Differential Gene Expression

RNA-seq Data Analysis Course, EBI, April 2020

Dr. Alexey Larionov

Department of Medical Genetics
Cambridge University, UK

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

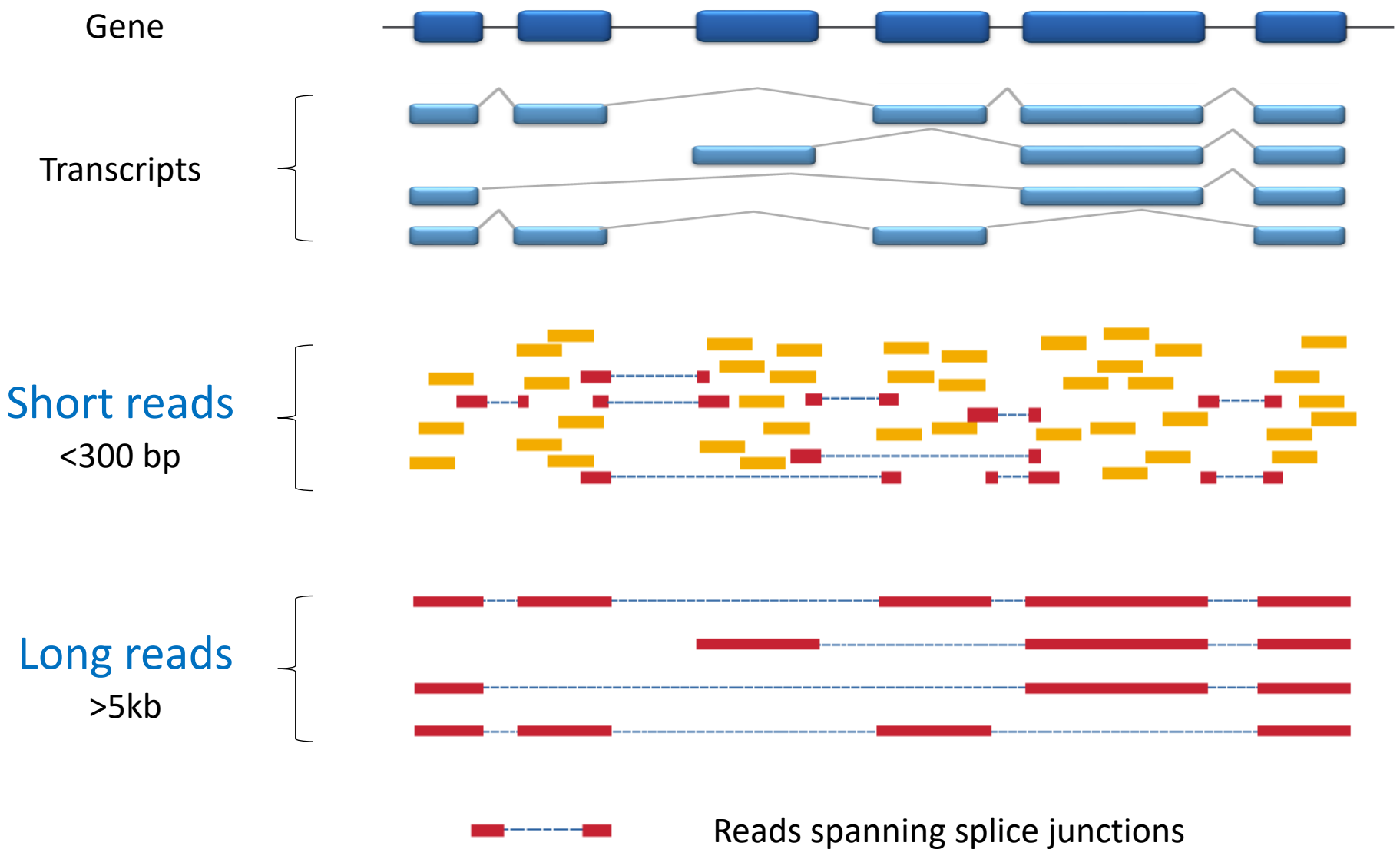
Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Short and long-reads in RNA-seq



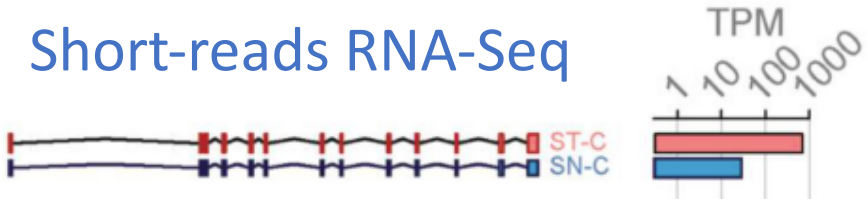
Complicated and inaccurate procedures for transcript assembly

No transcript assembly required

http://www.vib.be/en/training/research-training/courses/Archive_CourseRegistrations/GeneRegulation_Koenig.pdf

Short and long-reads in RNA-seq

Short-reads RNA-Seq

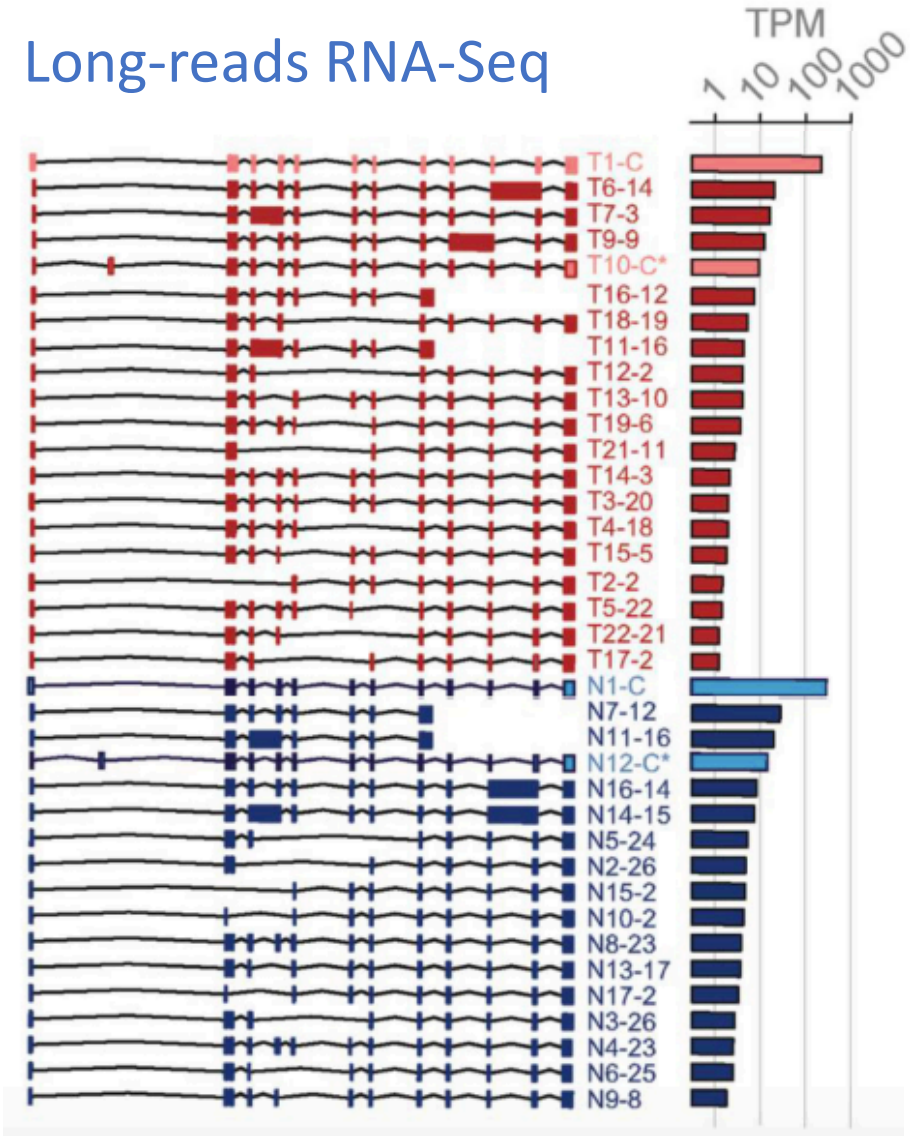


ANXA1 gene expression measurement
Sample A vs Sample B

… full-length RNA-sequencing … revealed a ~5-fold higher number of transcript isoforms than previously detected

Mays *et al* 2019

Long-reads RNA-Seq

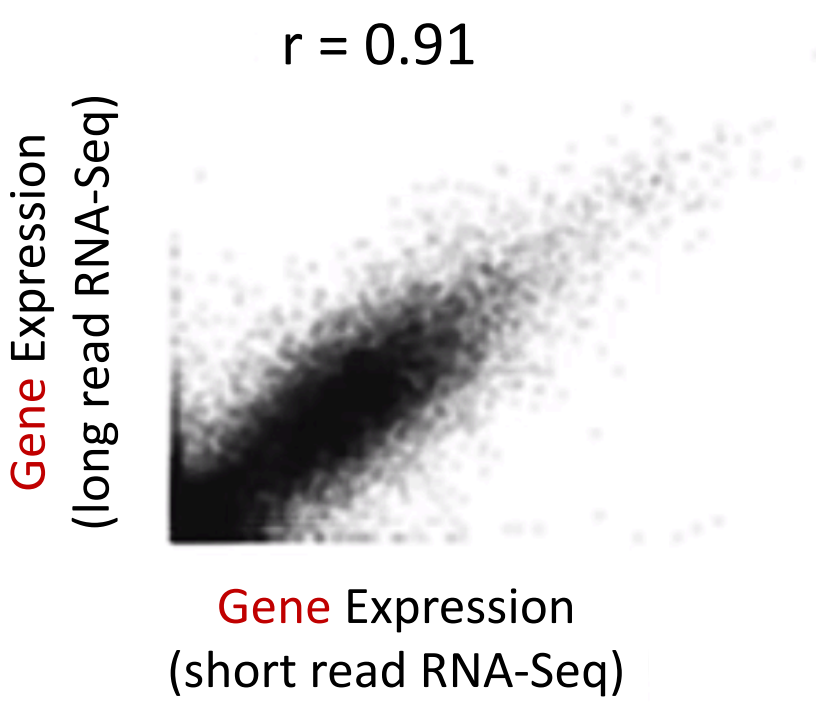


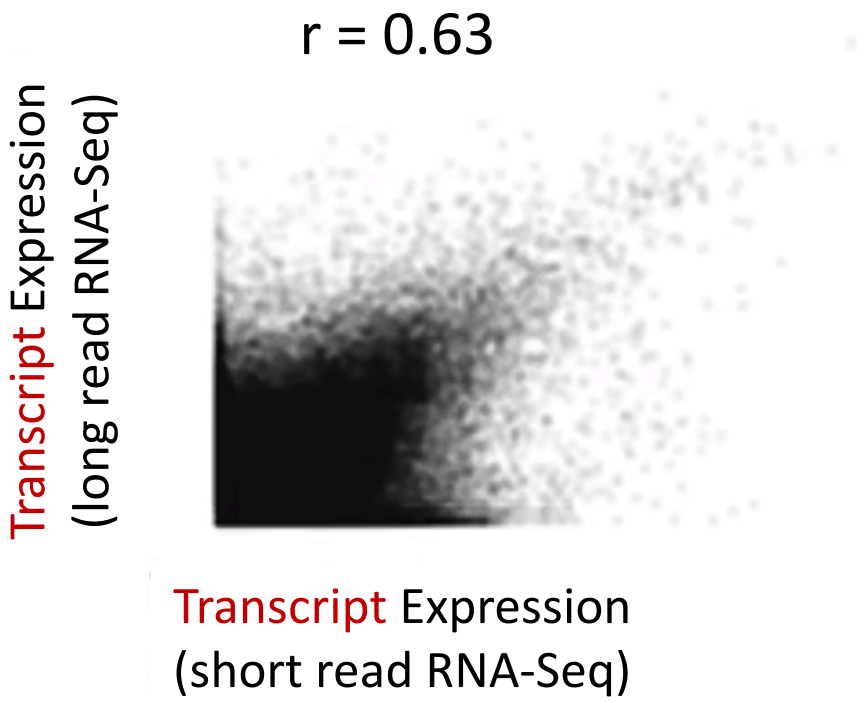
Tools for short reads are not good for long reads
Tools for long reads are yet emerging
(sometime platform-specific, e.g. ToFU for PacBio)

Mays et al 2019 Single-Molecule Real-Time (SMRT) Full-Length RNA-Sequencing Reveals Novel and Distinct mRNA Isoforms in Human Bone Marrow Cell Subpopulations Genes. 2019, 10, 253
ToFU: Gordon et al. 2015 Widespread Polycistronic Transcripts in Fungi Revealed by Single-Molecule mRNA Sequencing. PLoS ONE 10: e0132628
MISO: Katz et al 2010 Analysis and design of RNA sequencing experiments for identifying isoform regulation *Nature Methods* **7** 1009

Short and long-reads in RNA-seq

Gene expressions correlate well between short- and long – reads
Transcript expressions do not correlate well





Jonathan Göke, The SG-NEx project: nanopore long-read RNA-sequencing of human cancer cell lines
Nanopore Community Webinar, 28Feb 2019
https://globalmeet.webcasts.com/viewer/event.jsp?ei=1227094

Differential Gene Expression

RNA-seq Data Analysis Course, EBI, April 2020

Dr. Alexey Larionov

Department of Medical Genetics
Cambridge University, UK

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

RNA-Seq gene expression analysis

Reads

FastQC, Trimming adaptors
Removing poor bases and reads
Cutadapt, Trimmomatic

Alignment to genome

with gapped aligners
STAR, (r)subreads-align,
minimap2, GMAP

Transcripts assembly

from genome-alignment
StringTie, Cufflinks, ToFU

Alignment to transcriptome

with un-gapped aligners
Bowtie

Alignment-free quantification

Salmon , Kallisto, Sailfish

De-novo transcriptome assembly

becoming possible with
long reads
…

Counting reads

(r)subreads-featureCounts
RSEM, HTSeq-count

Transcripts/genes counts

Differential expression

DESeq2, edgeR, limma-voom, MISO

Functional analysis

GO, Pathways, GSEA

Classification

Signatures, linear models, co-clustering
Machine learning: Random Forrest, ANN

RNA-Seq gene expression analysis

Reads

FastQC, Trimming adaptors
Removing poor bases and reads
Cutadapt, Trimmomatic

Alignment to genome

with gapped aligners
STAR, (r)subreads-align,
minimap2, GMAP

Transcripts assembly

from genome-alignment
StringTie, Cufflinks, ToFU

Alignment to transcriptome

with un-gapped aligners
Bowtie

Alignment-free quantification

Salmon , Kallisto, Sailfish

De-novo transcriptome assembly

becoming possible with
long reads
…

Counting reads

(r)subreads-featureCounts
RSEM, HTSeq-count

Transcripts/genes counts

Differential expression

DESeq2, edgeR, limma-voom, MISO

Functional analysis

GO, Pathways, GSEA

Classification

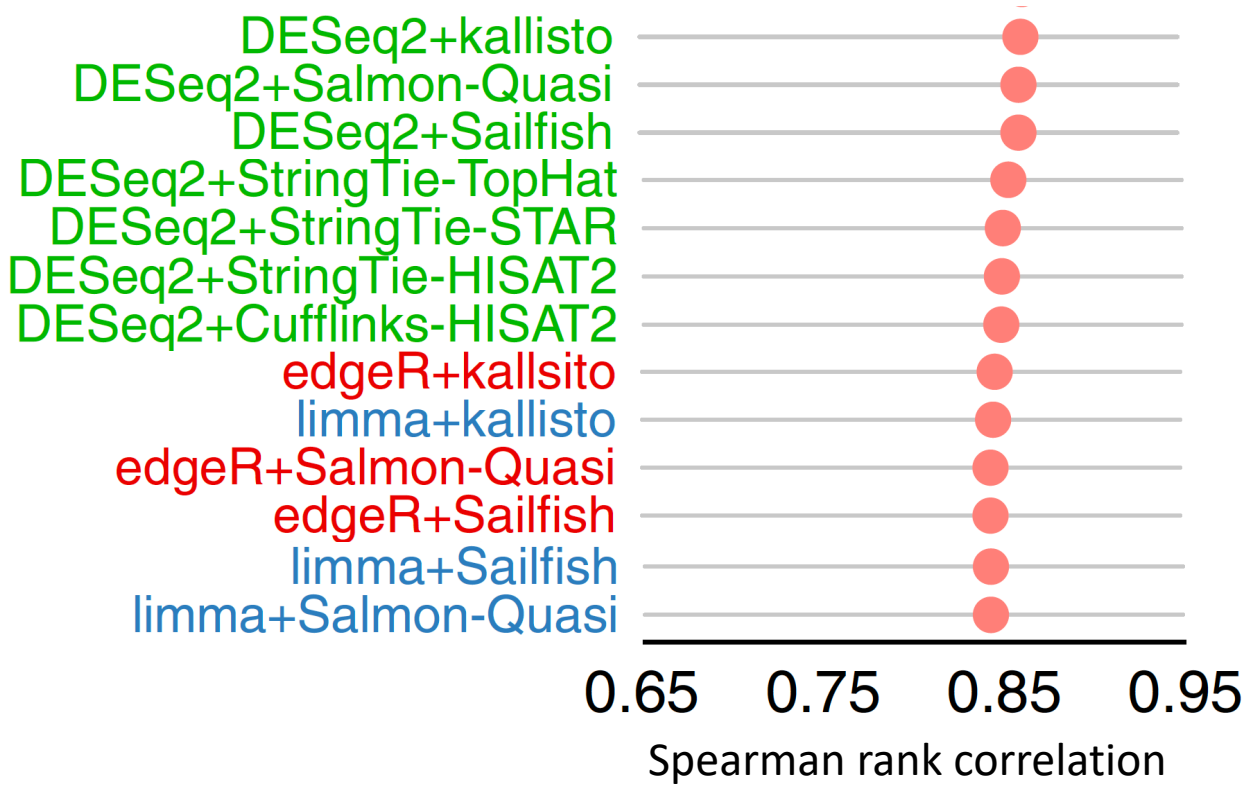
Signatures, linear models, co-clustering
Machine learning: Random Forrest, ANN

This lecture focuses only on genes differential expression although statistical principles could be extended to other types of counts

This slide focuses on options for short reads. Tools and practices for long reads are yet emerging

Comparison of tools for differential expression analysis in short-read RNA-Seq

RNA-Seq vs qRT-PCR



1,001 genes were measured in two samples by RNA-Seq and by qRT-PCR
Advantage of DESeq2 over edgeR/limma was even stronger in other comparosonns

Sahraeian et al 2017 Nature Communications 8:59 (Fig 6a)

Differential Gene Expression

Short reads RNA-seq Analysis with DESeq2 and edgeR

Dr. Alexey Larionov

Department of Medical Genetics
Cambridge University, UK

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

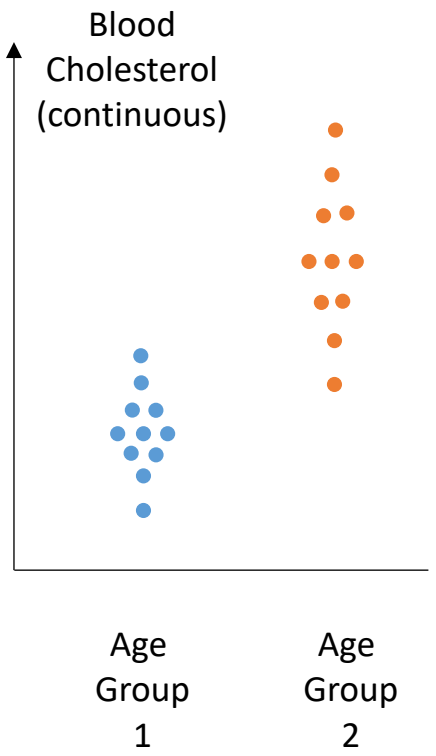
Comparison of methods
Comparison of results

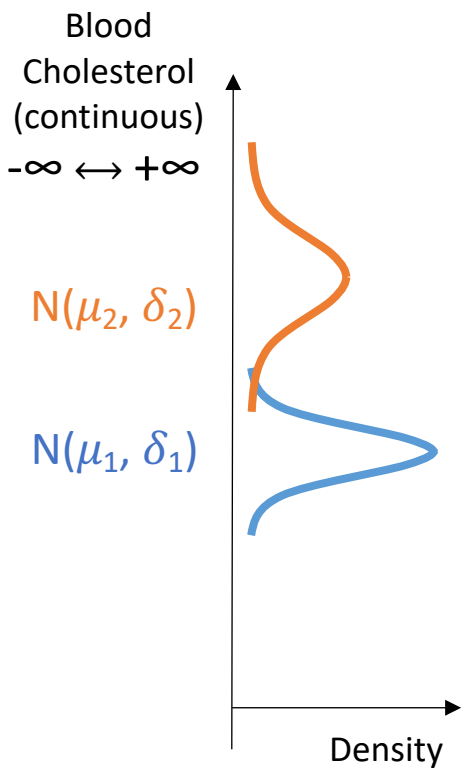
Recap of statistical approaches for detecting difference between groups

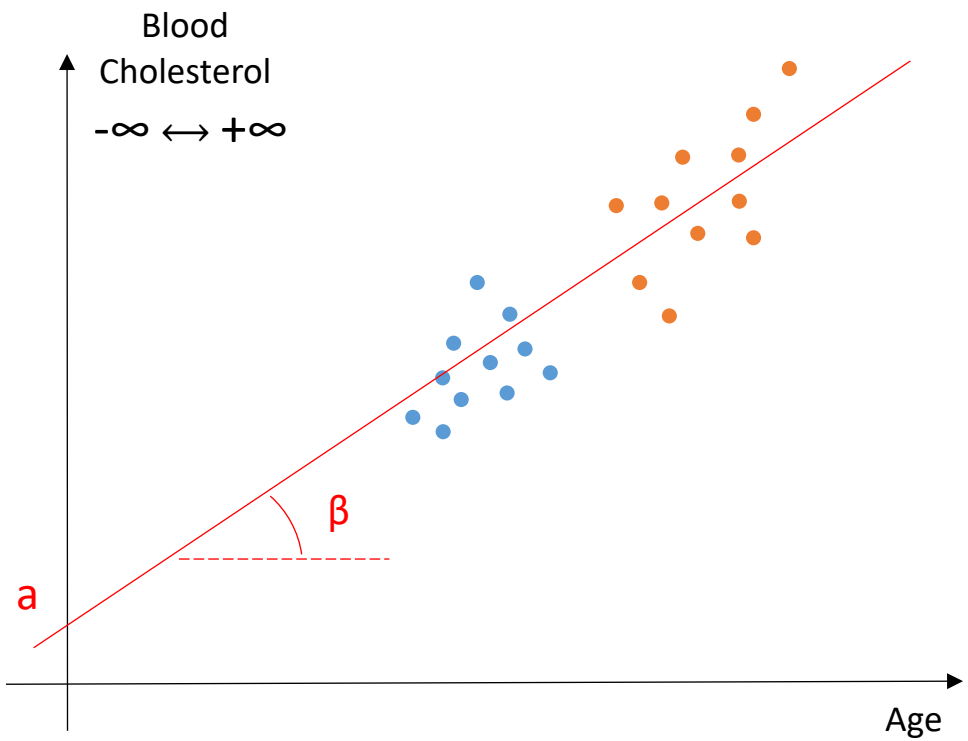
Source data → Modelling as Normal Distribution → Linear Regression Modelling

Cholesterol ~ N(μ, δ)

Cholesterol = a + β × Age







Visual assessment suggests a trend

Formal significance assessment
e.g. Student's test
$\mu_1 \neq \mu_2$ at $p < 0.05$

Significance of β ≠ 0;
Convenient to add covariates (confounders), e.g.:
Cholesterol = a + β1 × Weight + β0 × Age

Why can't we apply this framework for Differential Gene Expression Analysis ?

Problems

1) Raw counts in each sample depend on library size (depth of sequencing)
2) Low counts do not obey the "Normal" bell-shape distribution because they can't go below zero
3) The counts are discrete, which is better modelled by a discrete distribution
4) Small number of samples does not allow accurate estimation of dispersion (variance)
5) Testing for many genes at a time

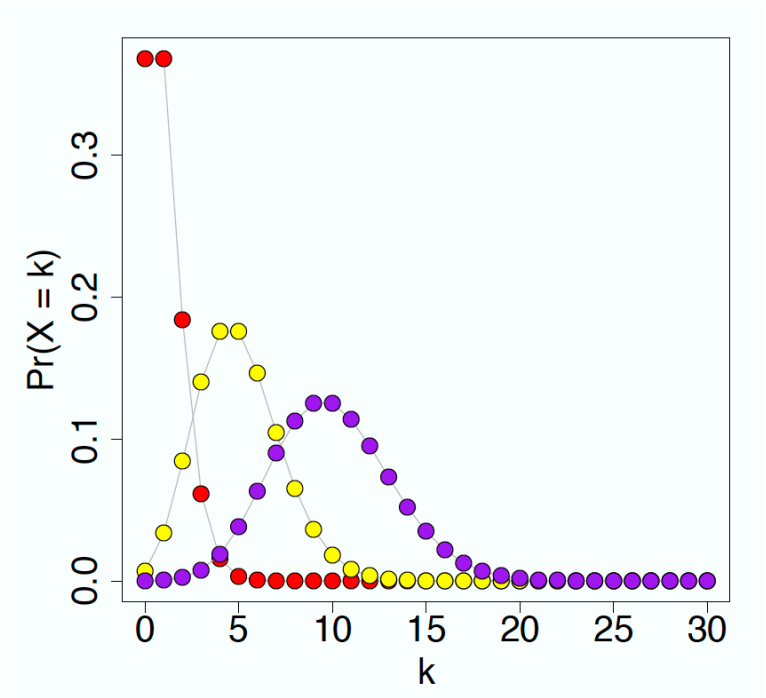
Solutions

1) Normalizing raw counts by the library size (discussed in later)
2) and 3) Choosing an appropriate discrete distribution
4) "Borrowing" data between genes for estimation of dispersion (discussed later)
5) Multiple testing correction (typically FDR)

Poisson distribution

Distribution of random independent events happening at a certain **mean** rate.
Mathematically, the dispersion(variance) is equal to the mean.

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$



By design describes the random sampling of molecules from a solution with given concentration.

Exactly matches the counts distribution in the technical replicas of RNA-seq:
e.g. sequencing of several aliquots from the same library.

Marioni *et al* 2008 RNA-seq: an assessment of technical reproducibility … https://genome.cshlp.org/content/18/9/1509

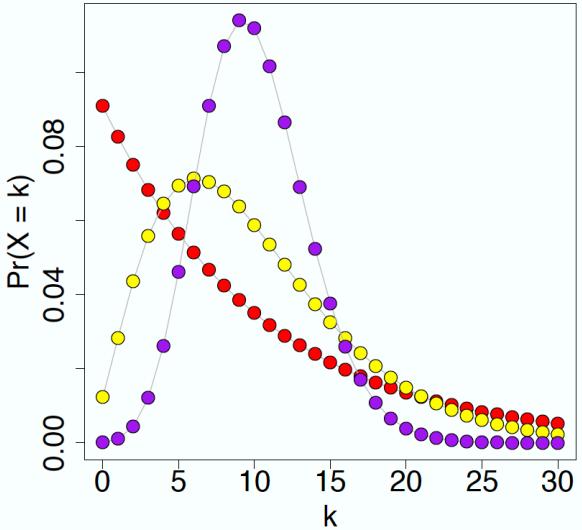
Overdispersion

Total variance	=	**Technical variance** i.e. between replicas within library, described by Poisson distribution	+	**Additional variance** e.g. between dishes of the same cell line or different tumors of the same type

Negative Binomial Distribution

Number of independent attempts until a certain number of successes
Mathematically, allows dispersion larger than mean

$$P(X = k) = \binom{k + r - 1}{k} \cdot (1 - p)^r p^k$$



"Similar" to Poisson: discrete and non-negative. However, unlike to Poisson allows to model the overdispersion.

Successfully used to model real-life RNA-seq data (details about the dispersion assessment will be discussed later).

Robinson and Smyth 2007 Moderated statistical tests … https://academic.oup.com/bioinformatics/article/23/21/2881/372869

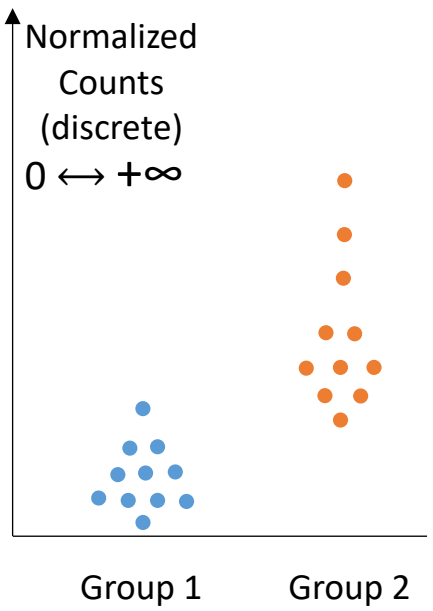
Overview of statistical approaches to Differential Genes Expression analysis

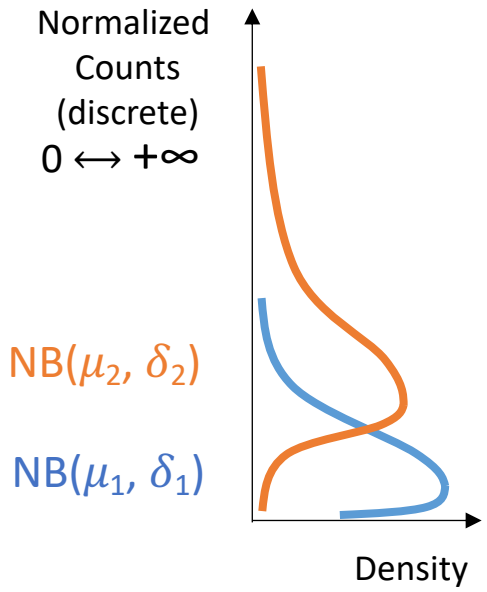
Source data
Normalized by library size

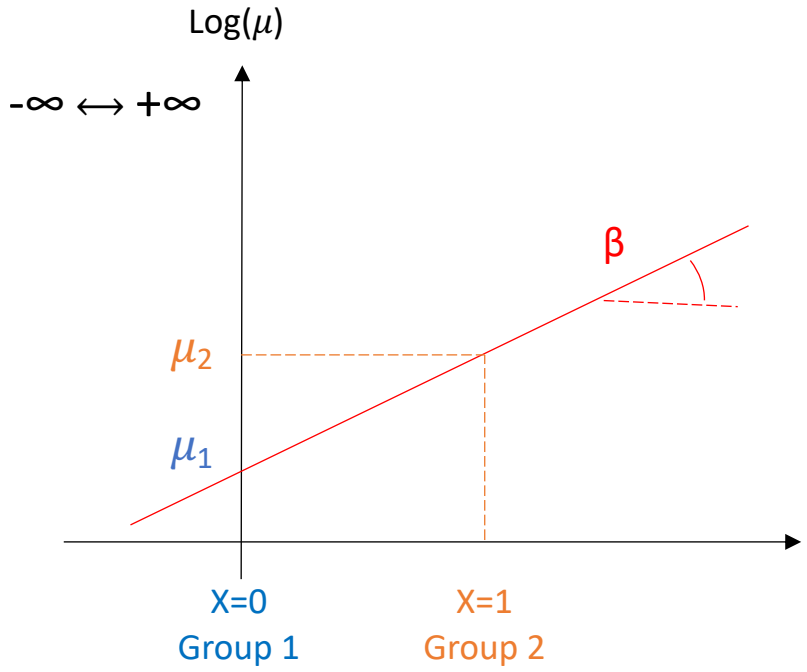
Modelling as Negative Binomial
Counts ~ NB(μ, δ)

Log-link

Generalized Linear Modeling
Log(μ) = a + β × Group







Visual assessment
suggests a trend

"Borrow" data for
"adaptive" variance evaluation:
for low number of cases
δ(*gene*) shrinks to
δ(*all genes with similar expression*)

Significance of β ≠ 0
Convenient adding covariates (confounders):
Counts = a + β1 × Batch + β0 × Group

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

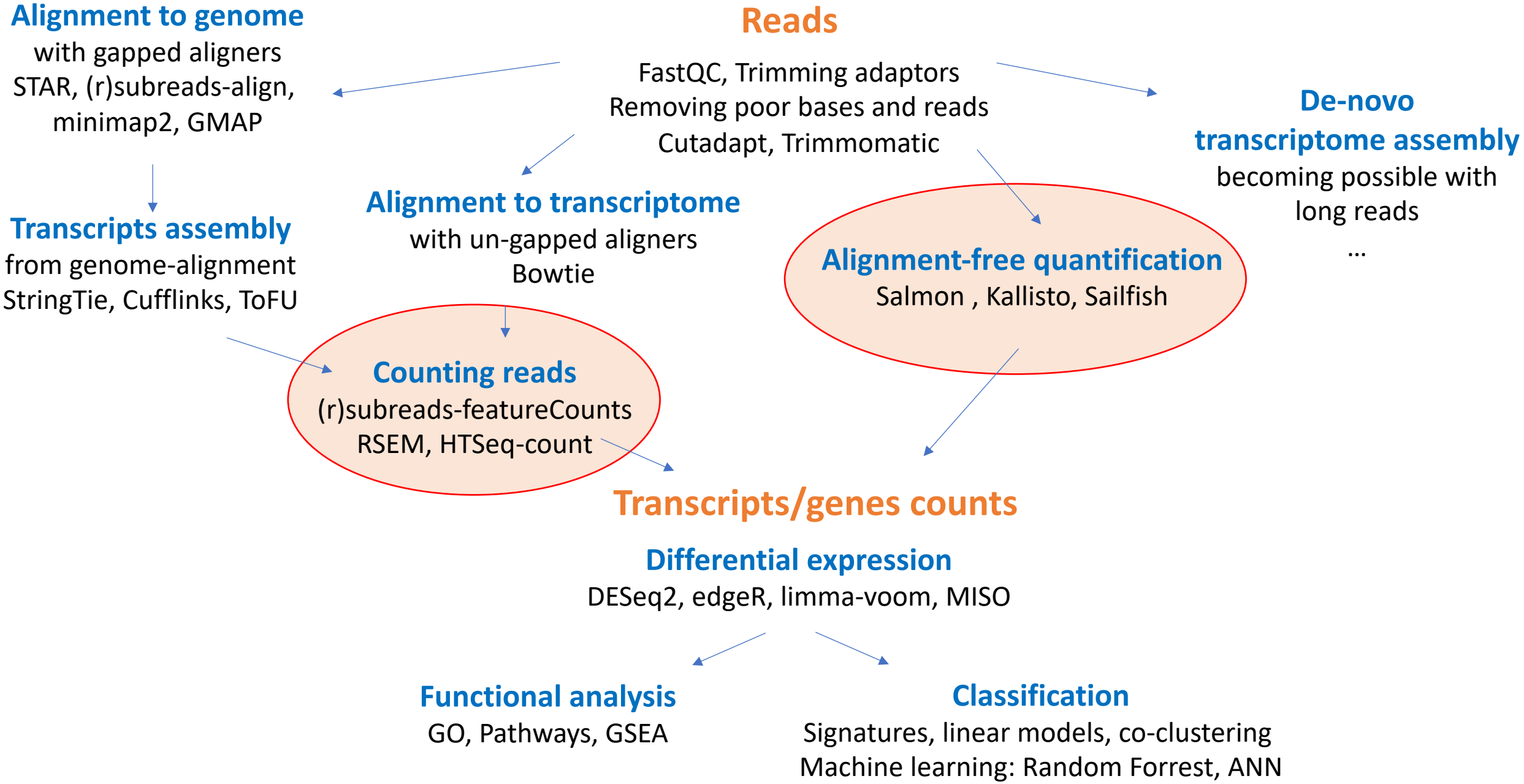
Visualizing results

MA- and Volcano plots

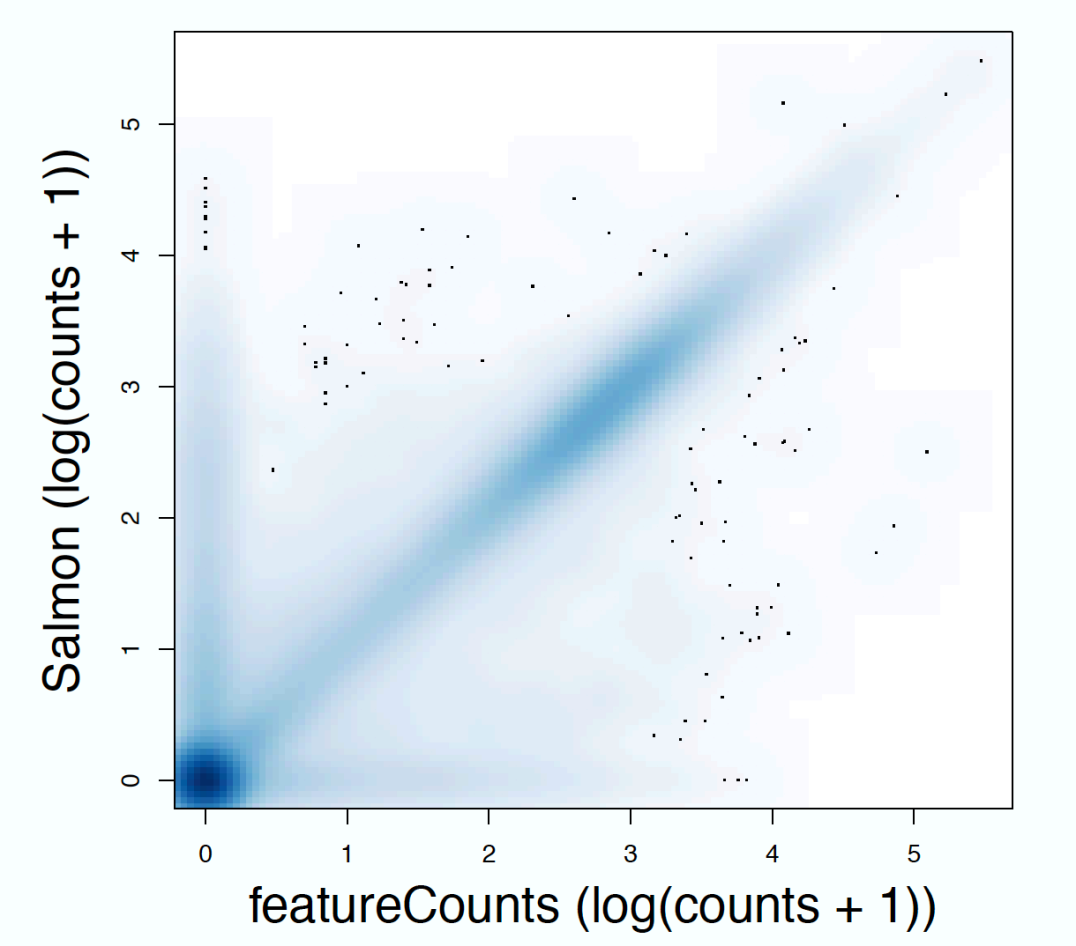
DESeq2 vs edgeR

Comparison of methods
Comparison of results

RNA-Seq gene expression analysis



Gene-level counts are reasonably close but not identical between different approaches



Caveats in getting raw counts …

For long-reads:

count of transcripts is reasonably natural
read ~ transcript (within reason…)

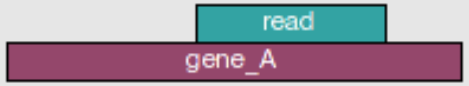
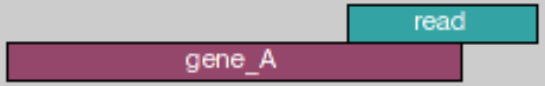
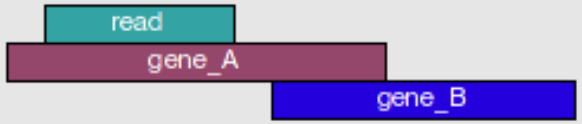
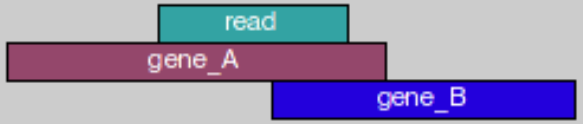
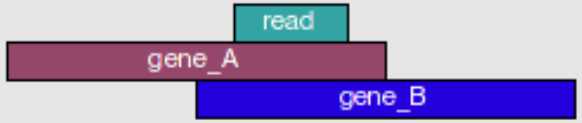
For short-reads:

count reads intersecting with features…

How to intersect ?
Options in ***htseq*** library :

http://htseq.readthedocs.io/en/master/count.html

What **features** to use ?
Transcripts, genes or exons …

	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

Units for read counts : raw counts + RPKM, FPKM, TPM

Historically used units

RPKM = Reads Per Kilobase of the feature (gene) per Million

FPKM = Fragments Per Kilobase of the feature (gene) per Million

$$R(F)PKM = \frac{\text{Number of reads (fragments) mapped for transcript} \times 10^3 \times 10^6}{\text{Transcript length} \times \text{Number of reads (fragments) mapped in sample}}$$

Currently recommended unit

TPM = Transcripts Per Million

$$TPM = \frac{\text{Number of fragments mapped for transcript} \times \text{Average fragment length} \times 10^6}{\text{Length of transcript} \times \text{Number of transcripts in sample}}$$

Accounts for transcript and library sizes

https://doi.org/10.1007/s12064-012-0162-3

Units for read counts : raw counts + RPKM, FPKM, TPM

Raw counts are needed for Differential Expression analysis !

Historically used units

RPKM = Reads Per Kilobase of the feature (gene) per Million

FPKM = Fragments Per Kilobase of the feature (gene) per Million

$$R(F)PKM = \frac{\text{Number of reads (fragments) mapped for transcript} \times 10^3 \times 10^6}{\text{Transcript length} \times \text{Number of reads (fragments) mapped in sample}}$$

Currently recommended unit

TPM = Transcripts Per Million

$$TPM = \frac{\text{Number of fragments mapped for transcript} \times \text{Average fragment length} \times 10^6}{\text{Length of transcript} \times \text{Number of transcripts in sample}}$$

Accounts for transcript and library sizes

https://doi.org/10.1007/s12064-012-0162-3

Different tools provide counts in different formats

HTSeq

Separate text file for each sample

example.htseq.counts

```
ENSG00000000003.13      2569
ENSG00000000005.5       1
ENSG00000000419.11      3180
ENSG00000000457.12      3332
ENSG00000000460.15      1621
ENSG00000000938.11      530
ENSG00000000971.14      7282
ENSG00000001036.12      3312
ENSG00000001084.9       2642
ENSG00000001167.13      3322
   ...
ENSGR0000270726.4       0
ENSGR0000275287.3       0
ENSGR0000276543.3       0
ENSGR0000277120.3       0
ENSGR0000280767.1       0
ENSGR0000281849.1       0
__no_feature    3069305
__ambiguous     3368739
__too_low_aQual 0
__not_aligned   0
__alignment_not_unique  23748640
```

Salmon

Separate text file for each sample in a folder with meta-information

- aux_info
- libParams
- logs
- cmd_info.json
- lib_format_counts.json
- quant.sf

quant.sf

Name	Length	EffectiveLength	TPM	NumReads
ENST00000456328.2	1657	1675.004	0.843814	17.896
ENST00000450305.2	632	477.000	0.000000	0.000
ENST00000488147.1	1351	1332.253	3.955642	66.727
ENST00000619216.1	68	11.000	0.000000	0.000
ENST00000473358.1	712	557.000	0.000000	0.000
ENST00000469289.1	535	381.000	0.000000	0.000
ENST00000607096.1	138	27.000	0.000000	0.000
ENST00000417324.1	1187	1032.000	0.000000	0.000
ENST00000461467.1	590	436.000	0.000000	0.000
ENST00000606857.1	840	685.000	0.000000	0.000
ENST00000642116.1	1414	1259.000	0.000000	0.000

...

(r)subreads featureCount

Matrix of counts with Genes in rows and Samples in columns

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Sources of Genes Annotations

GENCODE

HUMAN

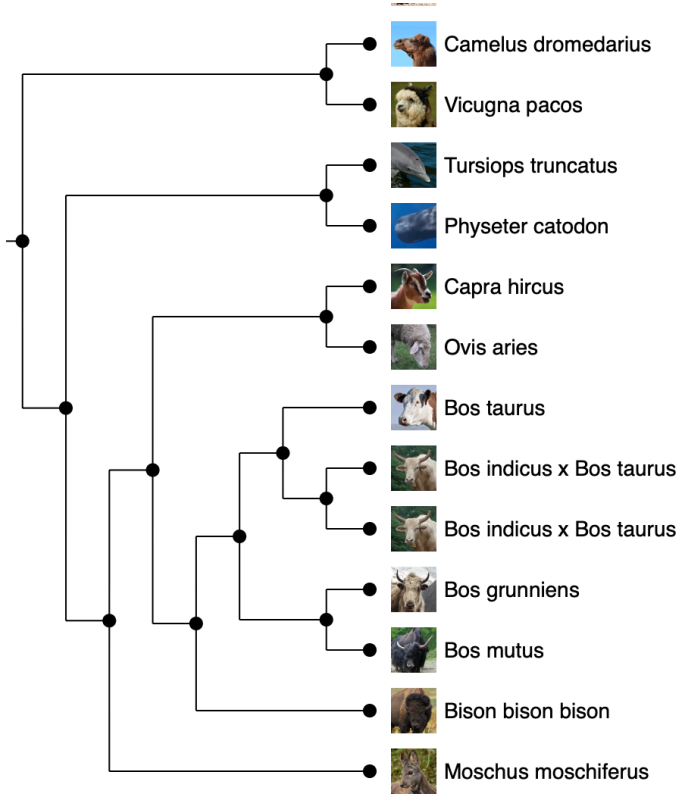
GENCODE 33 (16.01.20)

MOUSE

GENCODE M24 (16.01.20)

https://www.gencodegenes.org

e!Ensembl



https://www.ensembl.org

RefSeq: NCBI

March 6, 2020
RefSeq Release 99 is available for FTP

This release includes:

Proteins:	167,278,920
Transcripts:	29,869,155
Organisms:	99,842
Available at:	ftp://ftp.ncbi.nlm.nih.gov/refseq/release/
Documentation:	Release Notes

https://www.ncbi.nlm.nih.gov/refseq/

GFF / GTF: file format for genes annotations

https://www.ensembl.org/info/website/upload/gff.html

http://genome.ucsc.edu/FAQ/FAQformat.html#format3

GFF format

GFF (General Feature Format) lines are based on the Sanger **GFF2 specification**. GFF lines have nine required fields that *must* be tab-separated. If the fields are separated by spaces instead of tabs, the track will not display correctly. For more information on GFF format, refer to Sanger's **GFF page**.

gencode.v22.annotation.gtf - /Users/alexey/OneDrive/Documents/Teaching/Lecturing/rna-seq-ebi-2020/rna-seq-ebi-2020-tutorial/resources - Geany

gencode.v22.annotation.gtf

```
##description: evidence-based annotation of the human genome (GRCh38), version 22 (Ensembl 79)
##provider: GENCODE
##contact: gencode@sanger.ac.uk
##format: gtf
##date: 2015-03-06
chr1	HAVANA	gene	11869	14409	.	+	.	gene_id "ENSG00000223972.5"; gene_type "transcribed_
chr1	HAVANA	transcript	11869	14409	.	+	.	gene_id "ENSG00000223972.5"; transcript_id "ENST
chr1	HAVANA	exon	11869	12227	.	+	.	gene_id "ENSG00000223972.5"; transcript_id "ENST0000
chr1	HAVANA	exon	12613	12721	.	+	.	gene_id "ENSG00000223972.5"; transcript_id "ENST0000
chr1	HAVANA	exon	13221	14409	.	+	.	gene_id "ENSG00000223972.5"; transcript_id "ENST0000
chr1	HAVANA	transcript	12010	13670	.	+	.	gene_id "ENSG00000223972.5"; transcript_id "ENST
chr1	HAVANA	exon	12010	12057	.	+	.	gene_id "ENSG00000223972.5"; transcript_id "ENST0000
chr1	HAVANA	exon	12179	12227	.	+	.	gene_id "ENSG00000223972.5"; transcript_id "ENST0000
chr1	HAVANA	exon	12613	12697	.	+	.	gene_id "ENSG00000223972.5"; transcript_id "ENST0000
```

GTF files can be conveniently red into R data-frame using ***readGFF()*** function from **rtracklayer** package (see example in the practical session)

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Design formulas in R models

The "***design formulas***" are text strings, used to describe the required analysis for **edgeR** and **DESeq2**.
For simplicity, we will illustrate some "***design formulas***" using ***lm()*** function as an example.
Exactly the same "***formulas***" could be used for **edgeR** and **DESeq2**.

A simple design: Is the mean tumour size different between two groups ?

```r
# Toy data
size <- c(1.35, 1.42, 1.75, 1.14, 2.67, 2.54, 3.98, 2.27)
group <- c("A","A","A","A","B","B","B","B")

# Combine to a data.frame
experiment.df <- data.frame(size, group)
str(experiment.df)
```

```
'data.frame':   8 obs. of  2 variables:
 $ size : num  1.35 1.42 1.75 1.14 2.67 2.54 3.98 2.27
 $ group: Factor w/ 2 levels "A","B": 1 1 1 1 2 2 2 2
```

	size	group
1	1.35	A
2	1.42	A
3	1.75	A
4	1.14	A
5	2.67	B
6	2.54	B
7	3.98	B
8	2.27	B

Note that the "group" is a Factor with base level "A".
This level will be used for the intercept (also called "reference" level).

Is a mean tumour size different between two groups ?

Fit a linear model using ” formula” to specify the model’s “design”

Intercept =
the mean size of tumours
in **Group A**

Design formula

size ~ group

The **Intercept** = size of tumours
In **Group A** is significantly
different from 0

```
> fit <- lm(size ~ group, data=experiment.df)
> coefficients(summary(fit))
            Estimate Std. Error t value    Pr(>|t|)
(Intercept)    1.415  0.2837986 4.98593 0.002486972
groupB         1.450  0.4013519 3.61279 0.011193446
```

The difference between
the mean size of tumours
in **Group B** and the **intercept**

The difference between
Group B and the **intercept**
is significantly different from 0

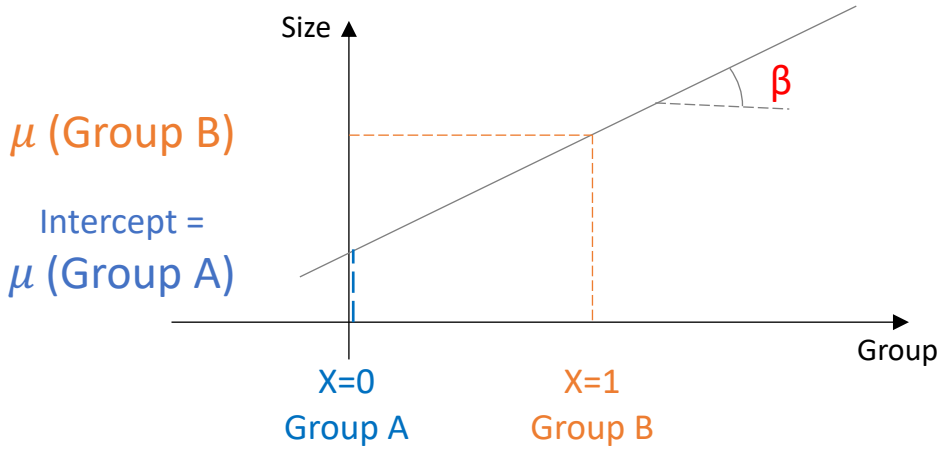
```
> fit <- lm(size ~ group, data=experiment.df)
> coefficients(summary(fit))
            Estimate Std. Error t value    Pr(>|t|)
(Intercept)    1.415  0.2837986 4.98593 0.002486972
groupB         1.450  0.4013519 3.61279 0.011193446
```

Behind the scene

1) Coding the groups as 0 and 1 and (making "design matrix")
2) Fitting an Intercept (default: the group with the base level in the factor)
3) Calculating the slope (β)

```
> model.matrix(~ group, data=experiment.df)
  (Intercept) groupB
1           1      0
2           1      0
3           1      0
4           1      0
5           1      1
6           1      1
7           1      1
8           1      1
```

Size = Intercept + β * Group



Setting Intercept in R linear models

Default:
implicit modeling with intercept
"Y ~ Group" in fact means *"Y ~ Intercept + Group"*

```
> fit <- lm(size ~ group, data=experiment.df)
> coefficients(summary(fit))
            Estimate Std. Error t value    Pr(>|t|)
(Intercept)    1.415  0.2837986 4.98593 0.002486972
groupB         1.450  0.4013519 3.61279 0.011193446
```

```
> model.matrix(~ group, data=experiment.df)
  (Intercept) groupB
1           1      0
2           1      0
3           1      0
4           1      0
5           1      1
6           1      1
7           1      1
8           1      1
```

Overriding the default:
explicit modeling without intercept
Y ~ 0 + Group

```
> fit <- lm(size ~ 0 + group, data=experiment.df)
> coefficients(summary(fit))
       Estimate Std. Error  t value     Pr(>|t|)
groupA    1.415  0.2837986  4.98593 2.486972e-03
groupB    2.865  0.2837986 10.09519 5.487252e-05
```

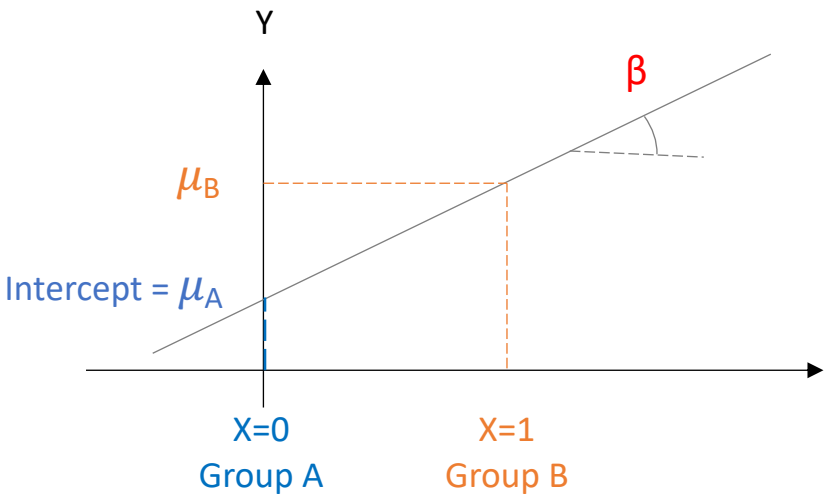
```
> model.matrix(~ 0 + group, data=experiment.df)
  groupA groupB
1      1      0
2      1      0
3      1      0
4      1      0
5      0      1
6      0      1
7      0      1
8      0      1
```

Setting Intercept in R linear models

Understand what you are asking for ...

Don't override the defaults without asking a statistician ….

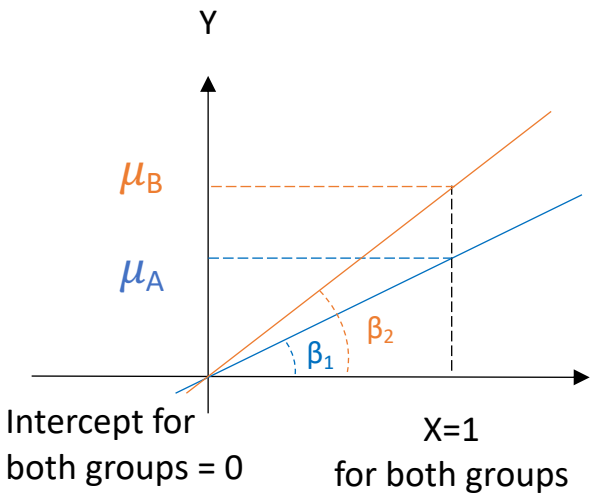
Default:
implicit modeling with intercept
"Y ~ Group" in fact means *"Y ~ Intercept + Group"*



Significance of $\beta \neq 0$

Look at the difference of mean values
between Group A and Group B

Overriding the default:
explicit modeling without intercept
Y ~ 0 + Group



Significance of $\beta 1 \neq 0$ or $\beta 2 \neq 0$

Hmm…
This is interesting …
But this is **NOT** what we
have been looking for !

Look at means in both groups:
whether each is different from zero

Design with a confounder (batch effect)

What if we measured the tumour sizes over two days, and on the second day we had a different calibration, so all the measurements on the second day somehow went higher?

```r
# Toy data
size <- c(1.42, 1.35, 1.95, 1.87, 2.67, 2.54, 3.98, 3.27)
group <- c("A","A","B","B","A","A","B","B")
day <- c("Mon","Mon","Mon","Mon","Tue","Tue","Tue","Tue")

# Combine to a data.frame
experiment.df <- data.frame(size, group, day)
str(experiment.df)
```

```
'data.frame':   8 obs. of  3 variables:
 $ size : num  1.42 1.35 1.95 1.87 2.67 2.54 3.98 3.27
 $ group: Factor w/ 2 levels "A","B": 1 1 2 2 1 1 2 2
 $ day  : Factor w/ 2 levels "Mon","Tue": 1 1 1 1 2 2 2 2
```

	size	group	day
1	1.42	A	Mon
2	1.35	A	Mon
3	1.95	B	Mon
4	1.87	B	Mon
5	2.67	A	Tue
6	2.54	A	Tue
7	3.98	B	Tue
8	3.27	B	Tue

We can see clearly the differences between groups within each day …
However, how to combine the data over both days (batches) ?

Including batch correction in the design formula

Formula without batch correction

size ~ group

No significant association between **Group** and **Size**

```
> fit <- lm(size ~ group, data=experiment.df)
> coefficients(summary(fit))
             Estimate Std. Error t value   Pr(>|t|)
(Intercept)    1.9950  0.4423317 4.51019 0.00405967
groupB         0.7725  0.6255514 1.23491 0.26302689
```

```
> model.matrix(~ group, data=experiment.df)
  (Intercept) groupB
1           1      0
2           1      0
3           1      1
4           1      1
5           1      0
6           1      0
7           1      1
8           1      1
```

Formula with batch correction

size ~ day + group

Detected an association between **Group** and **Size**

```
> fit <- lm(size ~ day + group, data=experiment.df)
> coefficients(summary(fit))
             Estimate Std. Error  t value     Pr(>|t|)
(Intercept)   1.26125  0.1707328 7.387274 0.0007148004
dayTue        1.46750  0.1971453 7.443750 0.0006899725
groupB        0.77250  0.1971453 3.918431 0.0111996428
```

```
> model.matrix(~ day + group, data=experiment.df)
  (Intercept) dayTue groupB
1           1      0      0
2           1      0      0
3           1      0      1
4           1      0      1
5           1      1      0
6           1      1      0
7           1      1      1
8           1      1      1
```

Including covariates / confounders in the design formula

The previous example in more detail

Predictors
(independent variables)

Response
(dependent variable)

size ~ day + group

Confounder

The factor
of intertest

By convention, the factor of interest is placed at the end of the formula

Other examples

Is a treatment significantly associated with change in blood pressure, controlling for age and ethnicity?

blood_pressure_change ~ age + ethnicity + treatment

Is a gene are differentially expressed in different types of tumour, controlling for batch and sex?

tumour_type ~ batch + sex + gene

Design with an interaction

May one factor change the response to another ?

Consider an imaginary experiment:
We start treatment of breast tumours in mice when they reached a certain size (e.g. 1cm).
Then we record the size after one week of treatment, and information about the type of tumour (ER-pos or ER-neg) and about the type of treatment (Tamoxifen or Placebo).

```r
# Make a toy dataset
size <- c(0.35, 0.42, 1.3, 1.4, 1.3, 1.2, 1.1, 1.4)
er <- c("Pos","Pos","Pos","Pos","Neg","Neg","Neg","Neg")
tamoxifen <- c("Yes","Yes","No","No","Yes","Yes","No","No")

# Combine to a data.frame
experiment.df <- data.frame(size,er,tamoxifen)
str(experiment.df)
```

```
'data.frame':   8 obs. of  3 variables:
 $ size     : num  0.35 0.42 1.3 1.4 1.3 1.2 1.1 1.4
 $ er       : Factor w/ 2 levels "Neg","Pos": 2 2 2 2 1 1 1 1
 $ tamoxifen: Factor w/ 2 levels "No","Yes": 2 2 1 1 2 2 1 1
```

	size	er	tamoxifen
1	0.35	Pos	Yes
2	0.42	Pos	Yes
3	1.30	Pos	No
4	1.40	Pos	No
5	1.30	Neg	Yes
6	1.20	Neg	Yes
7	1.10	Neg	No
8	1.40	Neg	No

We expect that ER status modifies the response to tamoxifen:

- ER-positive tumours should shrink upon Tamoxifen treatment while
- ER-negative tumours should keep growing

Including interaction in the design formula

Formula without interaction

size ~ er + tamoxifen

No significant **Size** change associated with **Tamoxifen** or **ER** alone

```
> fit <- lm(size ~ er + tamoxifen, data=experiment.df)
> coefficients(summary(fit))
              Estimate Std. Error   t value     Pr(>|t|)
(Intercept)    1.49125  0.1980649  7.529099 0.0006543625
erPos         -0.38250  0.2287056 -1.672456 0.1552945113
tamoxifenYes  -0.48250  0.2287056 -2.109699 0.0886557090
```

```
> model.matrix(~ er + tamoxifen, data=experiment.df)
  (Intercept) erPos tamoxifenYes
1           1     1            1
2           1     1            1
3           1     1            0
4           1     1            0
5           1     0            1
6           1     0            1
7           1     0            0
8           1     0            0
```

Formula with interaction

size ~ er * tamoxifen

Significant **Size** change because of interaction between **Tamoxifen** and **ER**

```
> fit <- lm(size ~ er * tamoxifen, data=experiment.df)
> coefficients(summary(fit))
                    Estimate Std. Error    t value     Pr(>|t|)
(Intercept)            1.250 0.08474226 14.7506103 0.0001229477
erPos                  0.100 0.11984365  0.8344205 0.4509786113
tamoxifenYes           0.000 0.11984365  0.0000000 1.0000000000
erPos:tamoxifenYes    -0.965 0.16948451 -5.6937356 0.0047004777
```

```
> model.matrix(~ er * tamoxifen, data=experiment.df)
  (Intercept) erPos tamoxifenYes erPos:tamoxifenYes
1           1     1            1                  1
2           1     1            1                  1
3           1     1            0                  0
4           1     1            0                  0
5           1     0            1                  0
6           1     0            1                  0
7           1     0            0                  0
8           1     0            0                  0
```

Another example of interpreting design formulas

RNA-seq was used to measure genes expression in three squamous cell carcinoma patients.
Tumour and paired Normal tissue was available for each patient.
Three types of model design was considered in the study:

Model	Interpretation	Genes detected
Patient	Baseline patient differences	
Patient + tissue	Consistent tumour differences	1276
Patient × tissue	Patient-specific tumour differences	202

McCarthy et al, 2012: Differential expression analysis of multifactor RNA-Seq experiments …

https://academic.oup.com/nar/article/40/10/4288/2411520

The design formula summary

- At the left side of the formula: The “Response” variable = “Dependent” variable (e.g. **Size** in the above examples)
- At the right side of the formula: “Predictors” = “Independent” variables (e.g. **Day**, **Group** etc in some above examples)
- The “predictors” include:
  - The variable of interest (e.g. **Group** in some above examples)
  - Confounders / covariates, which effect should be “controlled for” (e.g. **Day** in the batch effect example)
- By convention, most of the packages expect the variable of interest at the end of the formula
- Be aware about the reference level within the variable of interest (by default, the “base” level in the factor)
- Be aware about complex designs, such as designs with interactions, designs without intercepts.
  Other complex analyses may use user-defined contrasts etc. Ask a statistician if you consider using complex designs.

For simplicity, we illustrated design formulas using ***lm()*** function. However, many other R functions and packages, including ***glm()***, **edgeR** and **DESeq2** use the same way of specifying the design of analysis.

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

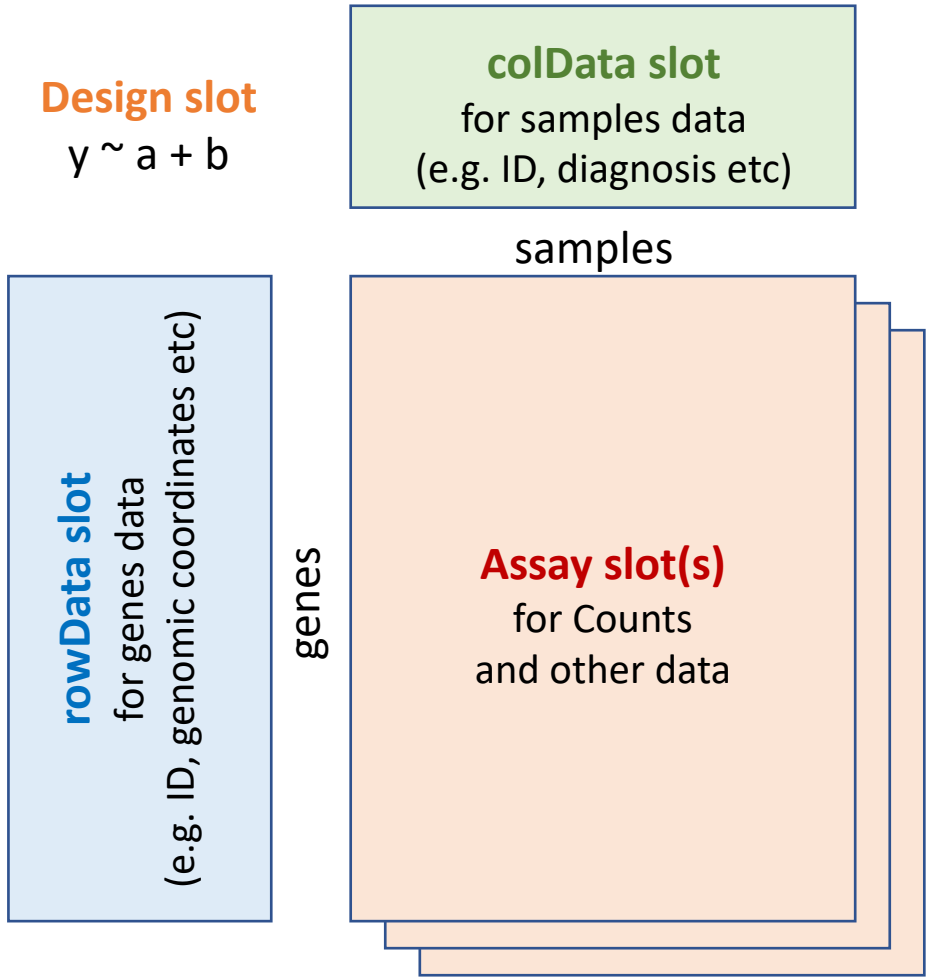
MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Internal representation of data in DESeq2 and edgeR

DESeq2 Data-Set



A modified ***Summarized Experiment***

edgeR DGEList

A list containing 3 synchronized objects:

- **Samples** data frame
- **Genes** data frame
- Counts matrix

Although is not implemented as ***Summarized Experiment*** provides similar functionality

https://bioconductor.org/packages/release/bioc/vignettes/SummarizedExperiment/inst/doc/SummarizedExperiment.html

Import functions and packages

readDGE function in edgeR

Makes a single matrix from multiple **text files of arbitrary format**, as long as the file contains columns with genes names and counts. Allows to add samples information etc. See example in the practical session.

https://master.bioconductor.org/packages/release/workflows/vignettes/RNAseq123/inst/doc/limmaWorkflow.html

DESeqDataSetFrom -Matrix, -HTSeqCount and *-Tximport* functions in DESeq2

Allow convenient import from matrix (**Rsubread**), **HTSeq** counts and from **Tximport** package respectively.

https://www.bioconductor.org/packages/devel/workflows/vignettes/rnaseqGene/inst/doc/rnaseqGene.html

Tximport package

Facilitates import from **Salmon**, **Kallisto** and some other tools to **DESeq2** or **edgeR**. May summarize transcripts to genes.

https://bioconductor.org/packages/devel/bioc/vignettes/tximport/inst/doc/tximport.html

Tximeta package

Currently facilitates import from **Salmon** to **DESeq2**, May summarize transcripts to genes.
Adds meta-data, including genes annotation in Genomic Ranges format.

https://bioconductor.org/packages/release/bioc/vignettes/tximeta/inst/doc/tximeta.html

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

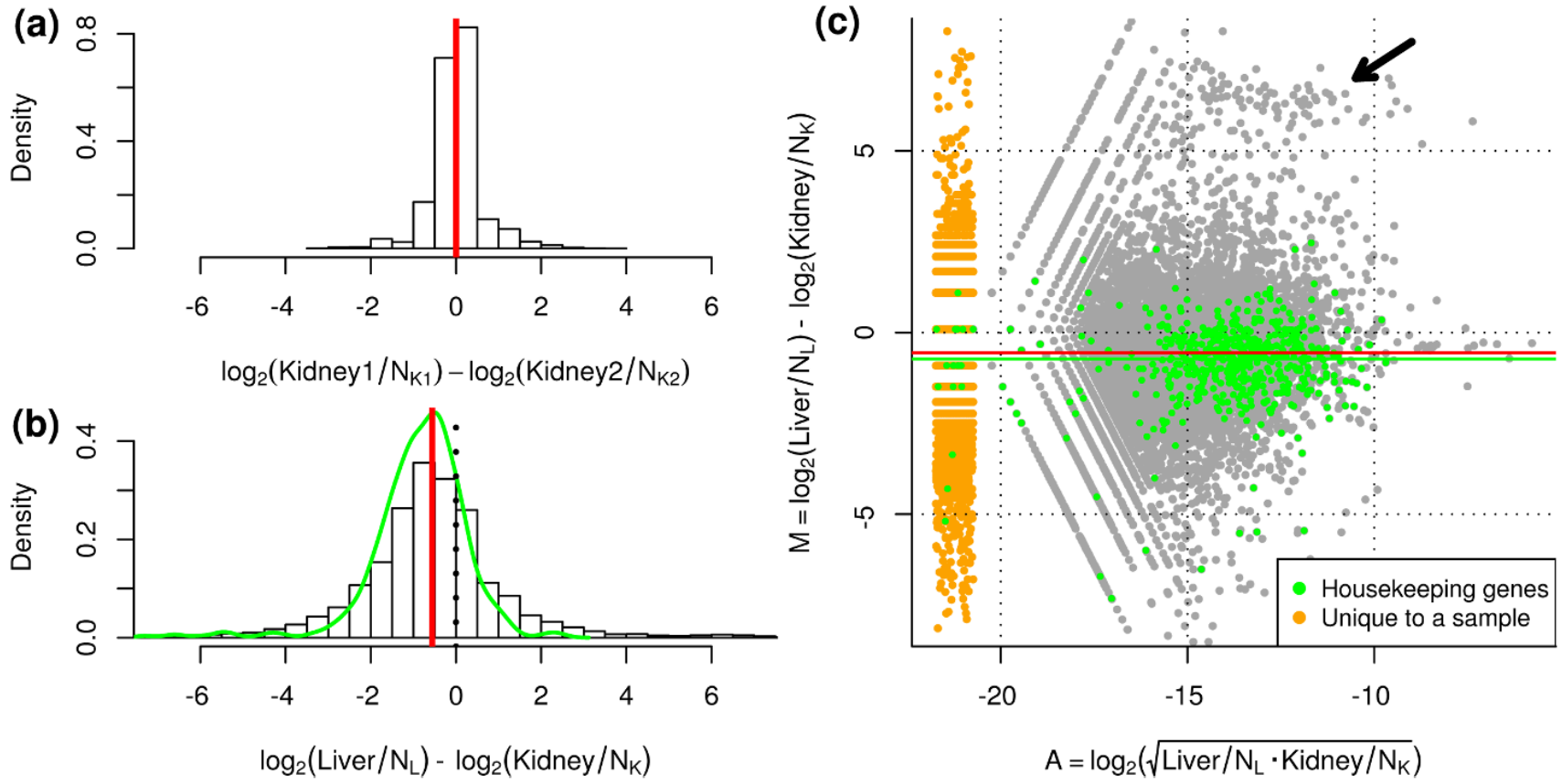
MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Normalizing by library size

Initially, a naïve normalization was suggested by the library size itself. However, later it was observed that, with such naïve normalization, a strong over-expression of some genes might be mis-interpreted as down-regulation of all the other genes:



Robinson and Oshlack 2010: A scaling normalization method for differential expression analysis of RNA-seq data http://genomebiology.com/2010/11/3/R25

TMM and Median-of-Ratios

To avoid such undue influence, **edgeR** excludes extremely expressed and extremely changed genes when calculating the normalization factors. The method is called "trimmed mean of M values" (**TMM**)

Robinson and Oshlack 2010: A scaling normalization method for differential expression analysis of RNA-seq data

http://genomebiology.com/2010/11/3/R25

For the same purpose, to avoid the undue influence of extremely changed genes, **DESeq2** takes **Median of the Genes Ratios** to estimate the size factors. The statistical properties of the **median** negate the effect of extremely changed genes.

Anders and Huber 2010: Differential expression analysis for sequence count data

http://genomebiology.com/2010/11/10/R106

TMM and Median of the Gene Ratios do not normalize by gene length

In contrast to some "normalized" units used for for RNA-seq counts representation (FPKM or TPM), **TMM** or **Median of the Gene Ratios** do not account for the gene/transcript length.

This implicitly assumes that the gene/transcript length does not change between the studied conditions. Although such assumption looks reasonable for Differential **Genes** Expression analysis on short-read data, changes in size of used transcripts may be incorporated in the modelling later.

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Variance-stabilizing transformations to explore source data

It is a common practice to perform unsupervised clustering of samples using all (or most variable) genes before the Differential Gene Expression analysis. Such exploration may suggest gross outliers, to exclude from analysis. Also, it shows whether the studied groups are well separated in the gene expression space.

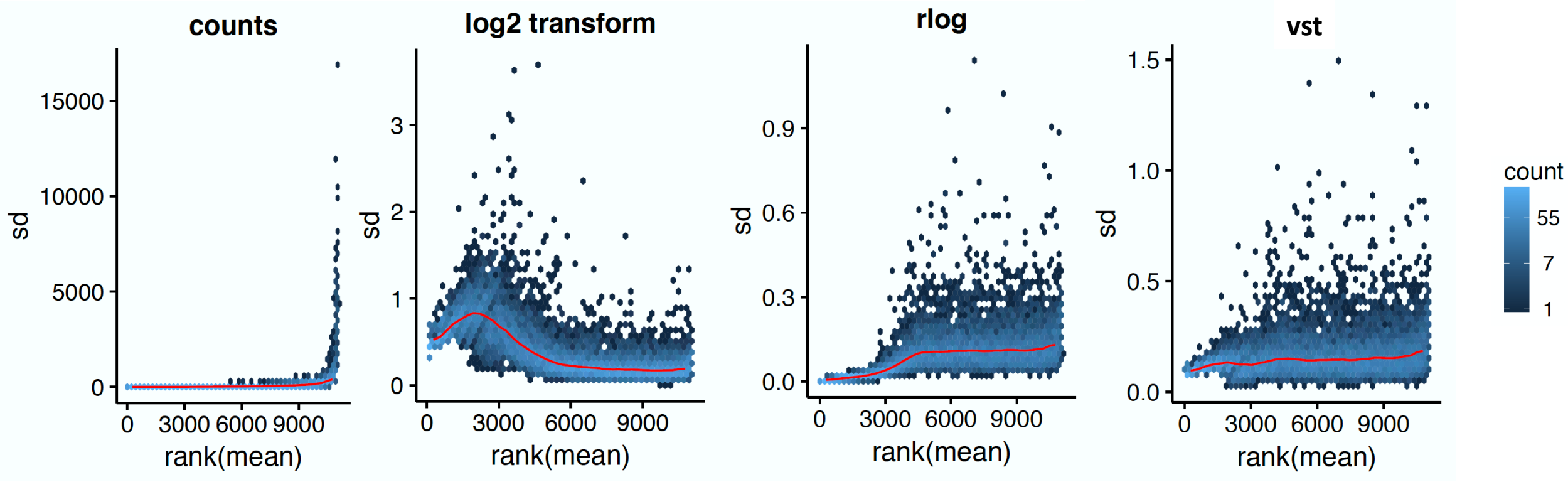
The most common methods for samples clustering include **Principal Component Analysis** (PCA) and **Hierarchical Clustering**. However, it has been empirically observed that variance in RNA-seq counts is higher in the highly-expressed genes (***heteroskedasticity*** of RNA-seq counts). Thus, **PCA** and **Hierarchical Clustering** might be dominated by the most expressed genes if the data are not transformed to make variance similar between the genes (make transformed data ***homoscedastic***).

Both **edgeR** and **DESeq2** provide transformations and plotting functions to facilitate the exploratory analysis:

	edgeR	DESeq2
Transformation(s)	Log(counts per million)	VST (Variance-Stabilizing transformation) rlog (regularized log-transformation
Plotting function	plotMDS*()	plotPCA()

* **MDS** stands for **Multi-Dimensional Scaling**: this is aprocedure very similar to PCA

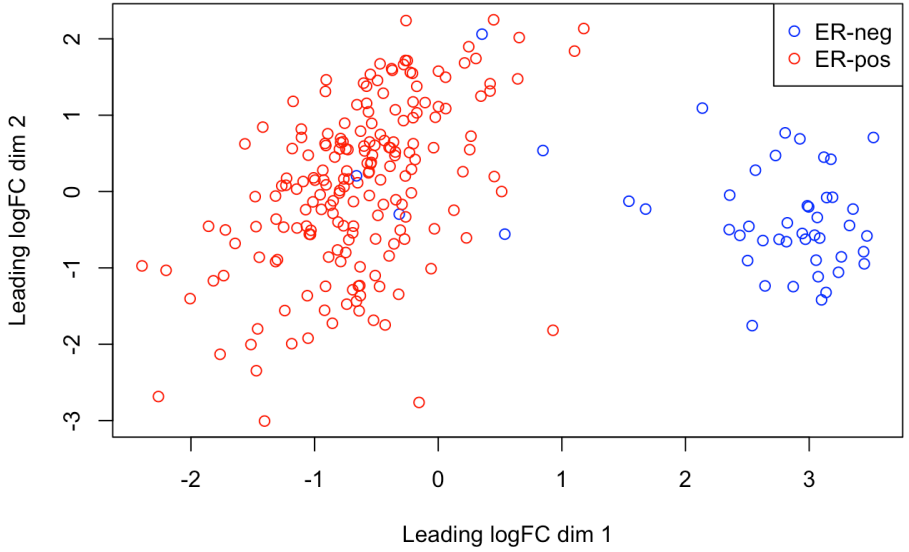
Variance-stabilizing transformations to explore source data



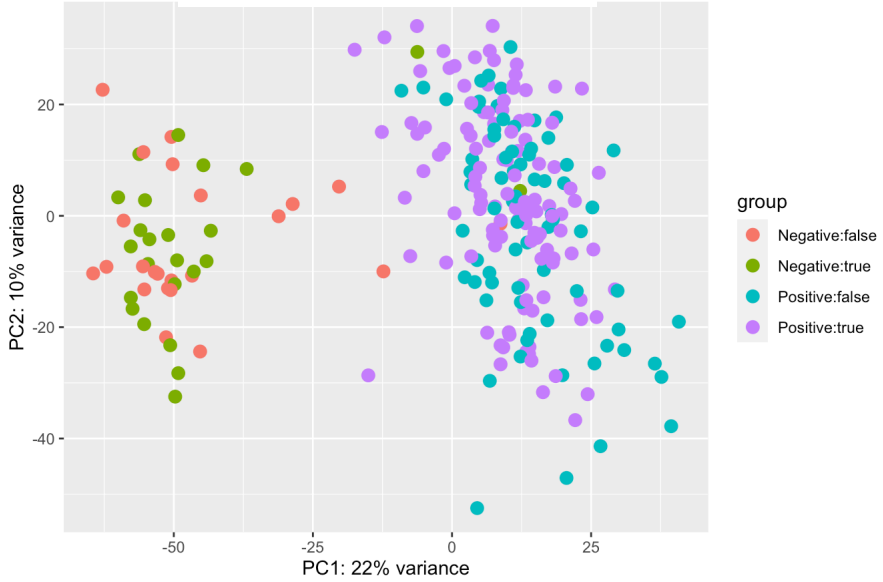
These transformations are used only to explore source data
They are NOT used during the Differential Expression Analysis

Examples of PCA and MDS plots

edgeR: MDS plot
(Log-cpm)



DESeq2: PCA plot
(vst)



These plots will be generated during the practical session (along with Hierarchical Clustering and Heatmap lots). The plots don't suggest any gross outliers to exclude. Also they suggest that some of the studied groups are clearly separated in the gene expression space.

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Dispersion estimation and adjustment

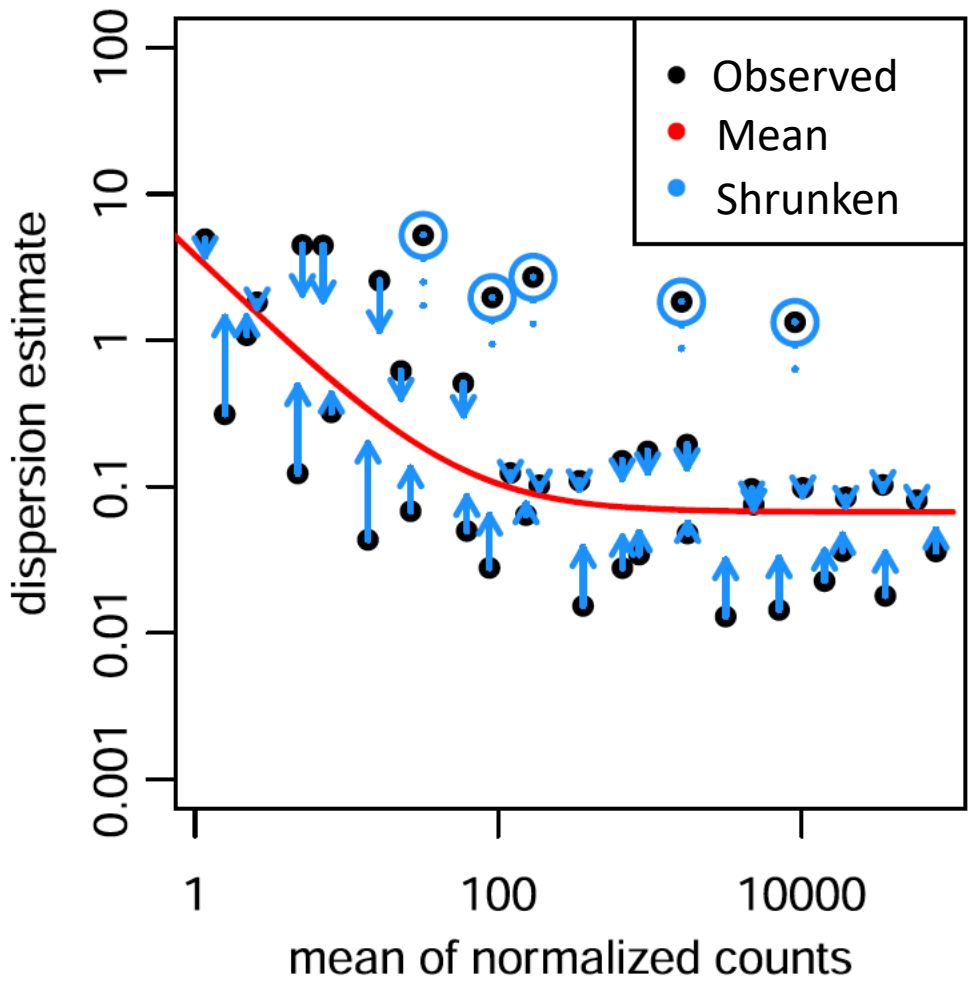
If dispersion for each single gene can not be accurately estimated because of a small number of samples (e.g. less than 10 replicates) then the data from other genes will be “borrowed”.

Simplified description of the procedure applied by DESeq2 :

1. **Observed** dispersions (●) are used to estimate **Mean** dispersions (●) for each level of expression.

2. Depending on the accuracy of the **Observed** dispersions they may be “**Shrunken**” (●) toward the **Mean** estimates. The more accurate is the observed dispersion, the less “shrinkage” will be applied.

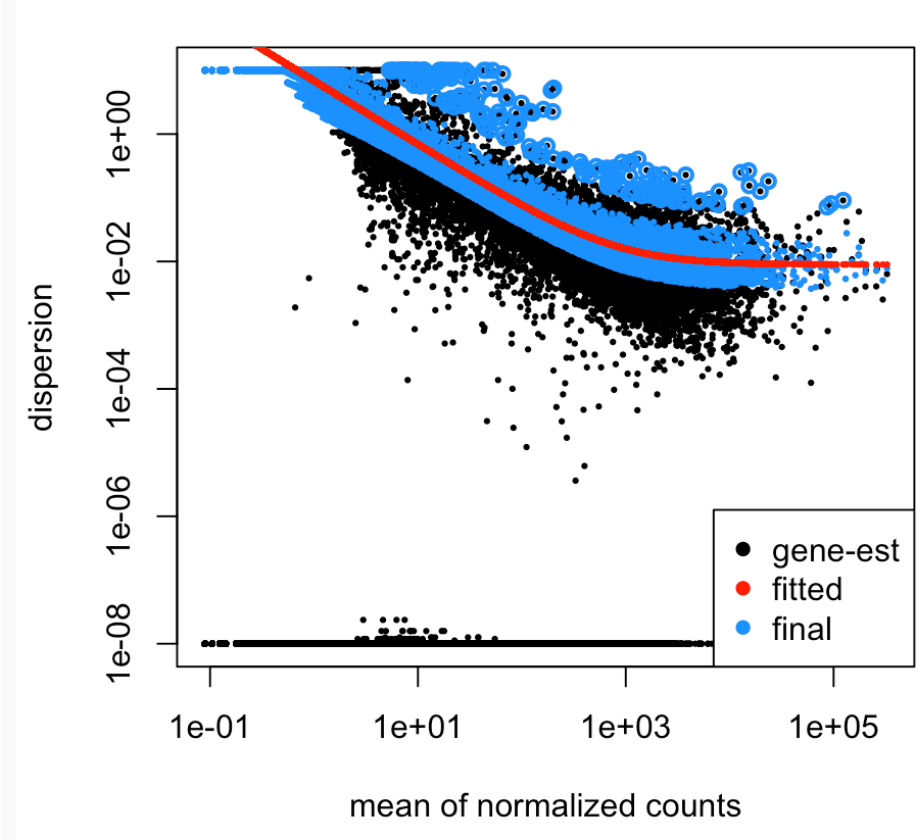
3. If the **Observed** dispersion extremely deviates from **Mean** (outliers encircled in blue) it does not shrunk.



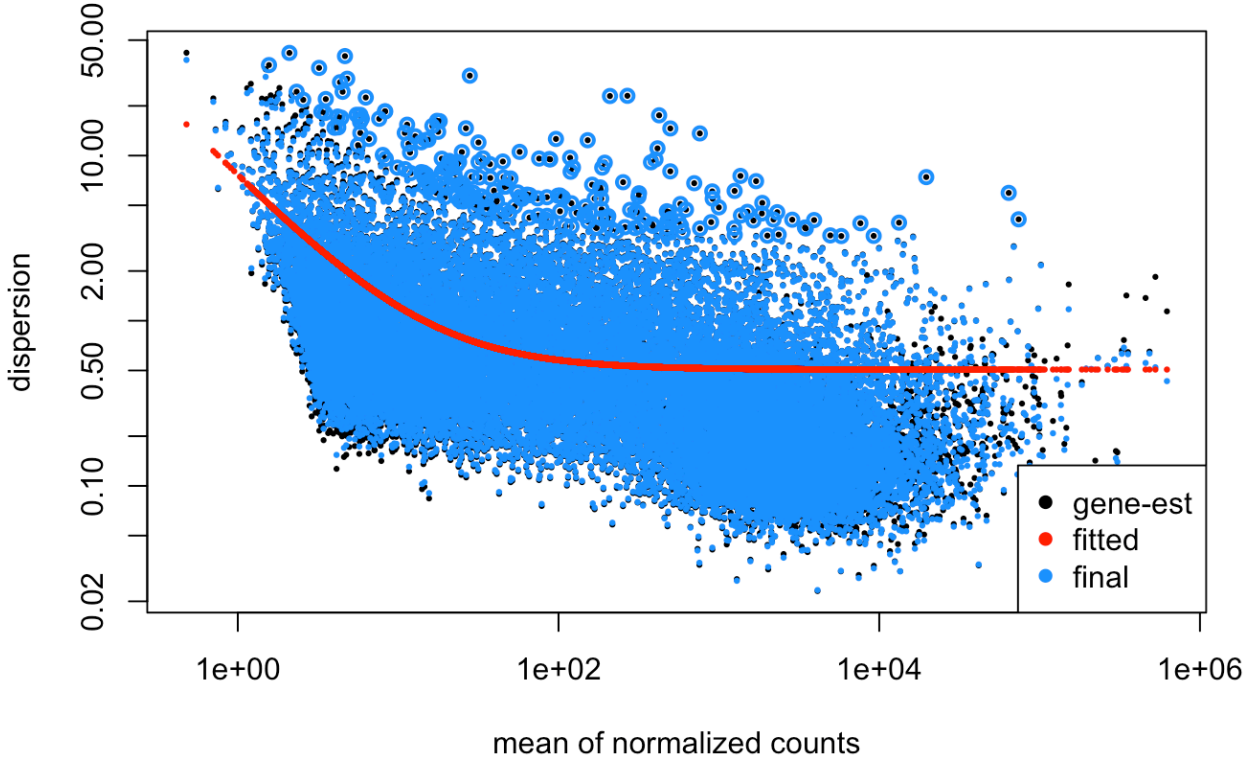
Love et al 2014: Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2

https://genomebiology.biomedcentral.com/articles/10.1186/s13059-014-0550-8

Dispersion estimation and adjustment



Dataset with small number of samples.
Many dispersions are shrunken toward the mean.



Dataset with large number of samples.
Most dispersions are not shrunken toward the mean.

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Default thresholds in DESeq2

It is a commonly accepted assumption that in most of experiments only a minority of genes change their expression. Thus, usually, the proportion of suggested Differentially Expressed Genes should not exceed 20%. However, in our practical session the default **DESeq2** settings would suggest that about 60% of all genes are differentially expressed … This is obviously an absurd result: the problem is in the default thresholds applied by DESeq2.

By default **DESeq2** performs testing for **any fold change at FDR < 0.1**.

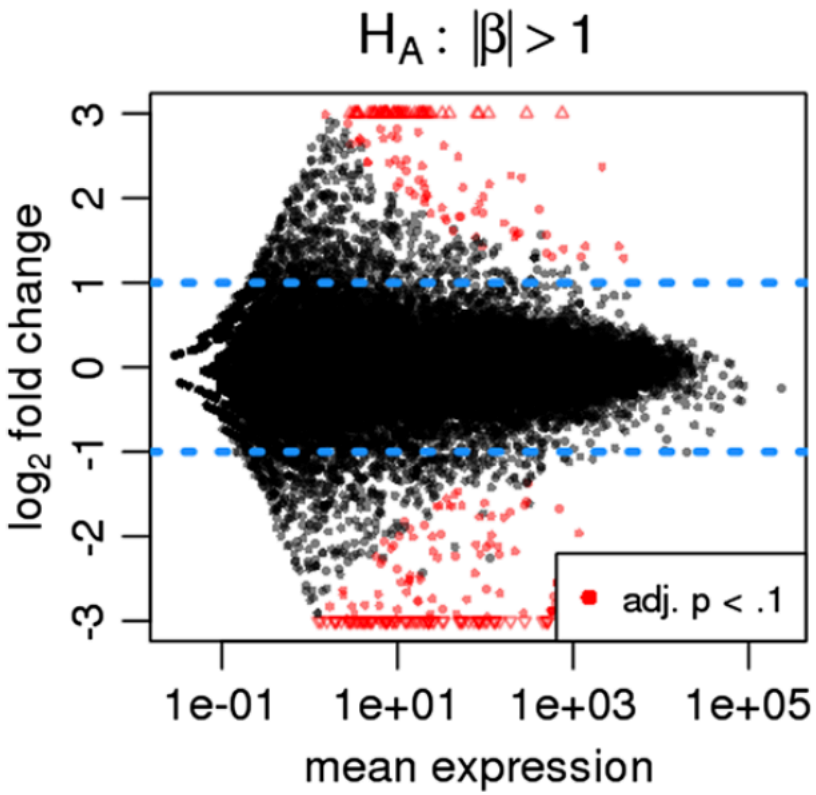
Such settings reflect the time, when RNA-seq was prohibitively expensive and experiments often included less than 10 samples. Our practical session analyses more than 200 samples. For such datasets, **DESeq2** allows to change the analysis thresholds. Thus, when we consider only genes with **at least 2-fold change at FDR < 0.01**, the proportion of suggested DEGs got below 10%. **edgeR** also allows testing for non-zero Fold-Change (see practical session for examples).

Importantly, testing against non-zero FC threshold is not the same as filtering by FC of the results obtained with default settings.

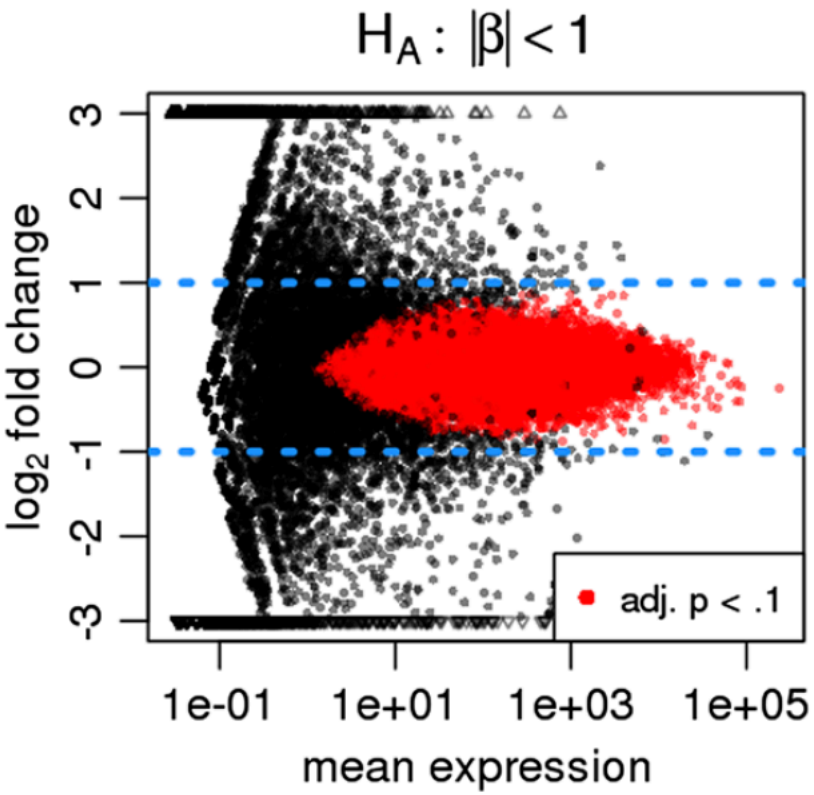
Testing against non-zero FC thresholds

MA-plots show Fold-Change (Y-axis) against Mean expression (X-axis) for individual genes; red shows significance

Genes with at least 2-fold change



Genes with less then 2-fold change



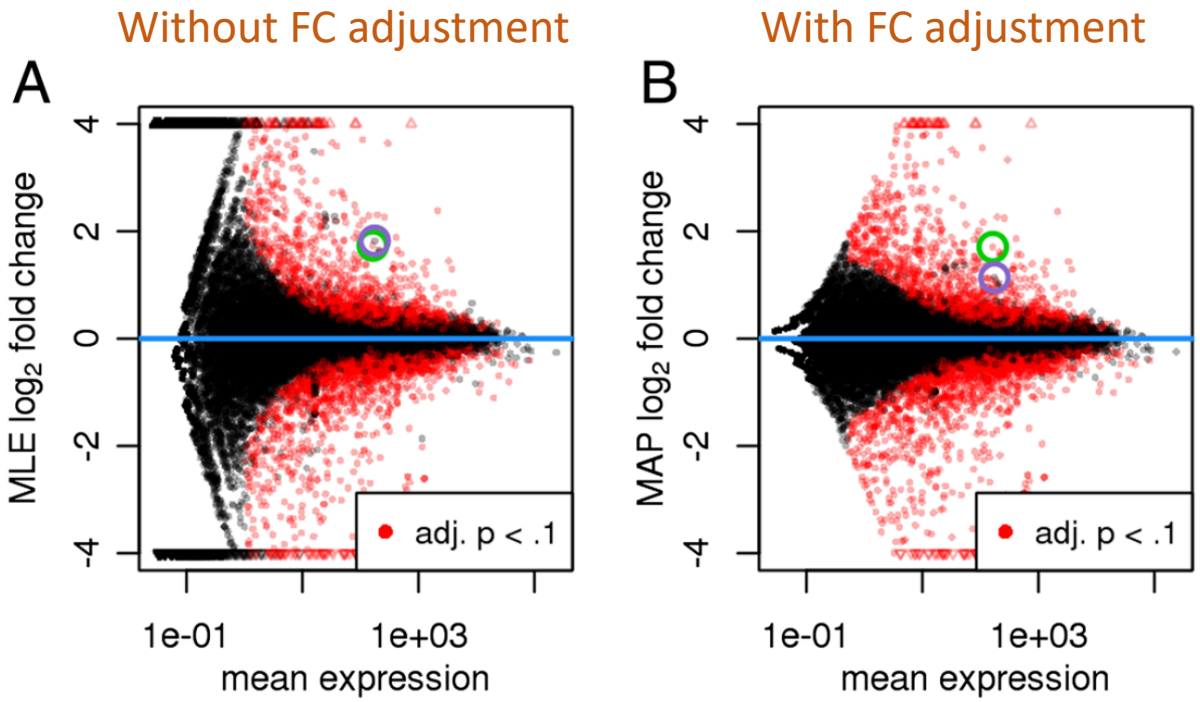
Love et al 2014: Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2

https://genomebiology.biomedcentral.com/articles/10.1186/s13059-014-0550-8

Additional features in DESeq2

Fold-Change (FC) adjustment in low-expressed genes

Noise may simulate high Fold-Changes in low-expressed genes. To avoid this artificially inflated "Changes" **DESeq2** uses an empirical algorithm that "shrinks" fold change in the genes with low expression.

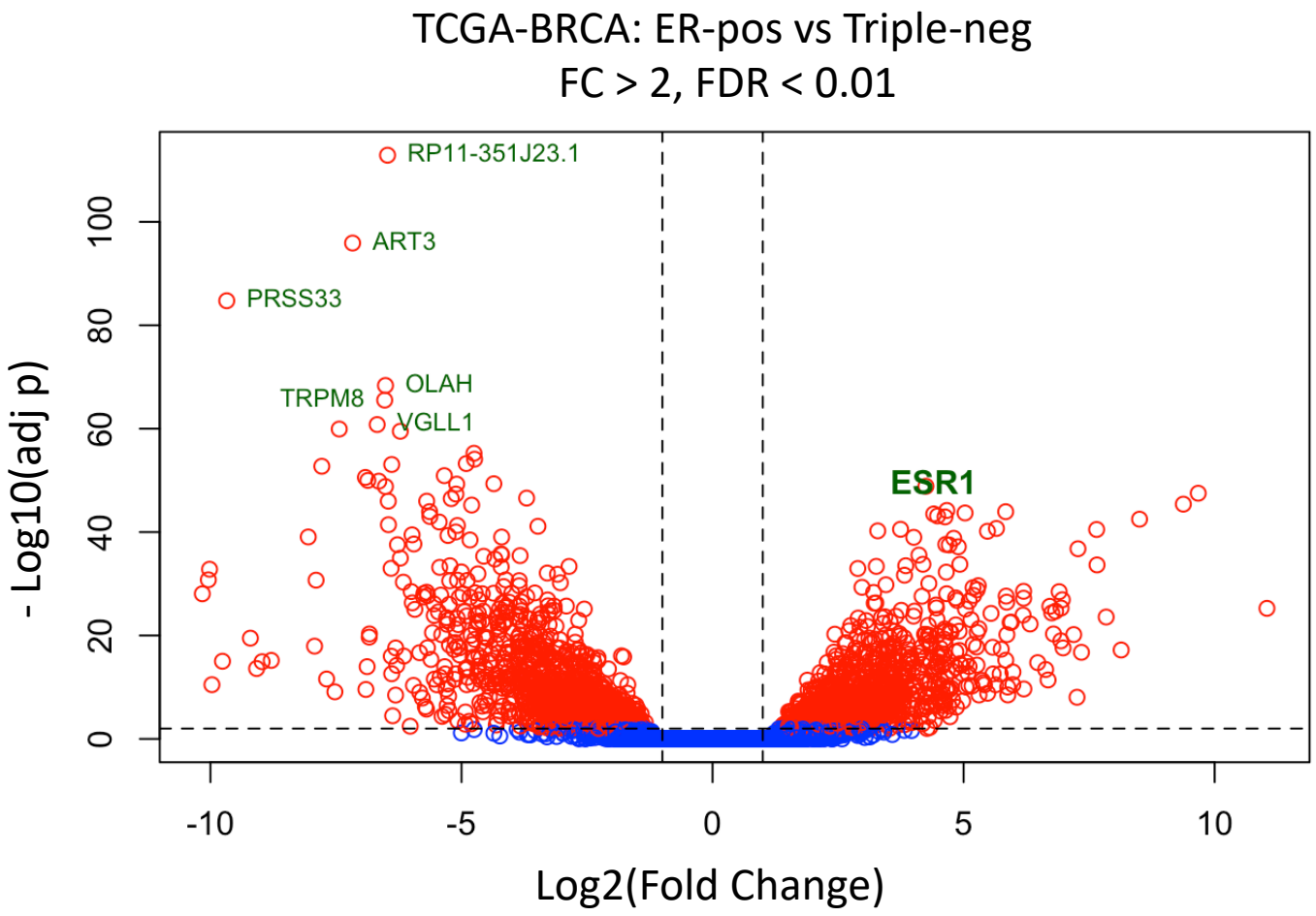


Removal of low-expressed genes before calculating FDR

DESeq2 automatically removes genes with low expression, applying adaptive threshold that maximises number of genes passing FDR. For the genes filtered out at this stage, NA is placed in the **p-adjusted** column.

Example of Volcano plot

Volcano-plot shows Significance (Y-axis) against Fold-Change (X-axis) for each gene.
Genes could be coloured according to FC and Significance thresholds



Result from the practical session

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial Distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

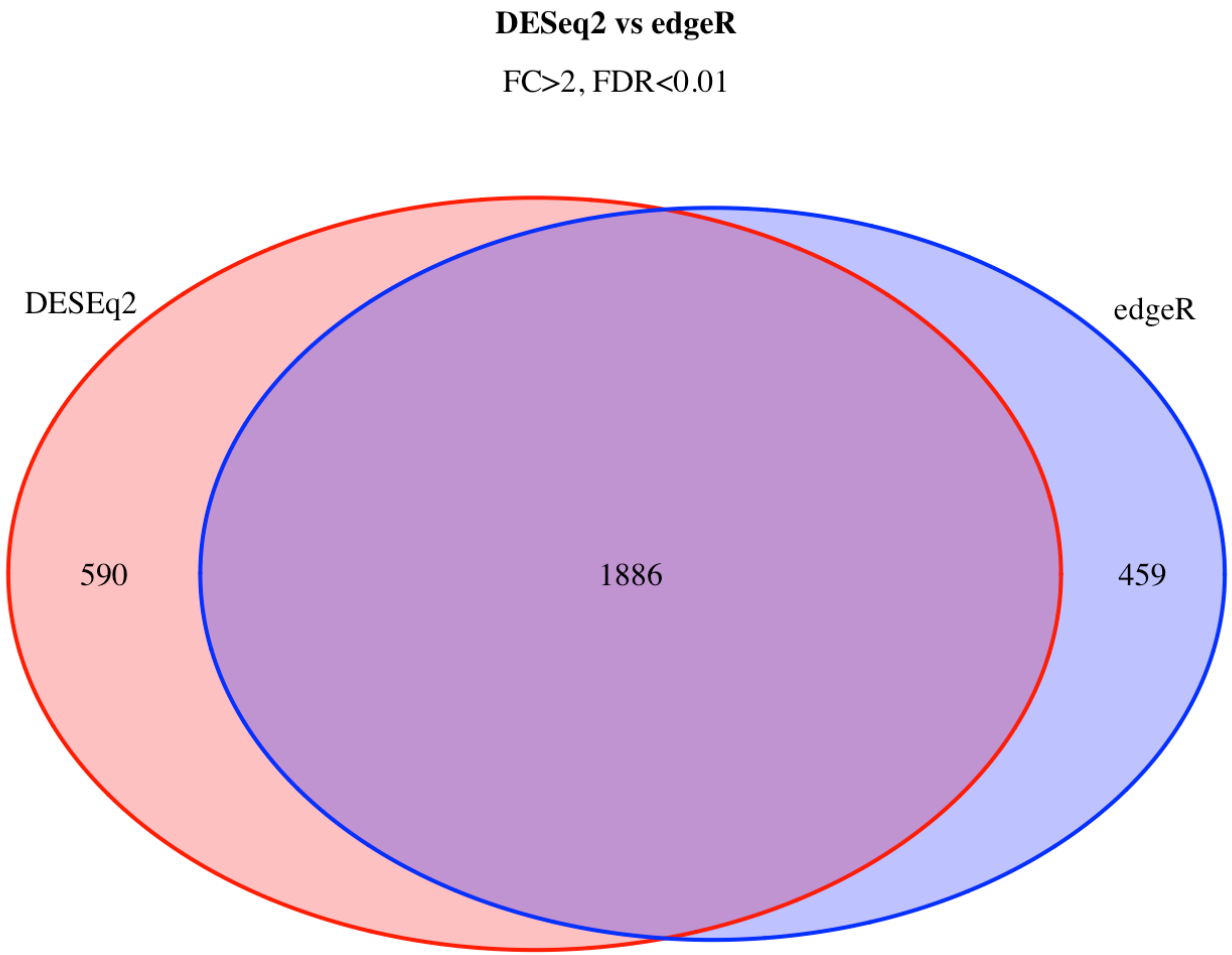
Statistical features of edgeR and DESeq2

Step	edgeR	DESeq2
Normalizing by library size	Trimmed Mean of M-values	Median of the Genes' Ratios
Distribution	Negative Binomial	Negative Binomial
Dispersion for a gene	An empirical custom procedure accounting for (i) dispersion over all genes, (ii) dispersion in the genes with similar expression and (iii) dispersion observed in the gene	An empirical custom procedure accounting for (i) dispersion in the genes with similar expression and (iii) dispersion observed in the gene
GLM	Log-link optimized algorithm for convergence	Log-link
Significance test	Log-likelihood ratio	Wald test
Multiple testing correction	FDR	FDR

Accessory features of edgeR and DESeq2

Step	edgeR	DESeq2
Internal data format	Customized list: DGEList	Modified Summarized Experiment
Data import	Data import function from a set of text files containing columns for **gene-ID** and **gene-Counts**	- Advanced import options are provided by **tximport** and **tximeta** packages - DESeq2 includes data import functions from different upstream tools (tximport, HTSeq, matrix)
Low-expressed genes	A function for filtering by low expression (applied before analysis)	- Automatic exclusion of low-expressed genes from multiple testing - Adjusting Fold Change for low-expressed genes
Testing against non-zero fold change (FC)	Yes	Yes, with an opportunity of testing for FC below or above the Threshold
Detecting counts outliers	No	Count outliers are detected and taken into account when calculating FC, p and when filtering genes
Data exploration functions	Log(counts per million) MDS plot	Variance-Stabilising transformation (VST) Regularized log-transformation (rlog) PCA plot

Example of DEGs detected by DESeq2 and edgeR



Result from the practical session

Differential Gene Expression

Introduction

Short and Long reads: Genes and Transcripts
Upstream and down-stream applications

Statistical summary

Recap of “standard” approaches
Problems with RNA-seq counts
Overdispersion: Negative Binomial distribution

Counts

Overview and software
Units: raw counts, (R)FPKM, TPM

Genes

GENCODE, Ensembl, Refseq
GTF file format

Design formula

A simple design
Accounting for covariates e.g. batch effect
Advanced designs: interactions

Data import

Summarized Experiment and DGEList
Data import packages and functions

Statistics in more details

Normalizing by library size
Exploring the source data
- Variance-stabilizing transformations
- PCA and Hierarchical clustering

Dispersion estimates
- Need for “borrowing” data
- Empirical adaptive estimates

Total number of DEGs and thresholds selection

Visualizing results

MA- and Volcano plots

DESeq2 vs edgeR

Comparison of methods
Comparison of results

Selected references

DESeq2

https://www.bioconductor.org/packages/devel/workflows/vignettes/rnaseqGene/inst/doc/rnaseqGene.html

Love *et al,* **2014**: Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2
https://genomebiology.biomedcentral.com/articles/10.1186/s13059-014-0550-8

Anders and Huber, **2010**: Differential expression analysis for sequence count data
http://genomebiology.com/2010/11/10/R106

edgeR

https://master.bioconductor.org/packages/release/workflows/vignettes/RNAseq123/inst/doc/limmaWorkflow.html

Robinson and Smyth, **2007**: Moderated statistical tests for assessing differences in tag abundance
https://academic.oup.com/bioinformatics/article/23/21/2881/372869

Robinson and Oshlack, **2010**: A scaling normalization method for differential expression analysis of RNA-seq data
http://genomebiology.com/2010/11/3/R25

McCarthy *et al*, **2012**: Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation
https://academic.oup.com/nar/article/40/10/4288/2411520

Practical session

- Using open-access RNA-seq data for several hundred samples from **TCGA dataset** the tutorial will provide step-by step instructions on how to detect genes differentially expressed between Estrogen-Receptor-positive and Triple-negative breast cancers.

- During the practical session you will
  - Import **HTSeq counts** to **DESeq2** and **edgeR** data formats
  - Add information about the **samples** and **genes**
  - Remove consistently low-expressed **genes**
  - Perform **normalization** to account for library sizes
  - Explore source data using **PCA** and **MDS** plots, perform **Hierarchical Clustering** and make **Heatmap** plot to show clustering of samples and genes
  - Identify Differentially Expressed Genes with at least **2-fold difference at FDR < 0.01**
  - Explore plots of the **dispersion estimates and adjustments**
  - Explore **MA- and Volcano plots** for the Differentially Expressed Genes

- The equivalent analyses will be performed by **DESeq2** and **edgeR**, and the results will be compared between these packages