Statistical Methods for High Dimensional Biology

Linear models with multiple factors

Keegan Korthauer

1 February 2021

with slide contributions from Gabriela Cohen Freue and Jenny Bryan

Recall from last class...

1. How to compare means of different groups (2 or more) using a linear regression model

- dummy/indicator variables to model the levels of a qualitative explanatory variable

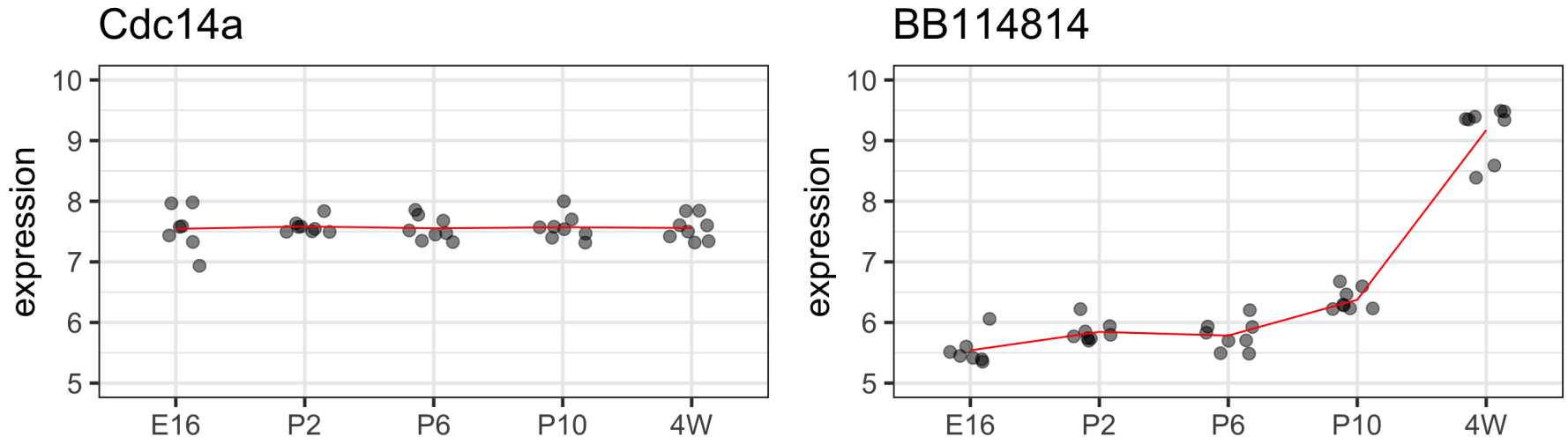
2. Write a linear model using matrix notation

- understand which matrix is built by R

3. Distinguish between **single** and **joint** hypotheses

- t-tests vs F-tests

Do we think that the expression levels at different developmental stages are generated by distributions with different location (mean)? Or a single common distribution?



Quick review: from t-test to linear regression

2-sample t-test

$$Y \sim F;\ E[Y] = \mu_Y;\ Z \sim G;\ E[Z] = \mu_Z$$

$$H_0 : \mu_Y = \mu_Z$$

$$\downarrow ?$$

Linear regression

$$Y = X\alpha + \epsilon; \qquad H_0 : \alpha_j = 0$$

HOW? WHY?

HOW?? : Cell means model using dummy variables

$$Y \sim F;\ E[Y] = \mu_Y;\ Z \sim G;\ E[Z] = \mu_Z$$

$$\downarrow$$

$$Y_{ij} = \mu_1 x_{ij1} + \mu_2 x_{ij2} + \varepsilon_{ij};\ i = 1, \ldots, n;\ j = 1, 2$$

$$x_{ij1} = \begin{cases} 1 \text{ if } j = 1 \\ 0 \text{ otherwise} \end{cases}, \quad x_{ij2} = \begin{cases} 1 \text{ if } j = 2 \\ 0 \text{ otherwise} \end{cases}$$

$$\downarrow$$

$$\begin{aligned} E[Y_{i1}] &= \mu_1 \\ E[Y_{i2}] &= \mu_2 \end{aligned}$$

$$\begin{bmatrix} Y_{11} \\ \vdots \\ Y_{n_1 1} \\ Y_{12} \\ \vdots \\ Y_{n_2 2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} \\ \vdots \\ \varepsilon_{n_1 1} \\ \varepsilon_{12} \\ \vdots \\ \varepsilon_{n_2 2} \end{bmatrix}$$

HOW?? : Changing the parameterization to reference-treatment using dummy variables

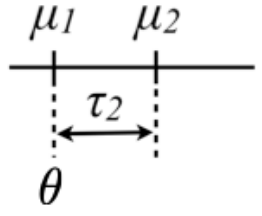
$$Y \sim F;\ E[Y] = \mu_Y;\ Z \sim G;\ E[Z] = \mu_Z$$

$$\downarrow$$

$$Y_{ij} = \theta + \tau_1 x_{ij1} + \tau_2 x_{ij2} + \varepsilon_{ij};\ i = 1, \ldots, n;\ j = 1, 2; \tau_1 = 0$$

$$x_{ij1} = \begin{cases} 1 \text{ if } j = 1 \\ 0 \text{ otherwise} \end{cases}, \quad x_{ij2} = \begin{cases} 1 \text{ if } j = 2 \\ 0 \text{ otherwise} \end{cases}$$

$$\downarrow$$



$$\begin{aligned} E[Y_{i1}] &= \theta = \mu_1 \\ E[Y_{i2}] &= \theta + \tau_2 = \mu_1 + (\mu_2 - \mu_1) = \mu_2 \end{aligned}$$

HOW?? : Changing the parameterization to reference-treatment using dummy variables

Removing the $\tau_1 x_{ij1}$ term since $\tau_1 = 0$:

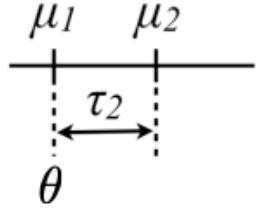
$$Y \sim F;\ E[Y] = \mu_Y;\ Z \sim G;\ E[Z] = \mu_Z$$

$$\downarrow$$

$$Y_{ij} = \theta + \tau_2 x_{ij2} + \varepsilon_{ij};\ i = 1, \ldots, n;\ j = 1, 2$$

$$x_{ij2} = \begin{cases} 1 \text{ if } j = 2 \\ 0 \text{ otherwise} \end{cases}$$

$$\downarrow$$



$$\begin{aligned} E[Y_{i1}] &= \theta = \mu_1 \\ E[Y_{i2}] &= \theta + \tau_2 = \mu_1 + (\mu_2 - \mu_1) = \mu_2 \end{aligned}$$

Using matrix notation ...

$$Y_{ij} = \theta + \tau_2 x_{ij2} + \varepsilon_{ij} \Rightarrow \mathbf{Y} = \mathbf{X}\alpha + \epsilon$$

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

- x_{ij2} is the second column of design matrix X
- $x_{112} = 0$ and $x_{122} = 1$

Red

$$Y_{11} = 1 * \theta + 0 * \tau_2 + \epsilon_{11} = \theta + \epsilon_{11}$$

Blue

$$Y_{12} = 1 * \theta + 1 * \tau_2 + \epsilon_{12} = \theta + \tau_2 + \epsilon_{12}$$

- Tip: examine design matrix in R with `model.matrix()`

... and similarly beyond 2 group comparisons (ANOVA)

WHY??

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

This gives us a VERY FLEXIBLE framework!!

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

1 categorical covariate

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ \vdots & \vdots & \vdots & \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

2 categorical covariates

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

1 continuous covariate

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

1 continuous 1 categorical

AND MANY MORE .....

Tip: ?model.matrix

Parameterizations

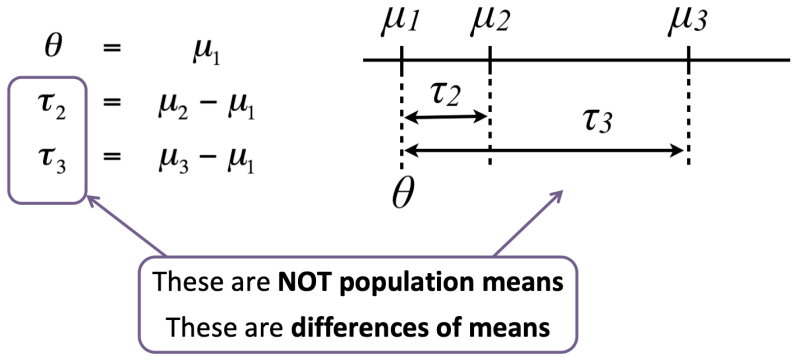
Different ways of writing this [design matrix, parameter vector] pair correspond to different **parameterizations** of the model

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

Understanding these concepts makes it easier ...

- to interpret and compare fitted models
- to fit models such that comparisons you care most about are directly addressed in the inferential "report"

Example: comparisons of mean expression between groups



By default, `lm` estimates group mean differences (with respect to a reference group):

```
filter(twoGenes, 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_stage4W  3.6323772  0.1398829 25.967276 5.303201e-24
```

We can tell R to use the cell-means parameterization

Write the formula as `Y ~ 0 + x` in the `lm` call to remove the intercept (θ) parameter and fit cell means parameters instead.

```
filter(twoGenes, gene == "BB114814") %>%
  lm(expression ~ 0 + dev_stage, data = .) %>%
  summary() %>% .$coef
```

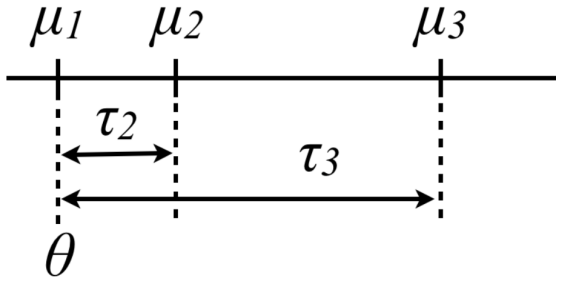
```
##                Estimate Std. Error  t value     Pr(>|t|)
## dev_stageE16 5.540916  0.1021560 54.23975 1.314828e-34
## dev_stageP2  5.844702  0.0955582 61.16379 2.303551e-36
## dev_stageP6  5.784196  0.0955582 60.53061 3.271123e-36
## dev_stageP10 6.375032  0.0955582 66.71361 1.230927e-37
## dev_stage4W  9.173293  0.0955582 95.99693 5.558604e-43
```

What null hypothesis does the *t*-test column now represent?

H_0: Each group mean is equal to zero

Recall that we can obtain one set of parameters from the other!

$$\mu_1 = \theta \qquad \theta = \mu_1$$
$$\mu_2 = \theta + \tau_2 \qquad \tau_2 = \mu_2 - \mu_1$$
$$\mu_3 = \theta + \tau_3 \qquad \tau_3 = \mu_3 - \mu_1$$



These are **population** means

These are **NOT population** means

These are **ref & TX** effects

```
filter(twoGenes, gene == "BB114814") %>%
  lm(expression ~ 0 + dev_stage, data = .) %>%
  summary() %>% .$coef
```

```
##              Estimate Std. Error  t value     Pr(>|t|)
## dev_stageE16 5.540916  0.1021560 54.23975 1.314828e-34
## dev_stageP2  5.844702  0.0955582 61.16379 2.303551e-36
## dev_stageP6  5.784196  0.0955582 60.53061 3.271123e-36
## dev_stageP10 6.375032  0.0955582 66.71361 1.230927e-37
## dev_stage4W  9.173293  0.0955582 95.99693 5.558604e-43
```

```
filter(twoGenes, 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_stage4W  3.6323772  0.1398829 25.967276 5.303201e-24
```

Learning objectives for today

1. Model more than one factor with multiple levels

- build models with multiple categorical variables and their interaction

2. Distinguish between **simple** and **main** effects

- `lm` vs `anova` tests

3. Test main effects using nested models

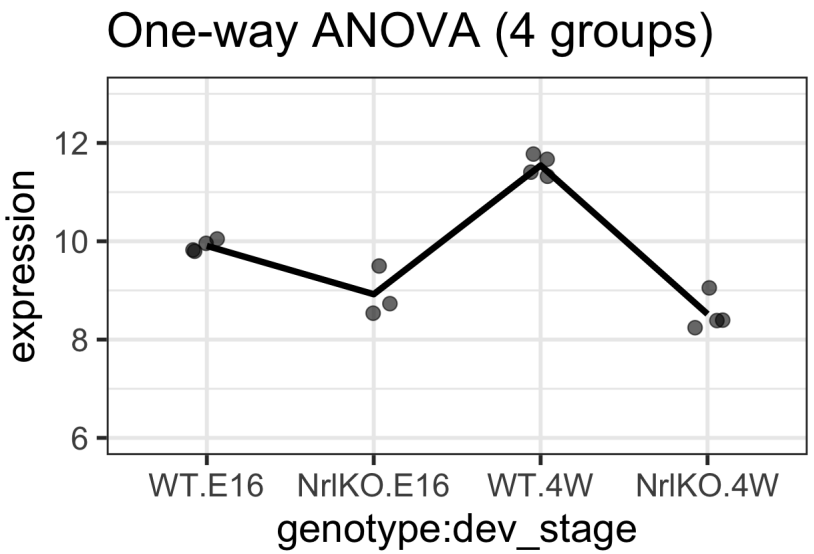
- t-tests vs F-tests

Increasing the complexity of the linear model ...

What if you have *two* categorical variables?

e.g., `genotype` and `dev_stage` (for simplicity, let's consider only E16 and 4W)

- ANOVA is usually used to study models with one or more categorical variables (factors)
- Can we combine 2 levels in each of 2 factors into 4 groups (treat as one factor)?

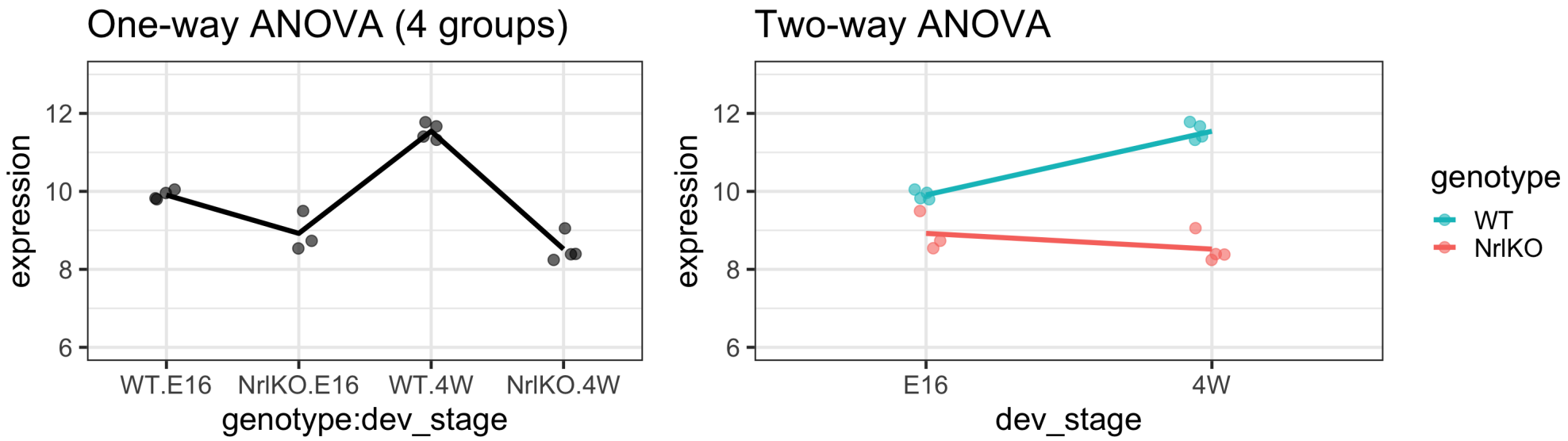


Increasing the complexity of the linear model ...

What if you have *two* categorical variables?

e.g., `genotype` and `dev_stage` (for simplicity, let's consider only E16 and 4W)

- ANOVA is usually used to study models with one or more categorical variables (factors)
- Can we combine 2 levels in each of 2 factors into 4 groups (treat as one factor)?
  - This would be a one-way ANOVA: we miss the interaction effect



Two-way ANOVA (or a linear model with interaction)

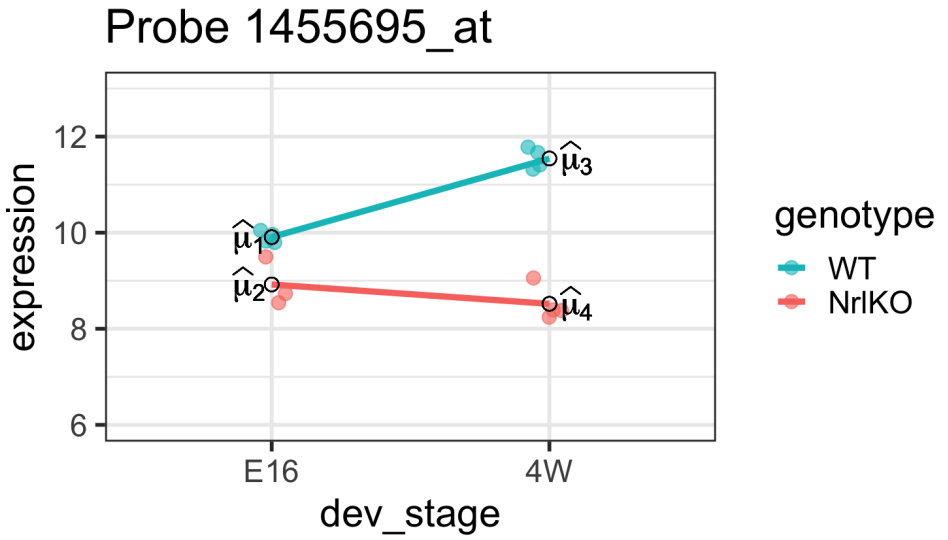
Which group means are we comparing in a model with 2 factors?

$$\mu_1 = E[Y_{(WT,E16)}]$$

$$\mu_2 = E[Y_{(NrlKO,E16)}]$$

$$\mu_3 = E[Y_{(WT,4W)}]$$

$$\mu_4 = E[Y_{(NrlKO,4W)}]$$



Reference-treatment effect parameterization

- By default, `lm` assumes a **reference-treatment effect** parameterization
- Mathematically, we need *more* dummy variables, see companion handout for more details

```
twoFactFit <- lm(expression ~ genotype * dev_stage, oneGene)
```

```
##                                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)                   9.9069542  0.1574053 62.939133 2.017456e-15
## genotypeNrlKO                -0.9844049  0.2404406 -4.094171 1.776894e-03
## dev_stage4W                   1.6366093  0.2226047  7.352087 1.444463e-05
## genotypeNrlKO:dev_stage4W -2.0403721  0.3276653 -6.227001 6.465669e-05
```

Cell-means and treatment effects in the two-way model - why do we need more dummy variables?

```
table(oneGene$dev_stage, oneGene$genotype)
```

```
##
##        WT NrlKO
##   E16   4     3
##   4W    4     4
```

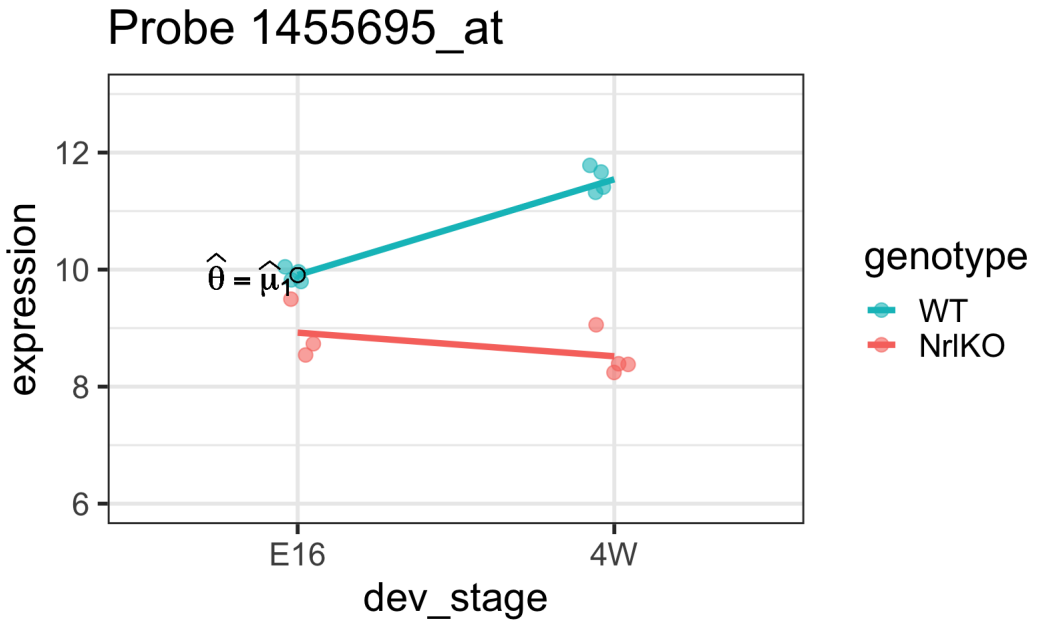
```
(means.2Fact <- group_by(oneGene, dev_stage, genotype) %>%
          summarize(cellMeans = mean(expression)) %>% ungroup() %>%
          mutate(txEffects = cellMeans - cellMeans[1],
                 lmEst = as.vector(summary(twoFactFit)$coeff[,1])))
```

```
## # A tibble: 4 x 5
##   dev_stage genotype cellMeans txEffects  lmEst
##   <fct>     <fct>        <dbl>     <dbl>  <dbl>
## 1 E16       WT            9.91     0      9.91
## 2 E16       NrlKO         8.92    -0.984 -0.984
## 3 4W        WT           11.5      1.64   1.64
## 4 4W        NrlKO         8.52    -1.39  -2.04
```

What is the reference group here?

WT & E16

As before, comparisons are relative to a reference but in this case there is a reference level *in each factor*: WT and E16



The reference: WT & E16

Mean of reference group: $\theta = E[Y_{WT,E16}]$

`lm` estimate: $\hat{\theta}$ is the sample mean of the group

```
##                              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)                 9.9069542  0.1574053 62.939133 2.017456e-15
## genotypeNrlKO              -0.9844049  0.2404406 -4.094171 1.776894e-03
## dev_stage4W                 1.6366093  0.2226047  7.352087 1.444463e-05
## genotypeNrlKO:dev_stage4W -2.0403721  0.3276653 -6.227001 6.465669e-05
```

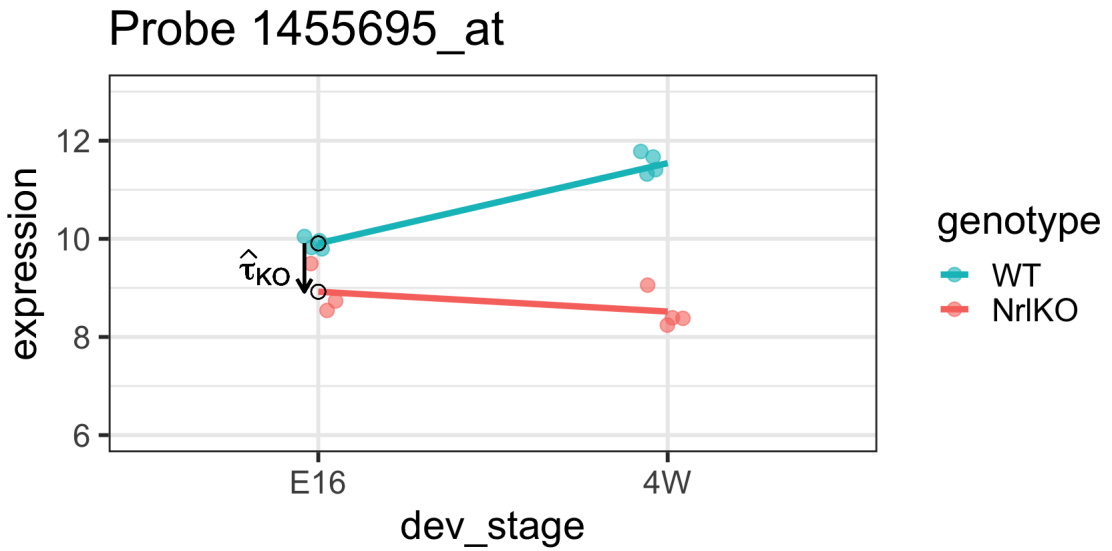
```
## # A tibble: 4 x 5
##   dev_stage genotype cellMeans txEffects  lmEst
##   <fct>     <fct>         <dbl>     <dbl>  <dbl>
## 1 E16       WT             9.91     0      9.91
## 2 E16       NrlKO          8.92    -0.984 -0.984
## 3 4W        WT            11.5      1.64   1.64
## 4 4W        NrlKO          8.52    -1.39  -2.04
```

In general, one is not interested in: $H_0 : \theta = 0$

Simple genotype effect: WT *vs* NrlKO at E16

And now the "treatment effects"...

> **Important**: effects are not marginal but *conditional* effects (at a given level of the other factor, e.g., at E16), usually called **simple effects**



Simple **genotype** effect: WT *vs* NrlKO at E16

Effect of genotype at E16: $\tau_{KO} = E[Y_{NrlKO,E16}] - E[Y_{WT,E16}]$

`lm` estimate: $\hat{\tau}_{KO}$ is the *difference* of sample respective means (check below)

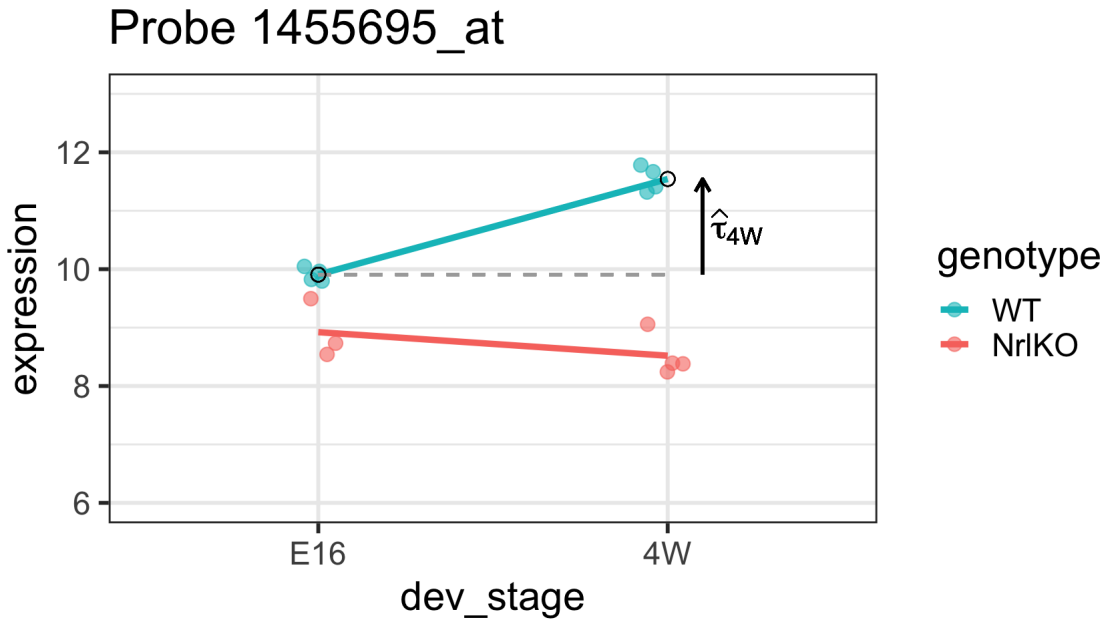
```
##                              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)                 9.9069542  0.1574053 62.939133 2.017456e-15
## genotypeNrlKO              -0.9844049  0.2404406 -4.094171 1.776894e-03
## dev_stage4W                 1.6366093  0.2226047  7.352087 1.444463e-05
## genotypeNrlKO:dev_stage4W -2.0403721  0.3276653 -6.227001 6.465669e-05
```

```
## # A tibble: 4 x 5
##   dev_stage genotype cellMeans txEffects  lmEst
##   <fct>     <fct>        <dbl>     <dbl>  <dbl>
## 1 E16       WT            9.91      0      9.91
## 2 E16       NrlKO         8.92     -0.984 -0.984
## 3 4W        WT           11.5       1.64   1.64
## 4 4W        NrlKO         8.52     -1.39  -2.04
```

But, do you want to test the *conditional* effect at E16: $H_0 : \tau_{KO} = 0$??

Simple **developmental** effect: E16 *vs* 4W in WT

Similarly, for the other factor: τ_{4W} is the effect of developmental time (4W vs E16) **in WT**



Simple developmental effect: E16 *vs* 4W in WT

Effect of development in WT: $\tau_{4W} = E[Y_{WT,4W}] - E[Y_{WT,E16}]$

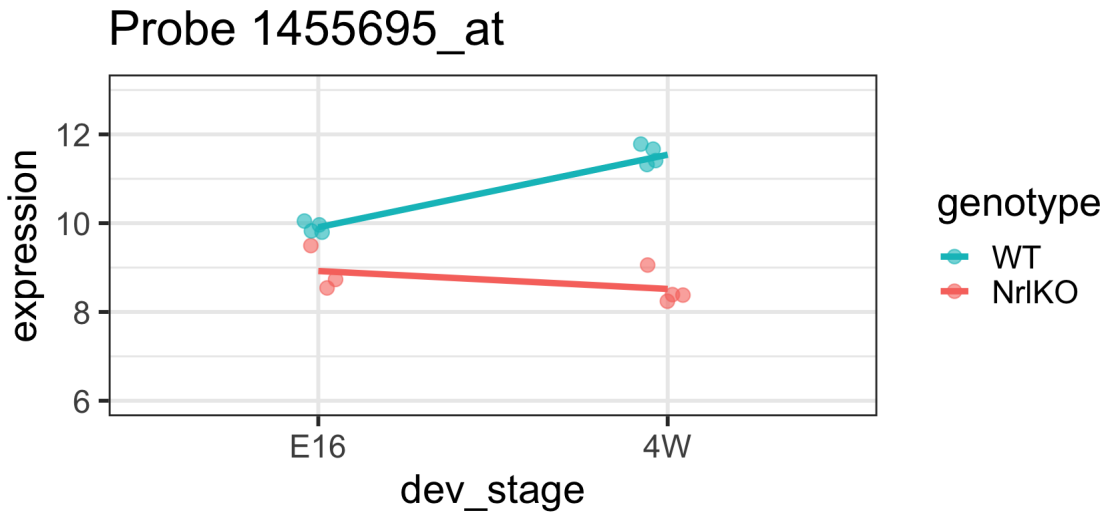
`lm` estimate: $\hat{\tau}_{4W}$ is the *difference* of respective sample means (check below)

```
##                              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)                 9.9069542  0.1574053 62.939133 2.017456e-15
## genotypeNrlKO              -0.9844049  0.2404406 -4.094171 1.776894e-03
## dev_stage4W                 1.6366093  0.2226047  7.352087 1.444463e-05
## genotypeNrlKO:dev_stage4W -2.0403721  0.3276653 -6.227001 6.465669e-05

## # A tibble: 4 x 5
##   dev_stage genotype cellMeans txEffects  lmEst
##   <fct>     <fct>        <dbl>     <dbl>  <dbl>
## 1 E16       WT            9.91     0      9.91
## 2 E16       NrlKO         8.92    -0.984 -0.984
## 3 4W        WT           11.5      1.64   1.64
## 4 4W        NrlKO         8.52    -1.39  -2.04
```

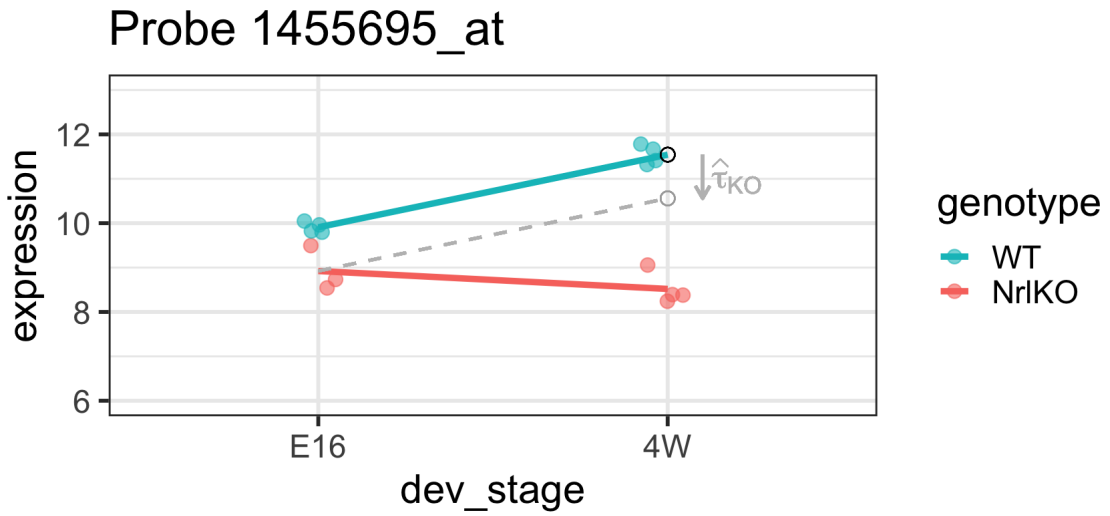
Interaction effect

Is the effect of genotype the same at different developmental stages? (or is the development effect the same for both genotypes?)



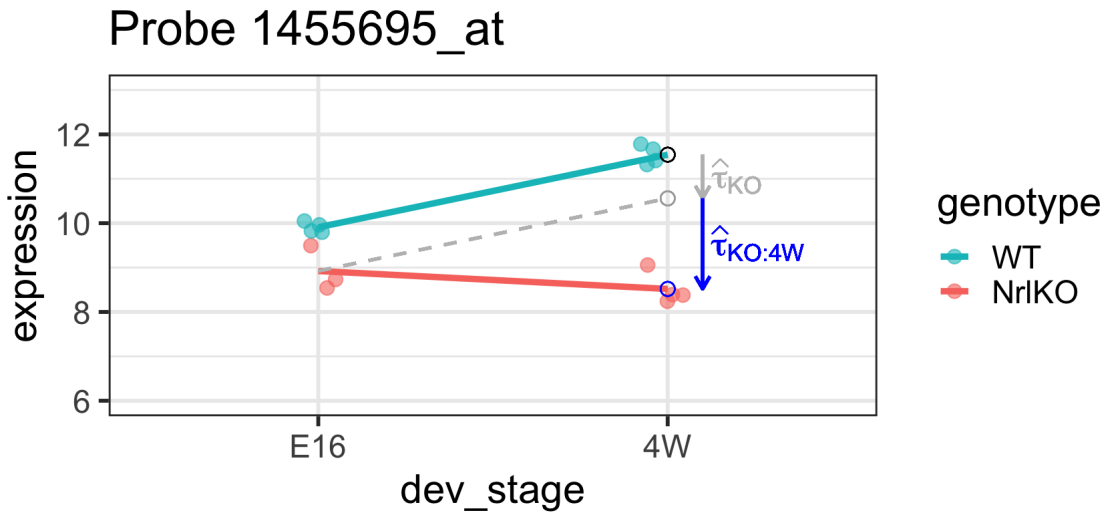
Interaction effect

Is the effect of genotype the same at different developmental stages? (or is the development effect the same for both genotypes?)



Interaction effect

Is the effect of genotype the same at different developmental stages? (or is the development effect the same for both genotypes?)



Yes, if there's no interaction effect, i.e., $\tau_{KO:4W} = 0$

The genotype effect at E16 is τ_{KO}. However, τ_{KO} does not seem to be the effect at 4W. The difference is the interaction effect!

Interaction effect

Difference of differences:

$$\tau_{KO:4W} = (E[Y_{NrlKO,4W}] - E[Y_{WT,4W}]) - (E[Y_{NrlKO,E16}] - E[Y_{WT,E16}])$$

```
##                                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)                   9.9069542  0.1574053 62.939133 2.017456e-15
## genotypeNrlKO                -0.9844049  0.2404406 -4.094171 1.776894e-03
## dev_stage4W                   1.6366093  0.2226047  7.352087 1.444463e-05
## genotypeNrlKO:dev_stage4W -2.0403721  0.3276653 -6.227001 6.465669e-05
```

```
(mean.4W.KO - mean.4W.WT) - (mean.E16.KO - mean.E16.WT)
```

```
## [1] -2.040372
```

Summary of model parameters: with interaction

model parameter	`lm` estimate	stats	interpretation
θ	`(Intercept)`	$E[Y_{WT,E16}]$	reference
τ_{KO}	`genotypeNrlKO`	$E[Y_{NrlKO,E16}] - E[Y_{WT,E16}]$	*conditional* effect of NrlKO at E16
τ_{4W}	`dev_stage4W`	$E[Y_{WT,4W}] - E[Y_{WT,E16}]$	*conditional* effect of 4W in WT
$\tau_{KO:4W}$	`genotypeNrlKO:dev_stage4W`	$E[Y_{NrlKO,4W}] - E[Y_{WT,4W}] - \tau_{KO}$	*interaction* effect of NrlKO and 4W

It is *important* to remember that `lm` reports **simple, not main** effects! Why? Because of the parameterization used! (see seecompanion handout)

It can also be shown that $\tau_{KO:4W} = E[Y_{NrlKO,4W}] - \tau_{4W} - \tau_{KO} - \theta$ (see previous slide and handout).

Let's examine these parameters closer and some examples

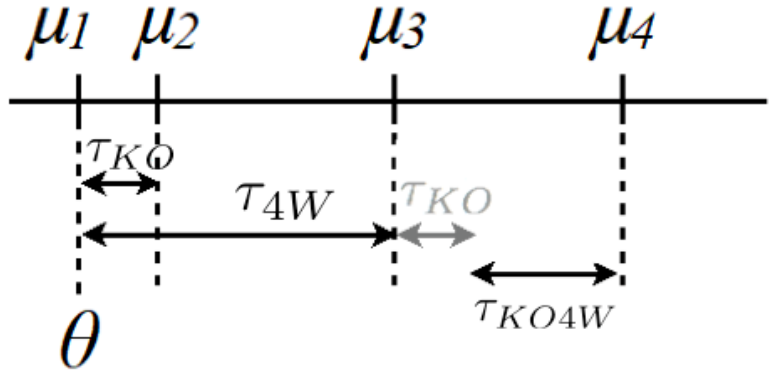
For our model, `lm` tests 4 hypotheses:

$$H_0 : \theta = 0$$

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

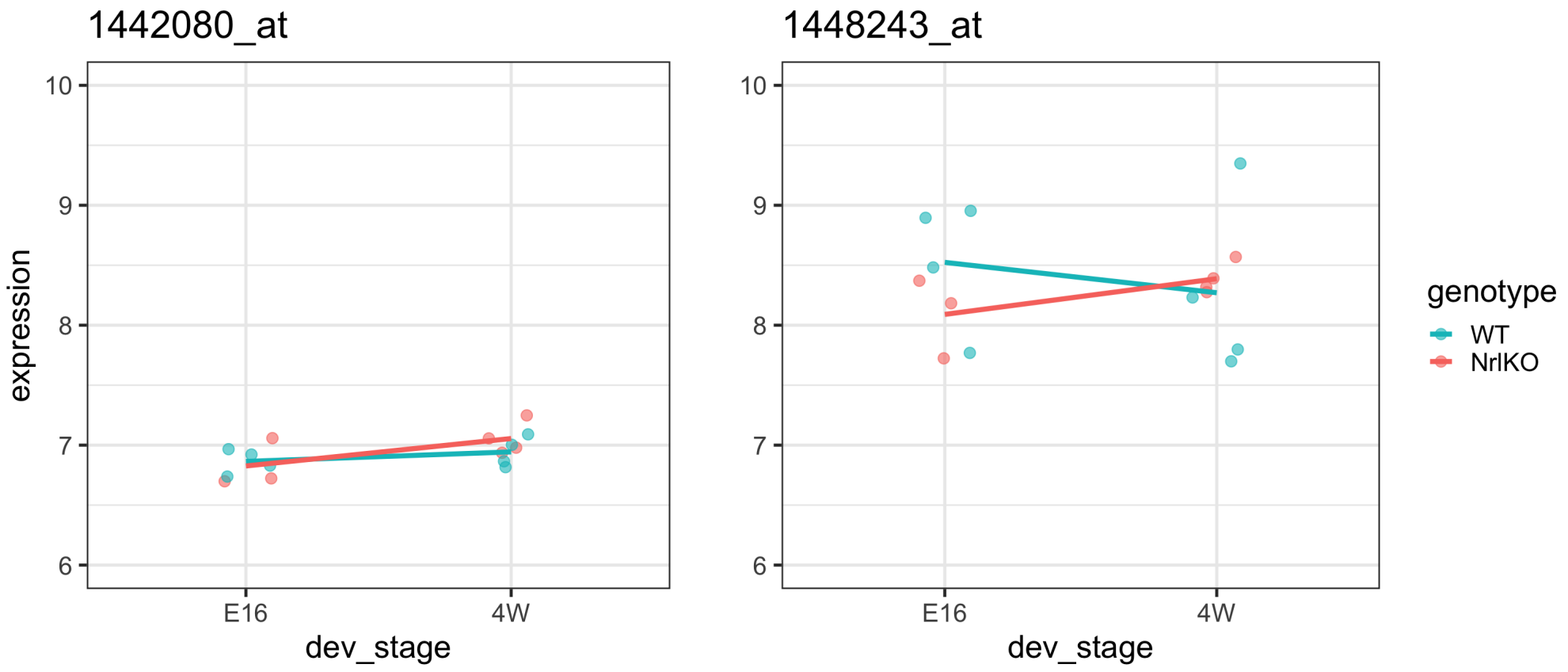
$$H_0 : \tau_{4W} = 0$$

$$H_0 : \tau_{KO:4W} = 0$$



We may not be interested in these hypotheses, e.g., τ_{KO} and τ_{4W} are *conditional* effects *at* a given level of a factor (*simple effects*)

Example 1: nothing is statistically significant, very flat genes



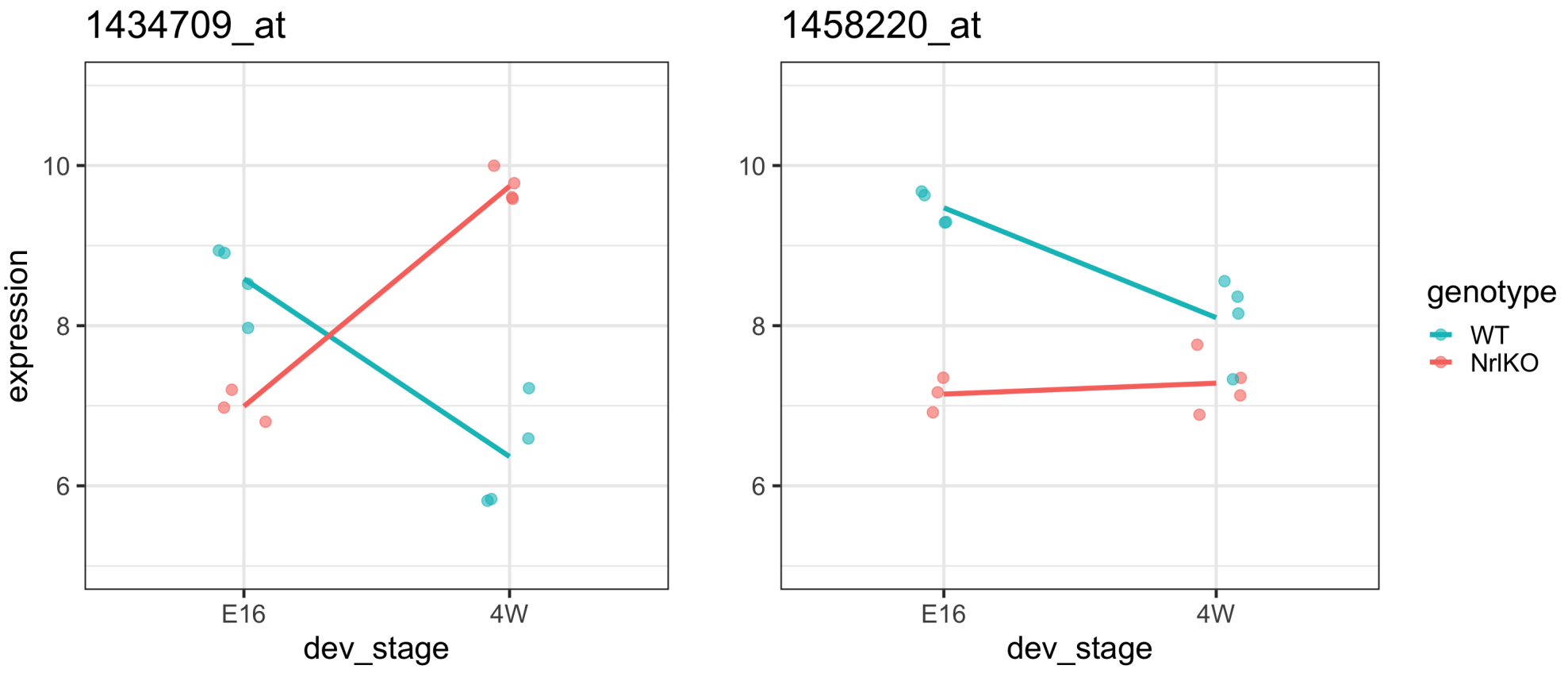
Example 1: nothing is statistically significant, very flat genes

Summary of `lm` for the gene in the right plot on previous slide:

```
filter(twoGenes, gene == "1448243_at") %>%
  lm(expression ~ genotype * dev_stage, data = .) %>%
  summary() %>% .$coeff
```

```
##                                Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)                   8.5241627  0.2561494 33.2780900 2.155313e-12
## genotypeNrlKO                -0.4340218  0.3912747 -1.1092511 2.909839e-01
## dev_stage4W                  -0.2534855  0.3622500 -0.6997531 4.986127e-01
## genotypeNrlKO:dev_stage4W     0.5511393  0.5332175  1.0336107 3.235080e-01
```

Example 2: statistically significant interaction (non-parallel)



Example 2: statistically significant interaction (non-parallel)

Summary of `lm` for the gene in the left plot on previous slide:

```
##                              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)                  8.585291  0.2214974 38.760231 4.080479e-13
## genotypeNrlKO               -1.592435  0.3383429 -4.706573 6.432625e-04
## dev_stage4W                 -2.220424  0.3132446 -7.088467 2.023126e-05
## genotypeNrlKO:dev_stage4W    4.969538  0.4610836 10.777954 3.479910e-07
```

- Note that interaction means the **simple** effects may not agree: compare the genotype effect @E16 (`genotypeNrlKO`) with that @4W
  - What is the effect of genotype at 4W?
- **Main** effects (overall): does genotype have an effect on gene expression?

> We can't (yet) answer this question! It depends (on the level of `dev_stage`)! (more later)

Example 3: **BALANCED** & only genotype @E16 is significant

For simplicity here, we'll add a random observation in the NrlKO & E16 group (close to its mean) to have a *balanced* design

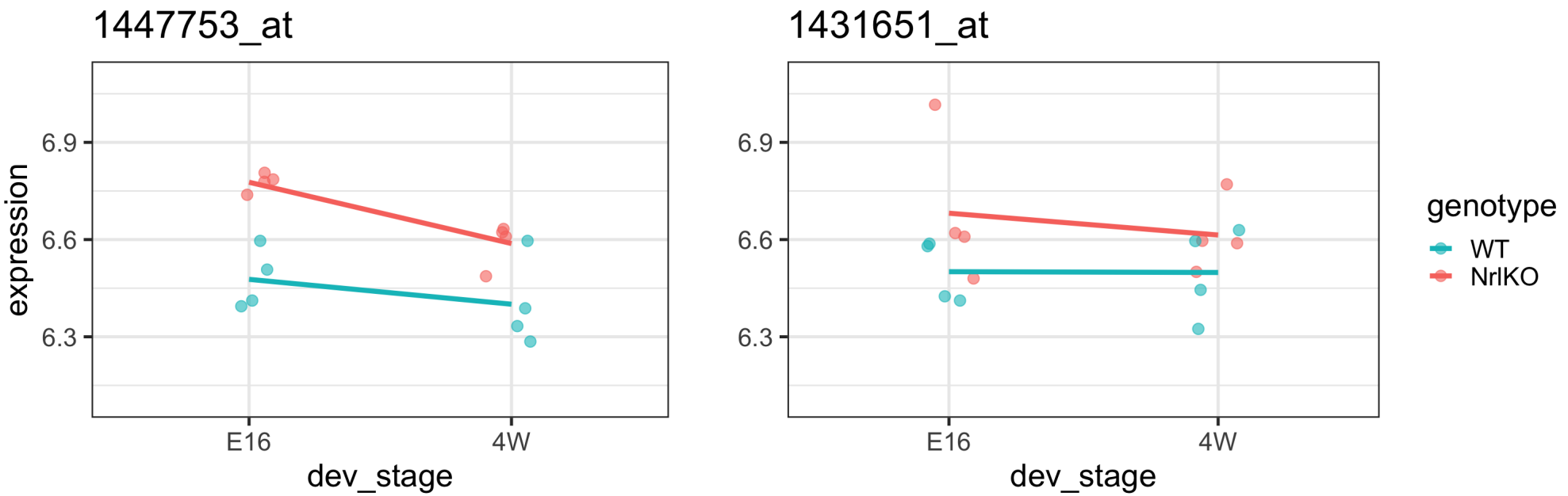
> In *unbalanced* designs the *main* effects are a *weighted* average of the simple effects, and the weights are not easy to interpret (beyond the scope of this course but worth noting the issue!)

```
# recall our unbalanced design
table(pData(eset)$genotype, pData(eset)$dev_stage)
```

```
##
##          E16 P2 P6 P10 4W
##   WT       4  4  4   4  4
##   NrlKO    3  4  4   4  4
```

```
# Duplicate sample GSM92615 (E16 NrlKO) and add noise expression
twoGenes <- filter(twoGenes, sample_id == "GSM92615") %>%
  mutate(expression = expression + rnorm(n(), 0, 0.1)) %>%
  rbind(twoGenes)
```

Example 3: only genotype @E16 is significant



- The interaction effect is not significant (almost parallel pattern)
- The effect of developmental stage is not significant for WT (almost flat pattern)

Example 3: only genotype @E16 is significant

```
filter(twoGenes, gene == "1447753_at") %>%
  lm(expression ~ genotype * dev_stage, data = .) %>%
  summary() %>% .$coeff
```

```
##                                Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)                  6.47735930 0.04533841 142.866929 9.285319e-21
## genotypeNrlKO                0.29971197 0.06411819   4.674367 5.374437e-04
## dev_stage4W                 -0.07678322 0.06411819  -1.197526 2.542215e-01
## genotypeNrlKO:dev_stage4W -0.11248197 0.09067682  -1.240471 2.385077e-01
```

- There is a genotype effect at E16
- There may be a genotype effect *regardless* of the developmental stage (**main** effect). However, that hypothesis is **not** tested here!!
- How do we test a **main** effect??

How do we test for a **main** effect?

- The main effect measures the *overall* association between the response and a factor. They are the (weighted) average of an effect over the levels of the other factor

> **Note**: a significant interaction means that the effect of a factor depends on the level of the other one. Thus, looking at main effects alone may mask interesting results!

- `anova()` can be used to test the main effects
- The following is the null hypothesis that there is no main effect of genotype:

$$H_0 : \frac{(E[Y_{KO,E16}] - E[Y_{WT,E16}]) + (E[Y_{KO,4W}] - E[Y_{WT,4W}])}{2} = 0$$

> **Note** that for unbalanced experiments $H_0 : w_1 \text{effect}_{E16} + w_2 \text{effect}_{4W} = 0$, where w_1 and w_2 are sample size weights

Main effects using `anova`

```
filter(twoGenes, gene == "1447753_at") %>%
  lm(expression ~ genotype * dev_stage, data = .) %>%
  anova() %>% tidy()
```

```
## # A tibble: 4 x 6
##   term                df  sumsq  meansq statistic   p.value
##   <chr>            <int>  <dbl>   <dbl>     <dbl>     <dbl>
## 1 genotype             1 0.237  0.237        28.8   0.000168
## 2 dev_stage            1 0.0708 0.0708        8.61  0.0125
## 3 genotype:dev_stage     1 0.0127 0.0127        1.54  0.239
## 4 Residuals             12 0.0987 0.00822     NA    NA
```

As we suspected, there is a **significant genotype effect** for this probe (1447753_at), i.e., its mean expression changes in NrlKO group (compared to WT), on average over developmental stages.

> **Technical note**: `anova()` uses *type I sums of squares* (sequential/conditional), thus order matters in unbalanced designs!

Main & interaction effects: important notes

- A **significant interaction effect** means that the effect of one factor depends on the levels of another
    - e.g., the effect of genotype depends on developmental stage
- **Main effects**: are the (weighted) average of an effect over the levels of the other factor.
- A **non-significant main effect** means that, on average, there's no evidence of a factor's effect
    - e.g., no evidence of a genotype effect, on average over both developmental stages
- **Note of caution**: if the interaction is significant, it is possible that one or both simple effects are significant but the average effect (i.e., the main effect) is not. This is because the effect of a factor *depends on* the level of the other one!

Additive models

- In some applications, we need to/want to test the interaction term
- However, additive models are easier and smaller
- If there are no statistical or biological grounds to include the interaction term, additive models are preferred
- Additive effects: $E[Y_{NrlKO,4W}] - E[Y_{WT,E16}] = \tau_{KO} + \tau_{4W}$

```
filter(twoGenes, gene == "1447753_at") %>%
  lm(expression ~ genotype + dev_stage, data = .) %>%
  summary() %>% .$coeff
```

```
##                  Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)    6.5054798 0.04006958 162.354570 6.917015e-23
## genotypeNrlKO  0.2434710 0.04626837   5.262148 1.535965e-04
## dev_stage4W   -0.1330242 0.04626837  -2.875057 1.301624e-02
```

Additive models and balanced designs

- In an additive model for a balanced design, the parameters are **average effects**, over the levels of the other factor. Now, same as in `anova()`!
    - Note the agreement! This is gone in unbalanced designs since weights are computed differently!
- The intercept parameter is now $\bar{Y} - \bar{x}_{ij,KO}\hat{\tau}_{KO} - \bar{x}_{ij,4W}\hat{\tau}_{4W}$

> **Note**: *Type III sum of squares* (marginal, conditional on all other terms in the model) are required for agreement in unbalanced designs (use `car::Anova` to obtain) - beyond our scope

Parameters in balanced additive models represent main effects

```
(fit <- filter(twoGenes, gene == "1447753_at") %>%
  lm(expression ~ genotype + dev_stage, data = .)) %>%
  summary() %>% .$coeff
```

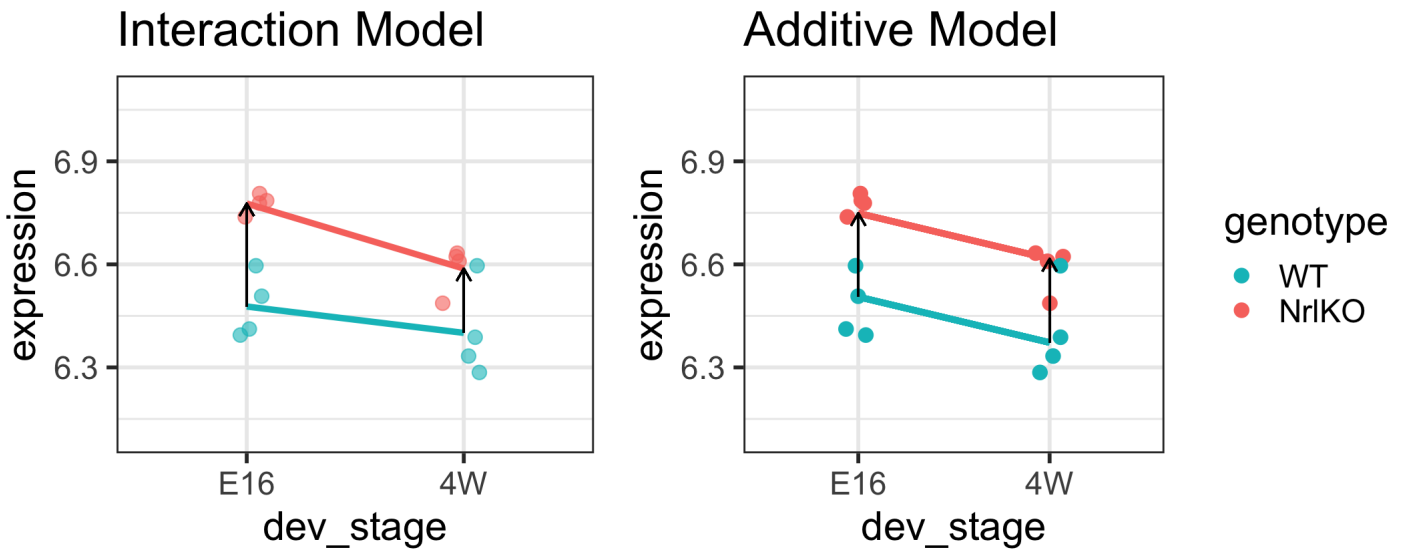
```
##                 Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)    6.5054798 0.04006958 162.354570 6.917015e-23
## genotypeNrlKO  0.2434710 0.04626837   5.262148 1.535965e-04
## dev_stage4W   -0.1330242 0.04626837  -2.875057 1.301624e-02
```

```
summary(fit)$coeff[2,3]^2
```

```
## [1] 27.6902
```

```
fit %>% anova() %>% tidy()
```

```
## # A tibble: 3 x 6
##   term         df  sumsq  meansq statistic   p.value
##   <chr>     <int>  <dbl>   <dbl>     <dbl>     <dbl>
## 1 genotype      1 0.237  0.237       27.7   0.000154
## 2 dev_stage     1 0.0708 0.0708       8.27  0.0130
## 3 Residuals    13 0.111  0.00856     NA    NA
```




```
addEst # additive model estimates
```

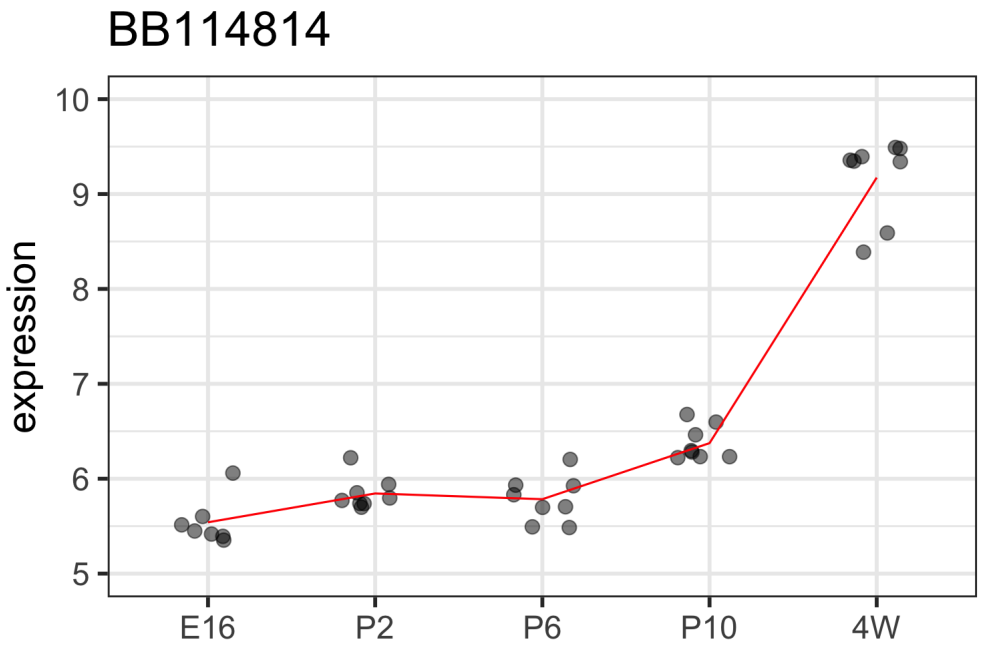
```
##   (Intercept) genotypeNrlKO   dev_stage4W
##     6.5054798     0.2434710    -0.1330242
```

```
multEst # interaction model estimates
```

```
##                (Intercept)               genotypeNrlKO                dev_stage4W
##                 6.47735930                  0.29971197                -0.07678322
## genotypeNrlKO:dev_stage4W
##                -0.11248197
```

Interactions with multi-level factors (more than 2 groups)

Back to our old friend the BB114814 gene



Interactions with multi-level factors (more than 2 groups)

We can generalize the regression model to factors with more levels (e.g., E16, P2, P10 and 4W): we just add more dummy variables (and parameters)!

With interaction

```
##                              Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)                5.43312590  0.1240473 43.7988184 4.763442e-28
## genotypeNrlKO              0.25151061  0.1894854  1.3273350 1.947534e-01
## dev_stageP2                0.39900048  0.1754294  2.2744220 3.051881e-02
## dev_stageP6                0.19534876  0.1754294  1.1135463 2.746187e-01
## dev_stageP10               0.91994107  0.1754294  5.2439391 1.287655e-05
## dev_stage4W                3.96129987  0.1754294 22.5805932 5.974687e-20
## genotypeNrlKO:dev_stageP2 -0.22636011  0.2582251 -0.8766000 3.879079e-01
## genotypeNrlKO:dev_stageP6  0.05993135  0.2582251  0.2320896 8.180985e-01
## genotypeNrlKO:dev_stageP10 -0.20757970  0.2582251 -0.8038712 4.280120e-01
## genotypeNrlKO:dev_stage4W -0.69377534  0.2582251 -2.6867078 1.181937e-02
```

Note that all the `dev_stage` parameters are still **simple** effects, but we now have more: one for each level compared to the reference

Factors with multiple levels (cont.)

Without interaction: additive

```
##                   Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)   5.52734211  0.1101244 50.1917911 9.624981e-33
## genotypeNrlKO 0.03167277  0.0878489  0.3605369 7.207433e-01
## dev_stageP2   0.30152313  0.1418465  2.1256996 4.110021e-02
## dev_stageP6   0.24101714  0.1418465  1.6991401 9.870275e-02
## dev_stageP10  0.83185393  0.1418465  5.8644640 1.437792e-06
## dev_stage4W   3.63011490  0.1418465 25.5918468 2.428361e-23
```

Parameters are now **main** effects (on average over the levels of the other factor) but we have more!

Is developmental stage a significant effect?

We haven't tested that!!

Simultaneous hypotheses again

We generally test two types of null hypotheses:

$$H_0 : \tau_j = 0$$

vs

$$H_0 : \tau_j \neq 0$$

for each *j* individually

e.g., Is gene A differentially expressed 2 days after birth compared to E16?

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

$$H_0 : \tau_j = 0$$

vs

$$H_0 : \tau_j \neq 0$$

for all *j* at the same time

e.g., Is gene A significantly affected by time (`dev_stage`)?

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

F-test and overall significance: a deja vu

- the *t*-test in linear regression allows us to test single hypotheses. Those are given in the summary of `lm`

$$H_0 : \tau_i = 0$$

$$H_A : \tau_j \neq 0$$

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

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

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

the *F*-test allows us to test such compound tests

Overall effects: compound tests

With interaction

> $H_0 : \tau_{KO} = 0$ (1 df)
> $H_0 : \tau_{P2} = \tau_{P6} = \tau_{P10} = \tau_{4W} = 0$ (**in WT!**, 4 df)
> $H_0 : \tau_{KO:P2} = \tau_{KO:P6} = \tau_{KO:P10} = \tau_{KO:4W} = 0$ (4 df)

```
anova(itxFit) %>% tidy()
```

```
## # A tibble: 4 x 6
##   term                df   sumsq  meansq statistic   p.value
##   <chr>            <int>   <dbl>   <dbl>     <dbl>     <dbl>
## 1 genotype             1  0.0693  0.0693      1.13  2.97e- 1
## 2 dev_stage            4 71.0    17.8       288.    6.72e-23
## 3 genotype:dev_stage   4  0.689   0.172       2.80  4.43e- 2
## 4 Residuals           29  1.78    0.0616     NA    NA
```

Tests of overall effects of a factor controlling for the previous ones

Overall effects: compound tests (cont.)

Without interaction (additive)

> $H_0 : \tau_{KO} = 0$ (1 df)
> $H_0 : \tau_{P2} = \tau_{P6} = \tau_{P10} = \tau_{4W} = 0$ (**on average!**, 4 df)

```
anova(addFit) %>% tidy()
```

```
## # A tibble: 3 x 6
##   term         df   sumsq  meansq statistic   p.value
##   <chr>     <int>   <dbl>   <dbl>     <dbl>     <dbl>
## 1 genotype      1  0.0693  0.0693     0.925  3.43e- 1
## 2 dev_stage     4 71.0    17.8      237.     8.45e-24
## 3 Residuals    33  2.47    0.0750    NA     NA
```

Tests of overall effects of a factor controlling for the other one

> Note: The t-test in `lm` and the F-test (1 df) in `anova` for genotype are not equivalent here due to unbalancedness (order matters)

These examples are just special cases of **nested models**

For example: does development have a significant effect on gene expression?

Compare the models with and without `dev_stage`!!

Model 1: `expression ~ genotype`

Model 2: `expression ~ genotype + dev_stage`

Mathematically:

Model 1: $Y_{ijk} = \theta + \tau_{KO} x_{KO,ijk} + \varepsilon$

Model 2: $Y_{ijk} = \theta + \tau_{KO} x_{KO,ijk} + \tau_{P2} x_{P2,ijk} + \tau_{P6} x_{P6,ijk} + \tau_{P10} x_{P10,ijk} + \tau_{4W} x_{4W,ijk} + \varepsilon$

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

The $x_{DD,ijk}$ are dummy variables (see companion handout)

More general!

F-test to compare nested models

$$H_0 : \alpha_{k+1} = \ldots = \alpha_{k+p}$$

$$F = \frac{(SS_{reduced} - SS_{full})/(p)}{SS_{full}/(n-p-k-1)} \sim \mathbf{F}_{p,\, n-p-k-1}$$

This F-statistic compares the following two models:

- Reduced (k + 1 parameters):

$$y_i = \alpha_0 + \alpha_1 x_{i1} + \ldots + \alpha_k x_{ik} + \epsilon_i$$

- Full (p + k + 1 parameters):

$$y_i = \alpha_0 + \alpha_1 x_{i1} + \ldots + \alpha_k x_{ik} + \ldots + \alpha_p x_{ip} + \epsilon_i$$

A *significant* F-statistic here means that the full model explains significantly more variation in the outcome variable than the reduced model.

Nested models in R

```
addReduced <- lm(expression ~ genotype, data = hitGene)
addFull <- lm(expression ~ genotype + dev_stage, data = hitGene)
anova(addReduced,addFull)
```

```
## Analysis of Variance Table
##
## Model 1: expression ~ genotype
## Model 2: expression ~ genotype + dev_stage
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     37 73.497
## 2     33  2.474  4    71.023 236.84 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(addFull) %>% tidy()
```

```
## # A tibble: 3 x 6
##   term         df   sumsq  meansq statistic   p.value
##   <chr>     <int>   <dbl>   <dbl>     <dbl>     <dbl>
## 1 genotype      1  0.0693  0.0693     0.925  3.43e- 1
## 2 dev_stage     4 71.0    17.8      237.     8.45e-24
## 3 Residuals    33  2.47    0.0750    NA     NA
```

Another special case: goodness of fit!

Compare the full *vs* the intercept-only models (compound test)!

$$H_0 : \tau_{KO} = \tau_{P2} = \tau_{P6} = \tau_{P10} = \tau_{4W} = 0, \ (5 \text{ df})$$

```
## Analysis of Variance Table
##
## Model 1: expression ~ 1
## Model 2: expression ~ genotype + dev_stage
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     38 73.566
## 2     33  2.474  5    71.092 189.66 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(addFull)$fstatistic # also given in the summary of lm
```

```
##    value    numdf    dendf
## 189.6573   5.0000  33.0000
```

Goodness of fit also given in output of `lm`

```
summary(addFull)
```

```
## Call:
## lm(formula = expression ~ genotype + dev_stage, data = hitGene)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.80137 -0.12454 -0.03212  0.17038  0.50036
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.52734    0.11012  50.192  < 2e-16 ***
## genotypeNrlKO  0.03167    0.08785   0.361   0.7207
## dev_stageP2    0.30152    0.14185   2.126   0.0411 *
## dev_stageP6    0.24102    0.14185   1.699   0.0987 .
## dev_stageP10   0.83185    0.14185   5.864 1.44e-06 ***
## dev_stage4W    3.63011    0.14185  25.592  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2738 on 33 degrees of freedom
## Multiple R-squared:  0.9664,    Adjusted R-squared:  0.9613
## F-statistic: 189.7 on 5 and 33 DF,  p-value: < 2.2e-16
```

Summary so far

- ***t*-tests** can be used to test the equality of **2** population means
- **ANOVA** can be used to test the equality of **more than 2** population means simultaneously (main effects)
- **Linear regression** provides a general framework for modelling the relationship between a response and different type of explanatory variables
  - *t*-tests are used to test the significance of **simple effects** (*individual* coefficients)
  - *F*-tests are used to test the significance of **main effects** (*simultaneously* multiple coefficients)
- *F*-tests are used to compare nested models
  - e.g., **overall** effects or *goodness of fit*
- Next time: continuous explanatory variables! Multiple genes!