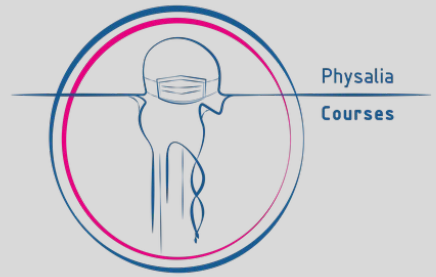
RNA-seq analysis

Epigenomics Data Analysis

Jacques Serizay

Physalia 2023

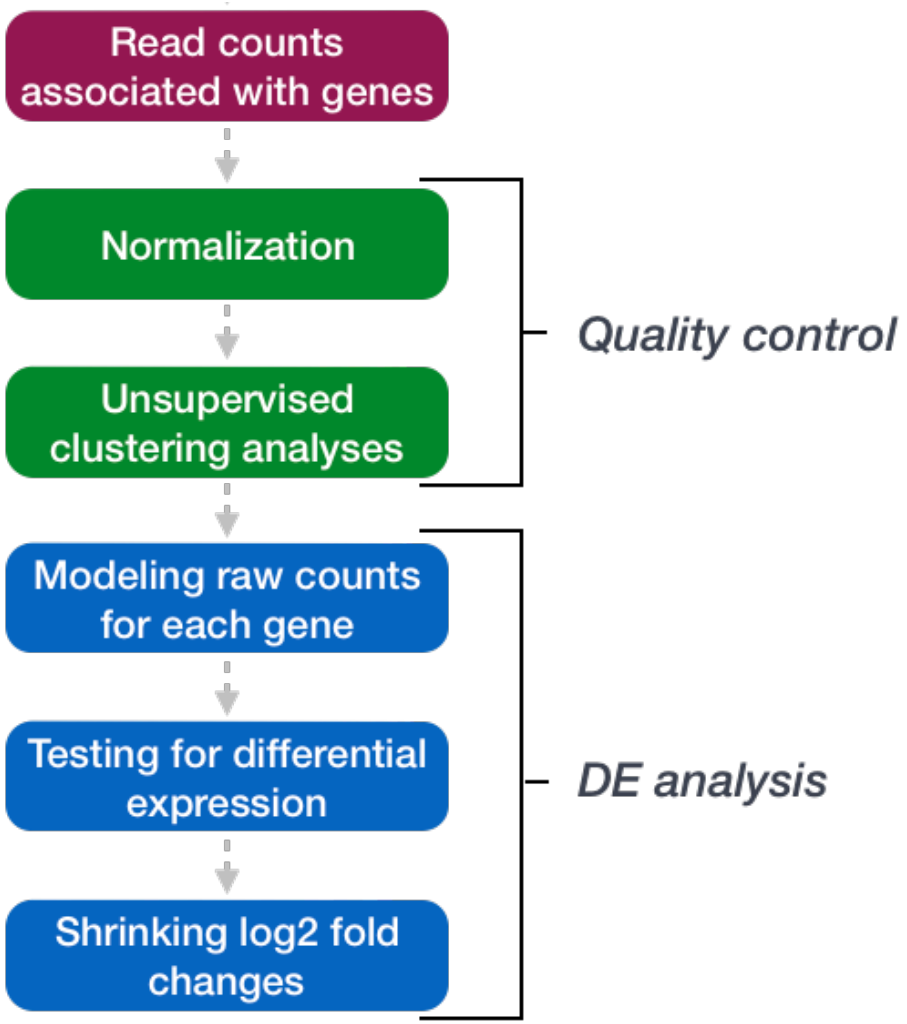


RNA-seq downstream analysis

- Get .bcl files 
- Create fastq files  Or **bcl2fastq**
- QC: remove/trim low quality reads — E.g. **cutadapt**
- Align fastq to BAM — E.g. **bowtie2, STAR**
- Filter duplicates, artifacts, … — E.g. **samtools**
- Generate tracks — E.g. **deepTools**
- Assay-specific downstream analysis

Transcript abundance quantification
Differential gene expression analysis
Gene ontology over-representation analysis

DESeq2 analysis workflow

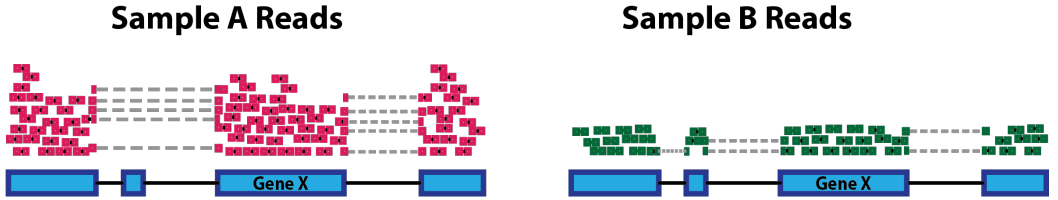


https://hbctraining.github.io/DGE_workshop/

Estimating gene/transcript abundance in RNA-seq

Important points when comparing gene abundance for different datasets/genes are:

- Sequencing depth

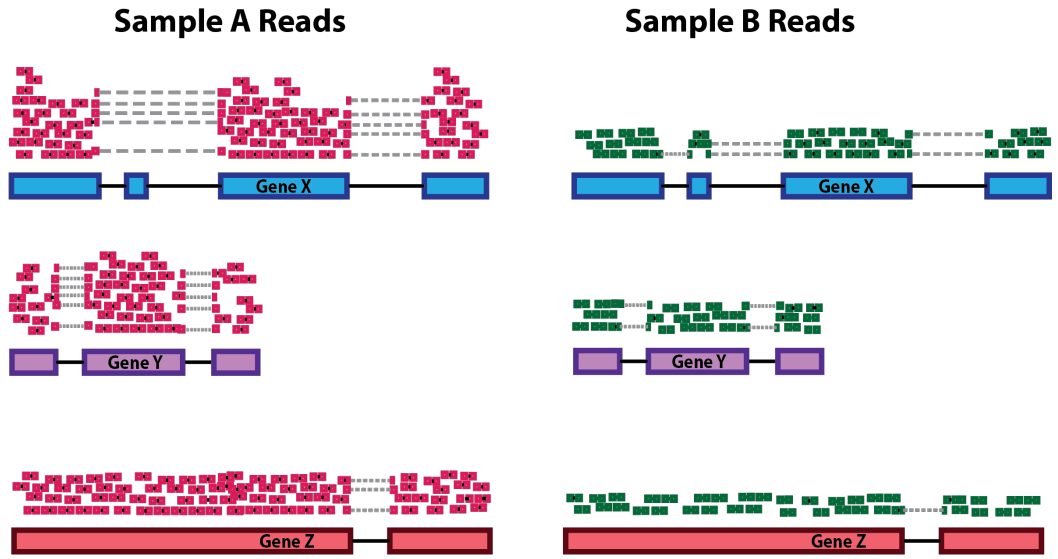


https://hbctraining.github.io/DGE_workshop/

Estimating gene/transcript abundance in RNA-seq

Important points when comparing gene abundance for different datasets/genes are:

- Sequencing depth
- Gene length

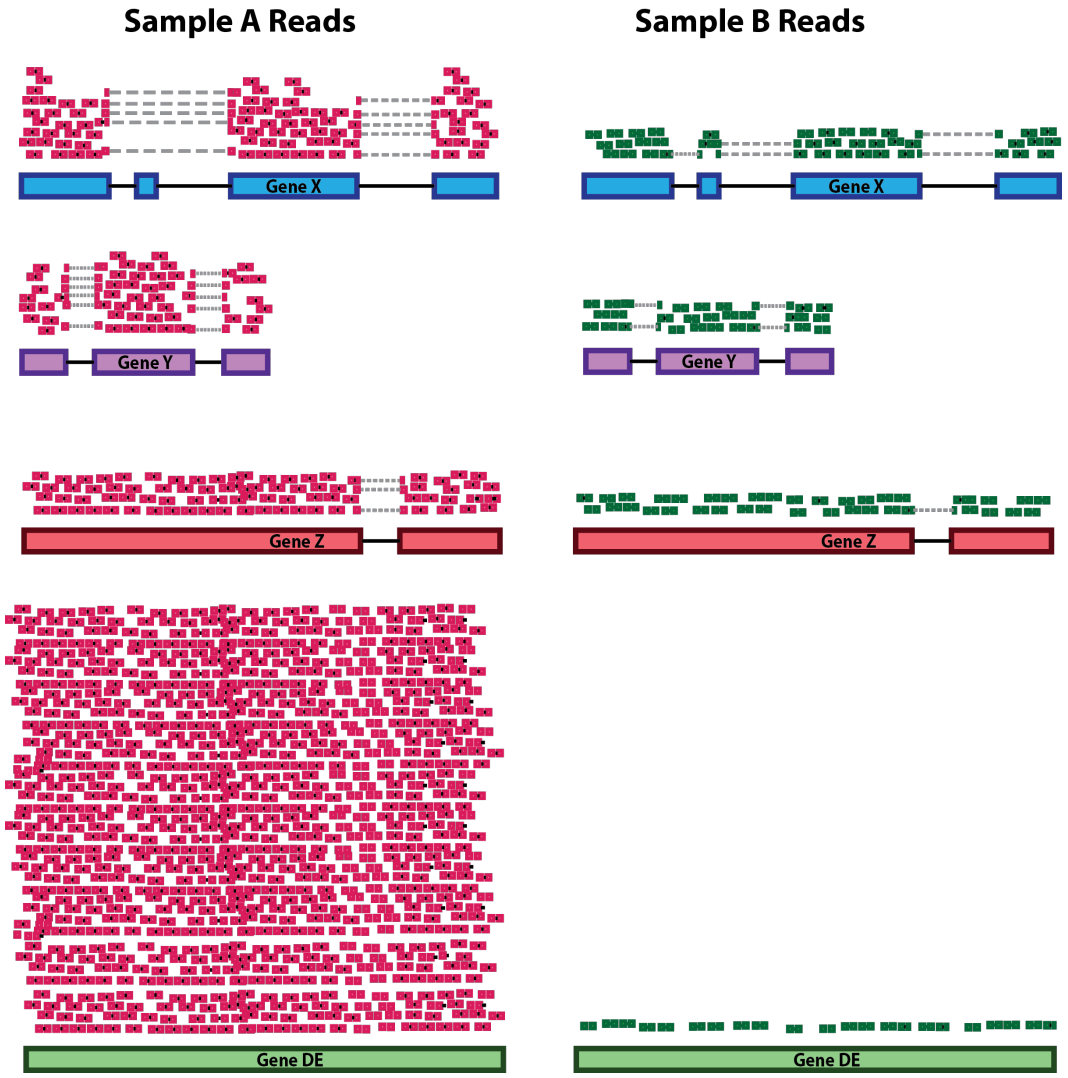


https://hbctraining.github.io/DGE_workshop/

Estimating gene/transcript abundance in RNA-seq

Important points when comparing gene abundance for different datasets/genes are:

- Sequencing depth
- Gene length
- RNA composition



https://hbctraining.github.io/DGE_workshop/

Estimating gene/transcript abundance in RNA-seq

- `featureCounts` can be used to count reads overlapping a set of gene annotations.

```
# in bash

featureCounts \
      -g gene_name \
      -s 2 \
      -p -B \
      -T 16 \
      -o data/counts/RNAseq_counts.tsv \
      -a data/counts/hg38_Gencodev41.gtf \
      data/mapping/RNA_ctl_1^hg38^filtered.bam \
      data/mapping/RNA_ctl_2^hg38^filtered.bam \
      data/mapping/RNA_ctl_3^hg38^filtered.bam \
      data/mapping/RNA_foxj1_1^hg38^filtered.bam \
      data/mapping/RNA_foxj1_2^hg38^filtered.bam
```

Estimating gene/transcript abundance in RNA-seq

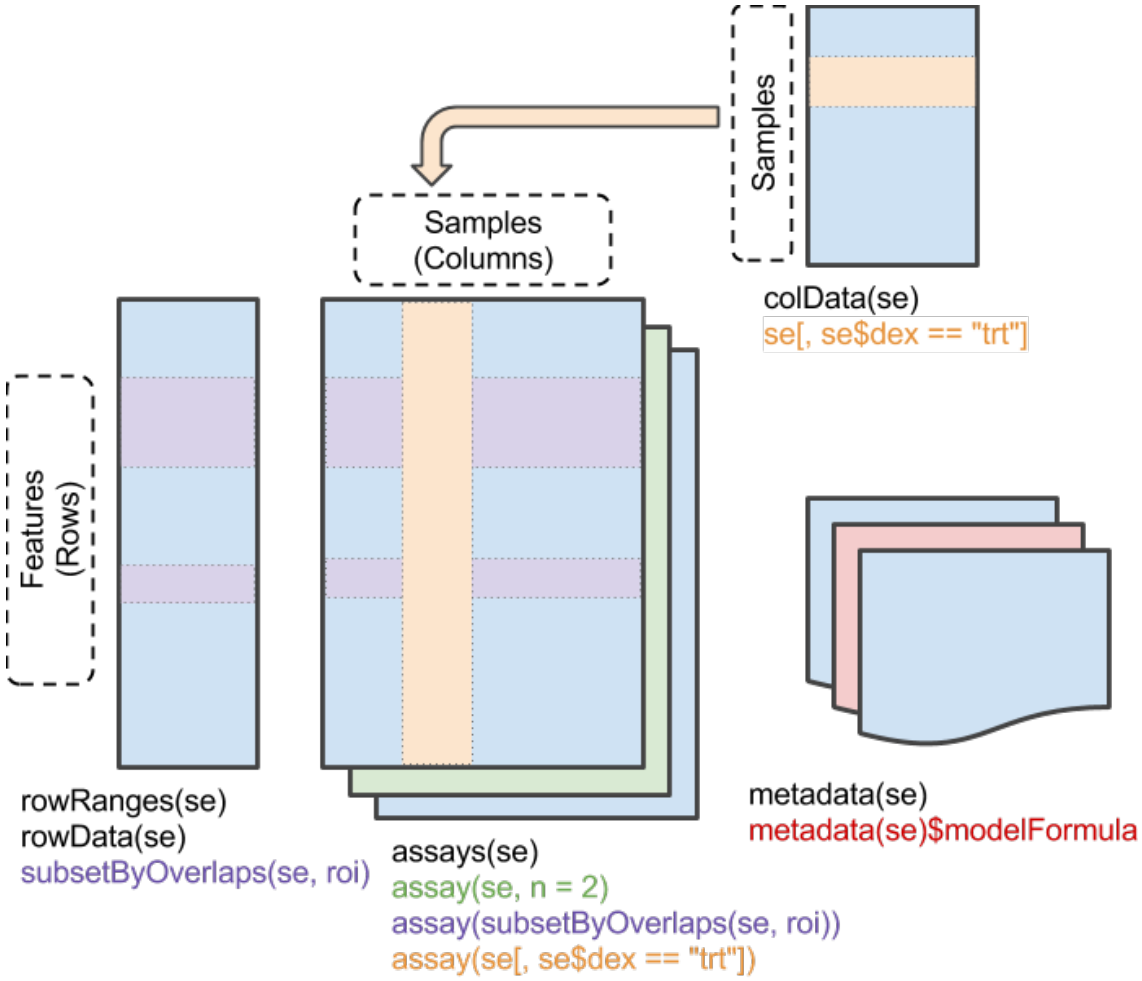
- `featureCounts` can be used to count reads overlapping a set of gene annotations.
- `Rsubread` is an R package which wraps `featureCounts` function in R.

```
# in bash

featureCounts \
      -g gene_name \
      -s 2 \
      -p -B \
      -T 16 \
      -o data/counts/RNAseq_counts.tsv \
      -a data/counts/hg38_Gencodev41.gtf \
      data/mapping/RNA_ctl_1^hg38^filtered.bam \
      data/mapping/RNA_ctl_2^hg38^filtered.bam \
      data/mapping/RNA_ctl_3^hg38^filtered.bam \
      data/mapping/RNA_foxj1_1^hg38^filtered.bam \
      data/mapping/RNA_foxj1_2^hg38^filtered.bam
```

```
# in R

cnts <- Rsubread::featureCounts(
    files = c('….bam', '….bam'),
    annot.ext = 'annots.gtf',
    isGTFAnnotationFile = TRUE,
    GTF.featureType = 'sequence_feature',
    GTF.attrType = 'id',
    isPairedEnd = TRUE,
    nthreads = 16
)
```

DESeq2 analysis workflow

- DESeq2 is based on a SummarizedExperiment object.



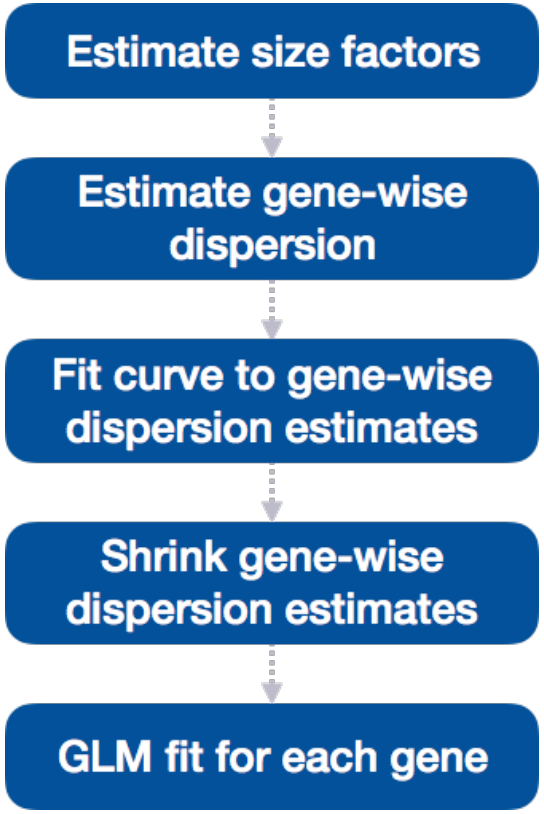
DESeq2 analysis workflow

- DESeq2 is based on a SummarizedExperiment object.
- A **formula** needs to be specified

```
> dds <- DESeqDataSet(counts, design = ~ timepoint)
```

DESeq2 analysis workflow

- Differential expression analysis is as simple as `DESeq()`



sample2	WT	2	No drug	
sample3	KO	1	Drug	
sample4	KO	2	drug	
sample5	WT	1	drug	
sample6	WT	2	drug	

- DE
- A f

- Differential expression analysis is as simple as DESeq( )

```
> dds <- DESeqDataSet(counts, design = ~ Condition + Treatment)
> dds <- DESeq(dds)
estimating size factors
estimating dispersions
gene-wise dispersion estimates
mean-dispersion relationship
final dispersion estimates
fitting model and testing
> dds
class: DESeqDataSet
dim: 6513 10
metadata(1): version
assays(4): counts mu H cooks
rownames: NULL
rowData names(37): summit peakID ... deviance maxCooks
colnames: NULL
colData names(5): sample timepoint replicate bam sizeFactor
```

DESeq2 analysis workflow

4) Extract results

- We can use `DESeq2::results()`
- DESeq2 extracts results for pairwise comparison between 2 conditions

```
> contrasts
     15_v_00    30_v_00    45_v_00    60_v_00    30_v_15    45_v_15    60_v_15    45_v_30    60_v_30    60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"
```

DESeq2 analysis workflow

```
> contrasts
     15_v_00     30_v_00     45_v_00     60_v_00     30_v_15     45_v_15     60_v_15     45_v_30     60_v_30     60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"

> results(dds, contrast = contrasts[, 2])
log2 fold change (MLE): timepoint 30 vs 00
Wald test p-value: timepoint 30 vs 00
DataFrame with 6513 rows and 6 columns
      baseMean log2FoldChange     lfcSE       stat      pvalue        padj
     <numeric>      <numeric> <numeric>  <numeric>   <numeric>   <numeric>
1    1643.7457      -0.900131  0.225114   -3.99856 6.37297e-05 0.002809493
2     106.1707      -0.876365  0.313066   -2.79930 5.12139e-03 0.066589377
3     134.4141      -0.753498  0.311437   -2.41942 1.55451e-02 0.133487340
4     203.8053      -1.094019  0.227780   -4.80296 1.56338e-06 0.000140911
5      67.2922      -1.001061  0.336647   -2.97362 2.94309e-03 0.046569804
...         ...            ...       ...         ...         ...         ...
6509    36.1437     -0.0313828  0.417826 -0.0751098   0.9401274    0.979001
6510   603.5468      -0.4020281  0.181778 -2.2116386   0.0269916    0.185475
6511    42.8792     -0.3776449  0.378523 -0.9976807   0.3184342    0.633629
6512    15.2541     -0.5818438  0.642764 -0.9052221   0.3653477          NA
6513    37.8941     -0.8396196  0.427293 -1.9649731   0.0494173    0.258224
```

DESeq2 analysis workflow

```
> contrasts
     15_v_00    30_v_00    45_v_00    60_v_00    30_v_15    45_v_15    60_v_15    45_v_30    60_v_30    60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"

> results(dds, contrast = contrasts[, 2])
log2 fold change (MLE): timepoint 30 vs 00
Wald test p-value: timepoint 30 vs 00
DataFrame with 6513 rows and 6 columns
      baseMean log2FoldChange     lfcSE      stat      pvalue       padj
     <numeric>      <numeric> <numeric> <numeric>   <numeric>  <numeric>
1    1643.7457      -0.900131  0.225114  -3.99856 6.37297e-05 0.002809493
2     106.1707      -0.876365  0.313066  -2.79930 5.12139e-03 0.066589377
3     134.4141      -0.753498  0.311437  -2.41942 1.55451e-02 0.133487340
4     203.8053      -1.094019  0.227780  -4.80296 1.56338e-06 0.000140911
5      67.2922      -1.001061  0.336647  -2.97362 2.94309e-03 0.046569804
...         ...            ...       ...       ...         ...        ...
6509   36.1437    -0.0313828  0.417826 -0.0751098   0.9401274   0.979001
6510  603.5468     -0.4020281  0.181778 -2.2116386   0.0269916   0.185475
6511   42.8792     -0.3776449  0.378523 -0.9976807   0.3184342   0.633629
6512   15.2541     -0.5818438  0.642764 -0.9052221   0.3653477         NA
6513   37.8941     -0.8396196  0.427293 -1.9649731   0.0494173   0.258224
```

DESeq2 analysis workflow

```
> contrasts
     15_v_00     30_v_00     45_v_00     60_v_00     30_v_15     45_v_15     60_v_15     45_v_30     60_v_30     60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"

> results(dds, contrast = contrasts[, 2])
log2 fold change (MLE): timepoint 30 vs 00
Wald test p-value: timepoint 30 vs 00
DataFrame with 6513 rows and 6 columns
      baseMean log2FoldChange     lfcSE       stat      pvalue       padj
     <numeric>      <numeric> <numeric>  <numeric>   <numeric>  <numeric>
1    1643.7457      -0.900131  0.225114   -3.99856 6.37297e-05 0.002809493
2     106.1707      -0.876365  0.313066   -2.79930 5.12139e-03 0.066589377
3     134.4141      -0.753498  0.311437   -2.41942 1.55451e-02 0.133487340
4     203.8053      -1.094019  0.227780   -4.80296 1.56338e-06 0.000140911
5      67.2922      -1.001061  0.336647   -2.97362 2.94309e-03 0.046569804
...         ...            ...       ...         ...         ...        ...
6509    36.1437    -0.0313828  0.417826 -0.0751098   0.9401274   0.979001
6510  603.5468     -0.4020281  0.181778 -2.2116386   0.0269916   0.185475
6511   42.8792     -0.3776449  0.378523 -0.9976807   0.3184342   0.633629
6512   15.2541     -0.5818438  0.642764 -0.9052221   0.3653477         NA
6513   37.8941     -0.8396196  0.427293 -1.9649731   0.0494173   0.258224
```

DESeq2 analysis workflow

```
> contrasts
     15_v_00     30_v_00     45_v_00     60_v_00     30_v_15     45_v_15     60_v_15     45_v_30     60_v_30     60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"

> results(dds, contrast = contrasts[, 2])
log2 fold change (MLE): timepoint 30 vs 00
Wald test p-value: timepoint 30 vs 00
DataFrame with 6513 rows and 6 columns
      baseMean log2FoldChange     lfcSE       stat      pvalue       padj
     <numeric>      <numeric> <numeric>  <numeric>   <numeric>  <numeric>
1    1643.7457      -0.900131  0.225114   -3.99856 6.37297e-05 0.002809493
2     106.1707      -0.876365  0.313066   -2.79930 5.12139e-03 0.066589377
3     134.4141      -0.753498  0.311437   -2.41942 1.55451e-02 0.133487340
4     203.8053      -1.094019  0.227780   -4.80296 1.56338e-06 0.000140911
5      67.2922     -1.001061  0.336647   -2.97362 2.94309e-03 0.046569804
...         ...           ...       ...        ...         ...        ...
6509   36.1437    -0.0313828  0.417826 -0.0751098   0.9401274   0.979001
6510  603.5468     -0.4020281  0.181778 -2.2116386   0.0269916   0.185475
6511   42.8792    -0.3776449  0.378523 -0.9976807   0.3184342   0.633629
6512   15.2541    -0.5818438  0.642764 -0.9052221   0.3653477         NA
6513   37.8941    -0.8396196  0.427293 -1.9649731   0.0494173   0.258224
```

Regularized log counts

- DESeq2 is primarily designed to **estimate fold-change** between conditions, **NOT** actual abundance!

Regularized log counts

- DESeq2 is primarily designed to **estimate fold-change** between conditions, **NOT** actual abundance!
- Rlog (regularized log) can be used to approximate gene expression levels.

Regularized log counts

- DESeq2 is primarily designed to **estimate fold-change** between conditions, **NOT** actual abundance!
- Rlog (regularized log) can be used to approximate gene expression levels.
- Rlog transforms the count data to the **log2 scale** in a way which

    1) minimizes differences between samples for rows with small counts

    2) normalizes with respect to library size

DESeq2::rlog()