

# Lesson 11

---

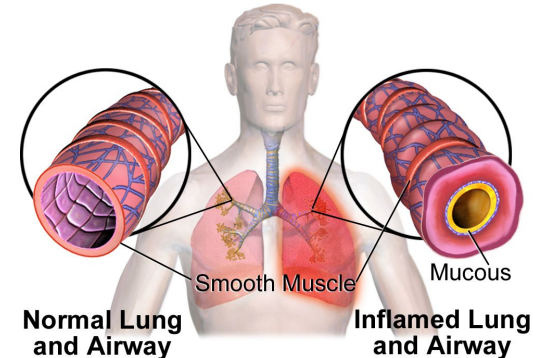
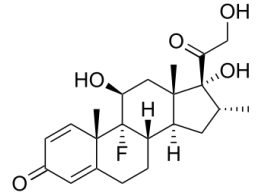
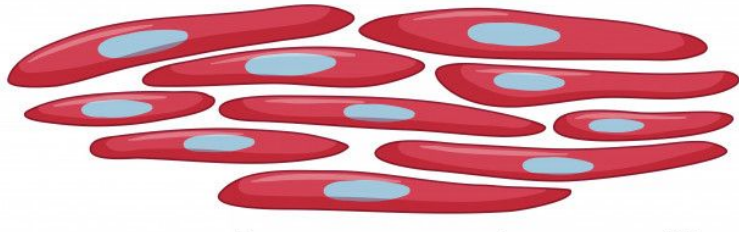
Expression data analysis 2

## By the end of this lesson you will...

- Understand the stages involved in DE analysis
- Be able to generate and explore RNA-seq read count tables
- Be familiar with the statistical models behind differential gene expression analysis
- Know how to perform differential gene expression analysis using the DESeq2 R package

# Experimental data

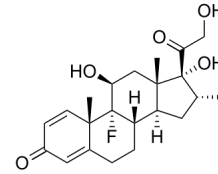
- We'll follow the analysis workflow of data from [Himes et al., 2014](#)<sup>1</sup>
- They studied the effect of the steroid Dexamethasone on human airway smooth muscle cells
- Four cell lines - treated/untreated
- Full analysis R code can be found in the [DESeq2 tutorial](#)



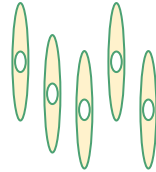
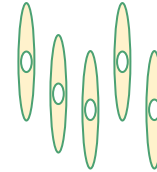
1. Himes, Blanca E., et al. "RNA-Seq transcriptome profiling identifies CRISPLD2 as a glucocorticoid responsive gene that modulates cytokine function in airway smooth muscle cells." *PloS one* 9.6 (2014).

**Goal:** Study the mechanism of dexamethasone action

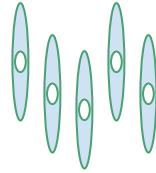
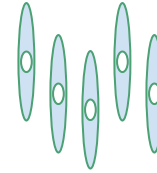
**Method:** Which genes are differentially-expressed between treated and untreated samples?



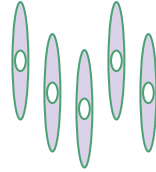
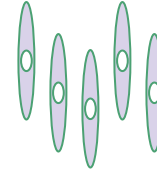
N61311



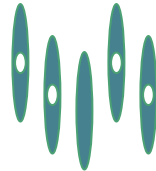
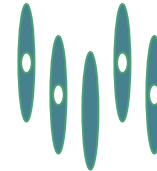
N052611



N080611

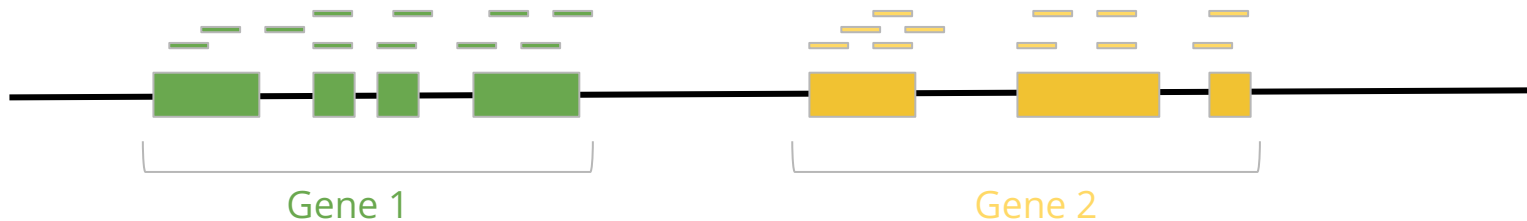


N061011



# Expression quantification from mapped reads

- Goal: determine how many RNA-seq reads mapped to each gene
- Reason: this is our proxy for gene expression level
- Input:
  - RNA-seq reads (spliced) mapping - BAM
  - Gene annotation - GFF
- Output: expression matrix  $M$   
 $M_{ij}$  = number of reads mapped to gene  $i$  in sample  $j$

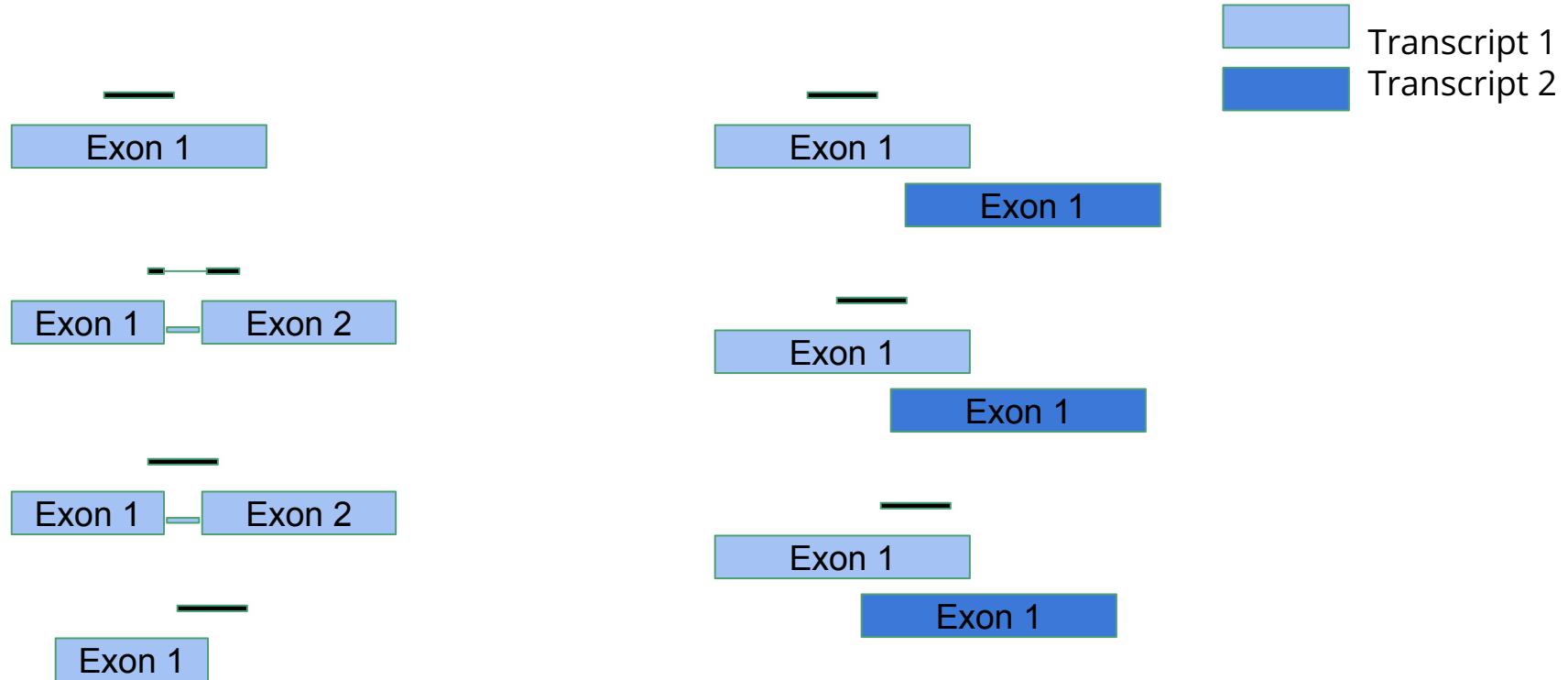


# Counts matrix

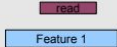
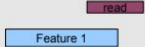





Cell line	N61311	N052611	N080611	N061011	N61311	N052611	N080611	N061011
Dex treatment	+	+	+	+	-	-	-	-
<i>C7</i>	34	512	66	121	25	344	297	76
<i>CCDC69</i>	5	8	8	3	12	7	10	7
<i>DUSP1</i>	1112	985	1003	898	214	128	188	203
<i>FKBP5</i>	33	94	111	42	46	98	57	85
...								

Total: ~64k transcripts (mRNAs)

# Assigning reads to genes - Not always straightforward



# featureCounts package - summarizeOverlaps

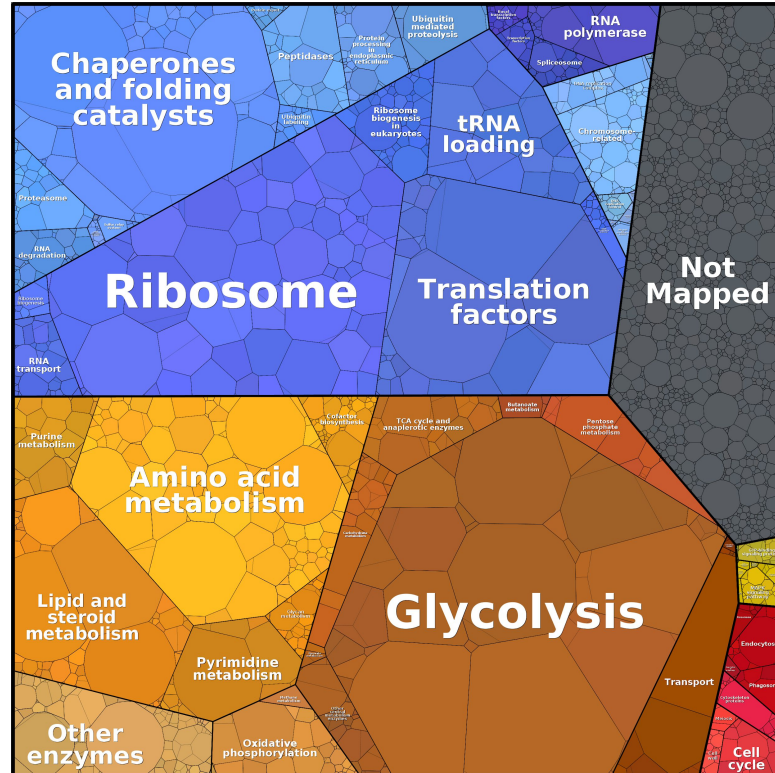
	Union	IntersectionStrict	IntersectionNotEmpty
	Feature I	Feature I	Feature I
	Feature I	No hit	Feature I
	Feature I	No hit	Feature I
	Feature I	Feature I	Feature I
	Feature I	Feature I	Feature I
	No hit	Feature 1	Feature I
	No hit	No hit	No hit



# Exploring expression levels data

- Useful as preparation for differential gene expression analysis
- Allows detection of trends in the counts data
- Basic QA
- **Main goal**: determine overall similarity between samples

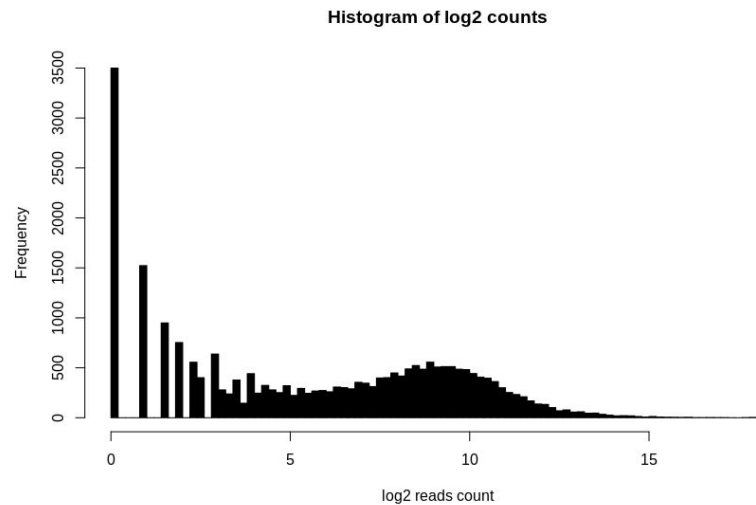
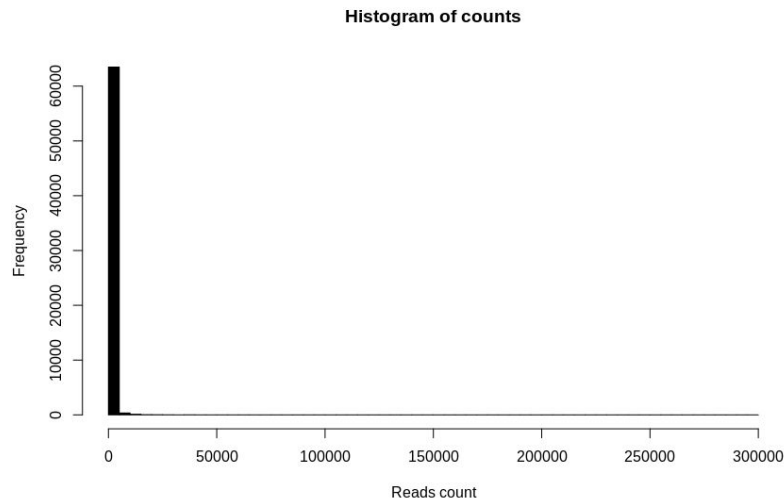
# Expression levels differ in orders of magnitude between genes



Liebermeister W., Noor E., Flamholz A., Davidi D., Bernhardt J., and Milo R. (2014), Visual account of protein investment in cellular functions. PNAS 111 (23), 8488-8493.

# The log transformation

- We usually apply a  $\log_2$  transformation to read counts
- Makes it easier to explore the data

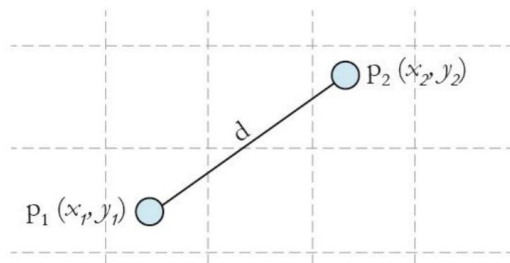


# QA of expression quantification by sample distances

- Which samples are overall similar/different from one another?
- Does it match our expectations, given the experimental design?
- We can use Euclidean distances:

**In n dimensions**

**In 2 dimensions**

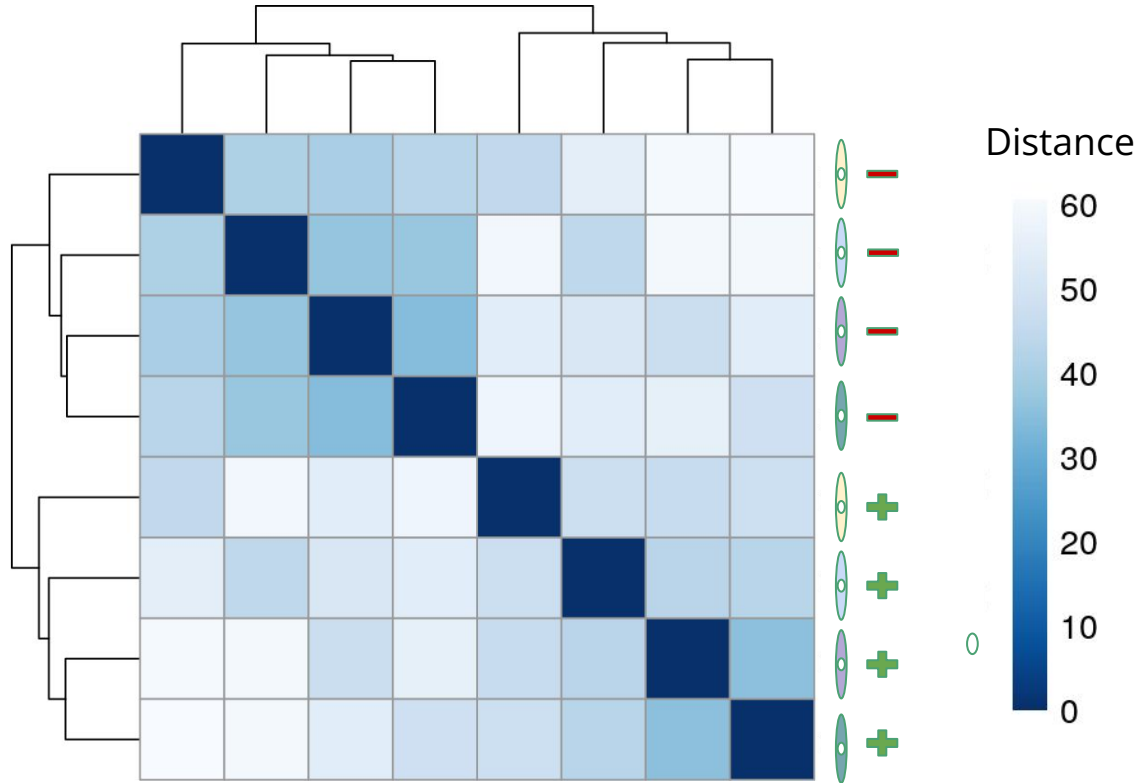


$$\text{Euclidean distance } (d) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

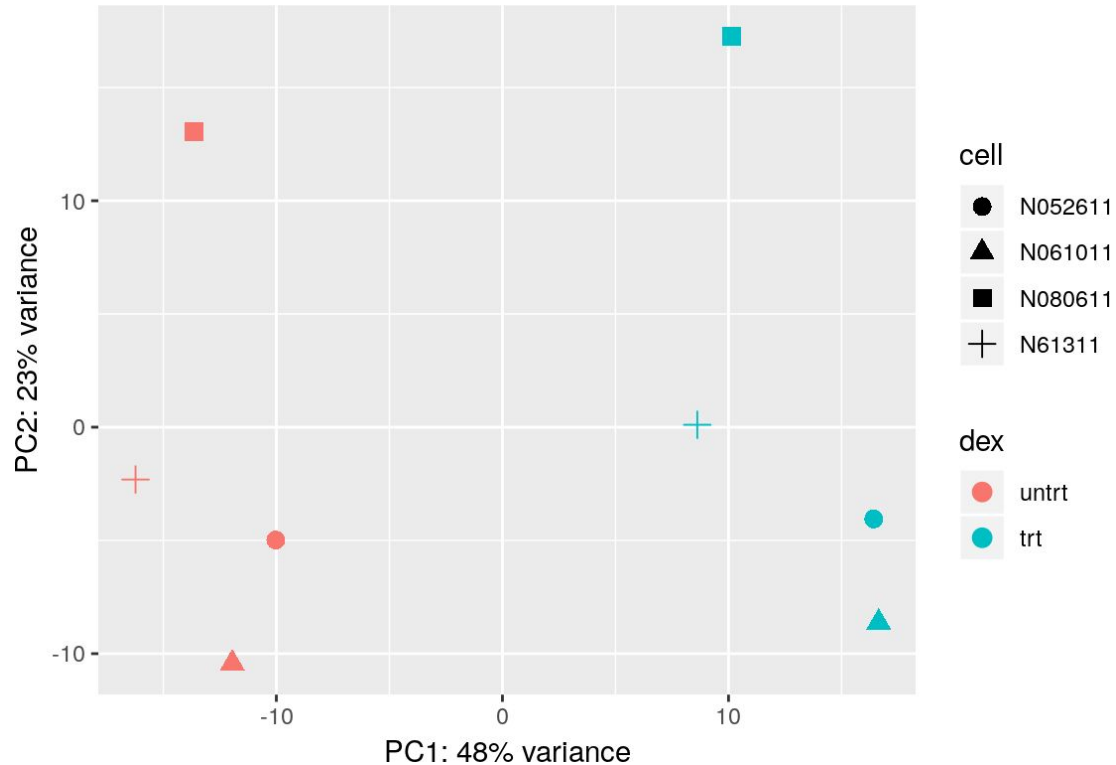
	sample1	sample2
gene1	$m_{11}$	$m_{12}$
gene2	$m_{21}$	$m_{22}$
...	...	...
gene n	$m_{n1}$	$m_{n2}$

$$d = \sqrt{(m_{12} - m_{11})^2 + (m_{22} - m_{21})^2 + \dots + (m_{n2} - m_{n1})^2}$$

# Hierarchical clustering and heatmap



# Principal component analysis (PCA)



# Filtering counts data

- It is useful to remove genes with very few reads from the analysis
  - Slow down the analysis
  - Reduce detection power for other genes
- We won't be able to detect DE anyway
- We can choose a count cutoff
- Or we can remove the  $X^{\text{th}}$  percentile
- In the Himes et. al data:
  - ~64k transcripts → Require  $\geq 10$  reads → ~20k transcripts

# Differential expression analysis

- Goal: detect genes that significantly differ in their expression levels between samples
- Input:
  - Normalized, filtered expression quantification matrix
  - Description of the experimental design
- Output: per gene - estimated difference between samples and **significance level**



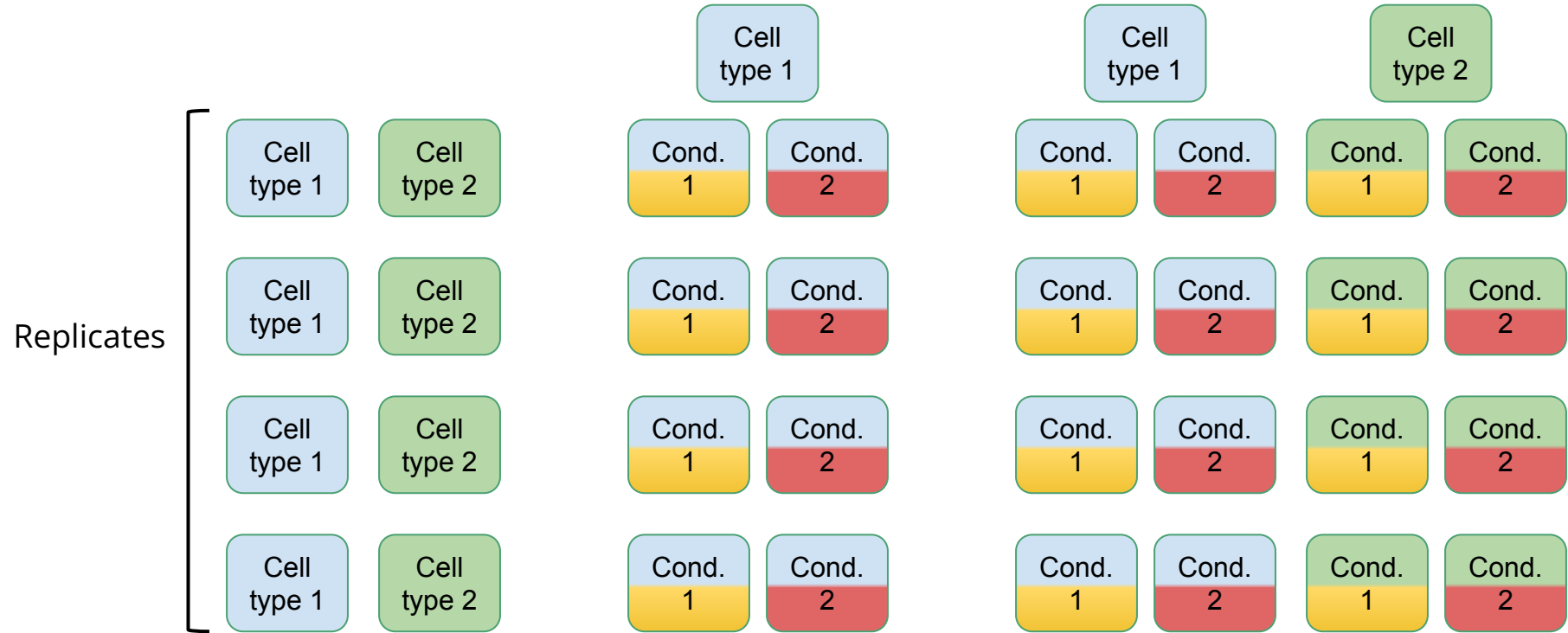
# Experimental design

- What samples were tested
- What conditions were tested
- What experimental replicates were made
- Experimental design must match the biological question
- Affects the statistical tests applied
- Experimental design is represented by the samples information table

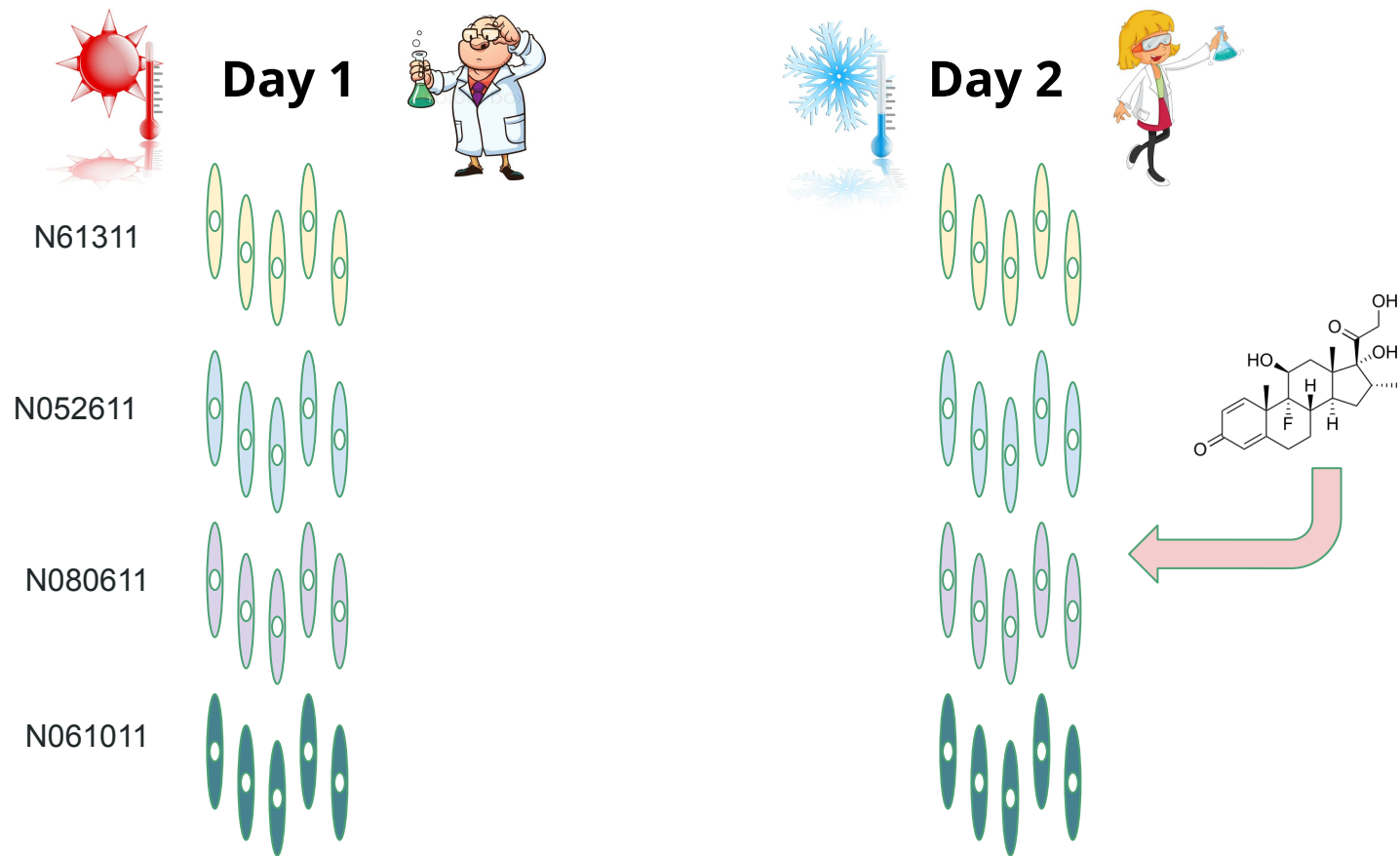
# The sample information table

Sample	Cell line	Dex
SRR1039508	N61311	Untreated
SRR1039509	N61311	Treated
SRR1039512	N052611	Untreated
SRR1039513	N052611	Treated
SRR1039516	N080611	Untreated
SRR1039517	N080611	Treated
SRR1039520	N061011	Untreated
SRR1039521	N061011	Treated

# Some design examples



# Batch effects



# Batch effects

- Introduced during sample handling and preparation
  - Technical factors
  - External factors
- Minimize by:
  - Use the same protocol for all samples
  - Prepare all samples together
- Not always possible
- We must test for batch effects when performing DE statistical tests

# Experiment design with batches

Sample	Cell line	Dex	Batch
SRR1039508	N61311	Untreated	1
SRR1039509	N61311	Treated	1
SRR1039512	N052611	Untreated	1
SRR1039513	N052611	Treated	1
SRR1039516	N080611	Untreated	2
SRR1039517	N080611	Treated	2
SRR1039520	N061011	Untreated	2
SRR1039521	N061011	Treated	2

# Fold changes

- The main measure used in DGE analysis is **fold change** - a.k.a ratio

$$R = \frac{Count_{sample1}}{Count_{sample2}}$$

- Ratios are highly non-symmetric
- Therefore we use log scaling - **log2 fold change (L2FC)**

$$L2FC = \log_2\left(\frac{Count_{sample1}}{Count_{sample2}}\right) = \log_2 Count_{sample1} - \log_2 Count_{sample2}$$

# Example - calculating fold changes

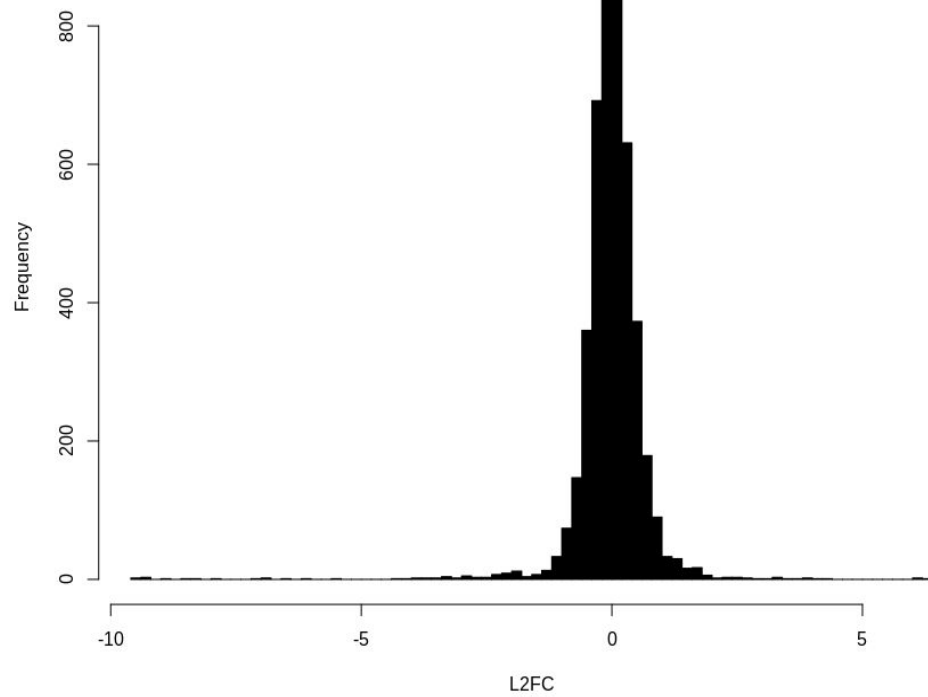
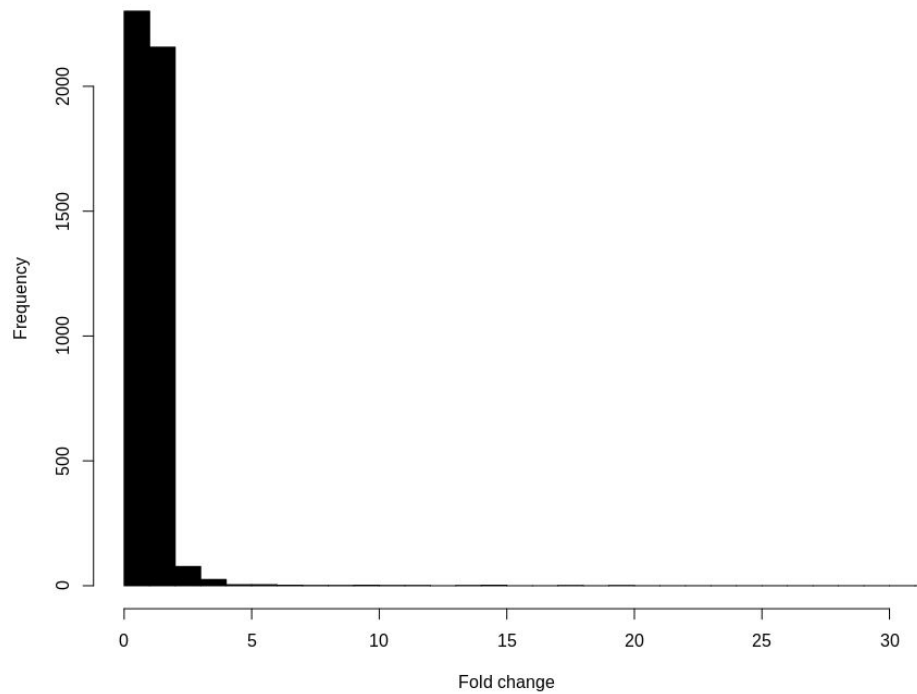
	N61311 Dex	N61311 Dex	N05261 1 Dex	N08061 1 Dex	N61311 Unt	N61311 Unt	N05261 1 Unt	N08061 1 Unt	Mean Dex	Mean Unt	FC	L2FC
<i>Gene 1</i>	34	512	66	121	25	344	297	76	183.25	185.50	0.99	-0.02
<i>Gene 2</i>	1112	985	1003	898	214	128	188	203	999.50	183.25	5.45	2.45
<i>Gene 3</i>	6	9	4	6	12	9	15	16	6.25	13.00	0.48	-1.06

L2FC = 0 : no difference

L2FC > 0 : sample1 expression > sample 2 expression

L2FC < 0 : sample1 expression < sample 2 expression

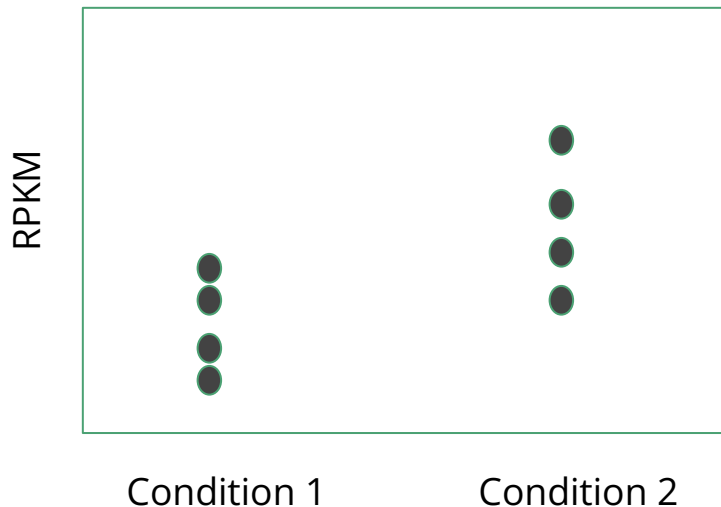




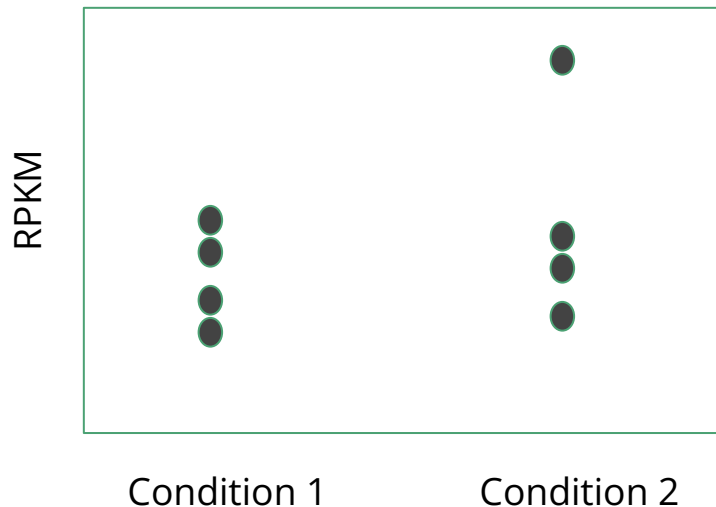
# Is this gene really differentially expressed?

- Can we tell just by looking at L2FC values?
- Maybe it's just random noise?
- Replicates can help

**Gene X** - L2FC = 0.5

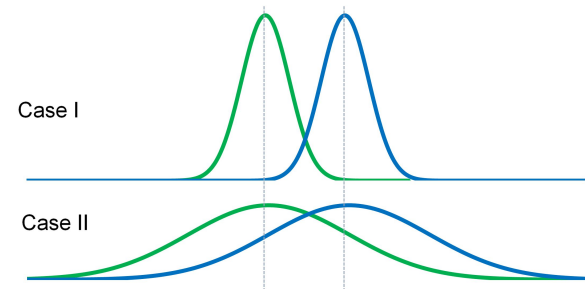


**Gene Y** - L2FC = 0.4



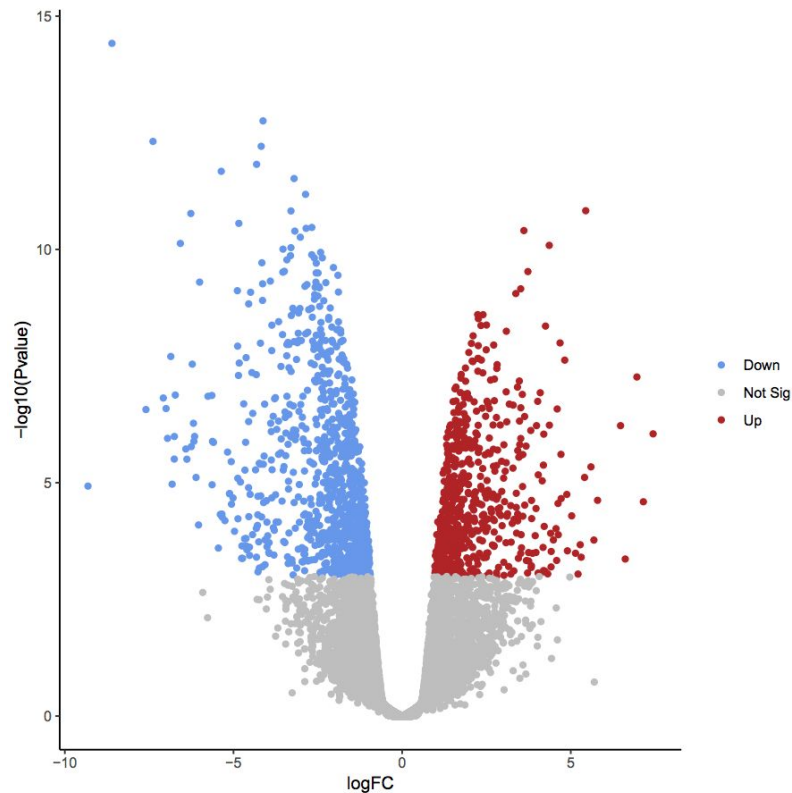
# Hypothesis testing

- $H_0$ : there is no difference in expression levels between samples
- $H_1$ : expression levels differ between samples
- We try and reject  $H_0$  with an appropriate statistical test, e.g.:
  - Parametric tests:  $t$ -test, ANOVA
  - Non-parametric tests: Mann-Whitney U test
  - Other modeling methods: linear models, GLM
- The result is a significance score - **per gene p-value**



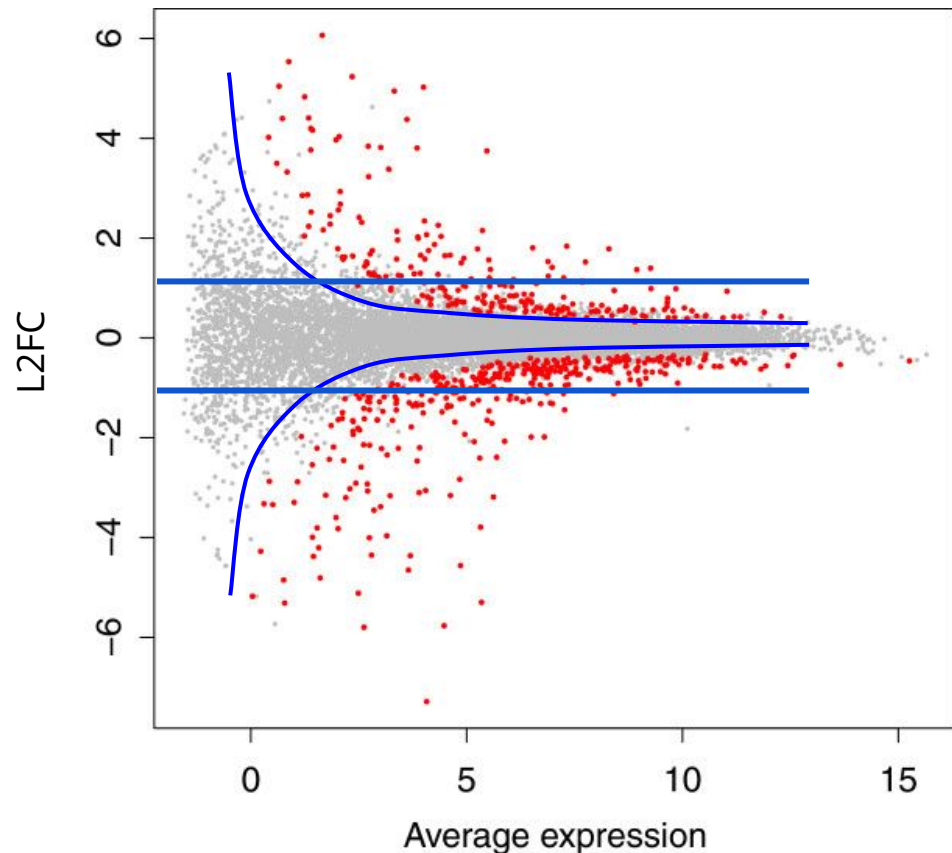
# Volcano plots - L2FC and p-value

- For each gene, we must consider both L2FC and p-value
- To get a global view - use a **volcano plot**
- We can choose a p-value cutoff, e.g. 0.05 or 0.01



# Choosing a L2FC cutoff

- We can choose an arbitrary cutoff, e.g. 1
- Lowly-expressed genes usually display higher variability in expression
- This can be seen in a **MA-plot**
- Can be used as a sanity-check
- Or to choose dynamic L2FC cutoffs



# Correcting for multiple testing

- Recall the meaning of a p-value...
- Since we are performing multiple tests, we must correct (adjust) p-values
- The simple and stringent way - **Bonferroni correction**

$$p_{adj} = p \times [\# \text{ of genes}]$$

- The common way - **False Discovery Rate** (FDR - BH procedure):
  1. Order p-values from smallest to largest

$$p_1, p_2, \dots, p_k, \dots, p_m$$

2.

$$p_{adj}^k = \frac{p^k \times [\# \text{ of genes}]}{k}$$

# DE with the pyDESeq2 Python package

- The DESeq2 package provides a full solution to DE analysis
  - Dedicated data structures
  - Statistical testing
  - Data transformations
  - Plotting
- Takes an **un-normalized** expression matrix - normalization performed internally
- Performs statistical tests using the negative Binomial distribution and GLMs
- Works great with small sample sizes (2-3 replicates)

# Specifying experimental designs in DESeq2

- We use a **design formula**
- The formula refers to a samples-info table provided by the user
- The formula describes a model for read counts
- Each column of the table is a factor
- Always starts with a ‘~’ symbol
- Then include all the relevant factors:

`~ factor1 + factor2 + ...`

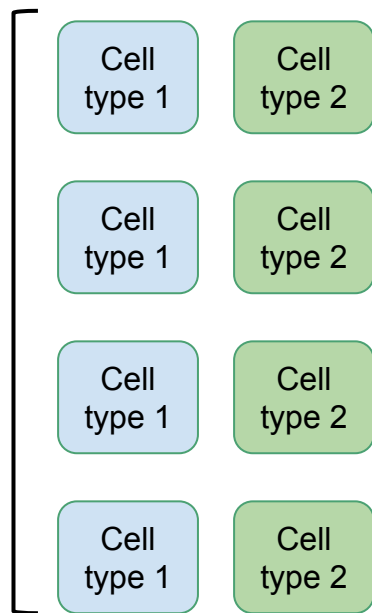
- Put the factor of interest **last**

Sample ID	Factor1	Factor2
s1	1	1
s2	1	2
s3	2	1
s4	2	2



# Design formulas - examples

Replicates

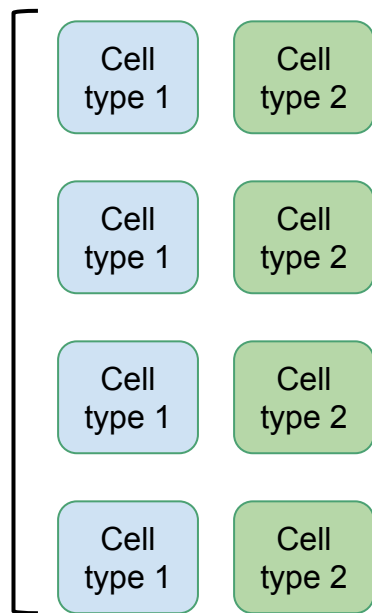


Sample ID	Cell-type	Batch
s1	1	1
s2	1	1
s3	1	1
s4	1	1
s5	2	1
s6	2	1
s7	2	1
s8	2	1

`~ celltype`

# Design formulas - examples

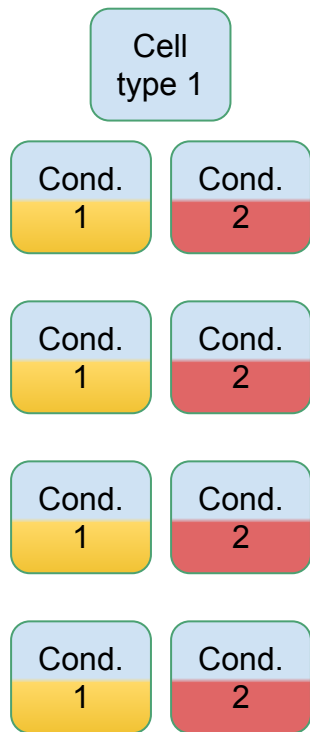
Replicates



Sample ID	Cell-type	Batch
s1	1	1
s2	1	1
s3	1	2
s4	1	2
s5	2	3
s6	2	3
s7	2	4
s8	2	5

`~ batch + celltype`

# Design formulas - examples



Sample ID	Cell-type	Batch	Condition
s1	1	1	1
s2	1	1	1
s3	1	2	1
s4	1	2	1
s5	1	3	2
s6	1	3	2
s7	1	4	2
s8	1	4	2

~ batch + condition

# Design formulas - examples

Sample	Cell line	Dex
SRR1039508	N61311	Untreated
SRR1039509	N61311	Treated
SRR1039512	N052611	Untreated
SRR1039513	N052611	Treated
SRR1039516	N080611	Untreated
SRR1039517	N080611	Treated
SRR1039520	N061011	Untreated
SRR1039521	N061011	Treated

`~ cell_line + Dex`

# Himes et al. - DE analysis results

- A total of 316 DE genes between treated and untreated samples
- Top 5:

Gene	Dex RPKM	Untreated RPKM	Ln[Fold Change]	Test Statistic	Adj. P-Value
C7	38.41	3.76	-3.35	8.74	0
CCDC69	47.39	6.24	-2.92	8.61	0
DUSP1	144.96	18.26	-2.99	8.99	0
FKBP5	53.05	3.43	-3.95	10.52	0
GPX3	613.37	45.18	-3.76	9.19	0

