Models and contrasts in `R/DESeq2`

UNIVERSITY OF CAMBRIDGE

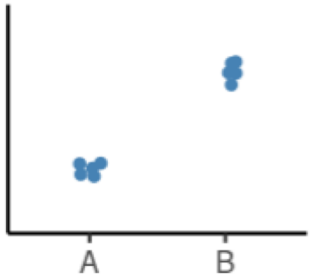
Bioinformatics Training Facility

In collaboration with:

CANCER RESEARCH UK | CAMBRIDGE CENTRE

Outline

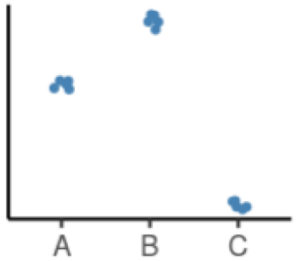
- How to interpret linear models coefficients
  - categorical variables & model matrix

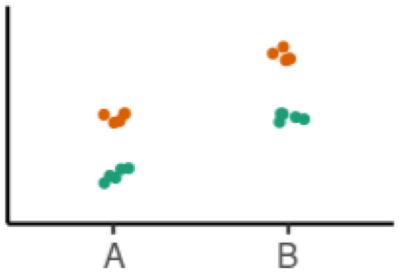


- How to specify models in R using the “formula syntax”

```
y ~ x
```

- How to interpret the results of different model designs
  - One factor, 3 levels
  - Two factors, additive
  - Two factors, interaction





- How *DESeq2* reports its results and how to interpret them

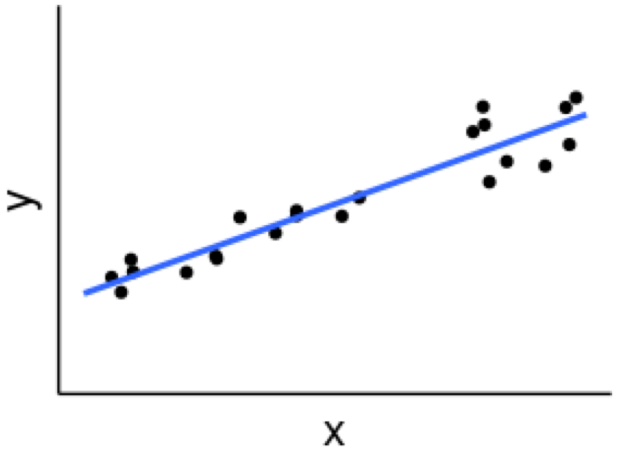
Linear Model

A model is a simplified representation of how we think different variables relate to each other.

Linear models are the most commonly used in statistical inference.

$$Y = \beta_0 + \beta_1 X + \epsilon$$

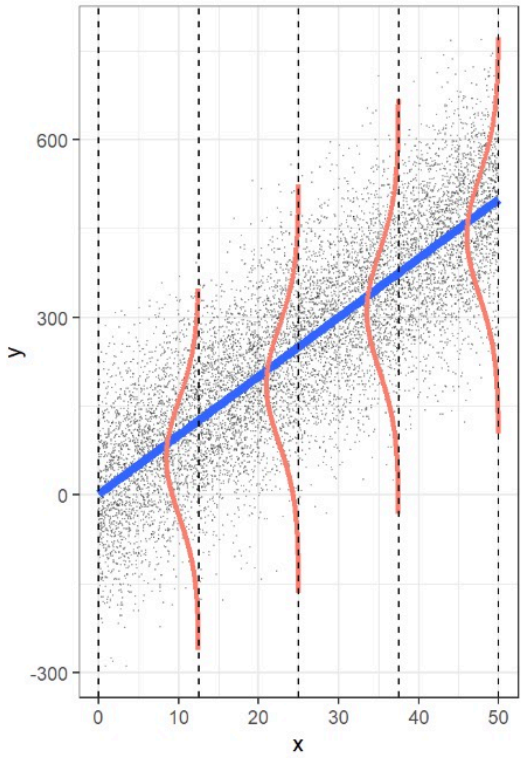
Intercept Slope Errors



X = Independent variable

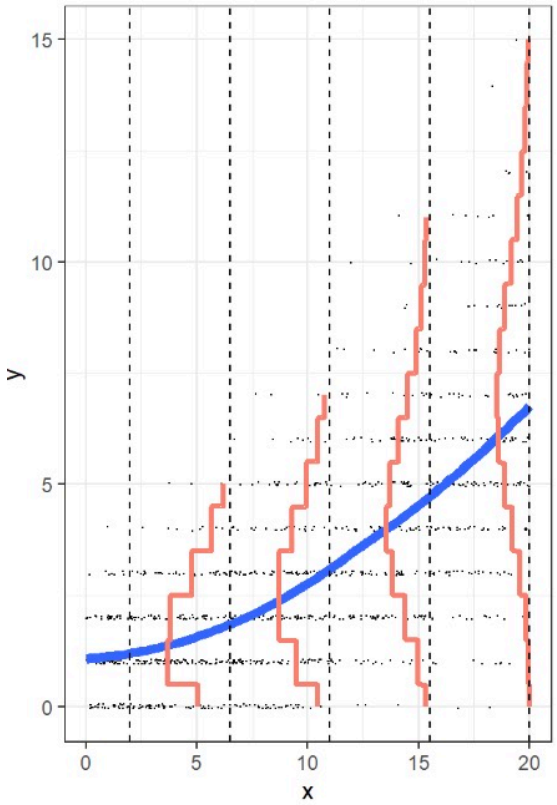
Y = Dependent variable

Linear Model assumptions



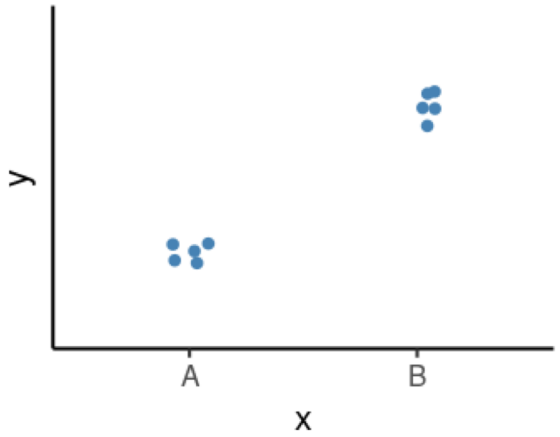
- **Important assumptions**
  - Errors / residuals are normally distributed around the fit line
  - Homoscedasticity: Error is constant along the values of the dependent variable.

Generalized Linear models (GLMs) in r



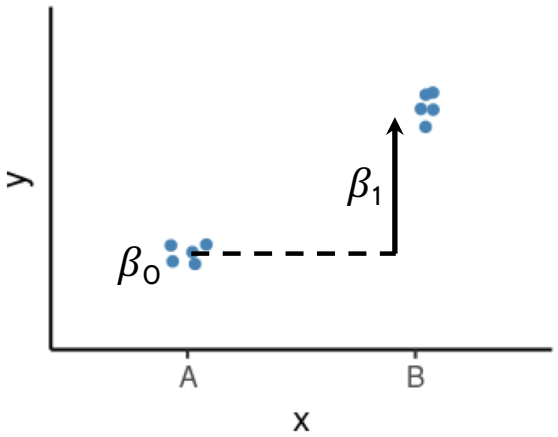
- In GLMs, linear model is generalized such that it can handle non-normal distribution of errors and heteroscedasticity.

```
          x
        <factor>
sample1   A
sample2   A
sample3   A
sample4   B
sample5   B
sample6   B
```



Linear Models in R | Categorical Variables

```
            x
        <factor>
sample1     A
sample2     A
sample3     A
sample4     B
sample5     B
sample6     B
```



Model:

$$Y = \beta_0 + \beta_1 X_B + \epsilon$$

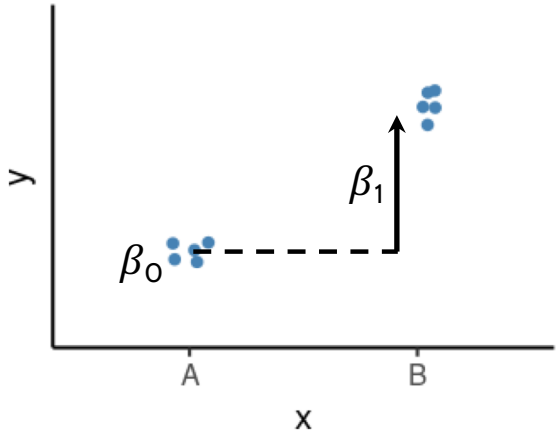
β_0 = *average of the reference group*

β_1 = ***difference*** to the reference group

Linear Models in R | Categorical Variables

	x	xB
	<factor>	
sample1	A	0
sample2	A	0
sample3	A	0
sample4	B	1
sample5	B	1
sample6	B	1

Indicator / Dummy variable



Model:

$$Y = \beta_0 + \beta_1 X_B + \epsilon$$

β_0 = *average of the reference group*

β_1 = ***difference*** to the reference group

Example:

$\beta_0 = 5;\ \beta_1 = 3$

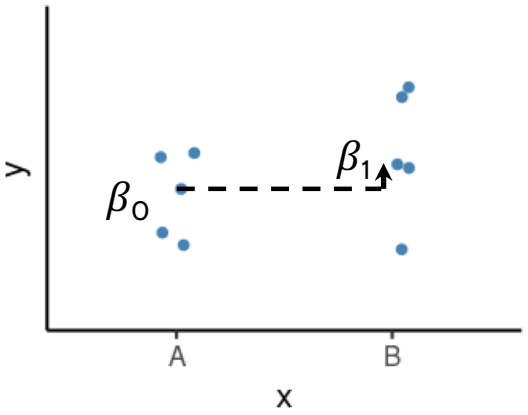
$Y = 5 + 3 * X_B$

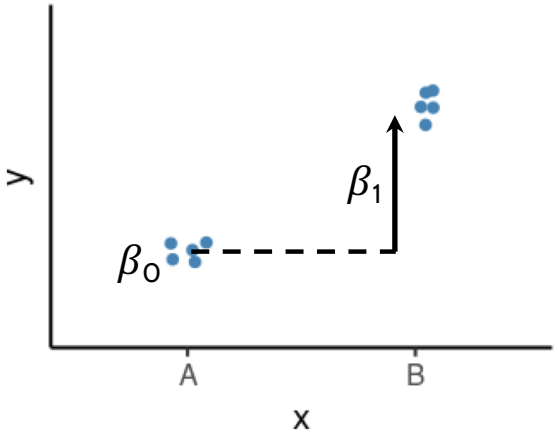
$$Y = 5 + \begin{cases} 3 * \mathbf{0} = 5 & \text{if "A"} \\ 3 * \mathbf{1} = 8 & \text{if "B"} \end{cases}$$

Linear Models in R | Null Hypothesis Testing

How compatible is my data with a "boring" hypothesis?

Null hypothesis: $\beta_1 = 0$





Model:

$$Y = \beta_0 + \beta_1 X_B + \epsilon$$

β_0 = *average of the reference group*

β_1 = ***difference*** to the reference group

Test statistic: $\beta_1 / \sigma_{\beta 1}$

(our estimate divided by the uncertainty in that estimate)

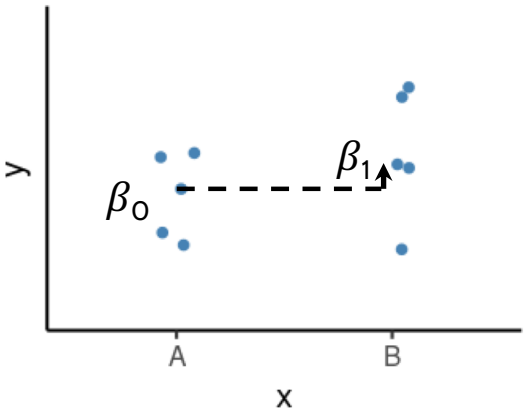
P-value calculated from the test statistic

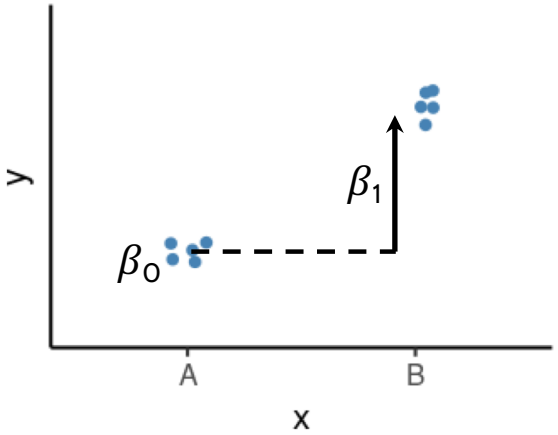
- Low p-value indicates that the data are not very compatible with the null hypothesis.

Linear Models in R | Null Hypothesis Testing

How compatible is my data with a "boring" hypothesis?

Null hypothesis: $\beta_1 = 0$





Model:

$$Y = \beta_0 + \beta_1 X_B + \epsilon$$

β_0 = *average of the reference group*

β_1 = ***difference*** to the reference group

Exercise 1

Test statistic: $\beta_1 / \sigma_{\beta 1}$

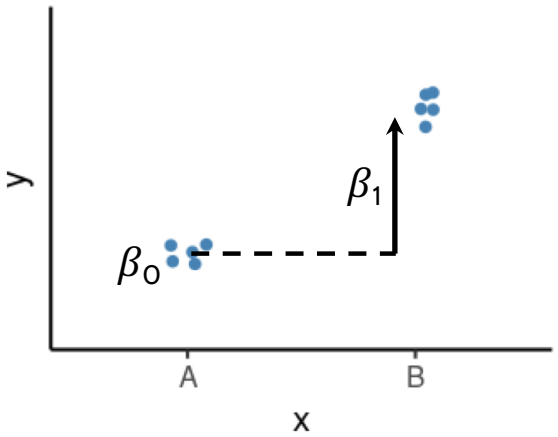
(our estimate divided by the uncertainty in that estimate)

P-value calculated from the test statistic

- Low p-value indicates that the data are not very compatible with the null hypothesis.

Linear Models in R | **Model Specification**

```
          x
        <factor>
sample1   A
sample2   A
sample3   A
sample4   B
sample5   B
sample6   B
```



Model:

$$Y = \beta_0 + \beta_1 X_B + \epsilon$$

β_0 = average of the reference group
β_1 = difference to the reference group

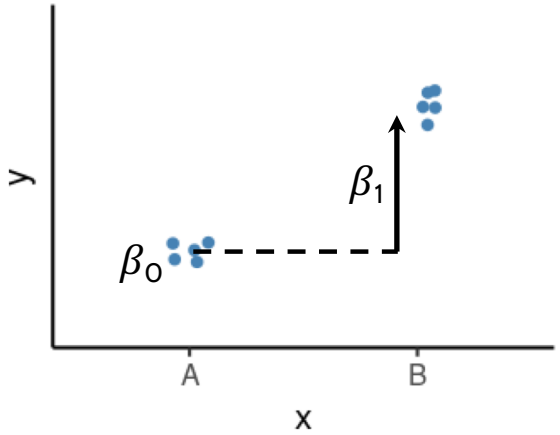
Formula syntax in R:

outcome ~ predictors

Linear Models in R | Model Specification

Formula syntax in R:

outcome ~ predictors



Model:

$$Y = \beta_0 + \beta_1 X_B + \epsilon$$

β_0 = *average of the reference group*

β_1 = ***difference*** to the reference group

```
          x
       <factor>
sample1    A
sample2    A
sample3    A
sample4    B
sample5    B
sample6    B
```

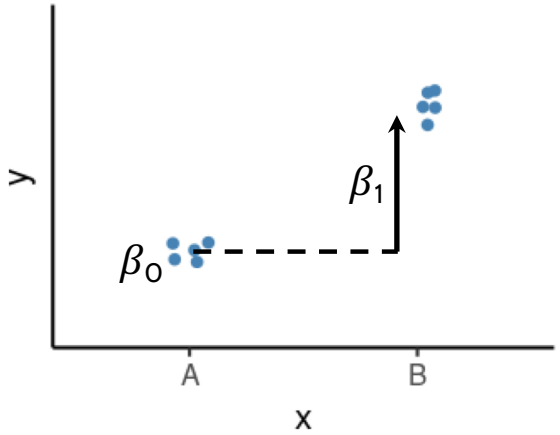
Design formula → ~ x → Model Matrix

Intercept	xB
1	0
1	0
1	0
1	1
1	1
1	1

Linear Models in R | **Model Specification**

Formula syntax in R:

outcome ~ predictors



Model:

$$Y = \beta_0 + \beta_1 X_B + \epsilon$$

β_0 = *average of the reference group*
β_1 = ***difference*** to the reference group

```
          x
       <factor>
sample1    A
sample2    A
sample3    A
sample4    B
sample5    B
sample6    B
```

Design formula → ~ x → Model Matrix

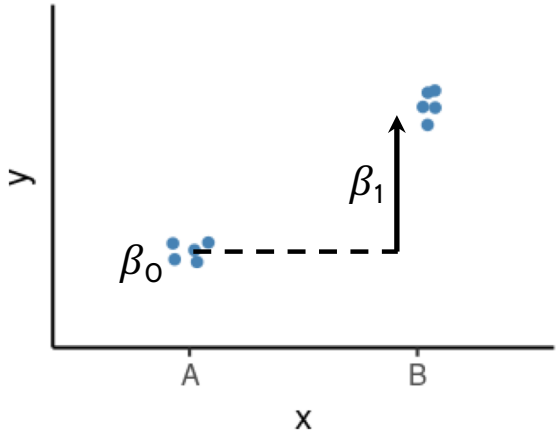
Intercept	xB
1	0
1	0
1	0
1	1
1	1
1	1

Example in R worksheet:
"Model Specification - Formula Syntax"

Linear Models in R | Model Specification

Formula syntax in R:

outcome ~ predictors



Model:

$$Y = \beta_0 + \beta_1 X_B$$

β_0 = *average of the reference group*

β_1 = ***difference*** to the reference group

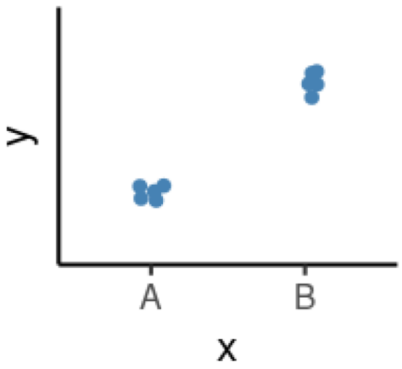
```
          x
       <factor>
sample1    A
sample2    A
sample3    A
sample4    B
sample5    B
sample6    B
```

Design formula → ~ x → Model Matrix

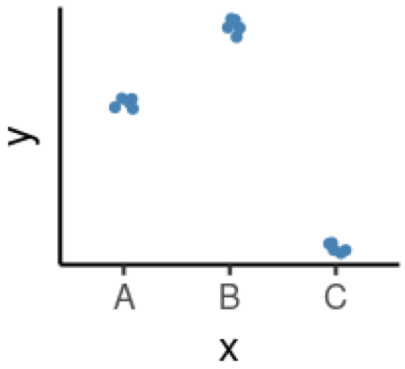
$$\begin{pmatrix} \text{Intercept} & \text{xB} \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} = \begin{pmatrix} 1*\beta_0 + 0*\beta_1 \\ 1*\beta_0 + 0*\beta_1 \\ 1*\beta_0 + 0*\beta_1 \\ 1*\beta_0 + 1*\beta_1 \\ 1*\beta_0 + 1*\beta_1 \\ 1*\beta_0 + 1*\beta_1 \end{pmatrix}$$

Common Designs

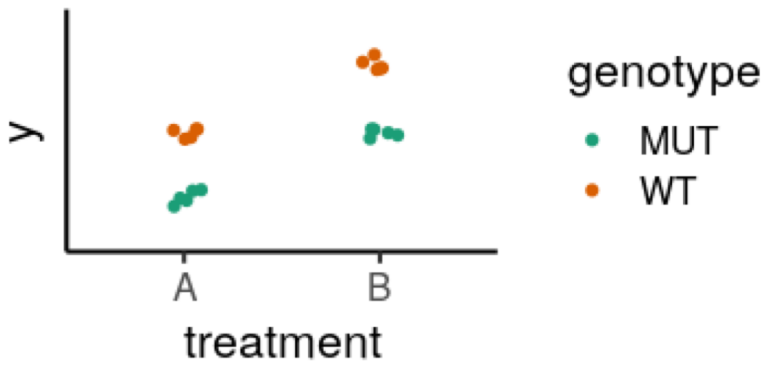
One factor, 2 levels



One factor, 3 levels



Two factors



- Define our model with formula syntax
- Categorical variables are encoded as indicator variables in a model matrix
  - R does this for us
- Interpret coefficients to define hypothesis of interest

Common Designs | One factor, 3 levels

	drug
sample1	Pink
sample2	Pink
sample3	Pink
sample4	Yellow
sample5	Yellow
sample6	Yellow
sample7	White
sample8	White
sample9	White

Design:

```
~ drug
```

Model matrix

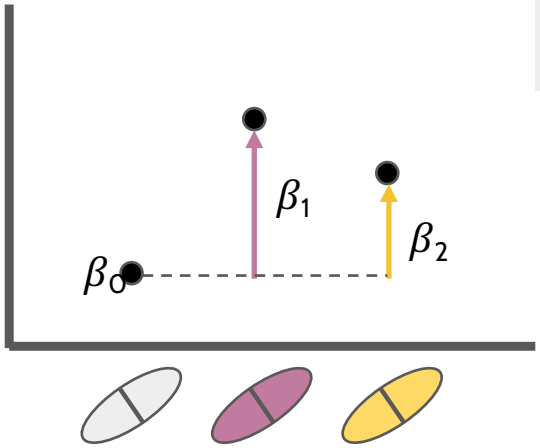
	(Intercept)	drugPink	drugYellow
1	1	1	0
2	1	1	0
3	1	1	0
4	1	0	1
5	1	0	1
6	1	0	1
7	1	0	0
8	1	0	0
9	1	0	0

Null hypothesis:

Pink vs White
$\beta_1 = 0$

Yellow vs White
$\beta_2 = 0$

Yellow vs Pink
$\beta_2 - \beta_1 = 0$



Expr = $\beta_0 + \beta_1 \text{drug}_{\text{Pink}} + \beta_2 \text{drug}_{\text{Yellow}}$

Model Designs | Two factors - additive model

	drug	genotype
sample1	Pink	WT
sample2	Pink	WT
sample3	Pink	MUT
sample4	Pink	MUT
sample5	White	WT
sample6	White	WT
sample7	White	MUT
sample8	White	MUT

Design:

```
~ drug + genotype
```

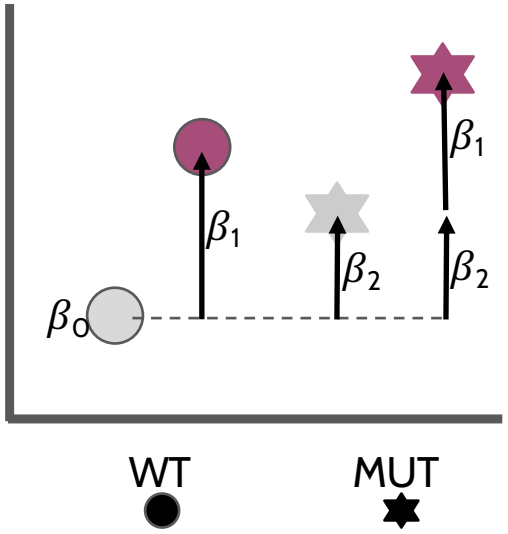
Model Matrix:

	(Intercept)	drugPink	genotypeMUT
1	1	1	0
2	1	1	0
3	1	1	1
4	1	1	1
5	1	0	0
6	1	0	0
7	1	0	1
8	1	0	1

Null hypothesis:

Pink vs White drug
$\beta_1 = 0$

WT vs MUT genotype
$\beta_2 = 0$



Expr = $\beta_0 + \beta_1 \text{drug}_{\text{Pink}} + \beta_2 \text{genotype}_{\text{MUT}}$

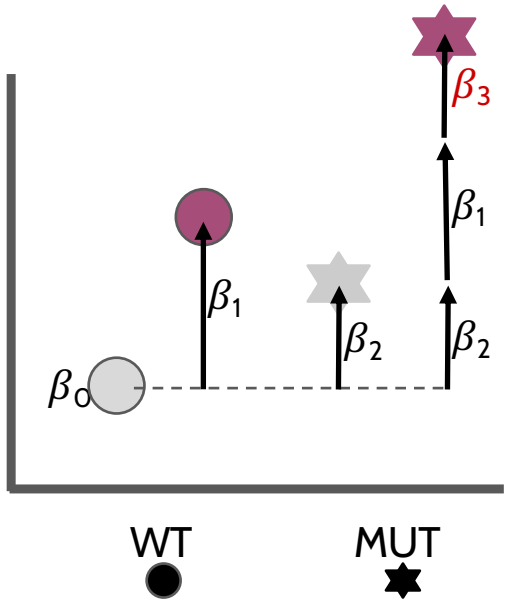
Model Designs | Two factors - interaction model

	drug	genotype
sample1	Pink	WT
sample2	Pink	WT
sample3	Pink	MUT
sample4	Pink	MUT
sample5	White	WT
sample6	White	WT
sample7	White	MUT
sample8	White	MUT

Design:

```
~ drug + genotype + drug:genotype
```

Expr = β_0 + β_1 $drug_{Pink}$ + β_2 $genotype_{MUT}$ + β_3 $drug_{Pink}genotype_{MUT}$



Null hypothesis:

Pink vs White (WT)
$\beta_1 = 0$

Pink vs White (MUT)
$\beta_1 + \beta_3 = 0$

WT vs MUT (White)
$\beta_2 = 0$

WT vs MUT (Pink)
$\beta_2 + \beta_3 = 0$

Interaction ("Difference of differences"):
$\beta_3 = 0$

Model Specification in *DESeq2*

- Create DESeqDataSet object
- Add model design:

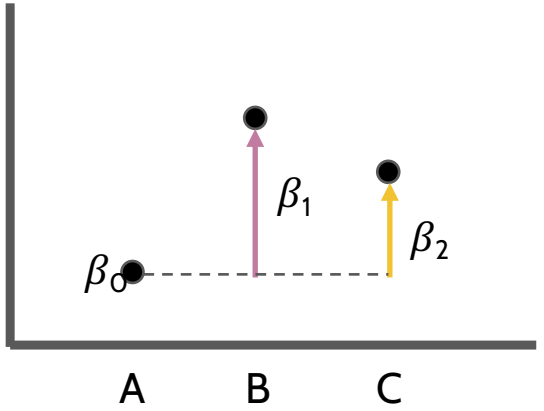
```
design(dds) ← ~ treatment
```

- Fit the statistical model

```
dds ← DESeq(dds)
```

- Check coefficients for hypothesis testing

```
resultsNames(dds)
```



Model Specification in *DESeq2*

- Create DESeqDataSet object
- Add model design:

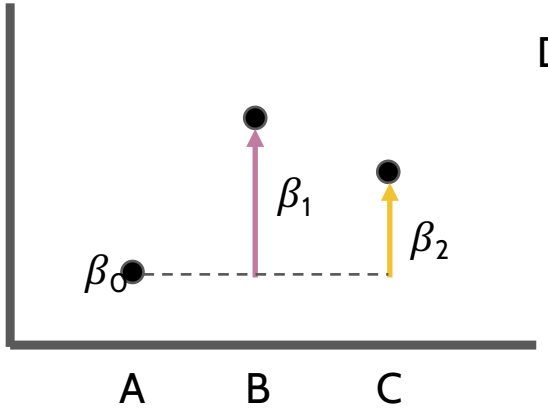
```
design(dds) ← ~ treatment
```

- Fit the statistical model

```
dds ← DESeq(dds)
```

- Check coefficients for hypothesis testing

```
resultsNames(dds)
```



DESeq coefficient names:

$\beta_0 \rightarrow$ Intercept
$\beta_1 \rightarrow$ treatment_B_vs_A
$\beta_2 \rightarrow$ treatment_C_vs_A

	Null Hypothesis
B vs A	$\beta_1 = 0$
C vs A	$\beta_2 = 0$
C vs B	$\beta_2 - \beta_1 = 0$

Model Specification in *DESeq2* | Interpreting the Results

```
results(dds, contrast = list("treatment_B_vs_A"))
```

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
gene1	32.80405	0.359444	0.598072	0.601004	0.5478372	0.923764
gene2	4.01072	3.407763	1.649827	2.065527	0.0388732	0.641407
gene3	7.01837	0.743337	0.994100	0.747749	0.4546118	0.923764
gene4	1.51006	2.814822	2.464686	1.142061	0.2534287	0.923764
gene5	11.23166	0.480522	0.894709	0.537071	0.5912189	0.923764
...	...	...	...	...	...	...
gene96	16.21864	0.684962	0.809892	0.845745	0.3976952	0.923764
gene97	2.91349	1.784327	1.790046	0.996805	0.3188590	0.923764
gene98	13.29915	-0.634070	0.768728	-0.824830	0.4094680	0.923764
gene99	82.45653	-0.963147	0.505109	-1.906810	0.0565452	0.799710
gene100	6.25763	1.673078	1.252839	1.335429	0.1817359	0.923764

baseMean → Mean across *all* samples

log2FoldChange → $\log_2$(B/A) i.e. the difference between treatments

lfcSE → the standard error of the log2FoldChange

stat → the test statistic = log2FoldChange/lfcSE

pvalue → the p-value of the Wald test

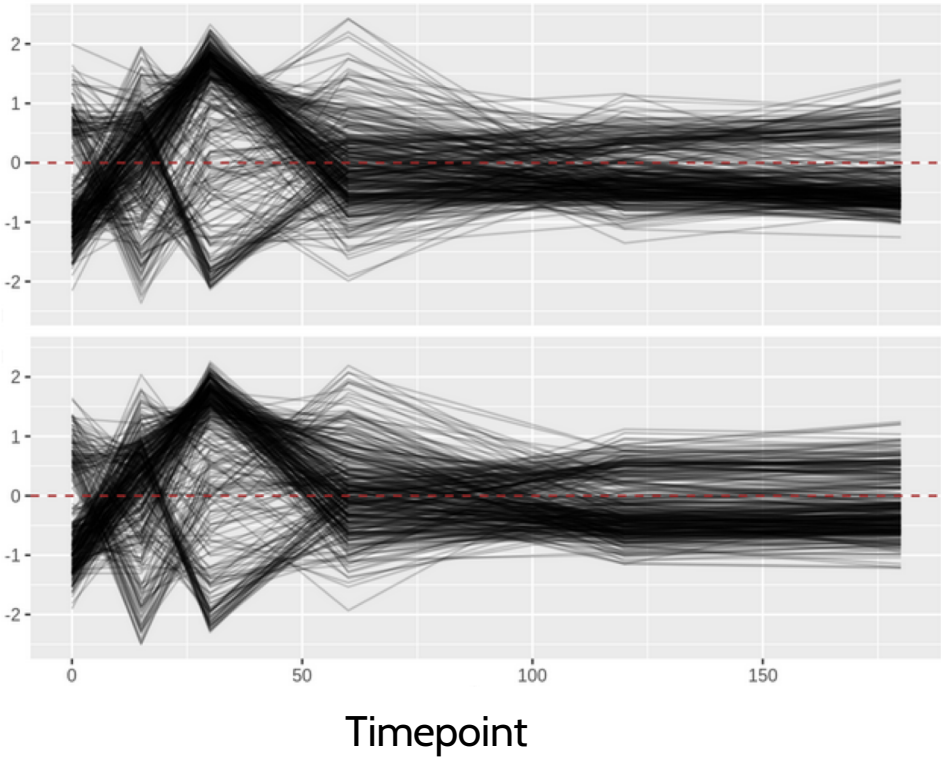
padj → the p-value adjusted for multiple testing (false discovery rate)

Model Specification in *DESeq2* | Likelihood-ratio Test

The default test in *DESeq2* is the Wald test, testing for null hypothesis that LFC = 0

And alternative is the **Likelihood Ratio Test**

$$LR = -2ln\left(\frac{L(m_1)}{L(m_2)}\right)$$



WT

Mutant

Example:

```
design(dds) ← ~ genotype + timepoint + genotype:timepoint

dds ← DESeq(dds, test = "LRT",
            reduced = ~ genotype)
```

Conclusions

- Differential expression tests are based on linear models, where the gene expression is modelled as an outcome of several variables of interest (e.g. treatment, genotype, infection status, etc.).
- Linear models use *indicator or dummy variables* to encode categorical variables in a model matrix.
- To define models in R/DESeq2 we use the formula syntax: `~ variables`
- Some common models are:
  - Single factor: `~ variable1`
  - Two factor, additive: `~ variable1 + variable2`
  - Two factor, interaction: `~ variable1 + variable2 + variable1:variable2`
- Interpreting our model coefficients allows us to define hypothesis/comparisons/contrasts of interest.
- In DESeq2 we use the `results()` function to obtain the log2(fold-change) in gene expression between groups of interest ("contrast").