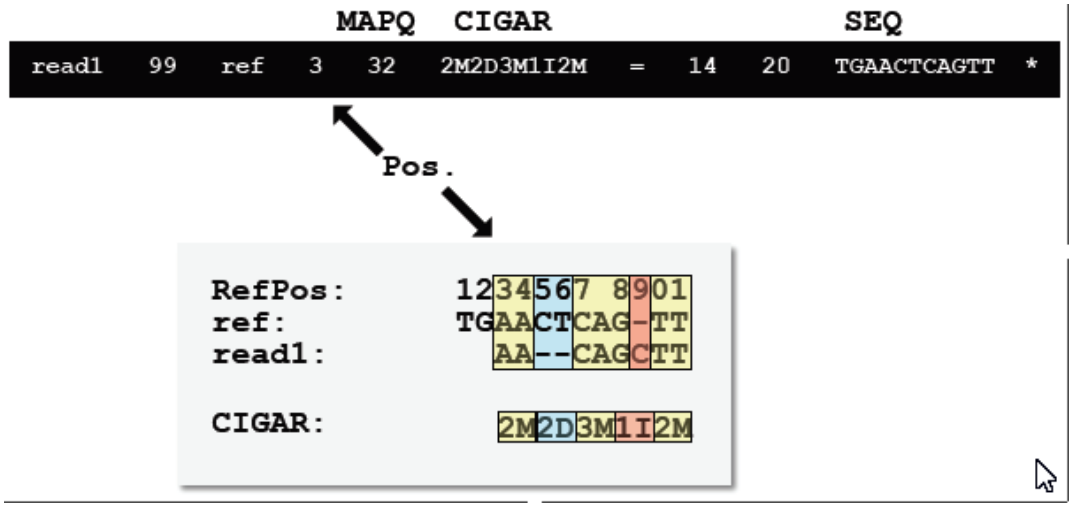


Table 9.2. CIGAR operations

Op	Description
M	alignment match (sequence match or mismatch)
I	insertion (additional non-reference base)
D	deletion (reference base missing in the read)
N	skipped region from the reference
S	soft clipping (clipped sequences still present in SEQ)
H	hard clipping (clipped sequences not present in SEQ)
P	padding (silent deletion from padded reference)
=	sequence match
X	sequence mismatch



- Inspect the BAM file in the **BAM viewer**. What is the mapping quality of the fourth read? How many alignments does it have (check the NH tag)? Is it a spliced read (check the CIGAR field for Ns)?

```
HWI-EAS229_1:2:40:1280:283/1  272    1      18506  1      49M6183N26M    *      0      0
        AGGGCCGATCTTGGTGCCATCCAGGGGGCCTCTACAAGGATAATCTGACCTGCTGAAGATGTCTCCAGAGACCTT
        ECC@EEF@EB:EECFEECCCBEEEE;>5;2FBB@FBFEEFCF@FFFFCEFFFFEE>FFEFC=@A;@>1@6.+5/5
        MD:Z:75 XG:i:0  NH:i:5  NM:i:0  XM:i:0  XN:i:0  XO:i:0  AS:i:0  XS:A:+  ZS:i:0  YT:Z:UU
```

If x = 1 in NH:i:x, i.e., when the alignment is unique, then MAPQ is calculated according

$$Q = -10 \log_{10} p \Leftrightarrow p = 10^{\frac{-Q}{10}}$$

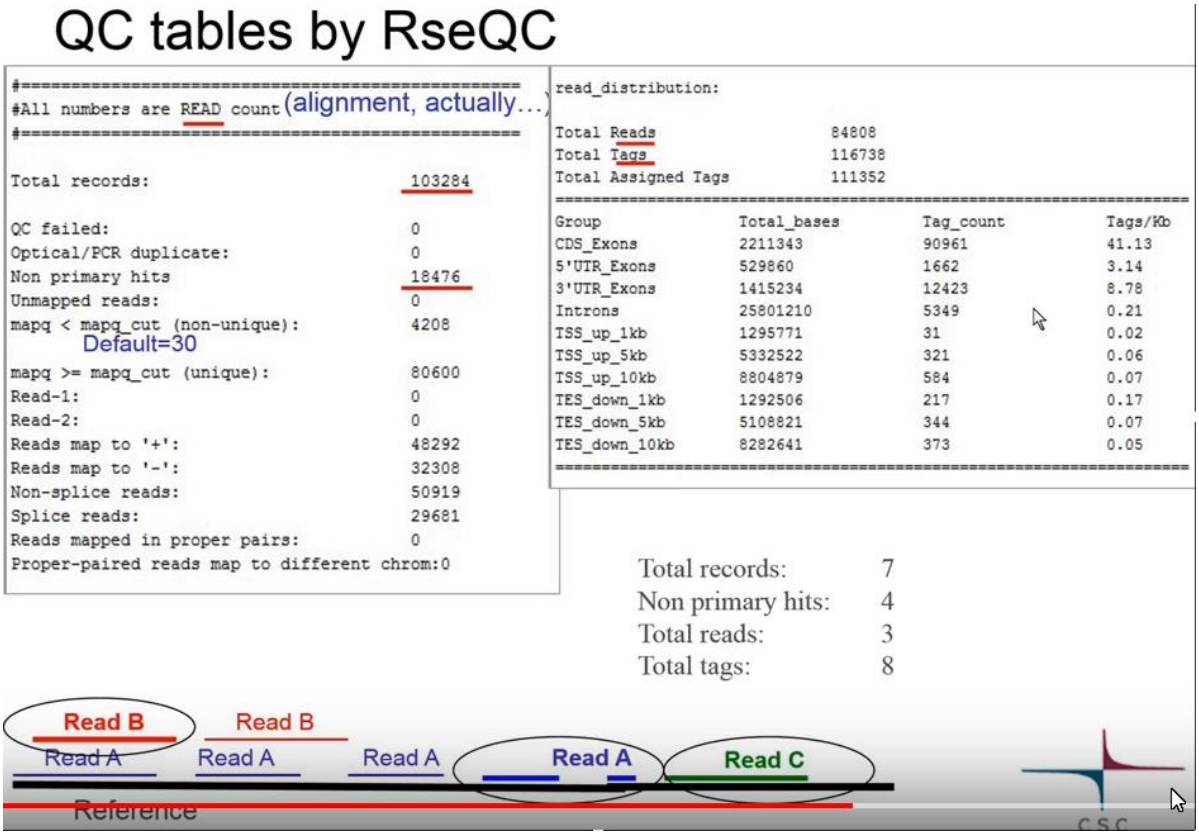
where ***p*** is the probability that the corresponding base call is **wrong**. When NH:i:x > 1, i.e., when the read is aligned to multiple locations on the reference genome, when the alignment is non-unique, then ***Q*** = 1 provided that the read is well aligned with just few mismatches, otherwise it is 0.

Table 9.4. SAM format: data types for the optional tags

Type	Description
A	Single character
Z	String
i	Signed 32-bit integer
f	Single-precision float (real number)
H	Hexadecimal number string
B	General array

IV. Alignment level QC

6. Perform alignment level quality check with RseQC. Select **hESC.bam** and the tool **Quality control / RNA-seq quality metrics with RseQC**. In parameters set **organism = Homo_sapiens.GRCh38.92**.
- Inspect the result file **hESC_rseqc.txt**. How many alignments does the BAM file contain? Is the tag (~read) density higher in exons than in introns?
- Inspect the result file **hESC_rseqc.pdf**. Is the coverage uniform along transcripts (check the first plot)? Were novel splice junctions found (check the splice junctions plot)?



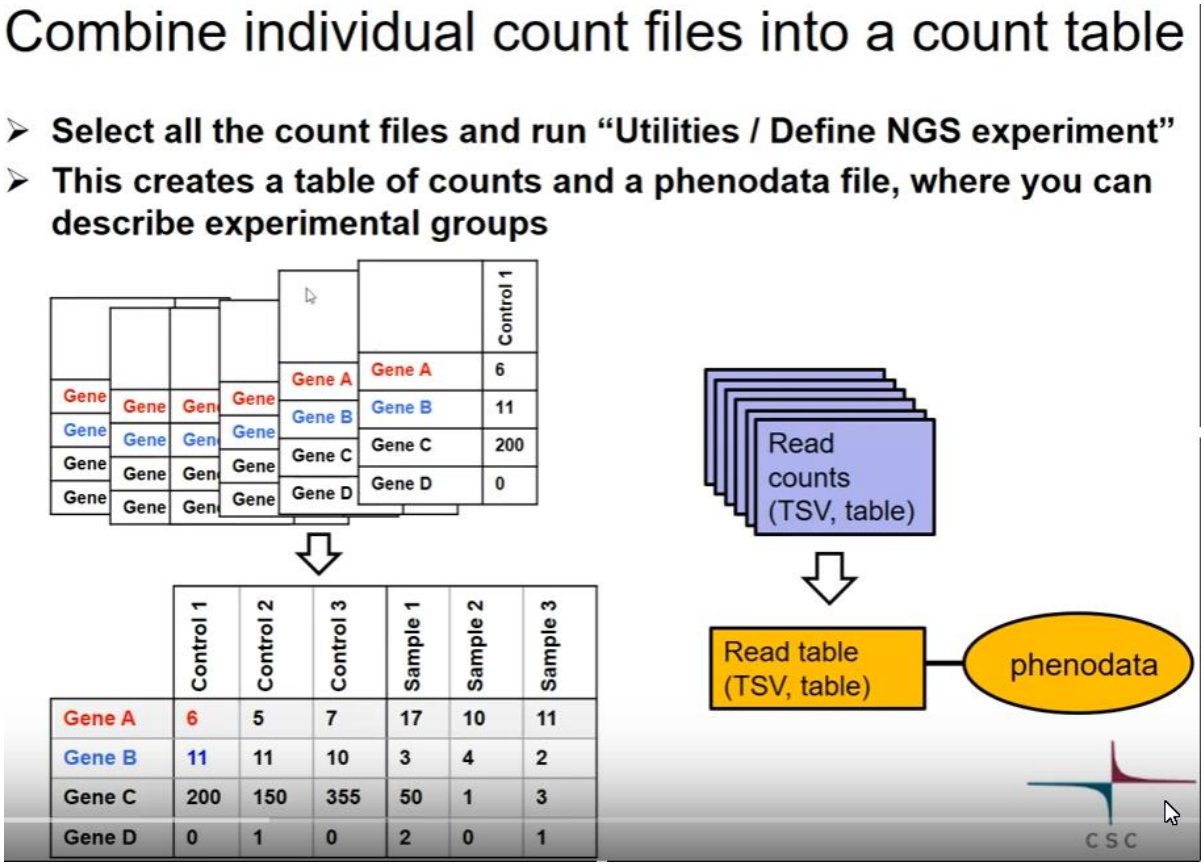
V. Quantitation

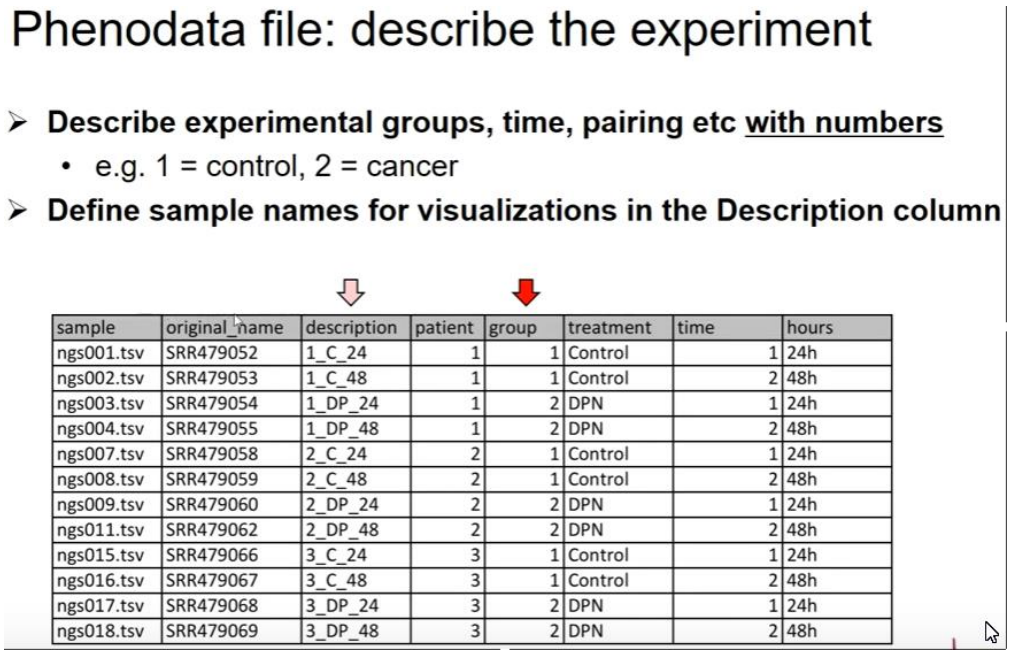
7. Count reads per genes using HTSeq. Select the **BAM** file and the tool **RNA-seq / Count aligned reads per genes with HTSeq**. Set the parameter **Is the data stranded and how** = "**yes" in HTSeq**.
- Inspect the result files. Which file contains the read counts per each gene? Can you find genes with counts (note that you can sort the table by clicking on the title of the count column)?
- How many alignments were not counted for any gene (check htseq-count-info.txt)?

8. Save session, get analysis history file, save and run an analysis workflow. Save session: Select **File / Save local session**. Give a name to your session and save it. Save a textual report: Select **hESC.tsv** and click on **Analysis history** in the visualization panel. Save an automatic workflow: Select file **hESC.fastq** and **Workflow / Save starting from selected**. Run workflow: Select **GM12878.fastq** and **Workflow / Run recent / yourName.bsh**.

VI. Describing the experiment with phenodata

9. Create count table and description file for the experiment. Select **both tsv files** containing the read counts, and the tool **Utilities / Define NGS experiment**. Set the parameters **Does your data contain genomic coordinates** = **yes** and **Count column** = **count**. In the resulting **phenodata.tsv** file, fill in the **group** column: enter **1** for hESC and **2** for GM12878. Save as local session.





sample	original_name	description	patient	group	treatment	time	hours
ngs001.tsv	SRR479052	1_C_24	1	1	Control	1	24h
ngs002.tsv	SRR479053	1_C_48	1	1	Control	2	48h
ngs003.tsv	SRR479054	1_DP_24	1	2	DPN	1	24h
ngs004.tsv	SRR479055	1_DP_48	1	2	DPN	2	48h
ngs007.tsv	SRR479058	2_C_24	2	1	Control	1	24h
ngs008.tsv	SRR479059	2_C_48	2	1	Control	2	48h
ngs009.tsv	SRR479060	2_DP_24	2	2	DPN	1	24h
ngs011.tsv	SRR479062	2_DP_48	2	2	DPN	2	48h
ngs015.tsv	SRR479066	3_C_24	3	1	Control	1	24h
ngs016.tsv	SRR479067	3_C_48	3	1	Control	2	48h
ngs017.tsv	SRR479068	3_DP_24	3	2	DPN	1	24h
ngs018.tsv	SRR479069	3_DP_48	3	2	DPN	2	48h

VII. Experiment level QC

It will be discussed in Experiments 2 and 3.

VIII. Differential expression analysis

10. Detect differentially expressed genes with edgeR. Select the file **ngs-data-table.tsv** and run the tool **RNA-seq / Differential expression using edgeR**.
- How many differentially expressed genes are detected (check the number of rows in de-list-edger.tsv)?

IX. Visualization of reads and results in genomic context

11. Annotate the list of differentially expressed genes. Select the file **de-list-edger.tsv** and run the tool **Utilities / Annotate Ensembl identifiers**.
- Which gene has the highest positive fold change?

12. Visualize differentially expressed genes in Chipster genome browser. We will use the file **de-list-edger.bed** as a navigation aid. Visualize it as a **spreadsheet**, and click **Detach** to open it in a separate window. Sort it by fold change (column4) so that the gene with the highest positive fold change is at the top. Put this new window aside for a moment. Select again **de-list-edger.bed**, both BAM files and the visualization method **Genome browser**. **Maximize** the visualization panel, select **genome = Homo sapiens hg38**, and click **Go**.
- Click on the start position of the first gene in the detached BED file to navigate to that gene. In the **Settings** tab, change coverage scale to 1000. Zoom in and out with a mouse wheel. You can turn off the reads (in the Options, untick the box Reads) and view just the coverage first.
- Does this gene (EEF2) seem to be differentially expressed? Are all the reads located in exons?