Analysis of bulk RNA-seq II: Reads to DGE

Analysis of Next-Generation Sequencing Data

Friederike Dündar

Applied Bioinformatics Core

Slides at https://bit.ly/2T3sjRg[1]

February 25, 2020

Weill Cornell Medicine

[1] https://physiology.med.cornell.edu/faculty/skrabanek/lab/angsd/schedule_2020/

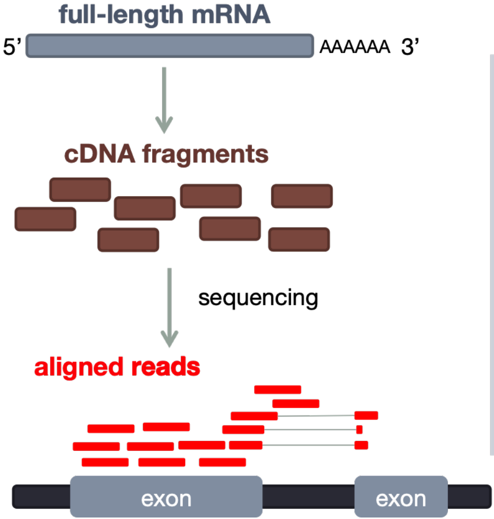
1. Gene expression quantification recap
2. Normalization of read counts
3. Exploratory analyses
4. Differential gene expression
5. Downstream analyses
6. References

Many slides today were inspired or directly taken from the excellent book **Data Analysis for the Life Sciences** by Rafael Irizarry and Michael Love, and training material developed by the **Harvard Chan Bioinformatics Core**.

Go and check them out for even more details! The Harvard Chan Bioinformatics Core's material can be found at their github page: https://github.com/hbctraining/DGE_workshop

Gene expression quantification recap

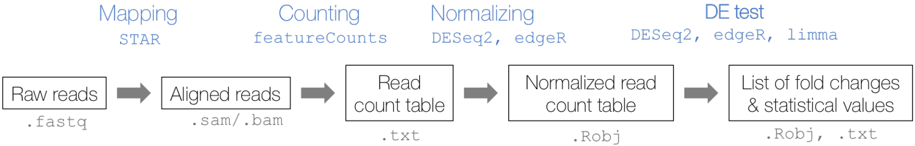
Alignment of NGS data is resource-intensive



Particular challenges of Illumina sequencing:

- the query sequences (= reads) are very short
- there are millions of them!
- cannot expect 100% exact matches
  - ➢ seq. errors
  - ➢ biological variation
  - ➢ reference errors
- RNA-seq: some cDNA fragments can only be aligned if one allows for gigantic gaps (= introns)

Quantification of gene expression



Mapping
STAR

Counting
featureCounts

Normalizing
DESeq2, edgeR

DE test
DESeq2, edgeR, limma

Raw reads
.fastq

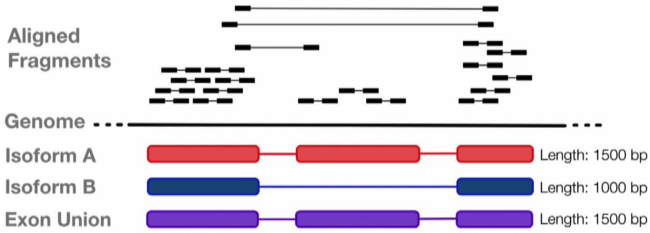
Aligned reads
.sam/.bam

Read count table
.txt

Normalized read count table
.Robj

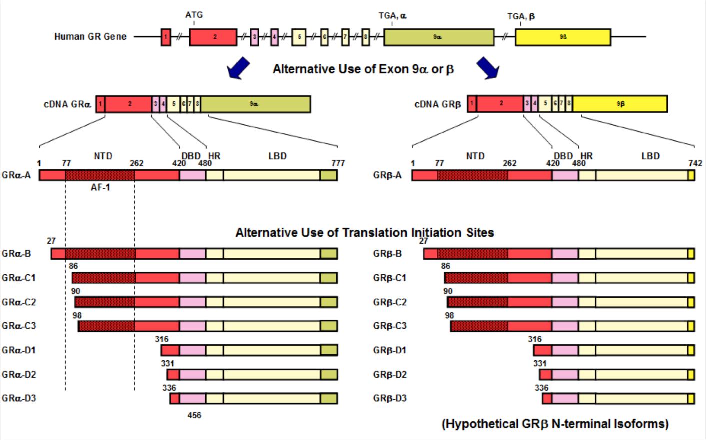
List of fold changes & statistical values
.Robj, .txt

1. **Align**
   - with splice-aware alignment tools! e.g. STAR
2. **Count** reads that overlap with annotated genes
   - complicated by alternative isoforms: **genes != transcripts**



Alternative isoforms are common in eukaryotic transcriptomes

Gene isoforms = mRNAs produced from the *same locus*, but with different final sequences (possibly giving rise to different protein sequences, too)



https://www.endotext.org/chapter/glucocorticoid-receptor

2 Philosophies of gene expression quantification

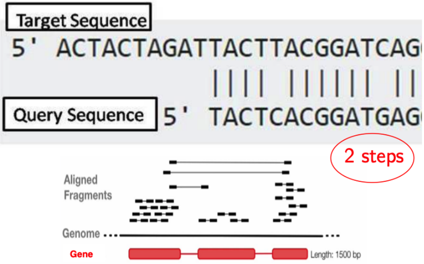
(A) Alignment + counting

Historically, the reads of RNA-seq experiments were treated the same way as reads of DNA-seq experiments, i.e. it was deemed important that we knew the precise location that each read had originated from.

The results of alignment, however, are not inherently quantitative, which is why a 2nd counting step was needed.

alignment followed by **counting** of reads overlapping with features

e.g. `STAR + featureCounts`



2 steps

2 Philosophies of gene expression quantification

(B) Pseudoalignment

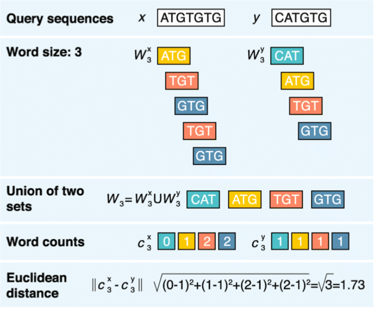
For standard bulk RNA-seq, we really just want the **number of reads** that are **compatible with a known transcript sequence**. If we decide to not care about the precise genome location, we can:

- reduce the size of our search space, i.e. our index of k-mers can be limited to cDNAs (no introns!)
- chop up the reference cDNAs AND our reads into fairly small k-mers
- perform a "simple" k-mer matching strategy and assign the read to the transcript that most of its k-mers matched to

See Zielezinski et al. [2017] for a good explanation of pseudo-alignment etc.

estimating expression levels of individual isoforms/genes based on **alignment-free k-mer matching**

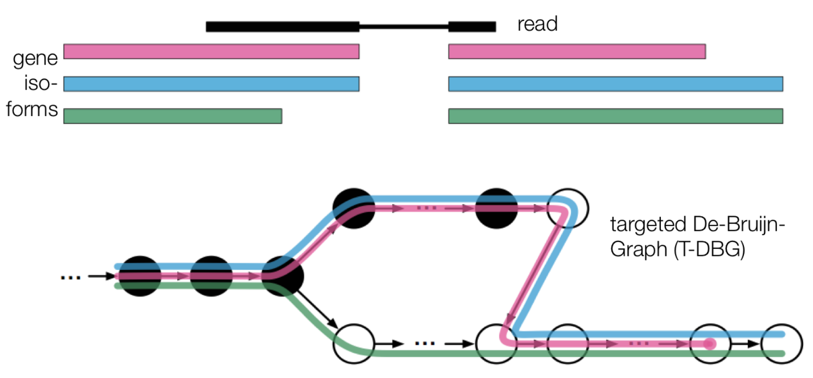
➔ `salmon, kallisto`



Zielezinski et al. (2017)

2 Philosophies of gene expression quantification

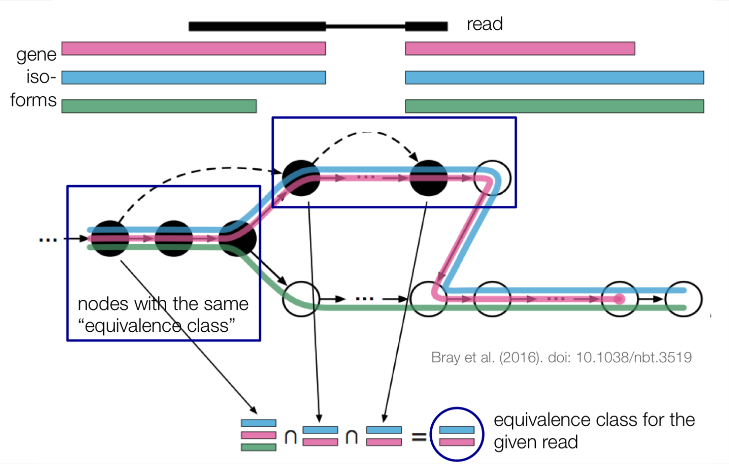
(B) Transcript abundance estimation via pseudoalignment



Bray et al. (2016). doi: 10.1038/nbt.3519
http://tinyheero.github.io/2015/09/02/pseudoalignments-kallisto.html

2 Philosophies of gene expression quantification

(B) Transcript abundance estimation via pseudoalignment

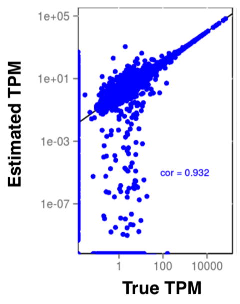


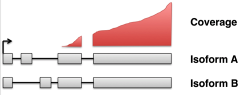
Bray et al. (2016). doi: 10.1038/nbt.3519

2 Philosophies of gene expression quantification

(B) Pseudoalignment caveats

- abundance estimates for **lowly expressed** transcripts are highly variable (not enough distinct k-mers)
- **short RNAs** have inherently fewer distinct k-mers
- problem when coverage of an isoform-defining region is low (or its sequence isn't distinct)
- any read that originated from somewhere else in the genome than cDNAs may be mapped spuriously





For very similar transcripts, collapsing all abundances per gene into a **gene-centric measure** is more robust and accurate. [Soneson et al., 2015]

2 Philosophies of gene expression quantification

(B) Transcript abundance estimates

If you decide to use abundance estimates rather than gene-read overlap counts, use the `tximport` package [Soneson et al., 2015] package for their use with Bioconductor differential gene expression packages.

The advantages of using the transcript abundance quantifiers **in conjunction with tximport to produce gene-level count matrices** and normalizing offsets, are:

- in-built correction for any potential changes in gene length across samples (e.g. from differential isoform usage) [Trapnell et al., 2012]
- increased speed and less memory and less disk usage compared to alignment-based methods
- it is possible to avoid discarding fragments that can align to multiple genes with homologous sequence

2 Philosophies of gene expression quantification

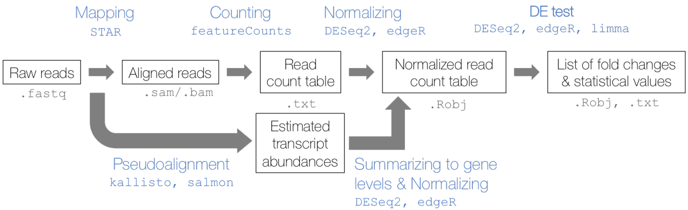
	Traditional	**Pseudoalignment**
Ex. workflow:	`STAR` + `featureCounts`	`kallisto` or `salmon`
Read mapping based on:	**Where** does a read match best?	Which **collection of unique k-mers** does a read match best?
Reference:	**Genome** seq. + exon boundaries	**cDNA** sequences
Mapping result:	Genome coordinates (`BAM`)	Table of expression level estimates (`txt`)
Expression quantification:	Counting how many reads *overlap* a gene[2].	Summing the values assigned to each collection of unique k-mers (equivalence class).
Output:	Read counts (integers)	Estimated transcript abundances (numeric)
Speed:	++ and +++	++++

[2] The read sequence is irrelevant at this point.

General bioinformatics workflow – updated

Understand your null hypothesis! (See Soneson et al. [2015], Love et al. [2018])

- **DGE**: Differential **Gene** Expression
  - Has the **total ouput** of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by DESeq2/edgeR/limma (see M. Love's protocols
- **DTU**:: Differential **Transcript** Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by DEXSeq (!)



Mapping
STAR

Counting
featureCounts

Normalizing
DESeq2, edgeR

DE test
DESeq2, edgeR, limma

Raw reads
.fastq

Aligned reads
.sam/.bam

Read count table
.txt

Normalized read count table
.Robj

List of fold changes & statistical values
.Robj, .txt

Pseudoalignment
kallisto, salmon

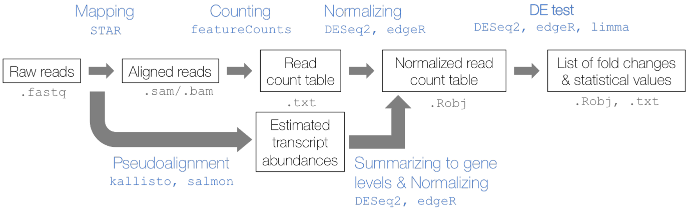
Estimated transcript abundances

Summarizing to gene levels & Normalizing
DESeq2, edgeR

General bioinformatics workflow – updated

Understand your null hypothesis! (See Soneson et al. [2015], Love et al. [2018])

- **DGE**: Differential **Gene** Expression
  - Has the **total ouput** of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by `DESeq2/edgeR/limma` (see M. Love's protocols
- **DTU**:: Differential **Transcript** Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by `DEXSeq` (!)



Mapping `STAR`

Counting `featureCounts`

Normalizing `DESeq2, edgeR`

DE test `DESeq2, edgeR, limma`

Raw reads `.fastq`

Aligned reads `.sam/.bam`

Read count table `.txt`

Normalized read count table `.Robj`

List of fold changes & statistical values `.Robj, .txt`

Pseudoalignment `kallisto, salmon`

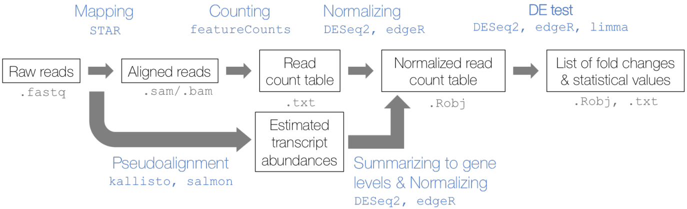
Estimated transcript abundances

Summarizing to gene levels & Normalizing `DESeq2, edgeR`

General bioinformatics workflow – updated

Understand your null hypothesis! (See Soneson et al. [2015], Love et al. [2018])

- **DGE**: Differential **Gene** Expression
  - Has the **total ouput** of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by `DESeq2/edgeR/limma` (see M. Love's protocols
- **DTU**:: Differential **Transcript** Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by `DEXSeq` (!)



Mapping
`STAR`

Counting
`featureCounts`

Normalizing
`DESeq2, edgeR`

DE test
`DESeq2, edgeR, limma`

Raw reads
`.fastq`

Aligned reads
`.sam/.bam`

Read count table
`.txt`

Normalized read count table
`.Robj`

List of fold changes & statistical values
`.Robj, .txt`

Pseudoalignment
`kallisto, salmon`

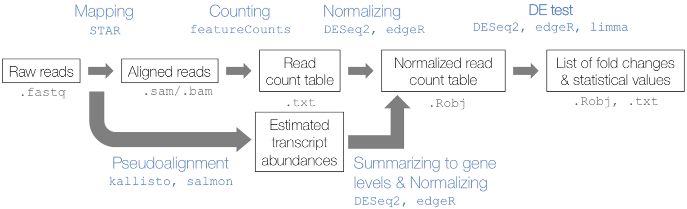
Estimated transcript abundances

Summarizing to gene levels & Normalizing
`DESeq2, edgeR`

General bioinformatics workflow – updated

Understand your null hypothesis!(See Soneson et al. [2015], Love et al. [2018])

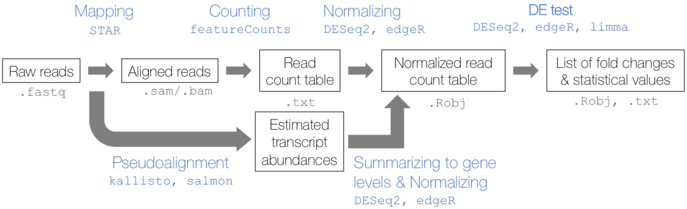
- **DGE**: Differential **Gene** Expression
  - Has the **total ouput** of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by DESeq2/edgeR/limma (see M. Love's protocols
- **DTU**:: Differential **Transcript** Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by DEXSeq (!)



General bioinformatics workflow – updated

Understand your null hypothesis!(See Soneson et al. [2015], Love et al. [2018])

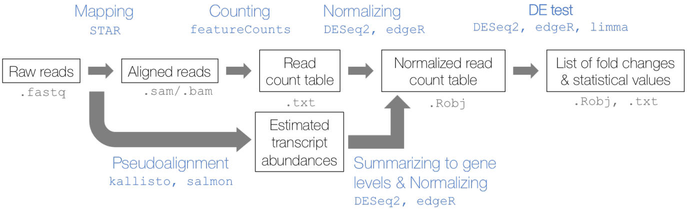
- **DGE**: Differential **Gene** Expression
  - Has the **total ouput** of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by `DESeq2/edgeR/limma` (see M. Love's protocols
- **DTU**:: Differential **Transcript** Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by `DEXSeq` (!)



General bioinformatics workflow – updated

Understand your null hypothesis! (See Soneson et al. [2015], Love et al. [2018])

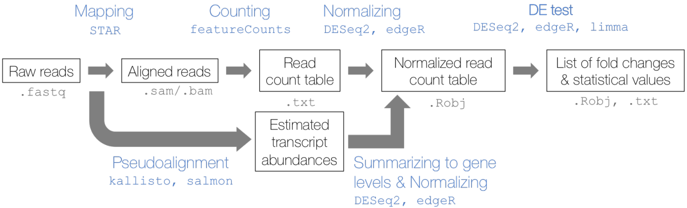
- **DGE**: Differential **Gene** Expression
  - Has the **total ouput** of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by `DESeq2/edgeR/limma` (see M. Love's protocols
- **DTU**:: Differential **Transcript** Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by `DEXSeq` (!)



General bioinformatics workflow – updated

Understand your null hypothesis!(See Soneson et al. [2015], Love et al. [2018])

- **DGE**: Differential **Gene** Expression
  - Has the **total ouput** of a gene changed?
  - input for the statistical testing: (estimated) counts per gene used by `DESeq2/edgeR/limma` (see M. Love's protocols
- **DTU**:: Differential **Transcript** Usage
  - Has the isoform composition for a given gene changed? I.e. are there different dominant isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by `DEXSeq` (!)



Normalization of read counts

Read counts are influenced by numerous factors, not just expression strength

Raw counts[3] = number of reads (or fragments) overlapping with the union of exons of a gene.

Raw count numbers are not just a reflection of the actual number of captured transcripts!

They are strongly influenced by:

- sequencing depth
- gene length
- DNA sequence content (% GC)
- expression of all other genes in the same sample

[3] also true for "estimated" gene counts from pseudoaligners

Read counts are influenced by numerous factors, not just expression strength

Raw counts[3] = number of reads (or fragments) overlapping with the union of exons of a gene.

Raw count numbers are not just a reflection of the actual number of captured transcripts!

They are strongly influenced by:

- sequencing depth
- gene length
- DNA sequence content (% GC)
- expression of all other genes in the same sample

[3] also true for "estimated" gene counts from pseudoaligners

Read counts are influenced by numerous factors, not just expression strength

Raw counts[3] = number of reads (or fragments) overlapping with the union of exons of a gene.

Raw count numbers are not just a reflection of the actual number of captured transcripts!

They are strongly influenced by:

- sequencing depth
- gene length
- DNA sequence content (% GC)
- expression of all other genes in the same sample

[3] also true for "estimated" gene counts from pseudoaligners

Read counts are influenced by numerous factors, not just expression strength

Raw counts[3] = number of reads (or fragments) overlapping with the union of exons of a gene.

Raw count numbers are not just a reflection of the actual number of captured transcripts!

They are strongly influenced by:

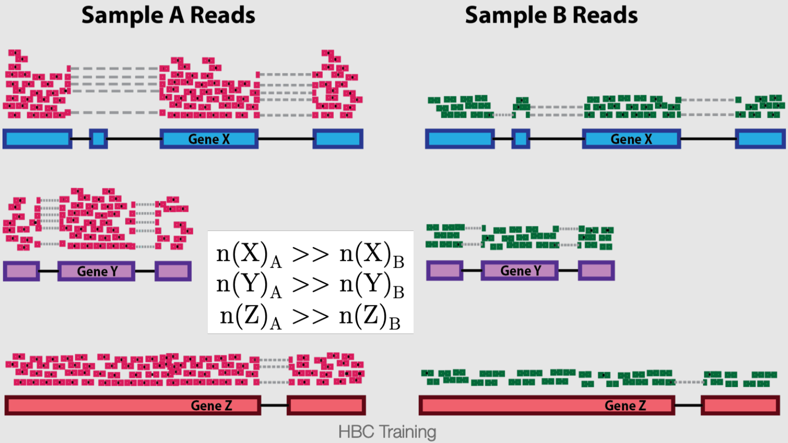
- sequencing depth
- gene length
- DNA sequence content (% GC)
- expression of all other genes in the same sample

[3] also true for "estimated" gene counts from pseudoaligners

Influences on read count numbers

1. Sequencing depth (= total number of reads per sample)

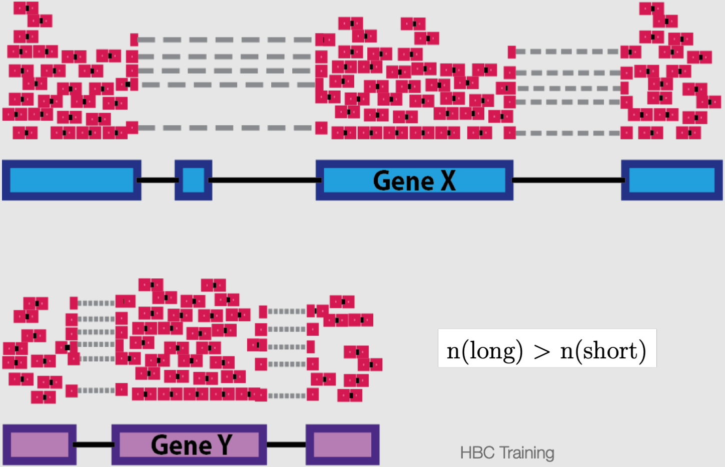
sequencing depth of Sample A $\gg$ Sample B



$$n(X)_A >> n(X)_B$$
$$n(Y)_A >> n(Y)_B$$
$$n(Z)_A >> n(Z)_B$$

Influences on read count numbers

2. Gene length (and GC bias)

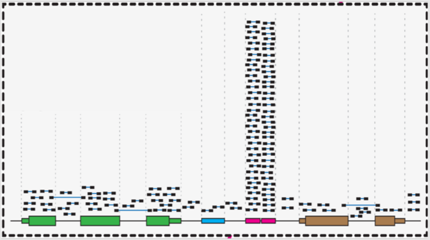


$n(\text{long}) > n(\text{short})$

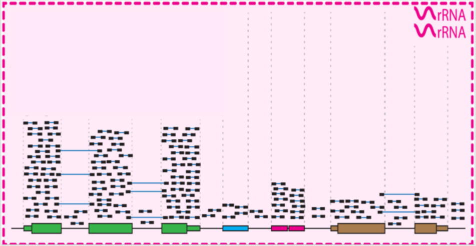
HBC Training

Influences on read count numbers

3. RNA composition - individual gene abundances



very highly expressed transcript soaks up significant portion of the reads reducing the range of read counts available for other transcripts



in the absence of that highly expressed transcript, the remaining transcripts' expression differences become more clear

All the numbers within a given sample are *relative* abundance measurements.

Influences on read count numbers - summary

- gene length
- transcript sequence (% GC)

need to be corrected when comparing different **genes**

- sequencing depth
- expression of all other genes within the same sample

need to corrected when comparing the same gene between different **samples**

Which biases are relevant for comparing different samples?

Different units for expression values

- **Raw counts**: number of reads/ fragments overlapping with the union of exons of a gene

$$X_i$$

- **[RF]PKM**: Reads/Fragments per Kilobase of gene per Million reads mapped – AVOID!

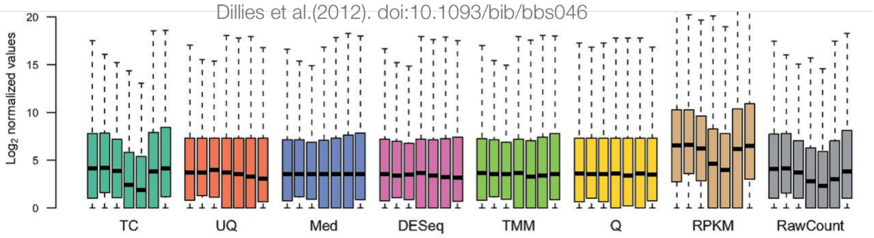
$$RPKM_i = \frac{X_i}{\underbrace{(\frac{l_i}{10^3})}_{\text{gene length}}\underbrace{(\frac{N}{10^6})}_{\text{seq. depth}}}$$

- **TPM**: Transcripts Per Million

$$TPM_i = \underbrace{\frac{X_i}{l_i}}_{\text{gene read counts per bp}} * \frac{1}{\underbrace{\sum_j \frac{X_j}{l_k}}_{\textit{all}\text{ gene counts over }\textit{all}\text{ gene bp}}} * 10^6$$

- **rlog**: log2-transformed count data normalized for small counts and library size (DESeq2)

Why not RPKMs?



- [RF]PKM values are not comparable between samples – Do NOT use them!
- if you need normalized expression values for exploratory plots, use TPM or DESeq2's rlog values

Working with read counts

- Download the featureCounts results to your laptop.
- Read the featureCounts results into R.
- Let's normalize!

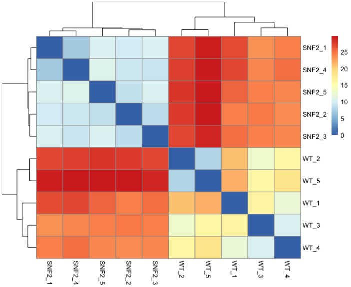
Exploratory analyses

Exploratory analyses

Exploratory analyses **do not test a null hypothesis**! They are meant to familiarize yourself with the data to discover biases and unexpected variability!

Typical exploratory analyses:

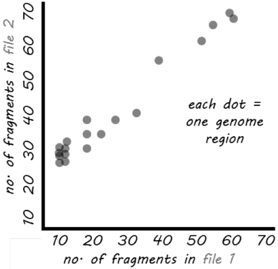
- **correlation** of gene expression between different samples
- (hierarchical) **clustering**
- **dimensionality reduction** methods, e.g. PCA
- dot plots/**box plots**/violin plots of individual genes

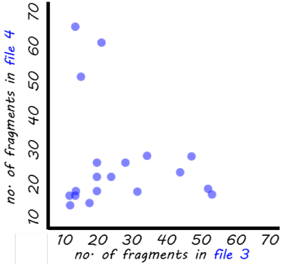


Use **normalized and transformed** read counts for data exploration!

Pairwise correlation of gene expression values

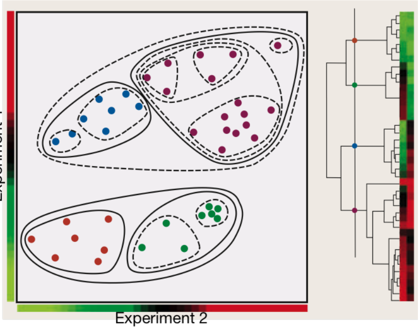
- replicates of the same condition should show high correlations (>0.9)
- **Pearson** method: *metric* differences between samples
  - influenced by outliers
  - covariance of two variables divided by the product of their standard deviation
  - suitable for normally distributed values
- **Spearman** method: based on *rankings*
  - less sensitive
  - less driven by outliers
- R function: `cor()`





Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

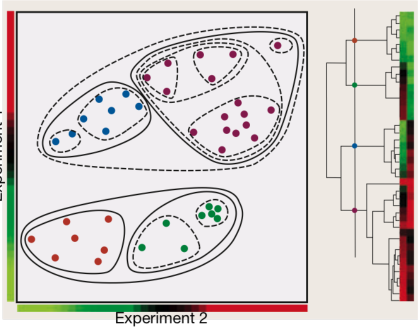


single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples

- Result: **dendrogram**
  - clustering is obtained by **cutting the dendrogram** at the desired level
- Similarity measure
  - Euclidean
  - Pearson
- Distance measure
  - Complete: largest distance
  - Average: average distance

Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

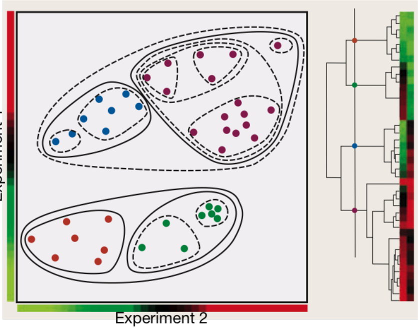


single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples

- Result: **dendrogram**
  - clustering is obtained by **cutting the dendrogram** at the desired level
- Similarity measure
  - Euclidean
  - Pearson
- Distance measure
  - Complete: largest distance
  - Average: average distance

Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

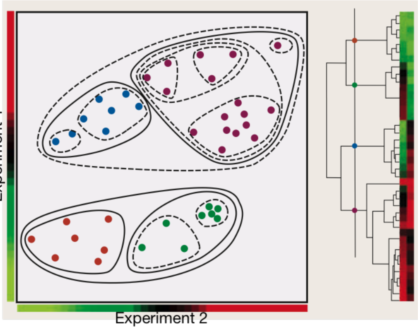


single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples

- Result: **dendrogram**
  - clustering is obtained by **cutting the dendrogram** at the desired level
- Similarity measure
  - Euclidean
  - Pearson
- Distance measure
  - Complete: largest distance
  - Average: average distance

Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

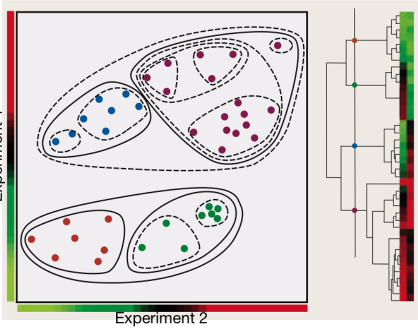


single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples

- Result: **dendrogram**
  - clustering is obtained by **cutting the dendrogram** at the desired level
- Similarity measure
  - Euclidean
  - Pearson
- Distance measure
  - Complete: largest distance
  - Average: average distance

Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

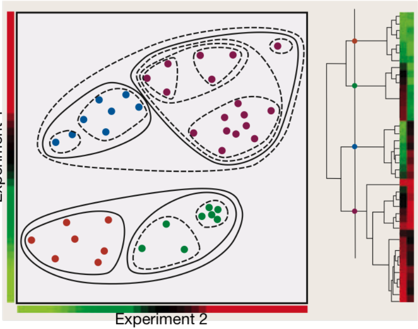


single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples

- Result: **dendrogram**
  - clustering is obtained by **cutting the dendrogram** at the desired level
- Similarity measure
  - Euclidean
  - Pearson
- Distance measure
  - Complete: largest distance
  - Average: average distance

Hierarchical clustering – grouping similar samples

Goal: partition the objects into homogeneous groups, such that the within-group similarities are large.

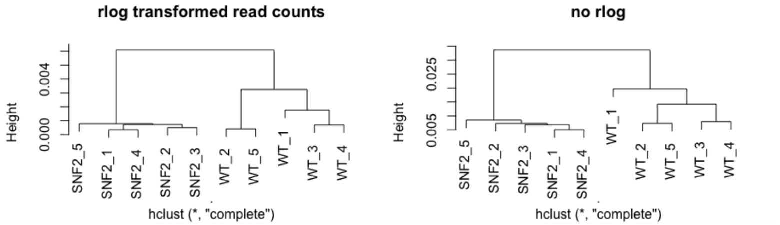


single-sample (or single-gene) clusters are successively joined, starting with the least dissimilar two samples

- Result: **dendrogram**
  - clustering is obtained by **cutting the dendrogram** at the desired level
- Similarity measure
  - Euclidean
  - Pearson
- Distance measure
  - Complete: largest distance
  - Average: average distance

Hiearchical clustering - R code

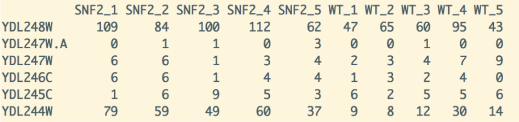
```
## calculate the correlation between columns of a matrix
pw_cor <- cor(rlog.norm.counts, method = "pearson" )

## use the correlation as a distance measure
distance.m_rlog <- as.dist(1 - pw_cor)

##  plot() can directly interpret the output of hclust() to generate
## a dendrogram
plot( hclust(distance.m_rlog),
      labels = colnames(rlog.norm.counts),
      main = "rlog transformed read counts")
```



Principal component analysis – capturing variability

Goal: reduce the dataset to have fewer dimensions, yet approx. preserve the distance between samples

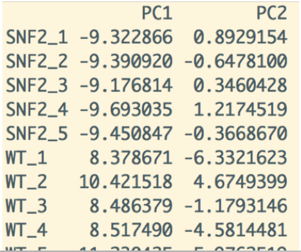
starting point: matrix with expression values per gene and sample, e.g. 6,600 genes x 10 samples



	SNF2_1	SNF2_2	SNF2_3	SNF2_4	SNF2_5	WT_1	WT_2	WT_3	WT_4	WT_5
YDL248W	109	84	100	112	62	47	65	60	95	43
YDL247W.A	0	1	1	0	3	0	0	1	0	0
YDL247W	6	6	1	3	4	2	3	4	7	9
YDL246C	6	6	1	4	4	1	3	2	4	0
YDL245C	1	6	9	5	3	6	2	5	5	6
YDL244W	79	59	49	60	37	9	8	12	30	14

```
assay(DESeq.rlog)[topVarGenes,])
%>% t %>% prcomp
```

transformed into 6,600 **principal components** x 10 samples

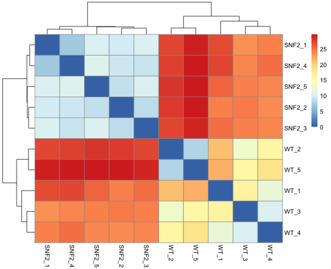


	PC1	PC2
SNF2_1	-9.322866	0.8929154
SNF2_2	-9.390920	-0.6478100
SNF2_3	-9.176814	0.3460428
SNF2_4	-9.693035	1.2174519
SNF2_5	-9.450847	-0.3668670
WT_1	8.378671	-6.3321623
WT_2	10.421518	4.6749399
WT_3	8.486379	-1.1793146
WT_4	8.517490	-4.5814481
WT_5	11.238425	5.0763510

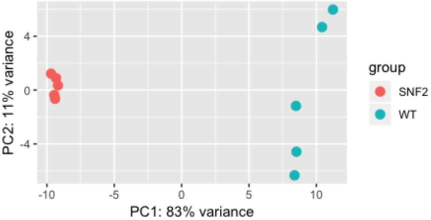
- linear combi of optimally weighted observed variables
- the vectors along which the variation between samples is maximal
- PC1-3 are usually sufficient to capture the major trends!

PCA vs. hierarchical clustering

- often similar results because both techniques should capture the most dominant patterns
- PCA will always be run on just a subset of the data!
- clustering will ALWAYS return clusters, PCA may not if the patterns of variation are too random



See `practical_exploratory.Rmd` R code to generate exploratory plots.
Use the `pcaExplorer` package!



See the chapter "Distance and Dimension Reduction" in Irizarry and Love [2015] for more details and the StatQuest video(s) on youtube.

Differential gene expression

Understand your null hypothesis!

- **DGE**: Differential **Gene** Expression
  - Has the total ouput of a gene changed?
  - input for the statistical testing: (estimated) **counts per gene** used by `DESeq2/edgeR/limma`
  - see Soneson et al. [2015] and bioconductor's `tximport` package vignette for details

- **DTU**: Differential **Transcript Usage**
  - Has the **isoform composition** for a given gene changed? I.e. are there different *dominant* isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by `DEXSeq` (!)
  - see Love et al. [2018] for details

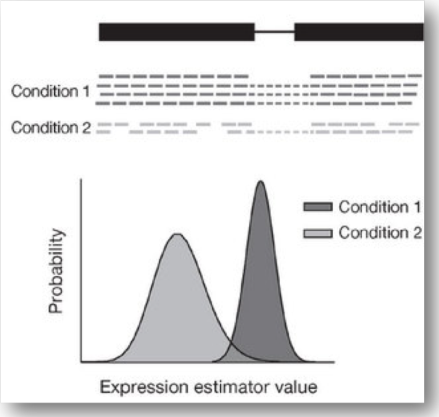
Understand your null hypothesis!

- **DGE**: Differential **Gene** Expression
  - Has the total ouput of a gene changed?
  - input for the statistical testing: (estimated) **counts per gene** used by `DESeq2/edgeR/limma`
  - see Soneson et al. [2015] and bioconductor's `tximport` package vignette for details

- **DTU**: Differential **Transcript Usage**
  - Has the **isoform composition** for a given gene changed? I.e. are there different *dominant* isoforms depending on the condition?
  - common when comparing different cell types (incl. healthy vs. cancer)
  - input for the statistical testing: (estimated) counts per transcript used by `DEXSeq` (!)
  - see Love et al. [2018] for details

DGE basics

H_0: There is no difference in the read distributions of the 2 conditions.



1 test per gene!

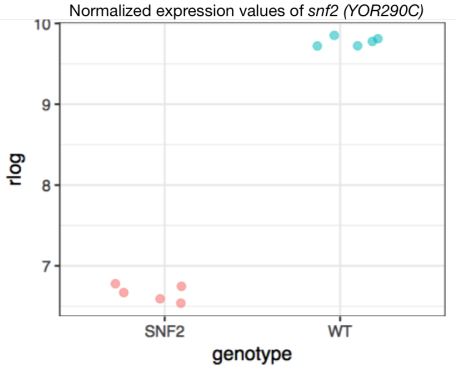
1. Estimate **magnitude** of DE taking into account differences in sequencing depth, technical, and biological read count variability.

logFC

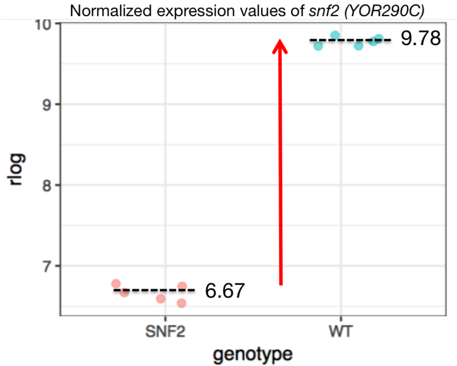
2. Estimate the **significance** of the difference accounting for performing thousands of tests.

(adjusted) p-value

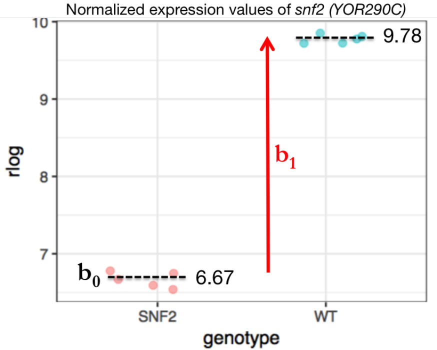
Applying linear models for read count modeling



Applying linear models for read count modeling



Applying linear models for read count modeling

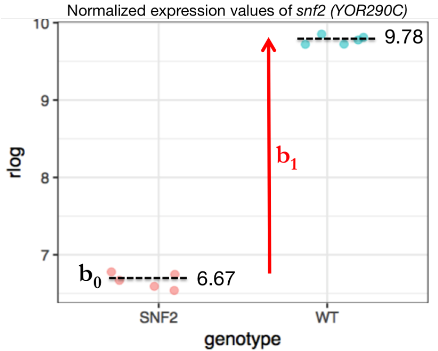


To describe all expression values of one (!) example gene (*snf2*), we can use a linear model like this:

$$\underbrace{Y}_{\text{expression values}} = \underbrace{b_0}_{\text{intercept}} + b_1 * \underbrace{x}_{\text{genotype (discrete factor here!)}} + e$$

Linear models model a response variable as a linear combination of predictors (betas), plus randomly distributed noise (e).

Applying linear models for read count modeling



To describe all expression values of one (!) example gene (*snf2*), we can use a linear model like this:

$$\underset{\text{expression values}}{Y} = \underset{\text{intercept}}{b_0} + b_1 * \underset{\text{genotype (discrete factor here!)}}{x} + e$$

Linear models model a response variable as a linear combination of predictors (betas), plus randomly distributed noise (e).

- b_0: **intercept**, i.e. average value of the baseline group
- b_1: **difference** between baseline and non-reference group
- x: 0 if genotype == "SNF2", 1 if genotype == "WT"

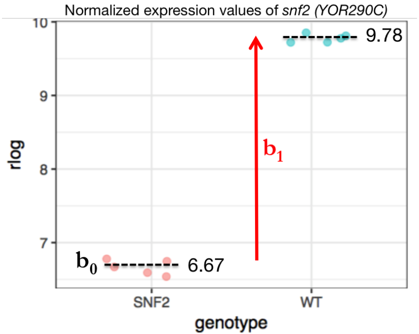
Model formulae syntax in R

- regression functions in R (e.g., `lm()`, `glm()`) use a “model formula” interface
- the basic format is:
  `response variable ~ explanatory variables`
  where tilde means “is modeled by” or “is modeled as a function of”.[4]
  e.g.: `lm( y ~ x )`

If you find yourself using linear models and somewhat complicated experimental designs more often than not, we strongly recommend to work through **chapters 4 and 5** of the PH525x series **Biomedical Data Science** [Irizarry and Love, 2016]

[4] See King [2016] for more details on the special meaning of mathematical operators within R formula contexts.

Applying linear models for read count modeling



- b_0: **intercept**, i.e. average value of the baseline group
- b_1: **difference** between baseline and non-reference group
- x: 0 if genotype == "SNF2", 1 if genotype == "WT"

Describe expression values *snf2* using a linear model:

$$\underset{\text{expression values}}{Y} = \underset{\text{intercept}}{b_0} + b_1 * \underset{\text{genotype (discrete factor here!)}}{x} + e$$

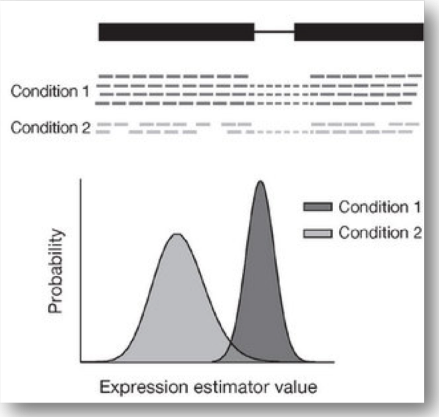
Factor of interest (b_1) can be estimated as follows:

```
# 1. FIT the model
> lmfit <- lm(rlog.norm ~ genotype)
# 2. ESTIMATE the coefficients
> coef(lmfit)
(Intercept)    genotypeWT
      6.666         3.111
```

Both values (b0, b1) are **estimates**! (They're spot-on because the values are so clear and the model is so simple!)

DGE basics

H_0: There is no difference in the read distributions of the 2 conditions.



1 test per gene!

1. Estimate **magnitude** of DE taking into account differences in sequencing depth, technical, and biological read count variability.

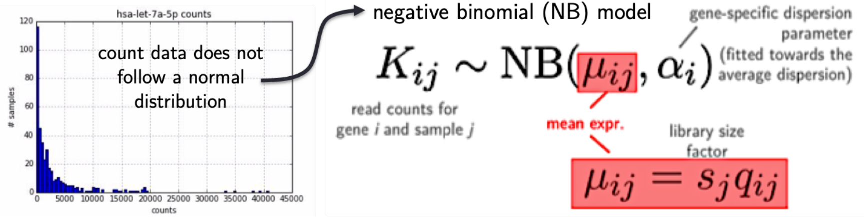
logFC

2. Estimate the **significance** of the difference accounting for performing thousands of tests.

(adjusted) p-value

DGE steps (à la DESeq2)

1. **Fitting** a sophisticated regression model to the read counts (per gene!)
   - library size factor
   - dispersion estimate using information across multiple genes
   - assuming neg. binomial distribution to describe read count distribution



count data does not follow a normal distribution

negative binomial (NB) model

$$K_{ij} \sim \mathrm{NB}(\boxed{\mu_{ij}}, \alpha_i)$$

gene-specific dispersion parameter (fitted towards the average dispersion)

read counts for gene *i* and sample *j*

mean expr.

library size factor

$$\boxed{\mu_{ij} = s_j q_{ij}}$$

DGE steps (à la DESeq2)

1. **Fitting** a sophisticated regression model to the read counts (done per gene; includes normalization)

gene-specific dispersion parameter (fitted towards the average dispersion)

$$K_{ij} \sim \mathrm{NB}(\mu_{ij}, \alpha_i)$$

read counts for gene *i* and sample *j*

2. Estimating **coefficients** to obtain the difference between the estimated mean expression of the different groups (⇒ log2FC)
   - define the **contrast of interest**, e.g. `Y ~ batchEffect + conditon`
   - always put the **factor of interest last**
   - order of the factor levels determines the direction of the log2FC values

DGE steps (à la DESeq2)

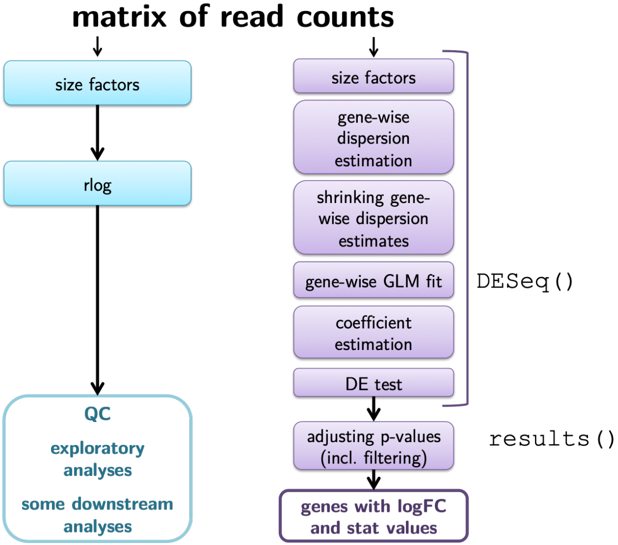
1. **Fitting** a sophisticated regression model to the read counts (done per gene; includes normalization)

$$\underbrace{K_{ij}}_{\text{read counts for gene } i \text{ and sample } j} \sim \mathrm{NB}(\mu_{ij}, \overset{\text{gene-specific dispersion parameter}}{\alpha_i})$$

(fitted towards the average dispersion)

2. Estimating **coefficients** to obtain the difference between the estimated mean expression of the different groups (⇒ log2FC)
3. **Test** whether the log2FC is "far away" from zero (remember H0!)
    - log-likelihood test or Wald test are offered by DESeq2
    - multiple hypothesis correction!

Summary: from read counts to DGE et al.



Comparison of additional tools for DGE analysis

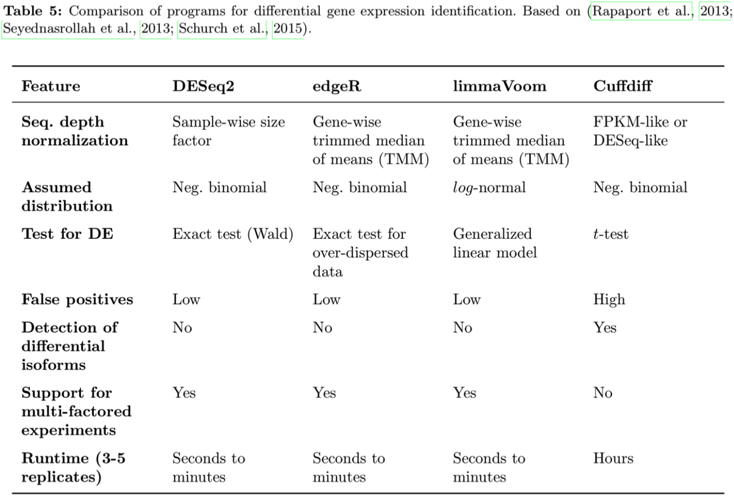


Table 5: Comparison of programs for differential gene expression identification. Based on (Rapaport et al., 2013; Seyednasrollah et al., 2013; Schurch et al., 2015).

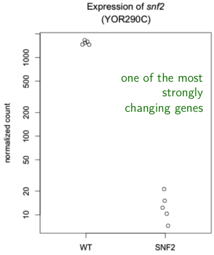
Feature	**DESeq2**	**edgeR**	**limmaVoom**	**Cuffdiff**
Seq. depth normalization	Sample-wise size factor	Gene-wise trimmed median of means (TMM)	Gene-wise trimmed median of means (TMM)	FPKM-like or DESeq-like
Assumed distribution	Neg. binomial	Neg. binomial	*log*-normal	Neg. binomial
Test for DE	Exact test (Wald)	Exact test for over-dispersed data	Generalized linear model	*t*-test
False positives	Low	Low	Low	High
Detection of differential isoforms	No	No	No	Yes
Support for multi-factored experiments	Yes	Yes	Yes	No
Runtime (3-5 replicates)	Seconds to minutes	Seconds to minutes	Seconds to minutes	Hours

When in doubt, compare the results of `limma`, `edgeR`, and `DESeq2` to get a feeling for how robust your favorite DE genes are. All packages can be found at Bioconductor.

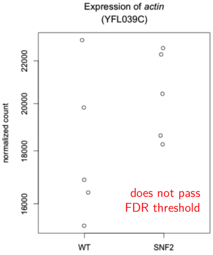
Downstream analyses

Understanding the RESULTS of the DGE analysis

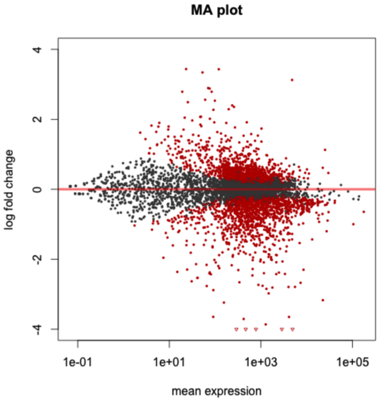
- Investigate the `results()` output:
  - How many DE genes? (FDR/q-value!)
  - How strongly do the DE genes change?
  - Directions of change?
  - Are your favorite genes among the DE genes?


Expression of *snf2* (YOR290C)

one of the most strongly changing genes


Expression of *actin* (YFL039C)

does not pass FDR threshold


MA plot

Understanding the FUNCTIONS of your DE genes

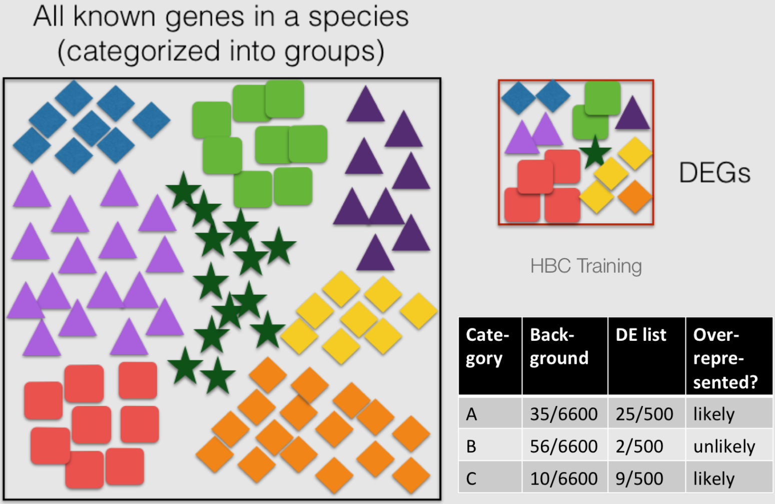
There are myriad tools for this – many are web-based, many are R packages, many will address very specific questions. Typical points of interest are:

- enriched gene ontology (GO) terms
  - ontology = standardized vocabulary
  - 3 classes of gene ontologies are maintained:
    - biological processes (BP), cell components (CC), and molecular functions (MF)
- enriched pathways
  - gene sets: e.g. from MSigDB [Liberzon et al., 2015]
  - physical interaction networks: e.g. from STRING [Szklarczyk et al., 2017]
  - metabolic (and other) pathways: e.g. from KEGG [Kanehisa et al., 2017]
- upstream regulators

> None (!) of these methods should lead you to make definitive claims about the role of certain pathways for your phenotype. These are **hypothesis-generating** tools! Also: make sure you use **shrunken logFC** values [Zhu et al., 2019].

Two typical approaches of enrichment analyses

1. Over-representation analysis (ORA)



Category	Background	DE list	Over-represented?
A	35/6600	25/500	likely
B	56/6600	2/500	unlikely
C	10/6600	9/500	likely

Two typical approaches of enrichment analyses

1. Over-representation analysis (ORA)

- “2x2 table method”
- assessing overlap of DE genes with genes of a given pathway
- statistical test: e.g. hypergeometric test
- limitations:
  - direction of change is ignored
  - magnitude of change is ignored
  - interprets genes as well as pathways as independent entities

See Khatri et al. [2012] for details!

Two typical approaches of enrichment analyses

1. Over-representation analysis (ORA)

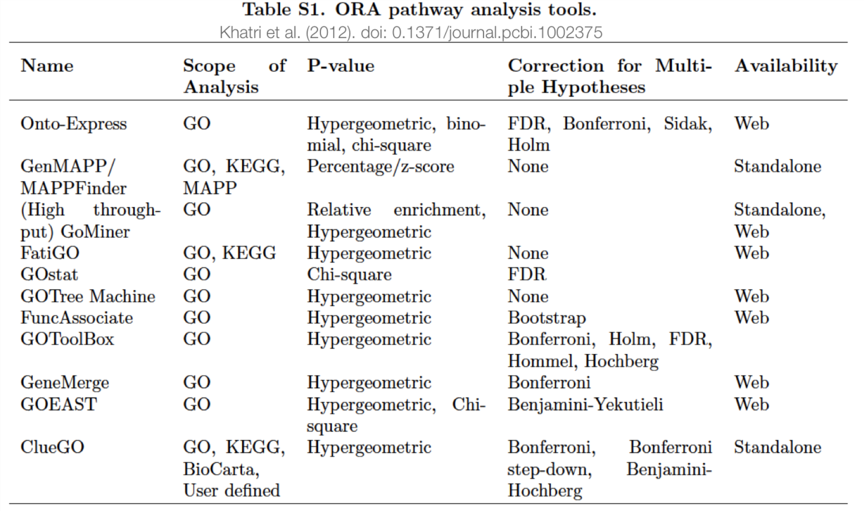


Table S1. ORA pathway analysis tools.

Khatri et al. (2012). doi: 0.1371/journal.pcbi.1002375

Name	Scope of Analysis	P-value	Correction for Multiple Hypotheses	Availability
Onto-Express	GO	Hypergeometric, binomial, chi-square	FDR, Bonferroni, Sidak, Holm	Web
GenMAPP/ MAPPFinder	GO, KEGG, MAPP	Percentage/z-score	None	Standalone
(High throughput) GoMiner	GO	Relative enrichment, Hypergeometric	None	Standalone, Web
FatiGO	GO, KEGG	Hypergeometric	None	Web
GOstat	GO	Chi-square	FDR	
GOTree Machine	GO	Hypergeometric	None	Web
FuncAssociate	GO	Hypergeometric	Bootstrap	Web
GOToolBox	GO	Hypergeometric	Bonferroni, Holm, FDR, Hommel, Hochberg	
GeneMerge	GO	Hypergeometric	Bonferroni	Web
GOEAST	GO	Hypergeometric, Chi-square	Benjamini-Yekutieli	Web
ClueGO	GO, KEGG, BioCarta, User defined	Hypergeometric	Bonferroni, Bonferroni step-down, Benjamini-Hochberg	Standalone

Two typical approaches of enrichment analyses

2. Functional Class Scoring ("Gene set enrichment")

- gene-level statistics for all genes in a pathway are aggregated into a single pathway-level statistic
- score will depend on size of the pathway, and the amount of correlation between genes in the pathway
- all genes are used
- direction and magnitude of change matter
- coordinated changes of genes within the same pathway matter, too

Two typical approaches of enrichment analyses

2. Functional Class Scoring ("Gene set enrichment")

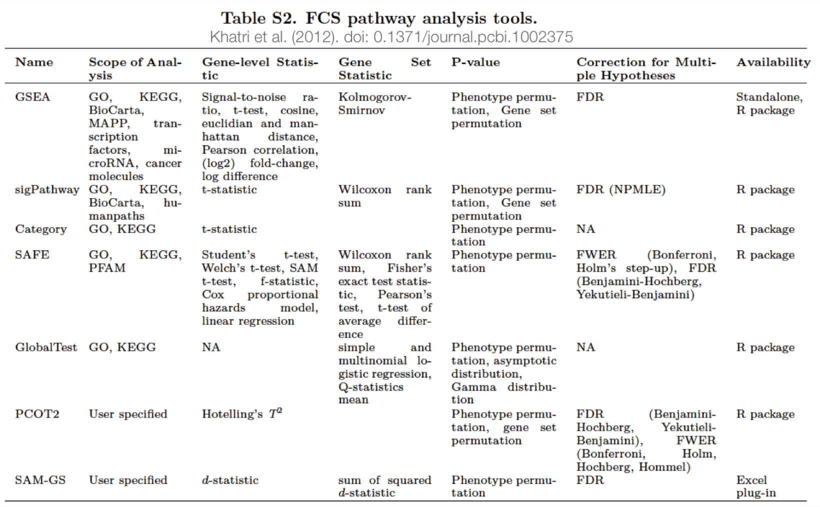


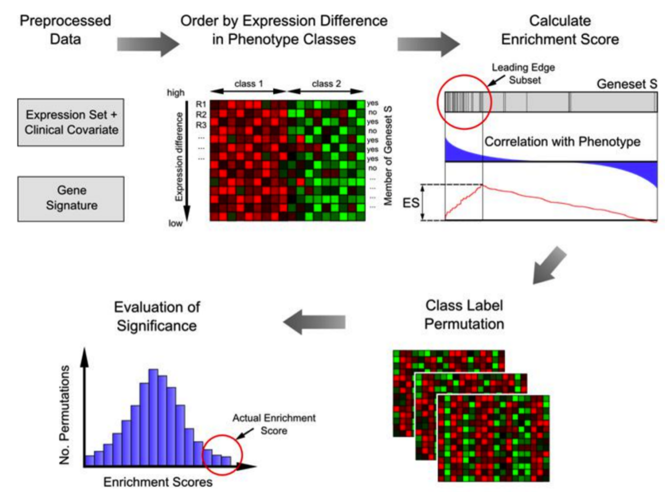
Table S2. FCS pathway analysis tools.

Khatri et al. (2012). doi: 0.1371/journal.pcbi.1002375

Name	Scope of Analysis	Gene-level Statistic	Gene Set Statistic	P-value	Correction for Multiple Hypotheses	Availability
GSEA	GO, KEGG, BioCarta, MAPP, transcription factors, microRNA, cancer molecules	Signal-to-noise ratio, t-test, cosine, euclidian and manhattan distance, Pearson correlation, (log2) fold-change, log difference	Kolmogorov-Smirnov	Phenotype permutation, Gene set permutation	FDR	Standalone, R package
sigPathway	GO, KEGG, BioCarta, humanpaths	t-statistic	Wilcoxon rank sum	Phenotype permutation, Gene set permutation	FDR (NPMLE)	R package
Category	GO, KEGG	t-statistic		Phenotype permutation	NA	R package
SAFE	GO, KEGG, PFAM	Student's t-test, Welch's t-test, SAM t-test, f-statistic, Cox proportional hazards model, linear regression	Wilcoxon rank sum, Fisher's exact test statistic, Pearson's test, t-test of average difference	Phenotype permutation	FWER (Bonferroni, Holm's step-up), FDR (Benjamini-Hochberg, Yekutieli-Benjamini)	R package
GlobalTest	GO, KEGG	NA	simple and multinomial logistic regression, Q-statistics mean	Phenotype permutation, asymptotic distribution, Gamma distribution	NA	R package
PCOT2	User specified	Hotelling's T^2		Phenotype permutation, gene set permutation	FDR (Benjamini-Hochberg, Yekutieli-Benjamini), FWER (Bonferroni, Holm, Hochberg, Hommel)	R package
SAM-GS	User specified	*d*-statistic	sum of squared *d*-statistic	Phenotype permutation	FDR	Excel plug-in

Two typical approaches of enrichment analyses

2. Functional Class Scoring: Example GSEA

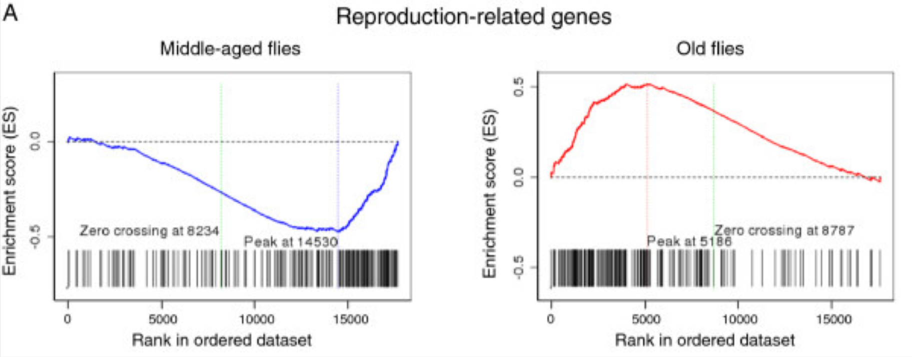


http://slideplayer.biz.tr/slide/2738467/10/images/20/Gene+Set+Enrichment+Analysis+(GSEA).jpg

Two typical approaches of enrichment analyses

2. Functional Class Scoring (“Gene set enrichment”)

Example GSEA results for positive and negative correlation



Doroszuk et al. (2012) doi: 10.1186/1471-2164-13-167

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use goseq to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietory software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use goseq to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietary software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use goseq to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietary software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use `goseq` to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietary software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use goseq to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as GOrilla, REVIGO [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietory software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use goseq to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietory software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use `goseq` to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietory software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use `goseq` to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietory software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use `goseq` to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietory software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use `goseq` to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as `GOrilla`, `REVIGO` [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietory software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

Summary – downstream analyses

Know your biological question(s) of interest!

- all enrichment methods potentially suffer from **gene length bias**
  - long genes will get more reads
- for **GO terms**:
  - use goseq to identify enriched GO terms [Young et al., 2010]
  - use additional tools, such as GOrilla, REVIGO [Eden et al., 2009, Supek et al., 2011] to summarize the often redundant GO term lists
- for **KEGG pathways**:
  - e.g. GAGE and PATHVIEW [Luo and Brouwer, 2013, Luo et al., 2017] [5]
- miscellaneous including attempts to predict upstream regulators
  - Enrichr [Chen et al., 2013]
  - RegulatorTrail [Kehl et al., 2017]
  - Ingenuity Pathway Analysis Studio (proprietory software!)

See the additional links and material on our course website!

[5] https://www.r-bloggers.com/tutorial-rna-seq-differential-expression-pathway-analysis-with-sailfish-deseq2-gage-and-pathview/

References

Edward Y. Chen, Christopher M. Tan, Yan Kou, Qiaonan Duan, Zichen Wang, Gabriela V. Meirelles, Neil R. Clark, and Avi Ma'ayan. Enrichr: Interactive and collaborative HTML5 gene list enrichment analysis tool. *BMC Bioinformatics*, 2013. doi: 10.1186/1471-2105-14-128. URL http://amp.pharm.mssm.edu/Enrichr.

Eran Eden, Roy Navon, Israel Steinfeld, Doron Lipson, and Zohar Yakhini. GOrilla: a tool for discovery and visualization of enriched GO terms in ranked gene lists. *BMC bioinformatics*, 10(1):48, jan 2009. doi: 10.1186/1471-2105-10-48. URL http://cbl-gorilla.cs.technion.ac.il.

R. Irizarry and M. Love. Leanpub, 2015. URL https://leanpub.com/dataanalysisforthelifesciences.

R. Irizarry and M. Love. Biomedical Data Science, 2016.

Minoru Kanehisa, Miho Furumichi, Mao Tanabe, Yoko Sato, and Kanae Morishima. KEGG: New perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Research*, 2017. doi: 10.1093/nar/gkw1092.

Tim Kehl, Lara Schneider, Florian Schmidt, Daniel Stöckel, Nico Gerstner, Christina Backes, Eckart Meese, Andreas Keller, Marcel H. Schulz, and Hans Peter Lenhof. RegulatorTrail: A web service for the identification of key transcriptional regulators. *Nucleic Acids Research*, 2017. doi: 10.1093/nar/gkx350. URL https://regulatortrail.bioinf.uni-sb.de/.

Purvesh Khatri, Marina Sirota, and Atul J. Butte. Ten years of pathway analysis: Current approaches and outstanding challenges. *PLoS Computational Biology*, 2012. doi: 10.1371/journal.pcbi.1002375.

William B. King. Model Formulae Tutorial, 2016. URL http://ww2.coastal.edu/kingw/statistics/R-tutorials/formulae.html.

Arthur Liberzon, Chet Birger, Helga Thorvaldsdóttir, Mahmoud Ghandi, Jill P. Mesirov, and Pablo Tamayo. The Molecular Signatures Database Hallmark Gene Set Collection. *Cell Systems*, 2015. doi: 10.1016/j.cels.2015.12.004.

Michael I Love, Charlotte Soneson, and Rob Patro. Swimming downstream: statistical analysis of differential transcript usage following Salmon quantification. *F1000Research*, 7(952), 2018. doi: 10.12688/f1000research.15398.1.

Weijun Luo and Cory Brouwer. Pathview: An R/Bioconductor package for pathway-based data integration and visualization. *Bioinformatics*, 2013. doi: 10.1093/bioinformatics/btt285.

Weijun Luo, Gaurav Pant, Yeshvant K. Bhavnasi, Steven G. Blanchard, and Cory Brouwer. Pathview Web: User friendly pathway visualization and data integration. *Nucleic Acids Research*, 2017. doi: 10.1093/nar/gkx372. URL https://pathview.uncc.edu/.

Charlotte Soneson, Michael I. Love, and Mark D. Robinson. Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences. *F1000Research*, 4(0):1521, 2015. doi: 10.12688/f1000research.7563.2.

Fran Supek, Matko Bošnjak, Nives Škunca, and Tomislav Šmuc. REVIGO summarizes and visualizes long lists of gene ontology terms. *PloS one*, 6 (7):e21800, jan 2011. doi: 10.1371/journal.pone.0021800. URL http://revigo.irb.hr/.

Damian Szklarczyk, John H. Morris, Helen Cook, Michael Kuhn, Stefan Wyder, Milan Simonovic, Alberto Santos, Nadezhda T. Doncheva, Alexander Roth, Peer Bork, Lars J. Jensen, and Christian Von Mering. The STRING database in 2017: Quality-controlled protein-protein association networks, made broadly accessible. *Nucleic Acids Research*, 2017. doi: 10.1093/nar/gkw937.

Cole Trapnell, Adam Roberts, Loyal Goff, Geo Pertea, Daehwan Kim, David R Kelley, Harold Pimentel, Steven L Salzberg, John L Rinn, and Lior Pachter. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nature Protocols*, 7(3): 562–78, March 2012. doi: 10.1038/nprot.2012.016.

Matthew D. Young, Matthew J. Wakefield, Gordon K. Smyth, and Alicia Oshlack. Gene ontology analysis for RNA-seq: accounting for selection bias. *Genome Biology*, 2010. doi: 10.1186/gb-2010-11-2-r14.

Anqi Zhu, Joseph G. Ibrahim, and Michael I. Love. Heavy-Tailed prior distributions for sequence count data: Removing the noise and preserving large differences. *Bioinformatics*, 35(12):2084–2092, 2019. doi: 10.1093/bioinformatics/bty895.

Andrzej Zielezinski, Susana Vinga, Jonas Almeida, and Wojciech M. Karlowski. Alignment-free sequence comparison: Benefits, applications, and tools. *Genome Biology*, 2017. doi: 10.1186/s13059-017-1319-7.