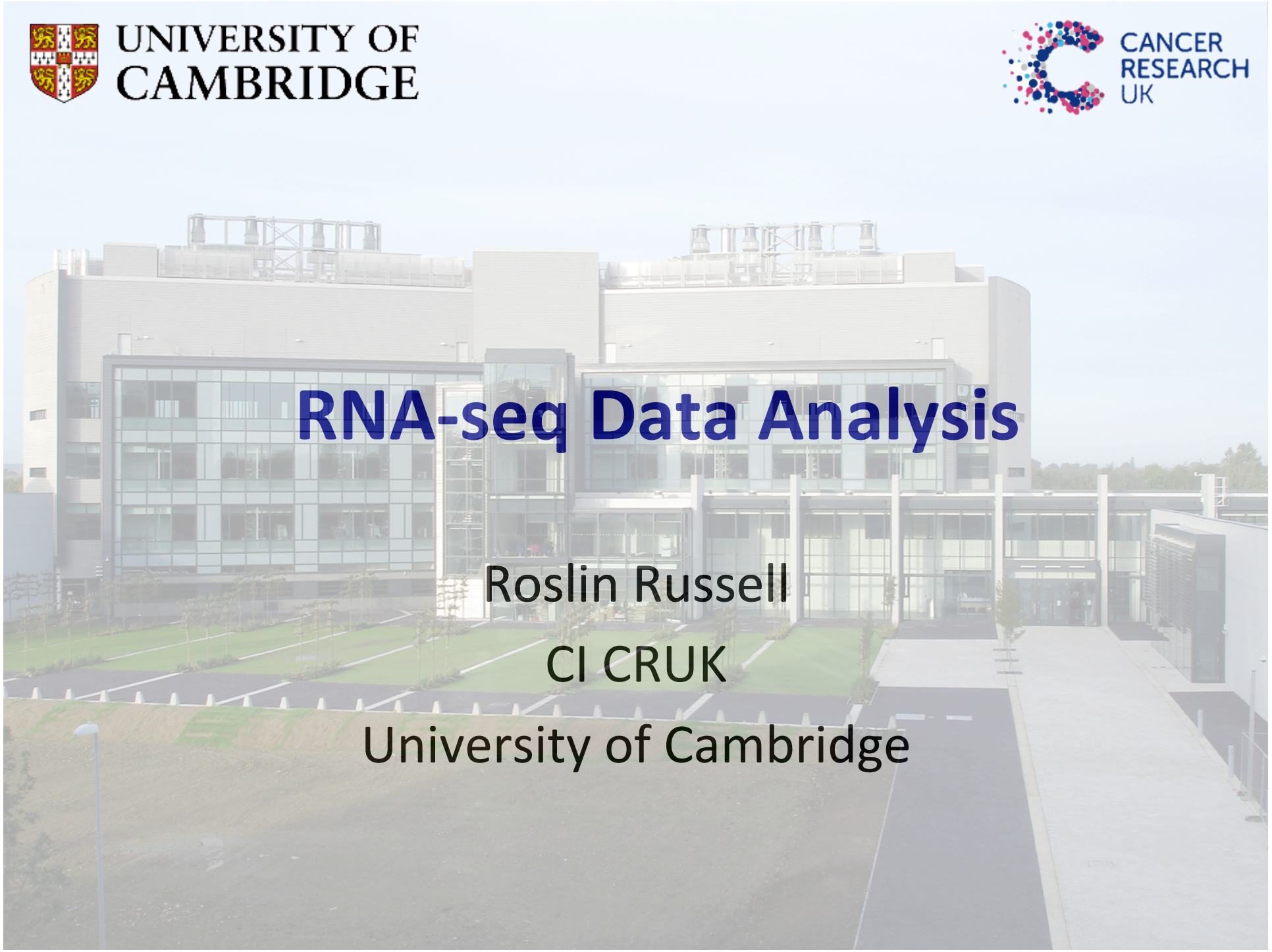


UNIVERSITY OF CAMBRIDGE

CANCER RESEARCH UK

RNA-seq Data Analysis

Roslin Russell

CI CRUK

University of Cambridge

Overview

- Applications
- Experimental Design Considerations
- RNA-seq Data Analysis Workflows
- What does the data look like?

- Pre-processing: aligning & mapping
- Feature Counting
- Normalization: ensuring data comparability
- Variance Estimation & Shrinkage

- Testing for Differential Expression
- Enrichment Analysis
- Data Clustering

Applications

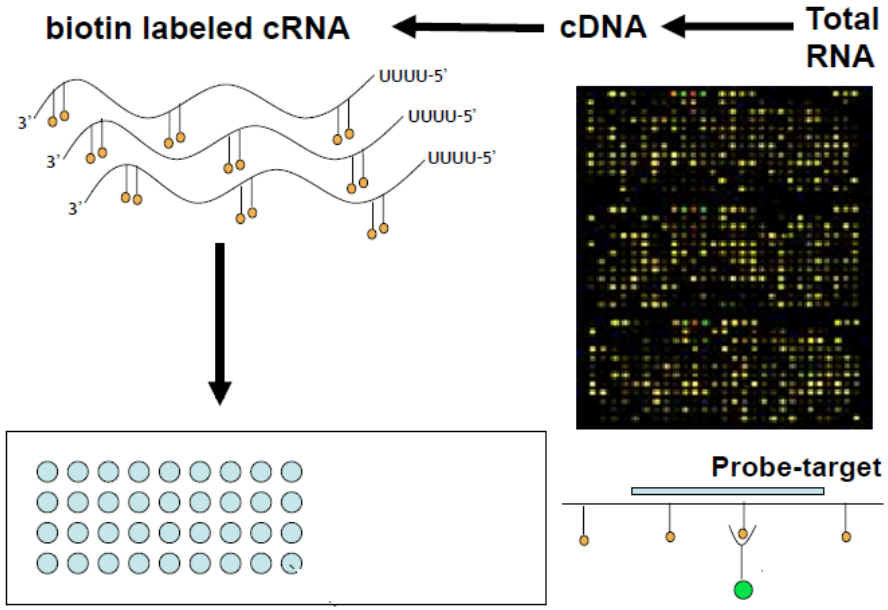
Two Main Groups of Applications

- **Discovery**
  - Find new transcripts
  - Find transcript boundaries
  - Find splice junctions

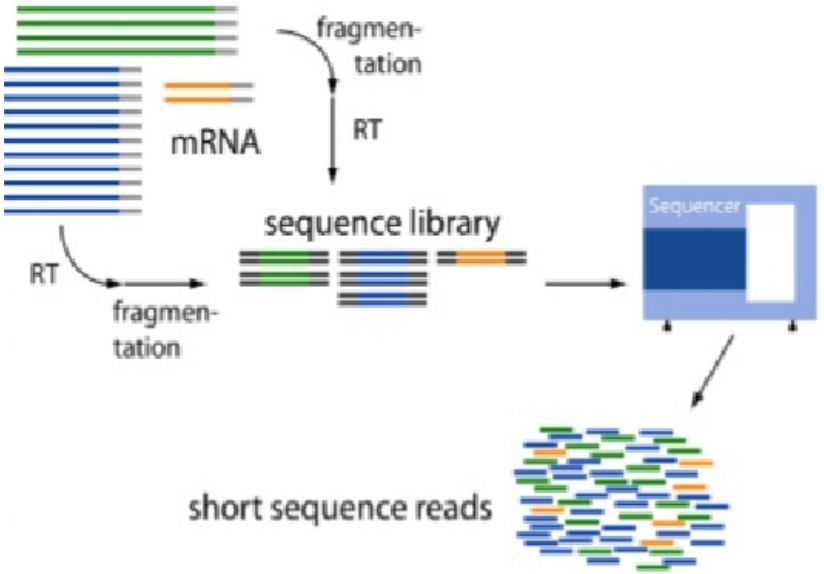
- **Comparison**

  Given samples from different experimental conditions, find effects of the treatment on
  - gene expression strengths
  - isoform abundance ratios, splice patterns, transcript boundaries

RNA-Seq versus Microarrays



Microarrays: Fluorescence intensities
Analogue Signal

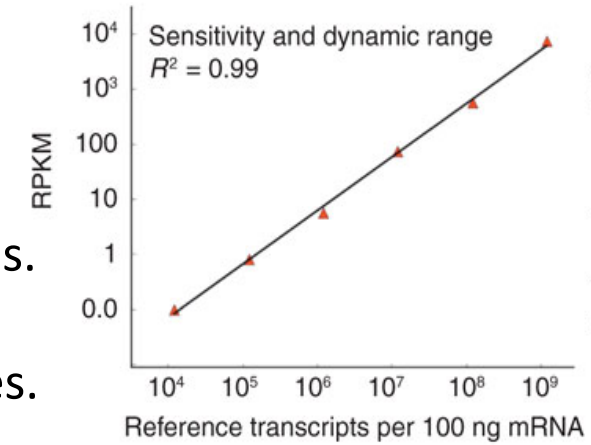


RNA-Seq: Sequence Reads
Digital Signal

Suraj Menon

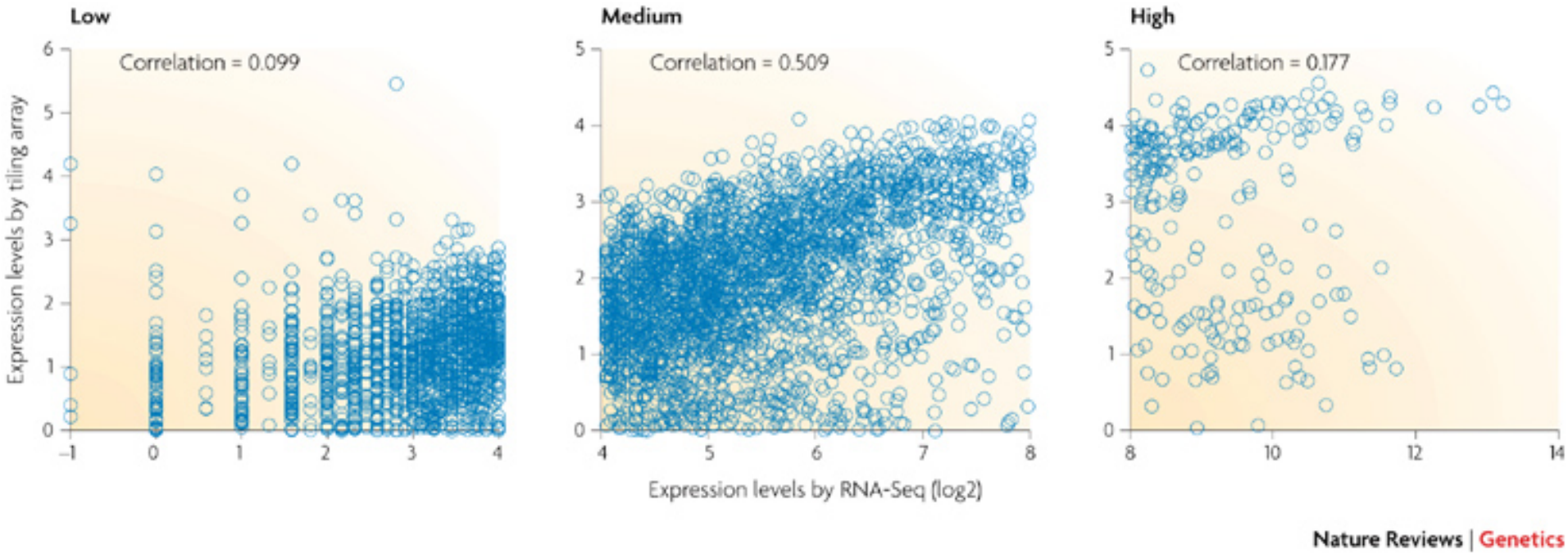
RNA-Seq versus Microarrays

- **Arrays suffer from fundamental 'design bias' :**
  - They only return results from those regions for which probes have been designed.
  - RNA-Seq allows for assay of novel transcripts/isoforms.
- **RNA-Seq has much lower noise levels and wider dynamic range:**
  - $10^4 – 10^9$ vs. a few hundred-fold
  - RNA-seq has greater sensitivity, better discrimination of DE for large expression values.
  - BUT bias for higher expressed and longer genes.



Mortazavi et al. Nature Methods, 2008

RNA-seq and microarray agree fairly well only for genes with medium levels of expression



Saccharomyces cerevisiae cells grown in nutrient-rich media. Correlation is very low for genes with either low or high expression levels.

Wang et al (2009) *Nature Reviews Genetics* 10, 57-63

RNA-seq Advantages Over Microarrays

Technical:

- Higher specificity
- Increased dynamic range of detection
- Lower technical variation
- More future-proof

Allows us to analyse **genome-wide** transcription multiple levels of information:

- Novel transcripts
- smRNA & miRNA
- Alternative splicing events
- Allele-specific transcripts
- Epigenetics
- Chimeric sequences
- Transcribed and non- translated regions

RNA-Seq versus Microarrays

- **Cost:**
  - RNA-Seq is more expensive but costs are dropping
  - Dependent on experimental design and feature.

- **Data Analysis:**
  - RNA-Seq data analysis is considerably more computationally and resource-intensive.
    - Storage, computation, computational analyst involvement.
  - Microarray workflows are well-established.

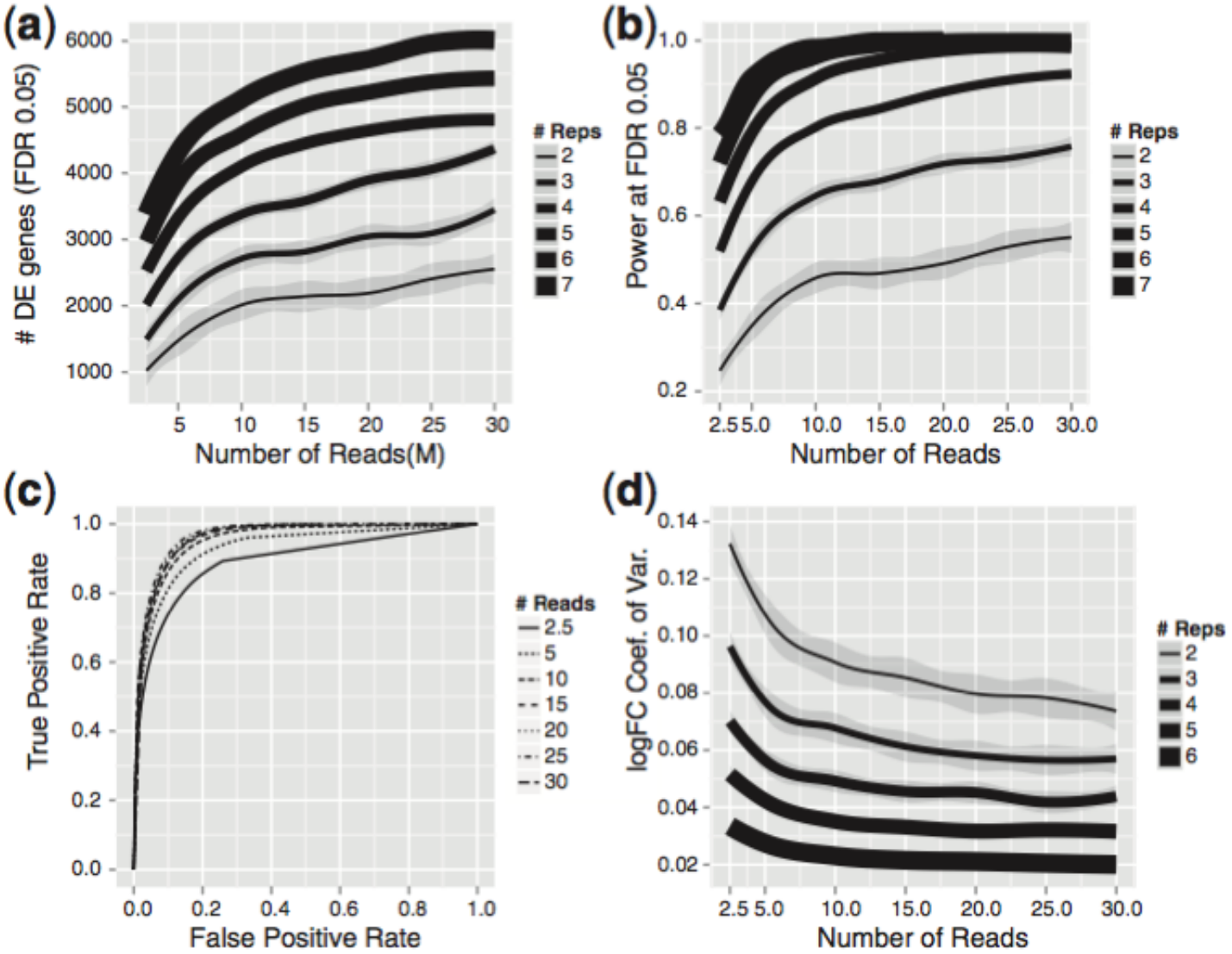
Experimental Design

RNA-seq Experimental Design Considerations

- **Number of Replicates**
  - Power to detect DE depends on sample size
- **Batch Effects**
  - Different batch of reagents, date, instrument, person?
- **Multiplexing**
  - If multiple pools – which samples in which pool?
- **Sequencing Depth/Coverage**
  - Depends on feature and application
  - Gene-level or transcript level?
  - Gene fusions, DAE?

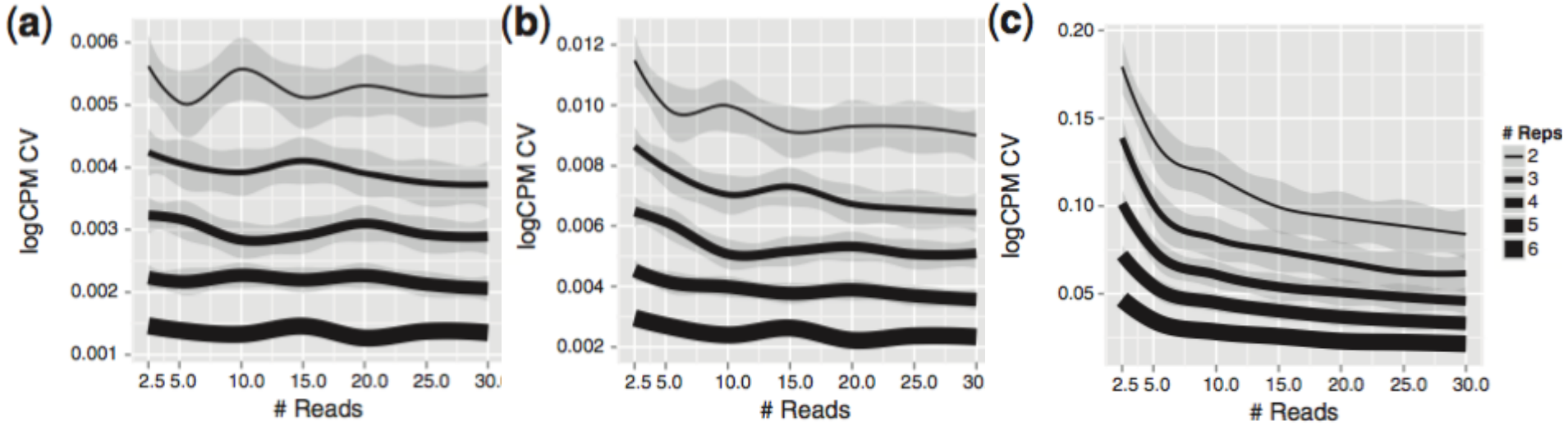
Randomisation, Blocking & Replication!!!!

Increase in biological replication significantly increases the number of DE genes identified.



Liu et al. 2014 Bioinformatics Feb 1;30(3):301-4

Biological replicates improve the accuracy in estimating expression level for all genes, regardless of expression level

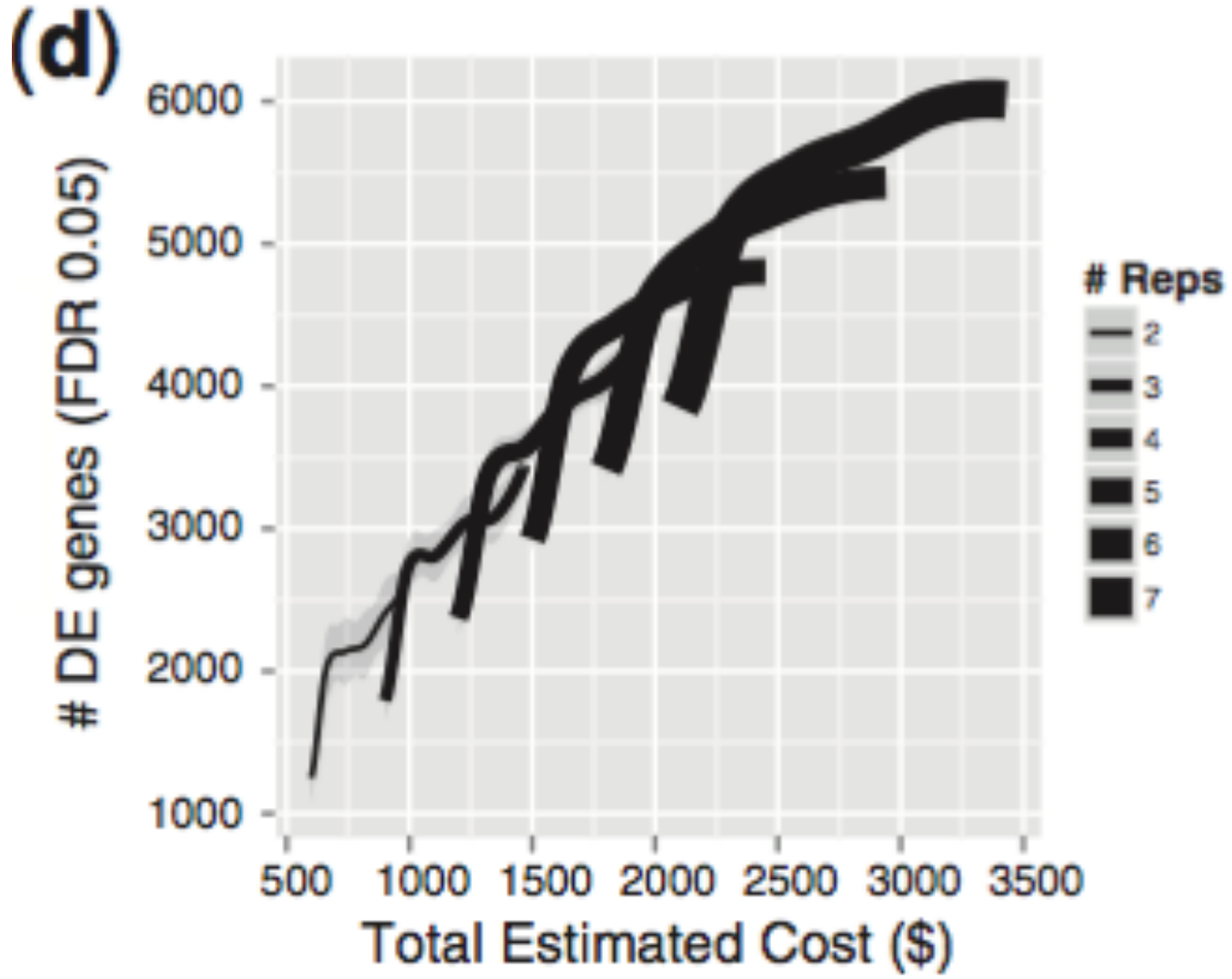


HIGH MEDIUM LOW

Whereas adding **sequencing depth** will improve estimation accuracy mostly for low expression genes.

Liu et al. 2014 Bioinformatics Feb 1;30(3):301-4

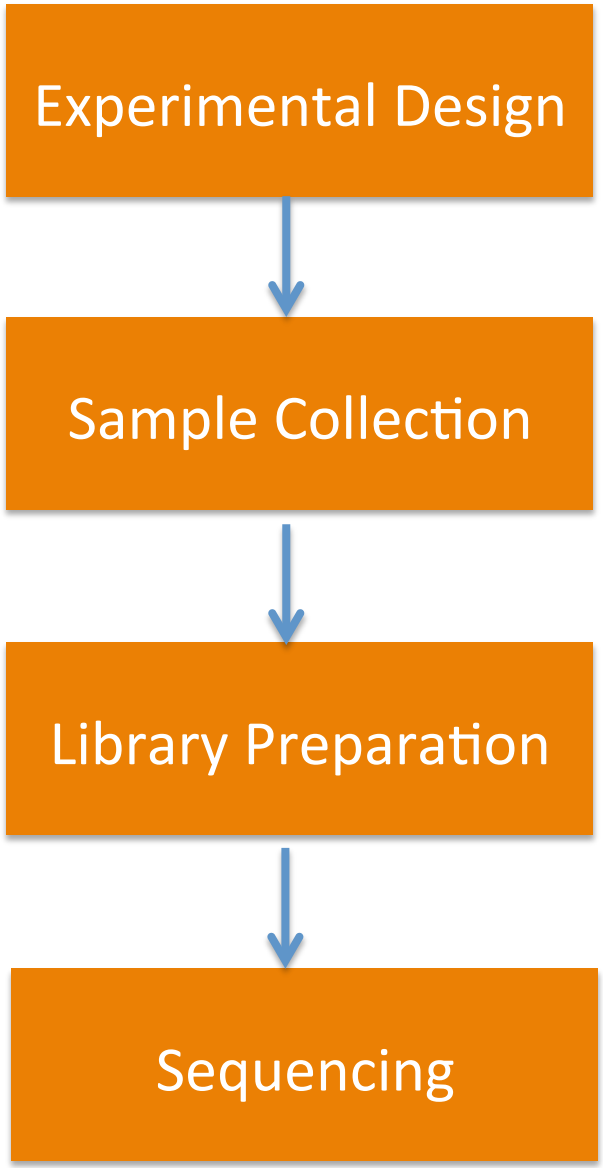
If higher numbers of DE genes are needed, increased biological replication should be used



Liu et al. 2014 Bioinformatics Feb 1;30(3):301-4

RNA-seq Data Analysis Workflows

General Sample Prep Workflow



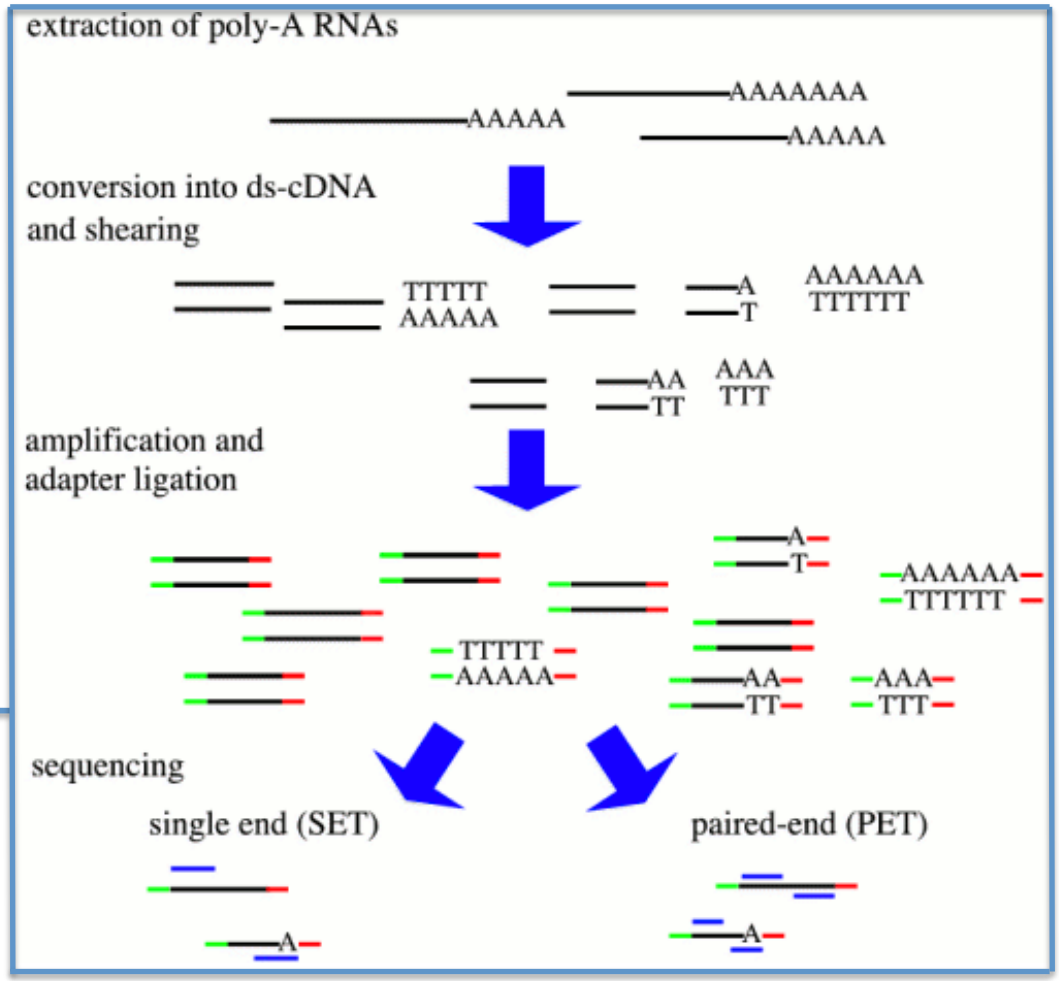
General Sample Prep Workflow

Experimental Design

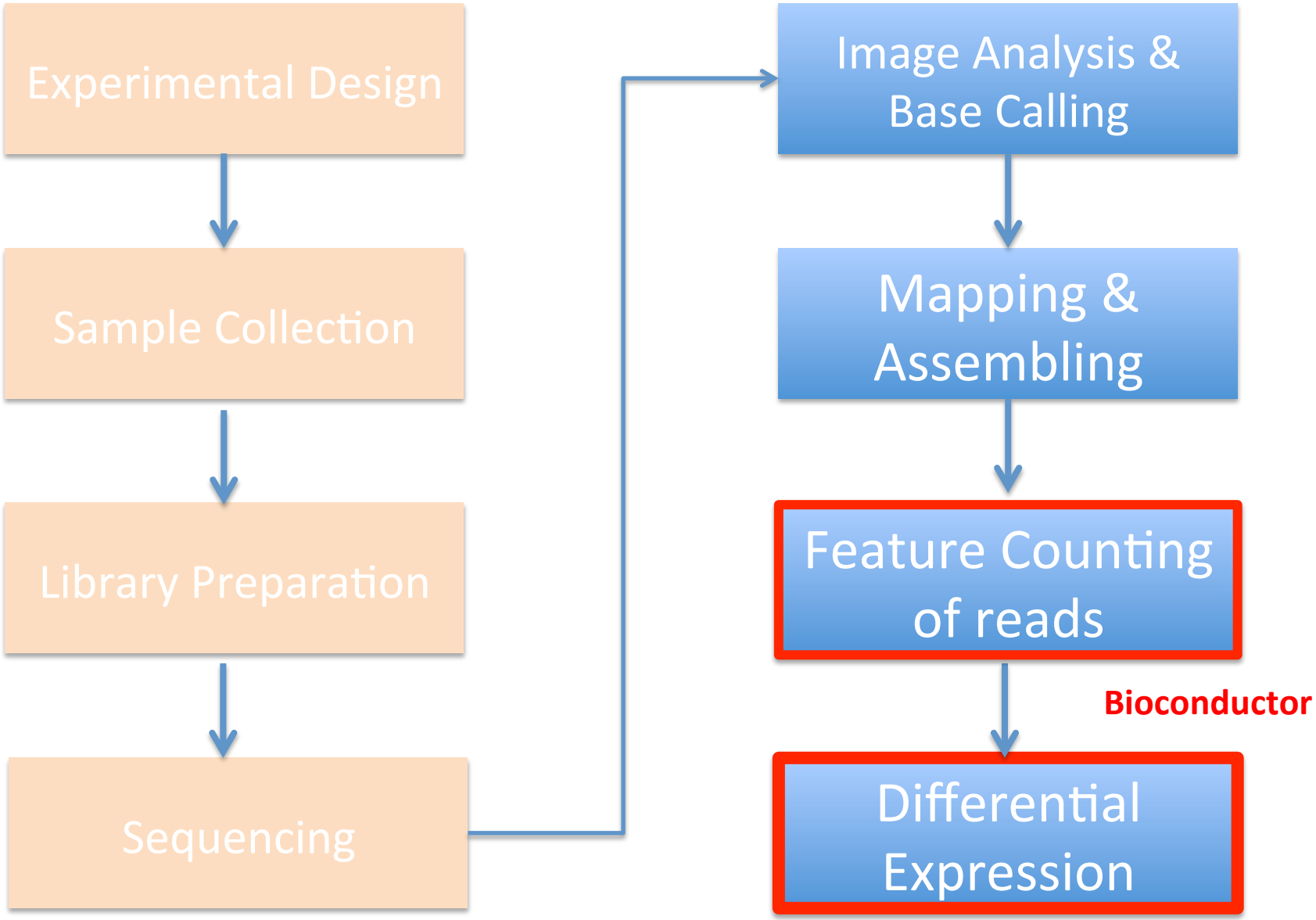
Sample Collection

Library Preparation

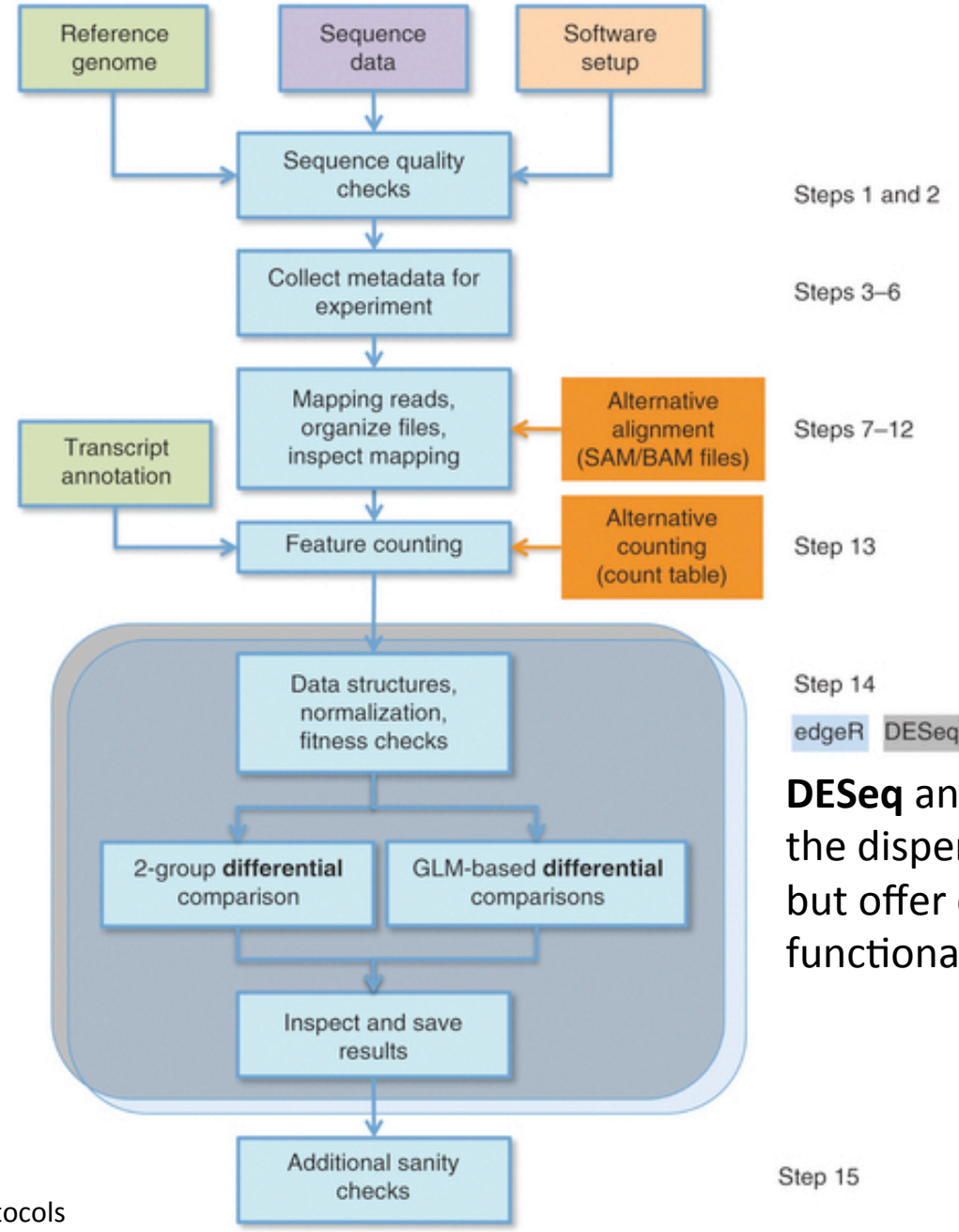
Sequencing



General Sample Prep Workflow



Standard RNA-seq DE Analysis Workflow



DESeq and **edgeR** estimate the dispersions differently but offer overlapping functionality

Simon Anders et al. (2013) Nature Protocols

Statistical issues in RNA-seq Data Analysis

- **Summarizing**:
  - Counts versus RPKM and other summaries
- **Normalization**:
  - Robust estimates of library size
- **Differential Expression**:
  - Appropriate error model (Negative Binomial, Poisson, . . .)
  - dispersion (under negative binomial) as parameter requiring estimation;
  - 'shrinkage' to balance accuracy of per-gene estimates with precision of experiment-wide estimates.
- **Testing**:
  - Filtering to reduce multiple comparisons & false discovery rate.

RNA-seq Data

Aligned Reads

- Data is a (large) set of sequences
- Typical file format is FASTQ

```
@HWI-EAS255_4_FC2010Y_1_43_110_790
TTAATCTACAGAATAGATAGCTAGCATATATTT
+
hhhhhhhhhhhhhhhdhhhhhhhhhhhdRehdh
```

Read identifier

Bases called

Base quality codes

- Alignment to genome is done by efficient indexing of seed sequences
- Aligned reads in SAM format

```
@HWI-… 163 chr19 9900 10000 16M2I25M
```

Read identifier

Where this read matched

Start and end positions

Codes for match:
16 matches, 2 extra,…

FastQC

- FastQC is free software under the GPLv3. You can download it from:

  http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc/
  http://www.youtube.com/watch?v=bz93ReOv87Y

- Perform QC checks on high throughput sequence data.

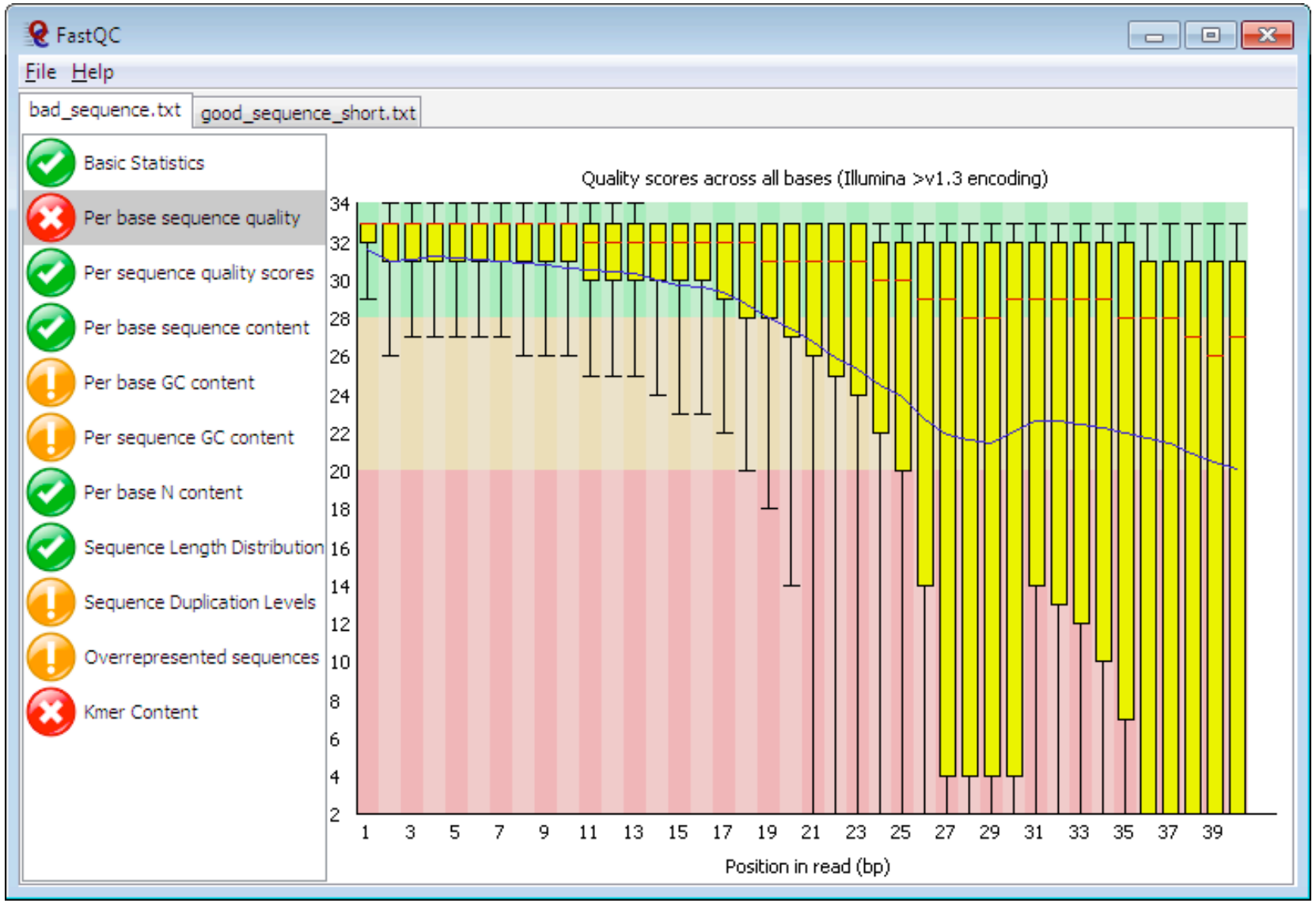
- FastQC runs a series of tests and will flag up and potential problems with your data.

- As with all of the plots in FastQC it's not there to tell you if your data is good or bad, it's there to tell you if your data looks unusual in some way.

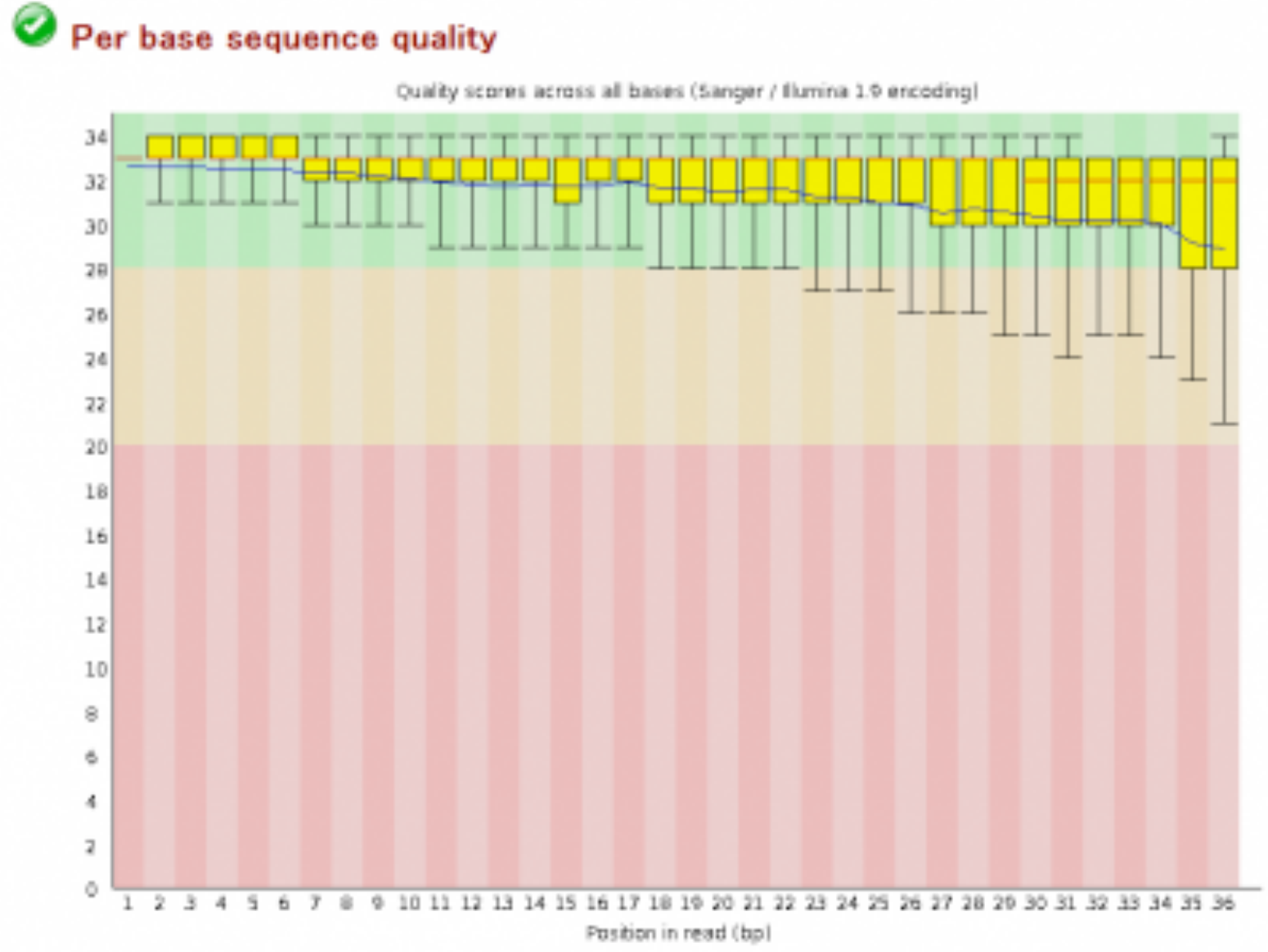
- Run as an interactive GUI application or run in an unattended offline mode where it generates HTML versions of its reports.

- Graphics: read length plots; read-quality plots; sequence duplication levels and many more

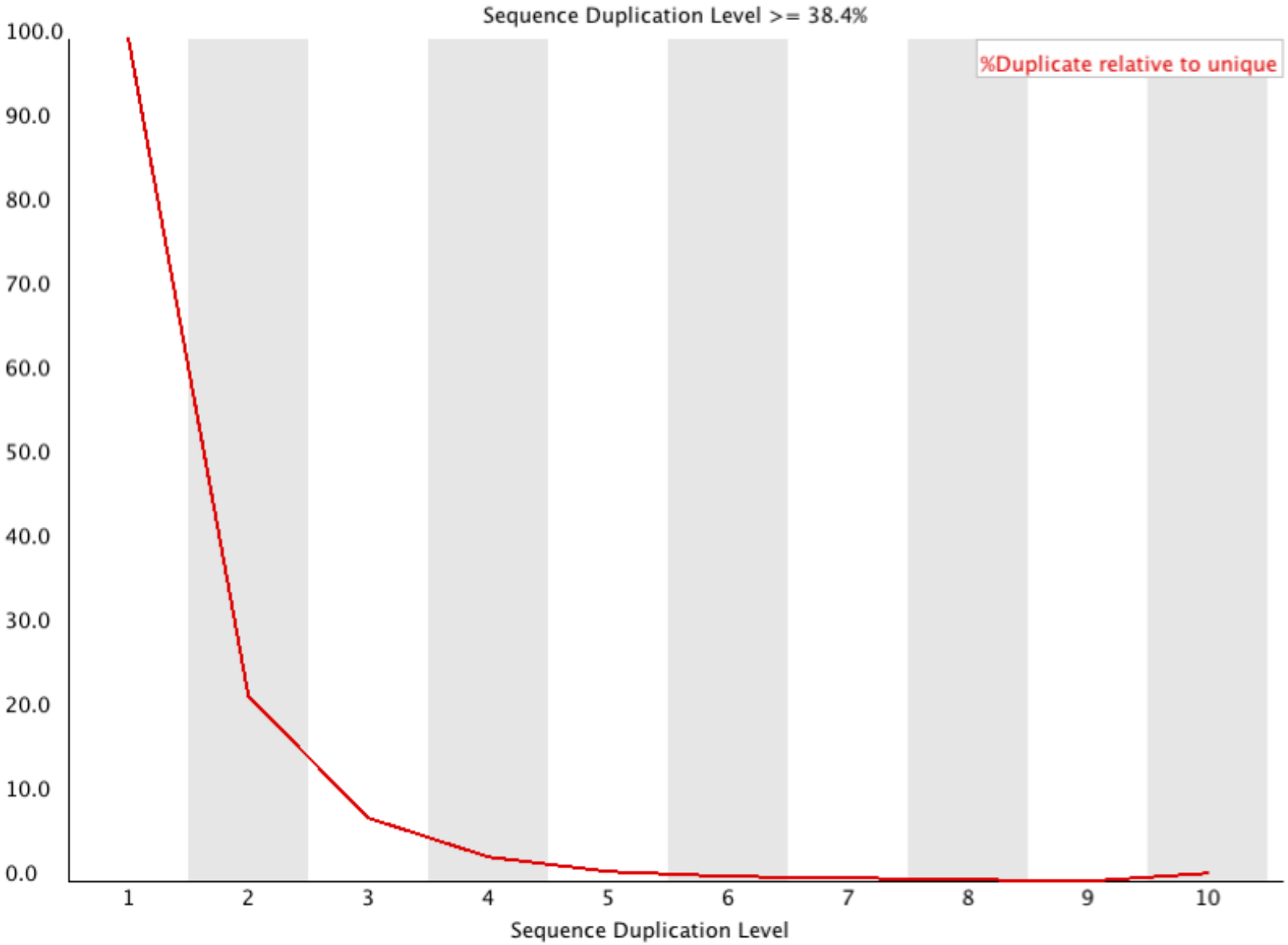
FastQC: Per base sequence quality (bad)



FastQC: Per base sequence quality (Ok)



FASTQC: PCR Duplication



http://proteo.me.uk/2011/05/interpreting-the-duplicate-sequence-plot-in-fastqc/

Read/PCR Duplication

- May be **artificial:** during the sequencing procedure a copy of the same read is created and sequenced
  - It may be an indication of **poor library complexity** caused by low sample input or over-amplification.
  - Duplicates often correlate with too little sample material, and/or difficulties in the lab, so worth checking
- Or **natural**: the same DNA fragment occurs and is sequenced twice.
  - Often due to very **high abundance of a small number of genes**.
  - Number of reads mapping to a gene is a measure of its expression, so by removing reads, you will bias the true expression measurements.
- Observing high rates of read duplicates in RNA-seq libraries is common, but removing PCR duplicates is not necessary.

.

Alignment of RNA-seq Data

Two major approaches for RNA-seq data processing to identify DEGs:

- Reads are mapped onto a **reference genome** and/or **transcriptome** so DEG results dependent on the aligner used.

- The ***de novo* assembly** of short reads (i.e. does not require a reference genome).

Alignment: genome or transcriptome?

Should one align to the **genome** or the **transcriptome**?

- transcriptome
  - easier, because no gapped alignment necessary
  - (but: splice-aware aligners are mature by now)

BUT

- Risk to miss possible alignments!
  (transcription is more pervasive than annotation claims)

→ Alignment to **genome** preferred.

Alignment of RNA-seq Data

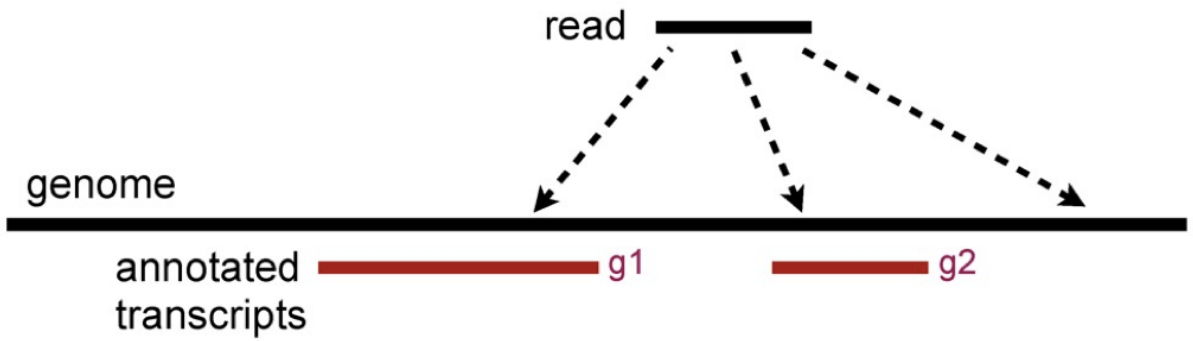
- Challenging and yet unsolved problem because:
  - Non-contiguous transcript structure
  - Relatively short read lengths

Computationally Intensive Tasks

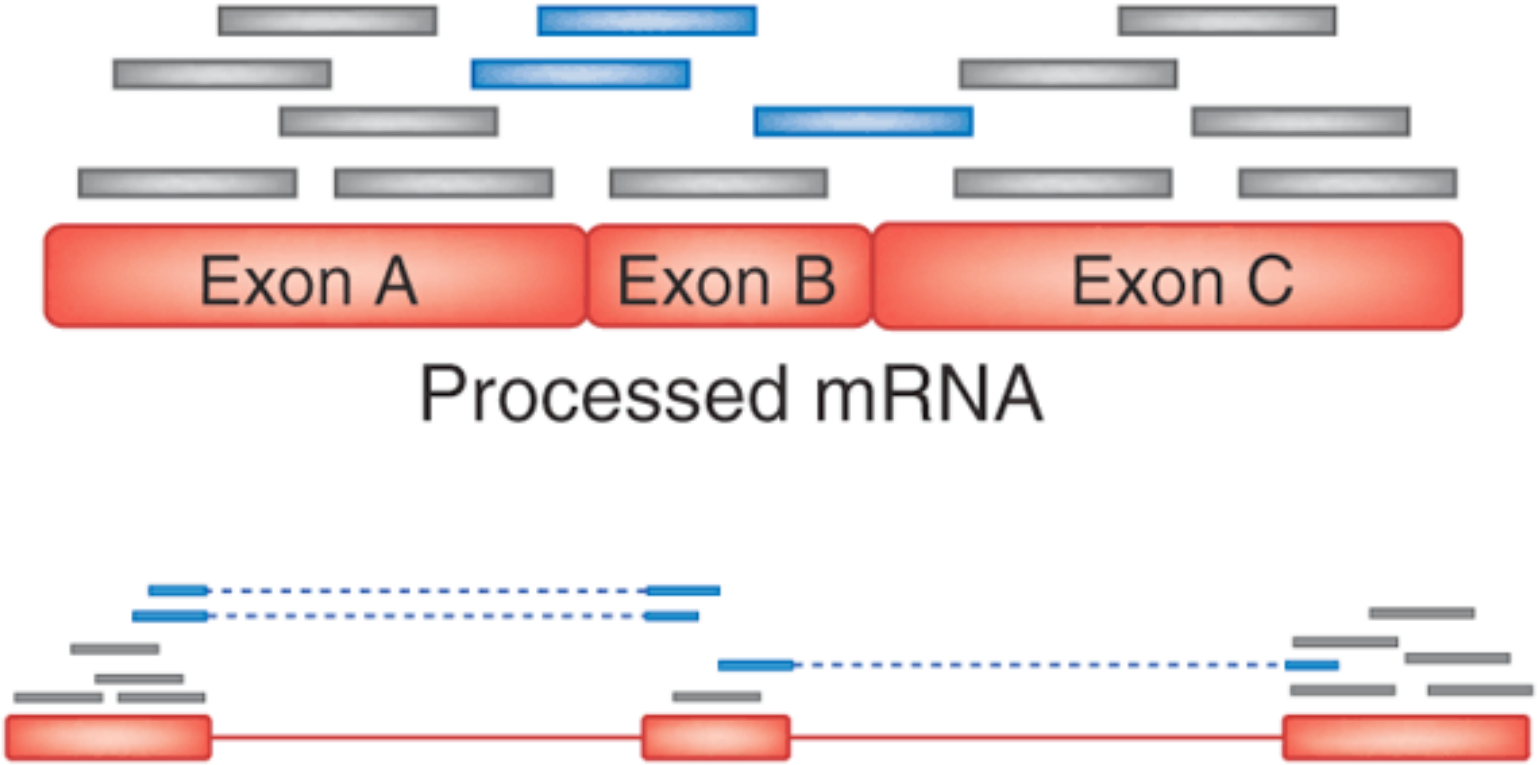
1. Accurate alignment of reads that contain ***mismatches***, ***insertions*** and ***deletions*** caused by genomic variations and sequencing errors.

2. Mapping sequences derived from non-contiguous genomic regions:

   - Unique – maps to one location
   - Multi-mappable:
     - spliced sequence modules that are joined together to form spliced RNAs.
     - Chimeric (fusion) sequences (same or different chromosome)



Mapping to the Genome



- Bowtie or Maq will produce the alignments shown in black but will fail to align the blue reads.
- TopHat or ERANGE will also report the (blue) alignments spanning intron boundaries.

Trapnell & Salzberg, Nature Biotech 2009

Mapping Quality

- **Confidence in read's point of origin**

- **Depends on**
  - length of alignment
  - number of mismatches and gaps
  - uniqueness of the aligned region in the genome

- **Expressed in Phred scores, like base qualities**
  - Q = -10 * log10 (probability that read was mapped to a wrong location)
  - Marked in numbers

Current Popular Aligners

Short or unspliced Aligners

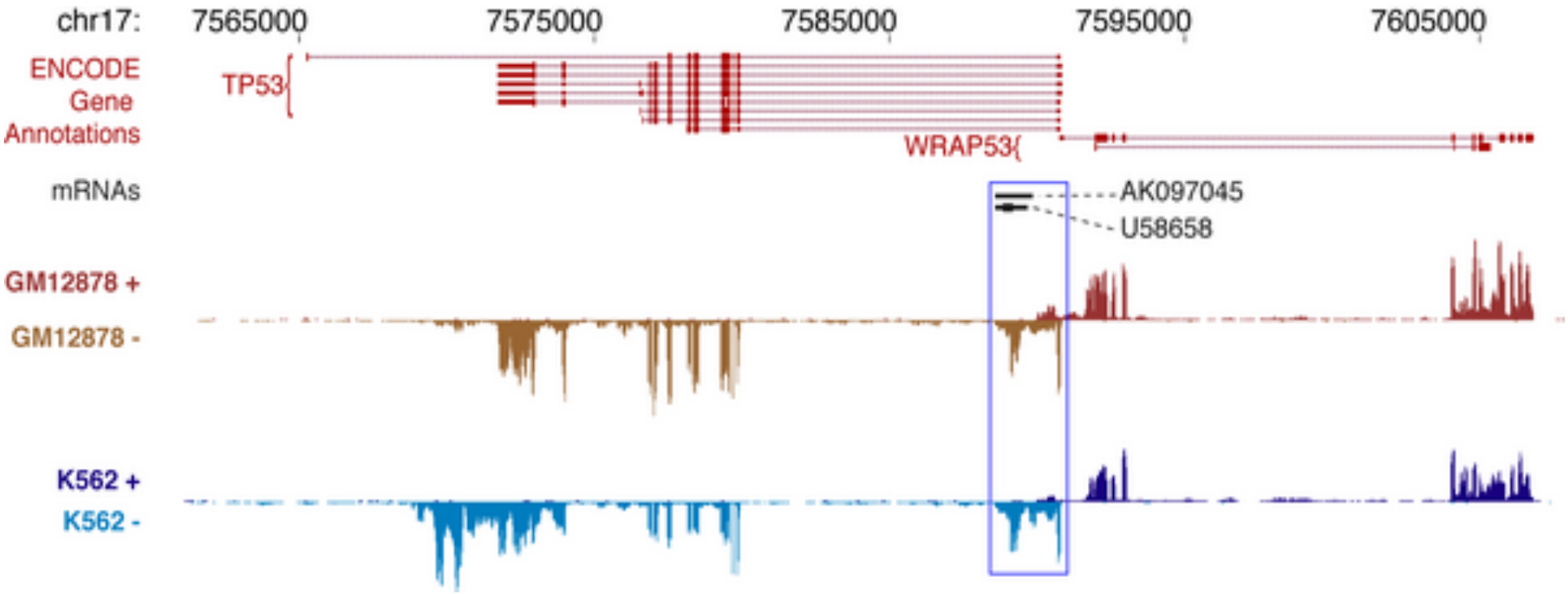
1. BWA
2. Stampy* - sensitive mapping
3. NovoAlign
4. Bowtie

Spliced Aligners

5. TopHat* - most popular
6. GSNAP* - very fast
7. MapSplice
8. RUM

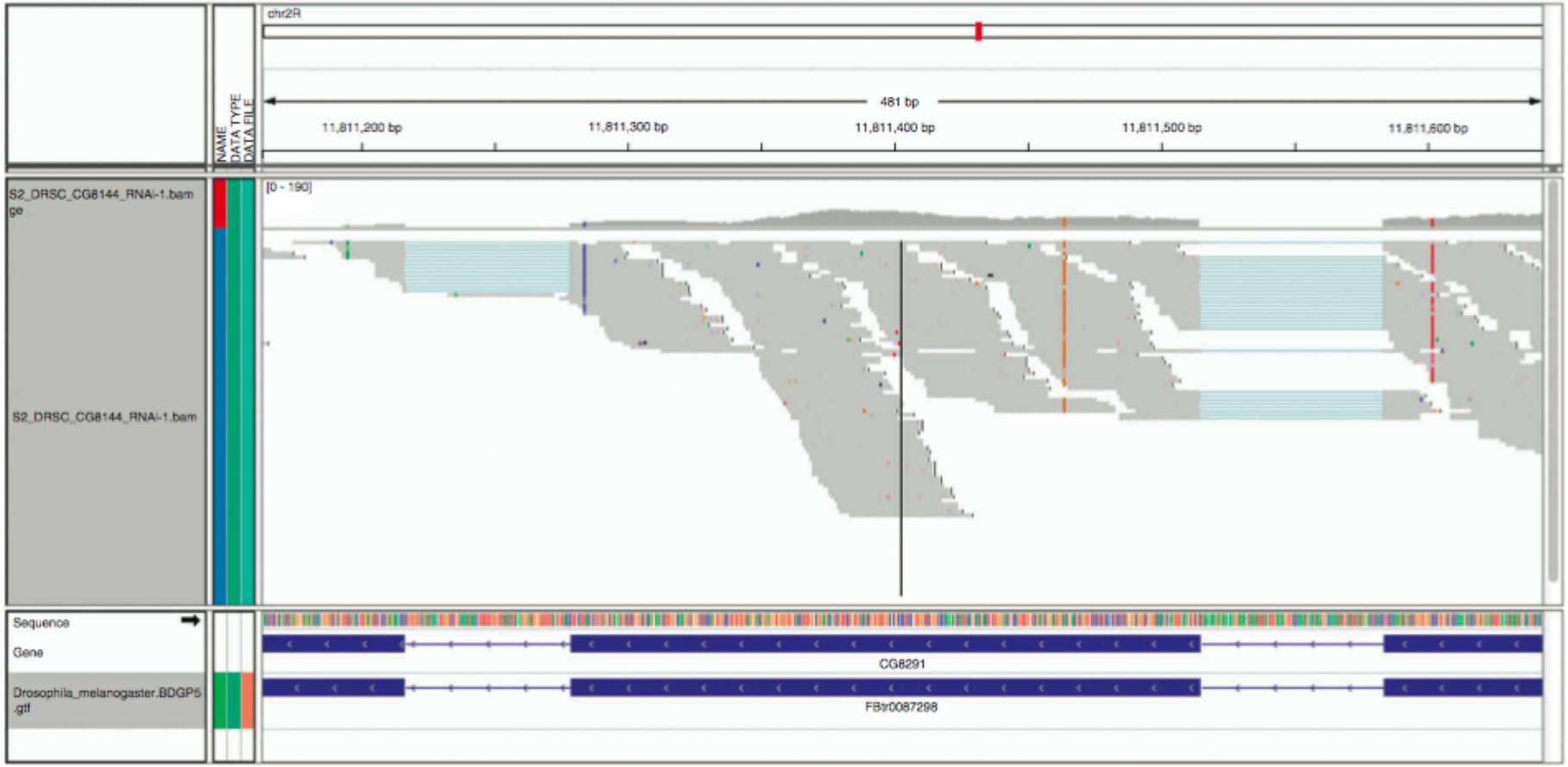
>20 aligners in the last 2 years!

RNA-Seq data is often represented by ‘pile-up’ diagrams



From: The ENCODE Project Consortium (2011) A User's Guide to the Encyclopedia of DNA Elements (ENCODE). PLoS Biol 9(4): e1001046.

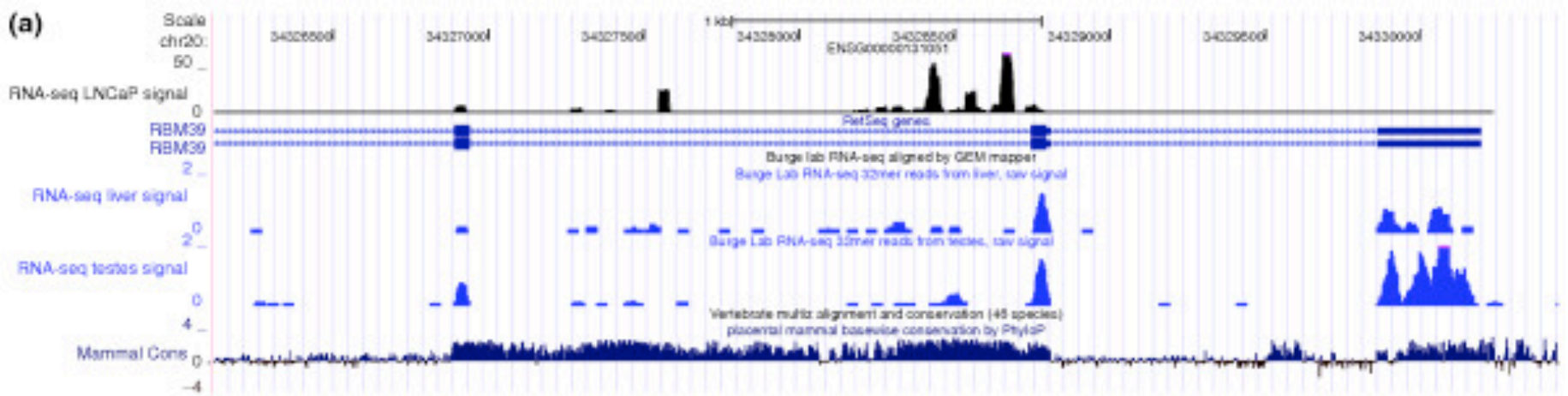
IGV Browser: screenshot of reads aligning across exon junctions

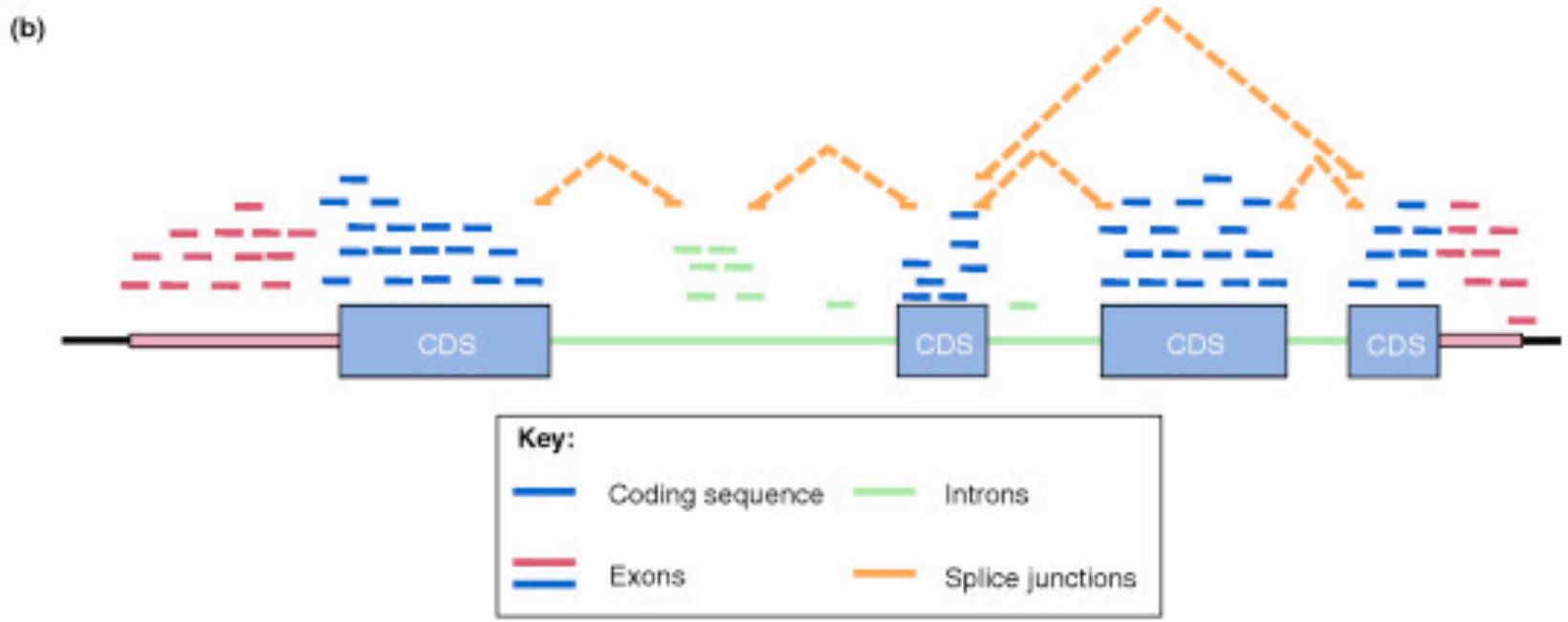


Load **BAM** files and the **GTF** file
http://www.broadinstitute.org/software/igv/home

Summarising
(Feature Counting; Quantification)

Summarising mapped reads into gene level counts





Feature Counting: gene-level

- Many methods exist inside and outside of Bioconductor to arrive at a table of counts given **BAM** (or SAM) files and a set of features (e.g., from a GTF file);

- Count the **number of reads** that fall into annotated genes.

- **Counting Rules**:
  - Count reads, not base-pairs
  - Count each read at most once.
  - Discard a read if
    - it cannot be uniquely mapped
    - its alignment overlaps with several genes
    - the alignment quality score is bad
    - (for paired-end reads) the mates do not map to the same gene

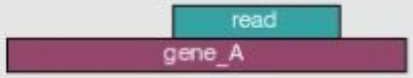
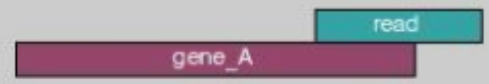
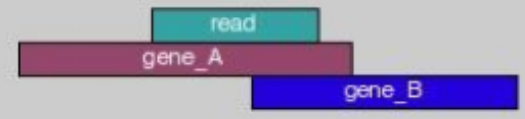
- **Perform stats** on the table of counts to identify DE
  - packages like **DESeq** and **edgeR** expect count data otherwise their model won't hold.
  - **Raw counts** are used to assess measurement precision.

Feature Counting: gene-level

- Reads that **cannot uniquely be assigned** to a gene because:
  - Reads aligned with gaps and gaps are inconsistent with known exon boundaries.
  - Reads position is annotated as part of several overlapping features.
  - For DE, such reads should be discarded!

- **hyseq-count** (from Python package **Htseq** – default union counting mode)
- **easyRNAseq**
- **SummarizeOverlaps** (GenomicRanges; gene & exon-level)
- **gCount** (QuasR)

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

Feature Counting: gene-level

	control-1	control-2	control-3	treated-1	treated-2
FBgn0000008	78	46	43	47	89
FBgn0000014	2	0	0	0	0
FBgn0000015	1	0	1	0	1
FBgn0000017	3187	1672	1859	2445	4615
FBgn0000018	369	150	176	288	383
[...]					

Count data in Htseq

Counting reads in genes: summarizeOverlaps (GenomicFeatures)

Download genes:

```
> library(GenomicFeatures)
> hse <- makeTranscriptDbFromBiomart(biomart="ensembl",
    dataset="hsapiens_gene_ensembl")
```

Produce a GRangesList object of all exons grouped by gene:

```
> exonsByGene <- exonsBy(hse, by="gene")
```

Counting reads in genes using `summarizeOverlaps` from the GenomicRanges and Rsamtools packages. library(Rsamtools)

#Read in BAM files

```
> bamDir <- system.file("extdata",package="parathyroidSE",mustWork=TRUE) >
fls <- list.files(bamDir, pattern="bam$",full=TRUE)
> bamlst <- BamFileList(fls)
```

#The protocol is not strand specific, so we set `ignore.strand=TRUE`. We counted "singletons" as well, reads with an unmapped mate, and added these counts to produce a total

```
> geneHitsPairs <- summarizeOverlaps(exonsByGene, bamlst, mode="Union",
      singleEnd=FALSE, ignore.strand=TRUE)
> geneHitsSingletons <- summarizeOverlaps(exonsByGene, bamlst,
      mode="Union", param=ScanBamParam(flag=scanBamFlag( isPaired=TRUE,
      hasUnmappedMate=TRUE)), singleEnd=TRUE, ignore.strand=TRUE)
> parathyroidGenesSE <- geneHitsPairs
> assay(parathyroidGenesSE) <- assay(geneHitsPairs) +
      assay(geneHitsSingletons)
```

Code taken from package parathyroidSE in Bioconductor

Normalization of RNA-Seq Data

Why Normalise?

- As with microarrays, analysis of RNA-seq data requires the careful accounting of factors that may introduce:
  - **Systematic effects**
    - e.g. differences is the amount of RNA; library prep; lane differences; sequencing depth; etc.
  - **Confounding variability** in the expression measurements
    - e.g. batch; day of processing; clinical covariates etc.

Why Normalise?

- As with microarrays, analysis of RNA-seq data requires the careful accounting of factors that may introduce:
  - **Systematic effects**
    - e.g. differences is the amount of RNA; library prep; lane differences; sequencing depth; etc.
  - **Confounding variability** in the expression measurements
    - e.g. batch; day of processing; clinical covariates etc.
- RNA-seq requires **robust estimates of sequencing depth** or library size:
  - *library sizes* (the total number of mapped reads) are typically different for different samples, which means that the observed counts are not directly comparable between samples.
- Otherwise, may result in *false positives* & *spurious correlations.*

Sequencing Issues

- Seq. technology is a **sampling procedure** from a population of transcripts:
  - Differences in transcript relative distributions between samples will affect the assessment of DE.

- Ability to detect **rare transcripts** is obscured by:
  - Wide dynamic range of mapped reads
  - The concentration of a large portion of the sequencing output in a number of highly expressed transcripts.

- Quantification of expression depends on the **length of the biological features** (gene/exon/transcript) under study.
  - Longer features will generate more reads than shorter ones.

Normalisation Methods

- **Global-scaling approaches:**
  - **RPKM** reads per kilobase of gene (or exon) per million reads;
  - **FPKM** (Cuffdiff, Trapnell 2010; Mortazavi, 2008)
  - **Trimmed Mean of M-values (TMM)** (**edgeR**)
  - **Relative Log Expression** approach (**DESeq**)
  - *Both edgeR and DESeq keep the raw counts and normalisation factors separate as this full information is needed to correctly model the data!*
- **Full-quantile approaches:**
  - a conditional quantile normalization (CQN) procedure (Hansen et al, 2011)
- **Generalised Linear Model (GLM)** approaches:
  - for sample specific GC content effects, offsets are presented to GLM while maintaining counts on the original scale (Risso et al, 2011; Hansen, 2012)

Normalisation: RPKM/FPKM

- **Expression estimates** in either:
  - **RPKM** (Reads Per Kilobase per Million mapped reads ; Mortazavi *et al*. 2008)
  - **FPKM** (Fragments Per Kilobase per Million mapped reads)

- A simple *global-scaling* **normalization** approach on raw counts.

- The goal of such transformations is to normalize the counts with respect to:

  - **Differing library Sizes,** numbers of counts per sample (sequencing depth).

  - the **length of the transcripts**, since a long transcript is expected to obtain more reads than a short transcript with the same expression level.

- Divide counts in a **region of interest** (a genomic region or a gene or an exon etc) by all counts (reads per million reads -RPM).

- Genes **have different lengths** so divide also by length of gene.

Normalisation: RPKM/FPKM

- **Expression estimates** in either:
  - **RPKM** (Reads Per Kilobase per Million mapped reads ; Mortazavi *et al*. 2008)
  - **FPKM** (Fragments Per Kilobase per Million mapped reads)

- A simple *global-scaling* **normalization** approach on raw counts.

- The goal of such transformations is to normalize the counts with respect to:

  - **Differing library Sizes,** numbers of counts per sample (sequencing depth).

  - the **length of the transcripts**, since a long transcript is expected to obtain more reads than a short transcript with the same expression level.

- Divide counts in a **region of interest** (a genomic region or a gene or an exon etc) by all counts (reads per million reads -RPM).

- Genes **have different lengths** so divide also by length of gene.

- However, this still widely-used approach has proven ineffective and more beneficial procedures have been proposed.

- Tends to be heavily affected by a relatively small proportion of highly-expressed genes and can lead to biased DE results.

Biases in RPKM Normalization

- Robinson et al noticed that most genes appeared less expressed in some liver samples in a landmark study

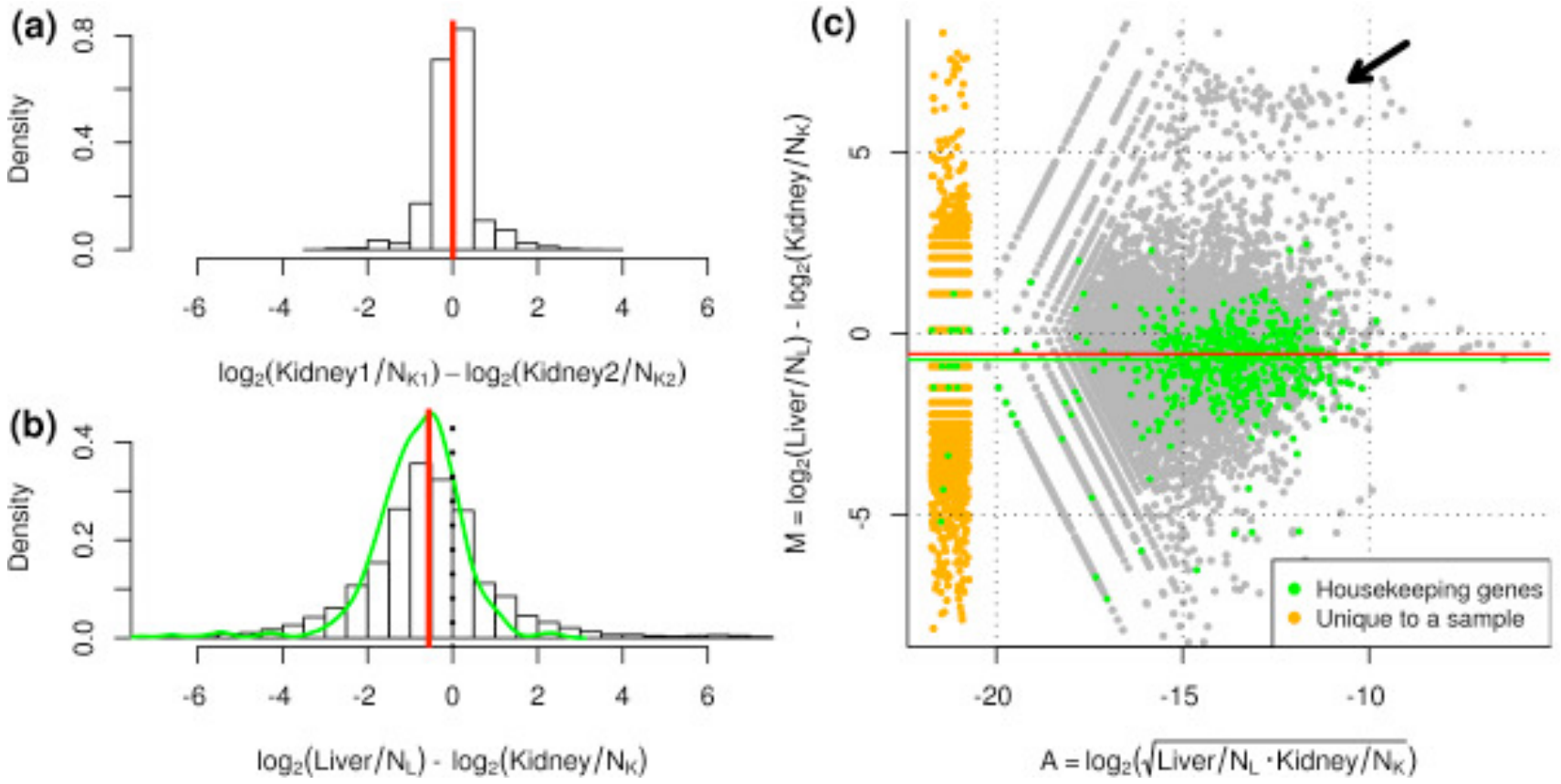
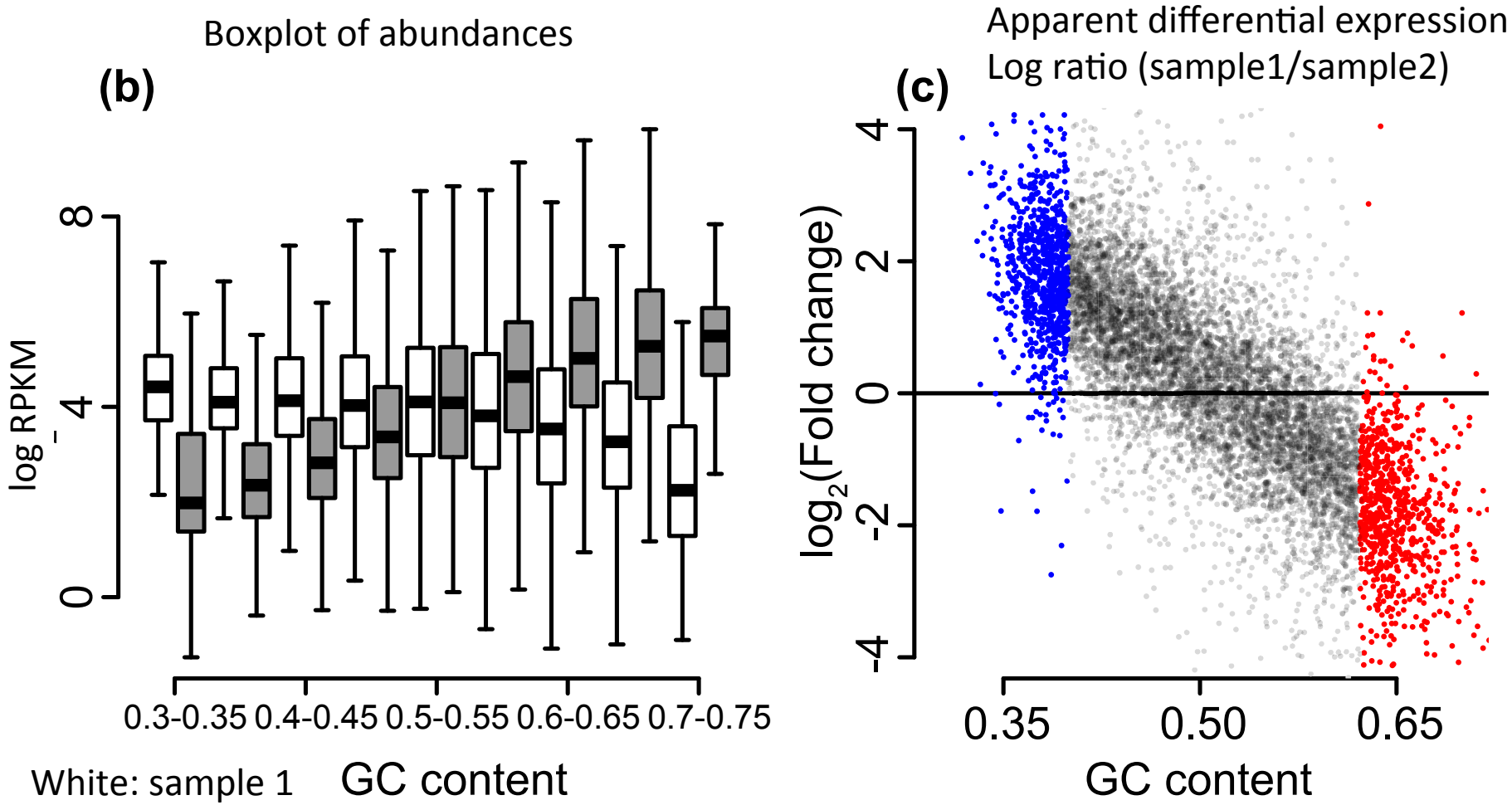


Fig 1 from Robinson & Oshlak, Genome Biology 2010

Their **Trimmed Mean of M-values (TMM)** procedure in **edgeR** centers trimmed log-ratios between samples: estimate the ratio of RNA production

GC Content May Affect Read Counts



From Hansen et al 2011

RPKM: Reads Per Kilobase per Million mapped reads

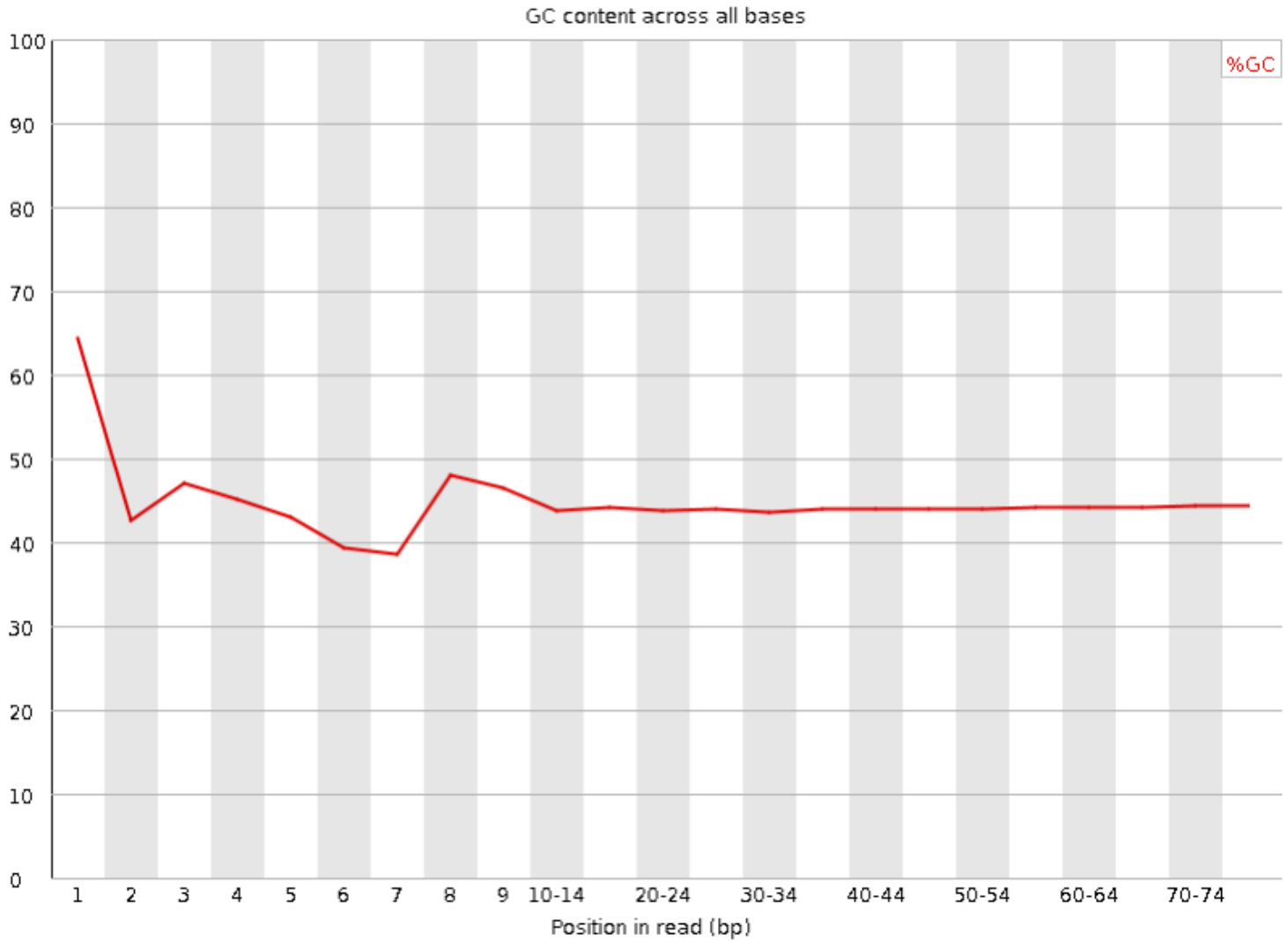
Other Sequencing Biases

- Hansen et al. Biases in Illumina transcriptome sequencing caused by **random hexamer priming**. NAR (2010)
  - Used to find sites for ploymerase to copy but results in bias in the nucleotide composition at the start of the sequence reads.
  - This bias influences the uniformity of the location of the reads along expressed transcripts.
  - They propose a re-weighting scheme that adjusts for this bias and makes the distribution of sequencing reads more uniform.

- Zheng et al. Bias detection and correction in RNA-sequencing data. BMC bioinf. (2011)

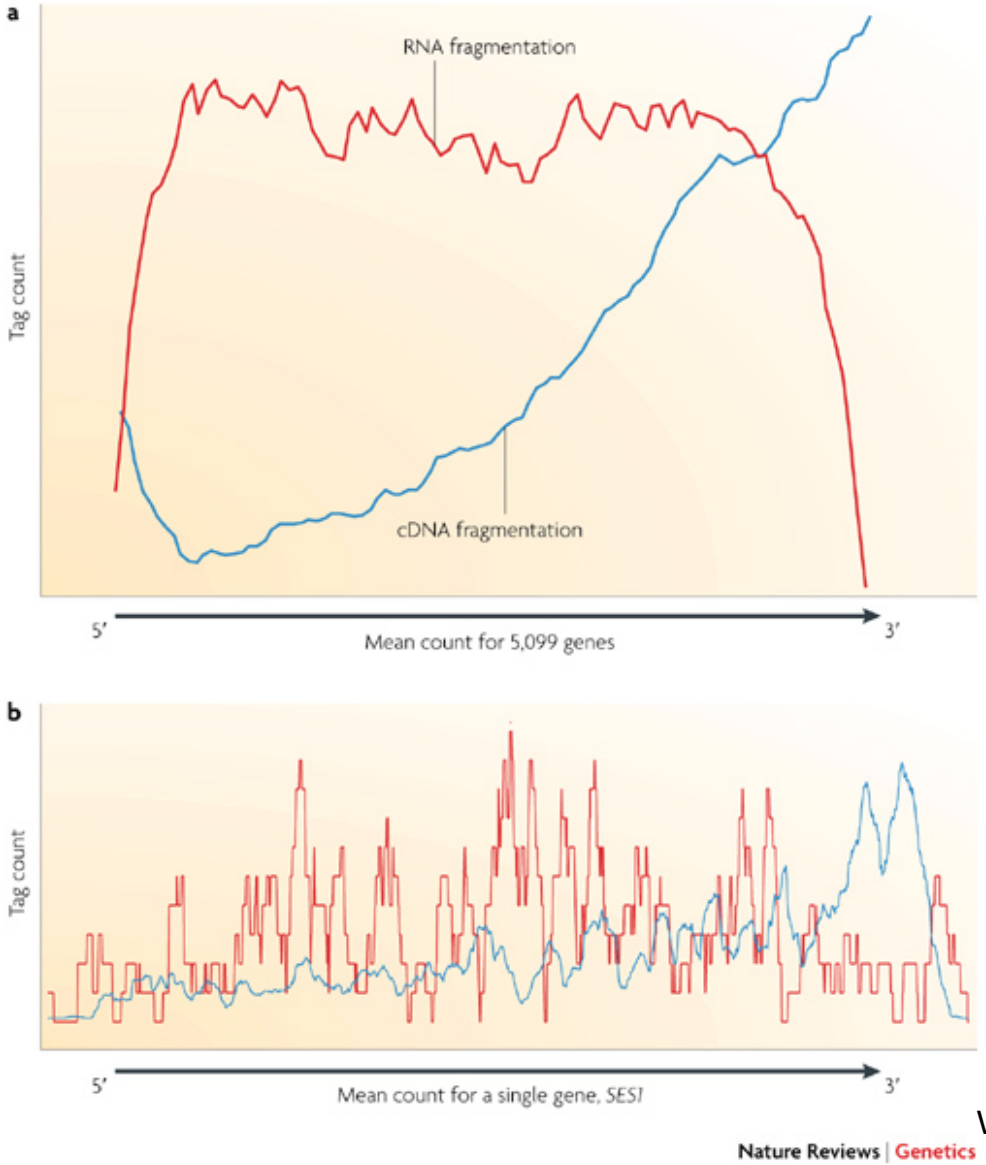
FASTQC

Per base GC content



RNA-seq library Prep

- Library Construction still a challenge!!!!



Fragmentation of oligo-dT primed cDNA (blue line) is more biased towards the 3' end of the transcript.

RNA fragmentation (red line) provides more even coverage along the gene body, but is relatively depleted for both the 5' and 3' ends.

A specific yeast gene, SES1 (seryl-tRNA synthetase)

Different fragmentation methods create different biases in the outcome.

Wang et al (2009) *Nature Reviews Genetics* 10, 57-63

Normalization by edgeR and DESeq

Trimmed Mean of M-values (TMM) (**edgeR**)
Relative Log Expression approach (**DESeq**)

- Aim to make normalized counts for non-differentially expressed genes similar 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

DESeq

- 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

Variance Estimation & Shrinkage

Differential Expression

DE requires:

- **Normalization**, in order to compare expression between samples
  - Different library sizes
  - RNA composition bias caused by sampling approach
- Appropriate **error model**
  - Negative Binomial accounts for overdispersion in biological replicates.
- **dispersion** (under negative binomial) as parameter requiring estimation;
- '**shrinkage**' to balance accuracy of per-gene estimates with precision of experiment-wide estimates.

Models for Count Data

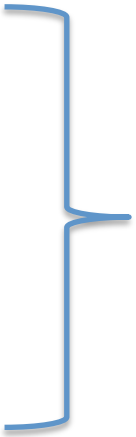
- **Poisson model**
  - Standard model for count data
- **Negative Binomial Model**
  - Higher variance than Poisson
- **Zero-inflated (mixture) NB model**
  - Allows excess 0 counts beyond either above

Statistical Methods Identifying DEGs:

Different statistical models based on discrete probability distribution:

- If apply overdispersed **Poisson model** (Robinson et al.)
  - implies the read counts follow a negative binomial (NB) model
  - NB model should be applied instead

- **edgeR**
- **Cuffdiff**
- **DESeq**
- **baySeq**
- **NOISeq**



Negative binomial model

Variance Estimation

- For DE, both **edgeR** & **DESeq** rely on an estimation of the typical relationship between the data's variance (**data's dispersion**) and their **mean**.

- Minimum variance of count data:

  $v = \mu$ (**Poisson**)

- Actual variance:

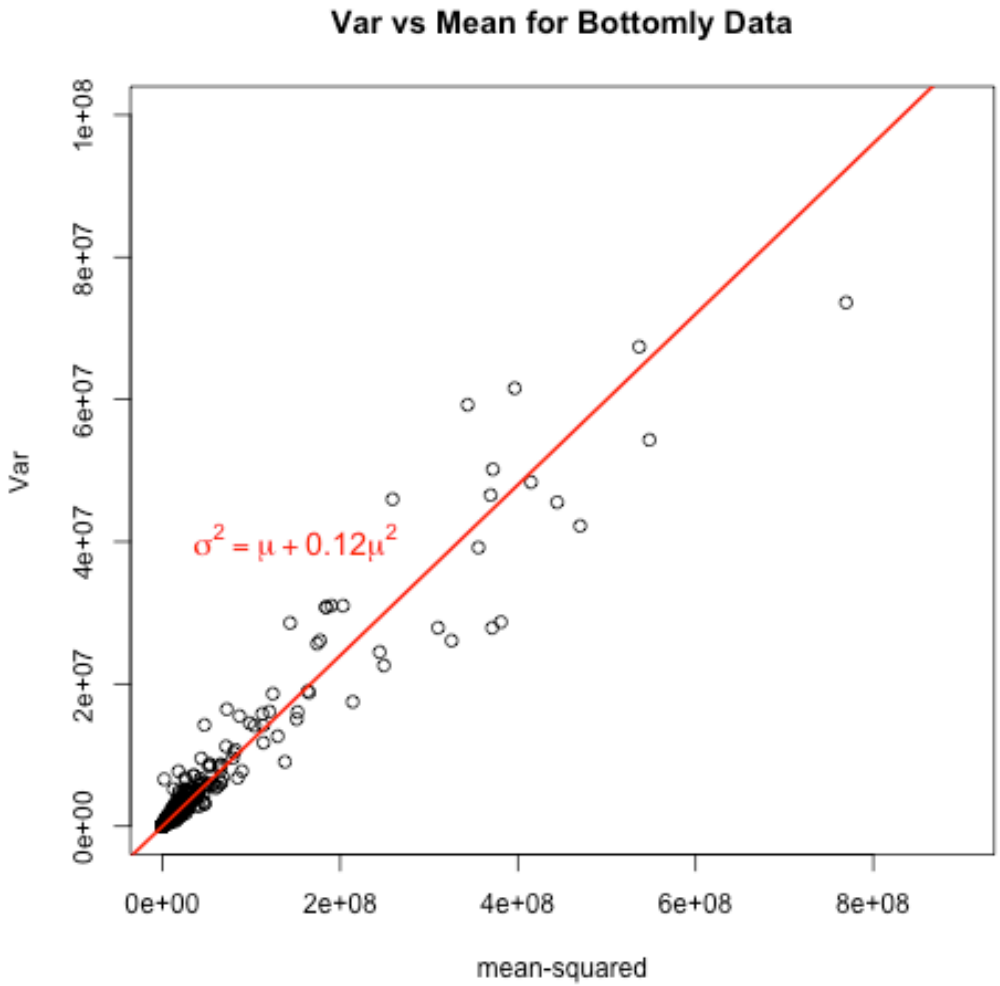
  $v = \mu + \alpha\mu^2$

- The "**dispersion**" α:

  $\alpha = (\mu - v) / \mu^2$

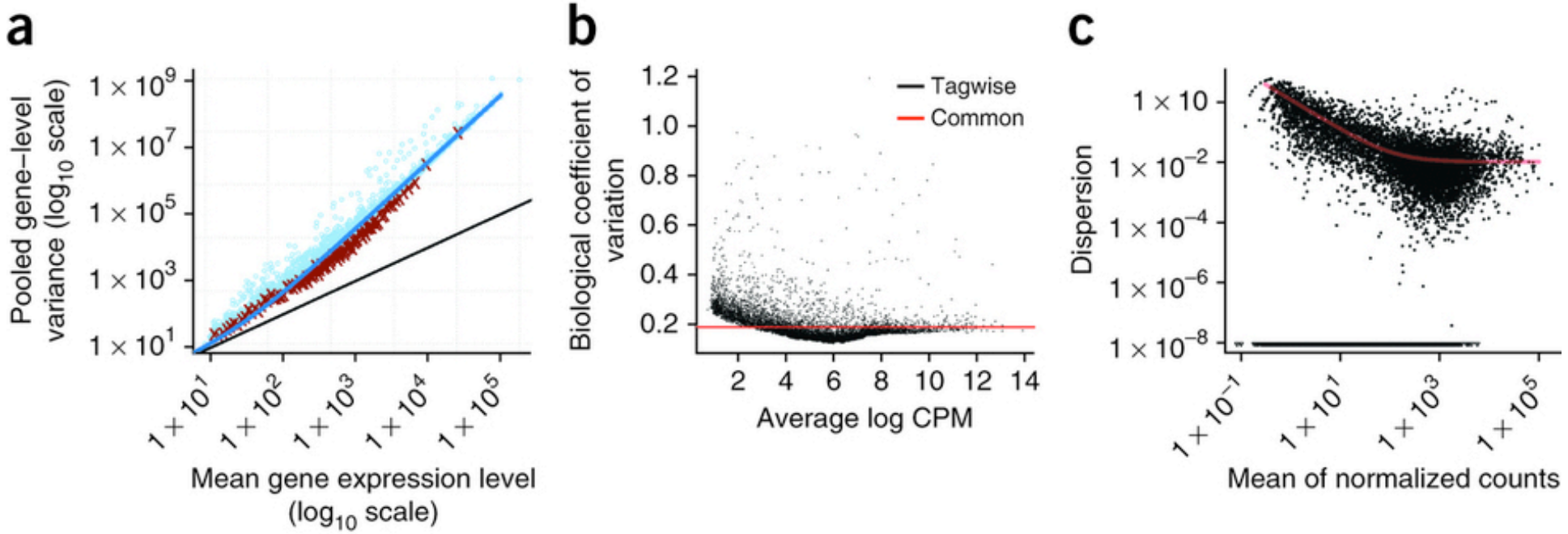
  = square of the coefficient of biological variation.

Variance Proportional to Mean



- Early RNA-seq studies found count variance across samples was proportional to count mean
- Among standard parametric distributions for counts, the negative binomial with fixed dispersion parameter has this property

Plots of mean-variance relationship



(a) edgeR's *plotMeanVar* can be used to explore the mean-variance relationship; each dot represents the estimated mean and variance for each gene, with binned variances as as the trended common dispersion overlaid.

(b) edgeR's *plotBCV* illustrates the relationship of biological coefficient of variation versus mean log CPM.

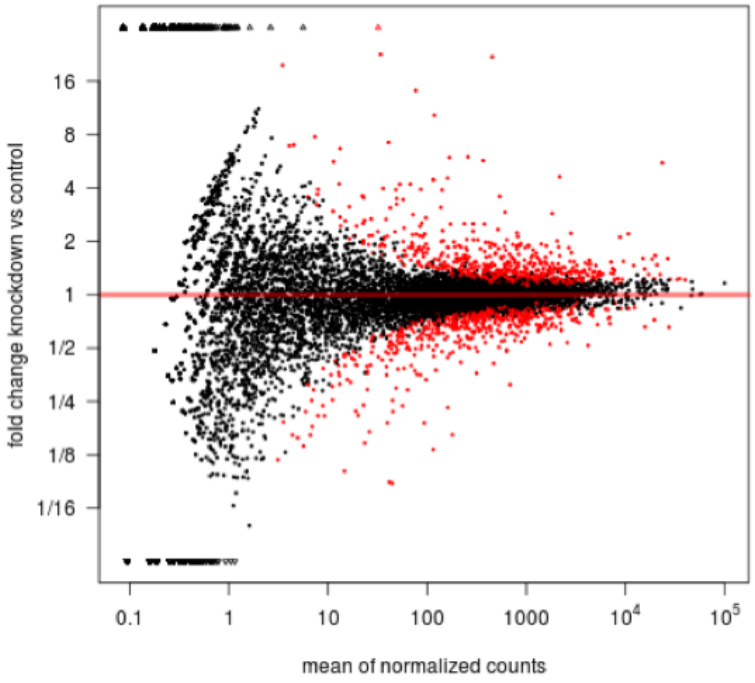
(c) DESeq's *plotDispEsts* shows the fit of dispersion versus mean. CPM, counts per million.

Variance Estimation & Shrinkage

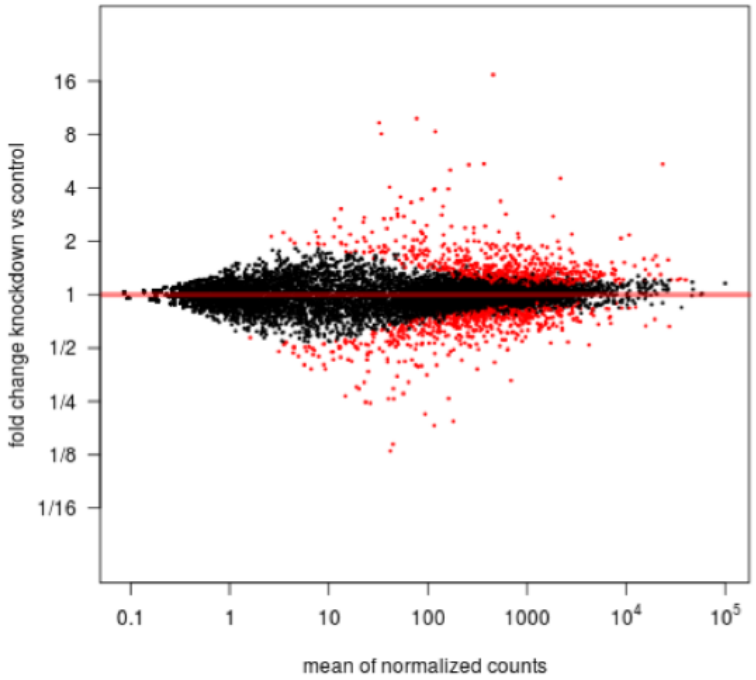
- A crucial input to the GLM procedure is the estimated strength of **within-group variability**.
- Sufficient **replicates** required to estimate variance and dispersion for each gene.
- Estimating variability from few samples requires information sharing across genes (**shrinkage**).
- Shrinkage can also regularize **fold-change estimates**. (New in DESeq2)
- This helps with interpretation, visualization, GSEA, clustering, ordination, etc.

Shrinkage estimation of effect sizes

Without shrinkage



With shrinkage



Simon Anders

Testing for Differential Expression

A comparison of methods for differential expression analysis of RNA-seq data

BMC Bioinformatics 2013, **14**:91 doi:10.1186/1471-2105-14-91

Charlotte Soneson (Charlotte.Soneson@isb-sib.ch)
Mauro Delorenzi (Mauro.Delorenzi@unil.ch)

Comprehensive evaluation of differential expression analysis methods for RNA-seq data

Franck Rapaport [1], Raya Khanin [1], Yupu Liang [1], Azra Krek [1], Paul Zumbo [2,4], Christopher E. Mason [2,4], Nicholas D. Socci [1], Doron Betel [3,4]

[1]Bioinformatics Core, Memorial Sloan-Kettering Cancer Center, New York
[2]Department of Physiology and Biophysics, Weill Cornell Medical College, New York
[3] Division of Hematology/Oncology, Department of Medicine, Weill Cornell Medical College, New York
[4] Institute for Computational Biomedicine, Weill Cornell Medical College, New York

January 24, 2013

DE on gene-level counts

Many tools using gene-level counts for DE:

- **edgeR**** } Top performers! (Soneson, 2013)
- **DESeq**** } Top performers! (Soneson, 2013)
- NPEBseq
- baySEQ
- BBSeq
- NoiSeq
- QuasiSeq

DE on gene-level counts

1. Filter Count Data
2. Estimate Normalisation factors
3. Estimate Dispersion
4. Specify the Experimental Design
5. Estimating Differential Representation

Downstream of DE:

- Gene Set Enrichment Analysis
- Clustering

DE on gene-level counts: step 1

Load matrix of counts:

http://www.bioconductor.org/help/course-materials/2013/useR2013/

```
> data(counts)
> dim(counts)
```

Create treatment group names from the column names of the counts matrix:

```
> grps <- factor(sub("[1-4].*", "", colnames(counts)), + levels=c("untreated", "treated"))
> pairs <- factor(c("single", "paired", "paired", "single", "single", "paired", "paired"))
> pData <- data.frame(Group=grps, PairType=pairs, + row.names=colnames(counts))
```

Use the edgeR package to create a DGEList object from the count and group data:

```
>library(edgeR)
> dge <- DGEList(counts, group=pData$Group)
> dge <- calcNormFactors(dge) #estimates relative library sizes for use as offsets in the generalized linear model.
```

Filter reads by scaling the counts by the library sizes and express the results on a per-million read scale:

```
#dividing each column by it's library size and multiplying by 1e6
> m <- sweep(dge$counts, 2, 1e6 / dge$samples$lib.size, `*`)

#gene is represented at a frequency of at least 1 read per million mapped (m > 1, below) in three or more samples (the n size of smallest group)
> ridx <- rowSums(m > 1) >= 3
> table(ridx)
> dge <- dge[ridx,]
```

```
#alternatively, convert to counts per million with function cpm
cpms = cpm(dge$counts)
keep = rowSums(cpms >1) >=3
dge = dge[keep,]
```

In edgeR, it is recommended to remove features without at least 1 read per million in *n* of the samples, where *n* is the size of the smallest group of replicates (here, *n* = 3 for the knockdown group).

DE on gene-level counts: steps 2, 3 & 4 (**standard** design)

Estimate normalization factors:

```
> dge = calcNormFactors(dge)
```

Estimate dispersion:

```
> dge <- estimateCommonDisp(dge)
> dge <- estimateTagwiseDisp(dge)
```

`estimateCommonDisp` is usually run before `estimateTagwiseDisp`

Create the model matrix for the experimental design:

```
> design <- model.matrix(~ Group, pData)
> design
            (Intercept) Grouptreated
treated1fb            1            1
treated2fb            1            1
treated3fb            1            1
untreated1fb          1            0
untreated2fb          1            0
untreated3fb          1            0
untreated4fb          1            0
attr(,"assign")
[1] 0 1
attr(,"contrasts")
attr(,"contrasts")$Group
[1] "contr.treatment"
```

The coefficient (column) labeled 'Intercept' corresponds to the first level of Group, i.e., 'untreated'.
The coefficient 'Grouptreated' represents the deviation of the treated group from untreated.

DE on gene-level counts: steps 2, 3 & 4 (**complex** design)

Estimate normalization factors:

```
> dge = calcNormFactors(dge)
```

Estimate dispersion for more complex design:

```
> Dge <- estimateGLMTrendedDisp(dge)
> Dge <- estimateGLMTagwiseDisp(dge)
```

For more complex designs, use `estimateGLMTrendedDisp()` and `estimateGLMTagwiseDisp()` (see Robinson et al. 2013)

Create the model matrix for the experimental design:

#include other factors expected to affect expression levels

```
> design <- model.matrix(~ PairType + Group, pData)
> colnames(design)[2:3] = c("LibraryLayoutSINGLE", "conditionKD")
> design
```

	(Intercept)	LibraryLayoutSINGLE	conditionKD
1	1	0	0
2	1	0	0
3	1	0	1
4	1	0	1
5	1	1	0
6	1	1	1
7	1	1	0

```
attr(,"assign")
[1] 0 1 2
attr(,"contrasts")
attr(,"contrasts")$LibraryLayout
[1] "contr.treatment"
attr(,"contrasts")$condition
[1] "contr.treatment"
```

DE on gene-level counts: step 5

Fit a generalized linear model to the data and experimental design, using the tagwise dispersion estimate:

```
> fit <- glmFit(dge, design)
```

#calculate a likelihood ratio test. The second coefficient captures the difference between treated and untreated groups, and the likelihood ratio test asks whether this term contributes meaningfully to the overall fit.

```
> lrTest <- glmLRT(fit, coef=2) #NB for complex design this is coef=3
> tt <- topTags(lrTest, n=10) #summarizes results across the experiment
> tt[1:3,]
```

```
Coefficient:  Grouptreated
                logFC   logCPM       LR        PValue           FDR
FBgn0039155 -4.697329 6.035726 564.1616 1.045516e-124 8.357851e-121
FBgn0029167 -2.233879 8.247571 247.1647  1.077884e-55  4.308302e-52
FBgn0034736 -3.499616 4.044214 232.5560  1.651719e-52  4.401281e-49
```

Sanity check: summarize the original data for the first several probes, confirming that the average counts of the treatment and control groups are substantially different.

```
> sapply(rownames(tt$table)[1:4], function(x) tapply(counts[x,], pData
$Group, mean))
```

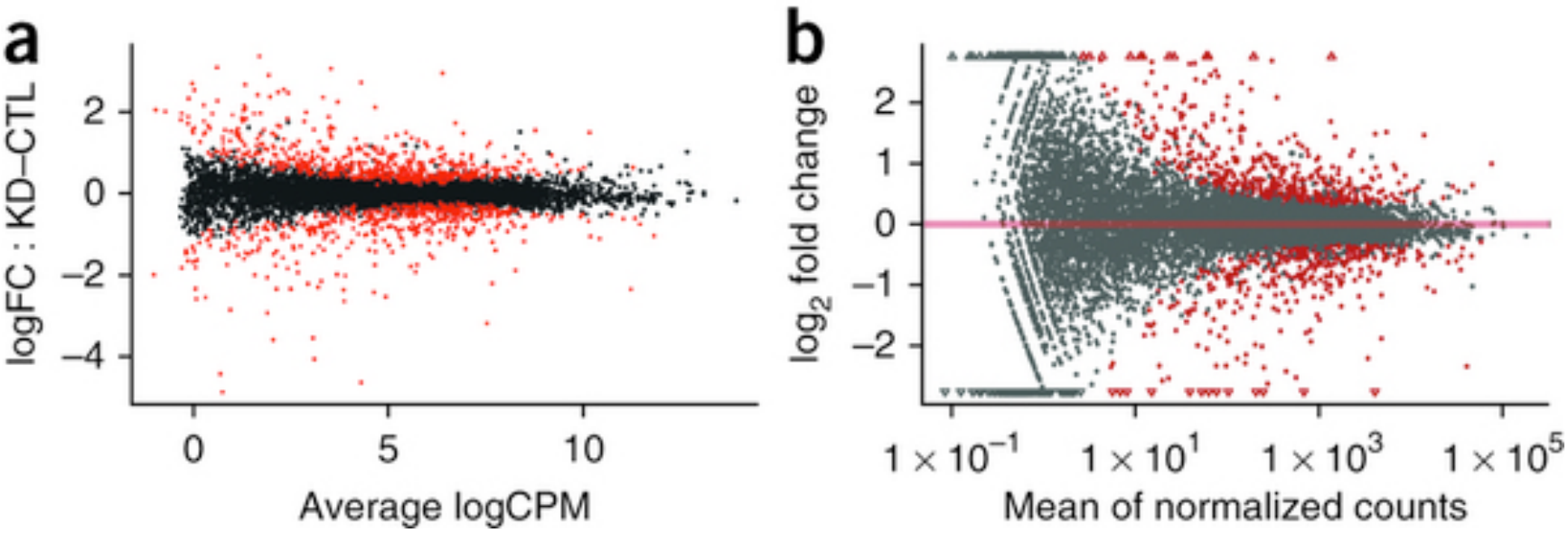
```
           FBgn0039155 FBgn0029167 FBgn0034736 FBgn0035085
untreated          1576    6447.000      382.25    994.2500
treated              64    1482.667       36.00    187.6667
```

M('minus') vs. A('add') plots for RNA-seq Data

displays differential expression (log-fold changes) vs. expression strength (log average read count)

edgeR's plotSmear() function

DESeq's plotMA() function



- Count data sets typically show a (left-facing) trombone shape, reflecting the higher variability of log ratios at lower counts.
- Points will typically be centered around a log ratio of 0 if the normalization factors are calculated appropriately

Robinson et al, Nature Protocols 2013

DE on exon/transcript-level counts

Various tools that use exon/transcript-level counts for DE:

- **DESeq**
- **edgeR**
- **DEXSeq+**
- **NPEBseq**

Exon-level estimates of expression

- BitSeq
- Cuffdiff*
- MMSEQ*

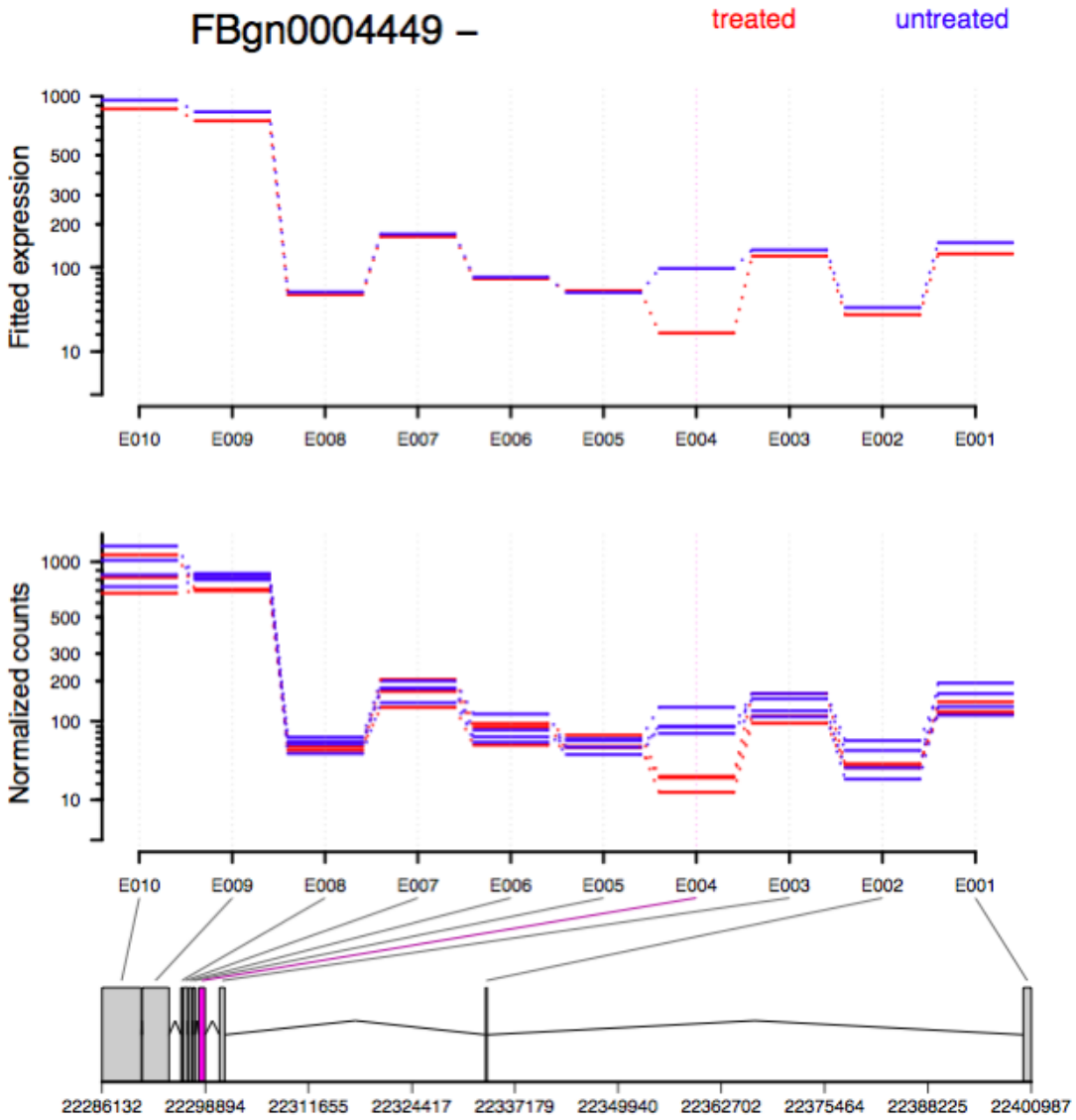
Transcript-level estimates of expression

*not available in R/Bioconductor

DE on exon/transcript-level counts

DEXseq:

- Within-gene differential expression approach.
- Tests for interaction between exon use and treatment.
- Knocking down the splicing factor ***pasilla*** affects the **fourth exon** (counting bin E004) of the gene Ten-m (CG5723).



Anders, Reyes & Huber (2012)

Enrichment Analysis

Enrichment Analysis

- Given the list of genes with strong effects in an experiment ("hits"):

  What do they mean?

- **Common approach**:
  - Take a collection of gene sets

    (e.g., GO, KEGG, Reactome, etc.)
  - look for sets that are enriched in hits.

Enrichment Analysis

Gene Sets from Databases of Biological Knowledge

- Biomedical literature
- Biochemical pathways
- Functional annotation
- Ontologies
- Sequence information
- Interaction data
- TF/regulatory information
- Experimental data

Suraj Menon

Enrichment Analysis

Two approaches:

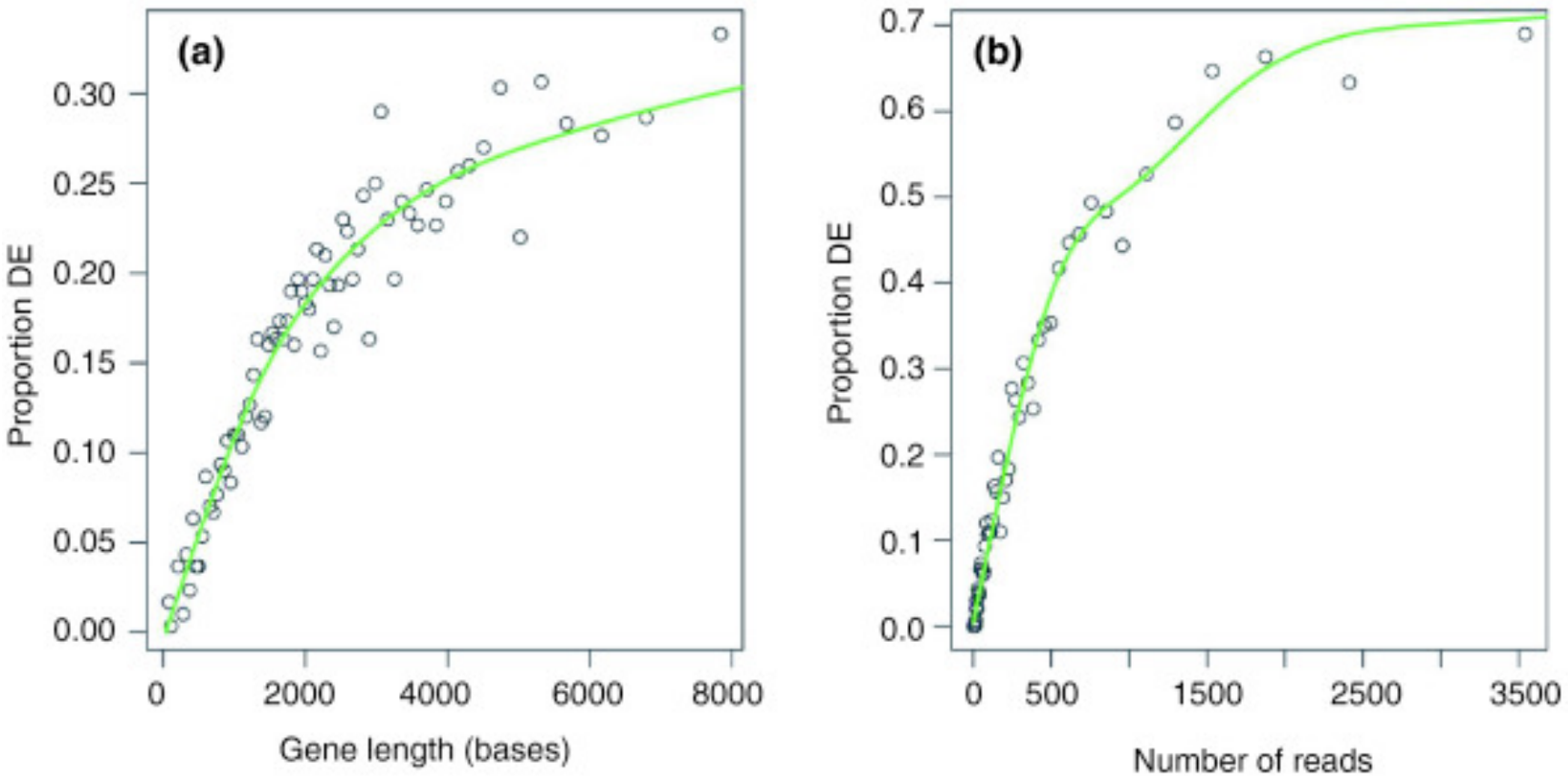
- **Categorical test:**
  - Over/under-representation analysis.
  - **Threshold-based list of genes** – usually by FDR cut-off DEGs e.g. P< 0.01
  - Is the gene set enriched for *significantly* differentially-expressed genes?
  - Bioconductor: `goseq`

- **Continuous test:**
  - Gene Set Enrichment Analysis (GSEA)
  - **Pre-ranked list of genes** – best to use estimated/moderated Fold Changes.
  - Are the fold changes of the genes in the set particularly strong?
  - Bioconductor: `limma`:
    - `WilcoxGST` & `geneSetTest`, the only ones that can be used as part of a negative binomial based analysis of count data.
    - NB roast and romer, make multivariate normal assumptions that are incompatible with count distributions.

DE, Gene Length and Read Count

- However, RNA-seq is affected by biases not present in microarray data.
- **Gene length bias** is an issue in RNA-seq data, in which longer genes have higher counts (at the same expression level)
- This results in greater statistical power to detect DE for long and highly expressed genes.

DE, Gene Length and Read Count



Goseq

Takes length bias into account - provides the probability that a gene will be DE based on its length alone (Probability Weighting Function, PWF)

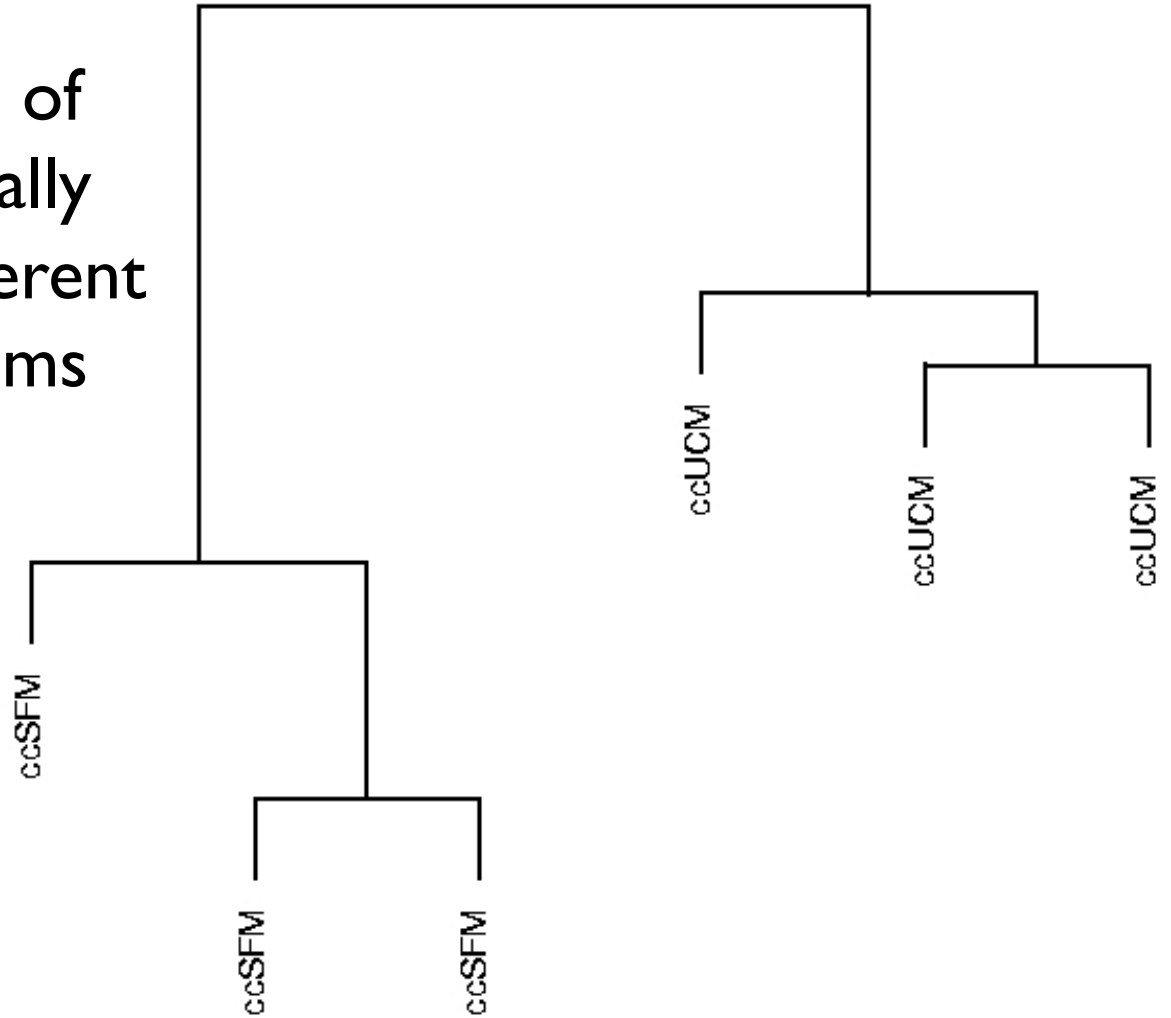
See vignette for more details and examples!

Young et al. *Genome Biology* (2010)

Data Clustering

Discovering groups

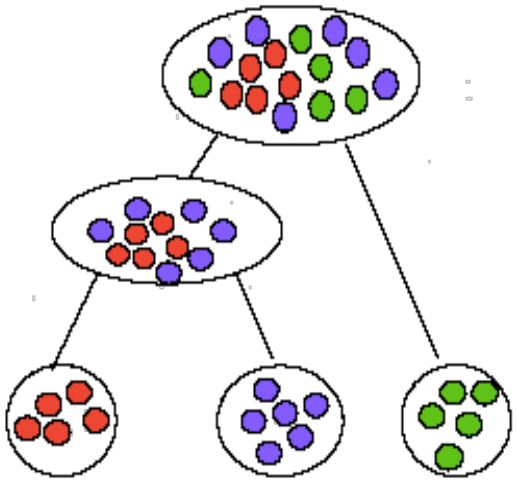
Replicate samples of cell lines accidentally grown in two different serum free mediums



Unsupervised Clustering Algorithms

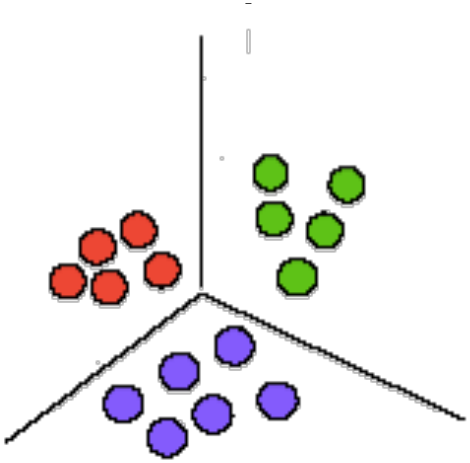
Heuristic

Hierarchical



Group genes that behave similarly under similar experimental conditions. Assume genes are functionally related. Separates genes & samples into mutually exclusive clusters.

Partitioning



Linearly decompose expression data into a dataset with a reduced number of dimensions, enabling the sets of genes that minimally discriminate the samples to be grouped and easily visualised.

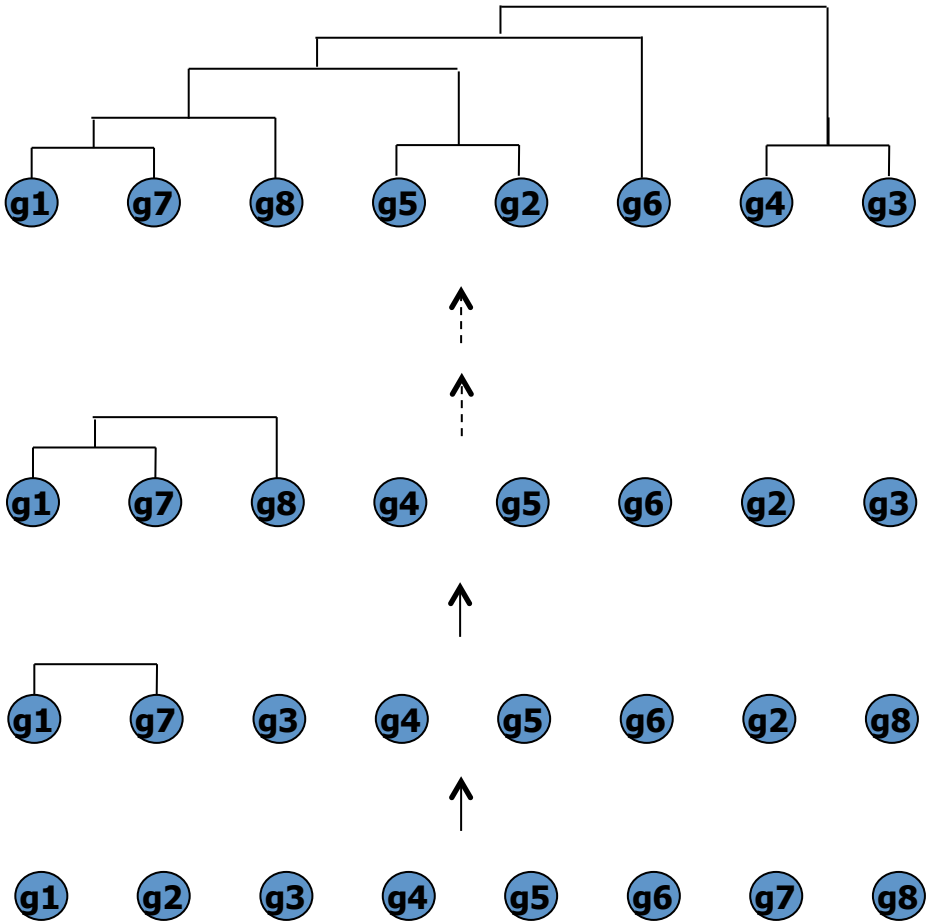
e.g. K-means; Self-Organised Maps (SOM)

Hierarchical clustering
Agglomerative '**bottom up**' Approach

3. The process continues until all genes (or samples) are linked in a single tree, with the length of the branches being an indication of the distance between each gene (or sample).

2. A combined measure of g1 and g7 is then used in subsequent comparisons, and g8 is the most similar etc.

1. A **similarity metric** determines that g1 and g7 have the closest expression profiles and groups these.



Agglomerative Hierarchical Clustering

R packages: `cclust, cluster, hclust`

Tree building:

determined to a large extent by the nature of the

- **similarity metrics** (distance between 2 data points)
- **linkage rules** (distance between 2 clusters)

R packages: `dist, bioDist, daisy`

Similarity/Dissimilarity Metric

- **Correlation coefficient:** *scale invariant*
  - Pearson's correlation:

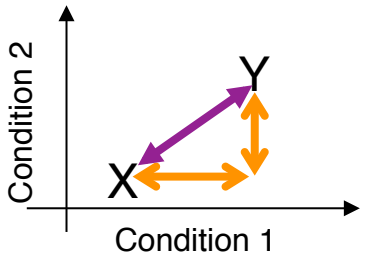
$$s(x_1, x_2) = \frac{\sum_{k=1}^{K} (x_{1k} - \bar{x}_1)(x_{2k} - \bar{x}_2)}{\sqrt{\sum_{k=1}^{K} (x_{1k} - \bar{x}_1)^2 \sum_{k=1}^{K} (x_{2k} - \bar{x}_2)^2}}$$

  - Spearman ρ: Pearson's correlation of ranks
  - Kendall's τ: probability of order concordance

$$d(X, Y) = \sqrt{\sum_i (x_i - y_i)^2}$$

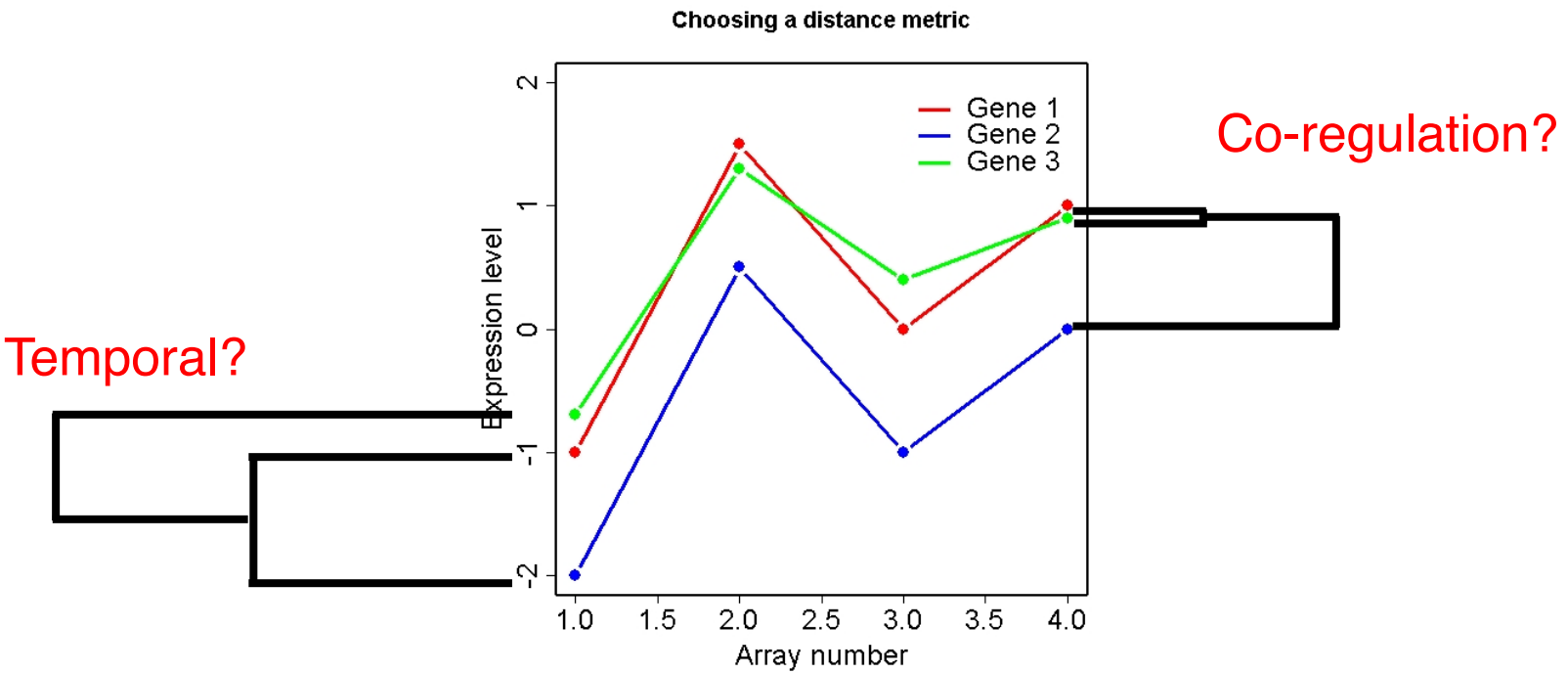
$$d(X, Y) = \sum_i |x_i - y_i|$$

- **Distance:** *scale dependent*
  - Euclidean distance
  - City block (Manhattan) distance
- Many others…



Distance (Euclidean) vs Correlation (Pearson)

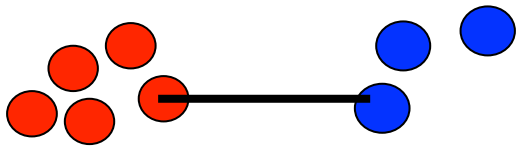
Think about the question you are asking



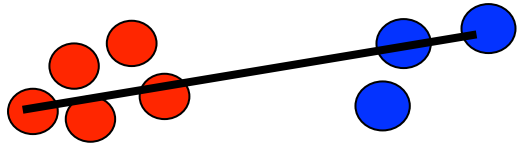
If shape is important, choose Pearson correlation
If level, choose Euclidean

Use **Pearson squared correlation** for genes that are anti-correlated

Between-cluster dissimilarity measures (Distance between clusters)

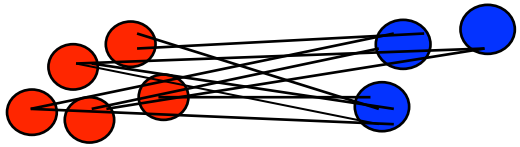


Single
(min. of pairwise distances)



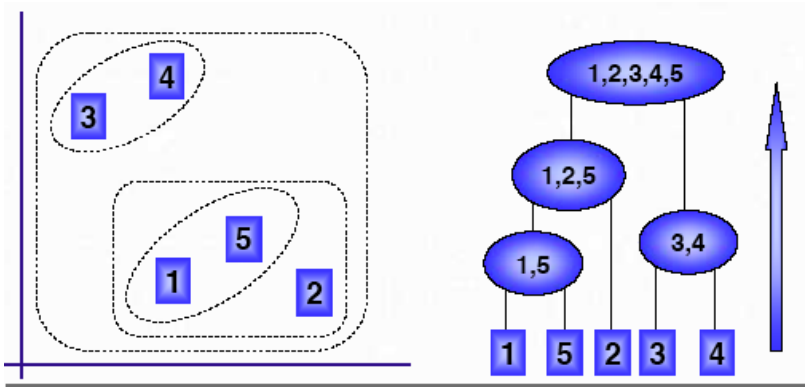
Complete
(max. of pairwise distances)

Distance between centroids



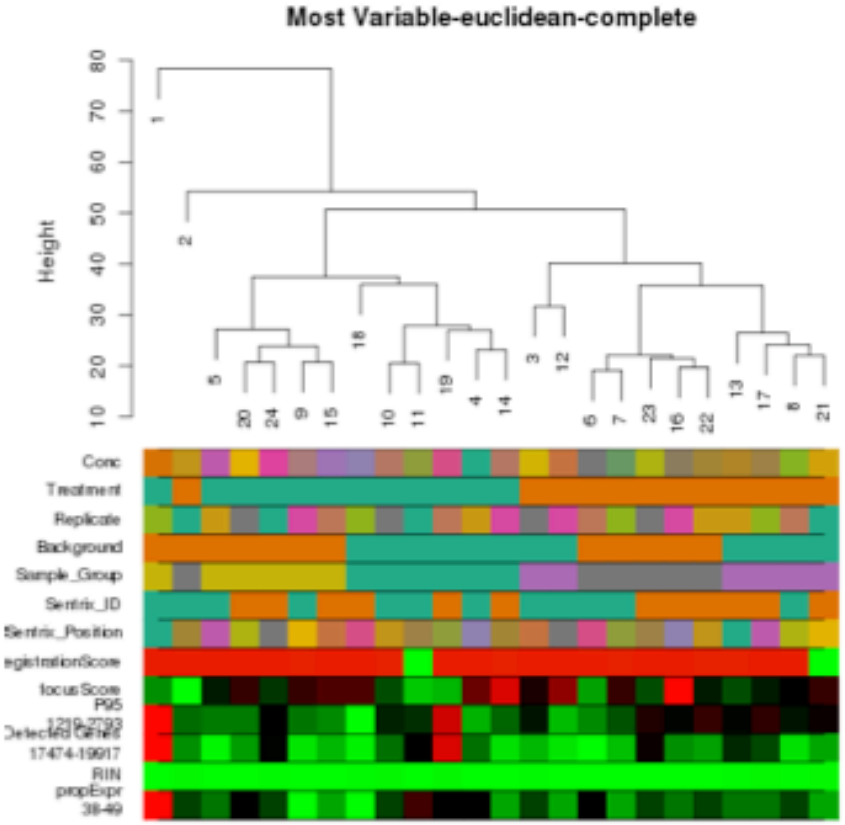
Average linkage
(mean of all pairwise distances)

Hierarchical Clustering: Dendrograms



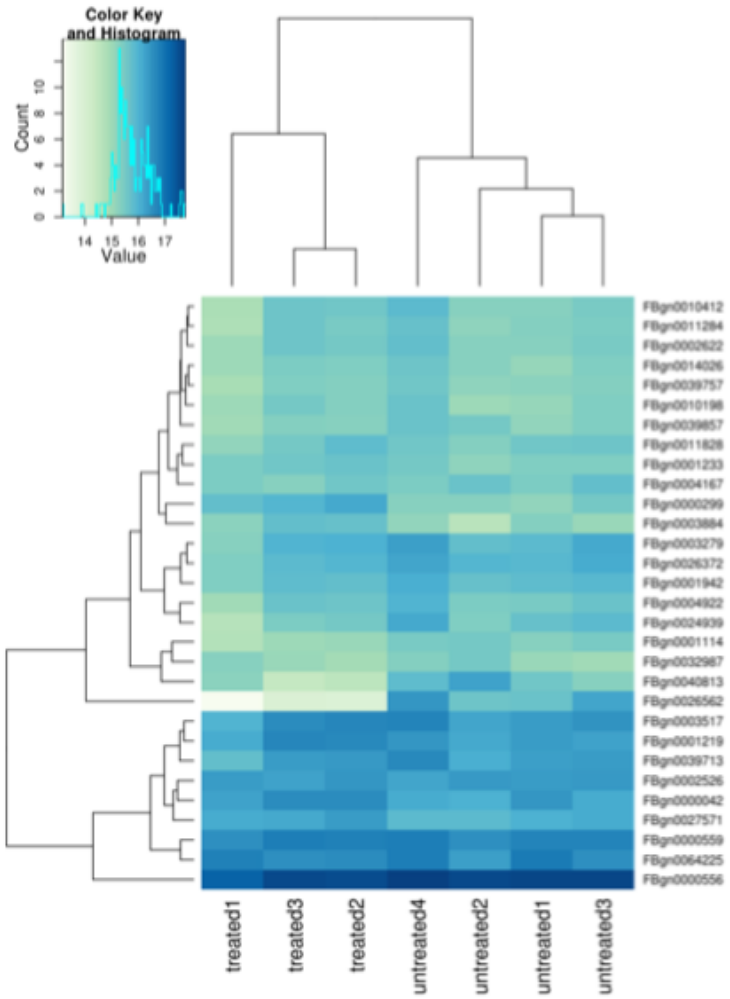
- Appealing because of apparent ease of interpretation, but they can be misleading.
- Dendrograms are good visual guides but **arbitrary**!
  - Nodes can be reordered: 2^{n-1} different dendrograms
  - Closer on dendogram ≠ more similar.
- Dendrograms **impose** structure on the data, instead of **revealing** structure in these data.

Using Clustering for QC

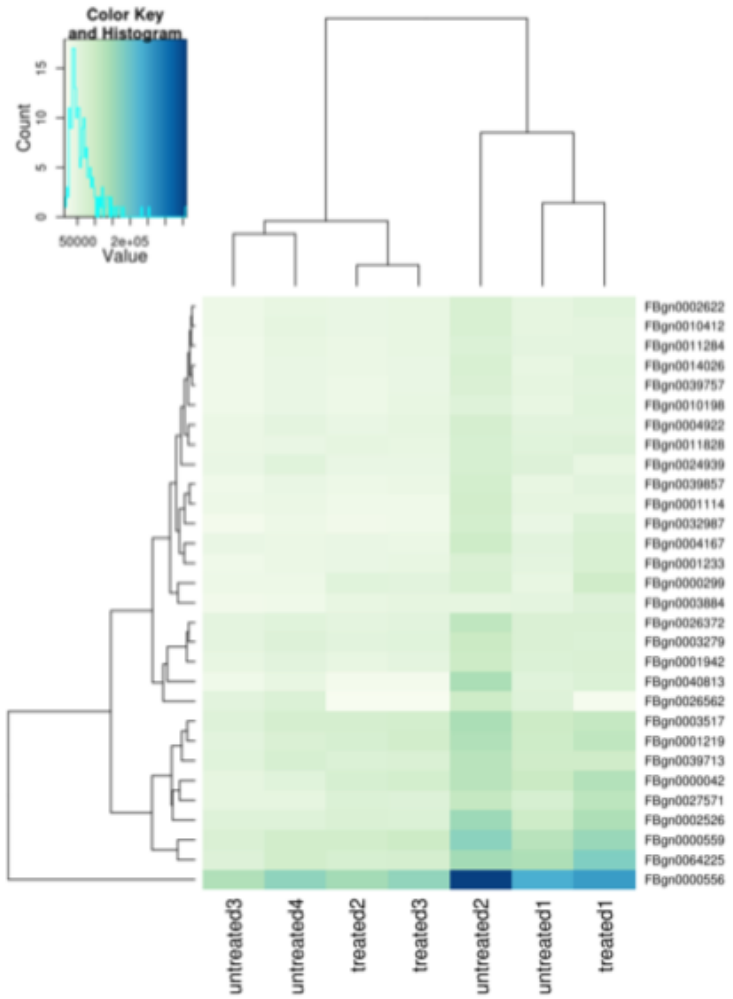



```
library(WGCNA)
d <- dist(t(eset), method = euclidean,])
dend <- hclust(d, method = complete)
plotDendroAndColors(dend, col_matrix,groupLabels=colnames(col_matrix))
```

Heatmaps



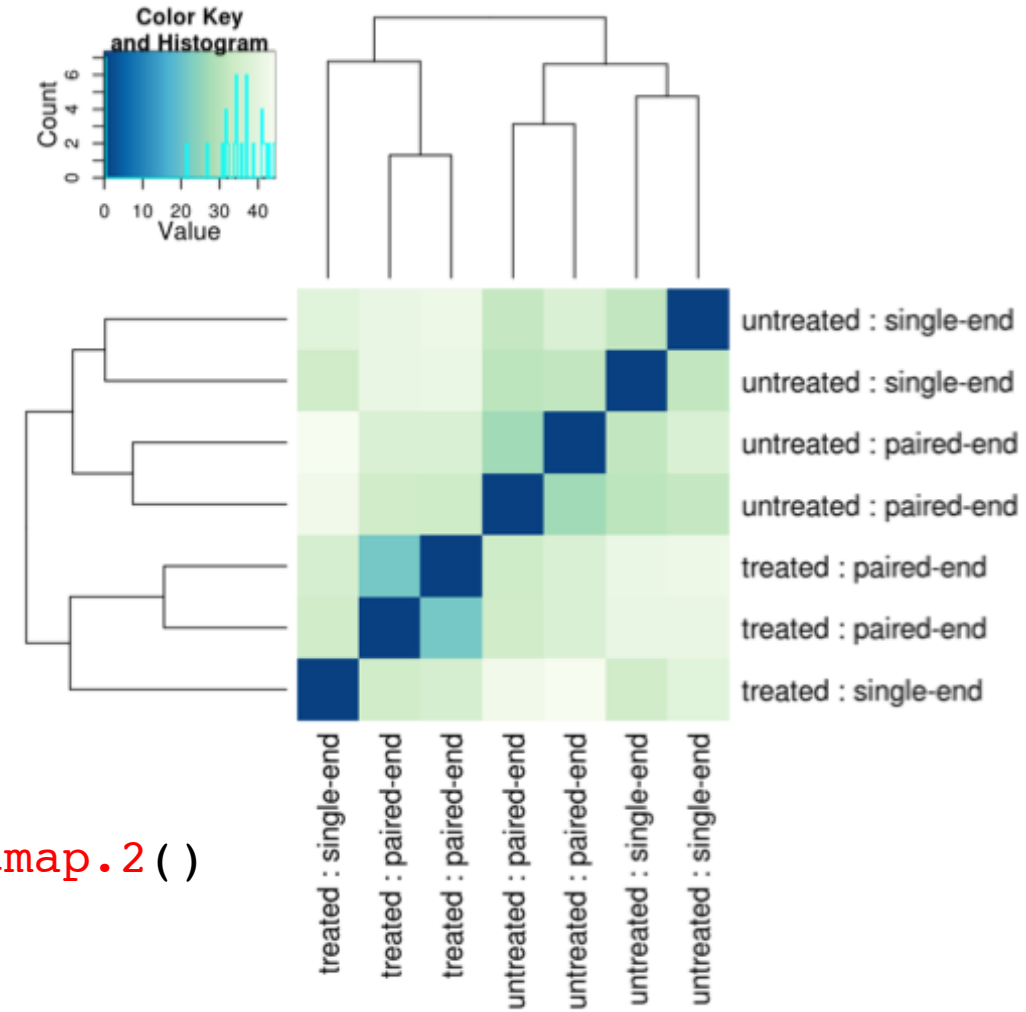
variance stabilisation transformed data
(vsdFull)



original count data
(cdsFull)

`heatmap.2()`

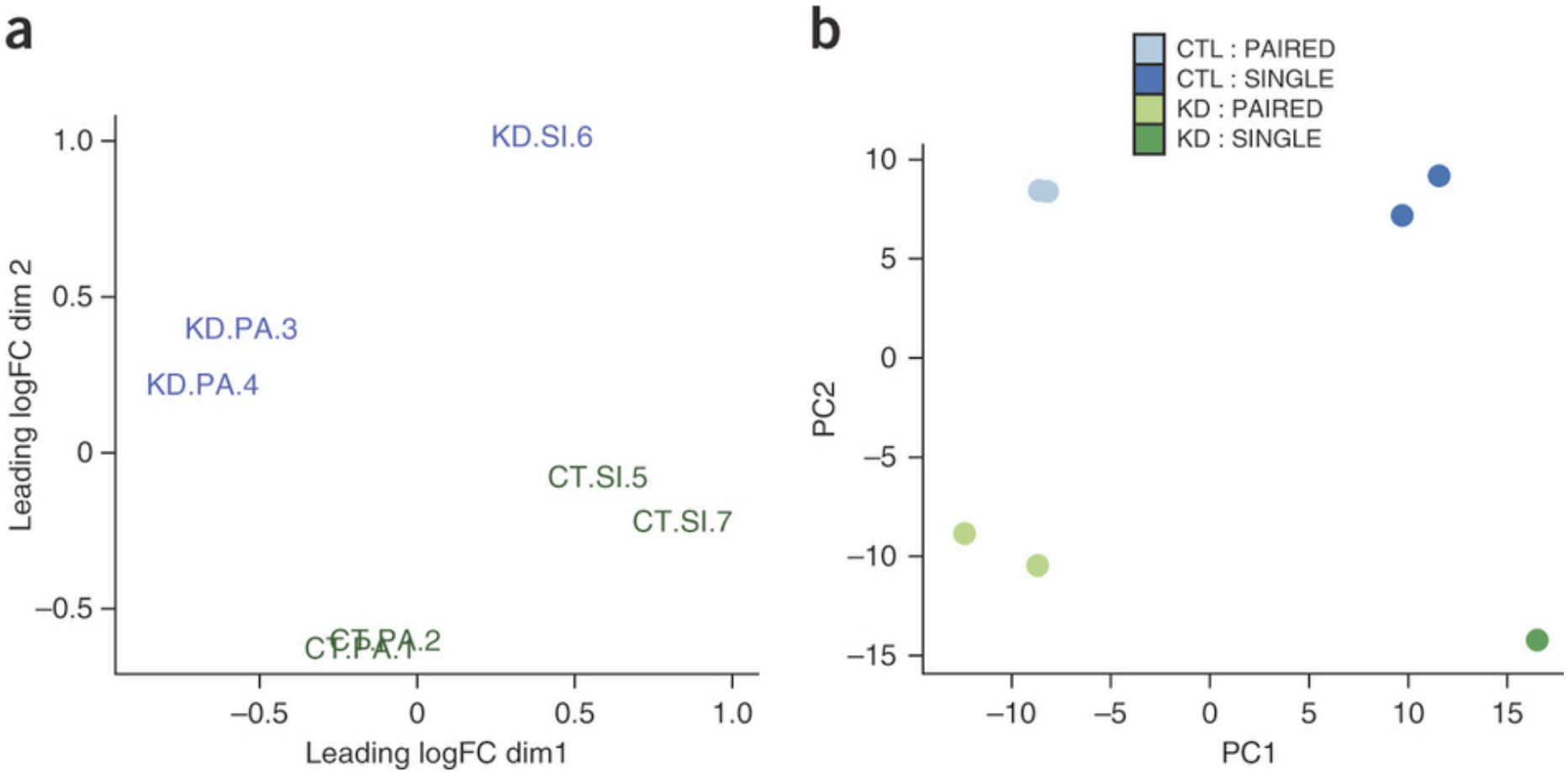
Heatmaps



`heatmap.2()`

Heatmap showing the Euclidean distances between the samples as calculated from the variance stabilising transformation of the count data.

PCA plot of samples



(a) By using a count-specific distance measure, **edgeR**'s *plotMDS* produces a multidimensional scaling plot showing the relationship between all pairs of samples

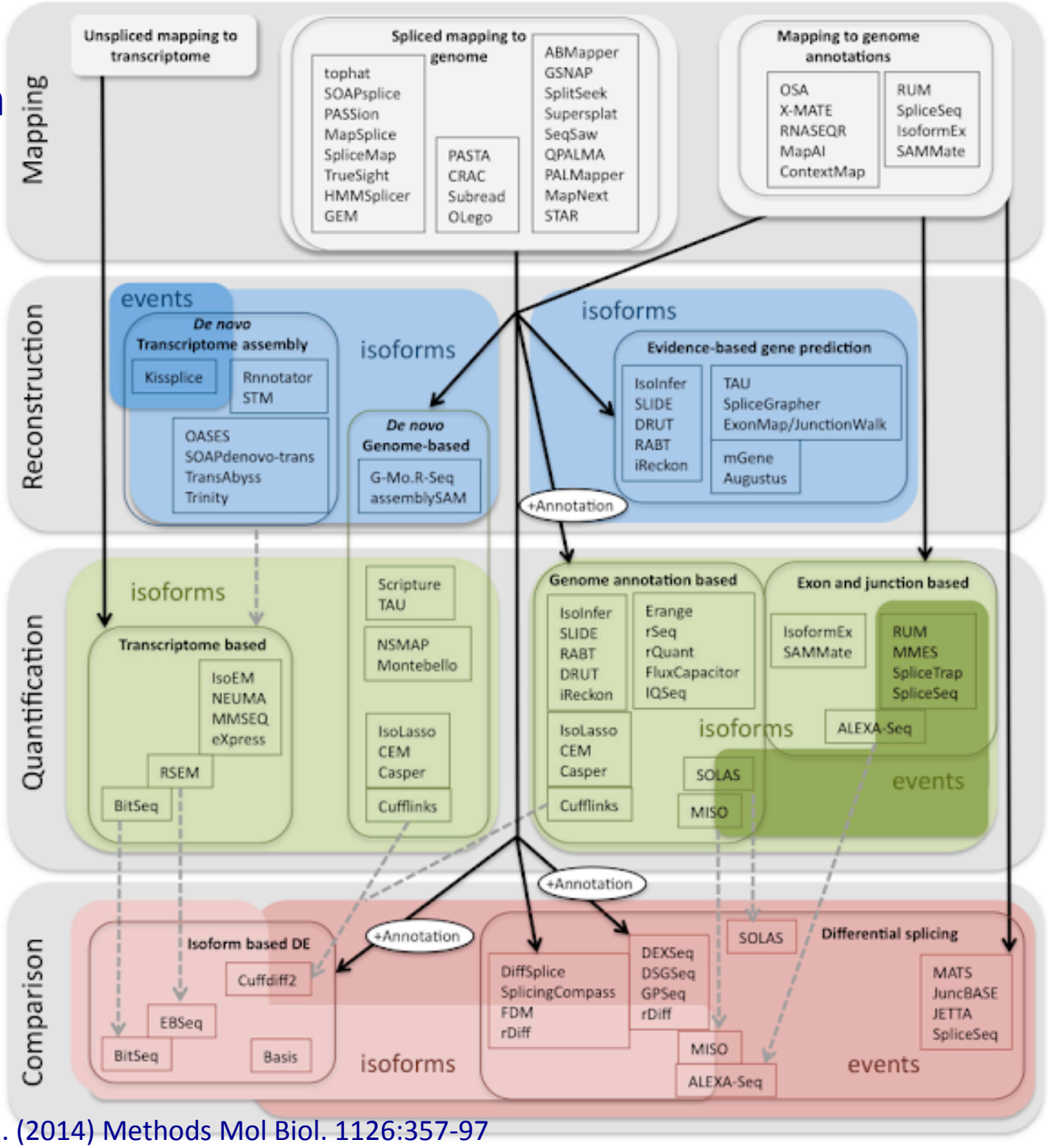
(b) DESeq's *plotPCA* makes a principal component (PC) plot of VST (variance-stabilizing transformation)-transformed count data. CT or CTL, control; KD, knockdown.

Splicing?

Overview of Methods To Study Splicing from RNA-seq Data

Entry point depends on specific analysis.

1. Assignment of the sequencing reads to their likely gene of origin.
2. Recovering the sequence of splicing events and isoforms.
3. Quantification of events and isoforms.
4. Providing an isoform or event view of differential splicing or expression.
5. Visualizing splicing regulation.



Eduardo Eryas' Group: Alamancos et al. (2014) Methods Mol Biol. 1126:357-97

RNA-seq Practical: edgeR