RNA-seq data analysis:

How to find differentially expressed genes? Using command line tools + R

Laxmana Yetukuri, Maria Lehtivaara
CSC – IT Center for Science, Finland
chipster@csc.fi

Welcome!

Some practical matters:

- **Keycard**
  - Please keep it with you at all times
  - Lunch ticket
  - QR reader on the door –knowing this, you can leave your belongings in the classroom at your own risk
- **Parking**
  - You need to get a permission from Info desk
- **Schedule**
  - Is on the course webpage –not set in stone ☺
- **GDPR!**
  - Careful when using the classroom computers
- **Foods & drinks**
  - We don’t allow those in the classroom –water bottles ok ☺
  - Coffee/tea breaks in the training lobby
  - Lunch at the two restaurants in this building

Schedule (draft)

- Thursday 6.2.
  - 9:00 First session: Welcome & Introductions
  - **10:00 Coffee break**
  - 10:30 Second session: Quality control and preprocessing
  - **12:00 Lunch**
  - 13:00 Third session: Alignment
  - **14:30 Coffee break**
  - 15:00 Fourth session: Quantitation, Experimental design, wrap up for the day

- Friday 7.2.
  - 9:00 First session: Differential expression analysis in R
  - **10:00 Coffee break**
  - 10:30 Second session: Annotations and enrichment analysis
  - **12:00 Lunch**
  - 13:00 Third session: Analysing in Puhti + Allas
  - **14:30 Coffee break**
  - 15:00 Fourth session: Other topics + wrap up

Understanding your data analysis - why?

- **You know your own experiments best**
  - Biology involved (e.g. genes, pathways, etc)
  - Potential batch effects etc

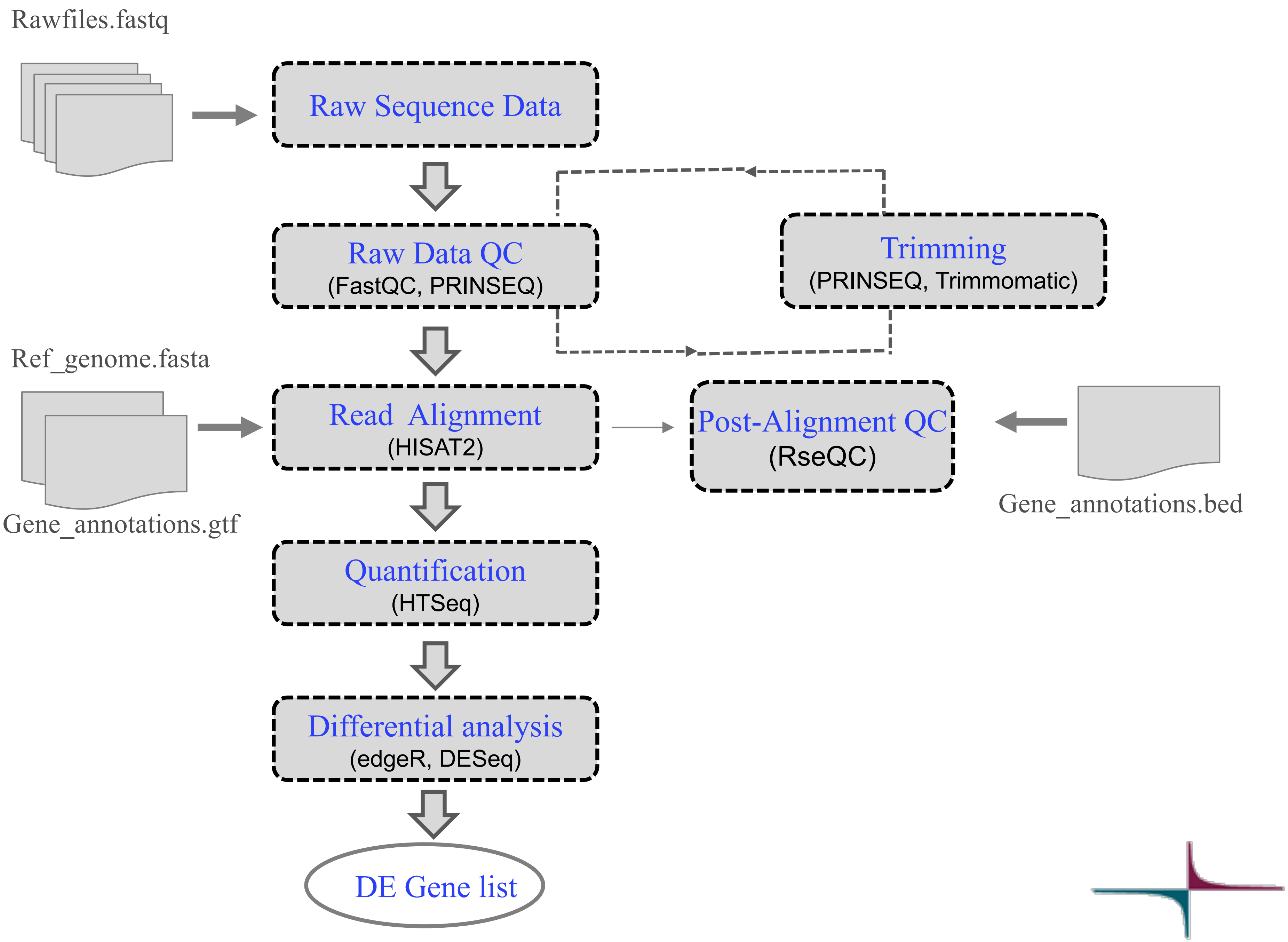
- **You can tune the parameters, ”play around” and learn more about your data**
  - Bioinformaticians might not always be available when needed

- **Allows you to design experiments better**
  - Enough replicates, reads etc → less money wasted

- **Allows you to discuss more easily with bioinformaticians**

What will I learn?

- **Introduction to RNA sequencing**
- **The basics in <u>differential gene expression analysis</u>**
  - Central concepts
  - Analysis steps
  - File formats
- **How to operate bunch of tools used in the exercises**
  - In command line (we use virtual machine that mimics CSCs Puhti environment)
  - In R (R included in the VM)
- **How to do the analysis effectively: running a batch job**
  - In CSC's Puhti supercomputer
- **Things to take into account when designing experiments**



Rawfiles.fastq

Raw Sequence Data

Raw Data QC
(FastQC, PRINSEQ)

Trimming
(PRINSEQ, Trimmomatic)

Ref_genome.fasta

Read Alignment
(HISAT2)

Post-Alignment QC
(RseQC)

Gene_annotations.gtf

Gene_annotations.bed

Quantification
(HTSeq)

Differential analysis
(edgeR, DESeq)

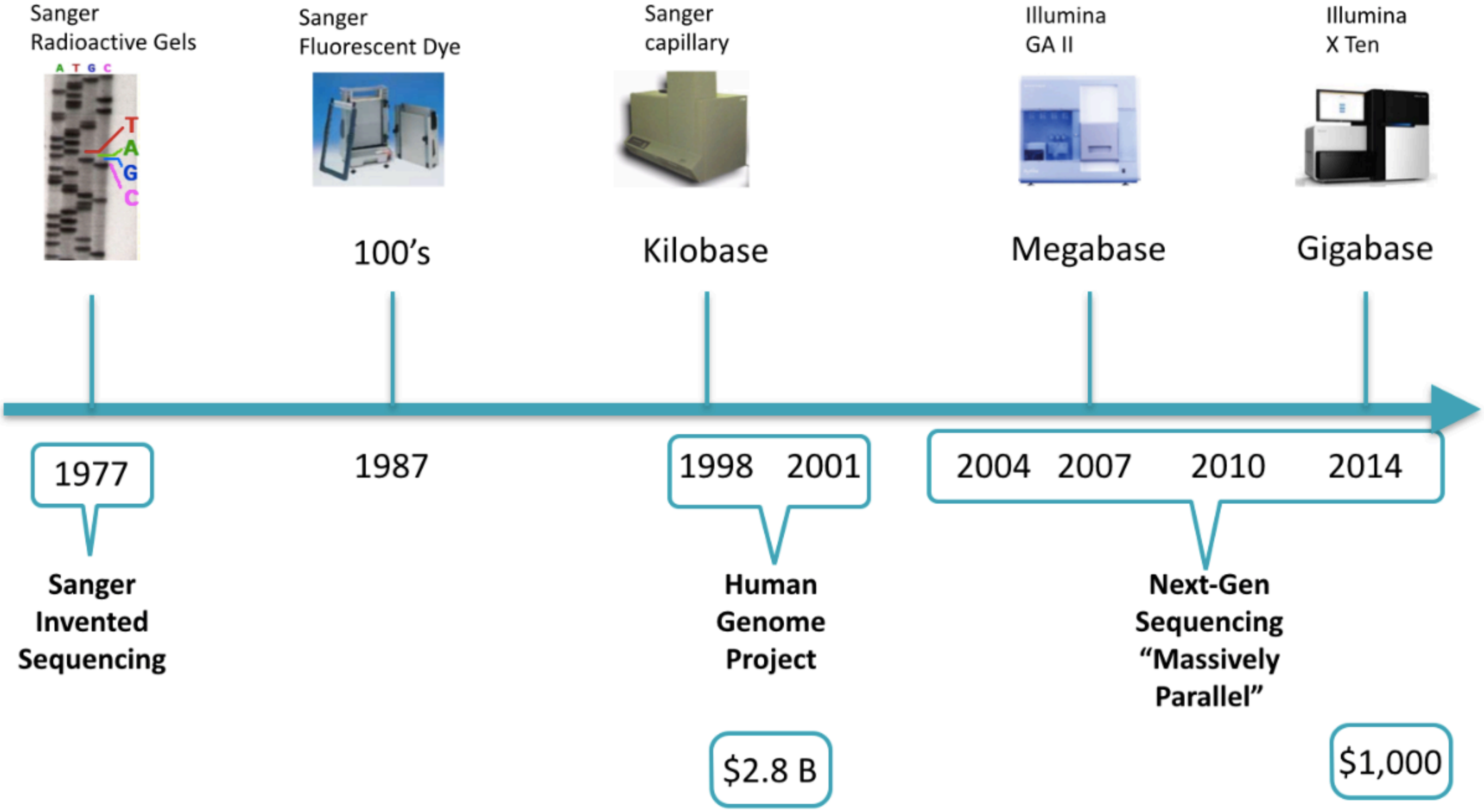
DE Gene list

Introduction to RNA-seq

What can I investigate with RNA-seq?

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

Development of sequencing methods



Sanger Radioactive Gels

Sanger Fluorescent Dye

Sanger capillary

Illumina GA II

Illumina X Ten

100's

Kilobase

Megabase

Gigabase

1977

1987

1998 2001

2004 2007 2010 2014

Sanger Invented Sequencing

Human Genome Project

Next-Gen Sequencing "Massively Parallel"

$2.8 B

$1,000

https://databricks.com/blog/2016/05/24/genome-sequencing-in-a-nutshell.html

Sequencing technologies

- **Sequencing by synthesis** → **Illumina**
- Pyrosequencing → Sanger, 454
- Ion semiconductor sequencing → Ion Proton
- Sequencing by ligation → SOLiD system
- Single molecule real time sequencing → PacBio

Illumina MiSeq

Illumina HiSeq

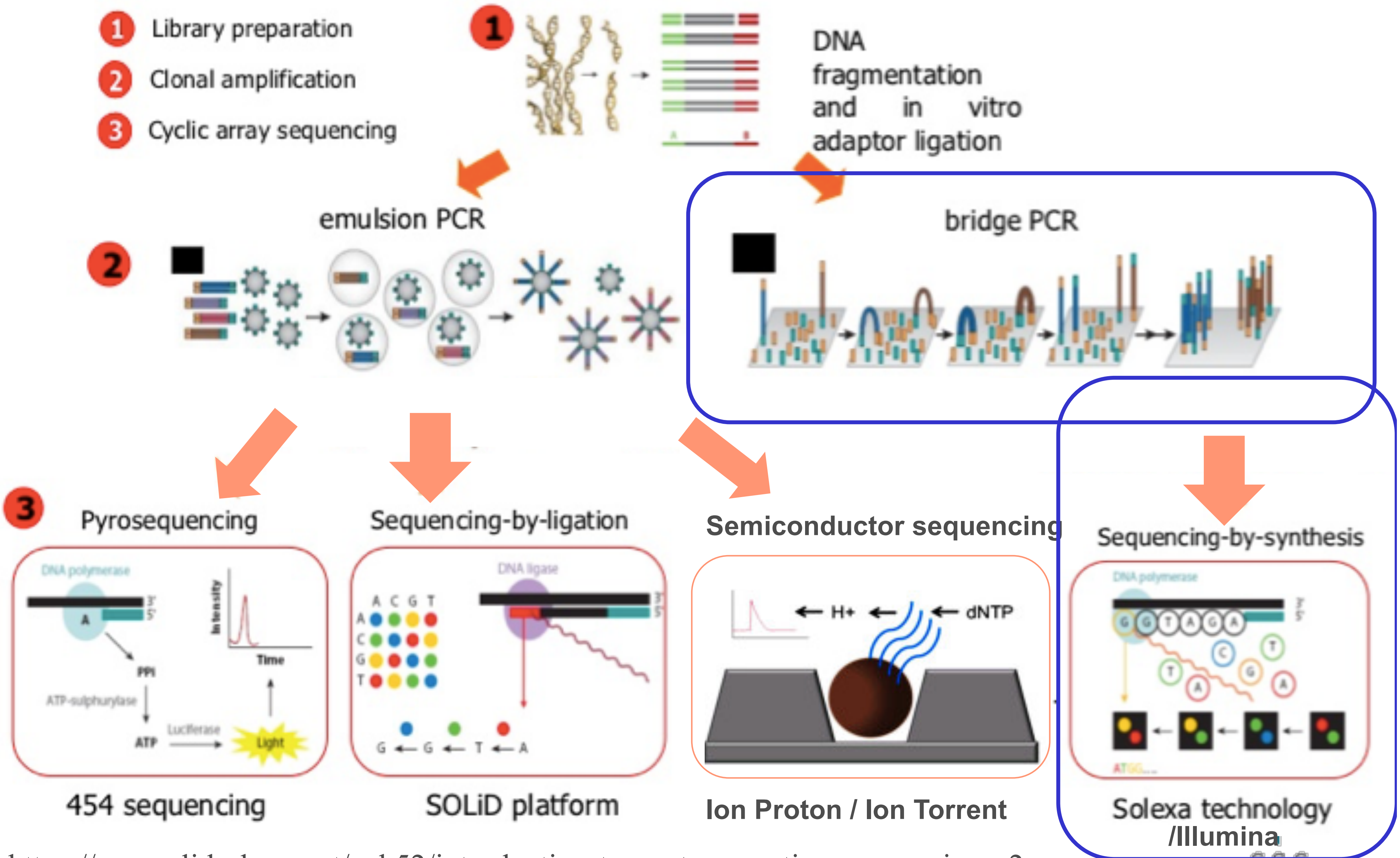
454 Sequencer

SOLiD system

PacBio

Ion Proton

Next-generation DNA sequencing



1 Library preparation

2 Clonal amplification

3 Cyclic array sequencing

DNA fragmentation and in vitro adaptor ligation

emulsion PCR

bridge PCR

Pyrosequencing

Sequencing-by-ligation

Semiconductor sequencing

Sequencing-by-synthesis

454 sequencing

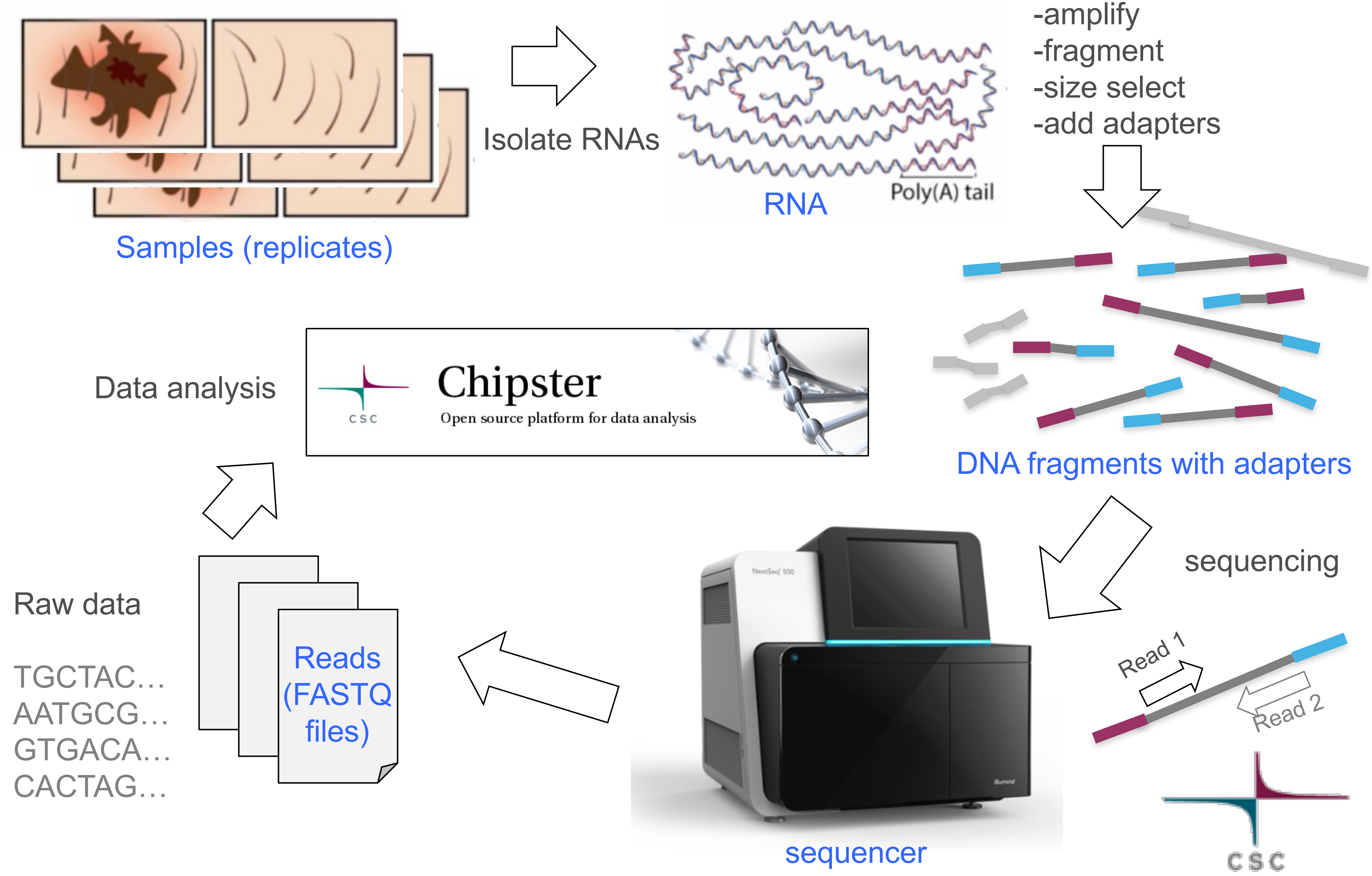
SOLiD platform

Ion Proton / Ion Torrent

Solexa technology /Illumina

https://www.slideshare.net/ueb52/introduction-to-next-generation-sequencing-v2

How is the data produced?



Samples (replicates)

Isolate RNAs

RNA

-generate cDNA
-amplify
-fragment
-size select
-add adapters

DNA fragments with adapters

sequencing

sequencer

Reads (FASTQ files)

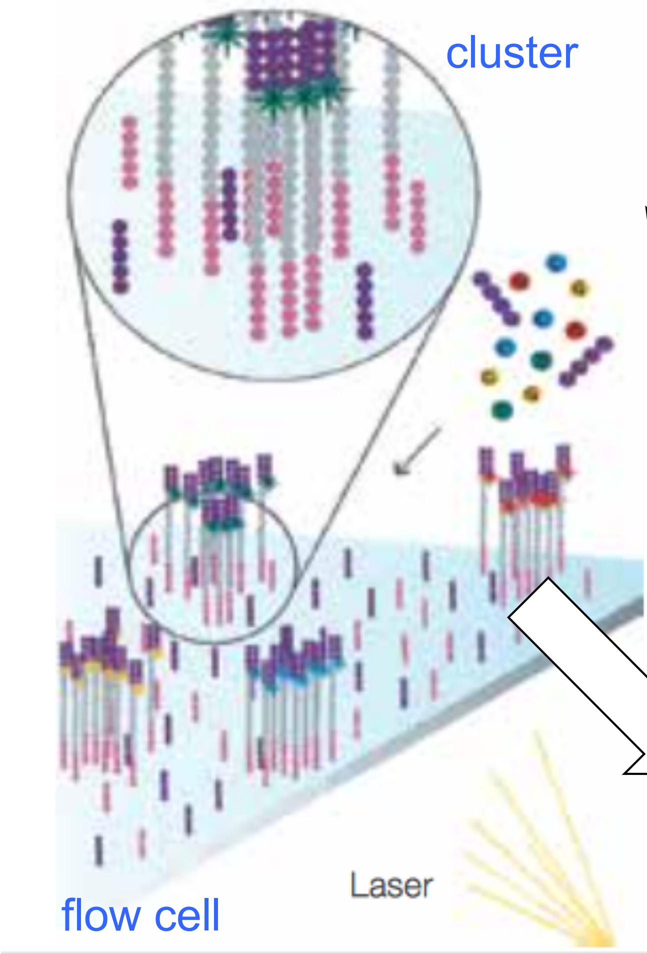
Raw data

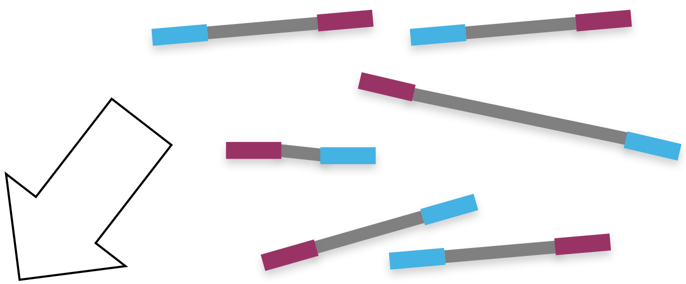
TGCTAC…
AATGCG…
GTGACA…
CACTAG…

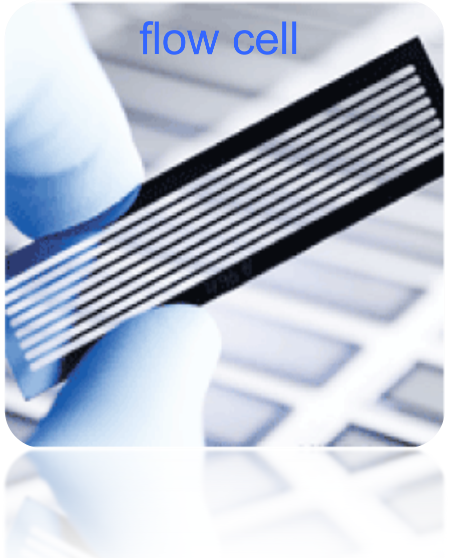
Data analysis

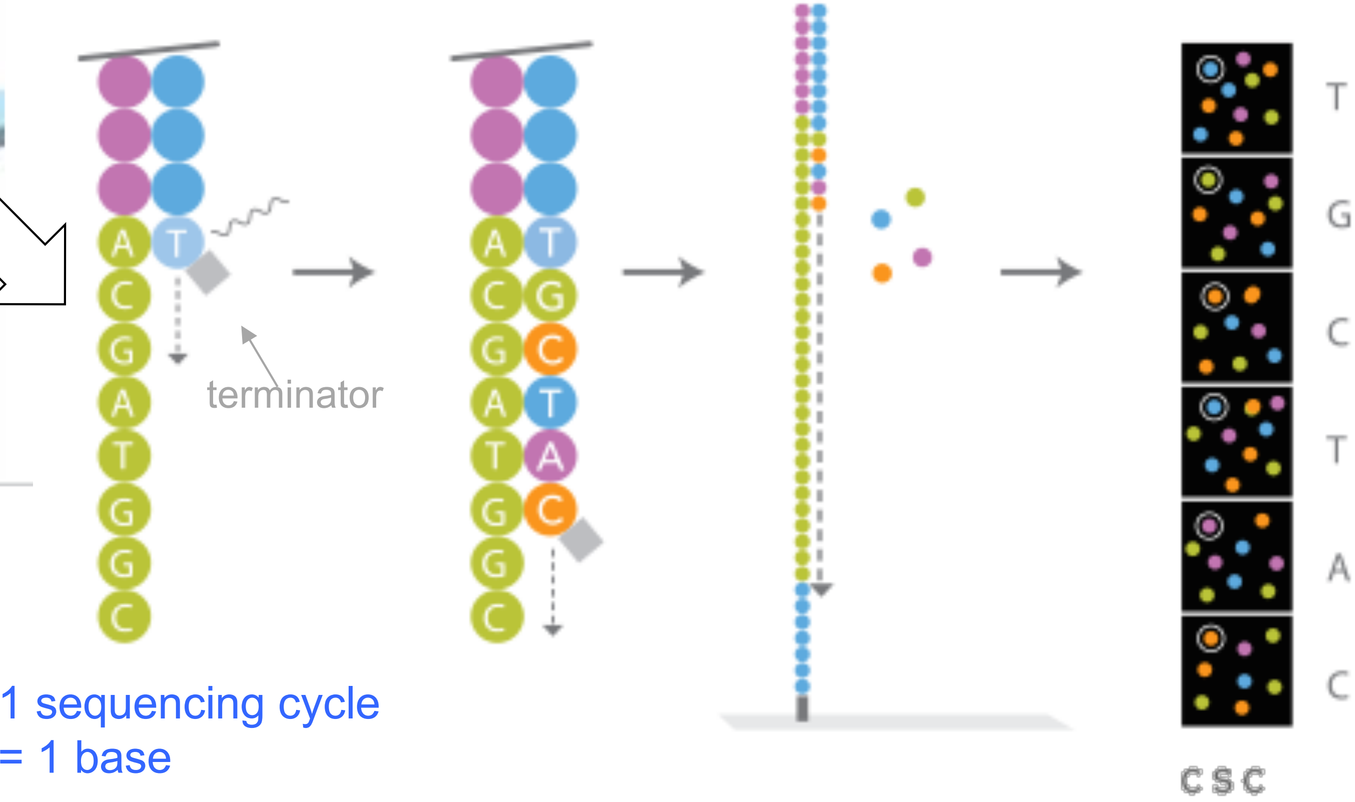
Chipster
Open source platform for data analysis

Sequencing by synthesis (Illumina)







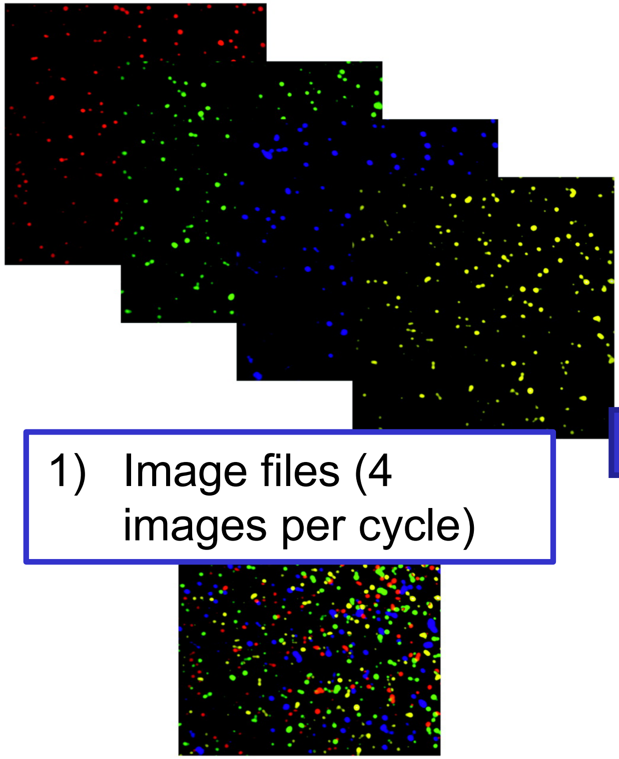


https://www.illumina.com/documents/products/techspotlights/techspotlight_sequencing.pdf

1 sequencing cycle
= 1 base

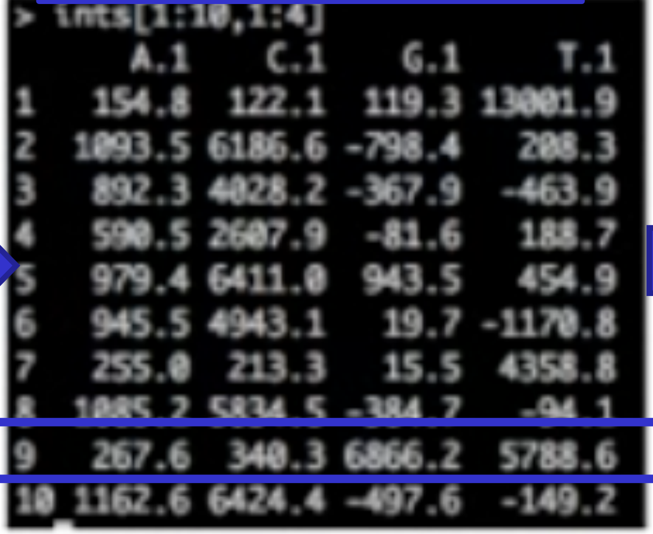
Sequencing by synthesis (Illumina)

- **From images to FASTQ file**



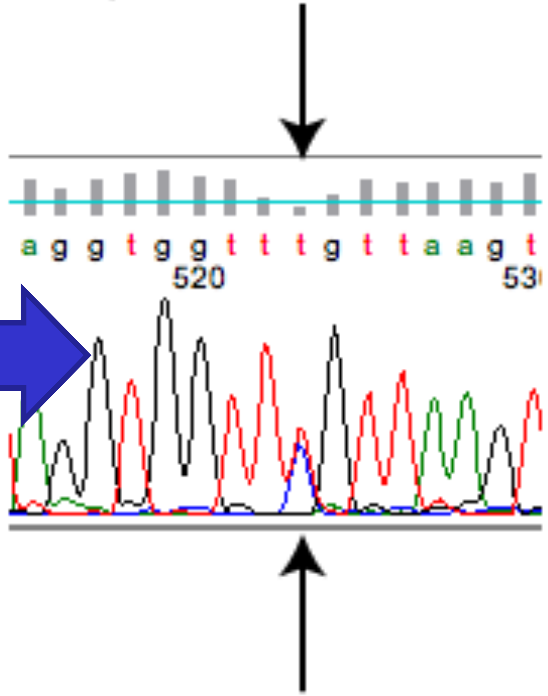
1) Image files (4 images per cycle)

2) Intensity table (4 values for each cluster & each cycle)



```
> ints[1:10,1:4]
      A.1    C.1    G.1     T.1
1   154.8  122.1  119.3 13001.9
2  1093.5 6186.6 -798.4   208.3
3   892.3 4028.2 -367.9  -463.9
4   590.5 2607.9  -81.6   188.7
5   979.4 6411.0  943.5   454.9
6   945.5 4943.1   19.7 -1170.8
7   255.0  213.3   15.5  4358.8
8  1085.2 5834.5 -384.7   -94.1
9   267.6  340.3 6866.2  5788.6
10 1162.6 6424.4 -497.6  -149.2
```

This is identified as a low quality base because there are two bases at this position.



This peak at position 523 shows both a T and a C.

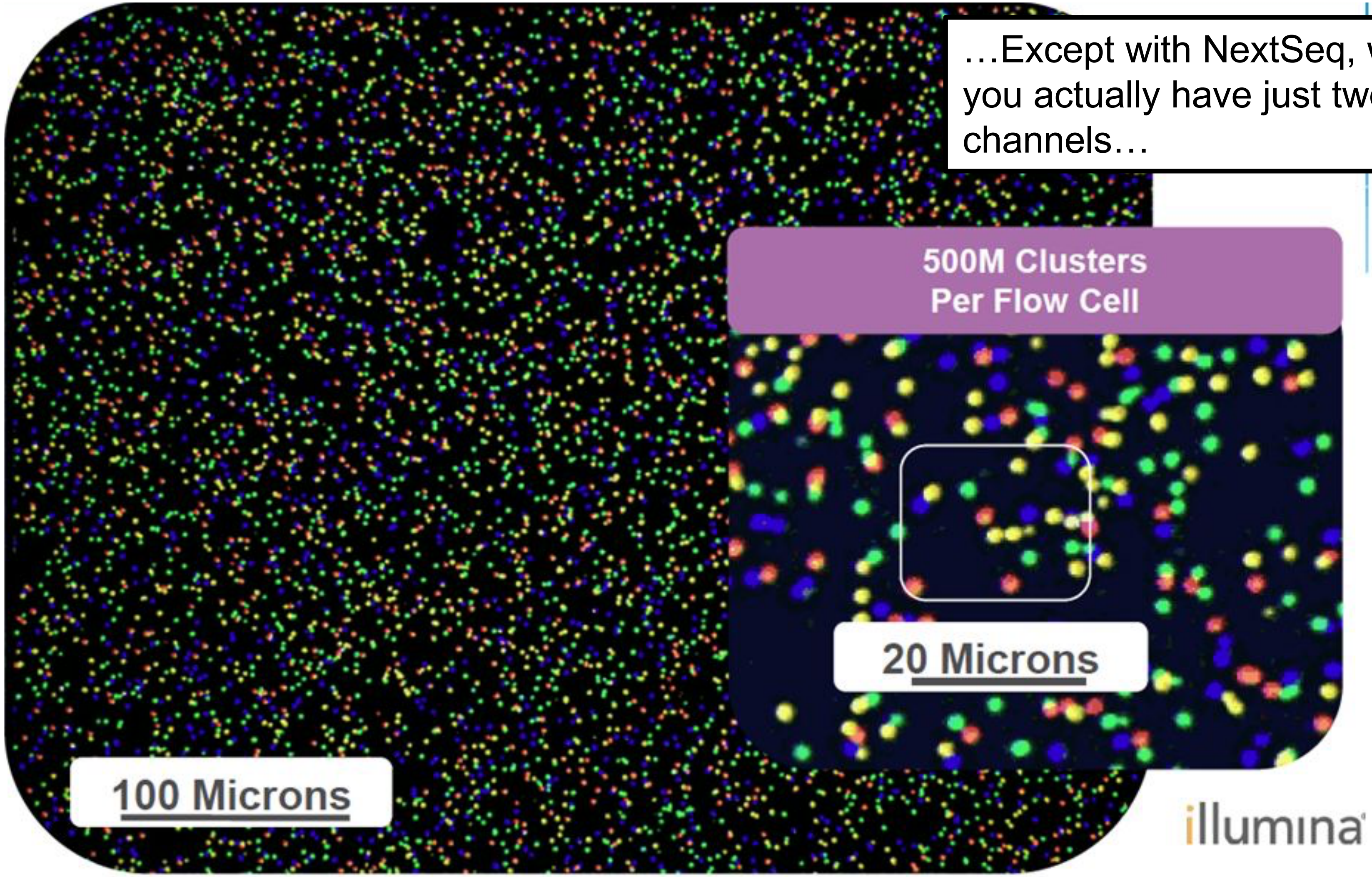
3) FASTQ file with the read sequences & quality values for each base

FASTQ file:

```
…
@read name
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+ read name
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
…
```

Sequencing by synthesis (Illumina)

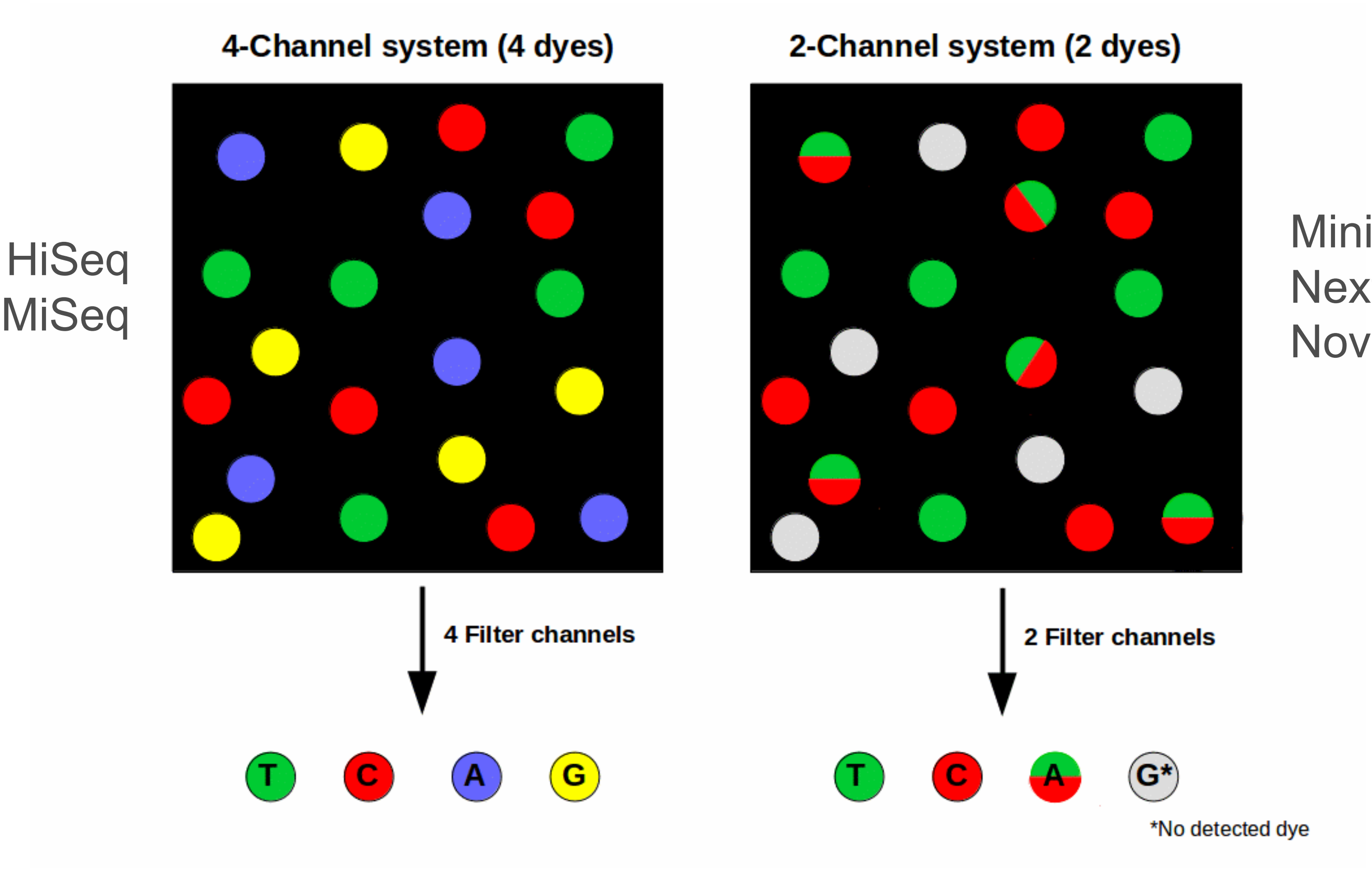
- **Now, how the flowcell and cluster ACTUALLY look like…**



…Except with NextSeq, where you actually have just two channels…

…and the same with two channels

HiSeq
MiSeq



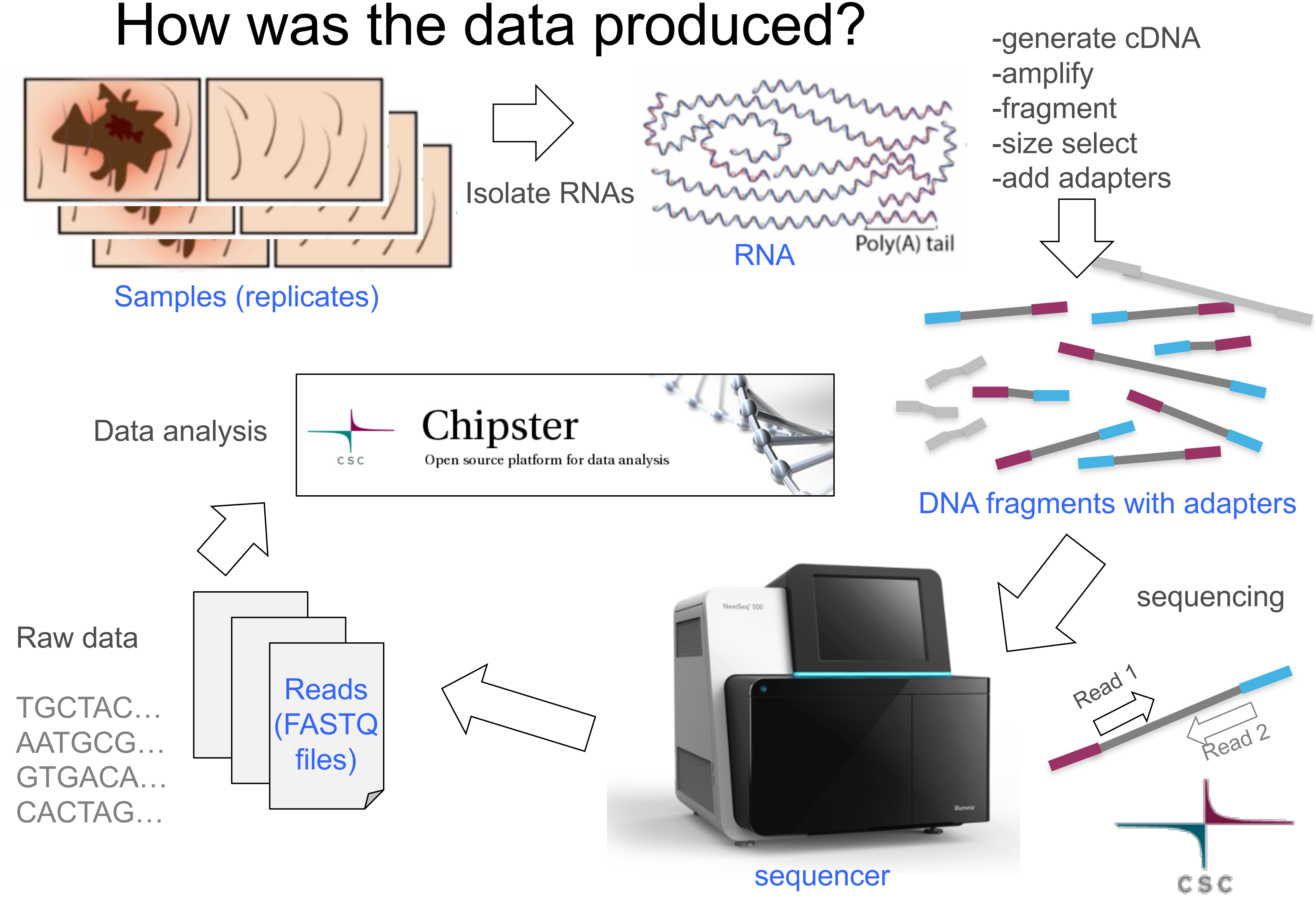
MiniSeq
NextSeq
NovaSeq

- https://www.ecseq.com/support/ngs/do_you_have_two_colors_or_four_colors_in_Illumina

Illumina devices

	What is it good for	**How long it takes**	**Reads per run**	**Max read lengt (bp)**	**Lanes**
MiSeq	Microbes, viruses, targeted panels	4-55 h	25 M	2 x 300	1
NextSeq	Exomes, transcriptomes	12-30 h	400 M	2 x 150	4 (all samples to all 4 lanes)
HiSeq 2500	Whole genomes	Up to 6 days	300 M – 4 G	2 x 250	8 (different samples to different lanes)
NovaSeq (2017)	Whole genomes, scalable	20 – 40 h	1.6 – 20 G	2 x 150	

How was the data produced?



Samples (replicates)

Isolate RNAs

RNA

Poly(A) tail

-generate cDNA
-amplify
-fragment
-size select
-add adapters

DNA fragments with adapters

sequencing

Read 1

Read 2

sequencer

Reads (FASTQ files)

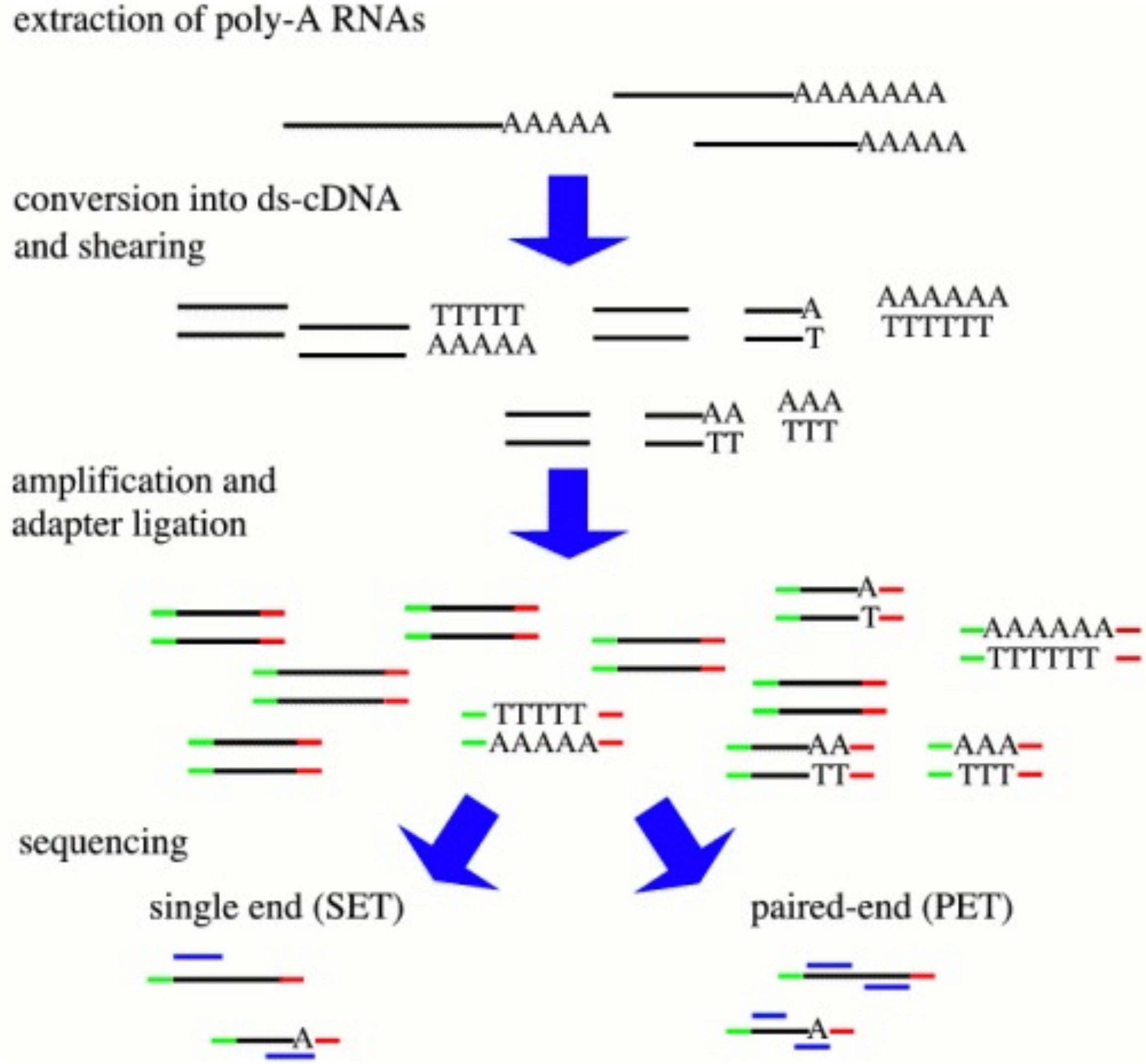
Raw data

TGCTAC…
AATGCG…
GTGACA…
CACTAG…

Data analysis

Chipster
Open source platform for data analysis

How was your data produced?



PolyA purification

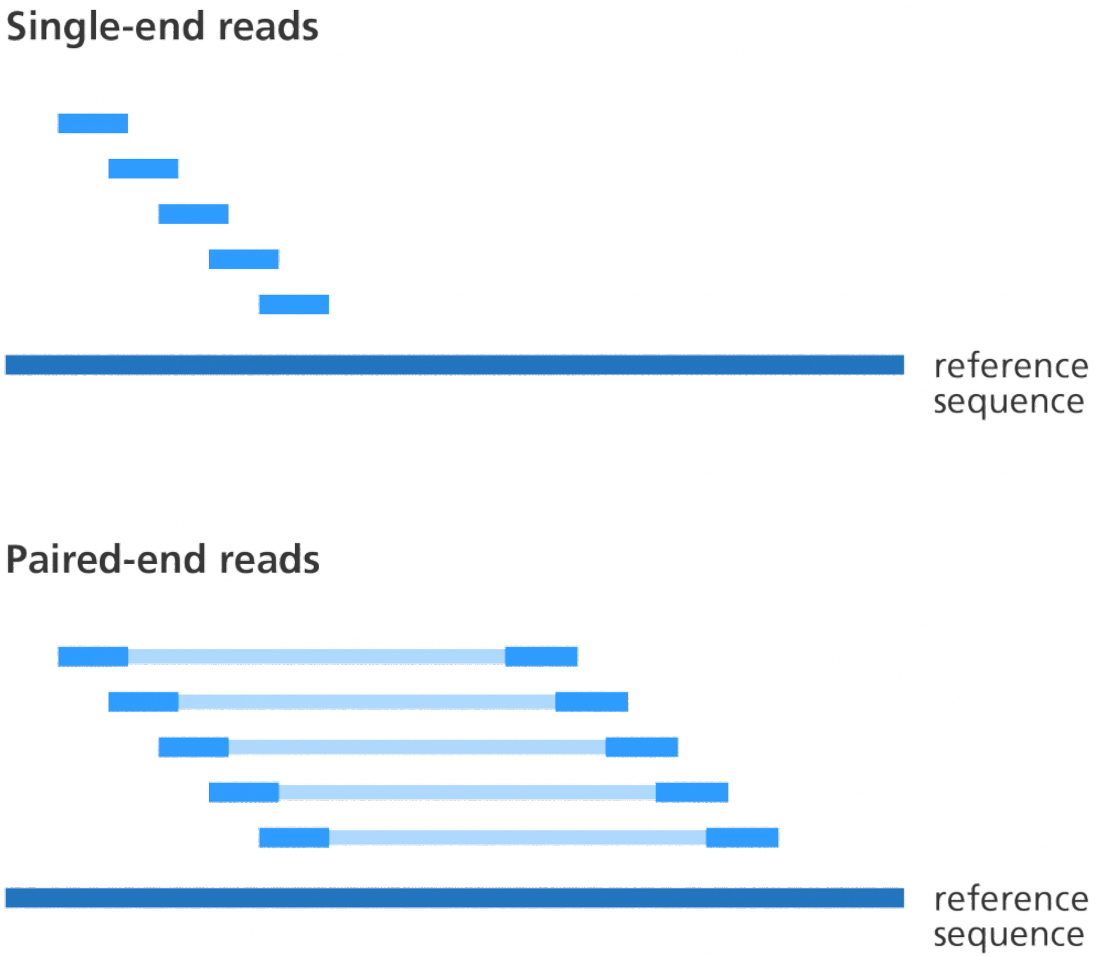
cDNA generation
& fragmentation

Library construction

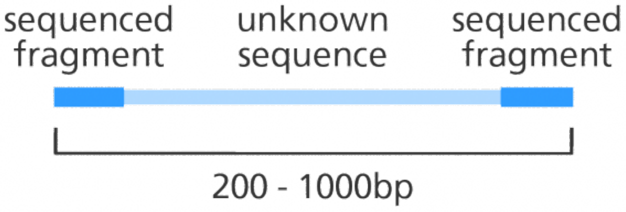
Size selection

http://cmb.molgen.mpg.de/2ndGenerationSequencing/Solas/RNA-seq.html

Paired-end vs single-end reads

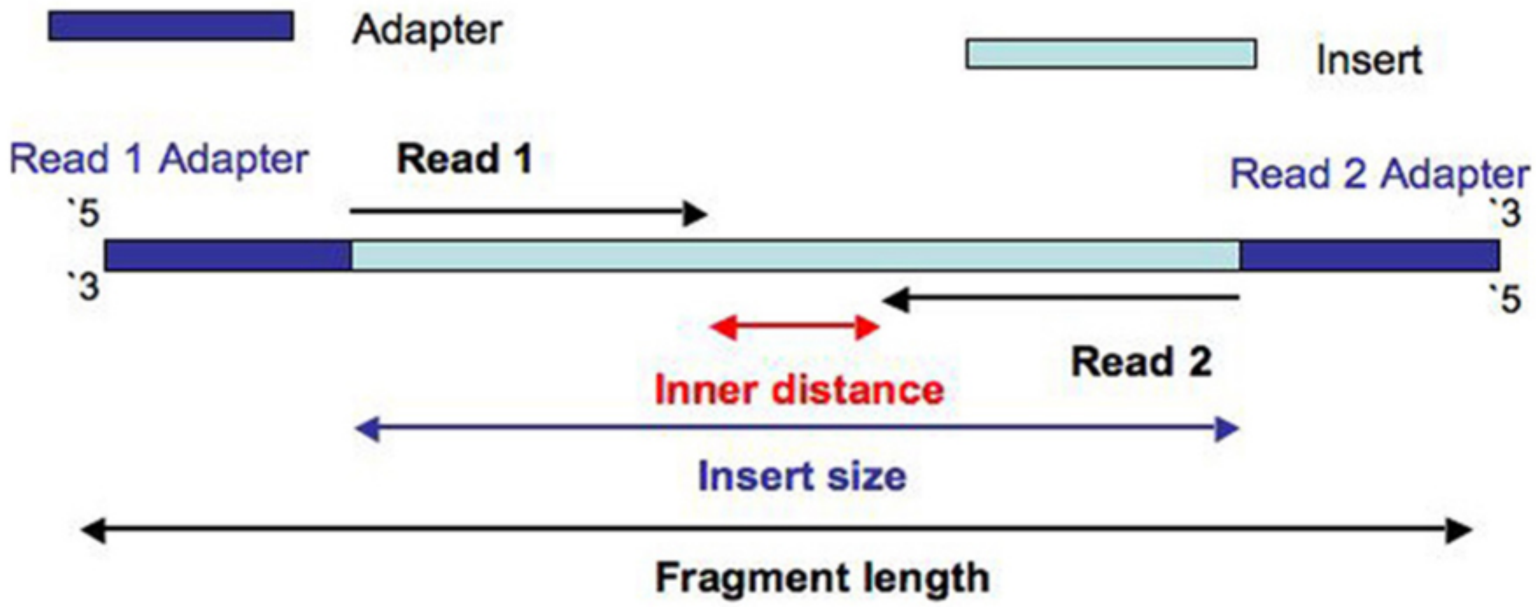


Insert length



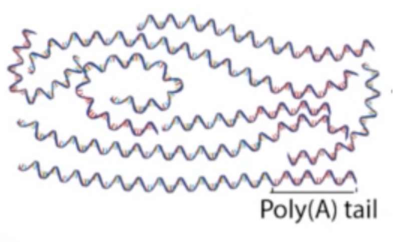
Differently sized fragments & inner distance

- **Illumina reads are always of same length**
- **But the size of the initial mRNA fragment (=insert) may vary**



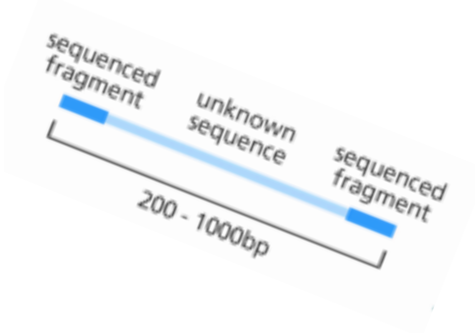
https://www.ecseq.com/support/ngs/why-do-the-reads-all-have-the-same-length-when-sequencing-differently-sized-fragments

Read length = number of sequencing cycles



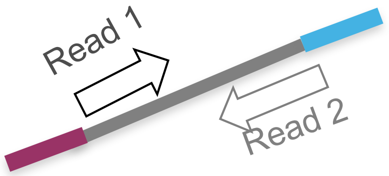
mRNA molecule (**3000** bases)

…ACTCACGTGTACGTAGCTAGTTTACGACTGACTCGCAGTAC
ATGCGCTCGTGGATCACTCGCTACTGCACTACGACTACGACAT
ATCAGCGGCATCGTGATCGGGCATGCATCGTACGCACTGATA
TACGCATAATCAGCTACGATCAGCATTATTCACCTACTATCACTC
CACATCACTTTAACCTGCGGGACTGACGTGACGTCACAAAAA…



mRNA fragment (**500** bases)

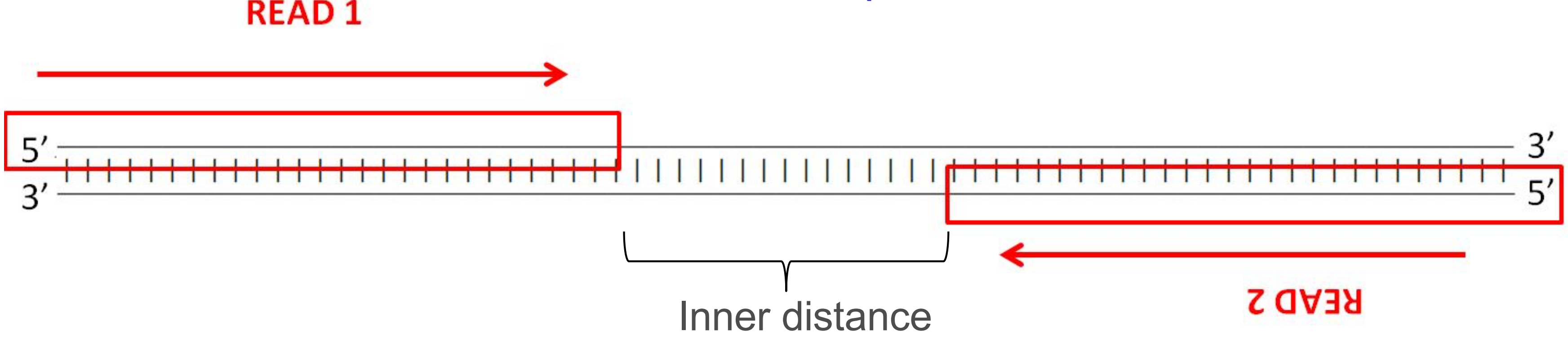
ADAPTER-CGTGGATCACTCGCTACTGCACTACGACTACGACAT
ATCAGCGGCATCGTGATCGGGCATGCATCGTACGCACTGATA
TACGCATAATCAGCTACGATCAGCATTAT-ADAPTER



Reads (**100** bases each)

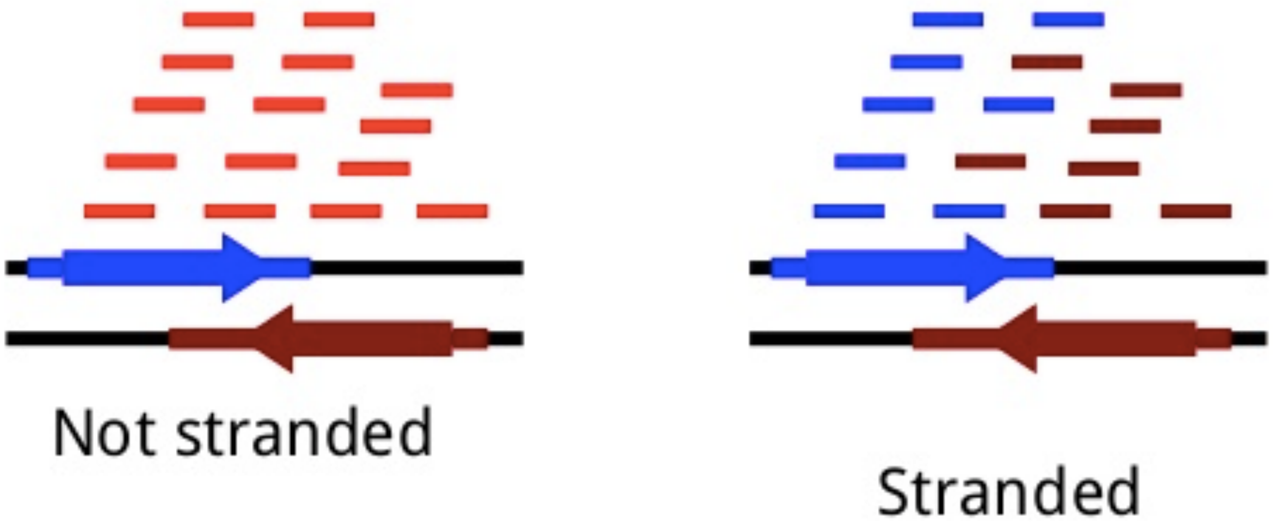
Read 1: CGTGGATCACTCGCTACTGCACTACGACTACGACA
Read 2: CTGATATACGCATAATCAGCTACGATCAGCATTATA

Question: In our example, what is the inner distance?



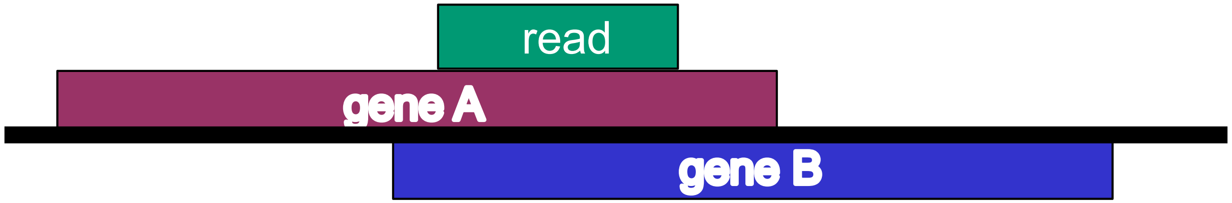
Strandedness

- **Several methods**
- **Stranded/directional method = you have the information of which strand the sequence originally came from**

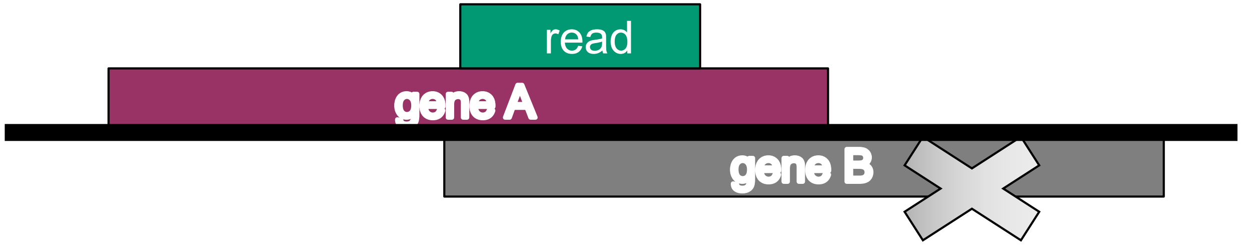


Stranded RNA-seq data

- **Tells if a read maps to the same strand where the parental gene is, or to the opposite strand**
  - Useful information when a read maps to a genomic location where there is a gene on both strands
- **Several lab methods, you need to know which one was used**
  - TruSeq stranded, NEB Ultra Directional, Agilent SureSelect Strand-Specific…



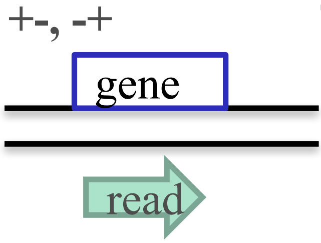
Unstranded data:
Does the read come from geneA or geneB?

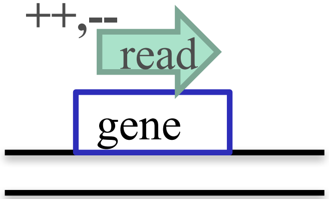


Stranded data
→ the read comes from geneA

Stranded / directional RNA-seq data

➢ **Important to indicate which one was used in some analysis tools**
- parameter naming differs in different tools
- You can check this with a RseQC tool

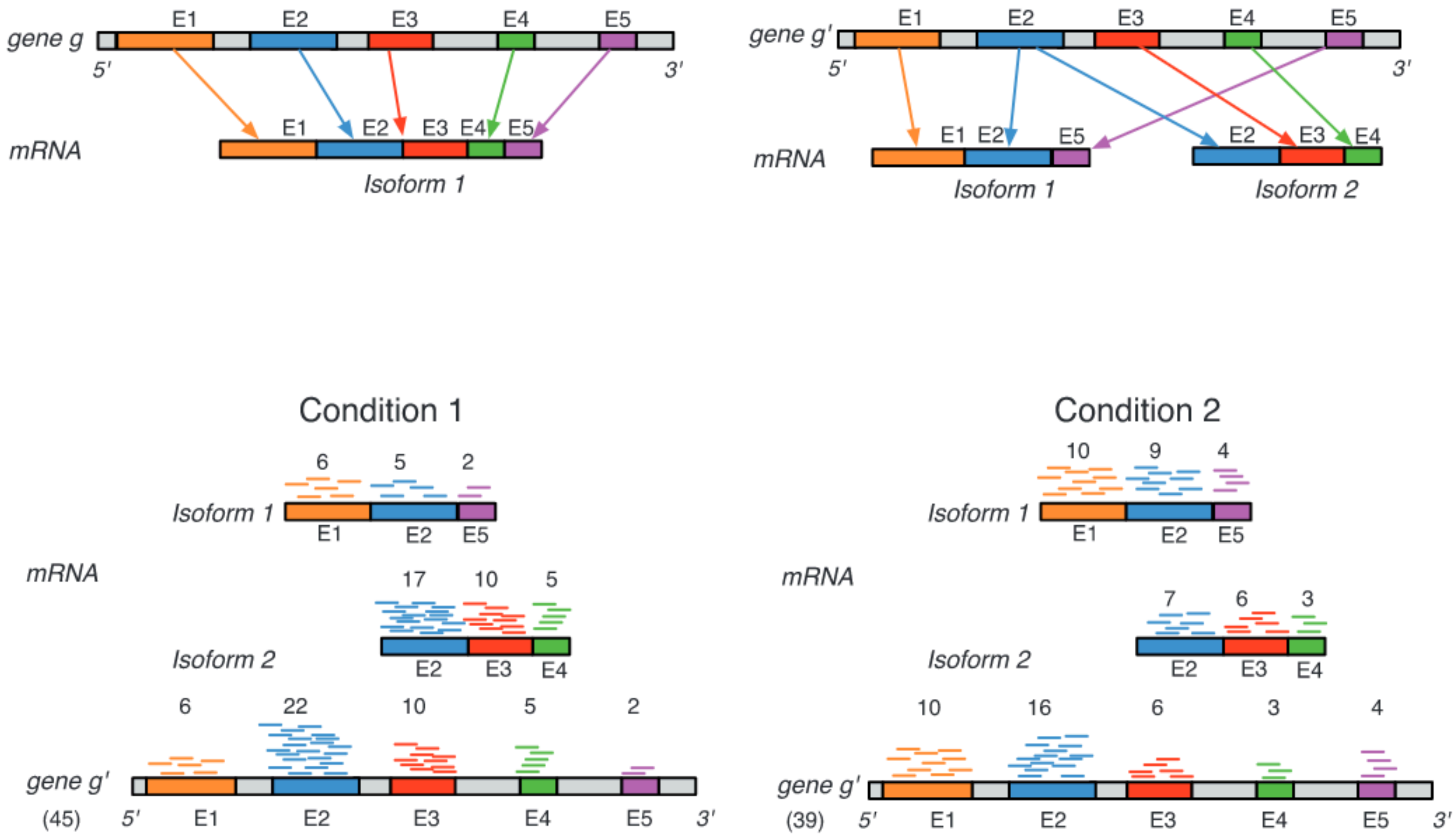




Strandedness:	**TopHat**	**HISAT2**	**HTSeq**
Read (1) and transcript on opposite strands	Fr-firststrand	--rna-strandedness R (SE) / RF (PE)	--stranded reverse
Read (1) and transcript on the same strand	Fr-secondstrand	--rna-strandedness F (SE) / FR (PE)	--stranded yes
No knowledge of where the read comes from	Fr-unstrand	default	--stranded no

Differential gene expression analysis

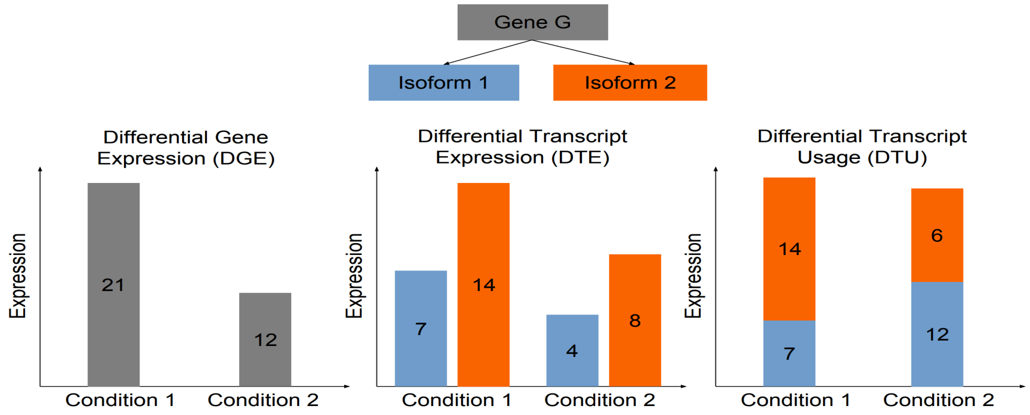
Gene vs. transcript/isoform level analysis



Leng et al. EBSeq: an empirical Bayes hierarchical model for inference in RNA-seq experiments, Bioinformatics, 2013

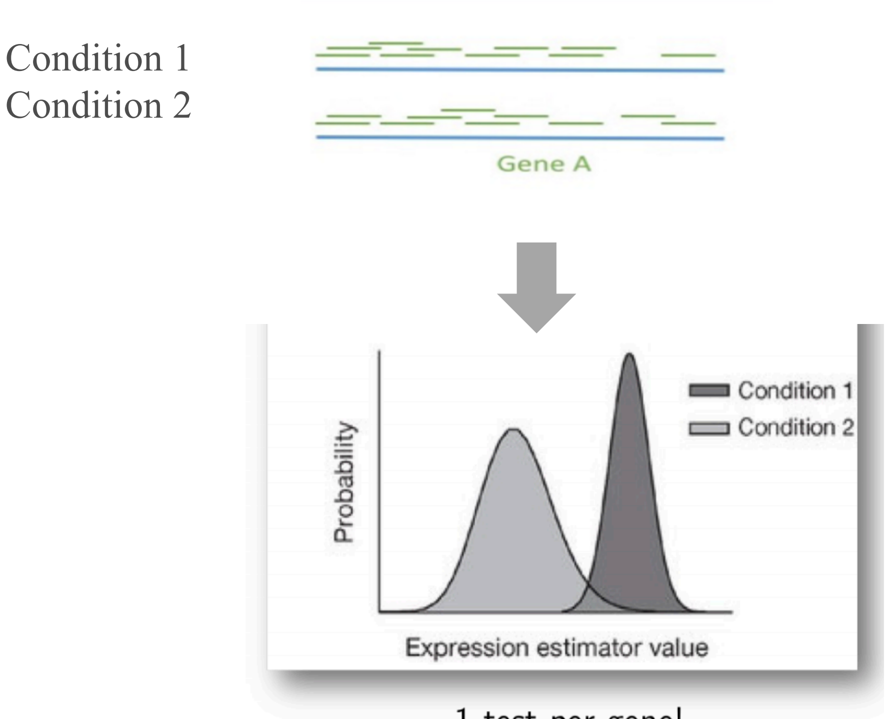
Types of differential expression analysis

- **DGE (differential gene expression): has the expression of a gene changed overall?**
- **DTE (differential transcript expression): has the expression of an individual transcript changed?**
- **DTU (differential transcript usage): has the <u>relative</u> expression of the different transcript isoforms of a gene changed?**



What is differential gene expression, DGE?

➢ Test whether collective abundance of transcripts levels from a gene change between conditions?



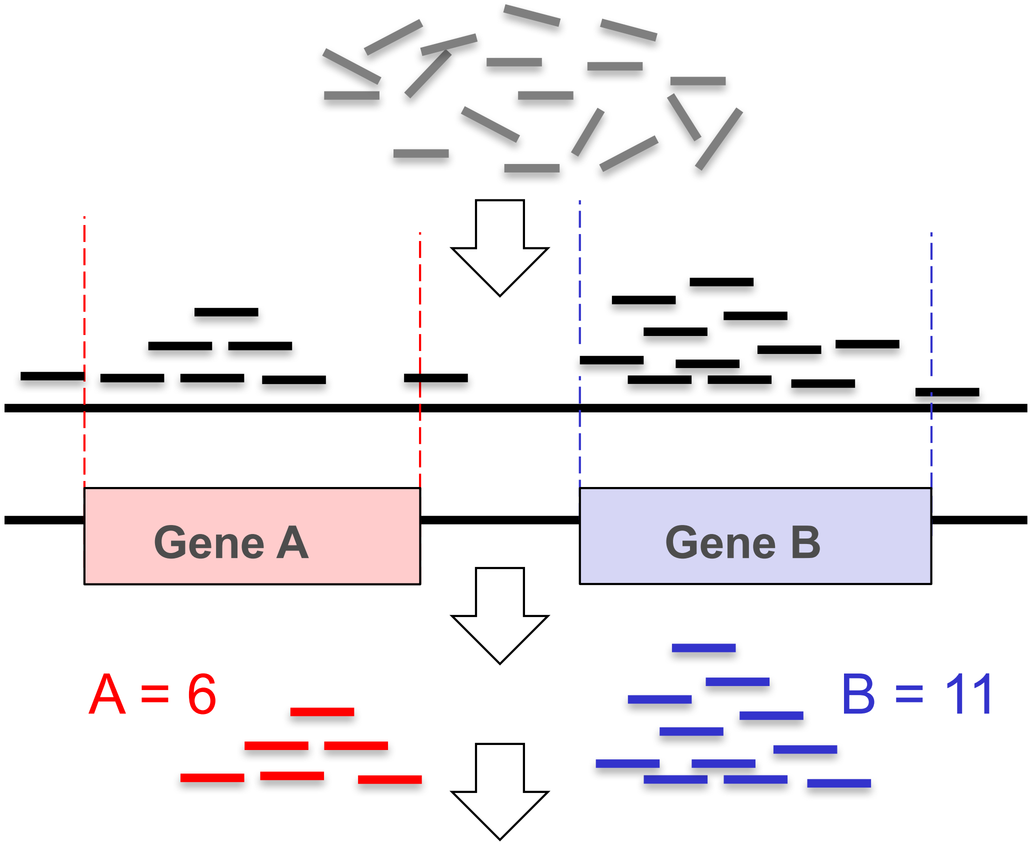
1 test per gene!

- Estimate the magnitude of differential expression between two or more conditions based on read counts from replicated samples

- Estimate the significance of the difference and correct for multiple testing

Null Hypothesis:
There is no difference in the read distributions in two conditions

DGE analysis: typical steps



Raw data (reads)

Align reads to reference genome

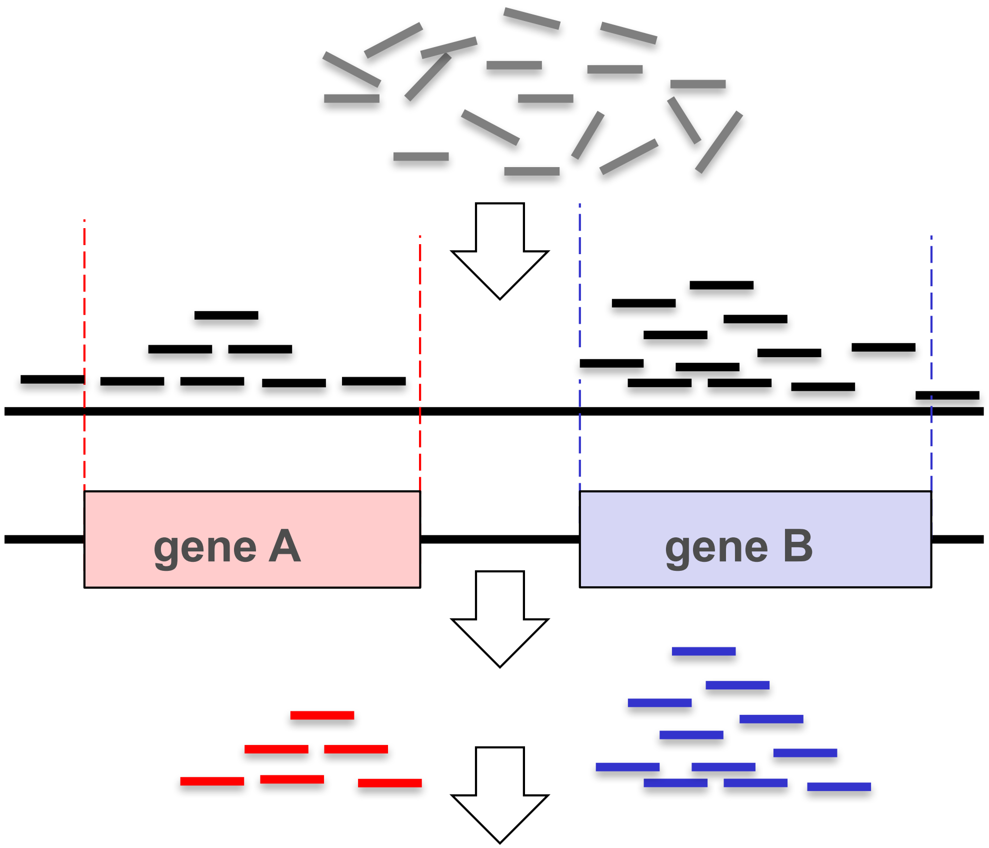
Match alignment positions with known gene positions

Count how many reads each gene has

	Control 1	Control 2	Control 3	Sample 1	Sample 2	Sample 3	
Gene A	6	5	7	170	100	110	⬆
Gene B	11	11	10	3	4	2	⬇
Gene C	200	150	355	50	1	3	⬇
Gene D	0	1	0	2	0	1	🟨

Compare sample groups: differential expression analysis

DGE analysis: steps, tools and files



	Control 1	Control 2	Control 3	Sample 1	Sample 2	Sample 3
Gene A	6	5	7	170	100	110
Gene B	11	11	10	3	4	2
Gene C	200	150	355	50	1	3
Gene D	0	1	0	2	0	1

STEP	TOOL	FILE
Quality control	FastQC	FASTQ
Pre-processing	Trimmomatic	FASTQ
Alignment	HISAT2	BAM
Quality control	RSeQC	
Quantitation	HTSeq	Read count file (TSV)
Combine count files to table		Read count table (TSV)
Quality control	PCA, clustering	
Differential expression analysis	DESeq2, edgeR	Gene lists (TSV)

Practical aspects of working environment in our course

Materials for the course

- **Slides**
  - available on the course webpage
- **Tutorial**
  - With the exercises
  - Available here: https://research.csc.fi/rnaseq-tutorial
- **Virtual machine image**
  - Ready on the classroom computers, downloadable from the tutorial page
- **Course data**
  - Downloadable from the tutorial page
  - Some data generated for you (like: indexes for alignment)
    1. Data for the VM practises
    2. Data for the Puhti practises
- **“Bonus” material: video lectures in Youtube**
  - Link in the tutorial page

Workflow for the course

- **Practising:**
  - Learning the analysis step by step
    - Little bit of theory (what & why)
    - Exercises in **command line** and in **R** (how)
    - Now, we are using virtual machine (mimics CSCs Puhti supercomputer)
  - Two datasets:
    - 2 ”toy samples” for the command line part -> only small part of the reads (this is to save time)
    - 10 ”real” samples for the R expression analysis part

- **How to really do the analysis effectively: running a batch job**
  - In CSC’s Puhti supercomputer

Working environment for course

- **RNAseq analysis: Interactive analysis**
  - Use virtual environment in the Oracle VirtualBox (= linux-like command line environment)
  - Make use of all course installations for running RNAseq analysis
  - Mimics CSCs Puhti supercomputer
  - Downloadable for your own use also after the course
  - Conda modules

- **RNAseq analysis: Batch analysis**
  - Puhti Supercomputer for running analysis with multiple samples (as an array job)

- **RNAseq analysis: Data navigation**
  - Allas environment at CSC for data navigation

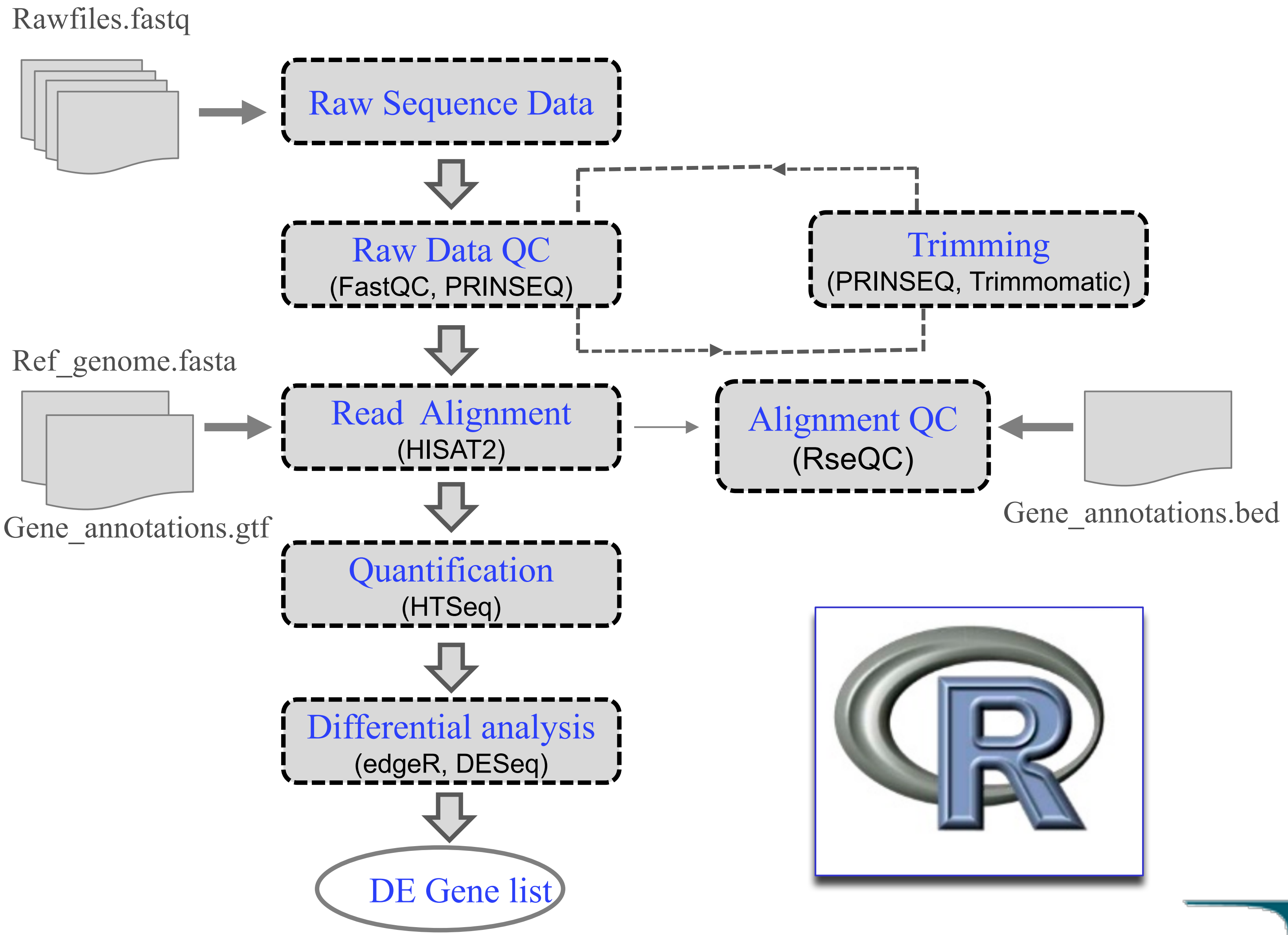
Logging in & getting started with VirtualBox

- **Log in to the classroom computer**
  - Password in the back of the classroom (turn your head)

- **Use virtual machine image from VirtualBox**
  1. Go to “Applications” -> “System Tools” -> Oracle VM VirtualBox
  2. Open image “RNAseq_v1” and click “Start”
  3. Log in to virtual machine: press enter, *password: rnaseq*
  4. Enter password (press: enter tab) : rnaseq
  5. Expect some glitches
  6. Tune the window so that it fits nicely on your screen
     1. View -> Virtual Screen 1 -> Scale Factor = 100%
     2. View -> Auto-Resize Guest Display
  7. Open Terminal

- **Note: copy/paste in terminal:**
  - Ctrl + shift + c = copy & ctrl + shift + v = paste
  - Or with mouse: paint the text to copy and double click to paste

What is inside VM: different software tools + R



Follow the tutorial page instructions in: https://research.csc.fi/rnaseq-tutorial

1. **(Virtual Machine image is already downloaded on the classroom computers)**
2. **Download the RNAseq bundle from *Allas object storage***
3. **"Untar" the raw data bundle**
4. **Rename the folder as *rnaseq***
5. **Check the kind of data/files in the folder**

Testing python and R environment in this VM

- Software tools are installed as *conda packages* and named as 'rnaseq' environment
  - rnaseq environment = all necessary programs are installed for doing RNAseq analysis
  - On the terminal, type: *conda activate rnaseq*

- To open Rstudio:
  - R packages needed in the course also readyly installed (**no need to run installation commands!**)
  - *conda activate base*
  - *Rstudio*
  - …under Applications -> Programming -> rstudio

Data analysis workflow

- **Quality control of raw reads**
- **Preprocessing if needed**
- **Alignment to reference genome**
- **Alignment level quality control**
- **Quantitation**
- **Experiment level quality control**
- **Differential expression analysis**

Our data is “toy data”

- Small subset of RNA-seq reads from chr19
- Illumina single-end reads
- from two human cell lines: h1-hESC and GM12878 (we practise with hESC sample).
- **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 to save resources:
  - running the exercises with full sample would take hours to complete
  - file sizes would require a lot of memory, making it difficult to run the analysis on a VM

Data analysis workflow

- **Quality control of raw reads**
- **Preprocessing if needed**
- **Alignment to reference genome**
- **Alignment level quality control**
- **Quantitation**
- **Experiment level quality control**
- **Differential expression analysis**

What and why?

- **Potential problems**
  - low confidence bases, Ns
  - sequence specific bias, GC bias
  - adapters
  - sequence contamination
  - …

Knowing about potential problems in your data allows you to

- **correct for them before you spend a lot of time on analysis**
- **take them into account when interpreting results**

Software packages for quality control

- **FastQC**
- **PRINSEQ**
- **MultiQC**
- FastX
- TagCleaner
- ...

Raw reads: FASTQ file format

- **Four lines per read:**

```
@read name
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+ read name
!"*((((***+))%%%++)(%%%%).1***-+*"))**55CCF>>>>>>CCCCCCC65
```

- **http://en.wikipedia.org/wiki/FASTQ_format**
- **Note: FASTQ files usually zipped (fastq.gz)**
  - Most analysis tools can cope with zipped files (.gz)
  - For some, you need to unzip files:

```
gunzip < hesc.fastq.gz > hesc.fastq
```

Base qualities

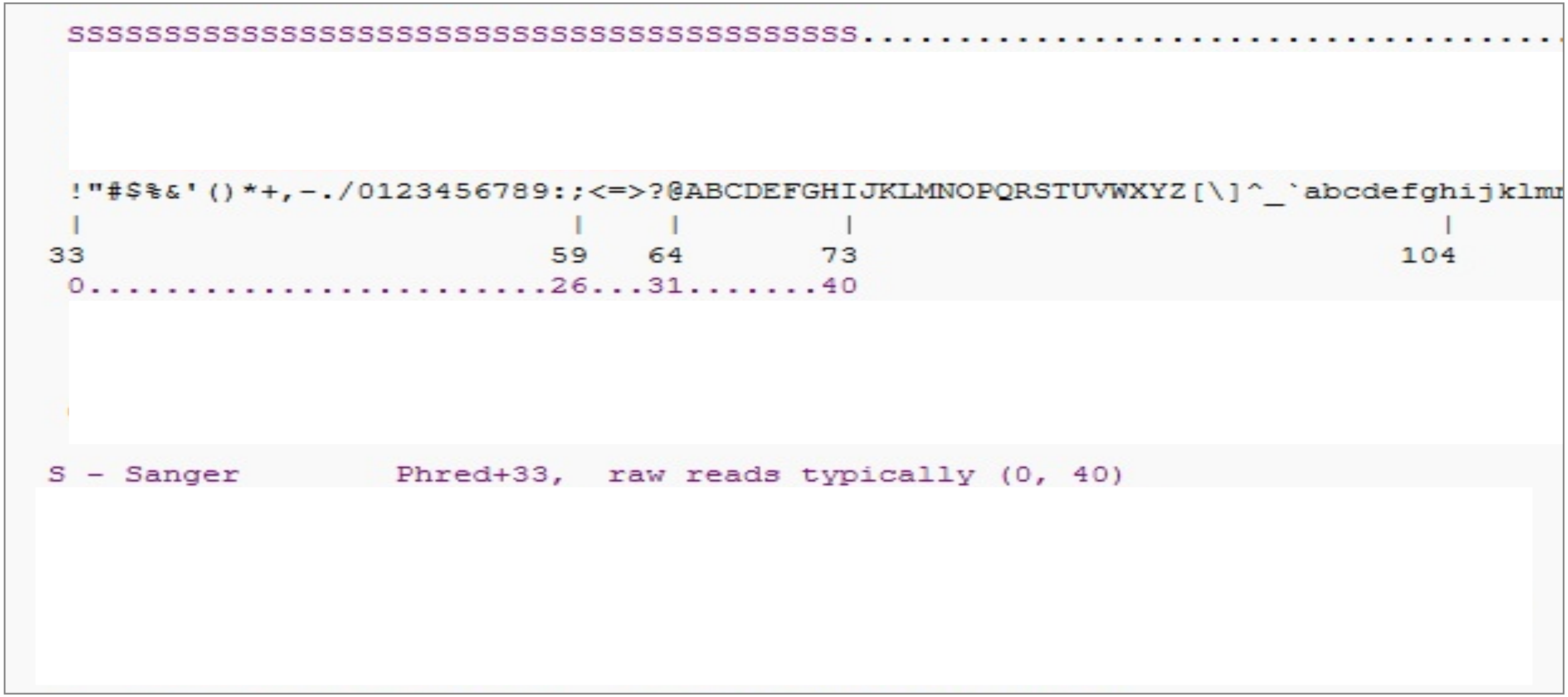
- **If the quality of a base is 20, the probability that it is wrong is 0.01.**
  - **Phred quality score** Q = -10 * $\log_{10}$ (probability that the base is wrong)

```
T  C  A  G  T  A  C  T  C  G
40 40 40 40 40 40 40 40 37 35
```

- **”Sanger” encoding: numbers are shown as ASCII characters**
  - Note that older Illumina data uses different encoding

Phred Quality Score	Probability of Incorrect Base Call	Base Call Accuracy	ASCII coding in FASTQ file
10	1 in 10	90%	+
20	1 in 100	99%	5
30	1 in 1,000	99.9%	?
40	1 in 10,000	99.99%	I

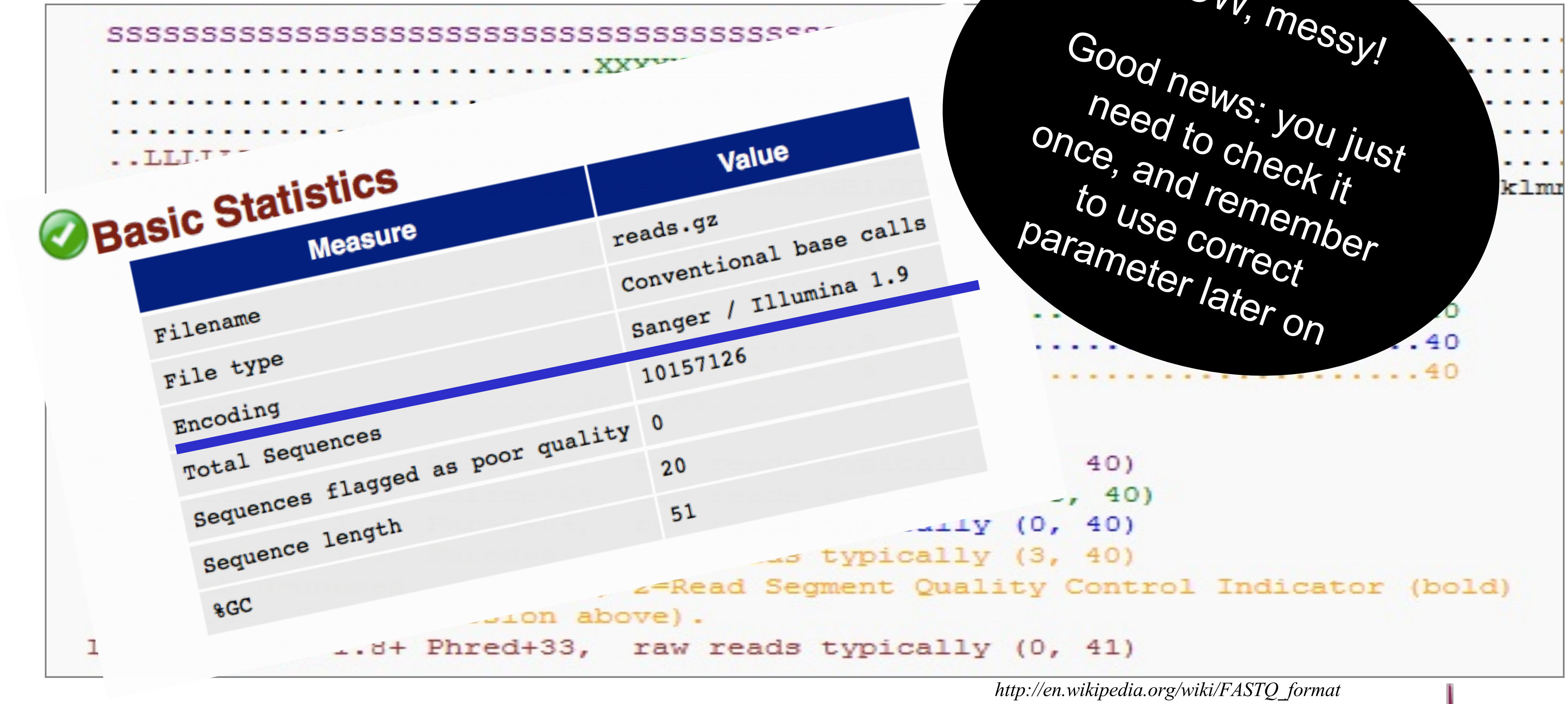
Base quality encoding systems



```
 SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS....................................
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklm
 |                         |    |        |                              |
33                        59   64       73                            104
 0.........................26...31.......40

S - Sanger        Phred+33,  raw reads typically (0, 40)
```

http://en.wikipedia.org/wiki/FASTQ_format

Base quality encoding systems



```
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS
..............................XXX
..LLL
klmn
.40
.40
.40
40)
40)
(0, 40)
typically (3, 40)
=Read Segment Quality Control Indicator (bold)
sion above).
1    .8+ Phred+33,  raw reads typically (0, 41)
```

✓ Basic Statistics

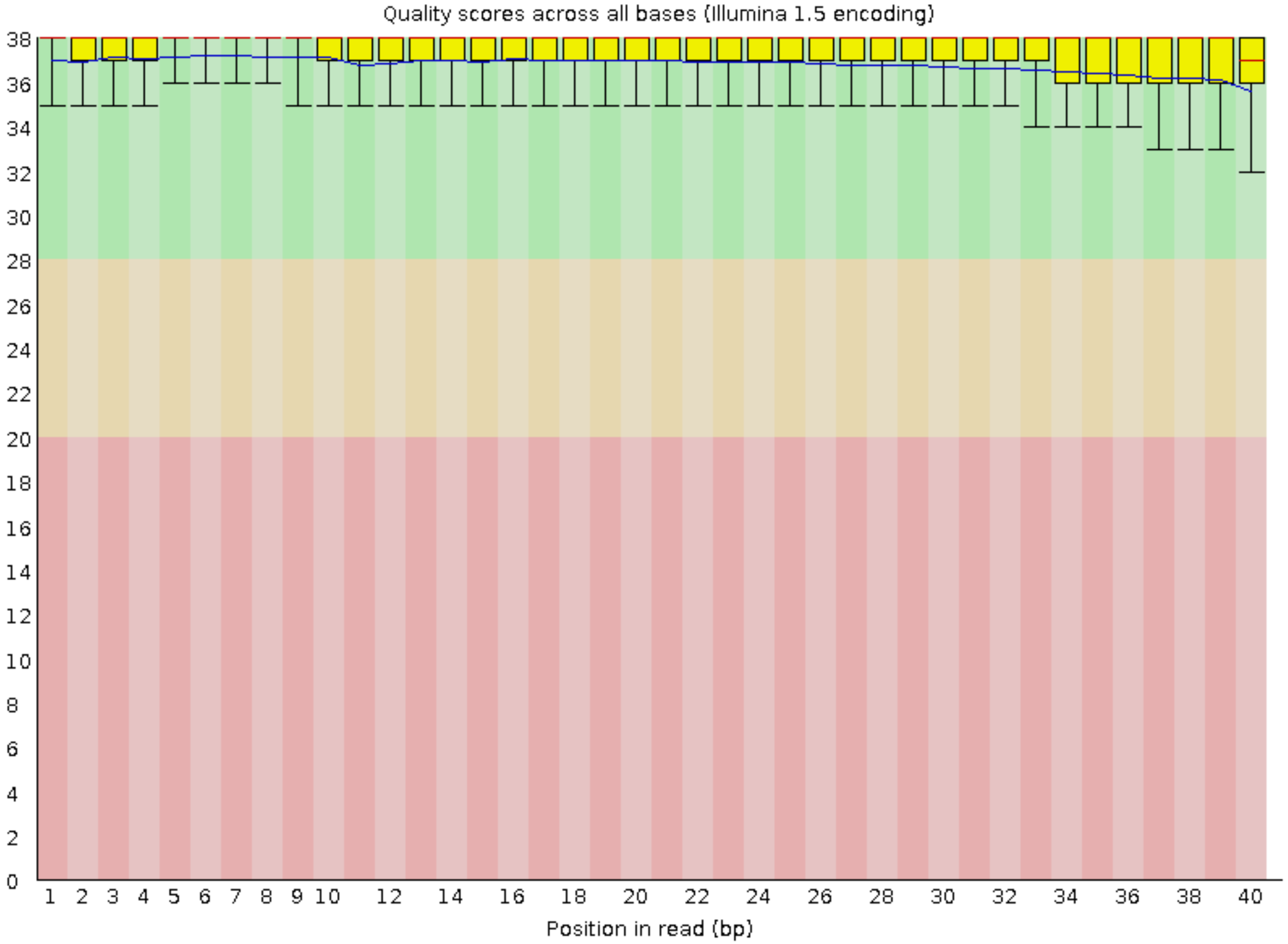
Measure	Value
Filename	reads.gz
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	10157126
Sequences flagged as poor quality	0
Sequence length	20
%GC	51

http://en.wikipedia.org/wiki/FASTQ_format

WOW, messy!

Good news: you just need to check it once, and remember to use correct parameter later on

Per position base quality (FastQC)

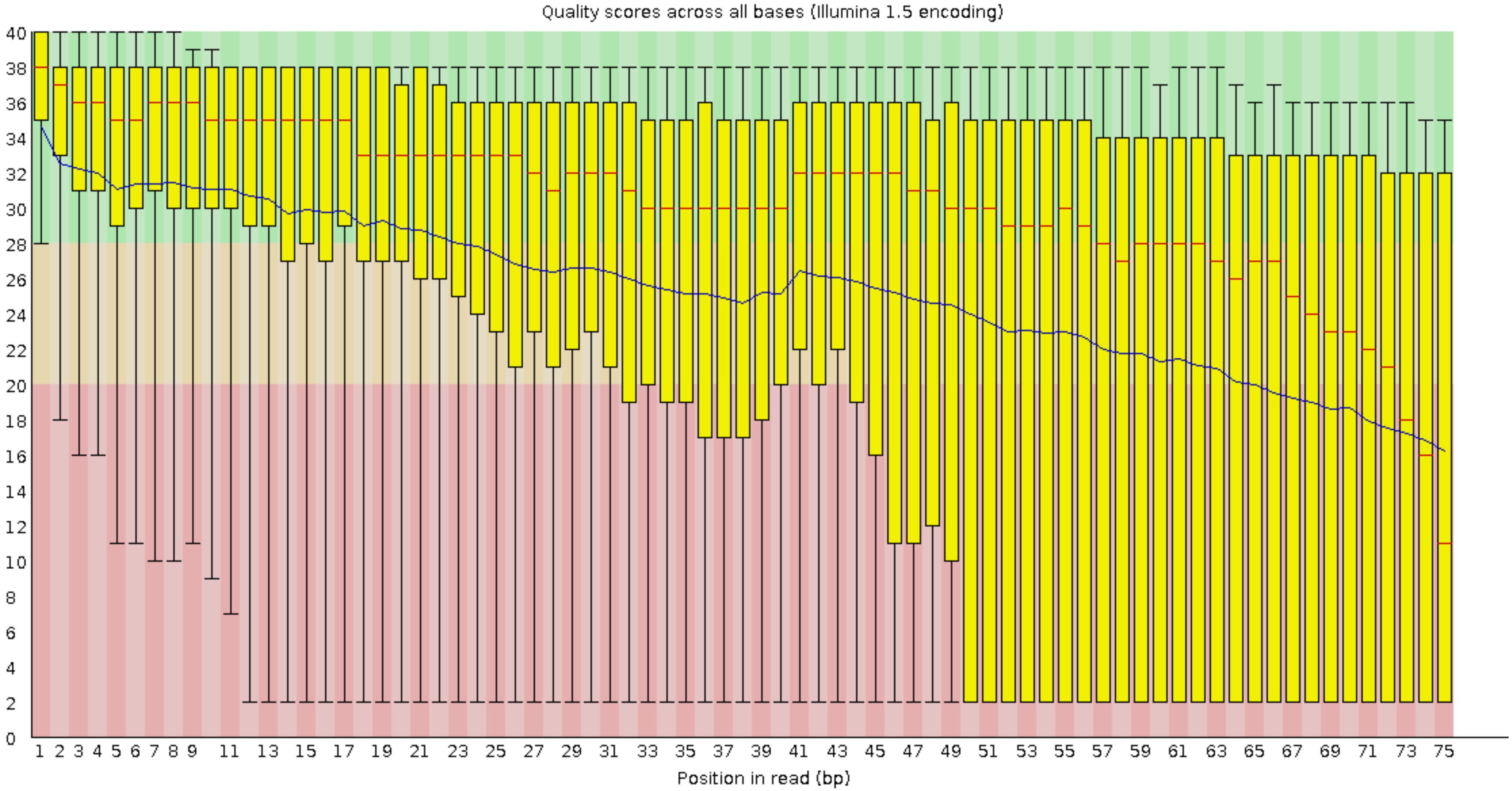


good

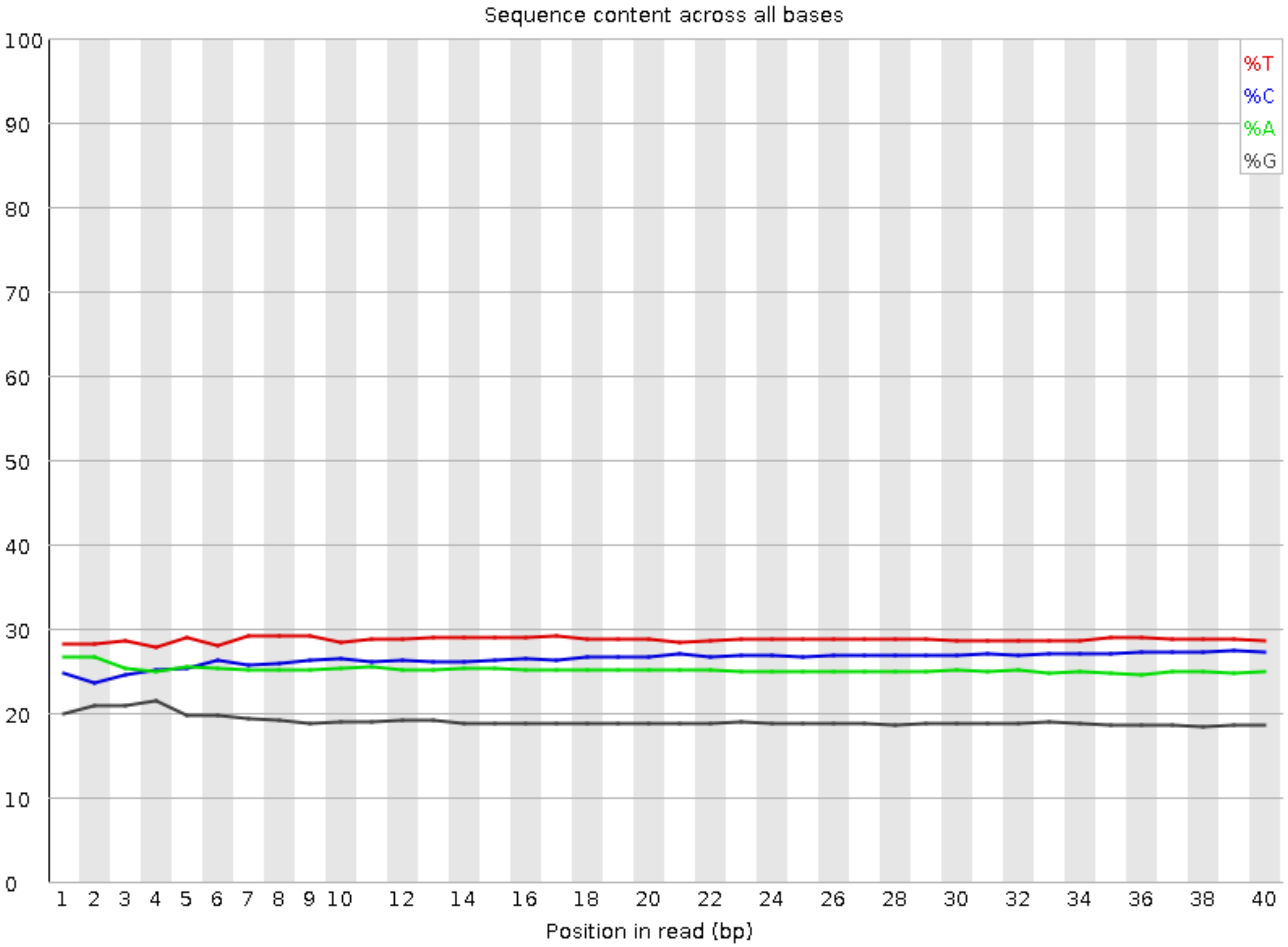
ok

bad

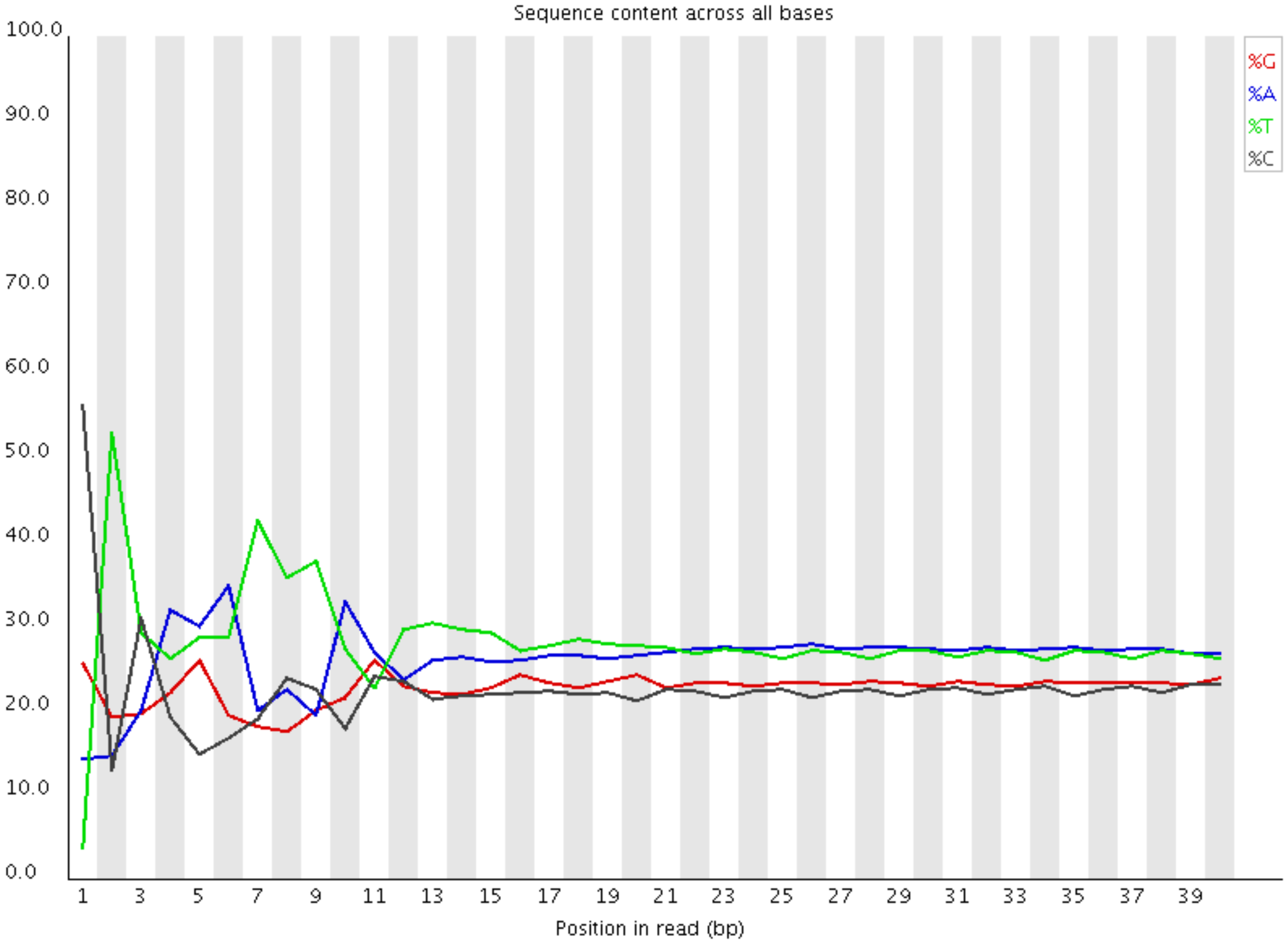
Per position base quality (FastQC)



Per position sequence content (FastQC)



Per position sequence content (FastQC)

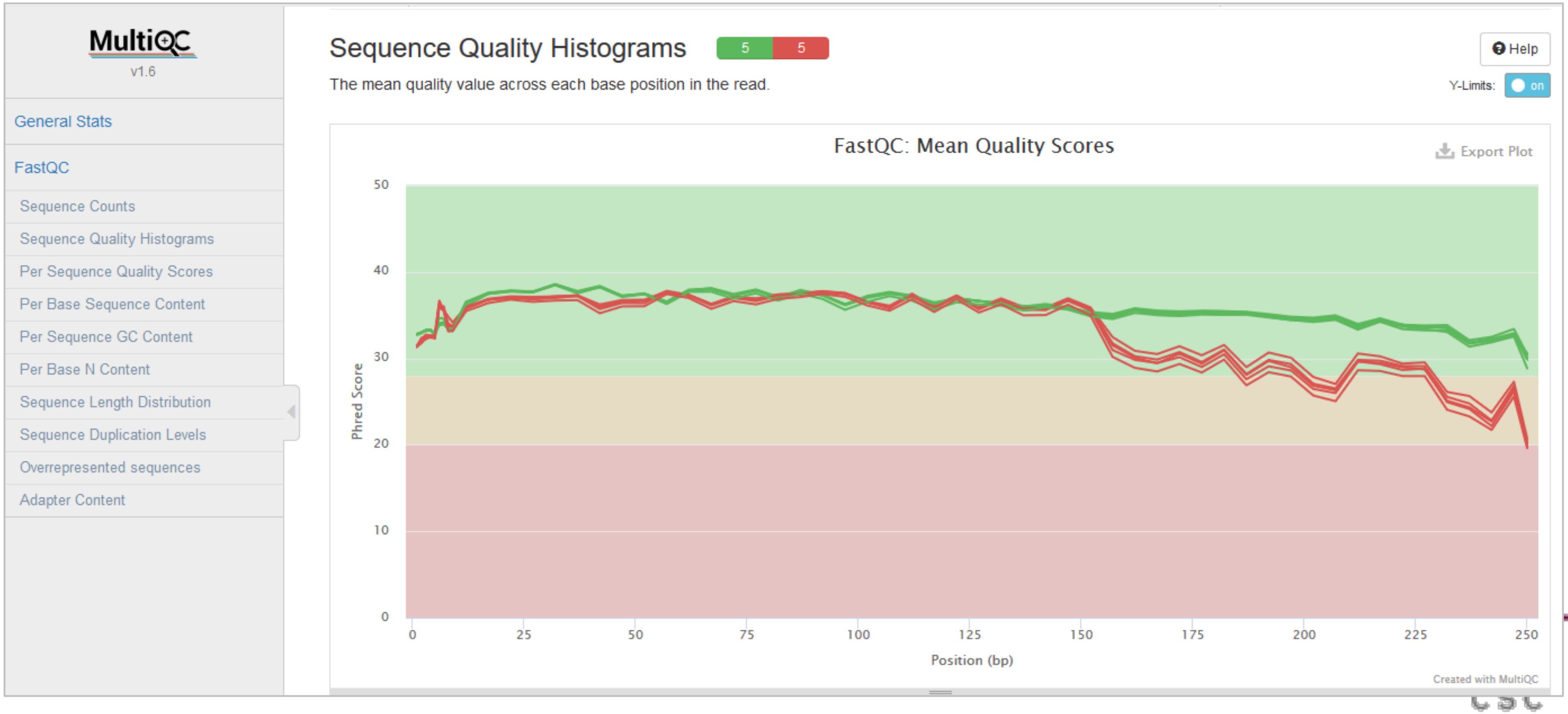


- **Enrichment of k-mers at the 5’ end due to use of random hexamers or transposases in the library preparation**
- **Typical for RNA-seq data**
- **Can’t be corrected, doesn’t usually effect the analysis**

I have many FASTQ files – how can I quickly check them all?

- MultiQC
- Just run in your working directory –this will collect all the relevant files

```
multiqc .
```



Sequence Quality Histograms

The mean quality value across each base position in the read.

Data analysis workflow

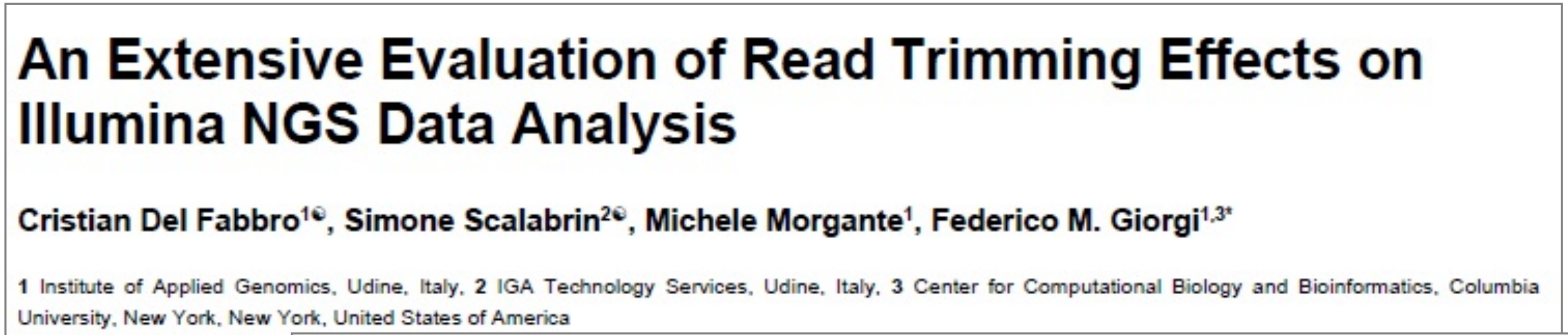
- Quality control of raw reads
- **Preprocessing (trimming / filtering) if needed**
- Alignment to reference genome
- Alignment level quality control
- Quantitation
- Experiment level quality control
- Differential expression analysis

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

What base quality threshold should be used?

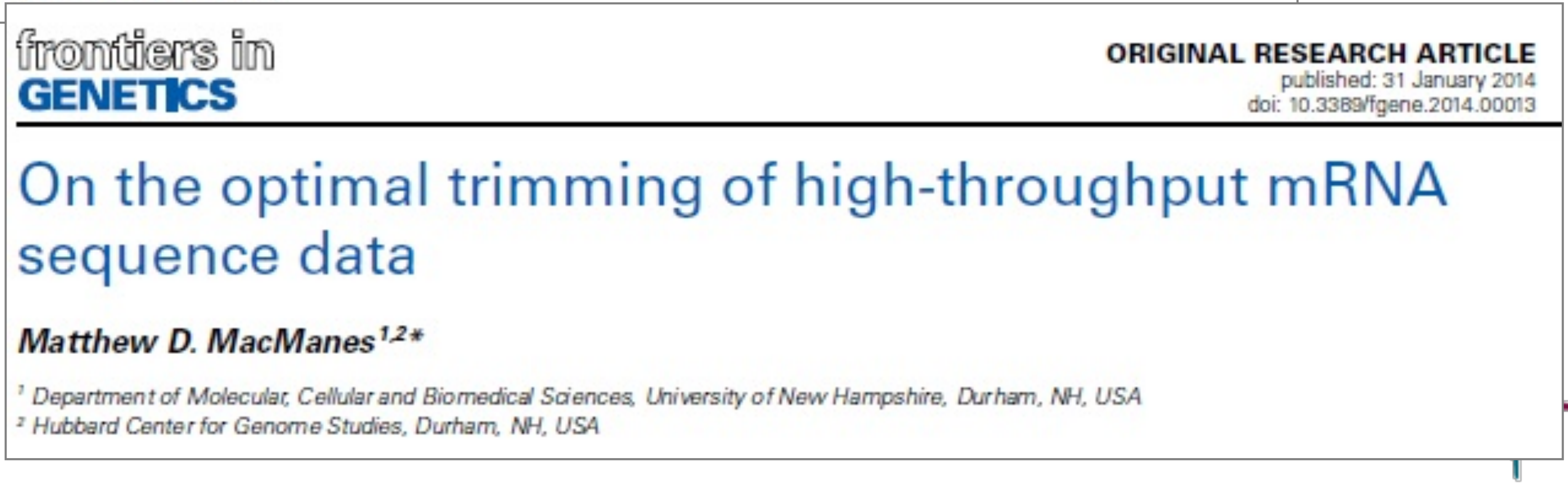
- ➢ **No consensus**
- ➢ **Trade-off between having good quality reads and having enough sequence**
- ➢ **Start with gentle trimming and check with FastQC**



An Extensive Evaluation of Read Trimming Effects on Illumina NGS Data Analysis

Cristian Del Fabbro[1☯], Simone Scalabrin[2☯], Michele Morgante[1], Federico M. Giorgi[1,3*]

[1] Institute of Applied Genomics, Udine, Italy, [2] IGA Technology Services, Udine, Italy, [3] Center for Computational Biology and Bioinformatics, Columbia University, New York, New York, United States of America



frontiers in GENETICS

ORIGINAL RESEARCH ARTICLE
published: 31 January 2014
doi: 10.3389/fgene.2014.00013

On the optimal trimming of high-throughput mRNA sequence data

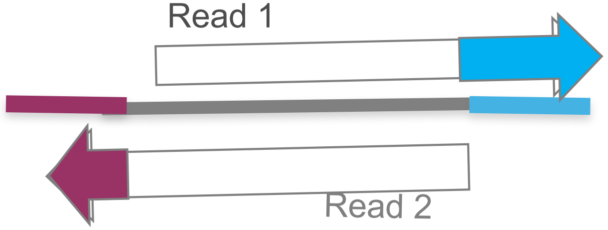
Matthew D. MacManes[1,2]*

[1] *Department of Molecular, Cellular and Biomedical Sciences, University of New Hampshire, Durham, NH, USA*
[2] *Hubbard Center for Genome Studies, Durham, NH, USA*

Software packages for preprocessing

- **Trimmomatic**
- FastX
- PRINSEQ
- TagCleaner
- ...

Trimmomatic options



- **Adapters**
  - Cause: reading through the (too short) fragment
  - You need: file listing the adapter sequences
- **Minimum quality**
  - Per base, one base at a time or in a sliding window, from 3’ or 5’ end
  - Per base adaptive quality trimming (balance length and errors)
  - Minimum (mean) base quality
- **Trim x bases from left/ right**
- **Minimum read length after trimming**
- **Copes with paired end data**

Terminology:

- LEADING edge = 5’ end = left side = the beginning of the read
- TRAILING edge = 3’ end = right side = the end of the read

Data analysis workflow

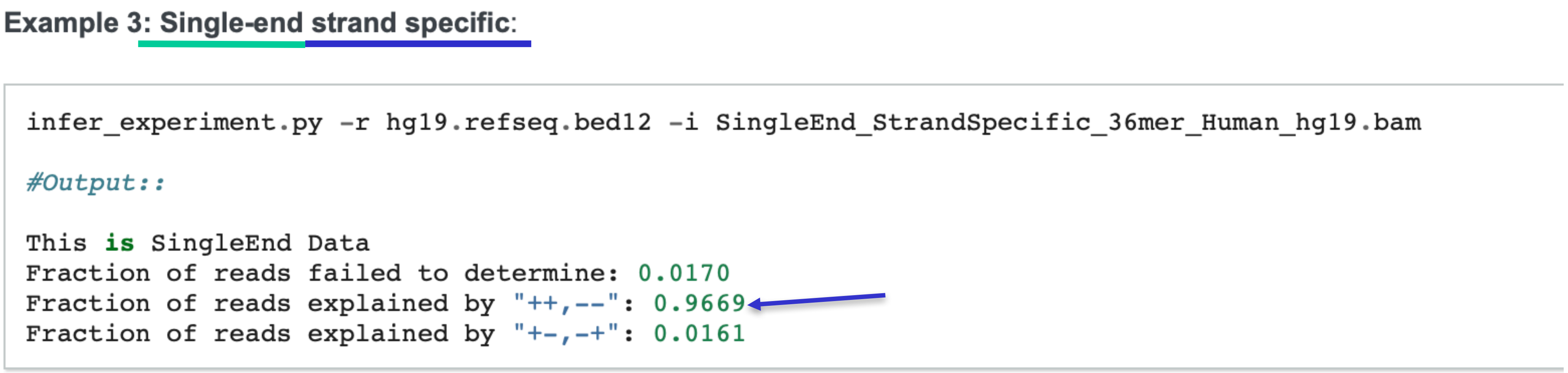
- Quality control of raw reads
- Preprocessing (trimming / filtering) if needed
- **Alignment to reference genome**
- Alignment level quality control
- Quantitation
- Experiment level quality control
- Differential expression analysis

Alignment:
Check the strandedness of your data

Was your data made with stranded protocol?

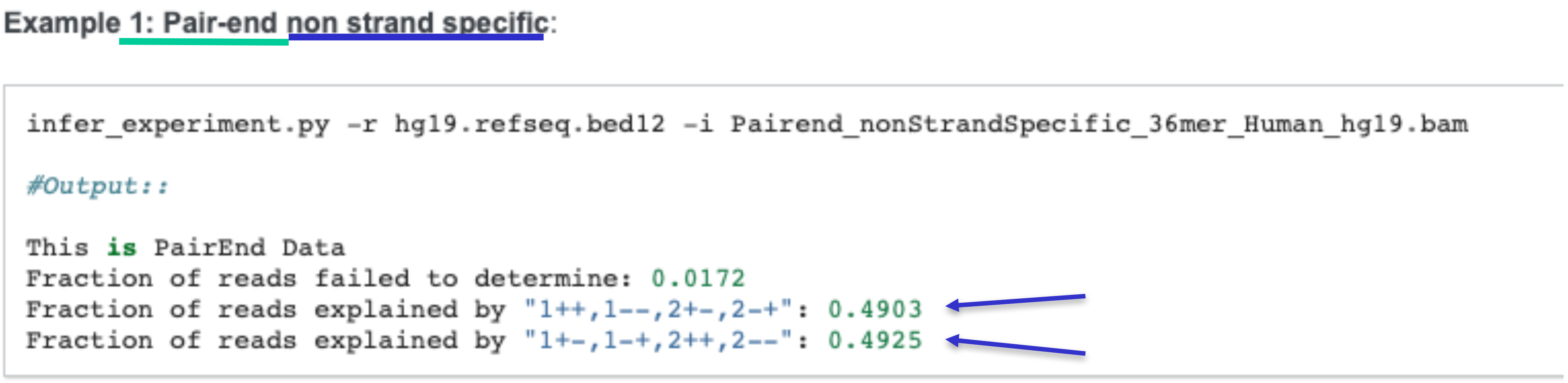
- **You need to indicate it when:**
  - aligning reads to genome (e.g. HISAT2)
  - counting reads per genes (e.g. HTSeq)
- **If you don't know if/which stranded sequencing protocol was used, you can check it:**
  - with RseQC tool infer_experiment.py:
    - First align a subset of the reads to genome, and then with infer_experiment.py compare the locations to reference annotation
    - http://rseqc.sourceforge.net/#infer-experiment-py
- **some help/summary collected here: https://chipster.csc.fi/manual/library-type-summary.html**

RseQC strandedness report



Example 3: Single-end strand specific:

```
infer_experiment.py -r hg19.refseq.bed12 -i SingleEnd_StrandSpecific_36mer_Human_hg19.bam

#Output::

This is SingleEnd Data
Fraction of reads failed to determine: 0.0170
Fraction of reads explained by "++,--": 0.9669
Fraction of reads explained by "+-,-+": 0.0161
```



Example 1: Pair-end non strand specific:

```
infer_experiment.py -r hg19.refseq.bed12 -i Pairend_nonStrandSpecific_36mer_Human_hg19.bam

#Output::

This is PairEnd Data
Fraction of reads failed to determine: 0.0172
Fraction of reads explained by "1++,1--,2+-,2-+": 0.4903
Fraction of reads explained by "1+-,1-+,2++,2--": 0.4925
```

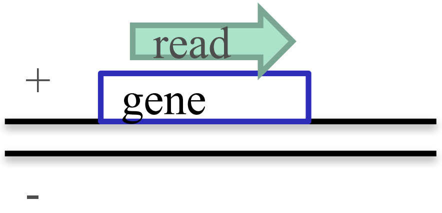
What does this ++, - - mean?

Single end:

++,--

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

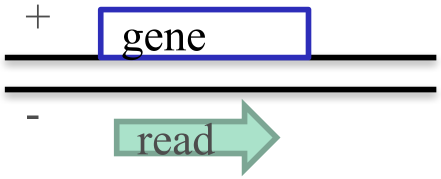
read mapped to '-' strand indicates parental gene on '-' strand



+-,-+

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

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



Paired end:

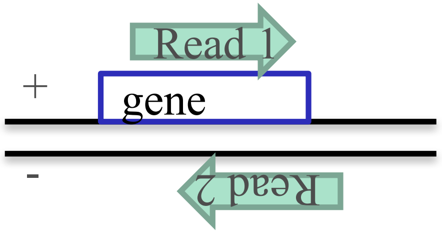
1++,1–,2+-,2-+

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



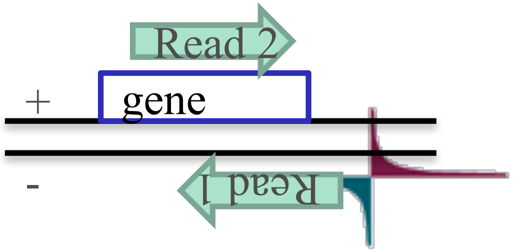
1+-,1-+,2++,2--

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

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

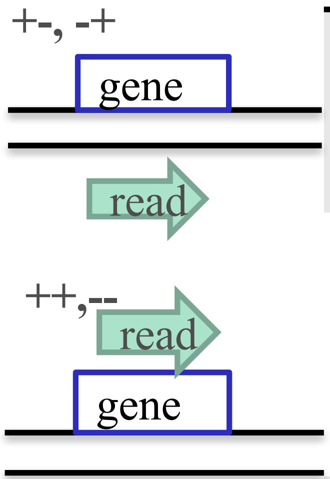
read2 mapped to '+' strand indicates parental gene on '+' strand

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



Stranded / directional RNA-seq data

- **Important to indicate which one was used in some analysis tools**
  - parameter naming differs in different tools
  - You can check this with a RseQC tool



Strandedness:	TopHat	HISAT2	HTSeq
Read (1) and transcript on opposite strands	Fr-firststrand	--rna-strandedness R / RF	--stranded reverse
Read (1) and transcript on the same strand	Fr-secondstrand	--rna-strandedness F / FR	--stranded yes
No knowledge of where the read comes from	Fr-unstrand	default	--stranded no

Understanding your data analysis - why?

- **You know your own experiments best**
  - Biology involved (e.g. genes, pathways, etc)
  - Potential batch effects etc

- **You can tune the parameters, ”play around” and learn more about your data**
  - Bioinformaticians might not always be available when needed

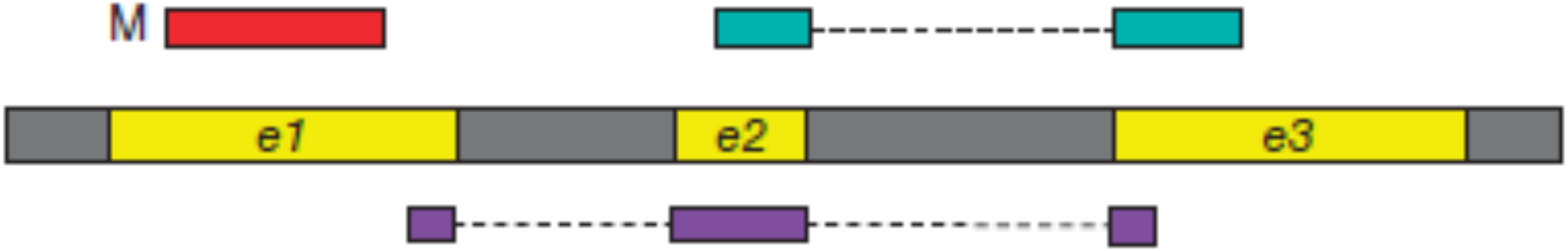
- **Allows you to design experiments better**
  - Enough replicates, reads etc → less money wasted

- **Allows you to discuss more easily with bioinformaticians**

RNAseq Alignment

Aligning reads to reference genome

- **The goal is to find the location where a read originated from**
- **Challenges**
  - Reads contain genomic variants and sequencing errors
  - Genomes contain non-unique sequence and introns
- **RNA-seq aligner needs to be able to map splice junction spanning reads to genome non-contiguously**
  - Spliced alignments are difficult because sequence signals at splice sites are limited, and introns can be thousands of bases long



Modified from Kim et al (2015) Nature methods 12:358

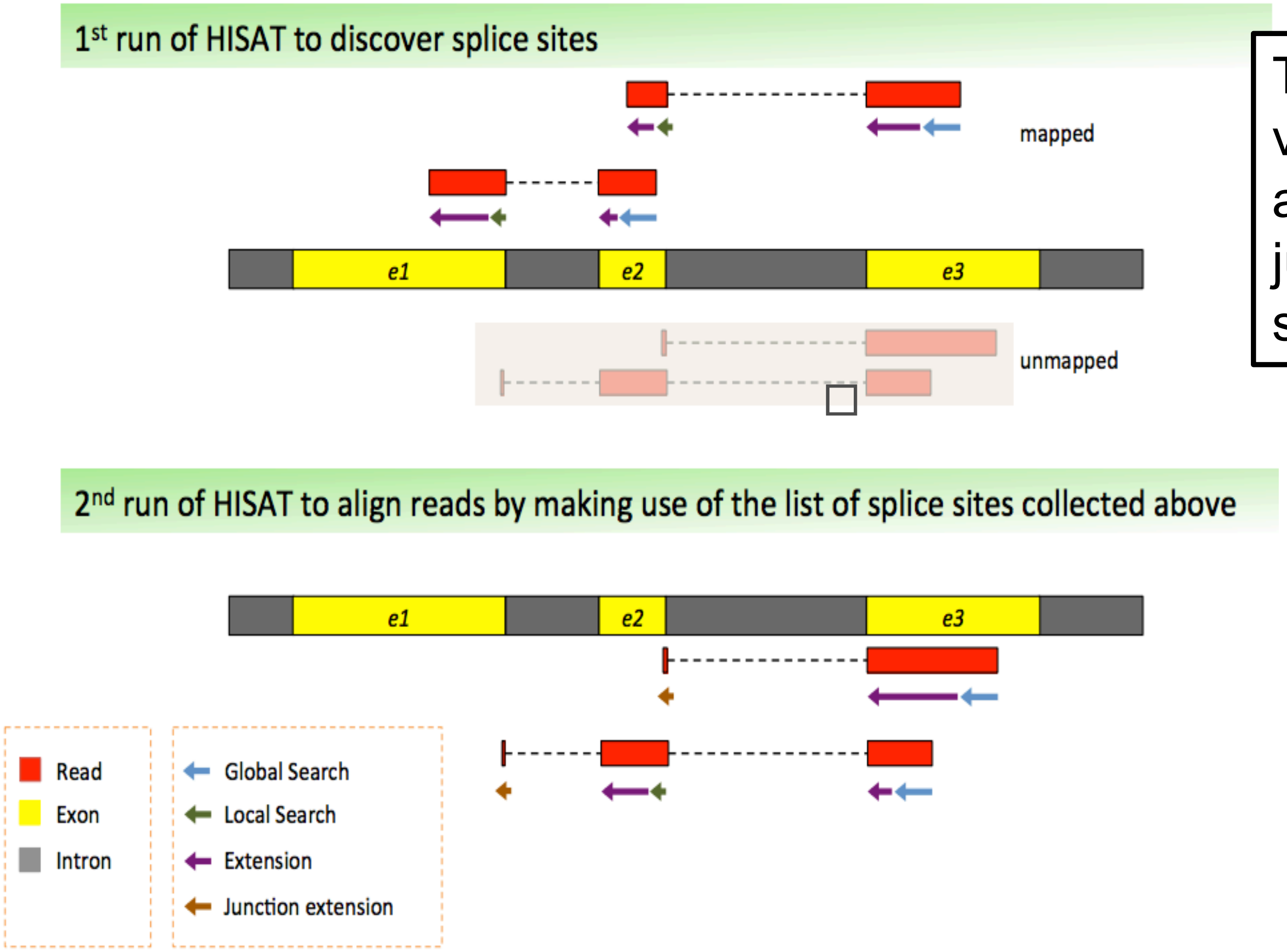
Alignment programs

- **Many aligners have been developed over the years**
  - Convert genome fasta file to a data structure which is faster to search (e.g. BWT index or suffix array)
  - Differ in speed, memory requirements, accuracy and ability to deal with spliced alignments
- **Use splice-aware aligner for mapping RNA-seq reads**
  - Examples:
    - STAR (fast and accurate, needs a lot of memory)
    - HISAT2 (fast and accurate, creating the genomic index needs a LOT of memory)
    - TopHat2 (slower, needs less memory)

HISAT2

- **HISAT = Hierarchical Indexing for Spliced Alignment of Transcripts**
- **Fast spliced aligner with low memory requirement**
- **Reference genome is (BWT FM) indexed for fast searching**
- **Uses two types of indexes**
  - A global index: used to anchor a read in genome (28 bp is enough)
  - Thousands of small local indexes, each covering a genomic region of 56 Kbp: used for rapid extension of alignments (good for spliced reads with short anchors)
- **Uses splice site information found during the alignment of earlier reads in the same run**

HISAT/HISAT2: How it works


1st run of HISAT to discover splice sites
mapped
e1
e2
e3
unmapped
2nd run of HISAT to align reads by making use of the list of splice sites collected above
e1
e2
e3
Read
Exon
Intron
Global Search
Local Search
Extension
Junction extension

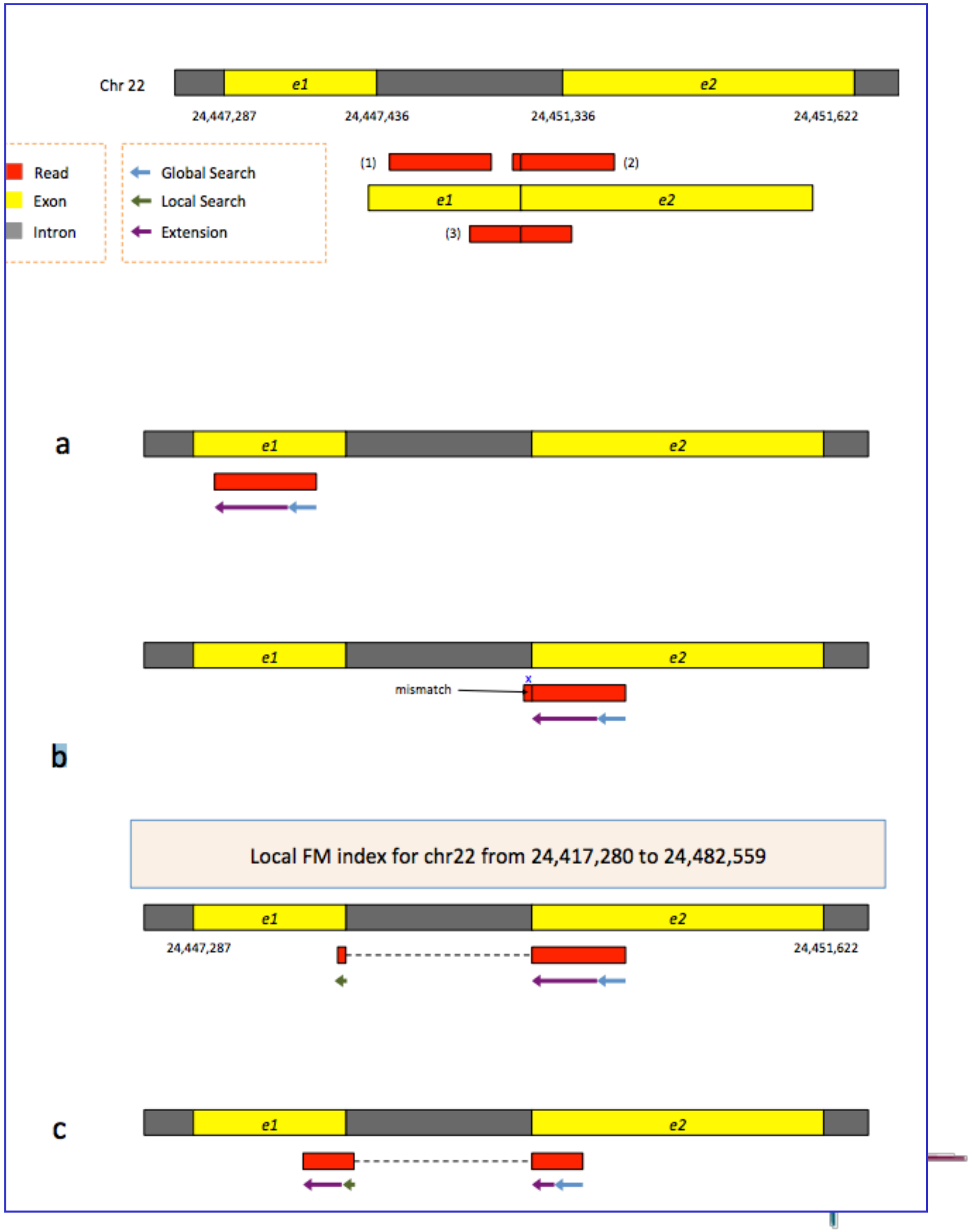
Two-step approach version of HISAT to allow alignment of junction reads with small anchors.

Kim et al (2015) Nature methods 12:358

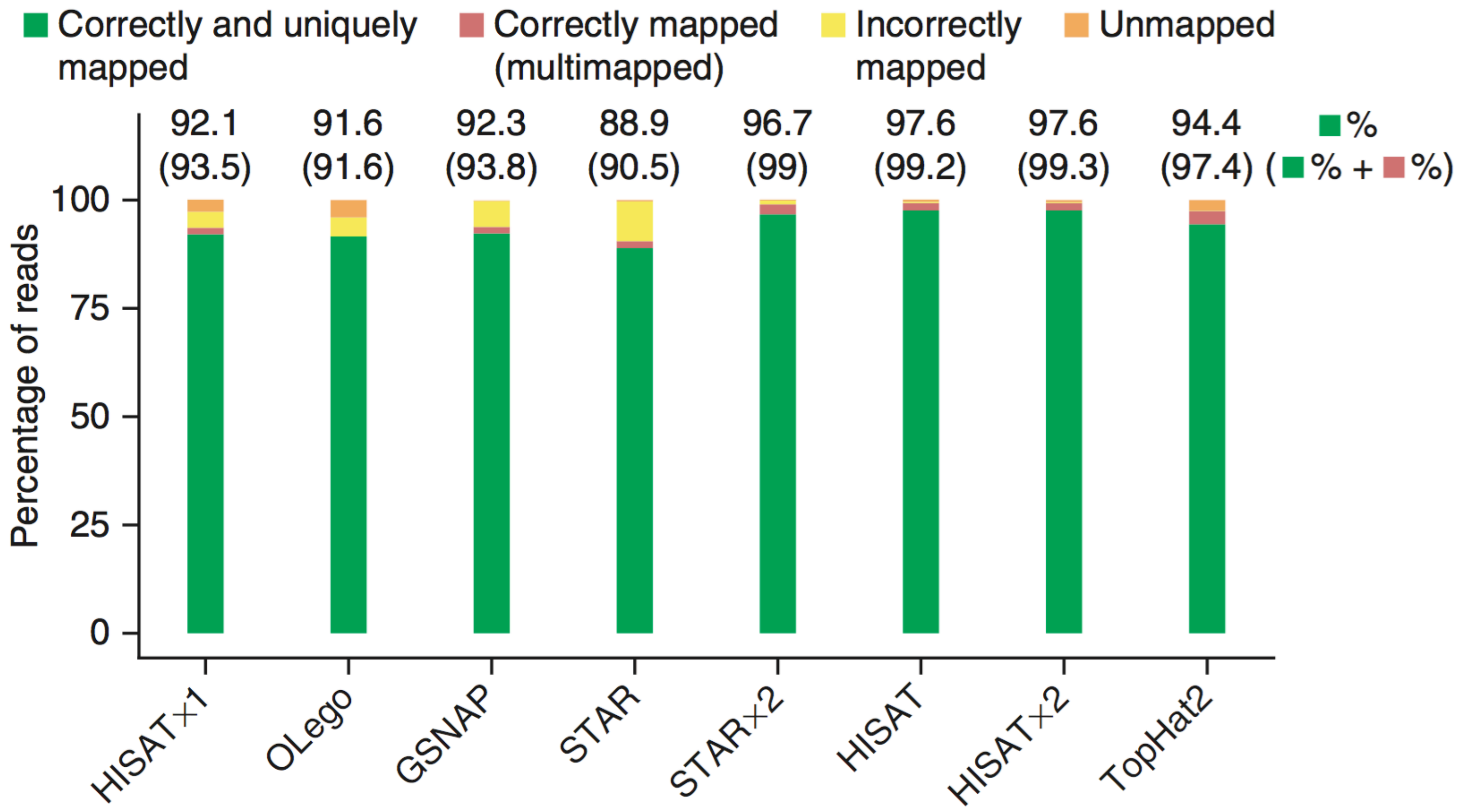
HISAT2 alignment: How it works?

- Uses an indexing scheme based on the Burrows-Wheelertransform and the Ferragina-Manzini (FM) index
- Use global search until exactly one match of at least 28bp (slower)
- Extend until mismatch is found (faster)
- Switch to local FM index to align remaining 8bp
- Extend again after junction if needed

Kim et al (2015) Nature methods 12:358



Use splice site information during read mapping to improve alignment accuracy



Kim D *et al.* HISAT: a fast spliced aligner with low memory requirements. Nat Methods. 2015 Apr;12(4):357-60.

HISAT2 – Indexing genome

- **Use splice sites and exon junction information**

```
hisat2_extract_splice_sites.py hg38chr19.gtf > splice_sites.txt
hisat2_extract_exons.py hg38chr19.gtf > exons.txt
```

- **Usage: hisat2-build [options]* <reference_in> <ht2_index_base>**

- **hisat2-build**

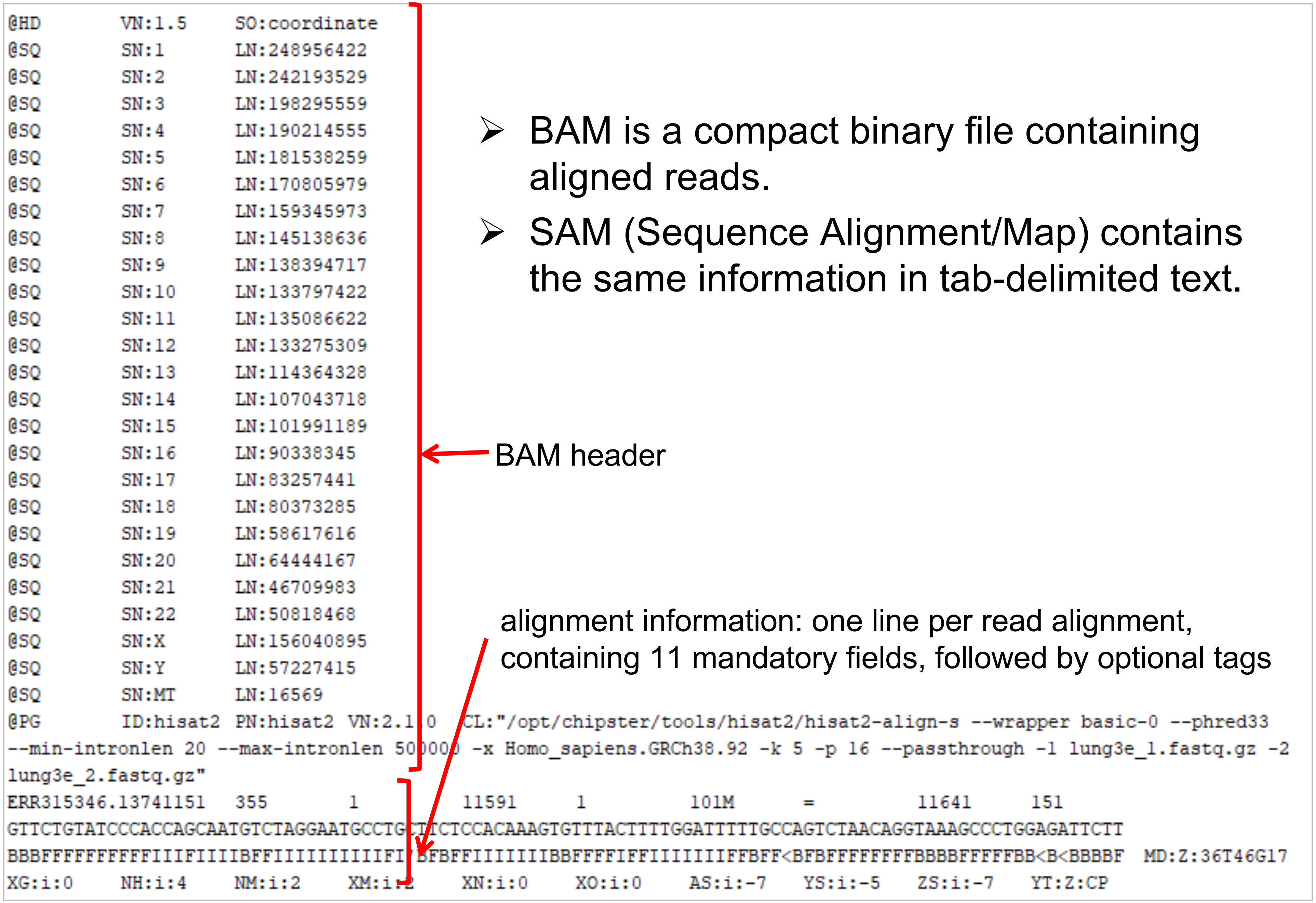
```
hisat2-build   – p 2
               --ss splice_sites.txt  \
               --exon exons.txt   \
               Homo_sapiens.GRCh38.dna.chromosome.19.fa \
               hs_19
```

HISAT2 – Read alignment

- **Usage: hisat2 [options]* -x <ht2-idx> {-1 <m1> -2 <m2> | -U <r> | --sra-acc <SRA accession number>} [-S <sam>]**

```
hisat2      - p  2  \
            - q    \
            -- rna-strandness F    \
            - x hisat-indexes/hs_19 \
            - U results-trimmomatic/hesc-trimmed.fq.gz   \
           - S results-hisat/hesc.sam
```

File format for mapped reads: BAM/SAM



```
@HD       VN:1.5    SO:coordinate
@SQ       SN:1      LN:248956422
@SQ       SN:2      LN:242193529
@SQ       SN:3      LN:198295559
@SQ       SN:4      LN:190214555
@SQ       SN:5      LN:181538259
@SQ       SN:6      LN:170805979
@SQ       SN:7      LN:159345973
@SQ       SN:8      LN:145138636
@SQ       SN:9      LN:138394717
@SQ       SN:10     LN:133797422
@SQ       SN:11     LN:135086622
@SQ       SN:12     LN:133275309
@SQ       SN:13     LN:114364328
@SQ       SN:14     LN:107043718
@SQ       SN:15     LN:101991189
@SQ       SN:16     LN:90338345
@SQ       SN:17     LN:83257441
@SQ       SN:18     LN:80373285
@SQ       SN:19     LN:58617616
@SQ       SN:20     LN:64444167
@SQ       SN:21     LN:46709983
@SQ       SN:22     LN:50818468
@SQ       SN:X      LN:156040895
@SQ       SN:Y      LN:57227415
@SQ       SN:MT     LN:16569
@PG       ID:hisat2 PN:hisat2 VN:2.1.0  CL:"/opt/chipster/tools/hisat2/hisat2-align-s --wrapper basic-0 --phred33
--min-intronlen 20 --max-intronlen 500000 -x Homo_sapiens.GRCh38.92 -k 5 -p 16 --passthrough -1 lung3e_1.fastq.gz -2
lung3e_2.fastq.gz"
ERR315346.13741151   355       1         11591     1         101M      =         11641     151
GTTCTGTATCCCACCAGCAATGTCTAGGAATGCCTGCTTCTCCACAAAGTGTTTACTTTTGGATTTTTGCCAGTCTAACAGGTAAAGCCCTGGAGATTCTT
BBBFFFFFFFFFFIIIFIIIIBFFIIIIIIIIIIIIFI'BFBFFIIIIIIIBBFFFFIFFIIIIIIIIFFBFF<BFBFFFFFFFFBBBBFFFFFBB<B<BBBBF  MD:Z:36T46G17
XG:i:0    NH:i:4    NM:i:2    XM:i:2    XN:i:0    XO:i:0    AS:i:-7   YS:i:-5   ZS:i:-7   YT:Z:CP
```

- BAM is a compact binary file containing aligned reads.
- SAM (Sequence Alignment/Map) contains the same information in tab-delimited text.

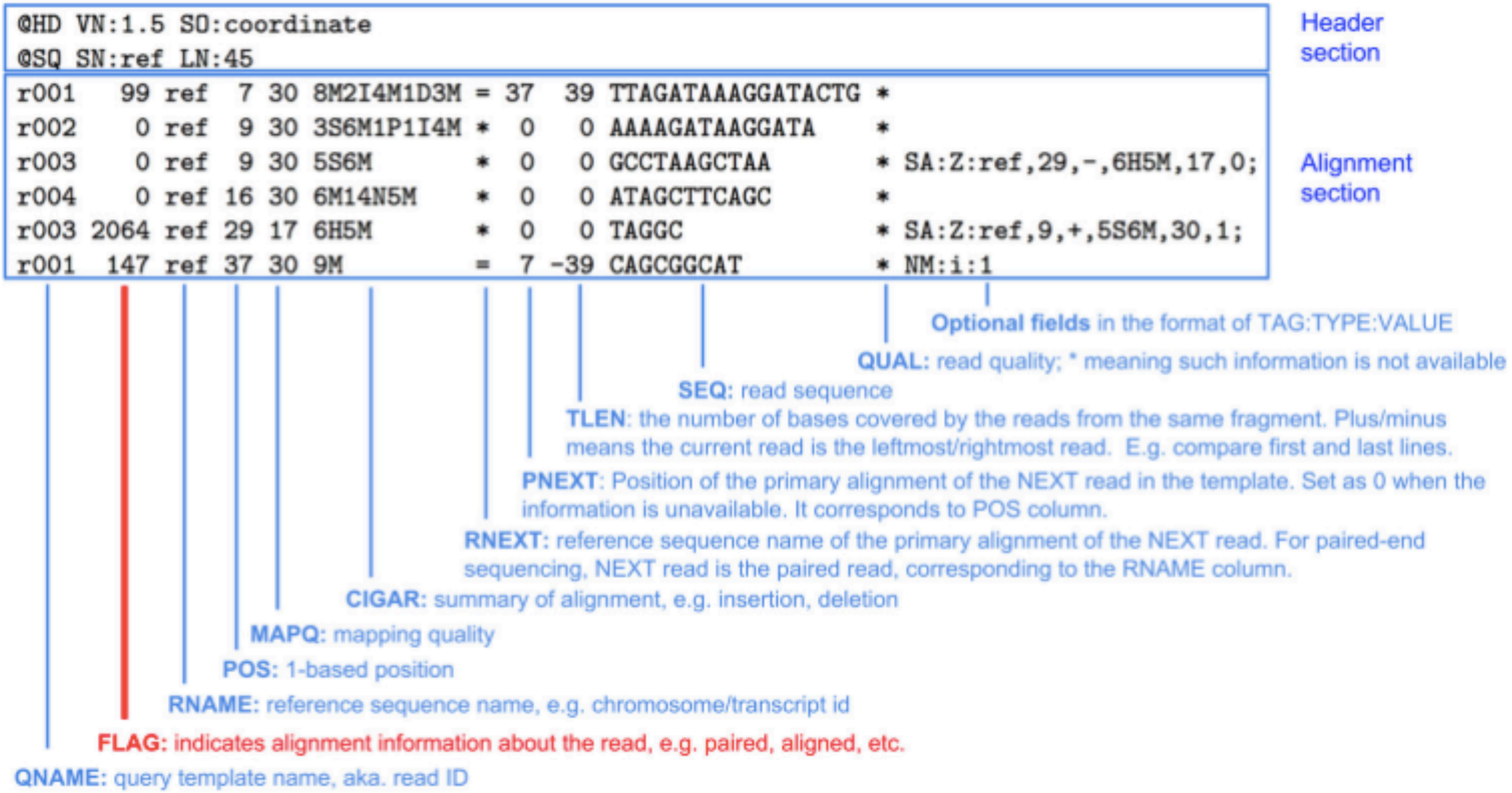
BAM header

alignment information: one line per read alignment, containing 11 mandatory fields, followed by optional tags

Fields in BAM/SAM files

➢ **read name**	HWI-EAS229_1:2:40:1280:283
➢ **flag**	272
➢ **reference name**	1
➢ **position**	18506
➢ **mapping quality**	0
➢ **CIGAR**	49M6183N26M
➢ **mate name**	*
➢ **mate position**	0
➢ **insert size**	0
➢ **sequence**	AGGGCCGATCTTGGTGCCATCCAGGGGGCCTCTACAAGGATAATCTGACCTGCTGAAGATGTCTCCAGAGACCTT
➢ **base qualities**	ECC@EEF@EB:EECFEECCCBEEEE;>5;2FBB@FBFEEFCF@FFFFCEFFFFFEE>FFEFC=@A;@>1@6.+5/5
➢ **tags**	MD:Z:75 NH:i:7 AS:i:-8 XS:A:-

- **Really nice pages for SAM/BAM interpretation: http://www.samformat.info**



```
@HD VN:1.5 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
```

Optional fields in the format of TAG:TYPE:VALUE

QUAL: read quality; * meaning such information is not available

SEQ: read sequence

TLEN: the number of bases covered by the reads from the same fragment. Plus/minus means the current read is the leftmost/rightmost read. E.g. compare first and last lines.

PNEXT: Position of the primary alignment of the NEXT read in the template. Set as 0 when the information is unavailable. It corresponds to POS column.

RNEXT: reference sequence name of the primary alignment of the NEXT read. For paired-end sequencing, NEXT read is the paired read, corresponding to the RNAME column.

CIGAR: summary of alignment, e.g. insertion, deletion

MAPQ: mapping quality

POS: 1-based position

RNAME: reference sequence name, e.g. chromosome/transcript id

FLAG: indicates alignment information about the read, e.g. paired, aligned, etc.

QNAME: query template name, aka. read ID

Mapping quality

- **Confidence in read's point of origin**
- **Depends on many things, including**
  - uniqueness of the aligned region in the genome
  - length of alignment
  - number of mismatches and gaps
- **Expressed in Phred scores, like base qualities**
  - Q = -10 * $\log_{10}$ (probability that mapping location is wrong)
- **Values differ in different aligners. E. g. unique mapping is**
  - 60 in HISAT2
  - 255 in STAR
  - 50 in TopHat
  - https://sequencing.qcfail.com/articles/mapq-values-are-really-useful-but-their-implementation-is-a-mess/

CIGAR string

- M = match or mismatch
- I = insertion
- D = deletion
- N = intron (in RNA-seq read alignments)
- S = soft clip (ignore these bases)
- H = hard clip (ignore and remove these bases)

- Example:

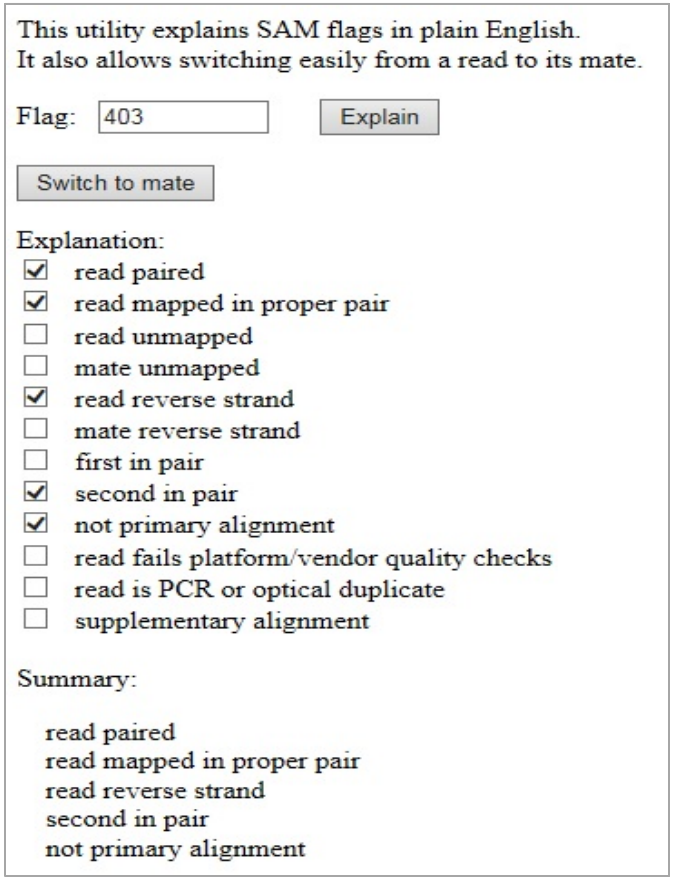
```
@HD VN:1.3 SO:coordinate
@SQ SN:ref LN:45
r001  163  ref  7  30  8M2I4M1D3M  =  37  39  TTAGATAAAGGATACTG  *
```

- The corresponding alignment

```
Ref   AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT
r001        TTAGATAAAGGATA*CTG
```

Flag field in BAM

- **Read's flag number is a sum of values**
  - E.g. 4 = unmapped, 1024 = duplicate
  - Explained in detail at http://samtools.github.io/hts-specs/SAMv1.pdf
  - You can interpret them at http://broadinstitute.github.io/picard/explain-flags.html



This utility explains SAM flags in plain English.
It also allows switching easily from a read to its mate.

Flag: 403 Explain

Switch to mate

Explanation:

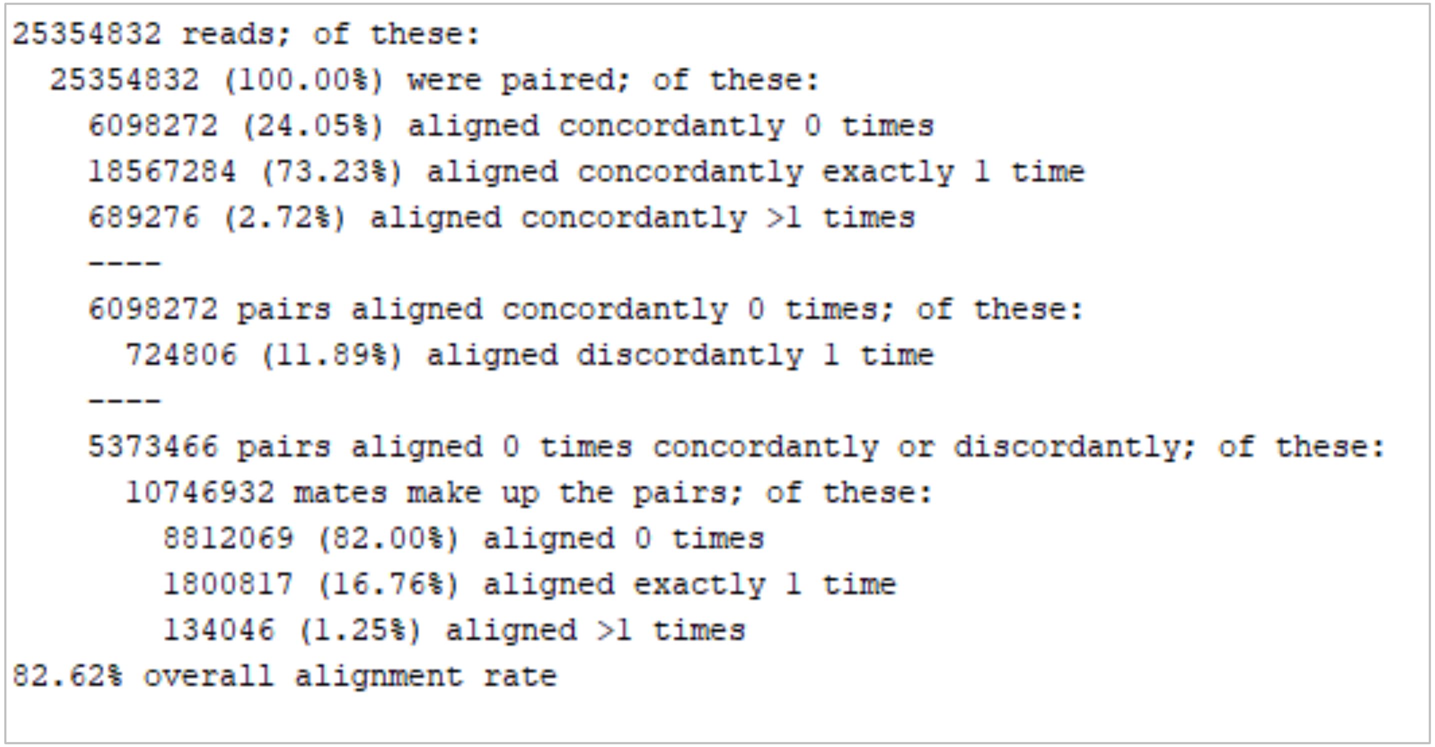
☑ read paired
☑ read mapped in proper pair
□ read unmapped
□ mate unmapped
☑ read reverse strand
□ mate reverse strand
□ first in pair
☑ second in pair
☑ not primary alignment
□ read fails platform/vendor quality checks
□ read is PCR or optical duplicate
□ supplementary alignment

Summary:

read paired
read mapped in proper pair
read reverse strand
second in pair
not primary alignment

How did the alignment go? Check the log file

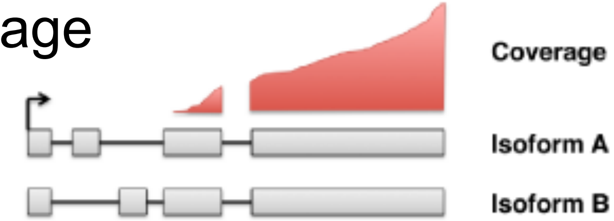
- **How many reads mapped to the reference?**
  - How many of them mapped uniquely?
- **How many pairs mapped?**
  - How many pairs mapped concordantly?
- **What was the overall alignment rate?**



```
25354832 reads; of these:
  25354832 (100.00%) were paired; of these:
    6098272 (24.05%) aligned concordantly 0 times
    18567284 (73.23%) aligned concordantly exactly 1 time
    689276 (2.72%) aligned concordantly >1 times
    ----
    6098272 pairs aligned concordantly 0 times; of these:
      724806 (11.89%) aligned discordantly 1 time
    ----
    5373466 pairs aligned 0 times concordantly or discordantly; of these:
      10746932 mates make up the pairs; of these:
        8812069 (82.00%) aligned 0 times
        1800817 (16.76%) aligned exactly 1 time
        134046 (1.25%) aligned >1 times
82.62% overall alignment rate
```

Full alignment or lightweight mapping?

- **Aligning reads to reference genome is slow → many quantitation tools offer now lightweight ”mapping”**
  - selective alignment (Salmon)
  - quasi-mapping (Sailfish, Salmon)
  - pseudoalignment (kallisto)
- **These tools match reads to transcripts and report transcripts that a read is compatible with (no base-to-base alignments)**
  - Difficult to assign reads to isoforms because they share exons, and technical biases cause non-uniform coverage
  - Need complete transcriptome



- **Srivastava et al 2019: Alignment and mapping methodology influence transcript abundance estimation**
  - Quantification accuracy is better when using traditional alignments

Alignment Practicals

- **Make an index file for HISAT2**
- **Align reads to reference genome with HISAT2**

Data analysis workflow

- Quality control of raw reads
- Preprocessing (trimming / filtering) if needed
- Alignment to reference genome
- **Alignment level quality control**
- Quantitation
- Experiment level quality control
- Differential expression analysis

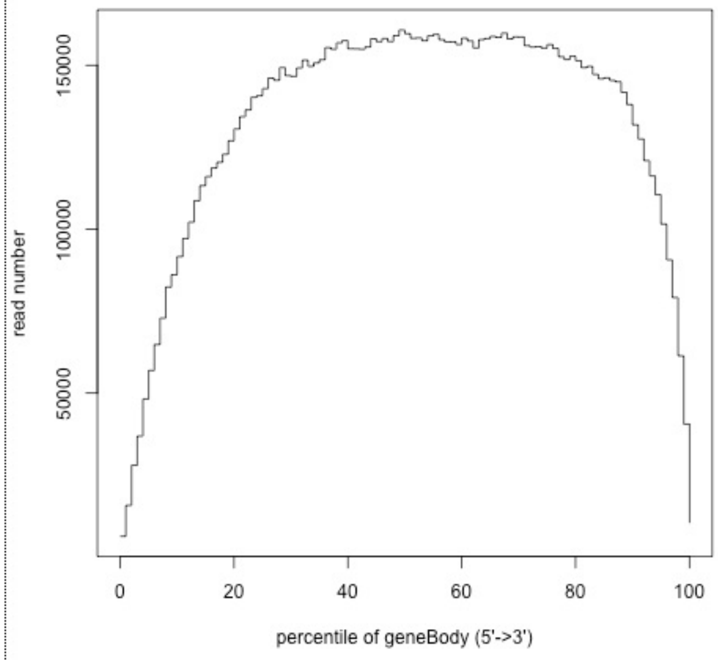
Alignment level quality control

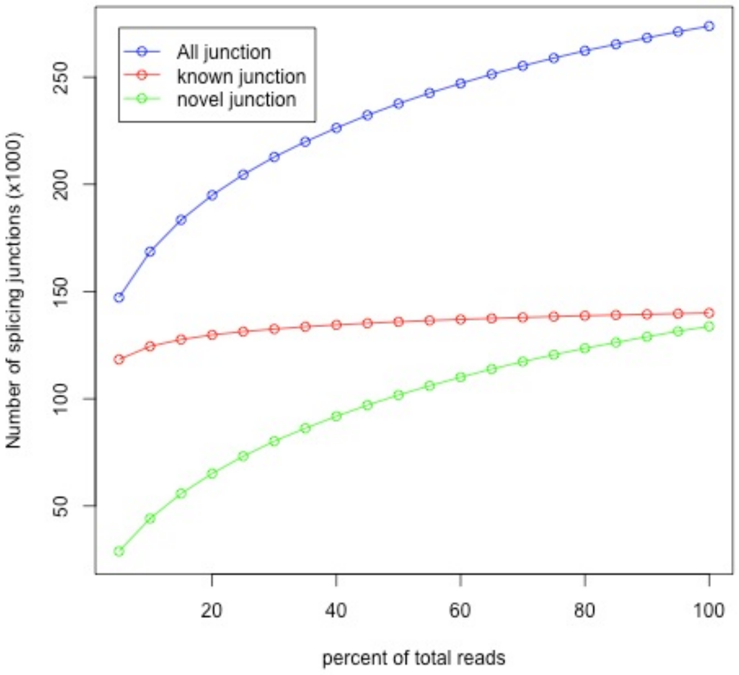
Annotation-based quality metrics

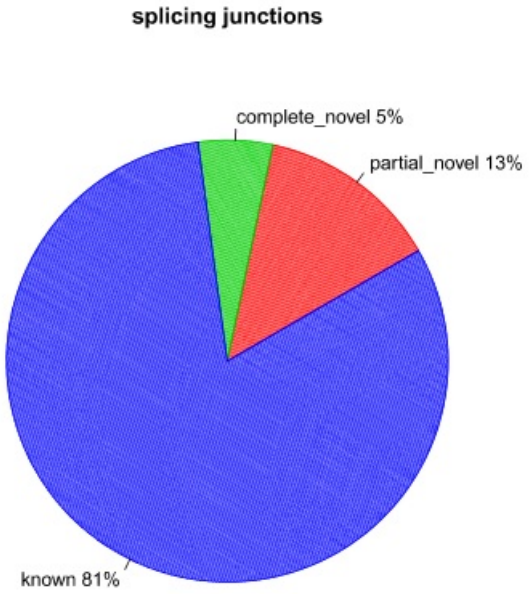
- **Saturation of sequencing depth**
  - Would more sequencing detect more genes and splice junctions?
- **Read distribution between different genomic features**
  - Exonic, intronic, intergenic regions
  - Coding, 3’ and 5’ UTR exons
  - Protein coding genes, pseudogenes, rRNA, miRNA, etc
- **Is read coverage uniform along transcripts?**
  - Biases introduced in library construction and sequencing
    - polyA capture and polyT priming can cause 3’ bias
    - random primers can cause sequence-specific bias
    - GC-rich and GC-poor regions can be under-sampled
  - Genomic regions have different mappabilities (uniqueness)

Quality assessment with RseQC

- Checks coverage uniformity, saturation of sequencing depth, novelty of splice junctions, read distribution between different genomic regions, etc.
- Takes a BAM file and a BED file
- Remember to check that the chromosome names match (chr1 vs 1)



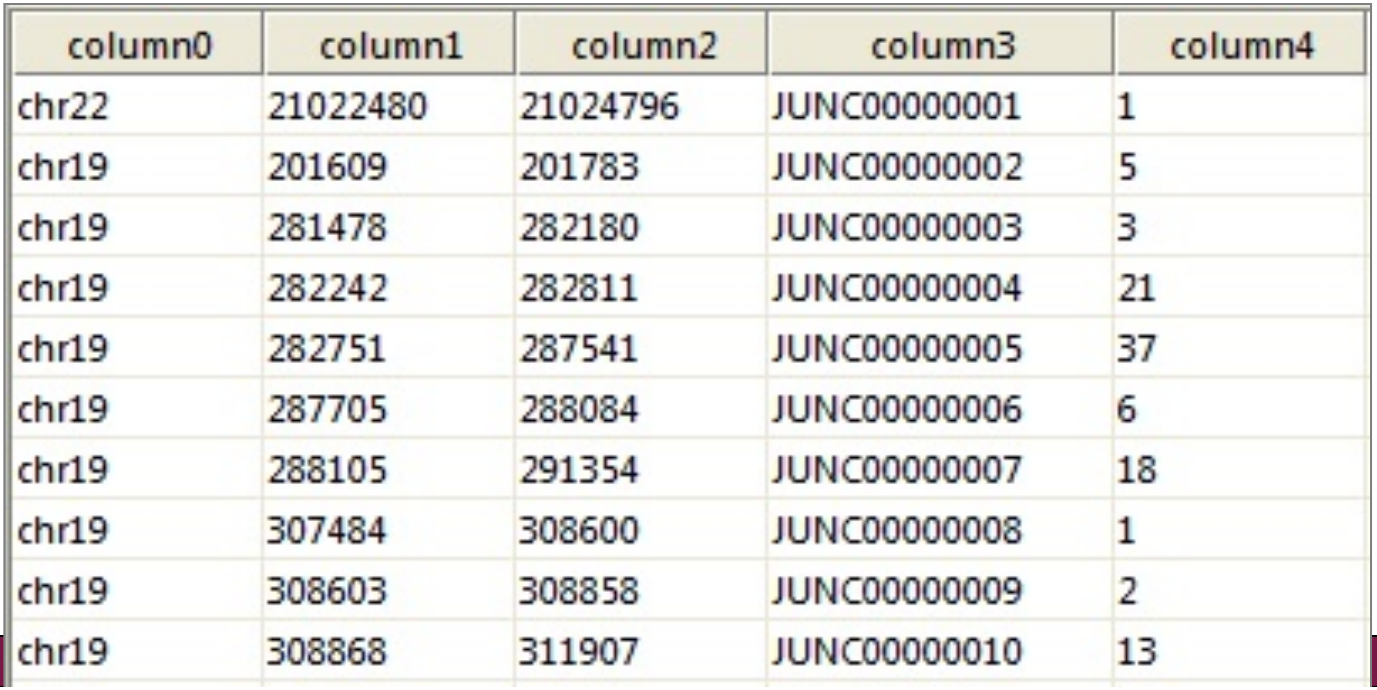




BED file format

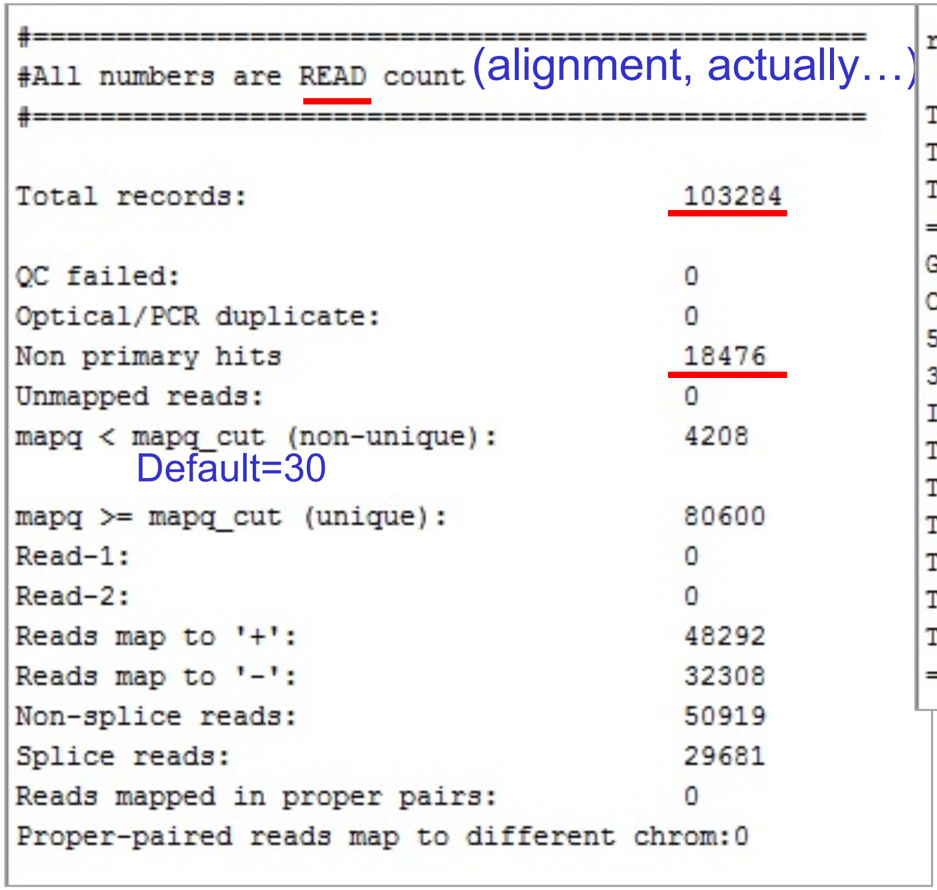
- BED (Browser extensible data) file format is used for reporting location of features (e.g. genes and exons) in a genome
- 5 obligatory columns: chr, start, end, name, score
- You can get a BED file with gene locations from UCSC Table

  Browser: https://genome.ucsc.edu/cgi-bin/hgTables
- Example of a BED file (with known junctions):



column0	column1	column2	column3	column4
chr22	21022480	21024796	JUNC00000001	1
chr19	201609	201783	JUNC00000002	5
chr19	281478	282180	JUNC00000003	3
chr19	282242	282811	JUNC00000004	21
chr19	282751	287541	JUNC00000005	37
chr19	287705	288084	JUNC00000006	6
chr19	288105	291354	JUNC00000007	18
chr19	307484	308600	JUNC00000008	1
chr19	308603	308858	JUNC00000009	2
chr19	308868	311907	JUNC00000010	13

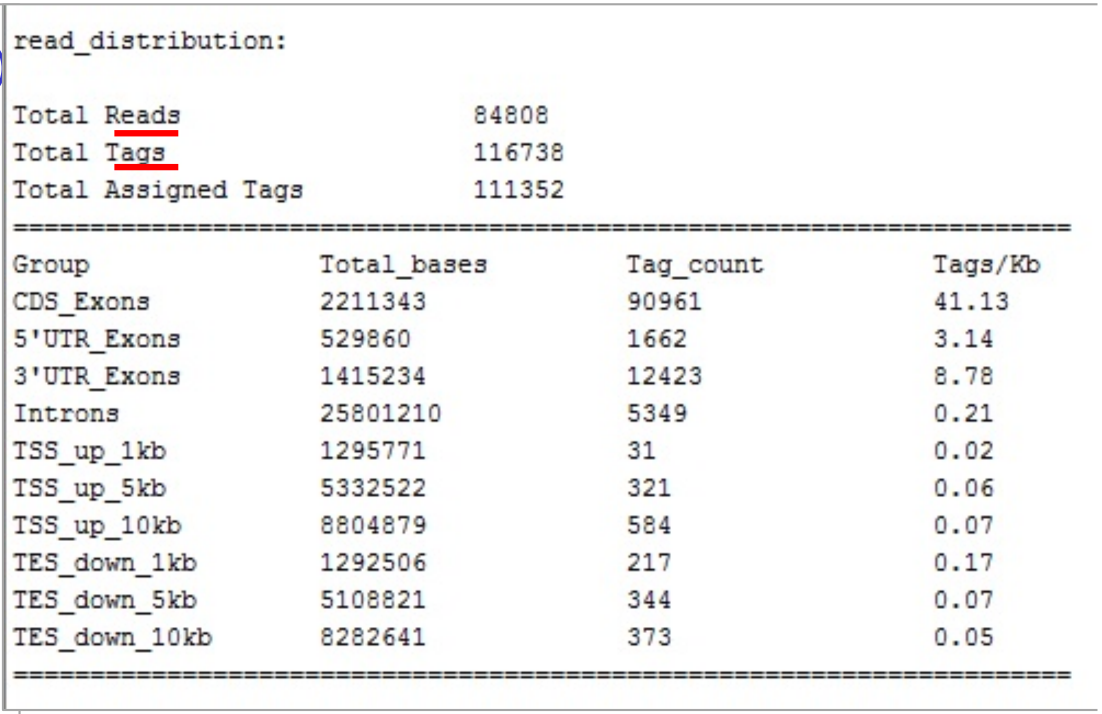
QC tables by RseQC



```
#==================================================
#All numbers are READ count
#==================================================

Total records:                          103284

QC failed:                              0
Optical/PCR duplicate:                  0
Non primary hits                        18476
Unmapped reads:                         0
mapq < mapq_cut (non-unique):           4208

mapq >= mapq_cut (unique):              80600
Read-1:                                 0
Read-2:                                 0
Reads map to '+':                       48292
Reads map to '-':                       32308
Non-splice reads:                       50919
Splice reads:                           29681
Reads mapped in proper pairs:           0
Proper-paired reads map to different chrom:0
```

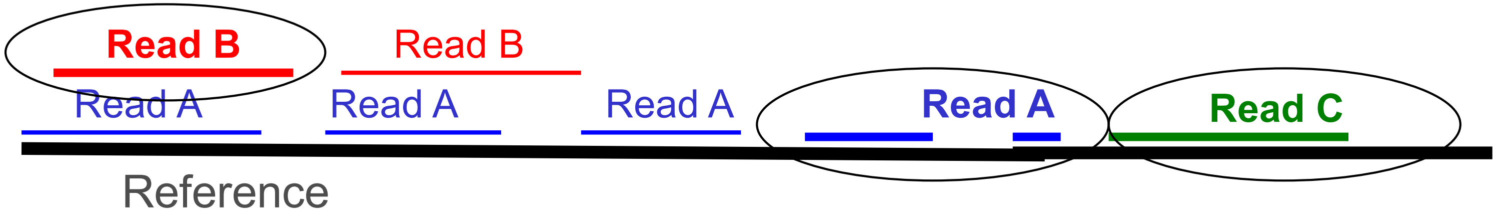
(alignment, actually…)

Default=30

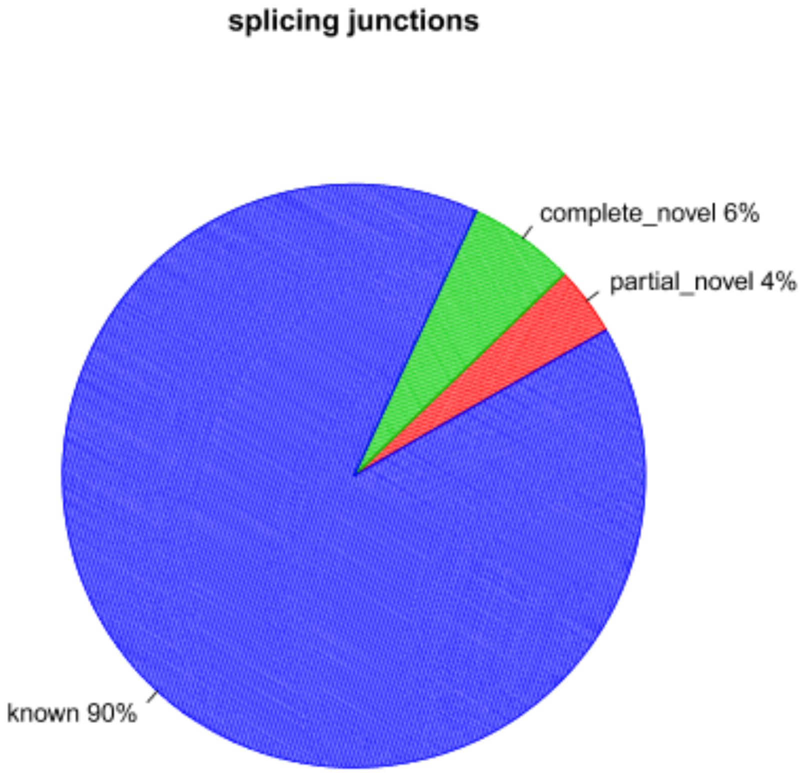


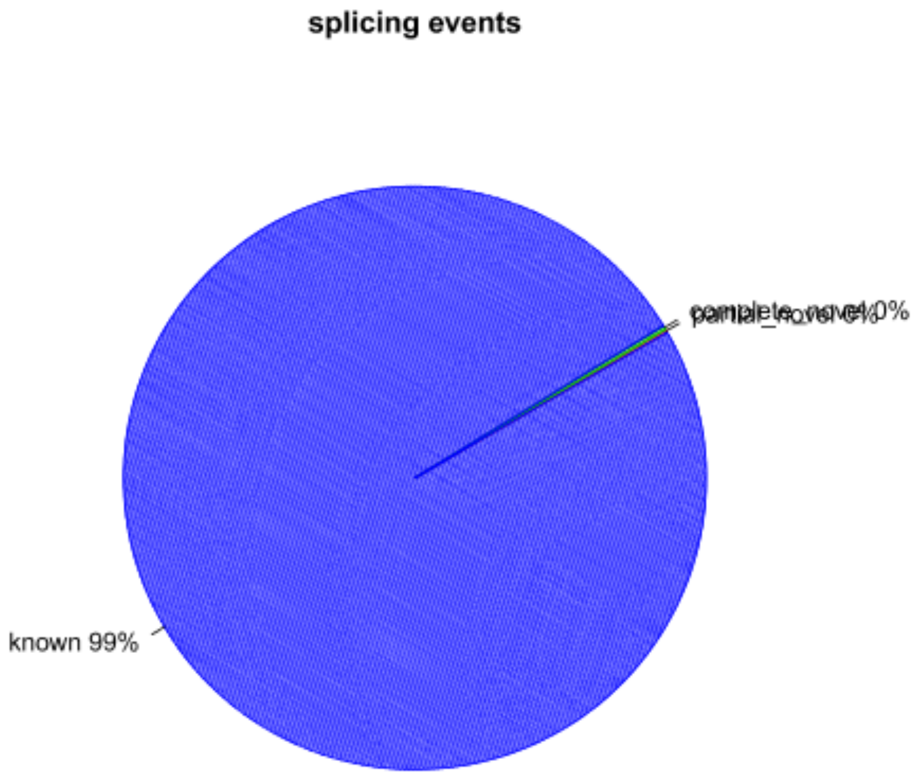
```
read_distribution:

Total Reads                   84808
Total Tags                    116738
Total Assigned Tags           111352
=====================================================================
Group               Total_bases         Tag_count           Tags/Kb
CDS_Exons           2211343             90961               41.13
5'UTR_Exons         529860              1662                3.14
3'UTR_Exons         1415234             12423               8.78
Introns             25801210            5349                0.21
TSS_up_1kb          1295771             31                  0.02
TSS_up_5kb          5332522             321                 0.06
TSS_up_10kb         8804879             584                 0.07
TES_down_1kb        1292506             217                 0.17
TES_down_5kb        5108821             344                 0.07
TES_down_10kb       8282641             373                 0.05
=====================================================================
```

Total records:	7
Non primary hits:	4
Total reads:	3
Total tags:	8



Splicing graphs by RseQC





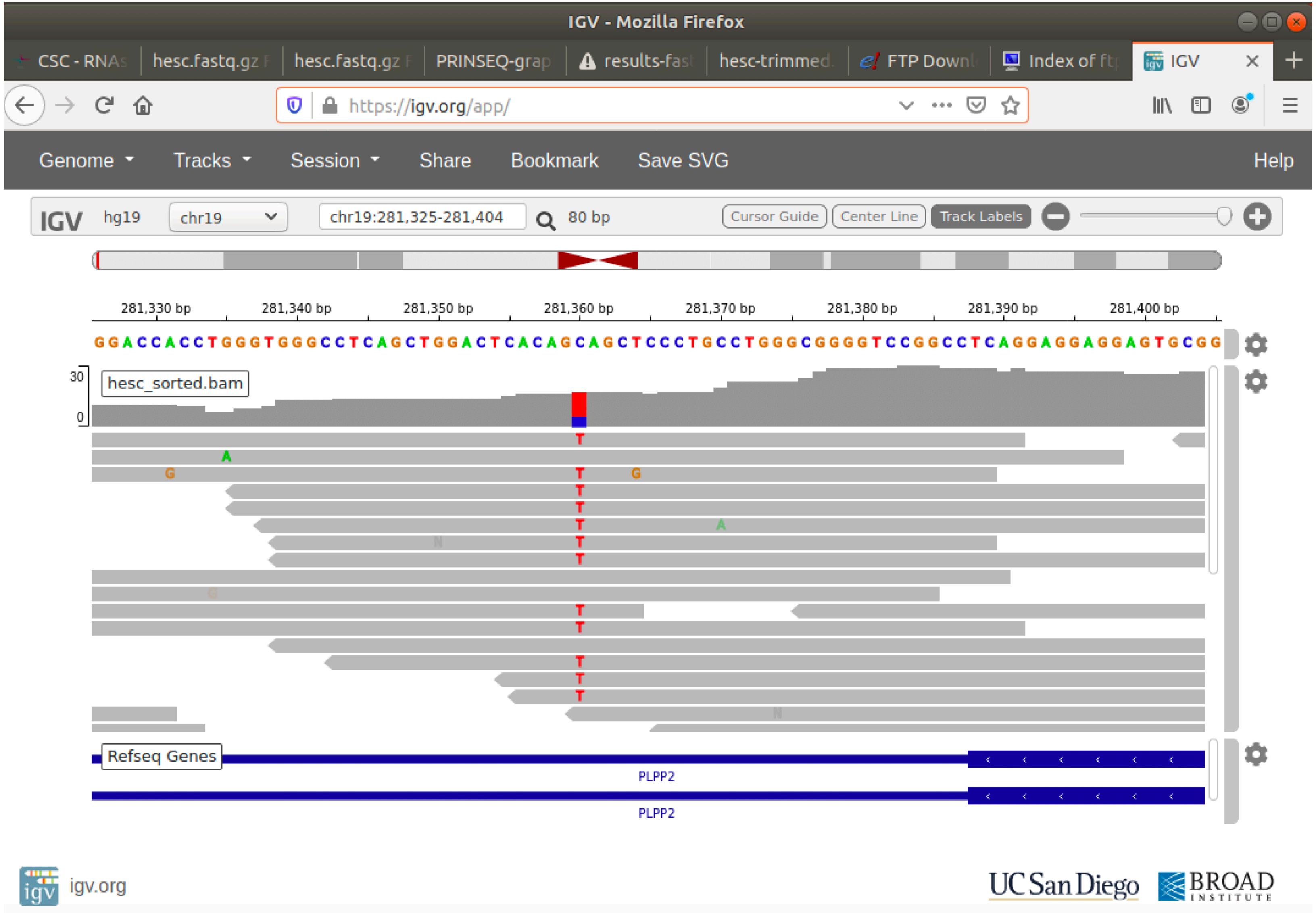
- **Splicing junction = exon-exon junction covered by one or more reads**
- **Splicing event = a read is split across a splice junction**

Visualisation: IGV Genomics Viewer

Visualisation: IGV Genomic Viewer

- **You can view your BAM files in IGV Genomic Viewer**
  1. Browse to https://igv.org/app/
  2. Upload your .bam and .bai files
  3. Go to known location, zoom in and out
- **What can you see?**
  - Reads/alignments
  - SNPs
  - Troubleshooting: your favorite gene should be expressed, but it is not counted by HTSeq. Are there any reads aligning to this location?
    - (If yes, the reason might be that they are aligning to other locations as well -non-unique- and thus not counted by HTSeq)

Visualisation: IGV Genomic Viewer



Did I accidentally sequence ribosomal RNA?

- **The majority of RNA in cells is rRNA**
- **Typically we want to sequence protein coding genes, so we try to avoid rRNA**
  - polyA capture
  - Ribominus kit (may not work consistently between samples)
- **How to check if we managed to avoid rRNA?**
  - RseQC might not be able to tell, if the rRNA genes are not in the BED file (e.g. in human the rRNA gene repeating unit has not been assigned to any chromosome yet)
  - You can map the reads to human ribosomal DNA repeating unit sequence (instead of the genome) with the Bowtie aligner, and check the alignment percentage

Data analysis workflow

- Quality control of raw reads
- Preprocessing (trimming / filtering) if needed
- Alignment to reference genome
- Alignment level quality control
- **Quantitation**
- Describing the experiment with phenodata
- Experiment level quality control
- Differential expression analysis

RNAseq quantification

Software for counting reads per genes or transcripts

- **HTSeq**

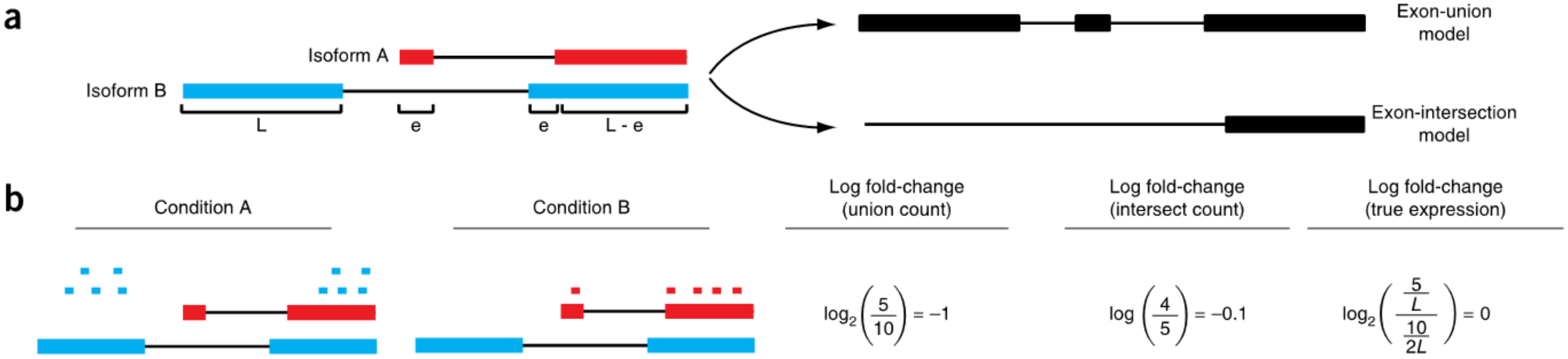
- StringTie
- Cufflinks

- Salmon
- Kallisto
- …

Counting reads per genes with HTSeq

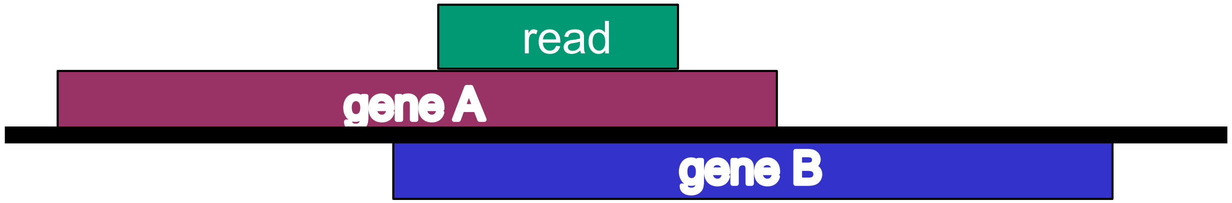
- **Given a BAM file and a GTF file with gene locations, counts how many reads map to each gene.**
  - A gene is considered as the union of all its exons.
  - Reads can be counted also per exons.
- **Use again Ensembl GTF files (or similar)**
  - Note that GTF and BAM must use the same chromosome naming
  - All exons of a gene must have the same gene_id (avoid UCSC GTFs)
- **Multimapping reads and ambiguous reads are not counted**
- **3 modes to handle reads which overlap several genes**
  - Union (default), Intersection-strict, Intersection-nonempty
- **Attention: was your data made with stranded protocol?**
  - You need to select the right counting mode!

Estimating gene expression at gene level

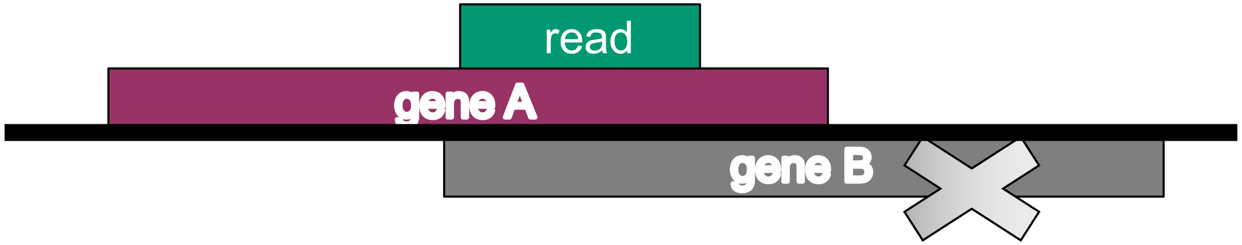

a
Isoform A
Isoform B
L
e
e
L - e
Exon-union model
Exon-intersection model
b
Condition A
Condition B
Log fold-change (union count)
Log fold-change (intersect count)
Log fold-change (true expression)
$\log_2\left(\frac{5}{10}\right) = -1$
$\log\left(\frac{4}{5}\right) = -0.1$
$\log_2\left(\frac{\frac{5}{L}}{\frac{10}{2L}}\right) = 0$

Trapnell et al. Nature Biotechnology 2013

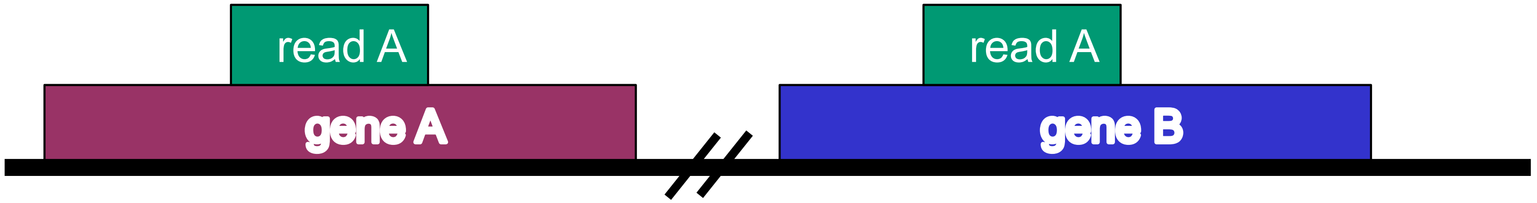
Not unique or ambiguous?



Ambiguous

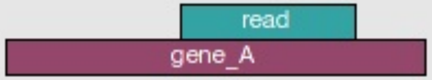
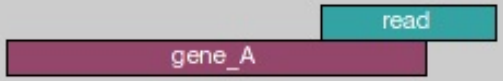
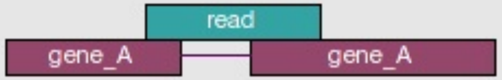
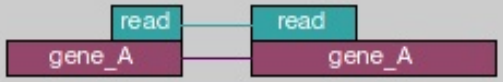
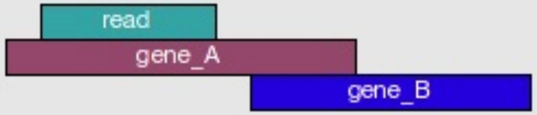
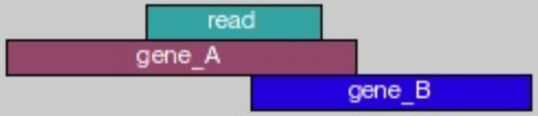
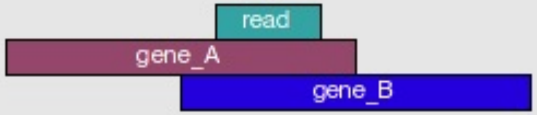


Stranded data
→ Not ambiguous



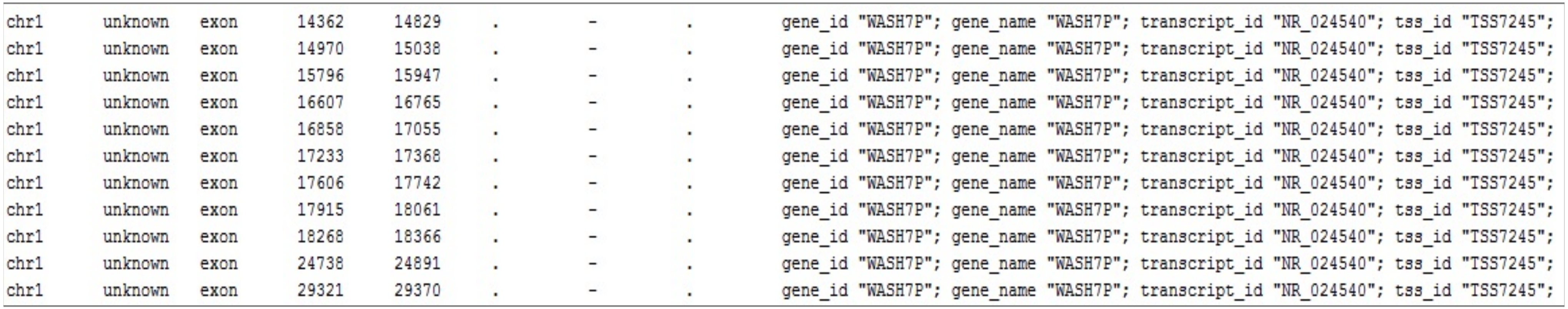
Multimapping
(not unique)

HTSeq count modes

	union	intersection _strict	intersection _nonempty
	gene_A	gene_A	gene_A
	gene_A	no_feature	gene_A
	gene_A	no_feature	gene_A
	gene_A	gene_A	gene_A
	gene_A	gene_A	gene_A
	ambiguous	gene_A	gene_A
	ambiguous	ambiguous	ambiguous

GTF file format

- **9 obligatory columns: chr, source, name, start, end, score, strand, frame, attribute**
- **1-based**
- **For HTSeq to work, all exons of a gene must have the same gene_id**
  - Use GTFs from Ensembl, avoid UCSC



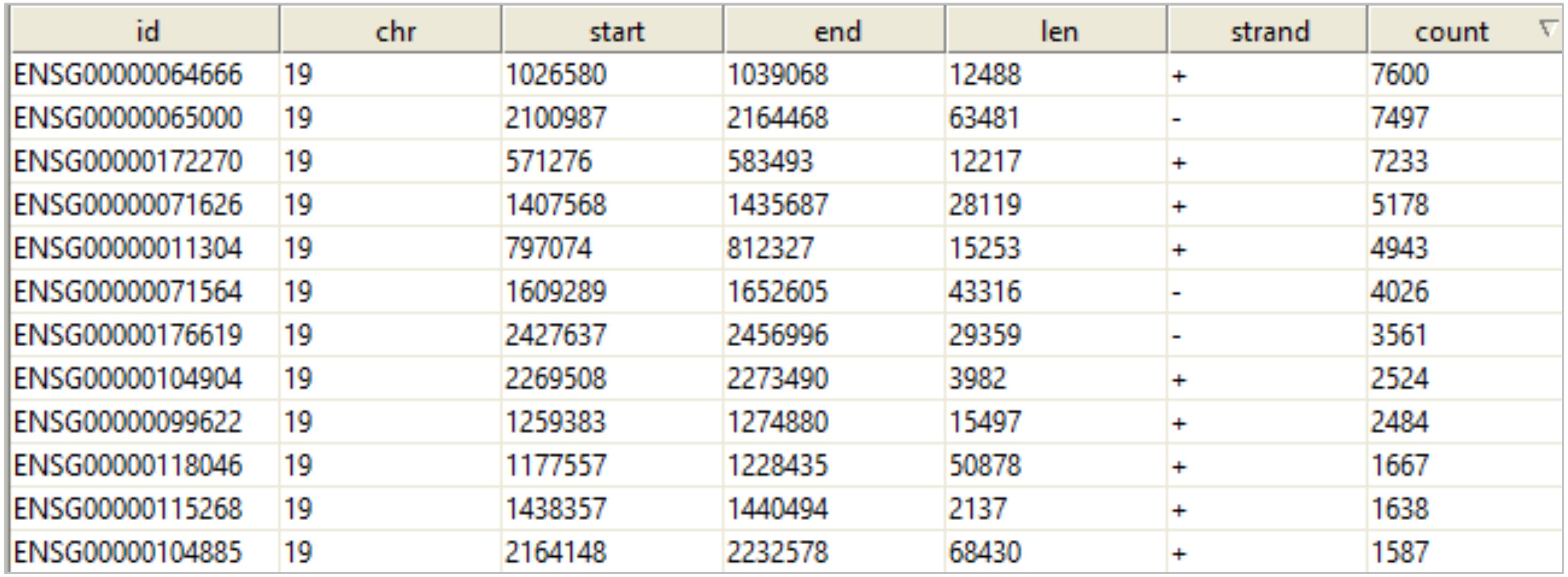
```
chr1	unknown	exon	14362	14829	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	14970	15038	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	15796	15947	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	16607	16765	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	16858	17055	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	17233	17368	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	17606	17742	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	17915	18061	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	18268	18366	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	24738	24891	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	29321	29370	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
```

HTSeq – Read counts per gene

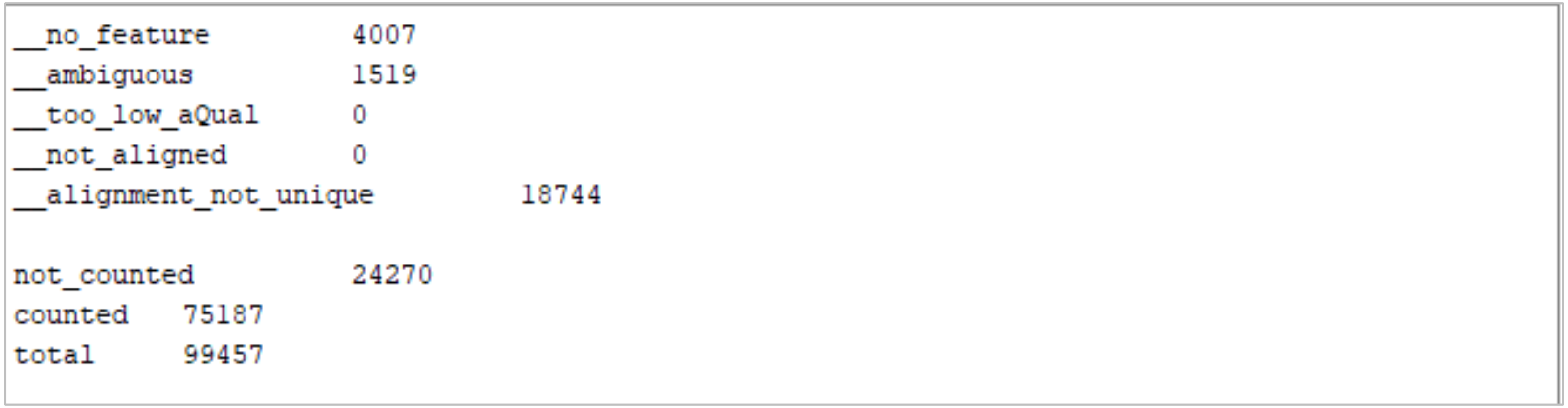
- **Usage: htseq-count [options] alignment_file gff_file**

-
```
htseq-count     --format=bam \
                --stranded=yes     \
                --mode=union     \
                --type=exon   \
                --idattr=gene_id  \
                results-hisat/hesc_sorted.bam  \
                hisat-indexes/hg38chr19.gtf
```

HTSeq result files: counts and info



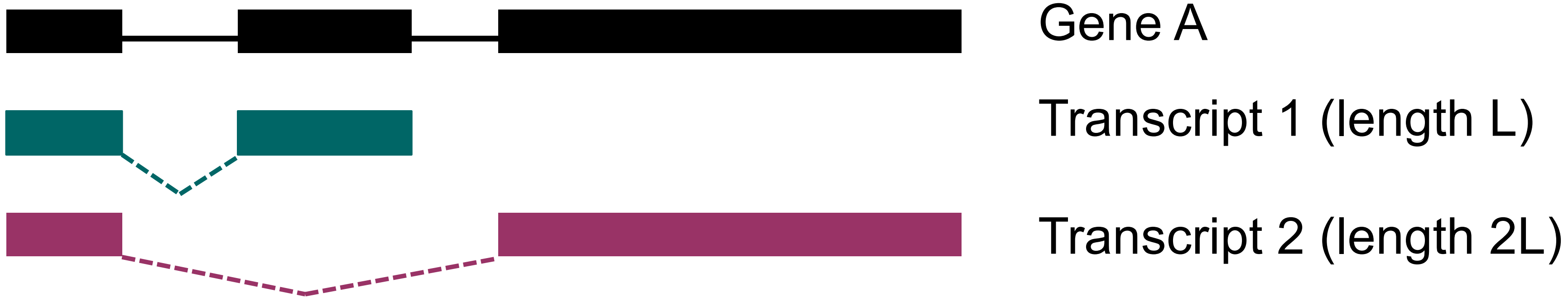
id	chr	start	end	len	strand	count
ENSG00000064666	19	1026580	1039068	12488	+	7600
ENSG00000065000	19	2100987	2164468	63481	-	7497
ENSG00000172270	19	571276	583493	12217	+	7233
ENSG00000071626	19	1407568	1435687	28119	+	5178
ENSG00000011304	19	797074	812327	15253	+	4943
ENSG00000071564	19	1609289	1652605	43316	-	4026
ENSG00000176619	19	2427637	2456996	29359	-	3561
ENSG00000104904	19	2269508	2273490	3982	+	2524
ENSG00000099622	19	1259383	1274880	15497	+	2484
ENSG00000118046	19	1177557	1228435	50878	+	1667
ENSG00000115268	19	1438357	1440494	2137	+	1638
ENSG00000104885	19	2164148	2232578	68430	+	1587

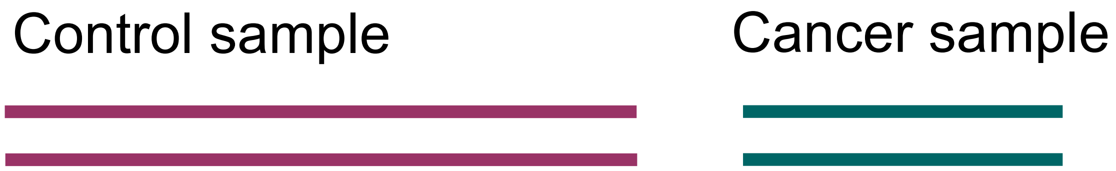


```
__no_feature          4007
__ambiguous           1519
__too_low_aQual       0
__not_aligned         0
__alignment_not_unique          18744

not_counted           24270
counted     75187
total       99457
```

Isoform switching can confound DGE analysis

- **The number of reads obtained from an expressed gene depends on the transcript length**
  - Longer transcripts produce more fragments and hence more reads
- **If a gene switches from one transcript isoform to another one, this can confound DGE analysis**

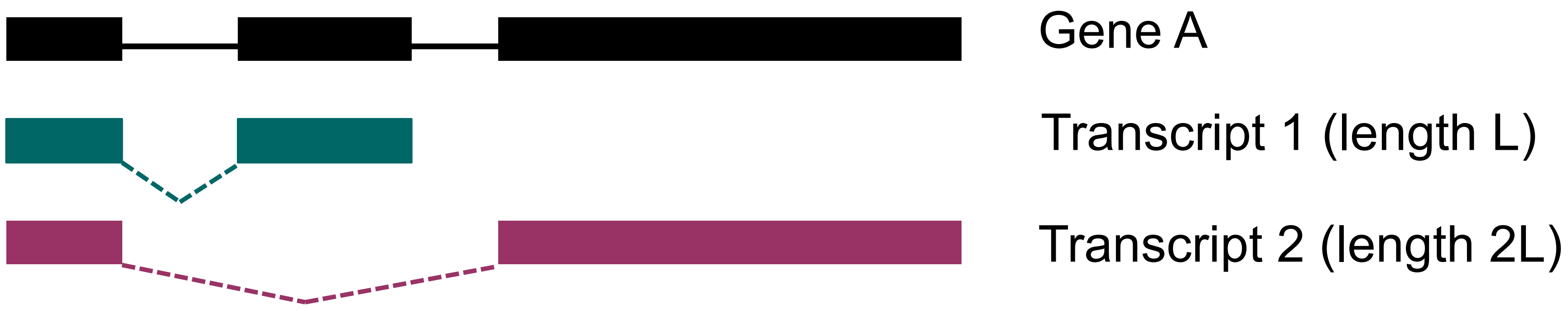


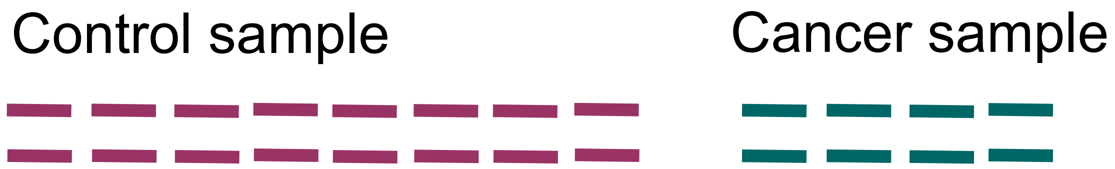


Expression level of gene A is the same in both samples, but cancer cells express the shorter isoform

Isoform switching can confound DGE analysis

- **The number of reads obtained from an expressed gene depends on the transcript length**
  - Longer transcripts produce more fragments and hence more reads
- **If a gene switches from one transcript isoform to another one, this can confound DGE analysis**



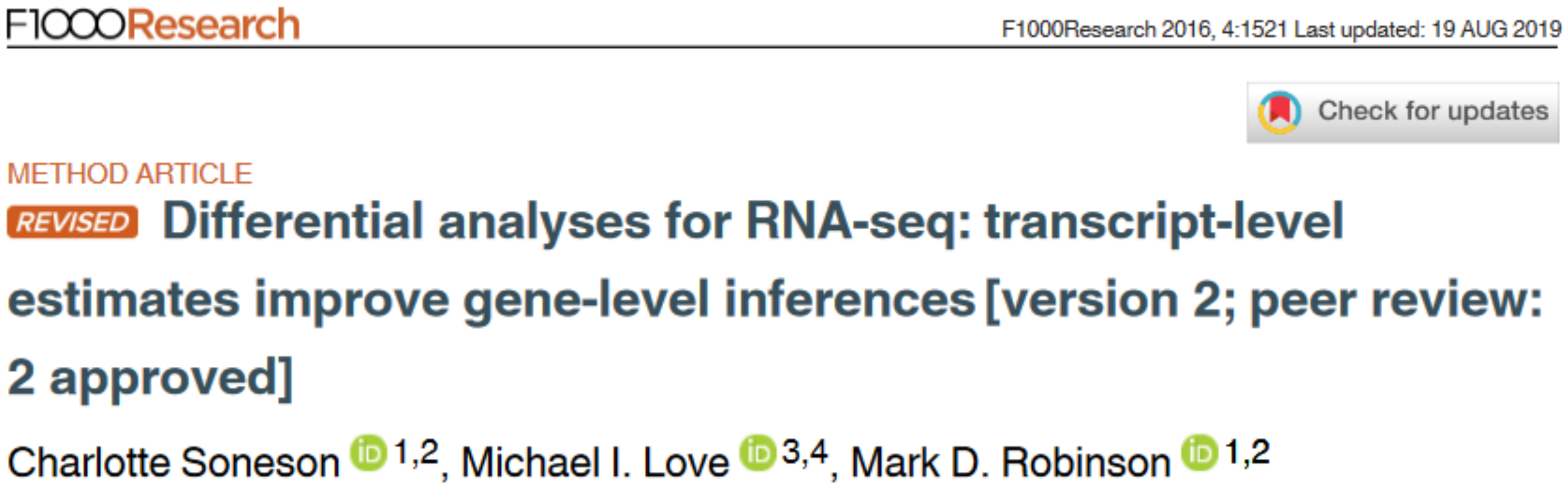


We get twice as many reads from the control sample
→ is gene A downregulated in cancer?

Should we quantitate at transcript level?

- **Gene-level quantitation is more accurate than transcript-level**
  - Technical biases cause non-uniform coverage → difficult to assign reads to different isoforms
  - High variation in abundance estimates of lowly expressed transcripts

- **BUT we can improve gene-level analysis by adjusting counts to reflect the underlying isoform composition!**



F1000Research

F1000Research 2016, 4:1521 Last updated: 19 AUG 2019

Check for updates

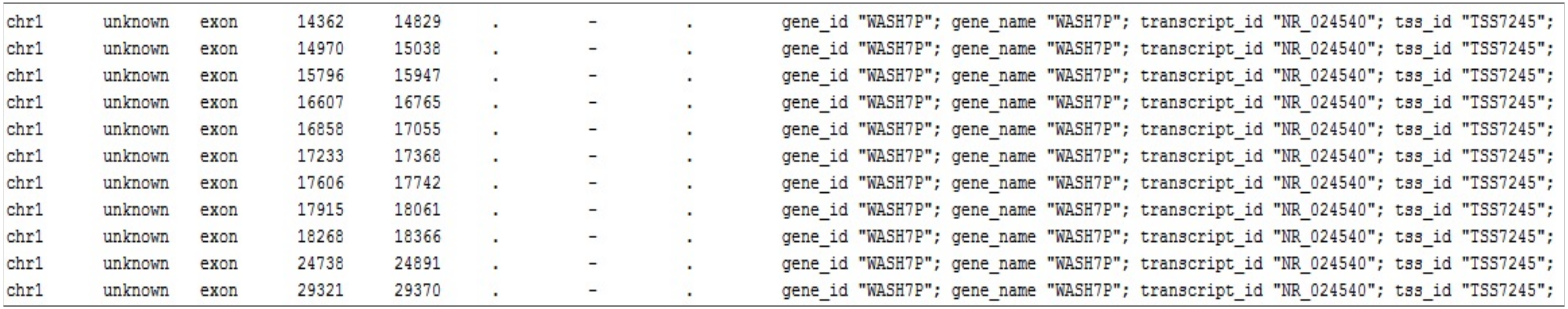
METHOD ARTICLE

REVISED **Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences [version 2; peer review: 2 approved]**

Charlotte Soneson [1,2], Michael I. Love [3,4], Mark D. Robinson [1,2]

GTF file format

- **9 obligatory columns: chr, source, name, start, end, score, strand, frame, attribute**
- **1-based**
- **For HTSeq to work, all exons of a gene must have the same gene_id**
  - Use GTFs from Ensembl, avoid UCSC



```
chr1	unknown	exon	14362	14829	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	14970	15038	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	15796	15947	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	16607	16765	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	16858	17055	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	17233	17368	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	17606	17742	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	17915	18061	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	18268	18366	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	24738	24891	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
chr1	unknown	exon	29321	29370	.	-	.	gene_id "WASH7P"; gene_name "WASH7P"; transcript_id "NR_024540"; tss_id "TSS7245";
```

Is isoform switching a major problem?

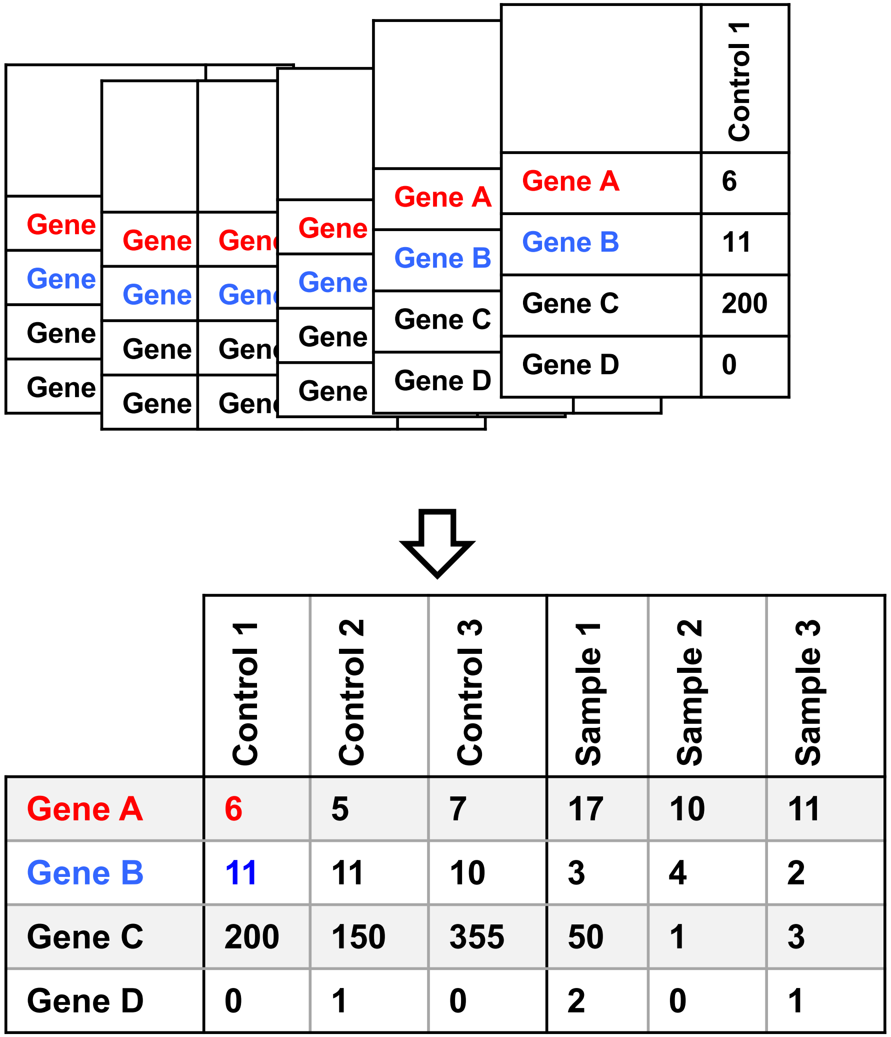
- **The magnitude of the effect depends on**
  - the extent of differential transcript usage (DTU)
  - the difference in length between the differentially expressed isoforms.
    - If the longer isoform is < 34% longer, false positives are controlled ok
    - Among all human transcript pairs in which both transcripts belong to the same gene, the median length ratio is 1.85
    - For one third of such pairs the longer isoform is < 38% longer

- **Many human genes express mainly one, dominant isoform**
  - → the global impact of isoform switching is relatively small in many real datasets (as opposed to simulated ones)

Soneson et al. F1000 Research 2016

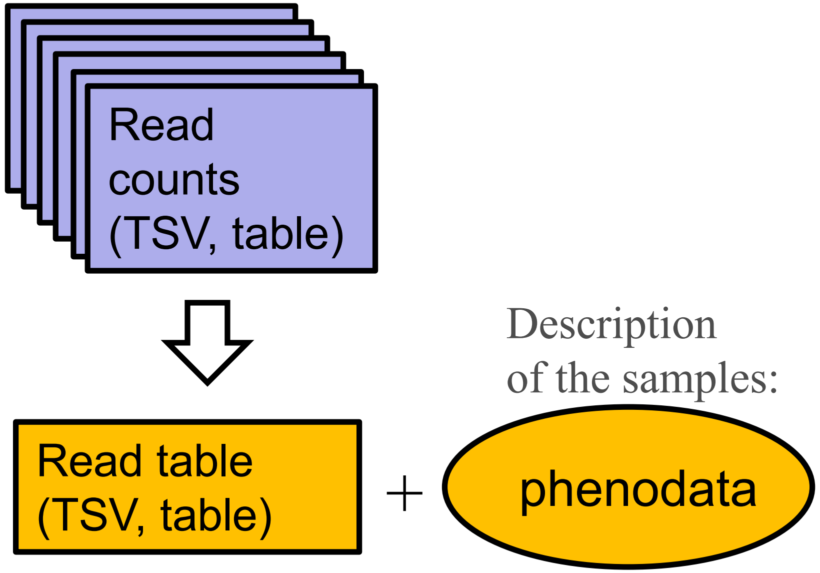
Combine individual count files into a count table

- **Combine the count files into one file**
- **We use a separate file for describing the experimental setup**



	Control 1
Gene A	6
Gene B	11
Gene C	200
Gene D	0

	Control 1	Control 2	Control 3	Sample 1	Sample 2	Sample 3
Gene A	6	5	7	17	10	11
Gene B	11	11	10	3	4	2
Gene C	200	150	355	50	1	3
Gene D	0	1	0	2	0	1



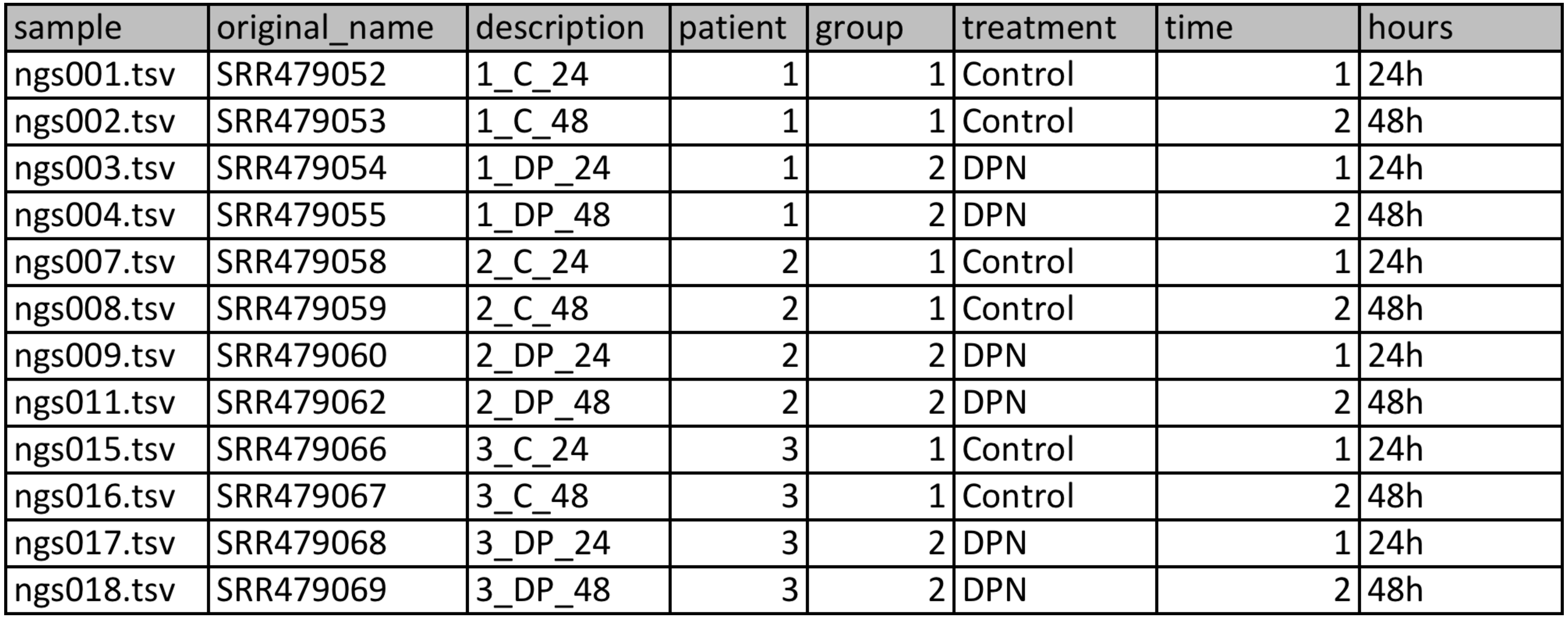
Read counts (TSV, table)

Read table (TSV, table) + phenodata

Description of the samples:

Describe the experiment (phenodata file)

- **Describe experimental groups, time, pairing etc with numbers**
  - e.g. 1 = control, 2 = cancer
- **Define sample names for visualizations**



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

Data analysis workflow

- Quality control of raw reads
- Preprocessing (trimming / filtering) if needed
- Alignment to reference genome
- Alignment level quality control
- Quantitation
- **Describing the experiment with phenodata**
- Experiment level quality control
- Differential expression analysis
- Visualization of reads and results in genomic context

Moving to R

- **So far we have used command line tools**
- **Now, we move the data to R and start using Bioconductor packages**
    - RStudio is installed in the VM used in the course
    - You can install R+RStudio on your own computer
    - …or use them in Puhti
- **The data is now MUCH smaller**
    - Instead of multiple sizable FASTQ and BAM files, we now have one table of gene counts
    - In our exercises, we now switch to a different dataset with 10 full-sized samples
- **It might be that you are starting the analysis at this point only**

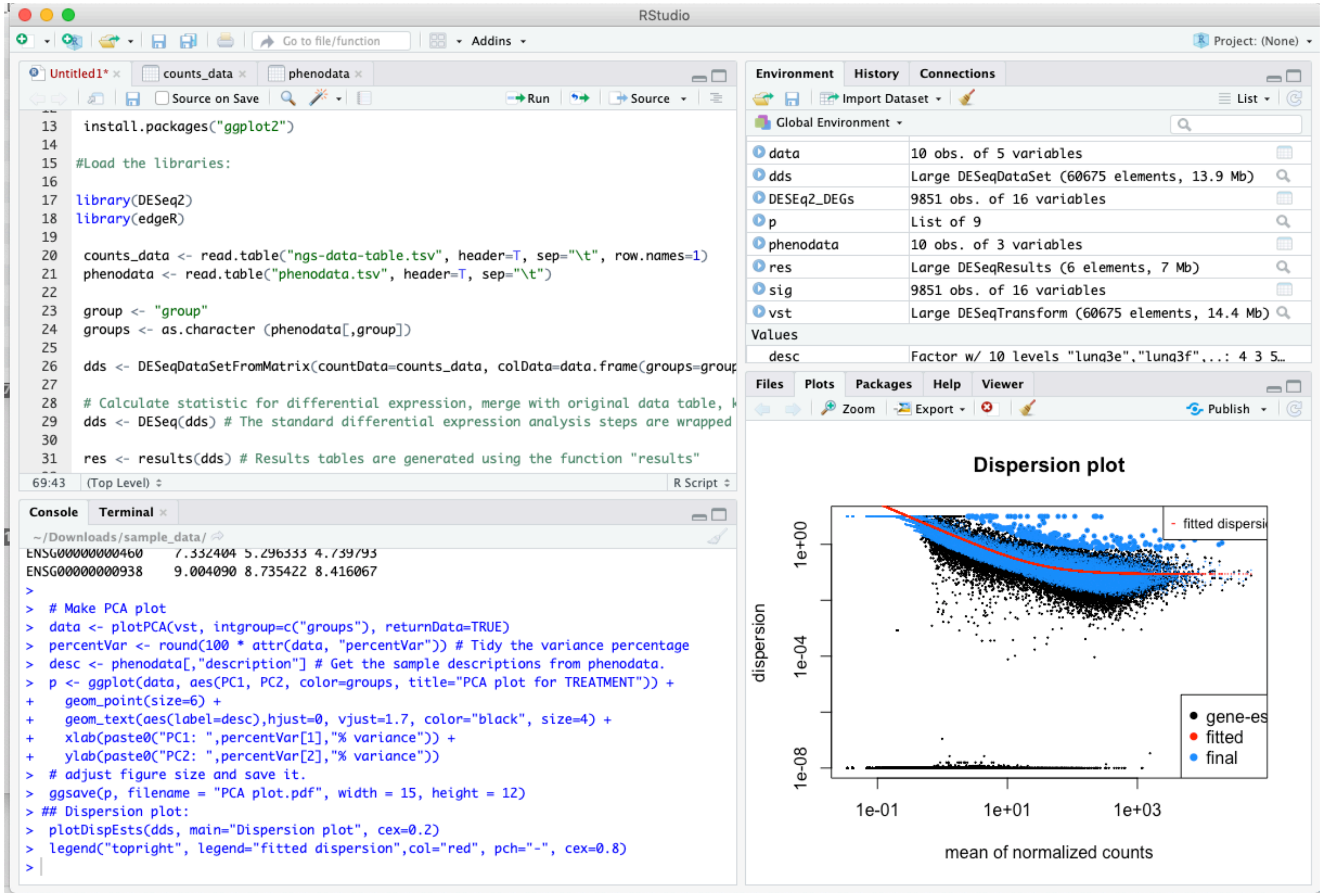
R, RStudio & Bioconductor

- **R:** free software environment for statistical computing and graphics
  - https://www.r-project.org
- **RStudio**: open source software for the R statistical computing environment –a GUI of sorts
  - https://rstudio.com
- **Bioconductor**: tools (“R packages”) for the analysis of high-throughput genomic data. Open source and open development.
  - We are using: DESeq2 and edgeR packages
  - https://www.bioconductor.org

Tiny recap of R

Your script: this is where you store your commands and notes!

The console: the commands go here



```
 install.packages("ggplot2")

#Load the libraries:

library(DESeq2)
library(edgeR)

 counts_data <- read.table("ngs-data-table.tsv", header=T, sep="\t", row.names=1)
 phenodata <- read.table("phenodata.tsv", header=T, sep="\t")

 group <- "group"
 groups <- as.character (phenodata[,group])

 dds <- DESeqDataSetFromMatrix(countData=counts_data, colData=data.frame(groups=grour

 # Calculate statistic for differential expression, merge with original data table, k
 dds <- DESeq(dds) # The standard differential expression analysis steps are wrapped

 res <- results(dds) # Results tables are generated using the function "results"
```

```
ENSG00000000460    7.332404 5.296333 4.739793
ENSG00000000938    9.004090 8.735422 8.416067
>
>  # Make PCA plot
>  data <- plotPCA(vst, intgroup=c("groups"), returnData=TRUE)
>  percentVar <- round(100 * attr(data, "percentVar")) # Tidy the variance percentage
>  desc <- phenodata[,"description"] # Get the sample descriptions from phenodata.
>  p <- ggplot(data, aes(PC1, PC2, color=groups, title="PCA plot for TREATMENT")) +
+    geom_point(size=6) +
+    geom_text(aes(label=desc),hjust=0, vjust=1.7, color="black", size=4) +
+    xlab(paste0("PC1: ",percentVar[1],"% variance")) +
+    ylab(paste0("PC2: ",percentVar[2],"% variance"))
>  # adjust figure size and save it.
>  ggsave(p, filename = "PCA plot.pdf", width = 15, height = 12)
> ## Dispersion plot:
>  plotDispEsts(dds, main="Dispersion plot", cex=0.2)
>  legend("topright", legend="fitted dispersion",col="red", pch="-", cex=0.8)
> 
```

data	10 obs. of 5 variables
dds	Large DESeqDataSet (60675 elements, 13.9 Mb)
DESEq2_DEGs	9851 obs. of 16 variables
p	List of 9
phenodata	10 obs. of 3 variables
res	Large DESeqResults (6 elements, 7 Mb)
sig	9851 obs. of 16 variables
vst	Large DESeqTransform (60675 elements, 14.4 Mb)
Values	
desc	Factor w/ 10 levels "lung3e","lung3f",..: 4 3 5…

Your variables

Visualisations, files, packages…

First…

- **Clean & format the data (examples given)**
- **(Install &) open the needed packages**

```
library(DESeq2)
```

- **Set the working directory to the folder where the data is**
- **Import the data**

```
data <-read.table(“my_data_table.tsv”)
```

Data analysis workflow

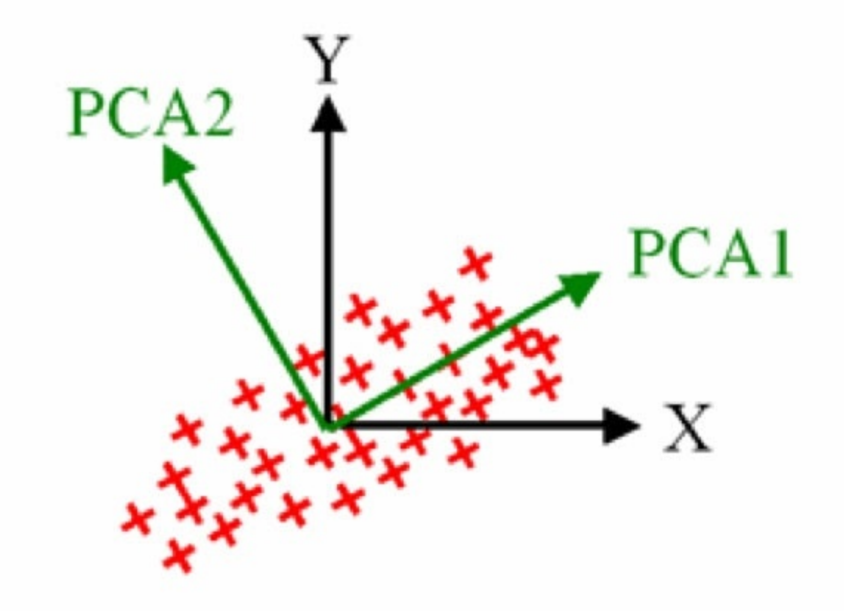
- Quality control of raw reads
- Preprocessing (trimming / filtering) if needed
- Alignment to reference genome
- Alignment level quality control
- Quantitation
- **Experiment level quality control**
- Differential expression analysis
- Visualization of reads and results in genomic context

Experiment level quality control

- **Getting an overview of similarities and dissimilarities between samples allows you to check**
  - Do the experimental groups separate from each other?
  - Is there a confounding factor (e.g. batch effect) that should be taken into account in the statistical analysis?
  - Are there sample outliers that should be removed?
- **To check this, we use PCA plot**

Dimension reduction / PCA

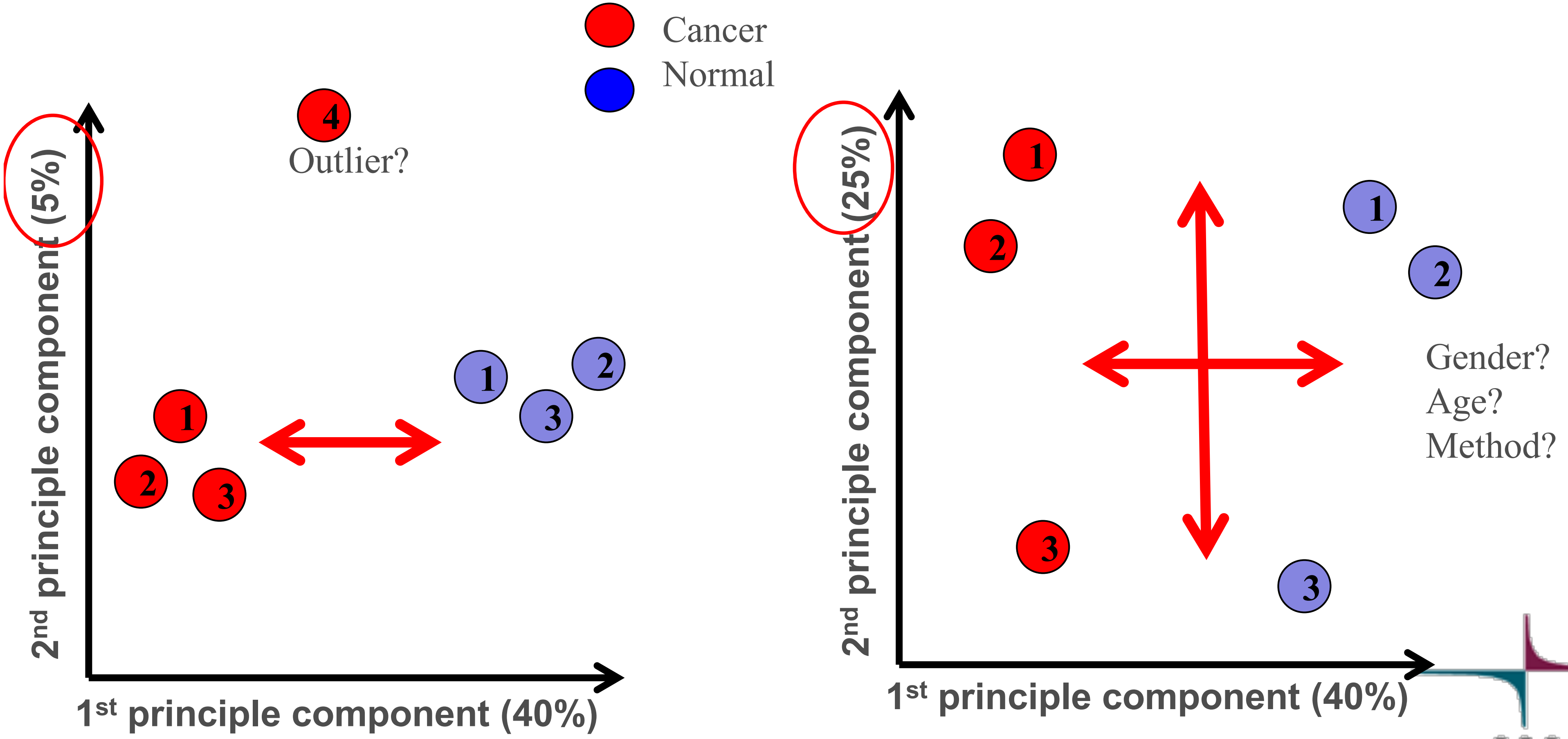
- **PCA = Principal Component Analysis**
- **finds the principal components of data**
- **PCs**
  - = the directions where there is the most variance
  - = the directions where the data is most spread out
- **Why we use PCA here?**
  - By reducing the dimensions the data can be visualized



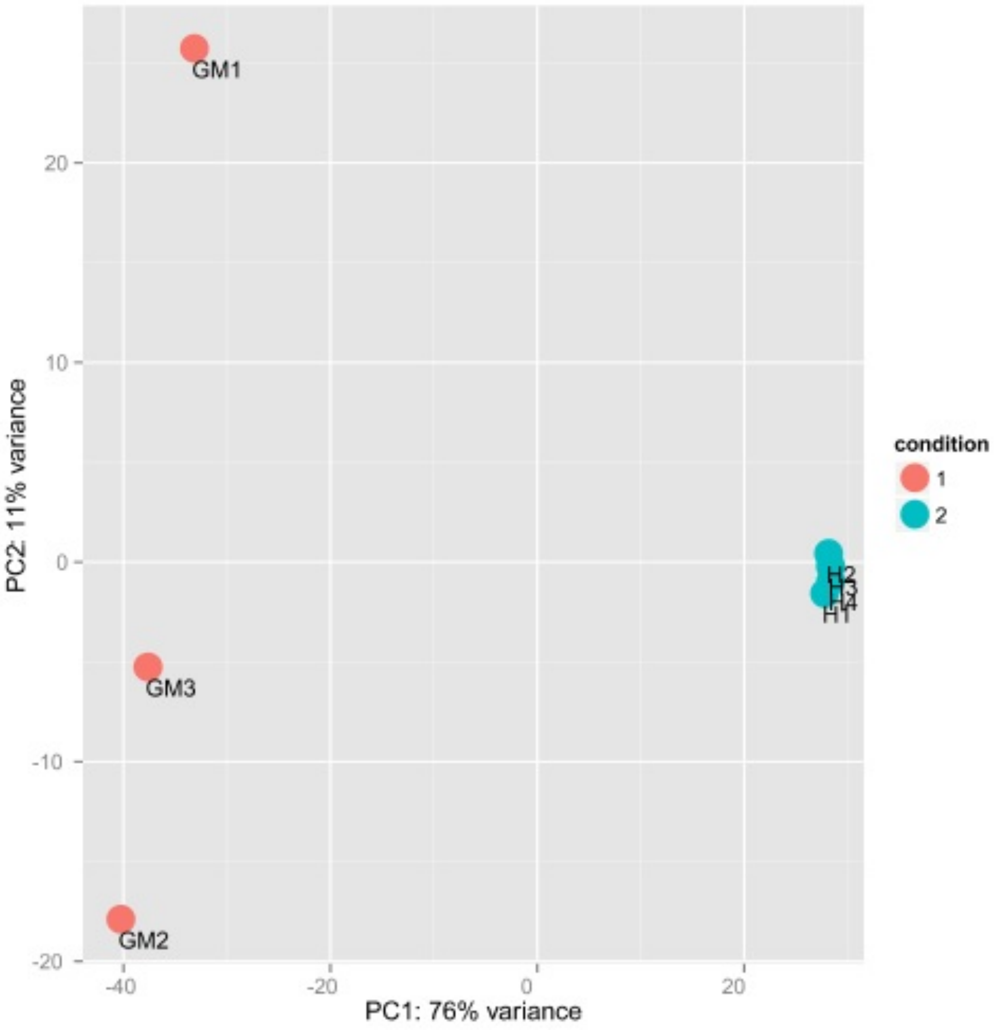
In our case instead of X and Y:
gene1, gene2, gene3… gene1838
= MANY dimensions!

What are we looking for?

- **Do the experimental groups separate from each other?**
- **Is there a confounding factor (e.g. batch effect) ?**
  - If the 2nd component explains only little variance, it can ignored
- **Outliers?**



PCA plot by DESeq2

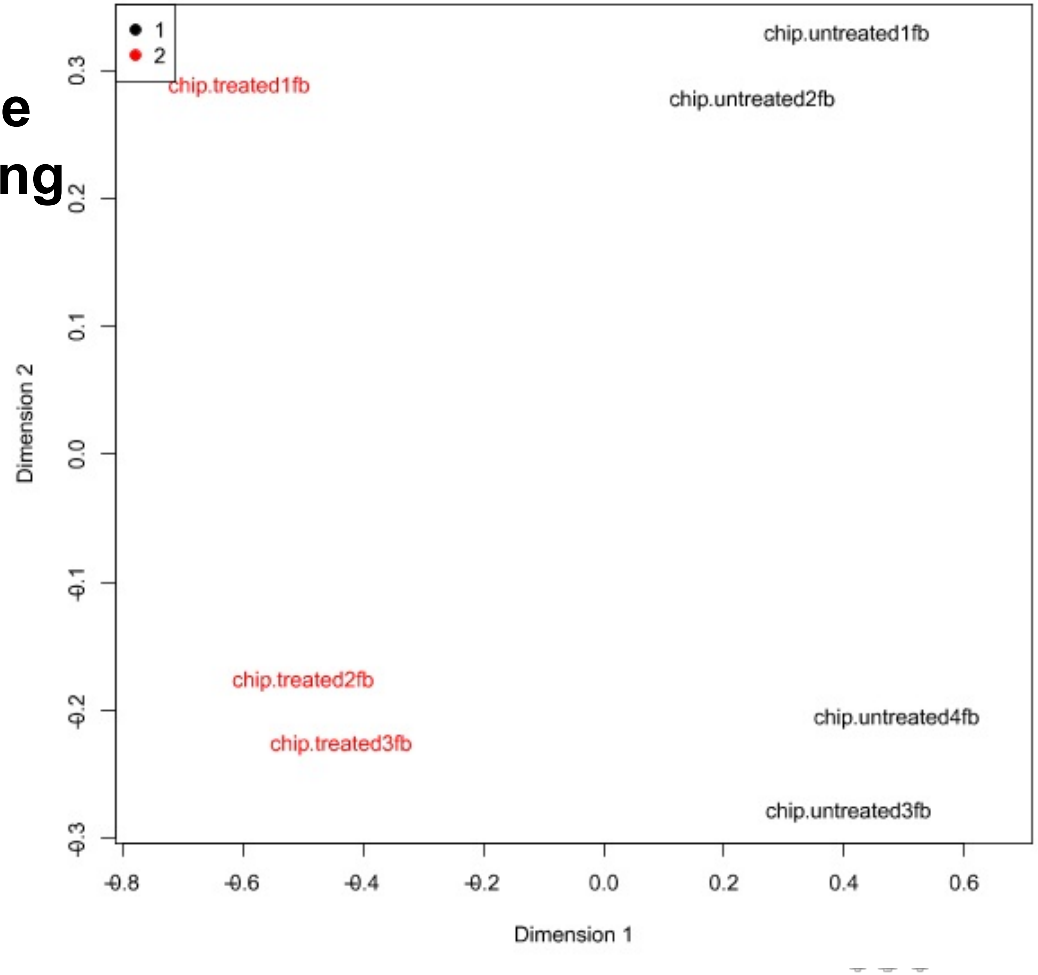


- **The first two principal components, calculated after variance stabilizing transformation**
- **Indicates the proportion of variance explained by each component**
  - If PC2 explains only a small percentage of variance, it can be ignored

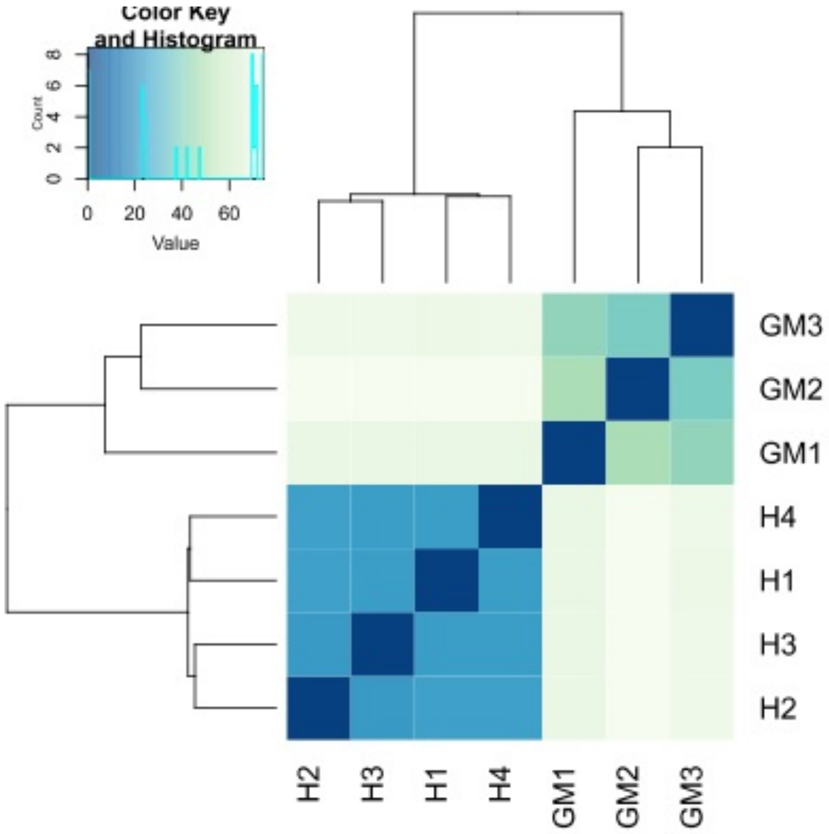
MDS plot by edgeR

- Distances correspond to the logFC or biological coefficient of variation (BCV) between each pair of samples

- Calculated using 500 most heterogenous genes (that have largest dispersion when treating all samples as one group)



Sample heatmap by DESeq2



- **Euclidean distances between the samples, calculated after variance stabilizing transformation**

Data analysis workflow

- Quality control of raw reads
- Preprocessing (trimming / filtering) if needed
- Alignment to reference genome
- Alignment level quality control
- Quantitation
- Experiment level quality control
- **Differential expression analysis**
- Visualization of reads and results in genomic context

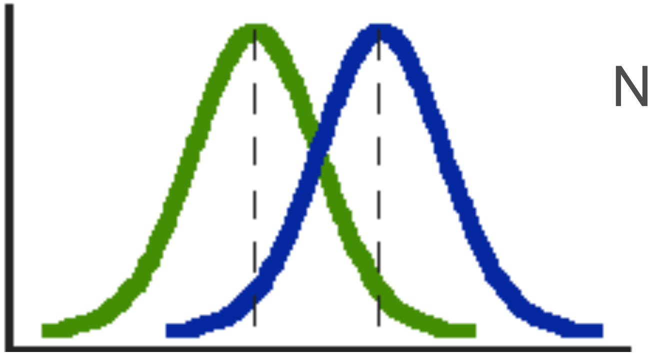
Differential expression analysis in RNA-seq

Differential expression analysis in RNA-seq

➢ **How differently is <u>gene A</u> expressed in <u>treatment</u> vs <u>controls</u>?**

	Control 1	Control 2	Control 3	Treatment 1	Treatment 2	Treatment 3
Gene A	**6**	**5**	**7**	**17**	**10**	**11**
Gene B	**11**	**11**	**10**	**3**	**4**	**2**
Gene C	**200**	**150**	**355**	**50**	**1**	**3**
Gene D	**0**	**1**	**0**	**2**	**0**	**1**

- Basic t-statistic: how far from each other are the means of the two groups? (In terms of deviance/variation/dispersion)



Not quite as simple as that…

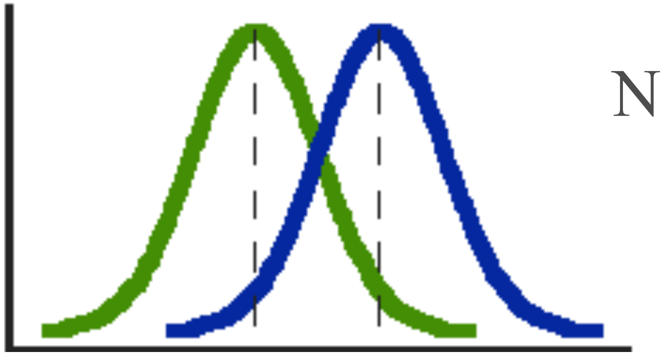
Differential expression analysis in RNA-seq

- **How differently is gene C expressed in treatment vs controls?**

	T1	T2	T3	C1	C2	C3
Gene A	1	0	1	6	7	5
Gene B	9	8	10	2	4	2
Gene C	10	12	11	7	20	16

???

- Basic t-statistic: how far from each other are the means of the two groups? (In terms of deviance/variation/dispersion)



Not quite as simple as that…

Software packages for DE analysis

- **edgeR**
- **DESeq2**
- Sleuth
- DRIMSeq
- DEXSeq
- Cuffdiff, Ballgown
- Limma + voom, limma + vst
- ...

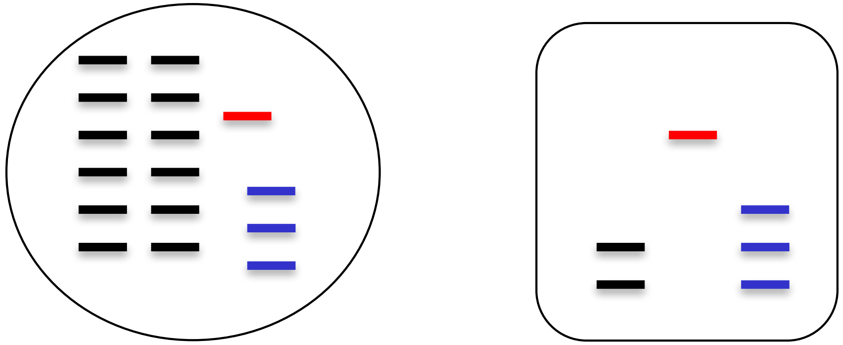
Differential gene expression analysis

- **Normalization**
- **Dispersion estimation**
- **Log fold change estimation**
- **Statistical testing**
- **Filtering**
- **Multiple testing correction**

Differential expression analysis:
Normalization

Normalization

- **For comparing gene expression between (groups of) samples, normalize for**
  - Library size (number of reads obtained)
  - RNA composition effect



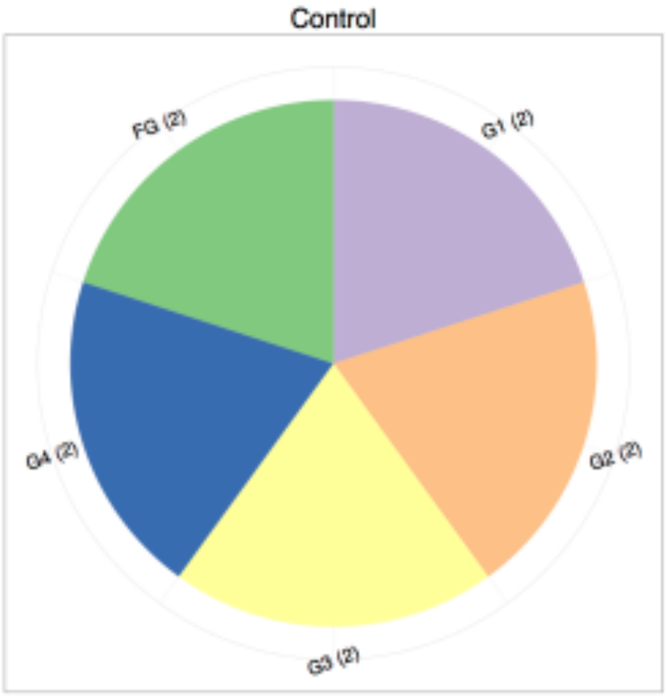
- **The number of reads for a gene is also affected by transcript length and GC content**
  - When studying differential gene expression we *assume that they stay the same*

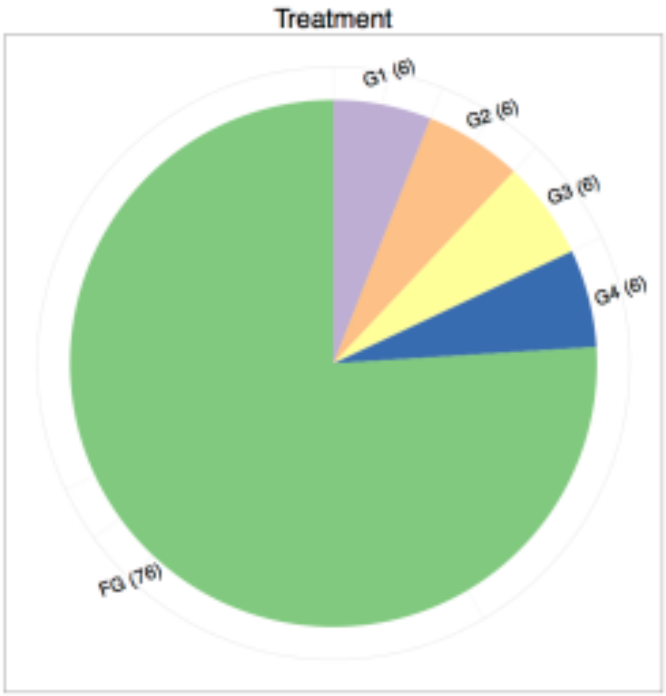
Metrics for quantifying gene expression levels

- **RPKM**
  - **R**eads **P**er **K**ilobase per **M**illion mapped reads
  - Normalize relative to sequencing depth and gene length
- **FPKM**
  - Similar to RPKM but count **DNA fragments** instead of reads
  - Used in paired end RNA-Seq experiments to avoid bias
- **TPM**
  - **T**ranscripts **P**er **M**illion
  - Normalize for gene length, then normalize by sequencing depth

Wagner GP *et al.* Measurement of mRNA abundance using RNA-Seq data: RPKM measure is inconsistent among samples. Theory Biosci. 2012 Dec;131(4):281-5.

Consideration for the between sample normalisation



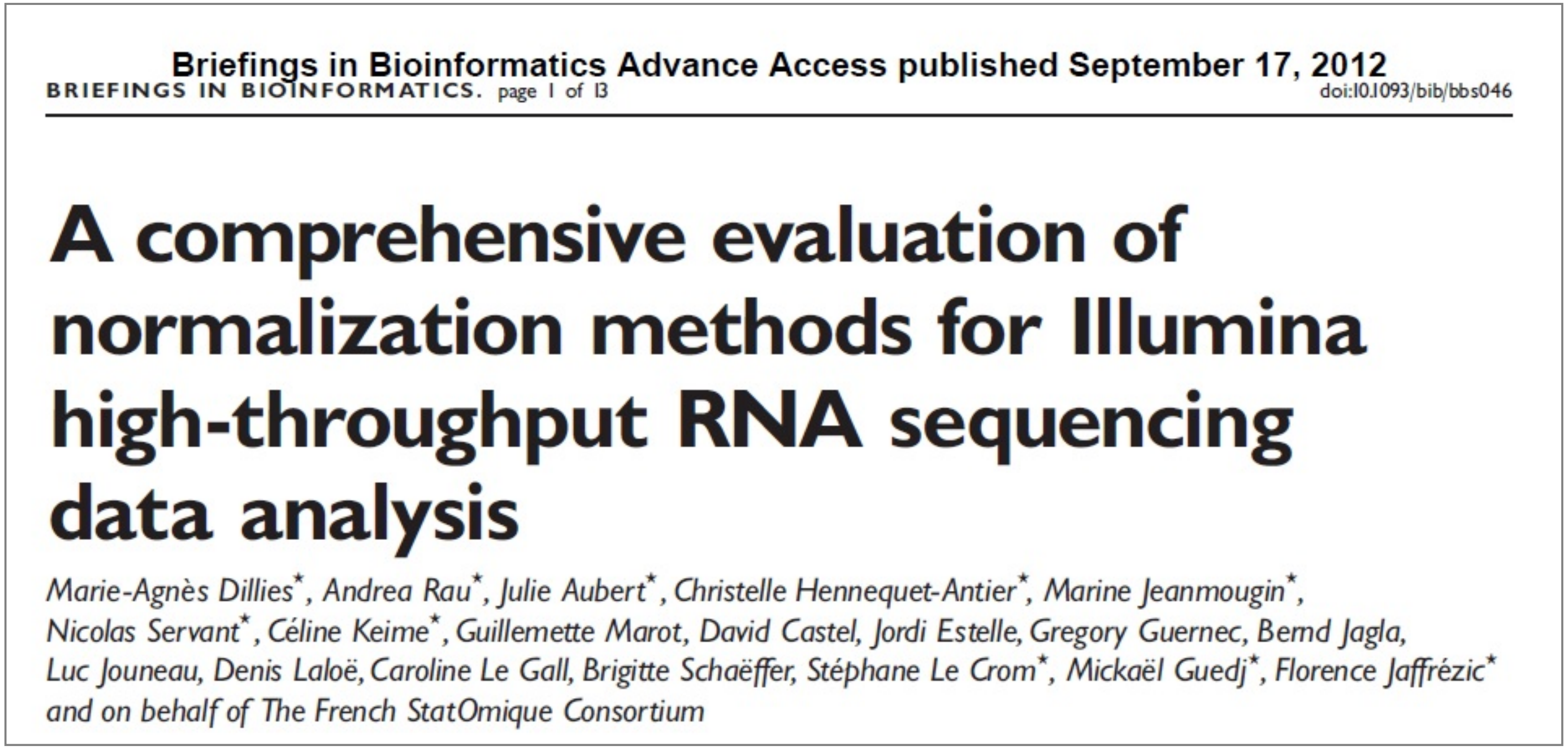


Gene	Control Counts	Treatment Counts	Control Normalized	Treatment Normalized
G1	2.00	6.00	0.25	0.25
G2	2.00	6.00	0.25	0.25
G3	2.00	6.00	0.25	0.25
G4	2.00	6.00	0.25	0.25
FG	2.00	76.00.00	0.25	3.17

(-FG) →

Gene	Control Counts	Treatment Counts	Control Normalized	Treatment Normalized
G1	2.00	6.00	0.20	0.06
G2	2.00	6.00	0.20	0.06
G3	2.00	6.00	0.20	0.06
G4	2.00	6.00	0.20	0.06
FG	2.00	76. 00	0.20	0.76

https://haroldpimentel.wordpress.com/2014/12/08/in-rna-seq-2-2-between-sample-normalization/



Briefings in Bioinformatics Advance Access published September 17, 2012

BRIEFINGS IN BIOINFORMATICS. page 1 of 13 doi:10.1093/bib/bbs046

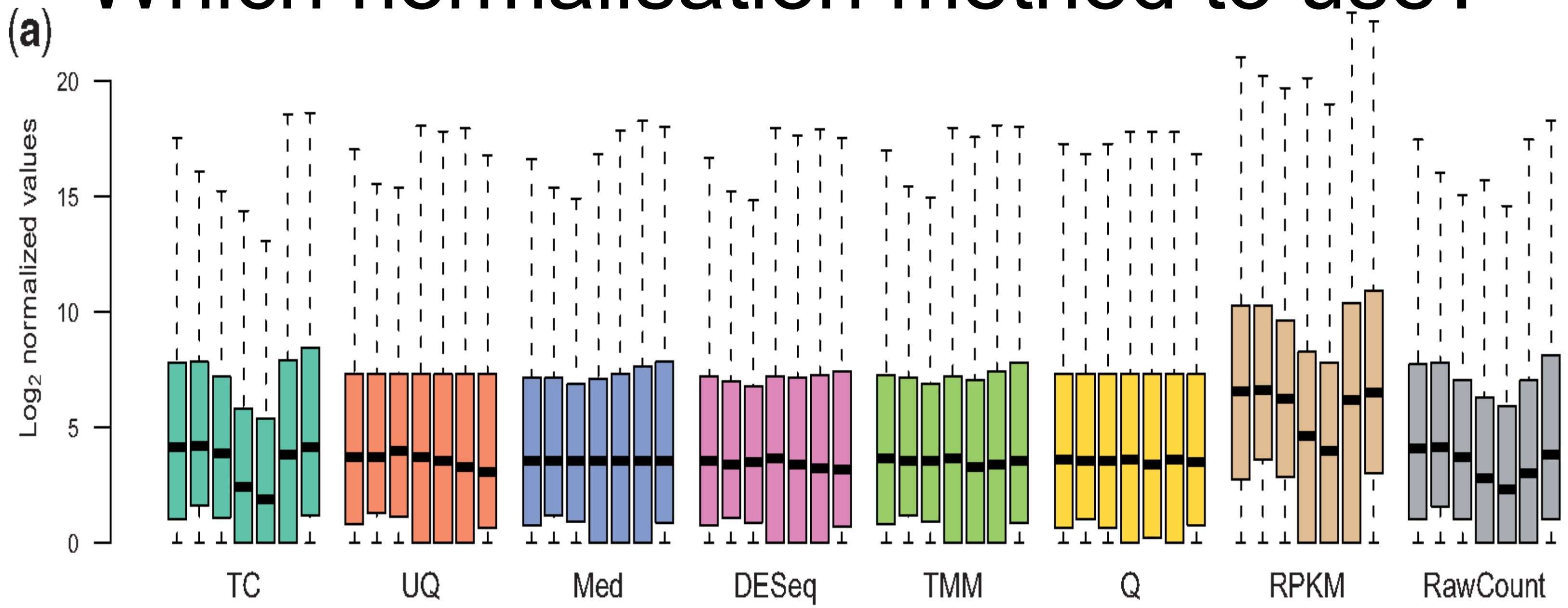
A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis

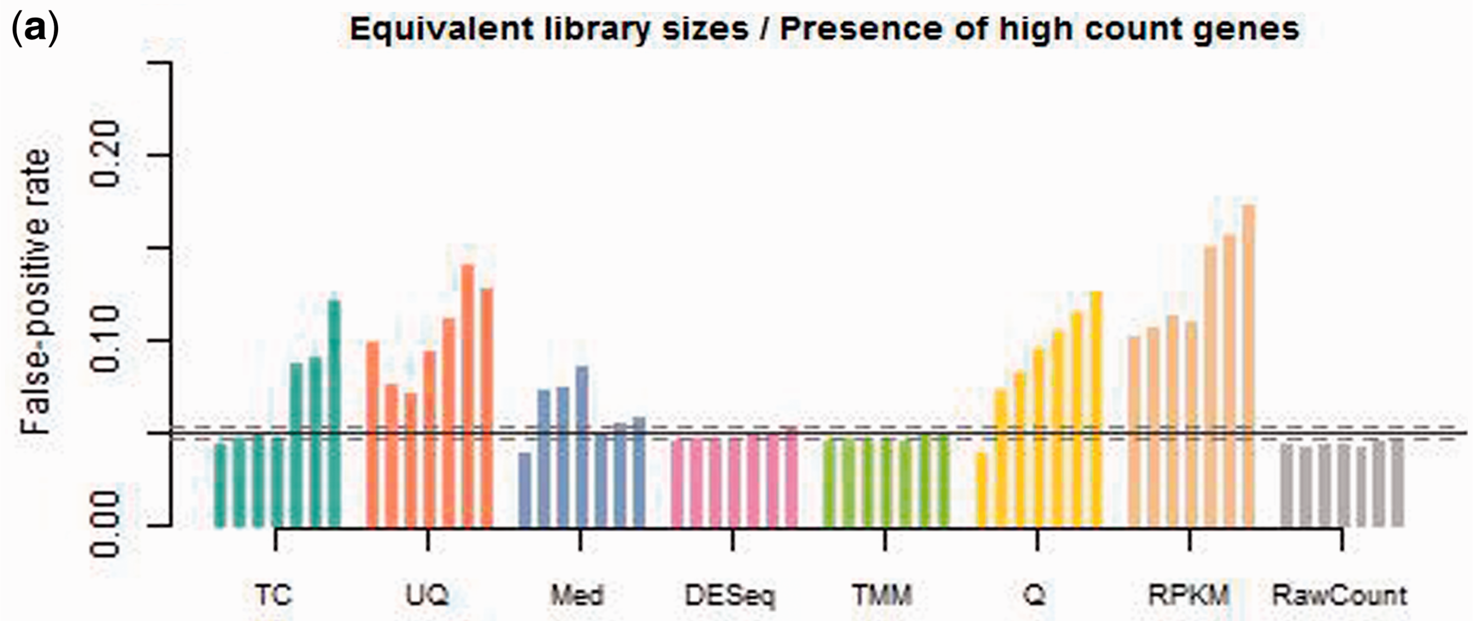
Marie-Agnès Dillies, Andrea Rau*, Julie Aubert*, Christelle Hennequet-Antier*, Marine Jeanmougin*, Nicolas Servant*, Céline Keime*, Guillemette Marot, David Castel, Jordi Estelle, Gregory Guernec, Bernd Jagla, Luc Jouneau, Denis Laloë, Caroline Le Gall, Brigitte Schaëffer, Stéphane Le Crom*, Mickaël Guedj*, Florence Jaffrézic* and on behalf of The French StatOmique Consortium*

- **“FPKM and TC are ineffective and should be definitely abandoned in the context of differential analysis”**

- **“In the presence of high count genes, only DESeq and TMM (edgeR) are able to maintain a reasonable false positive rate without any loss of power”**

Which normalisation method to use?





Dillies et. Al **A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis,** *Briefings in Bioinformatics*, Volume 14, Issue 6, November 2013, Pages 671–683, https://doi.org/10.1093/bib/bbs046

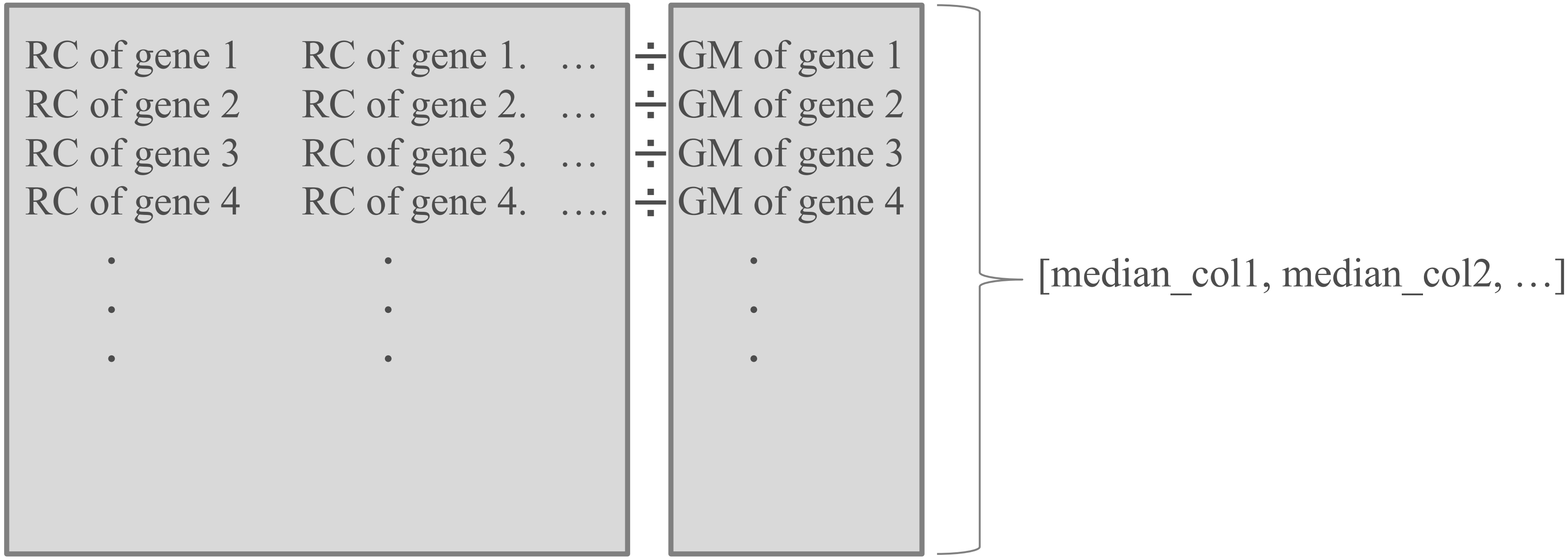
Normalization by edgeR and DESeq

- **Aim to make normalized counts for non-differentially expressed genes similar between samples**
  - Do not aim to adjust count distributions between samples
- **Assume that**
  - Most genes are not differentially expressed
  - Differentially expressed genes are divided equally between up- and down-regulation
- **Do not transform data, but use normalization factors within statistical testing**

Normalization by edgeR and DESeq – how?

- **DESeq(2)**
  - Take geometric mean of gene's counts across all samples
  - Divide gene's counts in a sample by the geometric mean
  - Take median of these ratios → sample's normalization factor (applied to read counts)
- **edgeR**
  - Select as reference the sample whose upper quartile is closest to the mean upper quartile
  - Log ratio of gene's counts in sample vs reference → M value
  - Take weighted trimmed mean of M-values (TMM) → normalization factor (applied to library sizes)
    - Trim: Exclude genes with high counts or large differences in expression
    - Weights are from the delta method on binomial data

Library size factor estimation in DESeq2



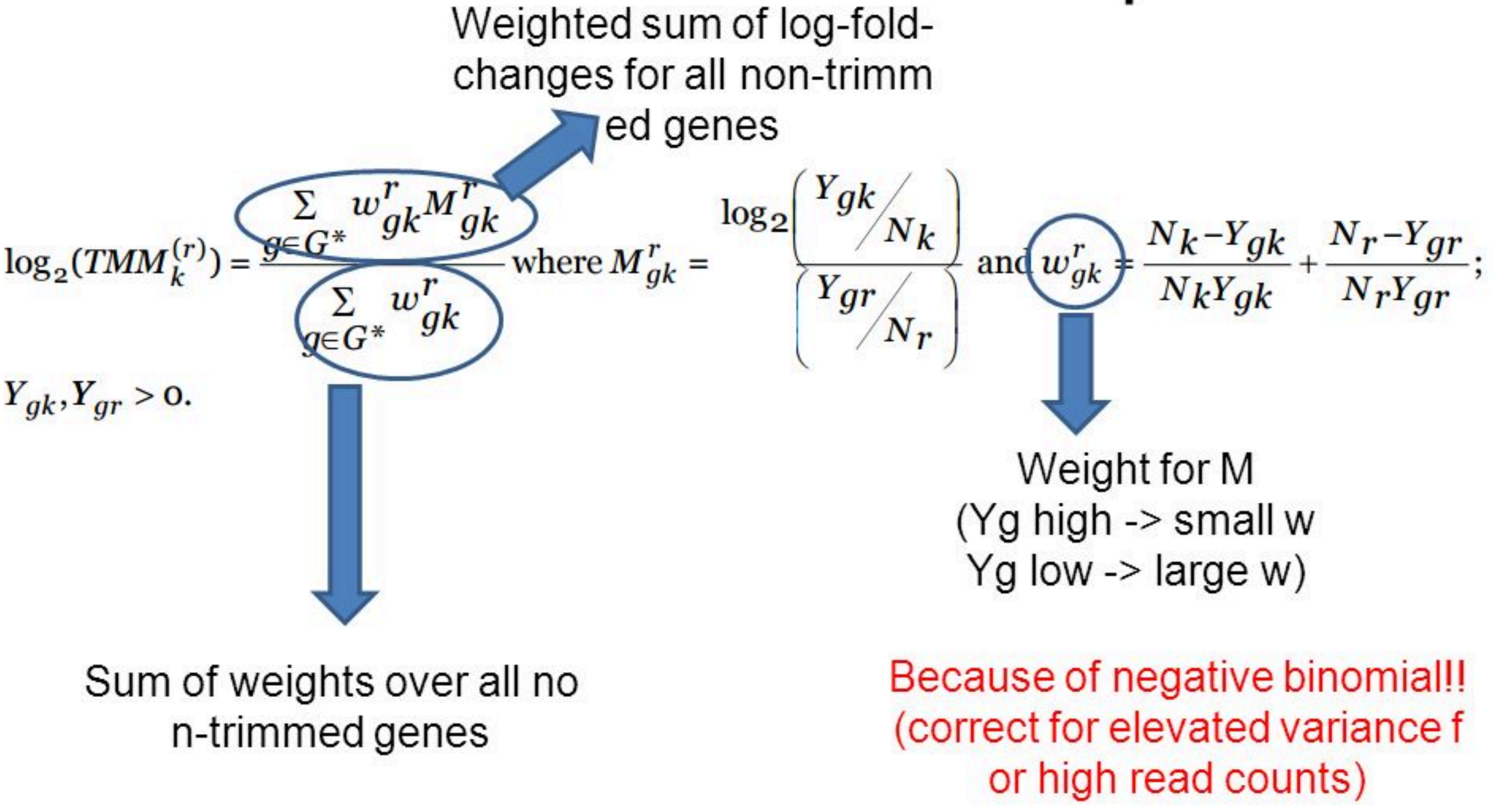
RC of gene 1 RC of gene 1. … ÷ GM of gene 1
RC of gene 2 RC of gene 2. … ÷ GM of gene 2
RC of gene 3 RC of gene 3. … ÷ GM of gene 3
RC of gene 4 RC of gene 4. …. ÷ GM of gene 4
. . .
. . .
. . .

[median_col1, median_col2, …]

Median value of each sample serves as scaling factor for that sample

Geometric mean (GM) is across all samples of respective gene read counts (RC) and then median value is obtained for each sample across ratios of all genes

TMM (trimmed mean of M values) normalization for RNA-seq data



Weighted sum of log-fold-changes for all non-trimmed genes

$$\log_2(TMM_k^{(r)}) = \frac{\sum_{g\in G^*} w_{gk}^r M_{gk}^r}{\sum_{g\in G^*} w_{gk}^r} \text{ where } M_{gk}^r = \log_2\frac{\left(Y_{gk}/N_k\right)}{\left(Y_{gr}/N_r\right)} \text{ and } w_{gk}^r = \frac{N_k - Y_{gk}}{N_k Y_{gk}} + \frac{N_r - Y_{gr}}{N_r Y_{gr}};$$

$$Y_{gk}, Y_{gr} > 0.$$

Sum of weights over all non-trimmed genes

Weight for M
(Yg high -> small w
Yg low -> large w)

Because of negative binomial!!
(correct for elevated variance for high read counts)

Robinson & Oshlack 2010

Source: https://slideplayer.com/slide/7647573/

edgeR and DESeq2 expect raw read counts

- **Raw counts are needed to assess the quantification uncertainty**
- **Uncertainty information is lost if counts are transformed to FPKM**
  - FPKM = fragments per kilobase per million mapped reads.
  - Normalizes for gene length and library size. Example:
    - 20 kb transcript has 400 counts, library size is 20 million reads: FPKM = (400/20) / 20
    - 0.5 kb transcript has 10 counts, library size is 20 million reads: FPKM = (10/0.5) / 20
    - → in both cases FPKM =1, but it is less likely to get 400 reads just by chance

- **The negative binomial assumption of edgeR and DESeq2 is flexible enough to deal with gene-level counts summarized from Salmon's transcript-level abundance estimates**

Differential expression analysis:

Dispersion estimation

Dispersion

- **When comparing gene's expression levels between groups, it is important to know also its within-group variability**

- **Dispersion = (BCV)^2**
  - BCV = gene's biological coefficient of variation
  - E.g. if gene's expression typically differs from replicate to replicate by 20% (so BCV = 0.2), then this gene's dispersion is 0.2^2 = 0.04

- **Note that the variability seen in counts is a sum of 2 things:**
  - Sample-to-sample variation (dispersion)
  - Uncertainty in measuring expression by counting reads

How to estimate dispersion reliably?

- **We cannot typically afford tens or hundreds of biological replicates**

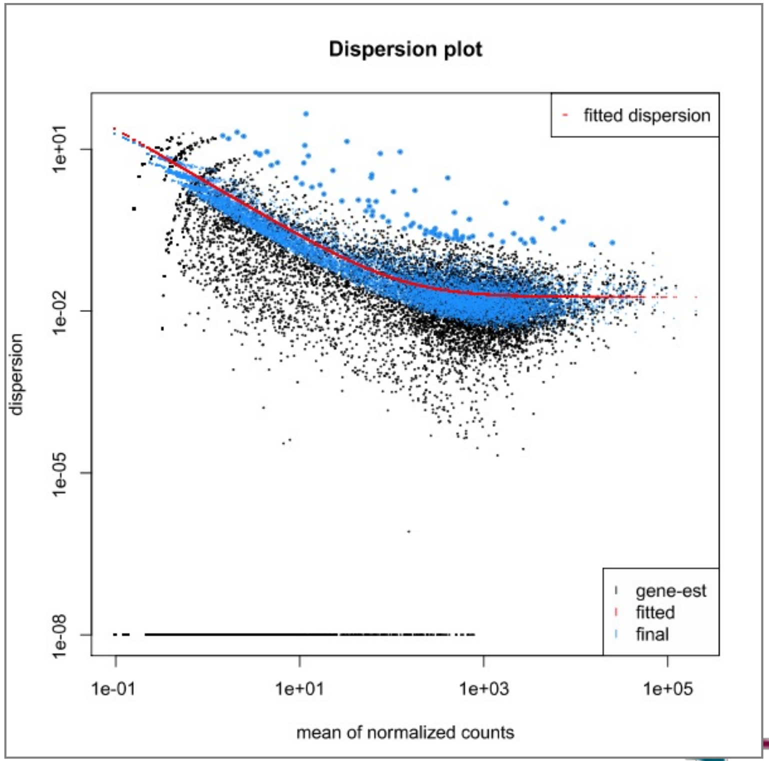
  → it is difficult to estimate within-group variability

- **Solution: pool information across genes which are expressed at similar level**
  - assumes that genes of similar average expression strength have similar dispersion

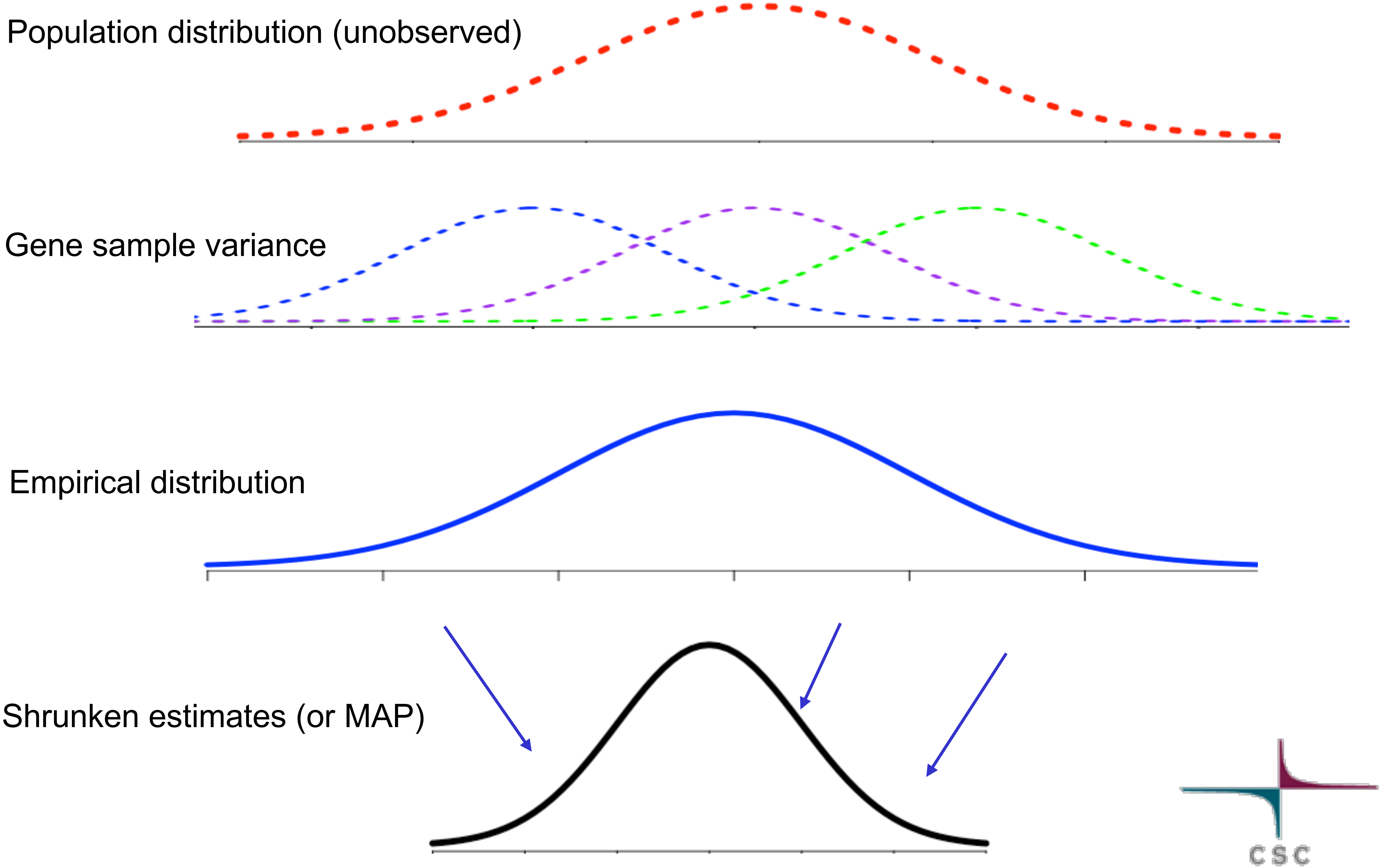
- **Different approaches**
  - edgeR
  - DESeq2

Dispersion estimation by DESeq2

- **Estimates genewise dispersions using maximum likelihood**
- **Fits a curve to capture the dependence of these estimates on the average expression strength**
- **Shrinks genewise values towards the curve using an empirical Bayes approach**
  - The amount of shrinkage depends on several things including sample size
  - Genes with high gene-wise dispersion estimates are dispersion outliers (blue circles above the cloud) and they are not shrunk



Shrinkage estimation



Population distribution (unobserved)

Gene sample variance

Empirical distribution

Shrunken estimates (or MAP)

Differential expression analysis:

Statistical testing

Generalized linear models

- **Model the expression of each gene as a linear combination of explanatory factors (eg. group, time, patient)**
  - y = a + (b · group) + (c · time) + (d · patient) + e

    y = gene's expression

    a, b, c and d = parameters estimated from the data

    a = intercept (expression when factors are at reference level)

    e = error term

- **Generalized linear model (GLM) allows the expression value distribution to be different from normal distribution**
  - Negative binomial distribution used for count data

Example of DESeq2 design matrix

Sex	Experiment
M	Treatment
M	Control
F	Treatment
M	Control
F	Treatment
F	Control

DESeq2 design = ~ sex+ experiment

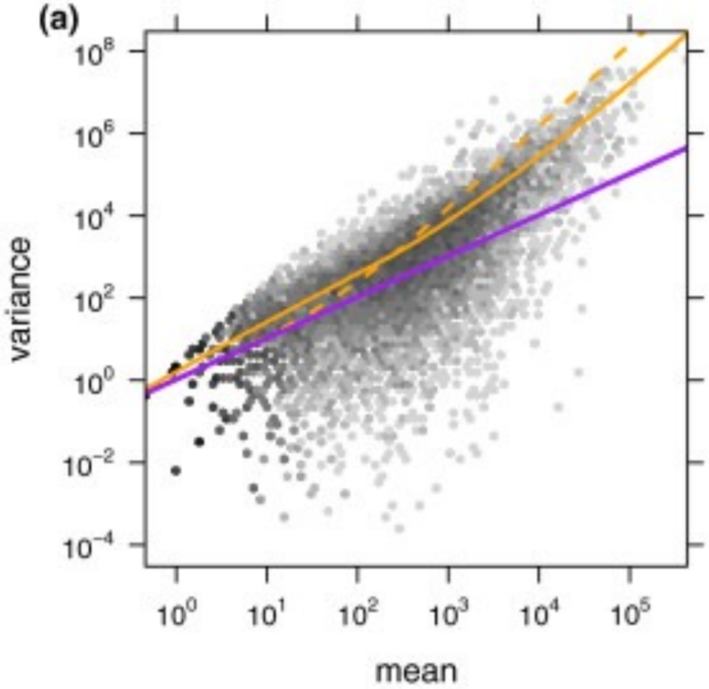
$$\begin{matrix}\log_2\mu_1\\ \log_2\mu_2\\ \log_2\mu_3\\ \log_2\mu_4\\ \log_2\mu_5\\ \log_2\mu_6\end{matrix} = \begin{pmatrix}1&1&1\\ 1&1&0\\ 1&0&1\\ 1&1&0\\ 1&0&1\\ 1&0&0\end{pmatrix}\begin{pmatrix}\beta_{\text{intercept}}\\ \beta_{\text{sex\_female\_vs\_male}}\\ \beta_{\text{experiment\_Ctr\_vs\_treat}}\end{pmatrix}$$

Statistical Distributions

gaussian, poisson, **negative binomial** -- what does all this mean?

- RNA-seq data fits a Negative Binomial (NB) distribution.
- But really, that's just saying that RNAseq looks like "counts" data with more variation than just statistical fluctuations– it also has biological variation in it.

- **How do we know?** Because, when you measure variance (per gene, between replicates), it's not equal to the mean, and it's not even a good linear fit



poisson

negative binomial fit

* For real data, even the NB fit isn't always great

M. Hammell

Anders & Huber, Genome Biology 2010

Statistical testing

- **edgeR**
  - Two group comparisons
    - Exact test for negative binomial distribution.
  - Multifactor experiments
    - Generalized linear model, likelyhood ratio test.
- **DESeq2**
  - Shrinks log fold change estimates toward zero using an empirical Bayes method
    - Shrinkage is stronger when counts are low, dispersion is high, or there are only a few samples
  - Generalized linear model, Wald test for significance
    - Shrunken estimate of log fold change is divided by its standard error and the resulting z statistic is compared to a standard normal distribution

Multiple testing correction

- **We tests thousands of genes, so it is possible that some genes get good p-values just by chance**
  - This problem is much bigger, if you test transcripts (DTE)

- **To control this problem of false positives, p-values need to be corrected for multiple testing**

- **Several methods are available, the most popular one is the Benjamini-Hochberg correction (BH)**

- **The adjusted p-value is FDR (false discovery rate)**

Filtering

- **Reduces the severity of multiple testing correction by removing some genes (makes n smaller)**

- **Filter out genes which have little chance of showing evidence for significant differential expression**
  - genes which are not expressed
  - genes which are expressed at very low level (low counts are unreliable)

- **Should be independent**
  - do not use information on what group the sample belongs to

- **DESeq2 selects filtering threshold automatically**

Summary of differential expression analysis

- **Means of the raw counts within groups can't be compared**
  - Because library size varies & the RNA composition effect messes things up

    → NORMALIZATION
  - Expression values don't follow normal distribution

    → GLM, NEGATIVE BINOMIAL DISTRIBUTION

- **Direct (gene-wise) estimation of dispersion is not good**
  - Because there are too few replicates

    → POOL DISPERSION INFORMATION ACROSS GENES

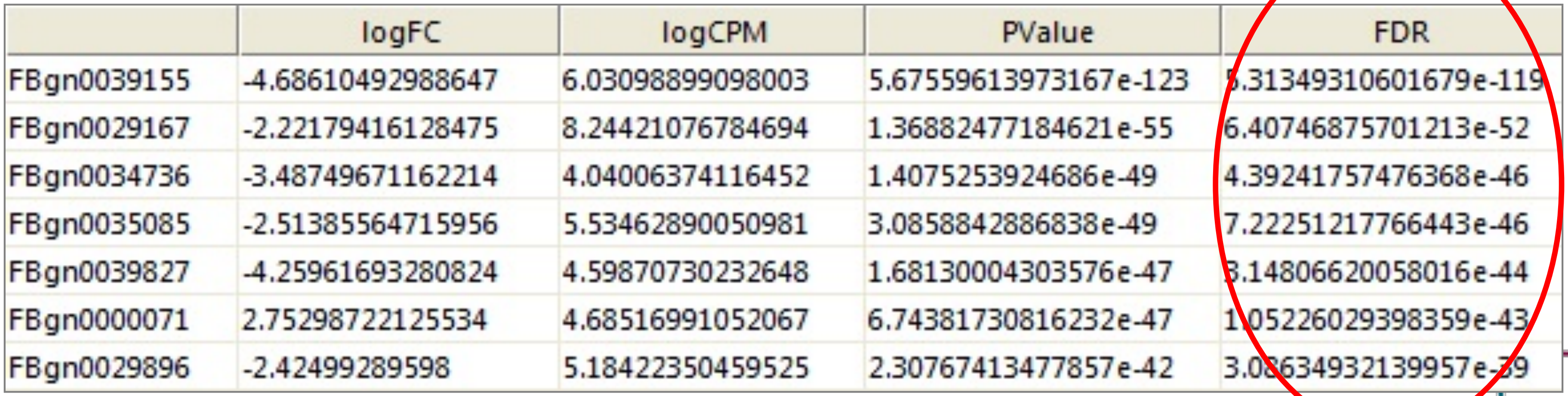
- **We are doing many comparisons**

    → MULTIPLE TESTING CORRECTION, FDR

Luckily the tools will take care of these things!

edgeR result table

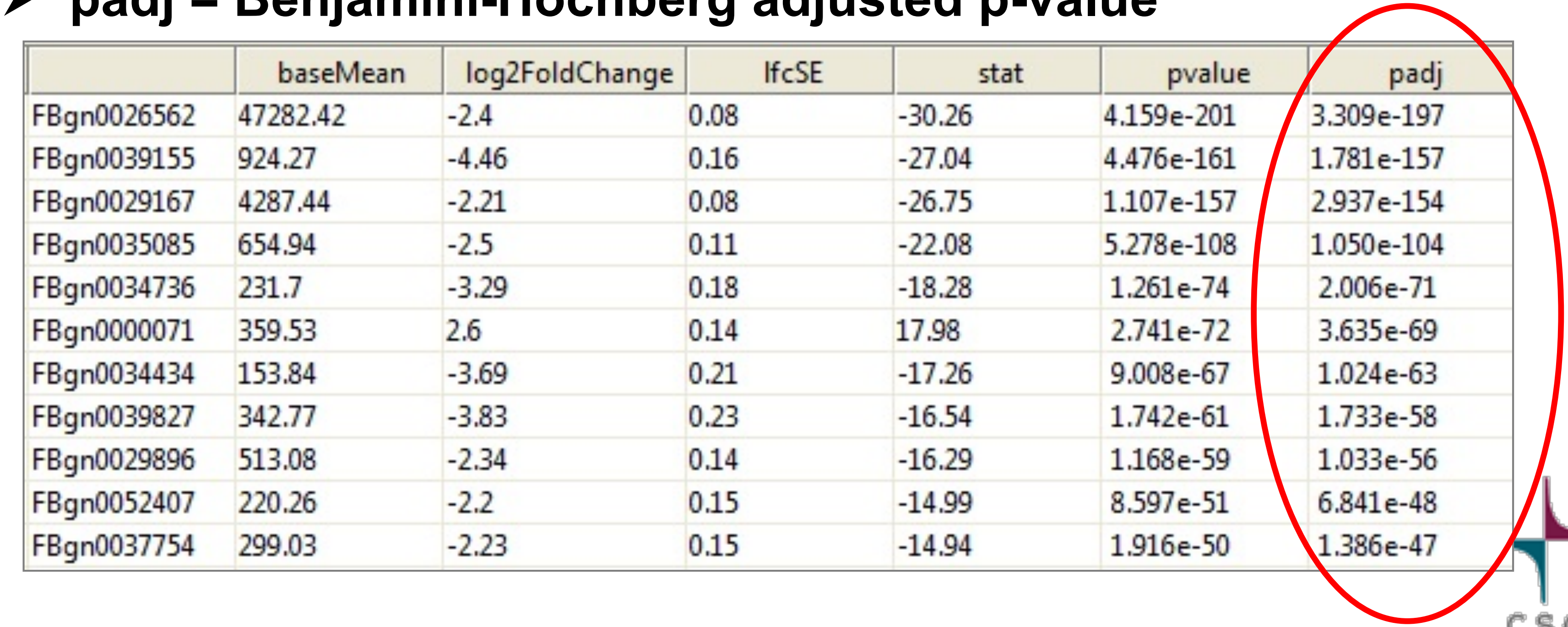
- **logFC = log2 fold change**
- **logCPM = average log2 counts per million**
- **Pvalue = raw p-value**
- **FDR = false discovery rate (Benjamini-Hochberg adjusted p-value)**



	logFC	logCPM	PValue	FDR
FBgn0039155	-4.68610492988647	6.03098899098003	5.67559613973167e-123	5.31349310601679e-119
FBgn0029167	-2.22179416128475	8.24421076784694	1.36882477184621e-55	6.40746875701213e-52
FBgn0034736	-3.48749671162214	4.04006374116452	1.4075253924686e-49	4.39241757476368e-46
FBgn0035085	-2.51385564715956	5.53462890050981	3.0858842886838e-49	7.22251217766443e-46
FBgn0039827	-4.25961693280824	4.59870730232648	1.68130004303576e-47	3.14806620058016e-44
FBgn0000071	2.75298722125534	4.68516991052067	6.74381730816232e-47	1.05226029398359e-43
FBgn0029896	-2.42499289598	5.18422350459525	2.30767413477857e-42	3.08634932139957e-39

DESeq2 result table

- baseMean = mean of counts (divided by size factors) taken over all samples
- **log2FoldChange = log2 of the ratio meanB/meanA**
- lfcSE = standard error of log2 fold change
- stat = Wald statistic
- pvalue = raw p-value
- **padj = Benjamini-Hochberg adjusted p-value**



	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
FBgn0026562	47282.42	-2.4	0.08	-30.26	4.159e-201	3.309e-197
FBgn0039155	924.27	-4.46	0.16	-27.04	4.476e-161	1.781e-157
FBgn0029167	4287.44	-2.21	0.08	-26.75	1.107e-157	2.937e-154
FBgn0035085	654.94	-2.5	0.11	-22.08	5.278e-108	1.050e-104
FBgn0034736	231.7	-3.29	0.18	-18.28	1.261e-74	2.006e-71
FBgn0000071	359.53	2.6	0.14	17.98	2.741e-72	3.635e-69
FBgn0034434	153.84	-3.69	0.21	-17.26	9.008e-67	1.024e-63
FBgn0039827	342.77	-3.83	0.23	-16.54	1.742e-61	1.733e-58
FBgn0029896	513.08	-2.34	0.14	-16.29	1.168e-59	1.033e-56
FBgn0052407	220.26	-2.2	0.15	-14.99	8.597e-51	6.841e-48
FBgn0037754	299.03	-2.23	0.15	-14.94	1.916e-50	1.386e-47

Analyzing differential gene expression: things to take into account

- **<u>Biological replicates are important!</u>**
- **Normalization is required in order to compare expression between samples**
  - Different library sizes
  - RNA composition bias caused by sampling approach
- **Raw counts are needed to assess measurement precision**
  - Counts are the ”the units of evidence” for expression
  - Gene-level counts summarized from Salmon’s transcript-level estimates seem to be ok
- **Multiple testing problem**

Annotations

- **We want to annotate our Ensembl identifiers with gene names + descriptions:**
  - ENSG00000122852 -> “SFTPA1”, “surfactant protein A1”
- **biomaRt tools allows to make queries to databases like Ensembl**
  1. Select database & dataset to use

```
ensembl <- useMart("ensembl",
dataset="hsapiens_gene_ensembl")
```

  2. Query:
     - attributes = what we retrieve
     - filters = restrictions for the query
     - values = values for the filter

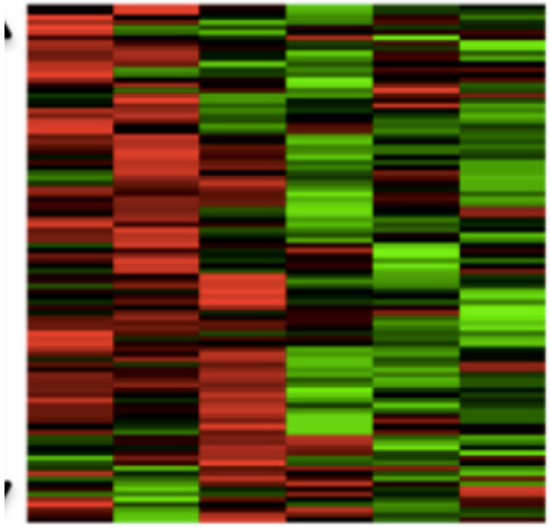
```
genes_ensembl_org <- getBM(attributes <-
c("ensembl_gene_id", "external_gene_name",
"description"), filters = "ensembl_gene_id", values =
genes, mart = ensembl)
```

  - List functions help to select: ListMarts, ListDatasets, ListAttributes…

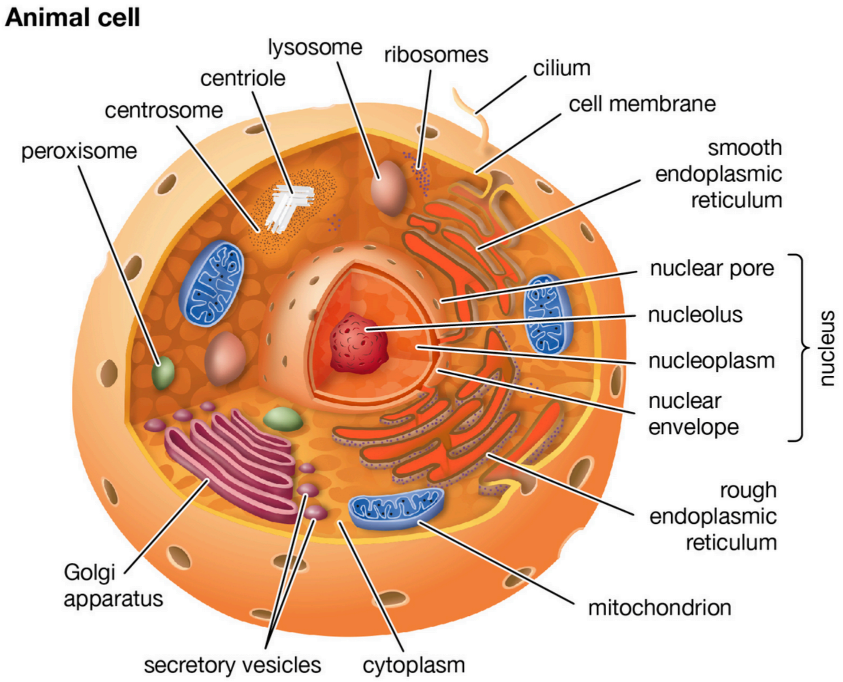
Enrichment Analysis analysis

A key challenge in omics' studies:
how to move from expression changes to biological functions

RNAseq data



?!

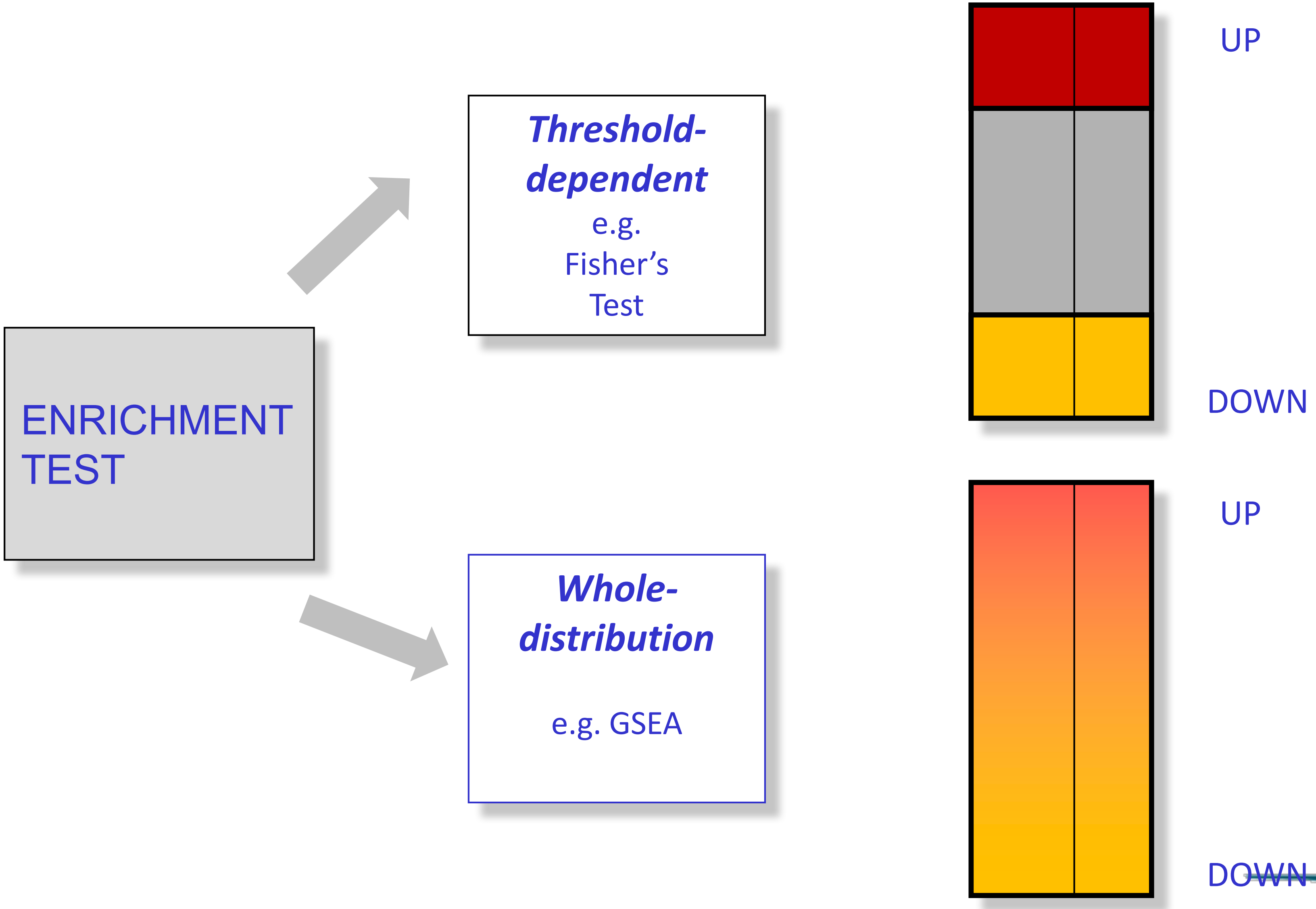


- ➢ **Are there any specific biological functions that are characterized by gene expression changes?**

Enrichment Analysis in General

- A genome-wide application tool
- Results are taken as indicative rather than conclusive
- Often done as secondary analysis to get high level understanding of biology
- Related names for this analysis: Functional interpretation analysis; GSEA, GSA, Pathway analysis

Mainly two types of enrichment analysis



ENRICHMENT TEST

Threshold-dependent

e.g. Fisher’s Test

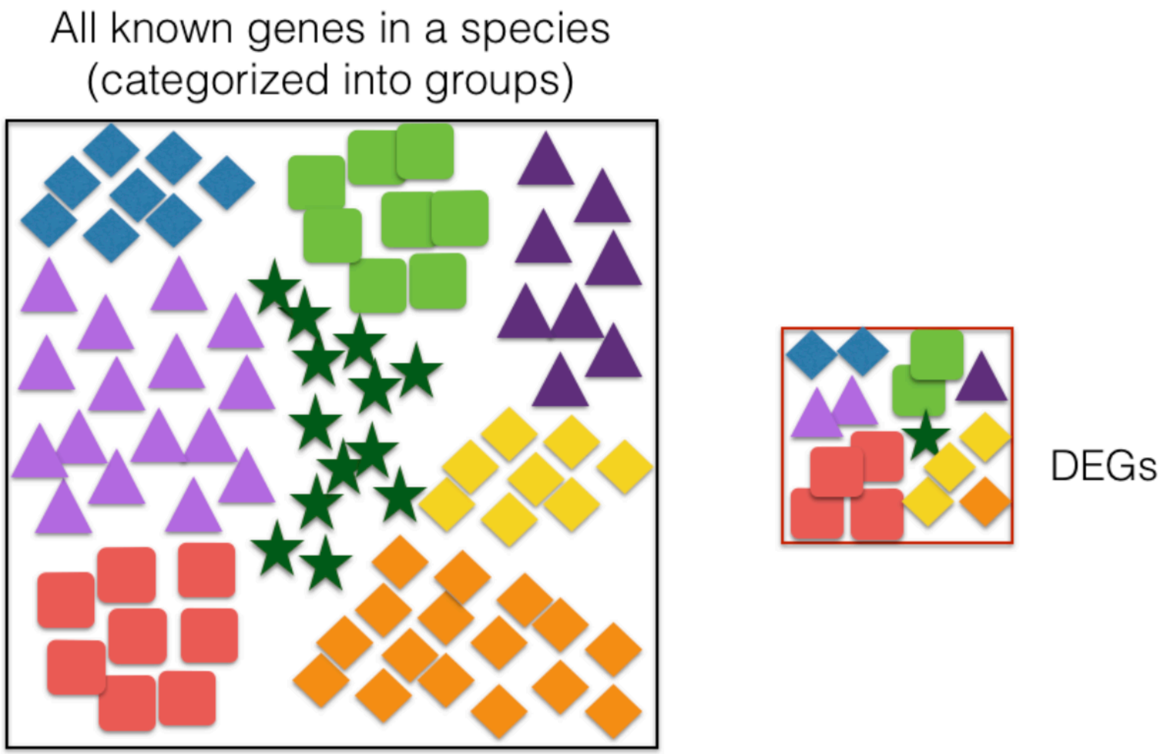
Whole-distribution

e.g. GSEA

Overrepresentation analysis
(= Threshold based analysis)

Enrichment Analysis: Introduction

- **Break down cellular function into gene sets**
  - Every set of genes is associated to a specific cellular function, process, component or pathway

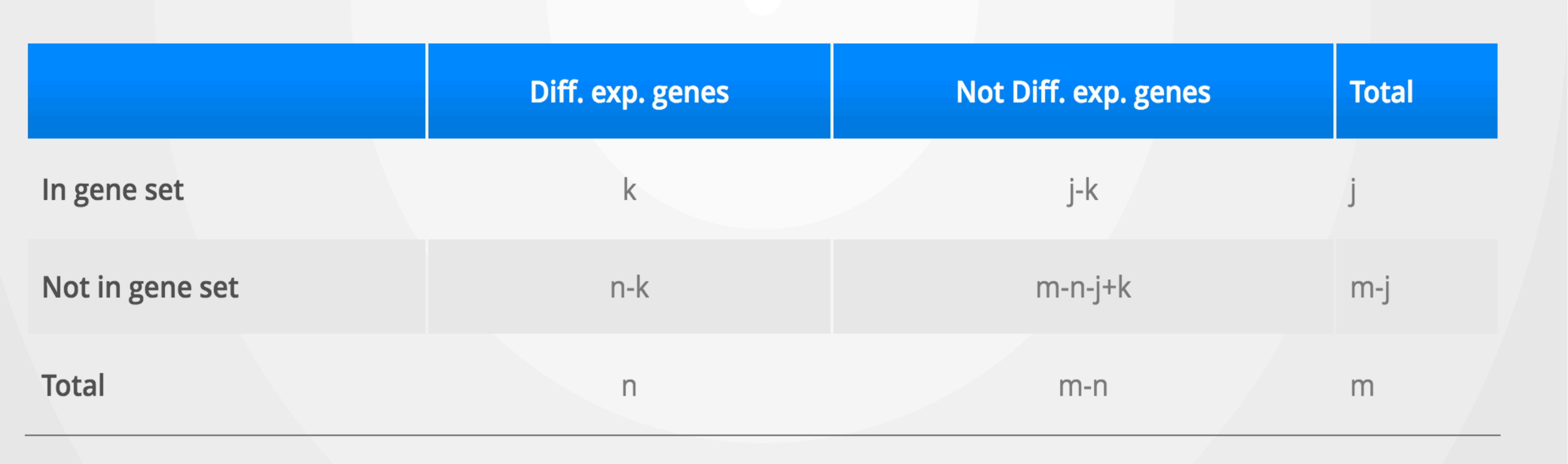


Gene set =
Predefined set of genes which are grouped by their gene function, pathway membership, etc.

Image source: https://github.com/hbctraining/DGE_workshop

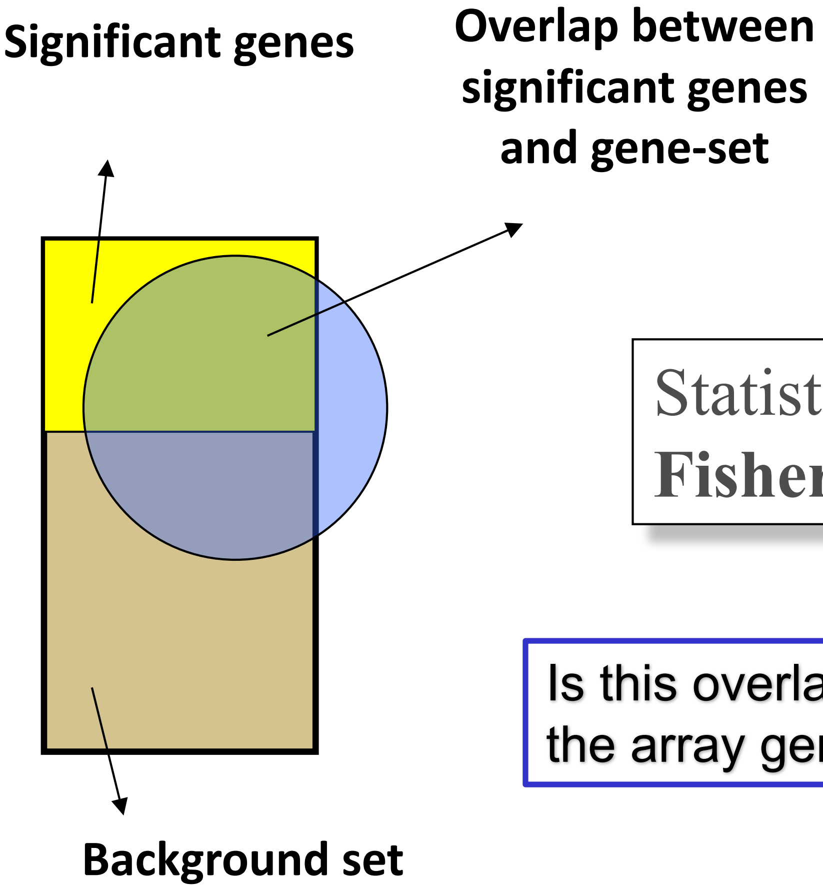
Hypergeometric testing of gene sets

- *m* is the total number of genes
- *j* is the number of genes are in the functional category
- *n* is the number of differentially expressed genes
- *k* is the number of differentially expressed genes in the category



	Diff. exp. genes	Not Diff. exp. genes	Total
In gene set	k	j-k	j
Not in gene set	n-k	m-n-j+k	m-j
Total	n	m-n	m

Enrichment Test



Statistical Model:
Fisher's Exact Test

Is this overlap larger than expected by random sampling the array genes?

How do we perform the gene set testing?

- **Find a set of differentially expressed genes (DEGs)**
- **Are *DEGs in a set* more common than *DEGs not in a set*?**
- **Fisher test stats::fisher.test()**
- **Conditional hypergeometric test, to account for directed hierachy of GO GOstats::hyperGTest()**

Fisher's exact test based methods are not optimal

- **The outcome of the overrepresentation test depends on the significance threshold used to declare genes differentially expressed**
  - cut-off is always somewhat arbitrary

- **Functional categories in which many genes exhibit small changes may go undetected.**

- **Genes are not independent, so a key assumption of the Fisher's exact tests is violated.**

- **Relative strength of DE changes is ignored**

Enrichment Analysis: Introduction

- **Main rationale – functionally related genes often display a coordinated expression to accomplish their roles in the cells**

- **Aims to identify gene sets even with "subtle but coordinated" expression changes that would be missed by DEGs threshold selection**

- **Gene Set Enrichment Analysis - Statistical methods determine significance of enrichment for gene set by comparing distribution of genes in set to 'background distribution'**

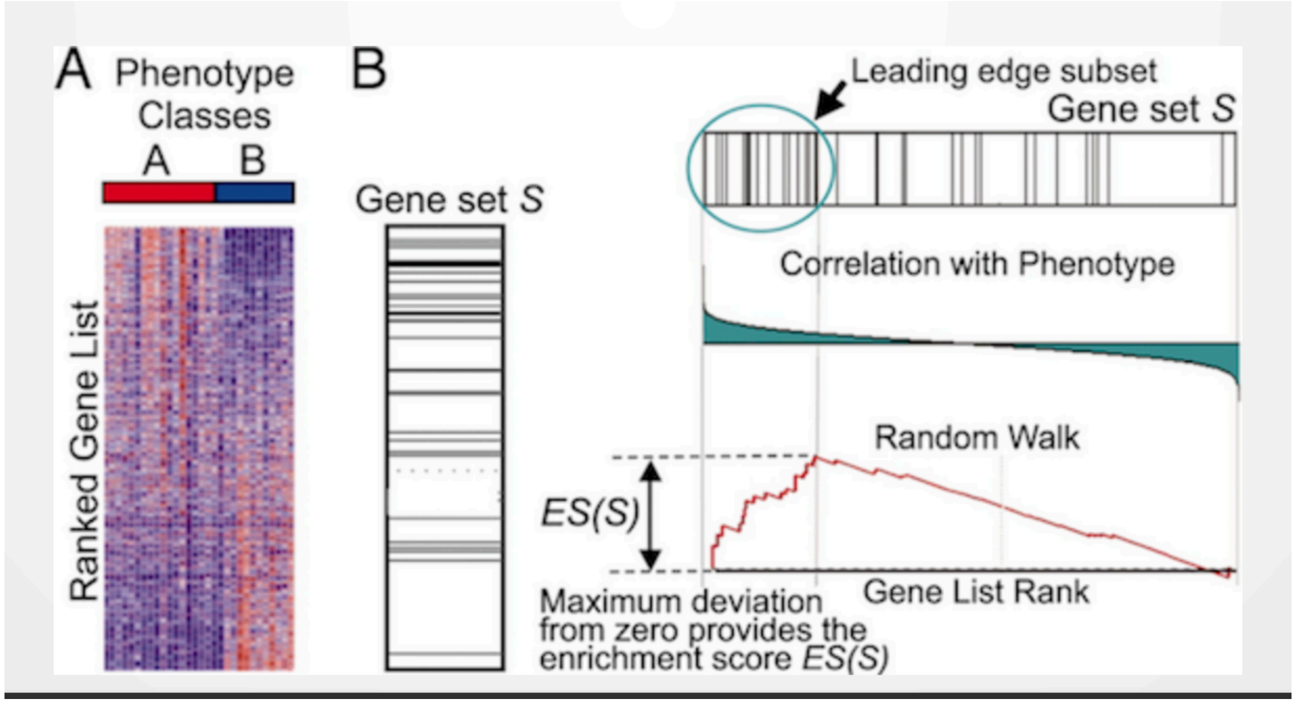
GSEA: Gene set enrichment analysis

- **The null hypothesis is that the rank ordering of the genes in a given comparison is random with regard to the case-control assignment.**

- **The alternative hypothesis is that the rank ordering of genes sharing functional/pathway membership is associated with the case-control assignment.**

GSEA: Gene set enrichment analysis

1. **Sort genes by log fold change**
2. **Calculate running sum - increment when gene in a set, decrement when not**
3. **Maximum of the runnig sum is the enrichment score - larger means genes in a set are toward top of the sorted list**
4. **Permute subject labels to calculate significance p-value**



Subramanian, Tamayo, et al. (2005, PNAS 102, 15545-15550)
Mootha, Lindgren, et al. (2003, Nat Genet 34, 267-273).

Advantages of using GSEA

- **Agnostic to the type of gene set and the source of annotation**
- **Operates on any ordered gene list**
- **Does not require the choice of a gene selection threshold or the explicit definition of a statistically significant marker set**
- **Uses distribution-free, non-parametric, permutation-based test procedures with increased statistical power**
- **Incorporates the permutation of phenotype labels thereby preserving the “biological” correlation structure of the markers**
- **Takes into account multiple hypotheses testing of multiple gene sets**
- **Less prone to false-positives than on the gene-level**

Many GO enrichment tools

- **GOStat, http://gostat.wehi.edu.au/**
- **GOrilla, Gene Ontology enRIchment anaLysis and visuaLizAtion tool http://cbl-gorilla.cs.technion.ac.il/**
- **g:Profiler, http://biit.cs.ut.ee/gprofiler/**
- **Metascape, http://metascape.org/**
- **ToppGene, https://toppgene.cchmc.org/**
- **WebGestalt - WEB-based GEne SeT AnaLysis Toolkit, http://www.webgestalt.org/**
- **R packages, clusterProfiler, https://www.bioconductor.org/packages/devel/bioc/html/clusterProfiler.html**

Analysing effectively

Parallel scripts in Puhti

Analysing effectively: let the script do the work for you

- **The steps learned today are repeated to all the samples in your dataset**
  - You don't want to type the same commands several times, and risk making mistakes between samples
- **Some analysis steps take hours to complete (alignment, read counting)**
  - Bit boring to just wait
- **Thus: first test, then automate = write a (batch) script!**
  - At CSC, you can then run the analysis effectively on our supercomputer Puhti
    - Parallelization

Puhti 101

- **Login with your username:**

```
ssh <csc_username>@puhti.csc.fi
```

- **Move to your projects SCRATCH directory**
  - SCRATCH directories are shared for the project
    - Make your own folder there!
  - Don’t run analysis on your HOME directory or in the login node
- **We are running the same tools for several samples**
  - = “awkwardly parallel” task => “array job”
  - Write a batch script (rnaseq_array_job_script.sh)

```
sbatch rnaseq_array_job_script.sh
squeue -l -u your_username
```

- **Modules: pre-installed tools are loaded in use with *module* command**

```
module load biokit
```

- **Store data in Allas! (SCRATCH is cleaned)**

```
cd /scratch/project_xxxxxxx
mkdir rnaseq_test_yourname
```

Array script

- https://docs.csc.fi/computing/running/array-jobs/

```
#!/bin/bash -l
#SBATCH --job-name=array_job
#SBATCH --output=array_job_out_%A_%a.txt
#SBATCH --error=array_job_err_%A_%a.txt
#SBATCH --account=<project>
#SBATCH --partition=small
#SBATCH --time=02:00:00
#SBATCH --ntasks=1
#SBATCH --mem-per-cpu=4000
#SBATCH –cpus-per-task=2
#SBATCH --array=1-50


# run the analysis command
my_prog data_${SLURM_ARRAY_TASK_ID}.inp data_${SLURM_ARRAY_TASK_ID}.out
```

Design of experiments

When planning an experiment, consider

- **The number of biological replicates needed. Depends on**
  - Biological variability and technical noise
  - Expression level, fold change and sequencing depth
- **Sample pairing**
- **Sequencing decisions**
  - Number of reads per sample (~sequencing depth)
  - Read length (longer is better)
  - Paired end or single end (PE is better)
  - Stranded or unstranded (stranded is better)
  - Batch effects

Relevant concepts

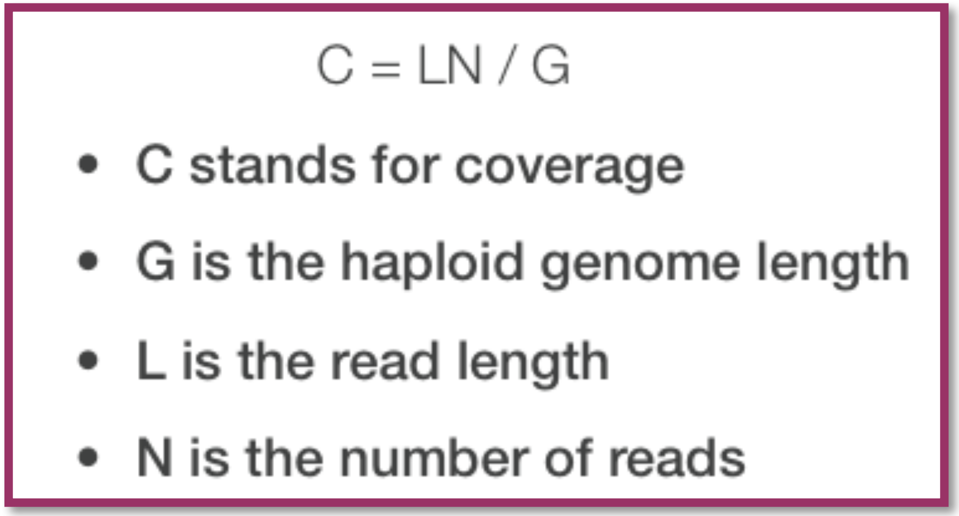
- **Read depth = coverage**

  = how many reads per each nucleotide, on average?

  - It depends… 30X – 300X ?
  - With RNASeq we count **reads per sample**. 20-200 M ?

- **Read length**? 75-300 bp?

- **Paired end**, **single end**?



C = LN / G

- C stands for coverage
- G is the haploid genome length
- L is the read length
- N is the number of reads

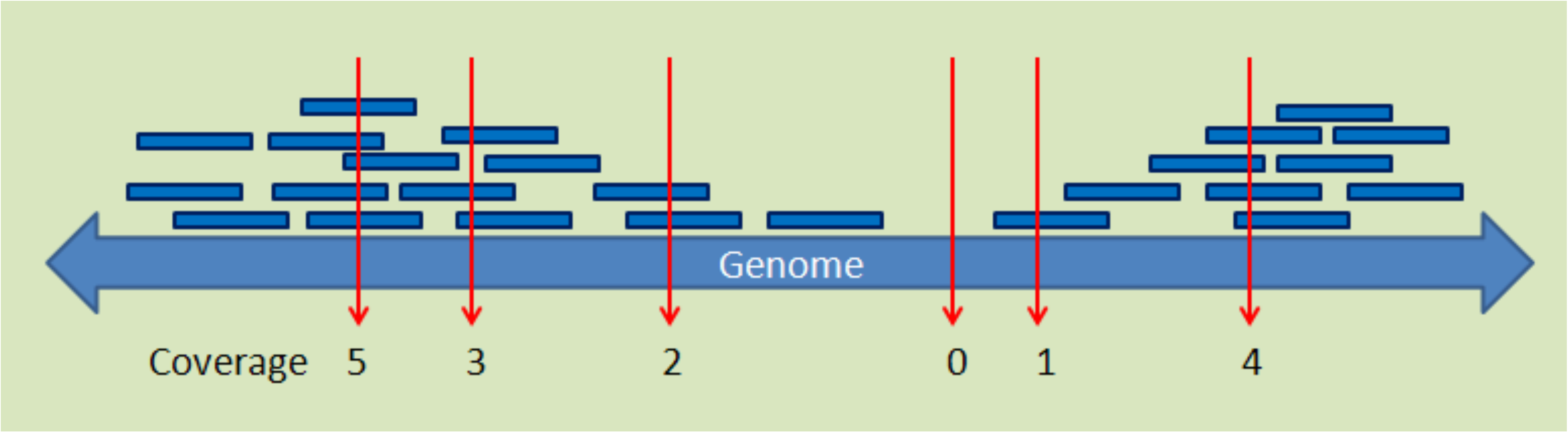
- **Sequencing capacity** = how many cycles, how many reads per flow cell, how many lanes in the flow cell? => depends on the device.

- **Heterogeneity** of the sample material (cell line vs. tumor sample)

https://genohub.com/recommended-sequencing-coverage-by-application/

Coverage

- **Whole Genome Sequencing (WGS):**
  - Genotype calls 35x, INDELs 60x, SNVs 30x
- **RNASeq:**
  - Differential expression profiling 10-25 M reads, allele specific expression 50-100 M
  - Alternative splicing 50-100 M
  - De novo assembly >100 M
- **ChiP-Seq: 10-15 M (sharp peaks), 20-40 M (broad)**



Relevant concepts

The device

- Read length = max. number of cycles
- Reads per flow cell
- Lanes per flow cell
- Multiplexing
- Paired end option
- Targeted sequencing = sequencing panels

The question

- Requirement for coverage / read depth
- SNP, indels, de novo assembly, variant discovery, novel mutation discovery, expression analysis…
- Mitochondria? Highly expressing genes active?

The sample

- Tumor sample, blood sample, model organism, cell line…
- Possible contaminations?
- How many samples available?

Time, money, experience, availability?

Technical vs. biological replicates

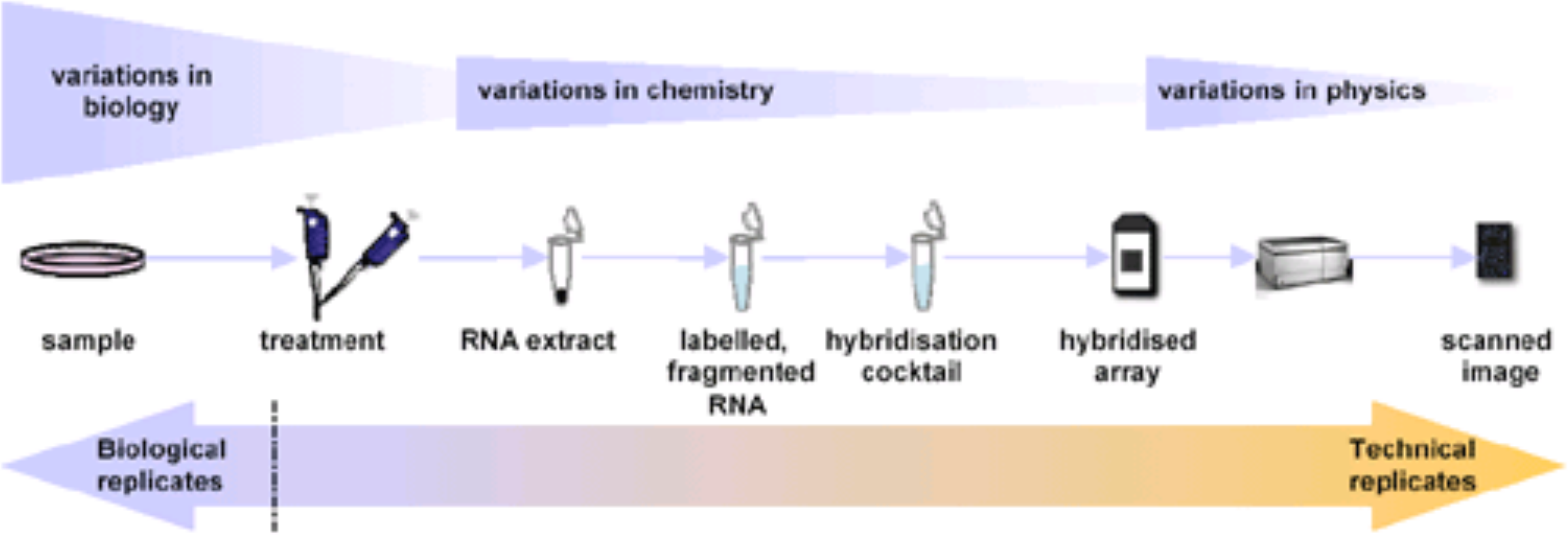
- **Biological replicates are separate individuals/samples**
  - Necessary for a properly controlled experiment

- **Technical replicates are repeated sequencing runs using the same RNA isolate or sample**
  - Waste of resources?
  - Can cause unnecessary variance reduction → increases number of false positives

- **Avoid mixing of biological and technical replicates!**

Technical vs. biological replicates

Distinction between technical and biological replicates is fuzzy.



Where do we stand with **cell lines**?

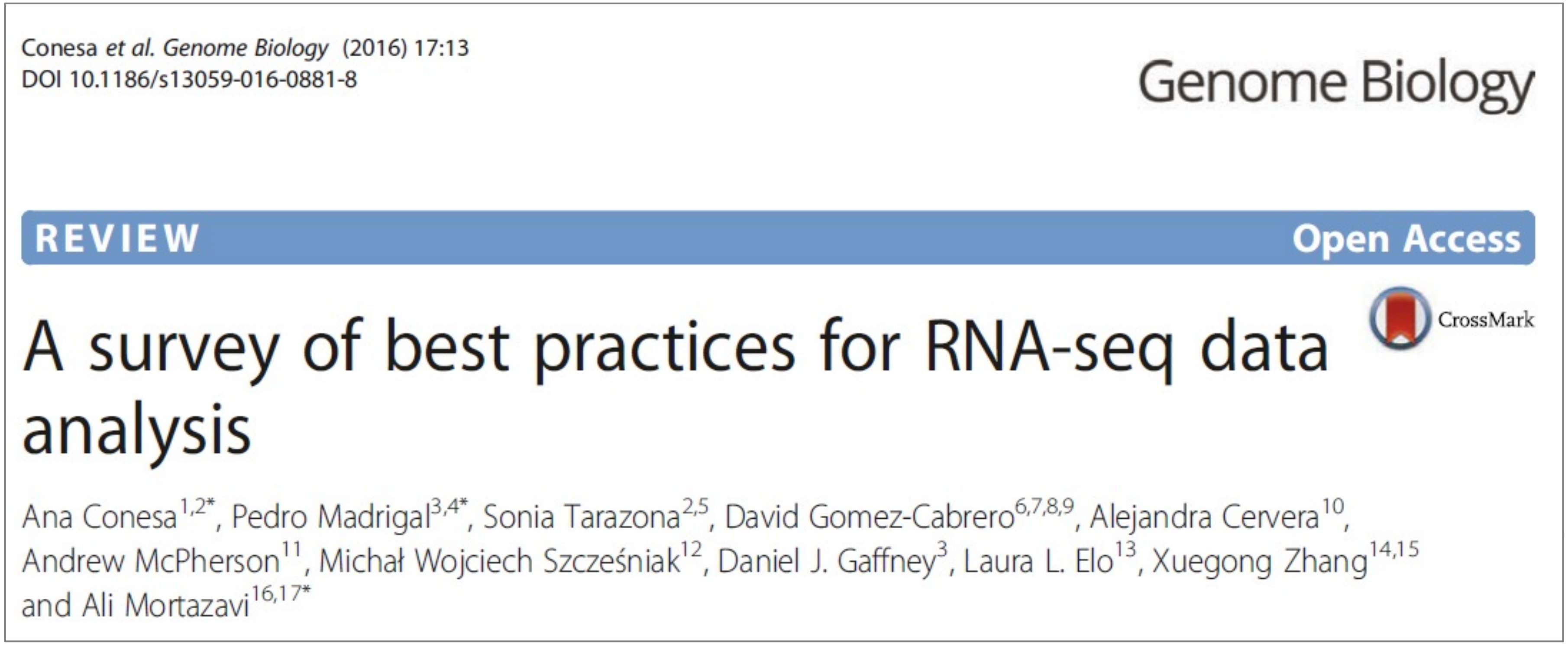
Replicate number

- **Publication quality data needs at least 3 biological replicates per sample group.**
  - This can be sufficient for cell-cultures and/or test animals

- **More reasonable numbers:**
  - Cell cultures / test animals: 3 is minimum, 4-5 OK, >7 excellent
  - Patients: 3 is minimum, 10-20 OK, >50 good
  - Power analysis can be used to estimate sample sizes

How many reads per sample do I need?

- **Depends on the transcriptome and what you want to investigate**
  - https://genohub.com/recommended-sequencing-coverage-by-application/
  - Differential expression 10-25 M reads
  - Allele specific expression 50-100 M
  - Alternative splicing 50-100 M
  - *De novo* assembly >100 M
- **More reads or more replicates?**



Conesa *et al. Genome Biology* (2016) 17:13
DOI 10.1186/s13059-016-0881-8

Genome Biology

REVIEW Open Access

A survey of best practices for RNA-seq data analysis

Ana Conesa[1,2*], Pedro Madrigal[3,4*], Sonia Tarazona[2,5], David Gomez-Cabrero[6,7,8,9], Alejandra Cervera[10], Andrew McPherson[11], Michał Wojciech Szcześniak[12], Daniel J. Gaffney[3], Laura L. Elo[13], Xuegong Zhang[14,15] and Ali Mortazavi[16,17*]

Read depth / number of reads per sample

- **Some recommendations available**
- **Heterogeneous sample => more depth needed**
  - For example tumor samples or when there is a doubt of contamination
- **RNASeq: some highly expressed transcript may hoard all the resources**
- **Targeted panels: how well are they targeting**
- **More depth or more replicates?**

Balance sample groups across batches

- **You can't account for a batch effect if all your control samples were run in one batch and the drug samples in the other**
  - DESeq2 would give an error: "*The model matrix is not full rank*"
- **Balance sample groups cross batches**

Problem: You have 8 samples, 4 controls and 4 treated samples. You can only fit 4 samples in one sequencing run, which means you will have 2 batches. How would you form the batches?

Option A:

sample	batch	treatment
1	1	control
2	1	control
3	1	control
4	1	control
5	2	drug
6	2	drug
7	2	drug
8	2	drug

✗

Option B:

sample	batch	treatment
1	1	control
2	1	control
5	1	drug
6	1	drug
3	2	control
4	2	control
7	2	drug
8	2	drug

✓

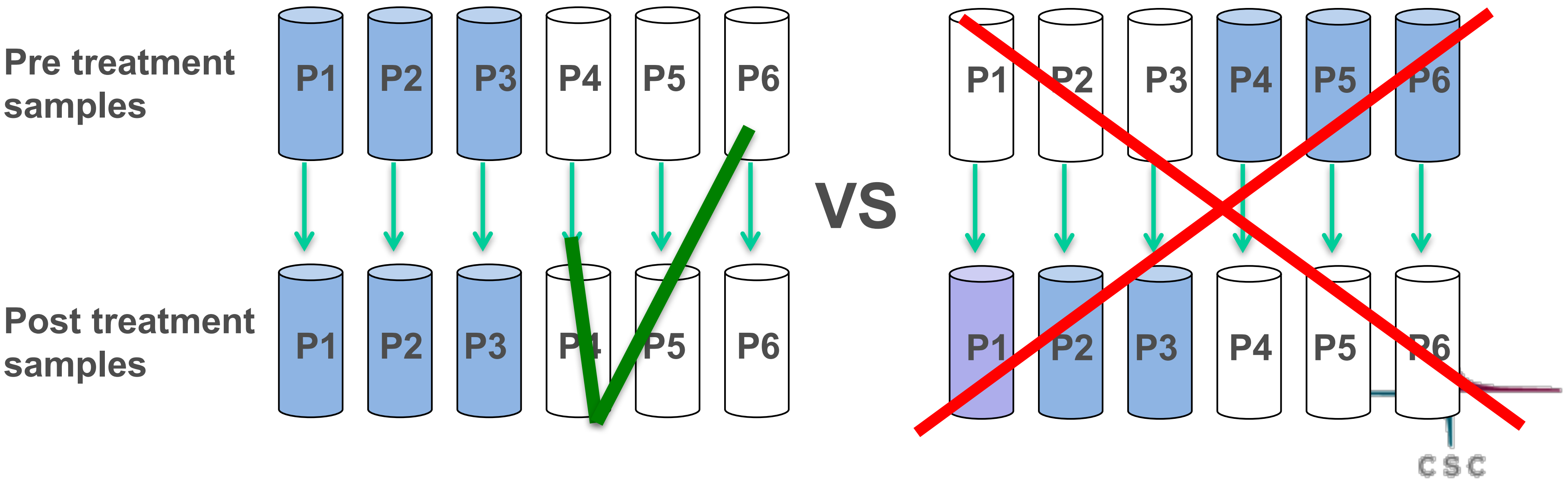
CSC

Paired samples

- **Use of matched samples reduces variance, as individual variation can be tackled using a matched control**
  - Pre vs. post treatment samples
  - Tumor vs. normal samples from the same patient

Problem: 6 patients, 2 samples from each. Enough resources to sequence only 6 samples. Which option do you choose?

Option A: **Option B:**



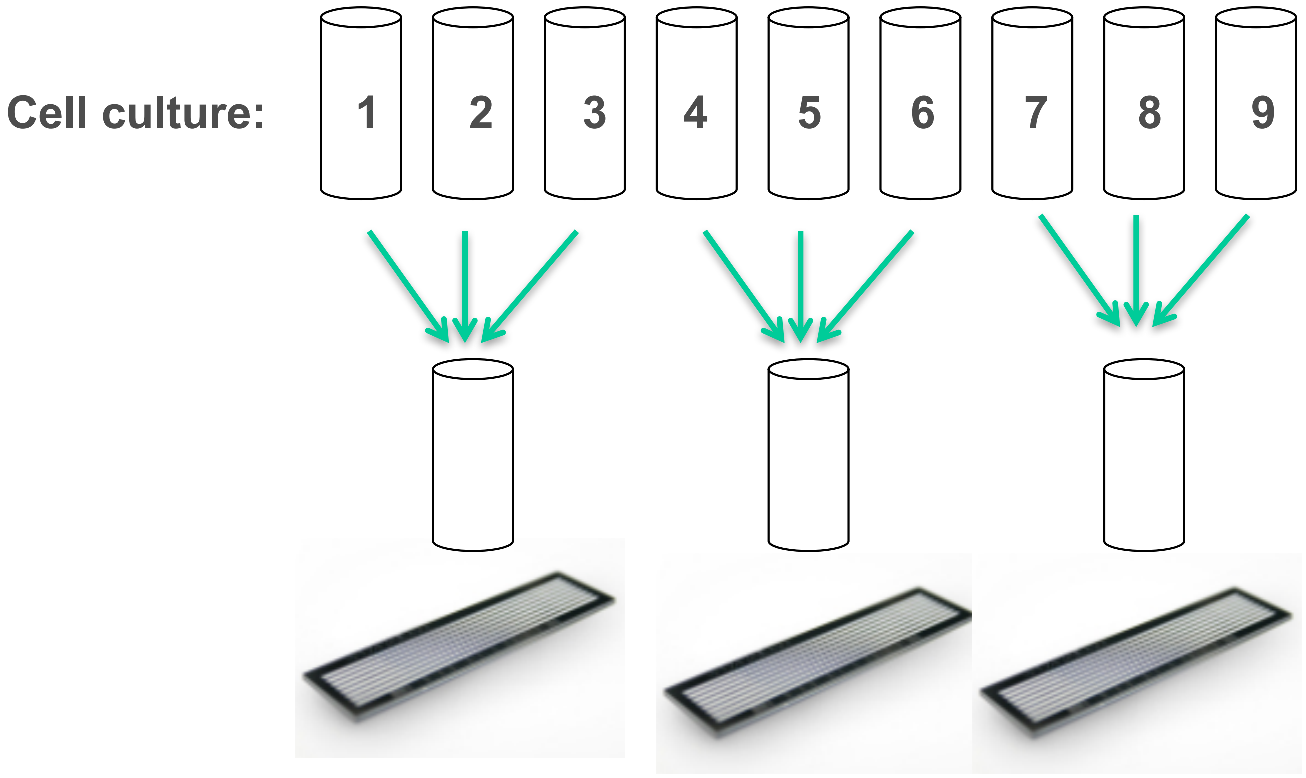
Pre treatment samples

Post treatment samples

VS

Pooling

- **When possible, measure each sample on its own.**
  - If this is not possible (too expensive or not enough material), samples can be pooled to reduce variance
  - Risk: If some of the samples are outliers, the pool is unusable
- **Make pools as similar as possible**

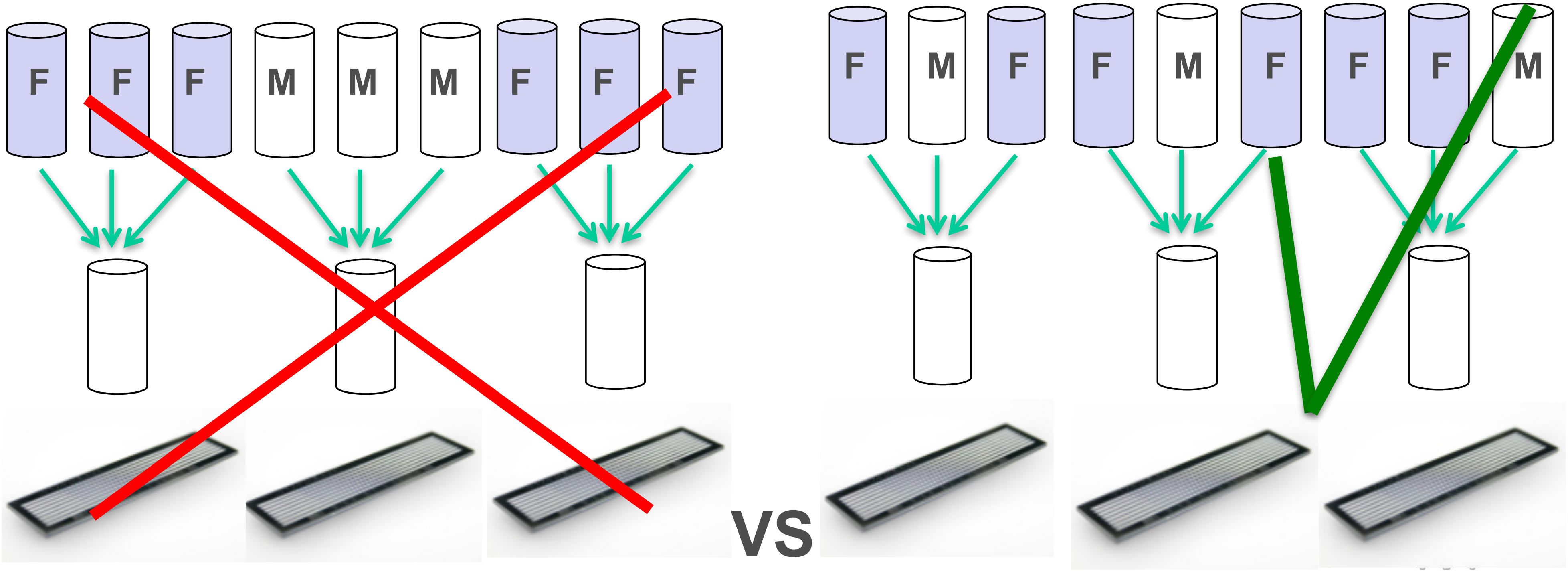


Pooling

- **Make pools as similar as possible**
- **Avoid pooling of similar kinds of samples into one pool**

Problem: We have 9 control samples, but we need to pool 3 samples together. 6 samples are from females and 3 from males.

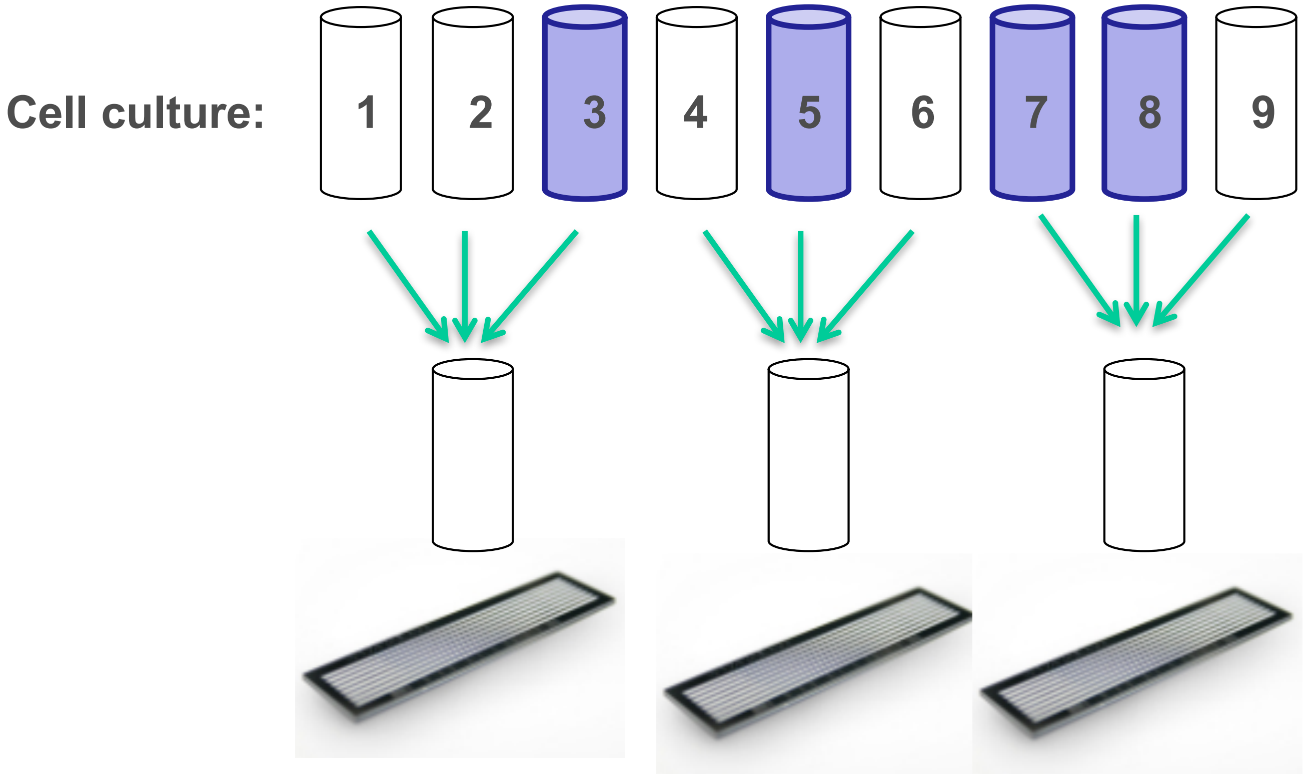
Option A: **Option B:**



Pooling

Something to consider:

What if some of your samples are outliers, or have a contamination?



Reference samples

- **Don’t compare apples to oranges!**
  - Cancer sample vs. normal sample –where do you get the “normal sample”?
    - Same tissue, “healthy” parts from the same patient?
    - Same, healthy tissue from another patient?
    - Similar tissue from the same patient?
    - Blood sample from the same patient?
    - Cell line?

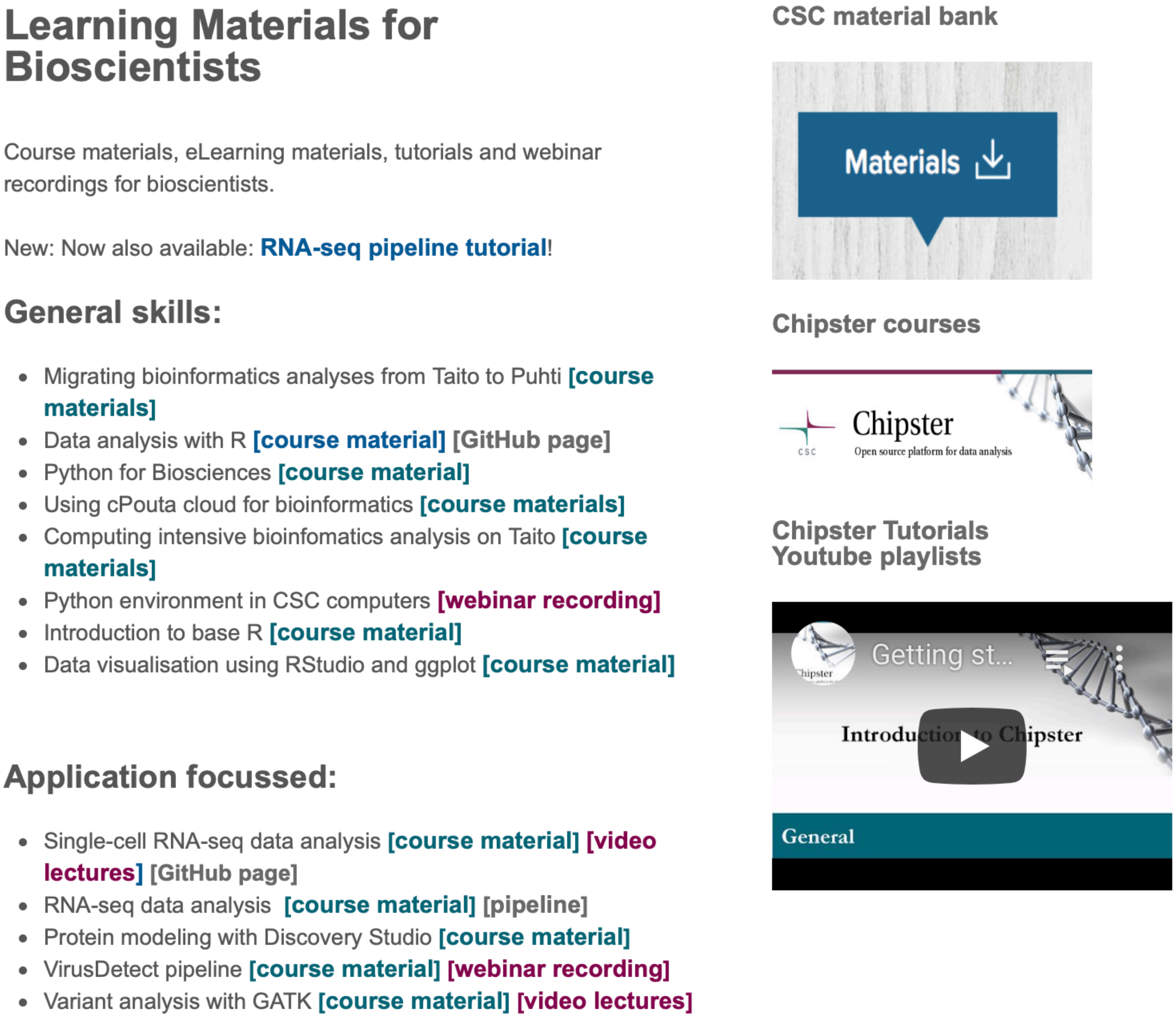
Getting started at CSC + other materials

Getting started at CSC

- **Overview and links to manual pages: https://research.csc.fi/accounts-and-projects**

- **Step 1. Create a user account**
  - Create a CSC account by logging in CSC's customer portal MyCSC with Haka or Virtu.
- **Step 2. Create or join a project**
  - to access Puhti, Mahti, Allas, cPouta, ePouta, Rahti, Kaivos and/or IDA.
  - A) Create a CSC project to access and invite users
  - B) Ask project manager to invite you
- **Step 3. Add service access for your project**
  - Only the project manager can add services.
- **Step 4. Apply for more resources/billing units, if needed**
- **Step 5. Renew your password annually**

https://research.csc.fi/bioscience-learning-materials



Learning Materials for Bioscientists

Course materials, eLearning materials, tutorials and webinar recordings for bioscientists.

New: Now also available: **RNA-seq pipeline tutorial**!

General skills:

- Migrating bioinformatics analyses from Taito to Puhti **[course materials]**
- Data analysis with R **[course material] [GitHub page]**
- Python for Biosciences **[course material]**
- Using cPouta cloud for bioinformatics **[course materials]**
- Computing intensive bioinfomatics analysis on Taito **[course materials]**
- Python environment in CSC computers **[webinar recording]**
- Introduction to base R **[course material]**
- Data visualisation using RStudio and ggplot **[course material]**

Application focussed:

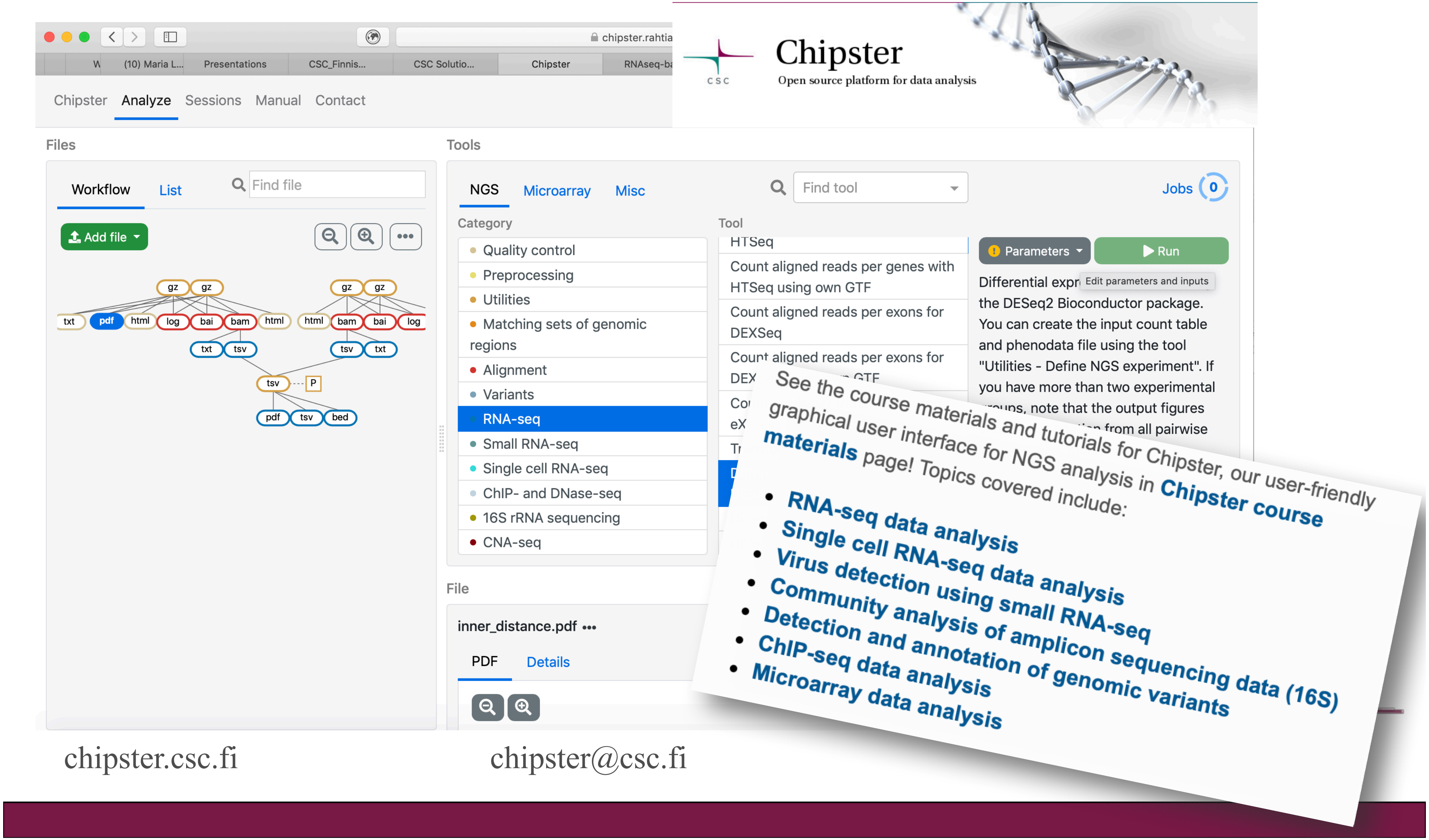
- Single-cell RNA-seq data analysis **[course material] [video lectures] [GitHub page]**
- RNA-seq data analysis **[course material] [pipeline]**
- Protein modeling with Discovery Studio **[course material]**
- VirusDetect pipeline **[course material] [webinar recording]**
- Variant analysis with GATK **[course material] [video lectures]**

CSC material bank

Chipster courses

Chipster Tutorials Youtube playlists

Chipster: Easy-to-use high-throughput data analysis tool



Differential expr... the DESeq2 Bioconductor package. You can create the input count table and phenodata file using the tool "Utilities - Define NGS experiment". If you have more than two experimental ...roups, note that the output figures ... from all pairwise

See the course materials and tutorials for Chipster, our user-friendly graphical user interface for NGS analysis in **Chipster course materials** page! Topics covered include:

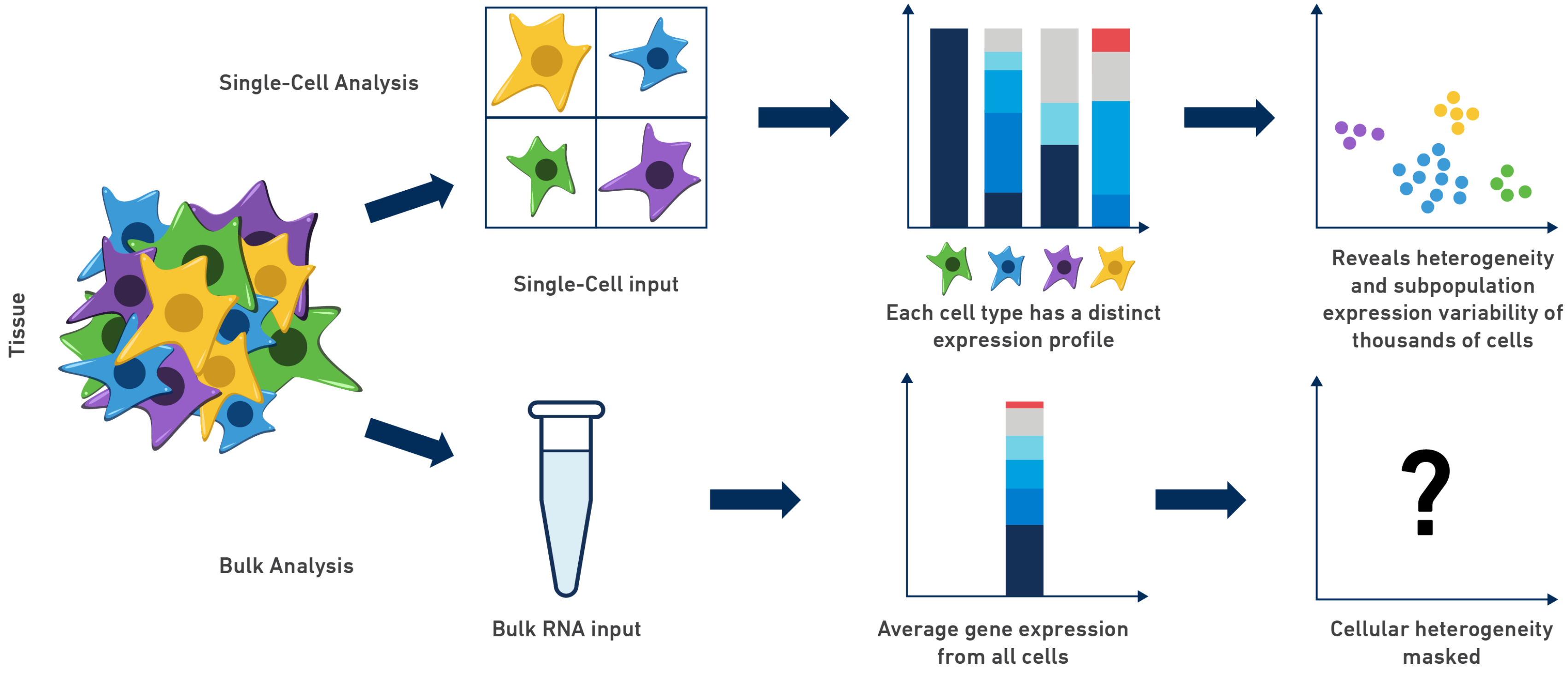
- **RNA-seq data analysis**
- **Single cell RNA-seq data analysis**
- **Virus detection using small RNA-seq**
- **Community analysis of amplicon sequencing data (16S)**
- **Detection and annotation of genomic variants**
- **ChIP-seq data analysis**
- **Microarray data analysis**

chipster.csc.fi chipster@csc.fi

Recent advances: single-cell RNAseq

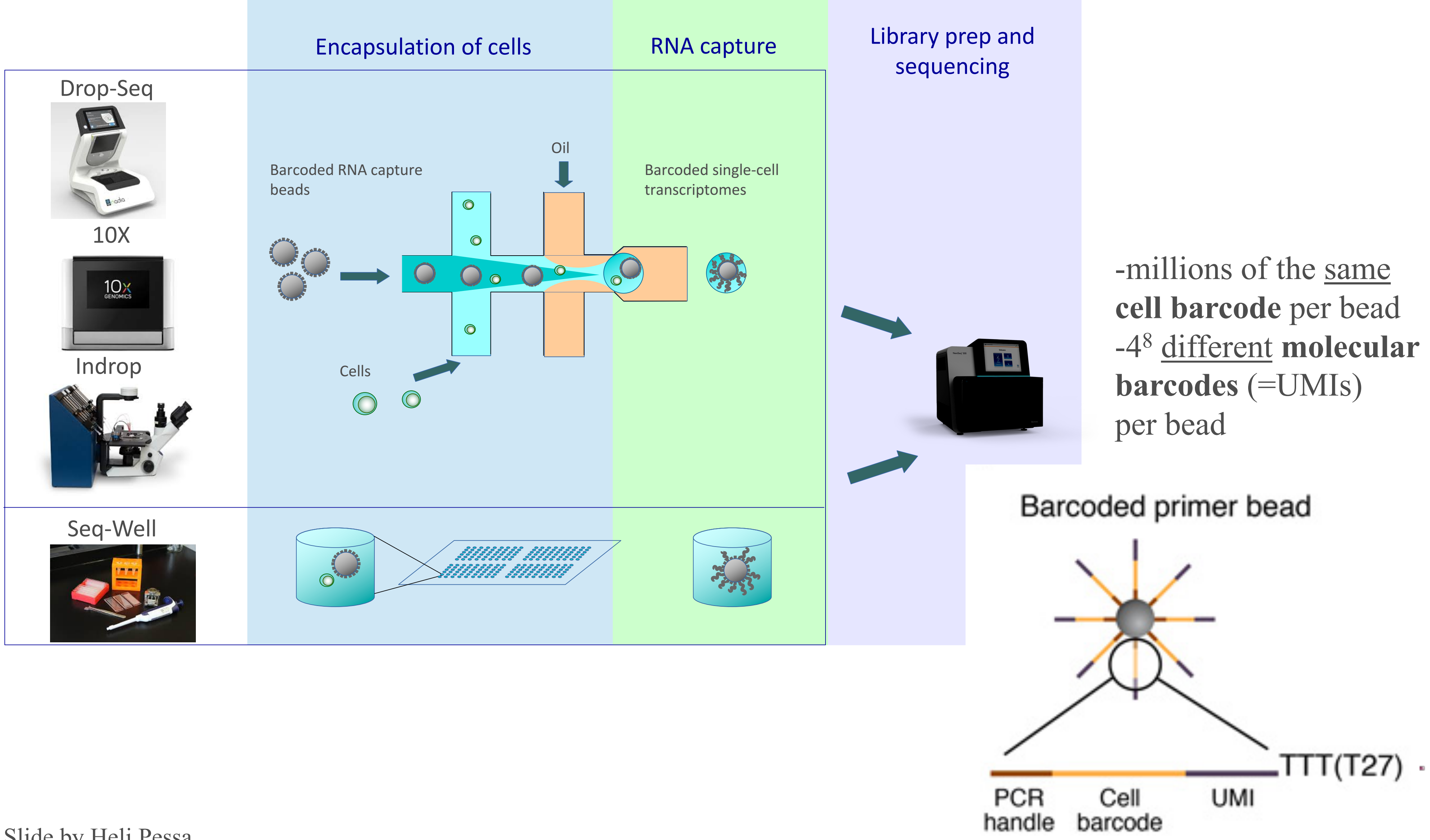
New directions: single cell RNA-seq

- **(Bulkl RNA-seq is very much in use, but scRNA-seq technology is warmly welcomed in many areas**
- **With scRNA-seq, cellular heterogeneity can be studied**


Single-Cell Analysis
Single-Cell input
Each cell type has a distinct expression profile
Reveals heterogeneity and subpopulation expression variability of thousands of cells
Tissue
Bulk Analysis
Bulk RNA input
Average gene expression from all cells
?
Cellular heterogeneity masked

https://www.biocompare.com/Bench-Tips/345311-Single-Cell-Set-Up-Sample-Preparation-Tips/

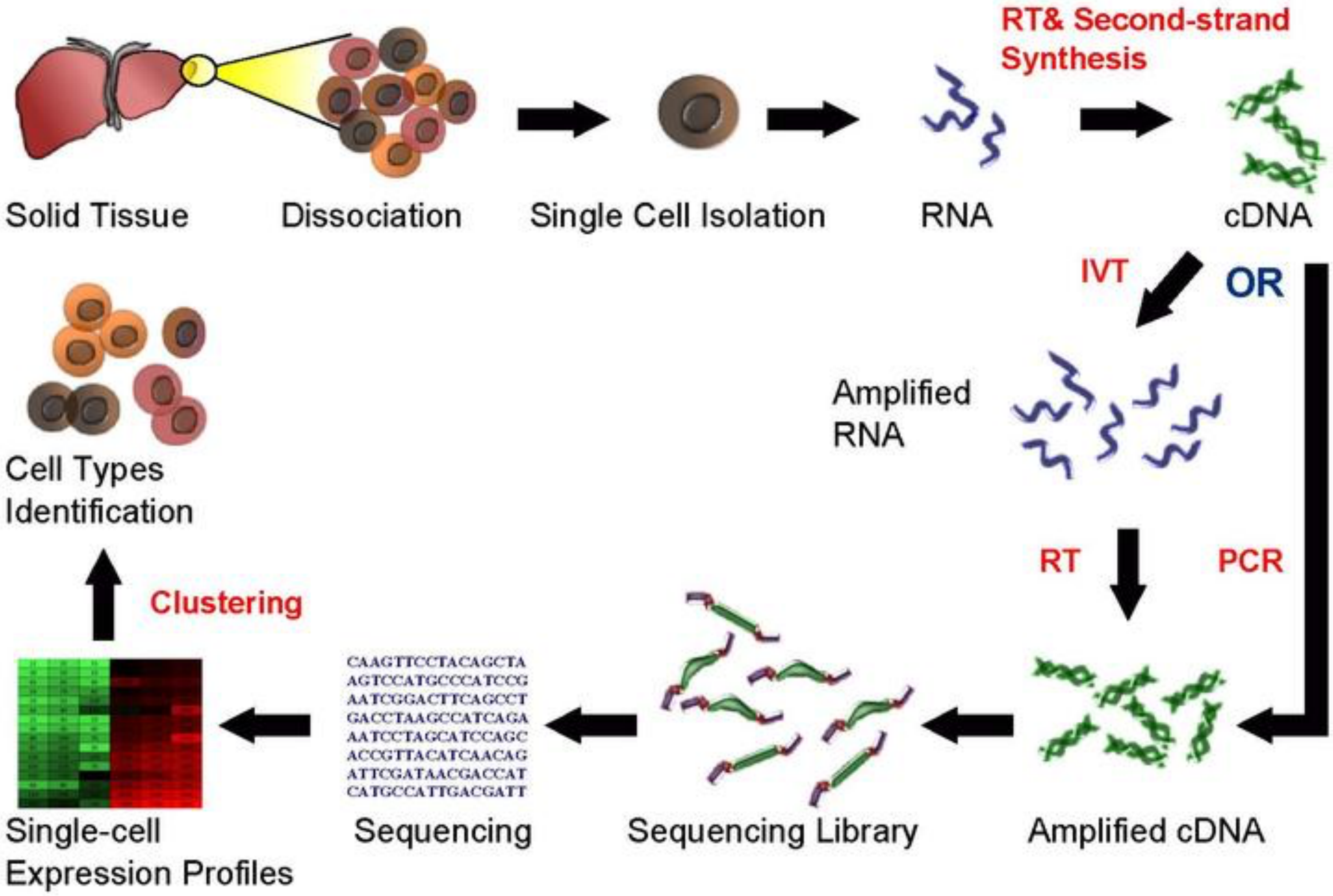
Different technologies for capturing single-cell transcriptomes



-millions of the same **cell barcode** per bead
-4^8 different **molecular barcodes** (=UMIs) per bead

Slide by Heli Pessa

Single Cell RNA Sequencing Workflow



https://hemberg-lab.github.io/scRNA.seq.course/introduction-to-single-cell-rna-seq.html

RNAseq vs scRNAseq data

RNAseq:

- 1 table, genes x samples
- Compare sample groups

	Sample 1	Sample 2	Sample 3	Control 1	Control 2	Control 3
Gene A	5	4	7	24	23	22
Gene B	50	54	52	12	12	11
Gene C	5	4	5	4	4	5
Gene D	33	34	32	21	32	43
…						

scRNAseq:

- Tables = samples
- Genes x cells (-> very wide tables)
- Lots of zeros
- Find clusters of similar cells in samples
- Compare clusters

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	…
Gene A	20	14	7	3	0	15	
Gene B	0	4					
Gene C	5	4					
Gene D	1	3					
…							

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	…
Gene A	5	0	2	0	1	0	
Gene B	50	51	52	12	12	11	
Gene C	5	0	5	0	0	1	
Gene D	0	1	2	0	7	0	
…							

Feedback

- **We would very much value your feedback!**
  - You will receive a course feedback link to your e-mail