

# Linear models and ANOVA

Keegan Korthauer

26 January 2022

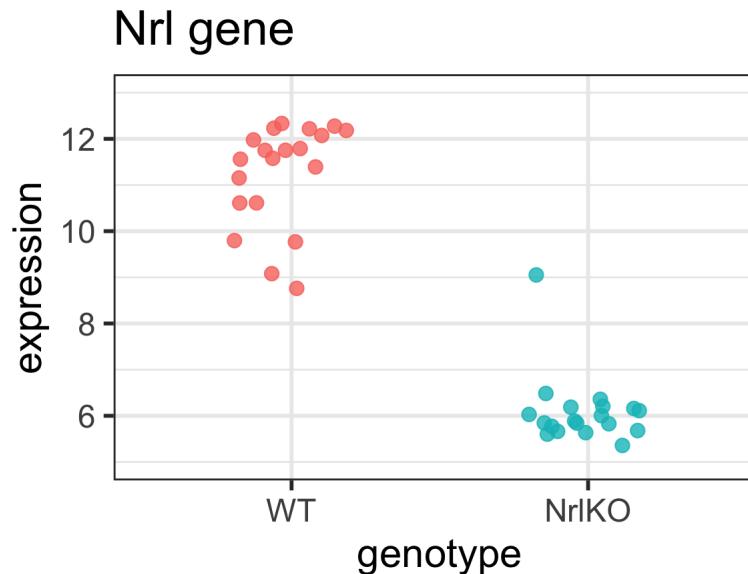
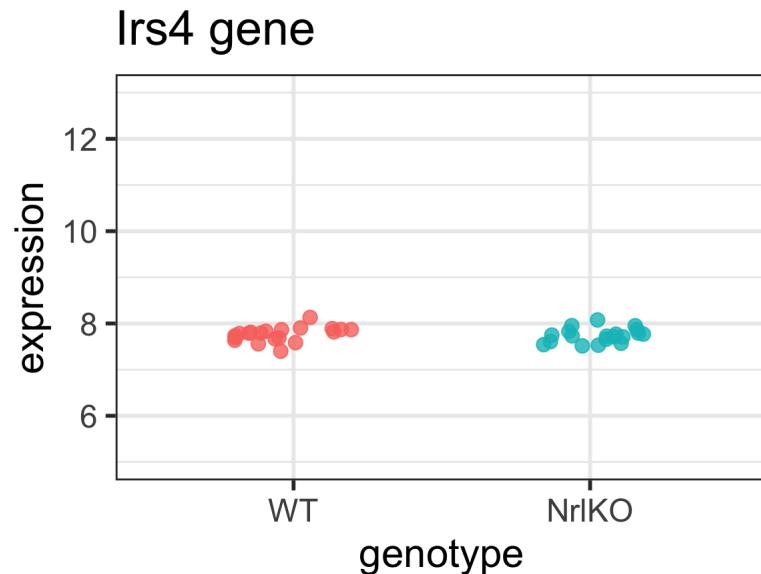
with slide contributions from Gabriela Cohen Freue and Jenny Bryan



# Recap: Are these genes different in NrlKO vs WT?

$H_0$ : the expression level of gene  $g$  is the same in both conditions

Is there **enough** evidence in the data to reject  $H_0$ ?



# Statistics: learn about a population from a random sample

Population (Unknown)

$$Y \sim F, Z \sim G$$

$$E[Y] = \mu_Y, E[Z] = \mu_Z$$

$$Var[Y] = \sigma_Y^2, Var[Z] = \sigma_Z^2$$

$$H_0 : \mu_Y = \mu_Z$$

$$H_A : \mu_Y \neq \mu_Z$$

Sample (Observed, with randomness)

$$(Y_1, Y_2, \dots, Y_{n_Y}) \text{ and } (Z_1, Z_2, \dots, Z_{n_Z})$$

$$\hat{\mu}_Y = \bar{Y} = \frac{\sum_{i=1}^{n_Y} Y_i}{n_Y}$$

$$\hat{\sigma}_Y^2 = S_Y^2 = \frac{1}{n_Y} \sum_{i=1}^{n_Y} (Y_i - \bar{Y})^2$$

(with similar quantities for  $Z$ :  $\bar{Z}$  and  $S_Z^2$ )

$$T = \frac{\bar{Y} - \bar{Z}}{\sqrt{\hat{Var}(\bar{Y} - \bar{Z})}}$$

$\bar{Y}, \bar{Z}, S_Y^2, S_Z^2$  and  $T$  are examples of **statistics** computed from the sample

# Summary: Hypothesis testing

1. Formulate scientific hypothesis as a **statistical hypothesis** ( $H_0$  vs  $H_A$ )
2. Define a **test statistic** to test  $H_0$  and compute its **observed value**. For example:
  - 2-sample  $t$ -test
  - Welch  $t$ -test (unequal variance)
  - Wilcoxon rank-sum test
  - Kolmogorov-Smirnov test
3. Compute the probability of seeing a test statistic as extreme as that observed, under the **null sampling distribution** (p-value)
4. Make a decision about the **significance** of the results, based on a pre-specified significance level ( $\alpha$ )

# We can run these tests in R

Example: use the `t.test` function to test  $H_0$  using a classical 2-sample  $t$ -test with equal variance.

```
filter(twoGenes, gene == "Irs4") %>%  
  t.test(expression ~ genotype, data = ., var.equal = TRUE)
```

```
##  
##      Two Sample t-test  
##  
## data: expression by genotype  
## t = 0.52854, df = 37, p-value = 0.6003  
## alternative hypothesis: true difference in means between group WT and group NrlKO is not equal to  
## 0  
## 95 percent confidence interval:  
## -0.07384018 0.12595821  
## sample estimates:  
## mean in group WT mean in group NrlKO  
## 7.765671 7.739612
```

# Which test should I use??

- How to decide which test to carry out (e.g. t-test, Wilcoxon, KS)?
- Are assumptions met for each one?
  - If so, parametric tests (e.g. t-test) tend to have slightly higher power (ability to reject  $H_0$  when  $H_0$  is false)
  - But if assumptions are potentially violated, non-parametric tests (Wilcoxon, KS) are a safer choice (albeit conservative)

# Today's Learning Objectives

1. Compare means of different groups (2 or more) using a **linear regression model**
  - Understand how 'indicator' variables represent the levels of a qualitative explanatory variable
2. Write a linear model using matrix notation
  - understand which matrix is built by R
3. Distinguish between **single** and **joint** hypothesis tests
  - e.g.  $t$ -tests vs  $F$ -tests

# 3 ways to test $H_0 : \mu_1 = \mu_2$

## 2-sample t-test (with equal variance)

```
filter(twoGenes, gene == "Irs4") %>%  
  t.test(expression ~ genotype, data = ., var.equal = TRUE)
```

## (One-way) Analysis of Variance (ANOVA)

```
filter(twoGenes, gene == "Irs4") %>%  
  aov(expression ~ genotype, data = .) %>%  
  summary()
```

## Linear regression model

```
filter(twoGenes, gene == "Irs4") %>%  
  lm(expression ~ genotype, data = .) %>%  
  summary()
```

# All three methods give the same result!\*

## 2-sample t-test (equal variance)

```
##  
## Two Sample t-test  
##  
## data: expression by genotype  
## t = 0.52854, df = 37, p-value = 0.6003  
## alternative hypothesis: true difference  
in means between group WT and group NrlKO  
is not equal to 0  
## 95 percent confidence interval:  
## -0.07384018 0.12595821  
## sample estimates:  
## mean in group WT mean in group NrlKO  
## 7.765671 7.739612
```

## (One-way) Analysis of Variance (ANOVA)

```
## Df Sum Sq Mean Sq F value Pr(>F)  
## genotype 1 0.0066 0.006617 0.279 0.6  
## Residuals 37 0.8764 0.023685
```

## Linear regression model

```
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 7.76567 0.03441 225.659 <2e-16  
***  
## genotypeNrlKO -0.02606 0.04930 -0.529 0.6
```

\*Note differences in sign between t-test & linear regression: pay attention to which group is 'reference'

# These are not coincidences!

## 2-sample t-test (equal variance)

```
## $`t statistic`  
##      t  
## 0.5285386  
##  
## $`p-value`  
## [1] 0.6002819  
##  
## $`mean difference`  
## [1] 0.02605902  
##  
## $`(t statistic)^2`  
##      t  
## 0.279353
```

## (One-way) Analysis of Variance (ANOVA)

```
## $`F statistic`  
## [1] 0.279353  
##  
## $`p-value`  
## [1] 0.6002819
```

## Linear regression model

```
## $`t statistic`  
## [1] -0.5285386  
##  
## $`p-value`  
## [1] 0.6002819  
##  
## $`coefficient estimate`  
## [1] -0.02605902
```

# *t*-test vs linear regression: why the same results?

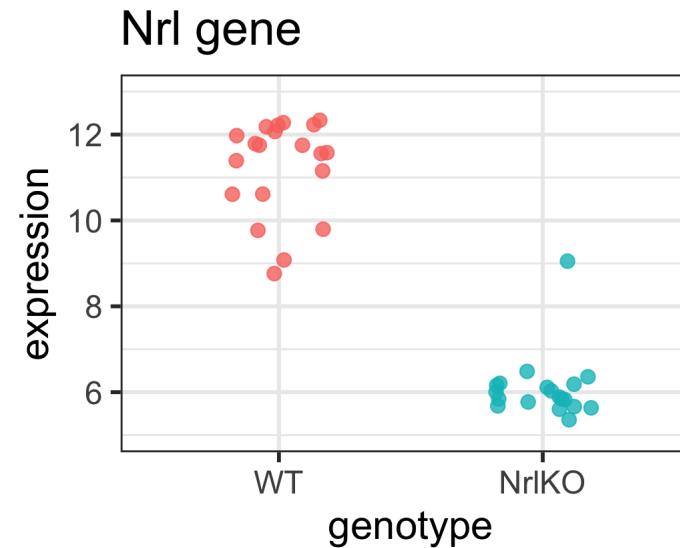
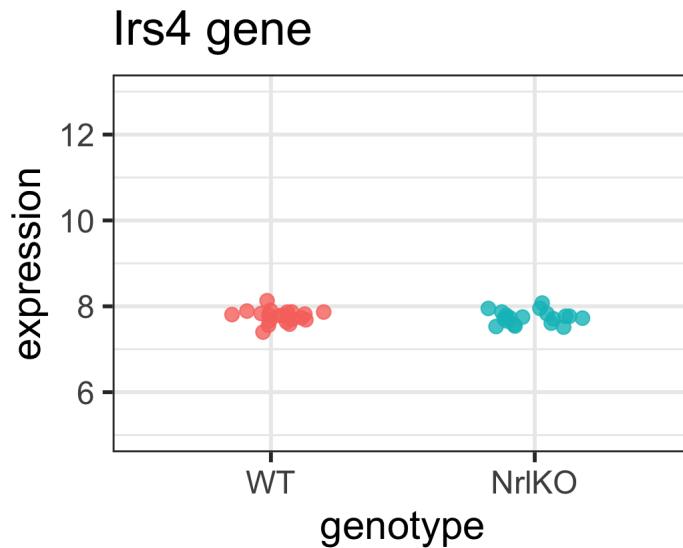
```
list("t statistic" = irs4.ttest$statistic,  
    "p-value" = irs4.ttest$p.value)
```

```
## $`t statistic`  
##      t  
## 0.5285386  
##  
## $`p-value`  
## [1] 0.6002819
```

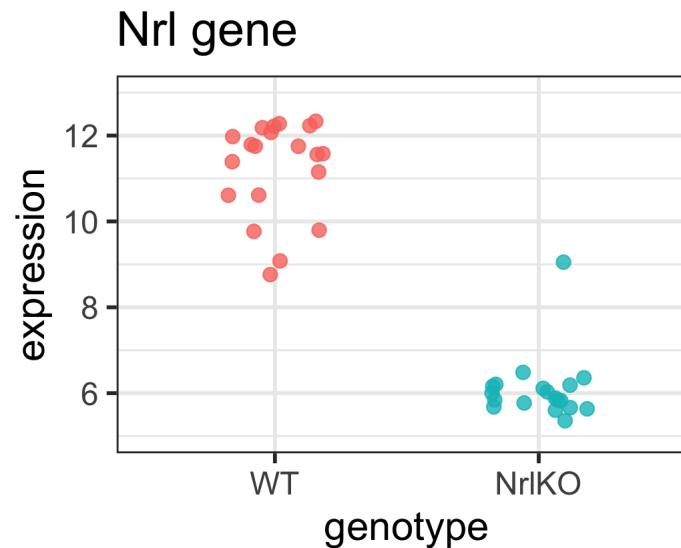
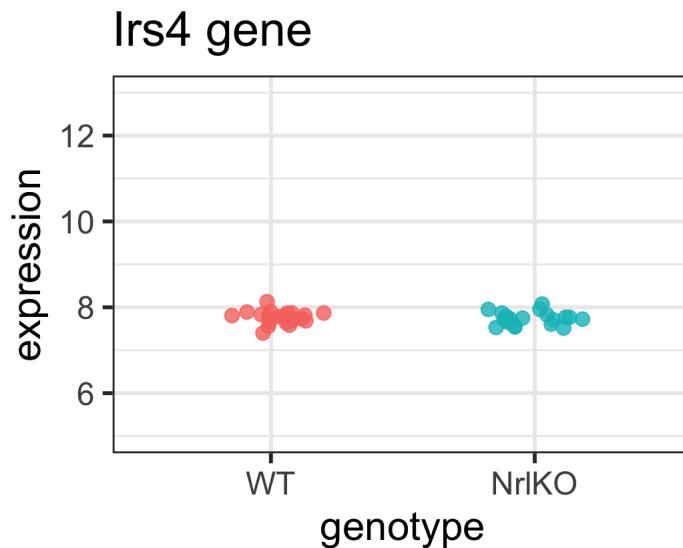
```
list("t statistic" = irs4.lm$coeff[2,3],  
    "p-value" = irs4.lm$coeff[2,4])
```

```
## $`t statistic`  
## [1] -0.5285386  
##  
## $`p-value`  
## [1] 0.6002819
```

# *t*-test vs linear regression: where's the *line*?



# *t*-test vs linear regression: where's the *line*?



Note that the *x*-axis in these plots is not numerical, thus a line in this space does not have any mathematical meaning.

Why can we run a t-test with a **linear** regression model?

# From $t$ -test to linear regression

Let's change the notation to give a common framework to all methods

$$Y \sim G; E[Y] = \mu_Y$$



$$Y = \mu_Y + \varepsilon_Y; \varepsilon_Y \sim G; E[\varepsilon_Y] = 0$$

# From $t$ -test to linear regression

Let's change the notation to give a common framework to all methods

$$Y \sim G; E[Y] = \mu_Y$$



$$Y = \mu_Y + \varepsilon_Y; \varepsilon_Y \sim G; E[\varepsilon_Y] = 0$$

## Why is this equivalent?

$$E[Y] = E[\mu_Y + \varepsilon_Y] = \mu_Y + E[\varepsilon_Y] = \mu_Y$$

We are just rewriting  $Y$  here

# From $t$ -test to linear regression

Let's change the notation to give a common framework to all methods

$$Y \sim G; E[Y] = \mu_Y$$



$$Y = \mu_Y + \varepsilon_Y; \varepsilon_Y \sim G; E[\varepsilon_Y] = 0$$

We can use indices to accommodate multiple groups, i.e.,

$$Y_{ij} = \mu_j + \varepsilon_{ij}; \varepsilon_{ij} \sim G_j; E[\varepsilon_{ij}] = 0;$$

where  $j = \{\text{WT, NrlKO}\}$  (or  $j = \{1, 2\}$ ) identifies the groups; and  $i = 1, \dots, n_j$  identifies the observations within each group

# From $t$ -test to linear regression

Let's change the notation to give a common framework to all methods

$$Y \sim G; E[Y] = \mu_Y$$



$$Y = \mu_Y + \varepsilon_Y; \varepsilon_Y \sim G; E[\varepsilon_Y] = 0$$

We can use indices to accommodate multiple groups, i.e.,

$$Y_{ij} = \mu_j + \varepsilon_{ij}; \varepsilon_{ij} \sim G_j; E[\varepsilon_{ij}] = 0;$$

where  $j = \{\text{WT, NrlKO}\}$  (or  $j = \{1, 2\}$ ) identifies the groups; and  $i = 1, \dots, n_j$  identifies the observations within each group

| For example:  $Y_{11}$  is the first observation in group 1 or WT

# This is called the cell-means model

The goal is to test  $H_0 : \mu_1 = \mu_2$

using data from the model

$$Y_{ij} = \mu_j + \varepsilon_{ij}; \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$

where  $j$  indexes groups (e.g. WT vs NrKO) and  $i$  indexes samples within group

- | We assume a common distribution  $G$  for all groups (equal variance assumption)

# This is called the cell-means model

The goal is to test  $H_0 : \mu_1 = \mu_2$

using data from the model

$$Y_{ij} = \mu_j + \varepsilon_{ij}; \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$

where  $j$  indexes groups (e.g. WT vs NrlKO) and  $i$  indexes samples within group

- | We assume a common distribution  $G$  for all groups (equal variance assumption)

**Note that the population means are given by  $E[Y_{ij}] = \mu_j$ , i.e., the model is written with a **cell-means** ( $\mu_j$ ) parametrization**

# This is called the cell-means model

The goal is to test  $H_0 : \mu_1 = \mu_2$

using data from the model

$$Y_{ij} = \mu_j + \varepsilon_{ij}; \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$

where  $j$  indexes groups (e.g. WT vs NrlKO) and  $i$  indexes samples within group

- | We assume a common distribution  $G$  for all groups (equal variance assumption)

**Note that the population means are given by  $E[Y_{ij}] = \mu_j$ , i.e., the model is written with a **cell-means** ( $\mu_j$ ) parametrization**

Why the name? 'Cell' here refers to a cell of a table - e.g. make a table of means by group, and  $\mu_j$  represents the population value for each cell  $j$  in the table

# Recall: sample mean estimator of population mean

Note that for each group, the **population** mean is given by

$$E[Y_{ij}] = \mu_j,$$

- A natural *estimator* of the population mean is the **sample** mean
- Classical hypothesis testing methods use the group sample means as estimators
- See, for example, the **t.test** function in R:

```
irs4.ttest$estimate
```

```
##      mean in group WT mean in group Nr1K0
##      7.765671        7.739612
```

# However, the `lm` function reports other estimates; why?

```
irs4.ttest$estimate
```

```
##      mean in group WT mean in group NrlKO  
##            7.765671          7.739612
```

```
irs4.lm$coefficients[,1]
```

```
## (Intercept) genotypeNrlKO  
##    7.76567142   -0.02605902
```

# However, the `lm` function reports other estimates; why?

```
irs4.ttest$estimate
```

```
##      mean in group WT mean in group NrlKO  
##            7.765671        7.739612
```

```
irs4.lm$coefficients[,1]
```

```
## (Intercept) genotypeNrlKO  
##    7.76567142   -0.02605902
```



(Intercept) estimate from `lm` is the **sample mean** of WT group

but `genotypeNrlKO` estimate from `lm` is **not** the sample mean of the NrlKO group. What is it then?

# Parameterization: how to write the model?

- By default, the `lm` function does not use the cell-means parameterization
- The goal is to *compare* the means, not to study each in isolation

Let's reformulate from **cell-means** ( $\mu_j$ ):

$$Y_{ij} = \mu_j + \varepsilon_{ij}; \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$



to **reference-treatment effect** ( $\theta, \tau_j$ ):

$$Y_{ij} = \theta + \tau_j + \varepsilon_{ij}; \quad \tau_1 = 0, \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$

# Parameterization: how to write the model?

- By default, the `lm` function does not use the cell-means parameterization
- The goal is to *compare* the means, not to study each in isolation

Let's reformulate from **cell-means** ( $\mu_j$ ):

$$Y_{ij} = \mu_j + \varepsilon_{ij}; \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$

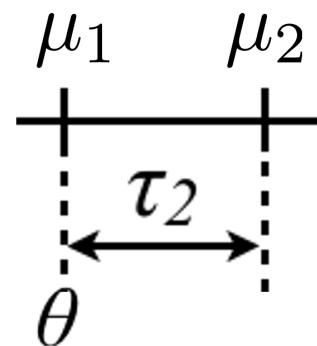
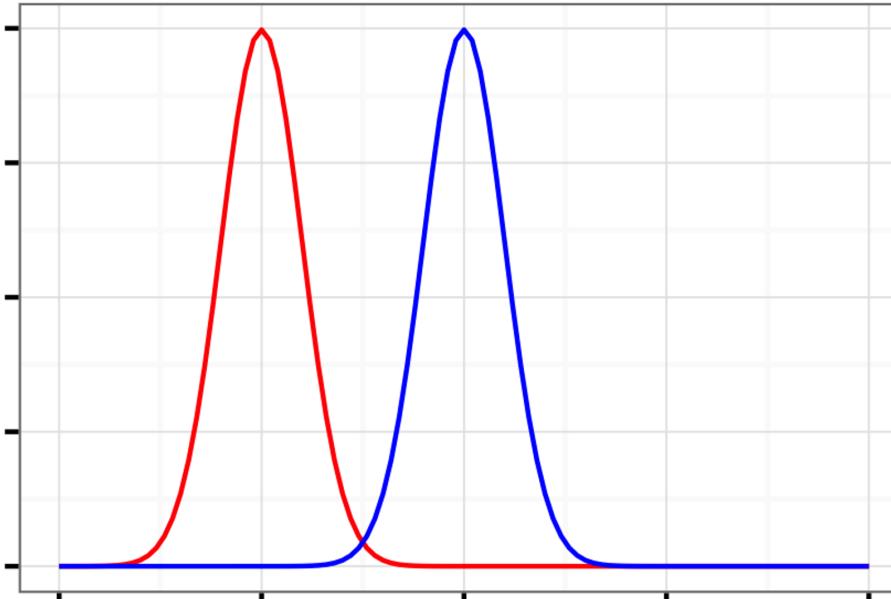


to **reference-treatment effect** ( $\theta, \tau_j$ ):

$$Y_{ij} = \theta + \tau_j + \varepsilon_{ij}; \quad \tau_1 = 0, \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$

- Note that for each group, the population mean is given by  $E[Y_{ij}] = \theta + \tau_j = \mu_j$ , and  $\tau_2 = \mu_2 - \mu_1 = E[Y_{i2}] - E[Y_{i1}]$  *compares* the means
- $\tau_1$  must be set to zero, since group 1 is the *reference* group

# Relation between parameterizations



$$H_0 : \mu_1 = \mu_2$$
$$H_0 : \tau_2 = 0$$

# lm output

- the sample mean of the **reference** group (WT):  $\hat{\theta}$
- **treatment effect**:  $\hat{\tau}_2$ 
  - i.e., difference between the sample means of both groups

For gene Irs4:

```
irs4.lm$coefficients[, 1]
```

```
## (Intercept) genotypeNr1KO
## 7.76567142 -0.02605902
```

```
irs4.means$meanExpr[irs4.means$genotype == "WT"]
```

```
## [1] 7.765671
```

```
irs4.means$meanExpr[irs4.means$genotype == "Nr1KO"] -
  irs4.means$meanExpr[irs4.means$genotype == "WT"]
```

```
## [1] -0.02605902
```

# lm output

- the sample mean of the **reference** group (WT):  $\hat{\theta}$
- **treatment effect**:  $\hat{\tau}_2$ 
  - i.e., difference between the sample means of both groups

For gene Nrl:

```
nrl.lm$coefficients[, 1]
```

```
## (Intercept) genotypeNrlKO  
## 11.244451 -5.154872
```

```
nrl.means$meanExpr[nrl.means$genotype == "WT"]
```

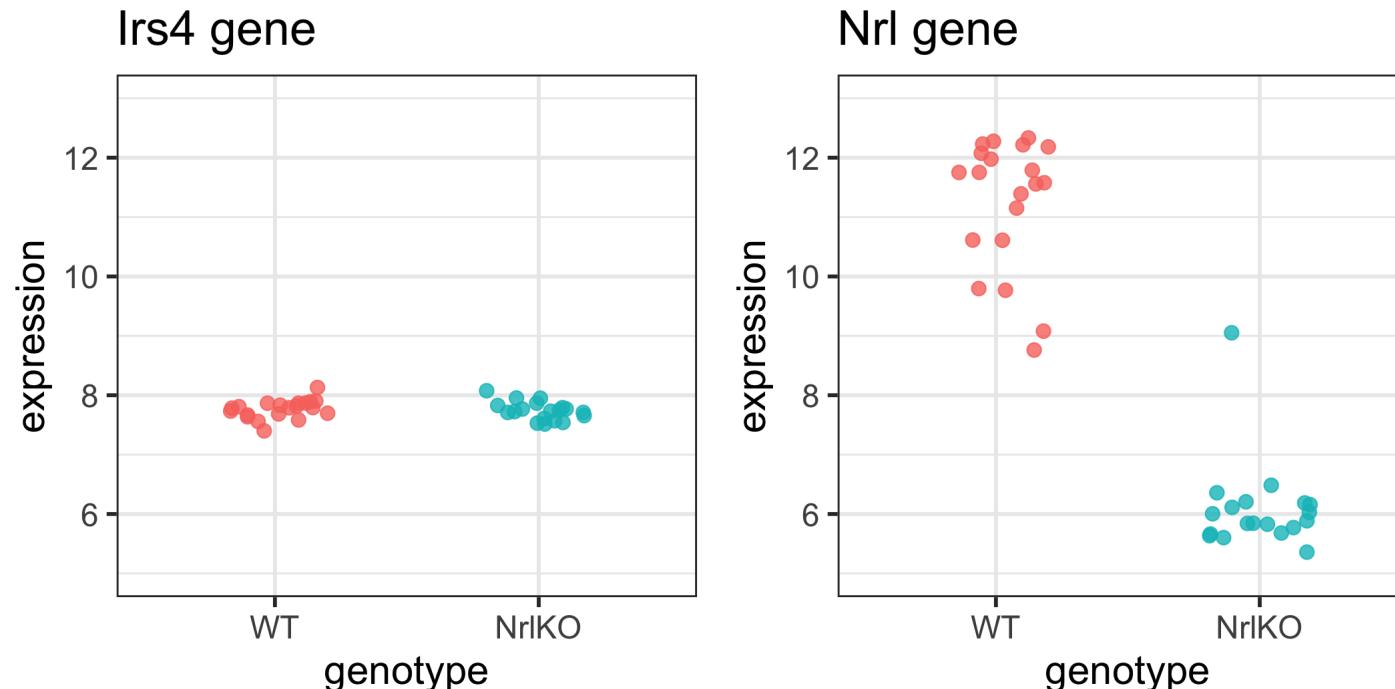
```
## [1] 11.24445
```

```
nrl.means$meanExpr[nrl.means$genotype == "NrlKO"] -  
nrl.means$meanExpr[nrl.means$genotype == "WT"]
```

```
## [1] -5.154872
```

# We still haven't answered our question ... where's the line??

$$Y_{ij} = \theta + \tau_j + \varepsilon_{ij}; \quad \tau_1 = 0, \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$



# Indicator variables

Let's re-write our model using **indicator** (aka 'dummy') variables:

$$Y_{ij} = \theta + \tau_j + \varepsilon_{ij} \text{ where } \tau_1 = 0; \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$



$$Y_{ij} = \theta + \tau_2 x_{ij} + \varepsilon_{ij} \text{ where } x_{ij} = \begin{cases} 1 & \text{if } j = 2 \\ 0 & \text{otherwise} \end{cases}$$

# Indicator variables

Let's re-write our model using **indicator** (aka 'dummy') variables:

$$Y_{ij} = \theta + \tau_j + \varepsilon_{ij} \text{ where } \tau_1 = 0; \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$



$$Y_{ij} = \theta + \tau_2 x_{ij} + \varepsilon_{ij} \text{ where } x_{ij} = \begin{cases} 1 & \text{if } j = 2 \\ 0 & \text{otherwise} \end{cases}$$

Note that  $Y_{i1} = \theta + \varepsilon_{i1}$ , because  $x_{i1} = 0$  and  $Y_{i2} = \theta + \tau_2 + \varepsilon_{i2}$ , because  $x_{i2} = 1$  (for all  $i$ )

The second form is written as a *linear* ( $y = a + bx + \varepsilon$ ) regression model, with a special **(indicator)** explanatory variable  $x_{ij}$

## Using indicator variables to model our categorical variable genotype we can perform a 2-sample *t*-test with a linear model

$$Y_{ij} = \theta + \tau_2 x_{ij} + \varepsilon_{ij} \text{ where } x_{ij} = \begin{cases} 1 & \text{if } j = 2 \\ 0 & \text{if } j = 1 \end{cases}$$

- Recall that  $\tau_2 = \mu_2 - \mu_1$
- The *t*-test in the linear model is carried out on  $H_0 : \tau_2 = 0$ , where  $\tau_2$  is the difference in population means (here NrlKO - WT)

```
list("t statistic"=irs4.ttest$stat,
     "p-value"=irs4.ttest$p.value)
```

```
## $`t statistic`
##      t
## 0.5285386
##
## $`p-value`
## [1] 0.6002819
```

```
list("t statistic"=irs4.lm$coeff[2,3],
     "p-value"=irs4.lm$coeff[2,4])
```

```
## $`t statistic`
## [1] -0.5285386
##
## $`p-value`
## [1] 0.6002819
```

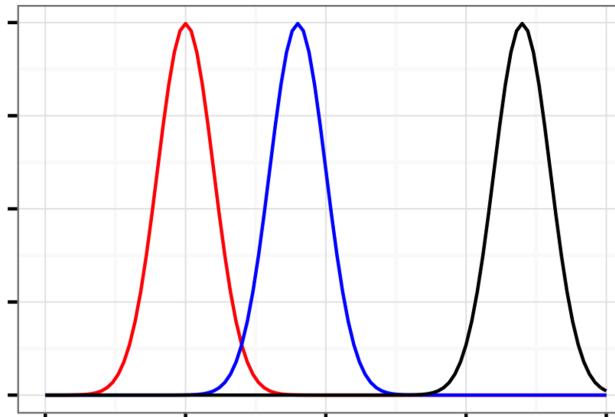
# Beyond 2-group comparisons

“cell-means”

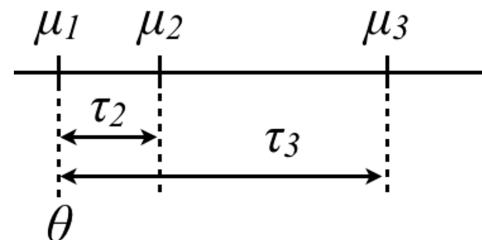
$$Y_{ij} = \mu_j + \varepsilon_{ij}$$

“reference-treatments”

$$Y_{ij} = \theta + \tau_j + \varepsilon_{ij}, (\tau_1 = 0)$$



More than 2 groups!



**Indicator variables can be used to model one *or more* categorical variables, each with 2 *or more* levels!**

2-sample *t*-test using a linear model

$$Y_{ij} = \theta + \tau_2 x_{ij} + \varepsilon_{ij} \text{ where } x_{ij} = \begin{cases} 1 & \text{if } j = 2 \\ 0 & \text{if } j = 1 \end{cases}$$

1-way ANOVA with many levels \* using a linear model - e.g for 3 groups:

$$Y_{ij} = \theta + \tau_2 x_{ij2} + \tau_3 x_{ij3} + \varepsilon_{ij} \text{ where } x_{ij2} = \begin{cases} 1 & \text{if } j = 2 \\ 0 & \text{otherwise} \end{cases} \text{ and } x_{ij3} = \begin{cases} 1 & \text{if } j = 3 \\ 0 & \text{otherwise} \end{cases}$$

This is why R can estimate all of them with `lm()`

\* in general; yet *another* parameterization can be used to present ANOVA

## t-test

Special case of ANOVA, but with ANOVA you can compare **more than two groups** and **more than one factor**.

## ANOVA

Special case of linear regression, but with linear regression you can include **quantitative variables** in the model.

## Linear regression

Provides a unifying framework to model the association between a response and **many quantitative and qualitative variables**.

In R: all three can be computed using the `lm()` function.

# Linear models using matrix notation

$$Y = X\alpha + \varepsilon$$

the column vector of the responses  
one element per experimental unit

a column vector  
of the errors

a (design) matrix that represents covariate  
info, one row per experimental unit

a column vector of the parameters in the  
linear model

It will become handy to write our model using matrix notation

# Let's form a design matrix ( $X$ ) for a 3-group comparison

$$Y_{ij} = \theta + \tau_2 x_{ij2} + \tau_3 x_{ij3} + \varepsilon_{ij}$$

First column in  $X$  for reference treatment parameterization is all 1s

Second & third columns contain  $x_{ij2}$  and  $x_{ij3}$ :

- $x_{i12} = x_{i13} = 0$  for the reference group
- $x_{i22} = 1$  for the 2nd group
- $x_{i33} = 1$  for the 3rd group

$$\begin{bmatrix} Y_{11} \\ \vdots \\ Y_{n_11} \\ Y_{12} \\ \vdots \\ Y_{n_22} \\ Y_{13} \\ \vdots \\ Y_{n_33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ \vdots & \vdots & \vdots \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta \\ \tau_2 \\ \tau_3 \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} \\ \vdots \\ \varepsilon_{n_11} \\ \varepsilon_{12} \\ \vdots \\ \varepsilon_{n_22} \\ \varepsilon_{13} \\ \vdots \\ \varepsilon_{n_33} \end{bmatrix}$$

Diagram illustrating the components of the regression equation:

- response  $Y$** : Points to the leftmost column of the matrix  $\mathbf{Y}$ .
- design matrix  $X$** : Points to the middle matrix  $\mathbf{X}$ .
- regression parameters**: Points to the vector  $\boldsymbol{\alpha}$ .
- error term**: Points to the rightmost column of the matrix  $\mathbf{\varepsilon}$ .

$$Y = X\alpha + \varepsilon$$

$$\begin{bmatrix} \underline{Y_{11}} \\ \vdots \\ Y_{n_1 1} \\ \underline{Y_{12}} \\ \vdots \\ Y_{n_2 2} \\ \underline{Y_{13}} \\ \vdots \\ Y_{n_3 3} \end{bmatrix} = \begin{bmatrix} \underline{1} & \underline{0} & \underline{0} \\ \vdots & \vdots & \vdots \\ 1 & 0 & 0 \\ \underline{1} & \underline{1} & \underline{0} \\ \vdots & \vdots & \vdots \\ 1 & 1 & 0 \\ \underline{1} & \underline{0} & \underline{1} \\ \vdots & \vdots & \vdots \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta \\ \tau_2 \\ \tau_3 \end{bmatrix} + \begin{bmatrix} \underline{\varepsilon_{11}} \\ \vdots \\ \varepsilon_{n_1 1} \\ \underline{\varepsilon_{12}} \\ \vdots \\ \varepsilon_{n_2 2} \\ \underline{\varepsilon_{13}} \\ \vdots \\ \varepsilon_{n_3 3} \end{bmatrix}$$

$$Y_{i1} = 1 \times \theta + 0 \times \tau_2 + 0 \times \tau_3 + \varepsilon_{i1} = \theta + \varepsilon_{i1}$$

$$Y_{i2} = 1 \times \theta + 1 \times \tau_2 + 0 \times \tau_3 + \varepsilon_{i2} = \theta + \tau_2 + \varepsilon_{i2}$$

$$Y_{i3} = 1 \times \theta + 0 \times \tau_2 + 1 \times \tau_3 + \varepsilon_{i3} = \theta + \tau_3 + \varepsilon_{i3}$$

$$Y_{ij} = \theta + \tau_2 x_{ij2} + \tau_3 x_{ij3} + \varepsilon_{ij}$$

$$Y = X\alpha + \varepsilon$$

$$\begin{bmatrix} Y_{11} \\ Y_{21} \\ \vdots \\ Y_{n_33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ \vdots & \vdots & \vdots \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta \\ \tau_2 \\ \tau_3 \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{21} \\ \vdots \\ \varepsilon_{n_33} \end{bmatrix}$$

Reference group:  $\mu_1$

$\mu_2 - \mu_1$

$\mu_3 - \mu_1$

The model is still written with a reference-treatment parameterization (difference of means)

$$E[Y_{i1}] = \theta$$

$$E[Y_{i2}] = \theta + \tau_2 \rightarrow \tau_2 = E[Y_{i2}] - E[Y_{i1}] = \mu_2 - \mu_1$$

$$E[Y_{i3}] = \theta + \tau_3 \rightarrow \tau_3 = E[Y_{i3}] - E[Y_{i1}] = \mu_3 - \mu_1$$

# Linear regression can include *quantitative & qualitative* covariates

$$Y = \boxed{X}\alpha + \varepsilon$$

This gives us a VERY FLEXIBLE framework!!

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 1.22 \\ 1 & 2.02 \\ 1 & 1.42 \\ \vdots & \vdots \\ 1 & 1.89 \\ 1 & 2.01 \\ \vdots & \vdots \\ 1 & 1.56 \\ 1 & 2.17 \\ 1 & 1.51 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 1.22 & 0 \\ 1 & 0 & 2.02 & 0 \\ 1 & 0 & 1.42 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1.89 & 0 \\ 1 & 1 & 2.01 & 2.01 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1.56 & 1.56 \\ 1 & 1 & 2.17 & 2.17 \\ 1 & 1 & 1.51 & 1.51 \end{bmatrix}$$

1 categorical covariate

2 categorical covariates

1 continuous covariate

1 continuous  
1 categorical

AND MANY MORE .....

Tip: ?model.matrix

Here we mean **linear** in the parameters  $\alpha$ ;  $X$  can contain  $x^2$ ,  $\log(x)$ , etc

# How it works in practice using lm() in R

$$Y = X\alpha + \varepsilon$$



```
lm(y ~ x, data = yourData)
```

`y ~ x`: formula

`y`: numeric

`x`: numeric and/or factor

`yourData`: `data.frame` (or `tibble`) in  
which `x` and `y` are to be found

By default, R uses the reference-treatment parametrization but you can control that!

# Special factor class in R

$$Y = X\alpha + \varepsilon$$

- Mathematically,  $X$  is a numeric matrix
- If your data contains categorical variables (e.g., `genotype`), you need to set them as **factors**
  - especially important if your categorical variables are encoded numerically!!
  - `lm` will automatically treat character variables as factors)
- R creates appropriate indicator variables for factors!

```
str(twoGenes$genotype)
```

```
## Factor w/ 2 levels "WT", "Nr1KO": 2 2 2 2 2 2 2 2 2 ...
```

# Under the hood, R creates a numeric $X$

```
mm <- model.matrix(~genotype, data =  
twoGenes)  
# show first 3 and last 3 rows of  
# model.matrix  
head(mm, 3)
```

```
## (Intercept) genotypeNr1KO  
## 1 1 1  
## 2 1 1  
## 3 1 1
```

```
tail(mm, 3)
```

```
## (Intercept) genotypeNr1KO  
## 76 1 0  
## 77 1 0  
## 78 1 0
```

```
# show first 3 and last 3 values of genotype  
twoGenes %>%  
  slice(c(1:3, (n() - 3):n())) %>%  
  pull(genotype)
```

```
## [1] Nr1KO Nr1KO Nr1KO WT     WT     WT     WT  
## Levels: WT Nr1KO
```

## Beyond 2-group comparisons in our case study:

Is the expression of gene X the same at all developmental stages?

## Beyond 2-group comparisons in our case study:

Is the expression of gene X the same at all developmental stages?

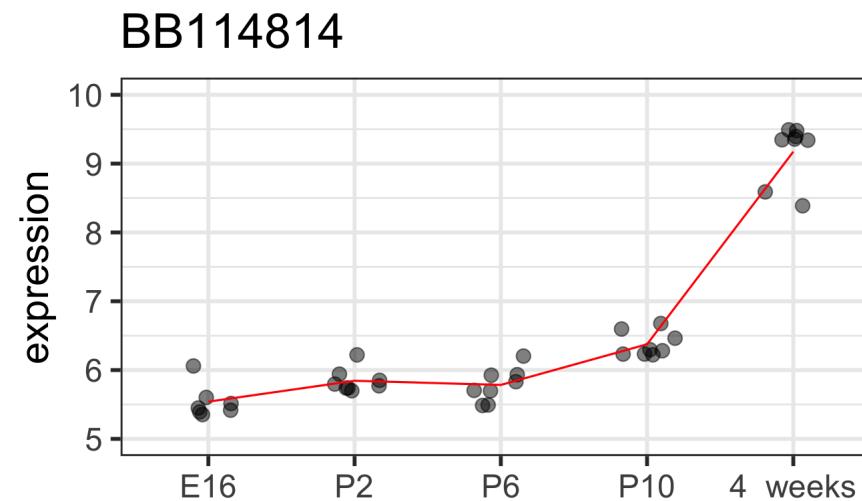
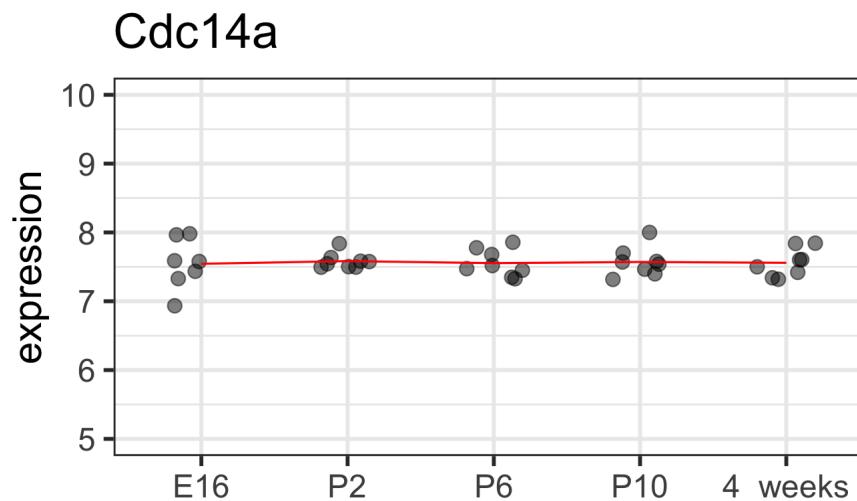
$$H_0 : \mu_{E16} = \mu_{P2} = \mu_{P6} = \mu_{P10} = \mu_{4W}$$

# Beyond 2-group comparisons in our case study:

Is the expression of gene X the same at all developmental stages?

$$H_0 : \mu_{E16} = \mu_{P2} = \mu_{P6} = \mu_{P10} = \mu_{4W}$$

Let's look at another two genes for some variety



Note: 4W = 4\_weeks

## The sample means: $\hat{\mu}_{E16}$ , $\hat{\mu}_{P2}$ , $\hat{\mu}_{P6}$ , $\hat{\mu}_{P10}$ , $\hat{\mu}_{4W}$

```
twoGenes %>%
  group_by(gene, dev_stage) %>%
  summarize(meanExpr = mean(expression)) %>%
  pivot_wider(values_from = meanExpr,
  names_from = gene)
```

```
## # A tibble: 5 × 3
##   dev_stage BB114814 Cdc14a
##   <fct>      <dbl>  <dbl>
## 1 E16        5.5409 7.5443
## 2 P2         5.8447 7.5836
## 3 P6         5.7842 7.5540
## 4 P10        6.3750 7.5710
## 5 4_weeks    9.1733 7.5590
```

# BB114814 gene with notable time effect

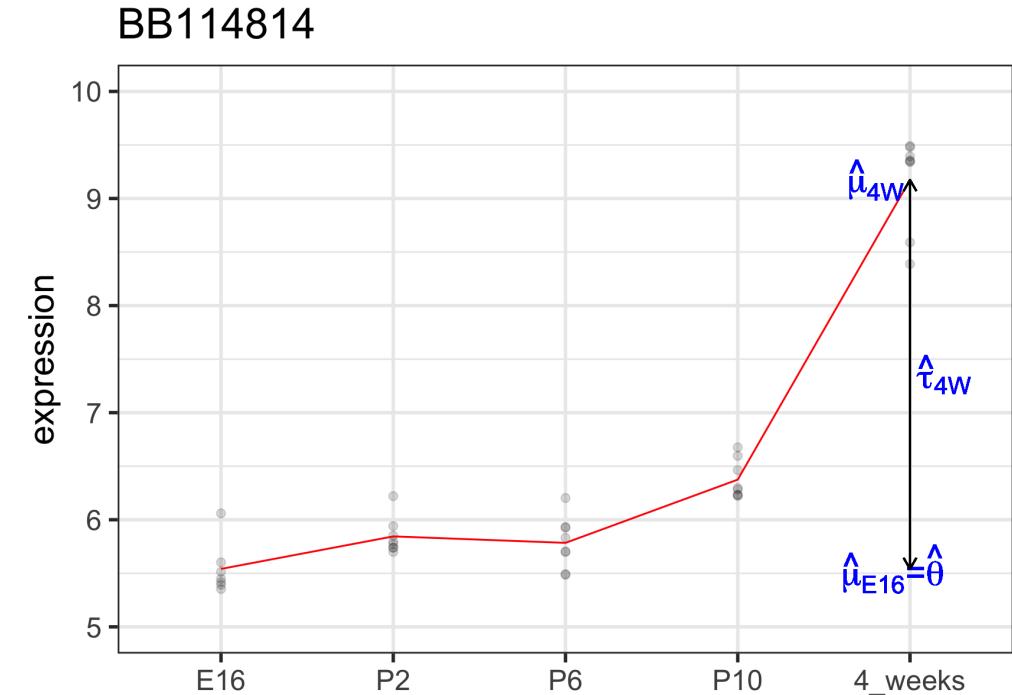
```
twoGenes %>% filter(gene == "BB114814") %>%
  group_by(dev_stage) %>%
  summarize(cellMeans = mean(expression)) %>%
  mutate(timeEffect = cellMeans - cellMeans[1])
```

```
## # A tibble: 5 × 3
##   dev_stage    cellMeans  timeEffect
##   <fct>        <dbl>      <dbl>
## 1 E16          5.5409      0
## 2 P2           5.8447     0.30379
## 3 P6           5.7842     0.24328
## 4 P10          6.3750     0.83412
## 5 4_weeks      9.1733     3.6324
```

"Effect" here means compared to reference/baseline (E16)

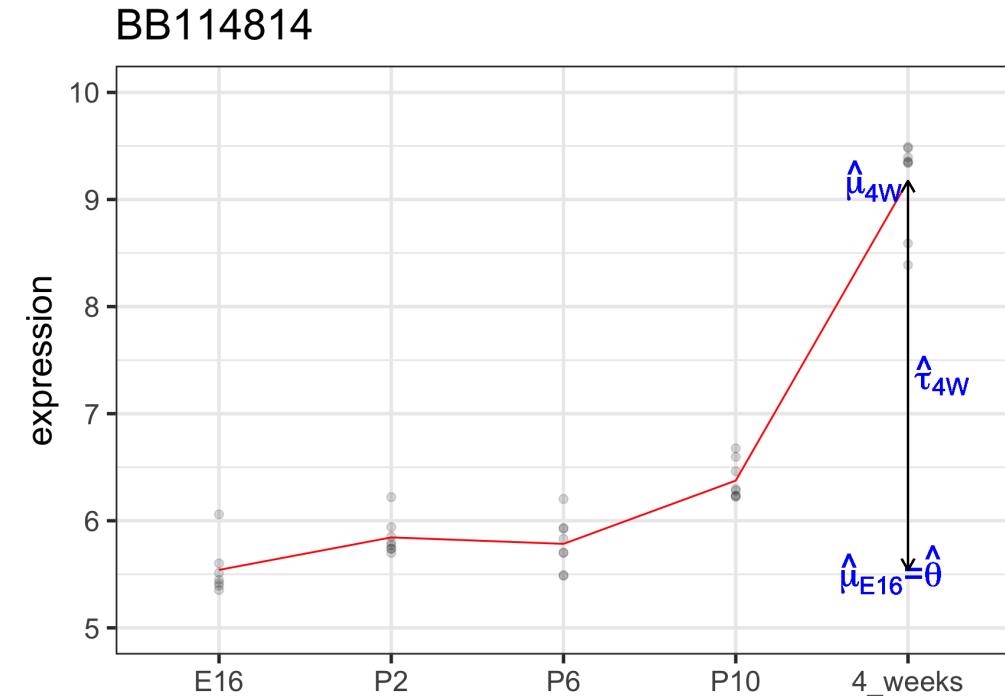
# BB114814 gene with notable time effect

```
## # A tibble: 5 × 3
##   dev_stage cellMeans timeEffect
##   <fct>      <dbl>     <dbl>
## 1 E16        5.5409     0
## 2 P2         5.8447    0.30379
## 3 P6         5.7842    0.24328
## 4 P10        6.3750    0.83412
## 5 4_weeks    9.1733    3.6324
```



# BB114814 gene with notable time effect

```
## # A tibble: 5 × 3
##   dev_stage cellMeans timeEffect
##   <fct>      <dbl>     <dbl>
## 1 E16        5.5409    0
## 2 P2         5.8447    0.30379
## 3 P6         5.7842    0.24328
## 4 P10        6.3750    0.83412
## 5 4_weeks    9.1733    3.6324
```



Can you guess the size of the  $X$  matrix?

| How many indicator variables do we need?

## Gene BB114814 with notable time effect

We need 4 indicator variables to estimate and test 4 time differences (between 5 time points):

$x_{P2}$ : P2 vs E16

$x_{P6}$ : P6 vs E16

$x_{P10}$ : P10 vs E16

$x_{4W}$ : 4W vs E16

## Gene BB114814 with notable time effect

We need 4 indicator variables to estimate and test 4 time differences (between 5 time points):

$x_{P2}$ : P2 vs E16

$x_{P6}$ : P6 vs E16

$x_{P10}$ : P10 vs E16

$x_{4W}$ : 4W vs E16

$$Y_{ij} = \theta + \tau_{P2}x_{ijP2} + \tau_{P6}x_{ijP6} + \tau_{P10}x_{ijP10} + \tau_{4W}x_{ij4W} + \varepsilon_{ij}$$

**Notation:**  $x_{ijk}$ :

- $i$  indexes for the observation/sample within group
- $j$  indexes the group (here: level of `dev_stage`)
- $k$  is the name of the indicator variable

# Under the hood, R creates a numeric $X$

```
model.matrix(~dev_stage, data = twoGenes) %>%  
  head(19)
```

```
##      (Intercept) dev_stageP2 dev_stageP6 dev_stageP10 dev_stage4_weeks  
## 1            1         0         0         0             1  
## 2            1         0         0         0             1  
## 3            1         0         0         0             1  
## 4            1         0         0         0             1  
## 5            1         0         0         0             0  
## 6            1         0         0         0             0  
## 7            1         0         0         0             0  
## 8            1         0         0         1             0  
## 9            1         0         0         1             0  
## 10           1         0         0         1             0  
## 11           1         0         0         1             0  
## 12           1         1         0         0             0  
## 13           1         1         0         0             0  
## 14           1         1         0         0             0  
## 15           1         1         0         0             0  
## 16           1         0         1         0             0  
## 17           1         0         1         0             0  
## 18           1         0         1         0             0  
## 19           1         0         1         0             0
```

## Hypothesis tests in lm output

```
## dev_stage cellMeans timeEffect
## 1 E16      5.5409    0
## 2 P2       5.8447   0.30379
## 3 P6       5.7842   0.24328
## 4 P10      6.3750   0.83412
## 5 4_weeks  9.1733   3.6324
```

```
twoGenes %>% filter(gene == "BB114814") %>%
  lm(expression ~ dev_stage, data = .) %>%
  summary() %>% $.coef
```

```
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.5409162 0.1021560 54.239748 1.314828e-34
## dev_stageP2 0.3037855 0.1398829  2.171713 3.694652e-02
## dev_stageP6 0.2432795 0.1398829  1.739166 9.105366e-02
## dev_stageP10 0.8341163 0.1398829  5.962962 9.620151e-07
## dev_stage4_weeks 3.6323772 0.1398829 25.967276 5.303201e-24
```

$$H_0 : \theta = 0 \text{ or } H_0 : \mu_{E16} = 0$$

**Estimate:**  $\hat{\theta} = \hat{\mu}_{E16} = \bar{Y}_{\cdot E16}$

we are not usually interested in testing this hypothesis: baseline mean = 0

## Hypothesis tests in lm output

```
## dev_stage cellMeans timeEffect
## 1 E16      5.5409    0
## 2 P2       5.8447   0.30379
## 3 P6       5.7842   0.24328
## 4 P10      6.3750   0.83412
## 5 4_weeks  9.1733   3.6324
```

```
twoGenes %>% filter(gene == "BB114814") %>%
  lm(expression ~ dev_stage, data = .) %>%
  summary() %>% $.coef
```

```
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.5409162  0.1021560 54.239748 1.314828e-34
## dev_stageP2      0.3037855  0.1398829  2.171713 3.694652e-02
## dev_stageP6      0.2432795  0.1398829  1.739166 9.105366e-02
## dev_stageP10     0.8341163  0.1398829  5.962962 9.620151e-07
## dev_stage4_weeks 3.6323772  0.1398829 25.967276 5.303201e-24
```

$$H_0 : \tau_{P2} = 0 \text{ or } H_0 : \mu_{P2} = \mu_{E16}$$

$$\text{Estimate: } \hat{\tau}_{P2} = \hat{\mu}_{P2} - \hat{\mu}_{E16} = \bar{Y}_{.P2} - \bar{Y}_{.E16}$$

we are usually interested in testing this hypothesis: change from E16 to 2 days old = 0

## Hypothesis tests in lm output

```
## dev_stage cellMeans timeEffect
## 1 E16      5.5409    0
## 2 P2       5.8447   0.30379
## 3 P6       5.7842   0.24328
## 4 P10      6.3750   0.83412
## 5 4_weeks  9.1733   3.6324
```

```
twoGenes %>% filter(gene == "BB114814") %>%
  lm(expression ~ dev_stage, data = .) %>%
  summary() %>% $.coef
```

```
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.5409162  0.1021560 54.239748 1.314828e-34
## dev_stageP2      0.3037855  0.1398829  2.171713 3.694652e-02
## dev_stageP6      0.2432795  0.1398829  1.739166 9.105366e-02
## dev_stageP10     0.8341163  0.1398829  5.962962 9.620151e-07
## dev_stage4_weeks 3.6323772  0.1398829 25.967276 5.303201e-24
```

$$H_0 : \tau_{4W} = 0 \text{ or } H_0 : \mu_{4W} = \mu_{E16}$$

**Estimate:**

$$\hat{\tau}_{4W} = \hat{\mu}_{4W} - \hat{\mu}_{E16} = \bar{Y}_{.4W} - \bar{Y}_{.E16}$$

we are usually interested in testing this hypothesis: change from E16 to 4 weeks old = 0

# Notice the standard error estimates

```
##             Estimate Std. Error   t value   Pr(>|t|)  
## (Intercept) 5.5409162  0.1021560 54.239748 1.314828e-34  
## dev_stageP2  0.3037855  0.1398829  2.171713 3.694652e-02  
## dev_stageP6  0.2432795  0.1398829  1.739166 9.105366e-02  
## dev_stageP10 0.8341163  0.1398829  5.962962 9.620151e-07  
## dev_stage4_weeks 3.6323772  0.1398829 25.967276 5.303201e-24
```

All data points are used to estimate the variance of the error term for the indicator variables

# Two types of null hypotheses: single vs joint

$$Y = X\alpha + \varepsilon$$

$$\alpha = (\theta, \tau_{P2}, \tau_{P6}, \tau_{P10}, \tau_{4W})$$

$$H_0 : \tau_j = 0 \text{ vs } H_0 : \tau_j \neq 0$$

for each  $j$  individually

For example: Is gene A differentially expressed 2 days after birth (compared to embryonic day 16)?

$$H_0 : \tau_{P2} = 0$$

This can be tested with a **t-test**

$$H_0 : \tau_j = 0 \text{ vs } H_0 : \tau_j \neq 0$$

for all  $j$  at the same time

For example: Is gene A significantly affected by time? In other words, is gene A differentially expressed at *any* time point?

$$H_0 : \tau_{P2} = \tau_{P6} = \tau_{P10} = \tau_{4W} = 0$$

How do we test this null hypothesis??

# F-test and overall significance of one or more coefficients

- the  $t$ -test in linear regression allows us to test single hypotheses:

$$H_0 : \tau_j = 0$$

$$H_A : \tau_j \neq 0$$

- but we often like to test multiple hypotheses *simultaneously*:

$$H_0 : \tau_{P2} = \tau_{P6} = \tau_{P10} = \tau_{4W} = 0 \text{ [AND statement]}$$

$$H_A : \tau_j \neq 0 \text{ for some } j \text{ [OR statement]}$$

- the **Ftest** allows us to test such compound tests
  - more on this type of test next week

# $H_0 : \tau_j = 0$ vs $H_0 : \tau_j \neq 0$ for each $j$ individually

```
##  
## Call:  
## lm(formula = expression ~ dev_stage, data = .)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -0.78553 -0.13324 -0.04796  0.17038  0.51846  
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 5.5409    0.1022  54.240 < 2e-16 ***  
## dev_stageP2 0.3038    0.1399   2.172  0.0369 *  
## dev_stageP6 0.2433    0.1399   1.739  0.0911 .  
## dev_stageP10 0.8341    0.1399   5.963 9.62e-07 ***  
## dev_stage4_weeks 3.6324    0.1399  25.967 < 2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 0.2703 on 34 degrees of freedom  
## Multiple R-squared:  0.9662,    Adjusted R-squared:  0.9623  
## F-statistic: 243.3 on 4 and 34 DF,  p-value: < 2.2e-16
```

# $H_0 : \tau_j = 0$ vs $H_0 : \tau_j \neq 0$ for all $j$ together

```
##  
## Call:  
## lm(formula = expression ~ dev_stage, data = .)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -0.78553 -0.13324 -0.04796  0.17038  0.51846  
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 5.5409     0.1022  54.240 < 2e-16 ***  
## dev_stageP2 0.3038     0.1399   2.172  0.0369 *  
## dev_stageP6 0.2433     0.1399   1.739  0.0911 .  
## dev_stageP10 0.8341     0.1399   5.963 9.62e-07 ***  
## dev_stage4_weeks 3.6324     0.1399  25.967 < 2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 0.2703 on 34 degrees of freedom  
## Multiple R-squared:  0.9662,    Adjusted R-squared:  0.9623  
## F-statistic: 243.3 on 4 and 34 DF,  p-value: < 2.2e-16
```

# To conclude

1. We can use different parametrizations to write statistical models

- **cell-means** ( $\mu_j$ ):  $Y_{ij} = \mu_j + \varepsilon_{ij}$  where  $\varepsilon_{ij} \sim G$ ;  $E[\varepsilon_{ij}] = 0$ ;
- **reference-treatment effect** ( $\theta, \tau_j$ ): (used by default by `lm`)

$$Y_{ij} = \theta + \tau_j + \varepsilon_{ij} \text{ where } \tau_1 = 0, \quad \varepsilon_{ij} \sim G; \quad E[\varepsilon_{ij}] = 0;$$

2. We can compare group means (2 or more) using a linear model

- **indicator variables** (e.g.,  $x_{ijP2}$ ) to model the levels of a qualitative explanatory variables

$$Y_{ij} = \theta + \tau_{P2}x_{ijP2} + \tau_{P6}x_{ijP6} + \tau_{P10}x_{ijP10} + \tau_{4W}x_{ij4W} + \varepsilon_{ij}$$

- qualitative variables need to be set as "factors" in the data → R creates the indicator variables

3. We can write a **linear model** using matrix notation:

$$Y = X\alpha + \varepsilon$$

4. Linear models can include **quantitative & qualitative covariates**

$$Y = X\alpha + \varepsilon$$

This gives us a VERY FLEXIBLE framework!!

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1.22 \\ 1 & 2.02 \\ 1 & 1.42 \\ \vdots & \vdots \\ 1 & 1.89 \\ 1 & 2.01 \\ \vdots & \vdots \\ 1 & 1.56 \\ 1 & 2.17 \\ 1 & 1.51 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1.22 & 0 \\ 1 & 0 & 2.02 & 0 \\ 1 & 0 & 1.42 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1.89 & 0 \\ 1 & 1 & 2.01 & 2.01 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1.56 & 1.56 \\ 1 & 1 & 2.17 & 2.17 \\ 1 & 1 & 1.51 & 1.51 \end{bmatrix}$$

1 categorical  
covariate

2 categorical  
covariates

1 continuous  
covariate

1 continuous  
1 categorical

AND MANY MORE .....

Tip: ?model.matrix

5. We use different tests to distinguish between **single** and **joint** hypotheses:

- e.g.  $t$ -tests vs  $F$ -tests