

Statistical Methods for High Dimensional Biology

STAT/BIOF/GSAT 540

Lecture 9 – Linear Models Part III

Rob Balshaw

01 February 2017

****based on slides from Dr. Jenny Bryan, with edits by Sara Mostafavi****

outline

- Review of previous lecture
- Linear regression III – general formulation and quantitative covariates
- Analyzing patterns across all genes

Developing mouse retina – time course for the experiment

5 distinct developmental stages:

Embryonic day 16 (E16)

Postnatal days 2, 6 and 10 (P2, P6, P10)

4 week spostnatal (4_weeks)

Experimental design

2 genotypes

wild-type (wt) vs. Nrl knockout (KO)



devStage	wt	NrlKO
E16	4	3
P2	4	4
P6	4	4
P10	4	4
4_weeks	4	4

- 1) What are the genes whose expression levels differ between the two genotypes?
- 2) What are the genes whose expression levels differ across developmental stages?
- 3) What are the genes whose expression *trajectories* across development *differs between the two genotypes*?

Other ways of asking the same questions

1. Does gene expression differ between the groups?
2. Does developmental stage matter to gene expression?
3. INTERACTION gType:devStage
 - a) Does the effect of genotype differ by developmental stage?
 - b) Are the developmental stage patterns different for the different genotypes?
 - c) Do the genotype differences vary by developmental stage?

And there are many more variations...

Two-way ANOVA

Gene i
Genotype j
Dev Stage k

Genotype effect

Interaction effect

$$y_{ijk} = \theta + \tau_j + \beta_k + (\tau\beta)_{jk} + \varepsilon_{ijk}$$

Reference (i.e., intercept)

"devStage" effect

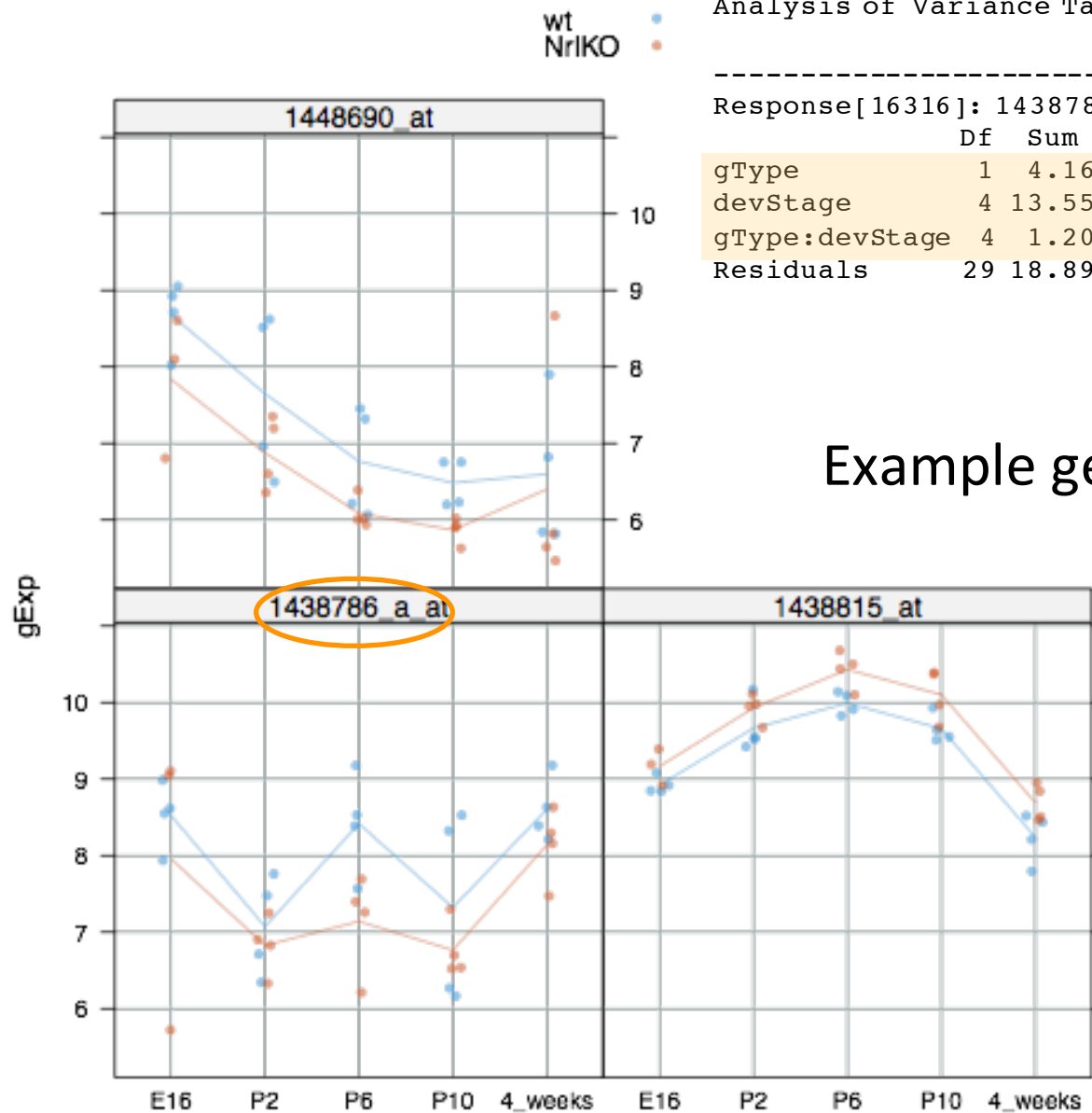
devStage	E16	P2	P6	P10	4_weeks
gType					
wt	θ	β_{P2}	β_{P6}	β_{P10}	β_{4_weeks}
NrlKO	τ_{NrlKO}	$(\tau\beta)_{NrlKO,P2}$	$(\tau\beta)_{NrlKO,P6}$	$(\tau\beta)_{NrlKO,P10}$	$(\tau\beta)_{NrlKO,4_weeks}$

Analysis of Variance Table

Response[16316]: 1438786_a_at

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
gType	1	4.1606	4.1606	6.3855	0.017216 *
devStage	4	13.5545	3.3886	5.2008	0.002774 **
gType:devStage	4	1.2014	0.3003	0.4610	0.763712
Residuals	29	18.8953	0.6516		

Example gene: only main effects

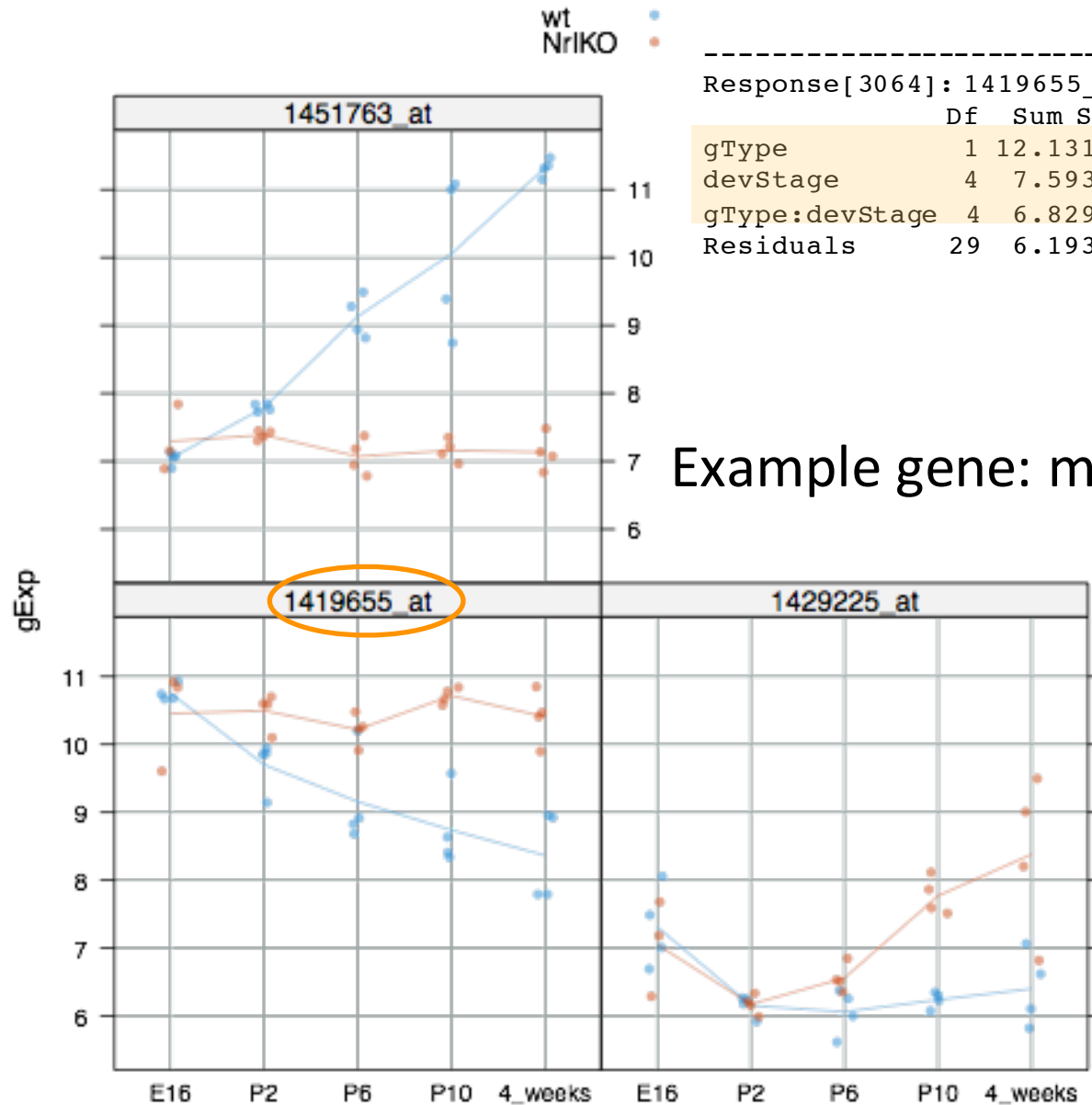


Analysis of Variance Table

Response[3064]: 1419655_at

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
gType	1	12.1312	12.1312	56.8008	2.623e-08	***
devStage	4	7.5937	1.8984	8.8888	8.210e-05	***
gType:devStage	4	6.8292	1.7073	7.9939	0.0001798	***
Residuals	29	6.1937	0.2136			

Example gene: main and interaction effects



F tests in regression

small model is nested within big -- it's a special case where some parameters are equal to zero

model	example	# params = DF	RSS
small	lm(y ~ gType + devStage)	p _{small} = 6	RSS _{small}
big	lm(y ~ gType * devStage)	p _{big} = 10	RSS _{big}

$$y_{ijk} = \theta + \tau_j + \beta_k + (\tau\beta)_{jk} + \varepsilon_{ijk} \text{ “big”}$$

$$y_{ijk} = \theta + \tau_j + \beta_k + (\cancel{\tau\beta})_{jk} + \varepsilon_{ijk} \text{ “small”}$$

F tests in regression

small model is **nested** within big -- it's a special case where some parameters are equal to zero

model	example	# params = DF	RSS
small	<code>lm(y ~ gType + devStage)</code>	$p_{\text{small}} = 6$	RSS_{small}
big	<code>lm(y ~ gType * devStage)</code>	$p_{\text{big}} = 10$	RSS_{big}

$$y_{ijk} = \theta + \tau_j + \beta_k + (\tau\beta)_{jk} + \varepsilon_{ijk} \text{ “big”}$$

$$y_{ijk} = \theta + \tau_j + \beta_k + (\tau\beta)_{jk} + \varepsilon_{ijk} \text{ “small”}$$

Plugging the observed F into the null distribution for F with given DOF, gives us the tail probability under the null

by definition:

$$p_{\text{small}} < p_{\text{big}}$$

$$RSS_{\text{small}} \geq RSS_{\text{big}}$$

$$F = \frac{\left(\frac{RSS_{\text{small}} - RSS_{\text{big}}}{p_{\text{big}} - p_{\text{small}}} \right)}{\frac{RSS_{\text{big}}}{n - p_{\text{big}}}} \sim_{H_0}$$

$$F_{(p_{\text{big}} - p_{\text{small}}, n - p_{\text{big}})}$$

Let's talk about linear regression as a very general framework.

How do we get the parameters?

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

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} = \begin{bmatrix} \alpha_0 \cdot 1 + \alpha_1 \cdot x_1 \\ \alpha_0 \cdot 1 + \alpha_1 \cdot x_2 \\ \vdots \\ \alpha_0 \cdot 1 + \alpha_1 \cdot x_n \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} = \begin{bmatrix} \alpha_0 + \alpha_1 x_1 + \varepsilon_1 \\ \alpha_0 + \alpha_1 x_2 + \varepsilon_2 \\ \vdots \\ \alpha_0 + \alpha_1 x_n + \varepsilon_n \end{bmatrix}$$

$$y_i = \alpha_0 + \alpha_1 x_i + \varepsilon_i$$

Here we are just fitting a line but using matrix notation to handle all n observations at once, more elegantly.

Big pay-offs ensue

Estimation of the parameter α

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

Two viewpoints:

- maximum likelihood estimation, assuming ε_i are iid $N(0, \sigma^2)$
- “ordinary least squares” (OLS), i.e. minimizing the sum of the squared residuals

both lead to the same estimator of α :

$$\hat{\alpha} = (X^T X)^{-1} X^T y = \min^{-1} \sum (y_i - x_i \alpha)^2$$

Linear models: maximum likelihood estimation

$$Y = X\alpha + \varepsilon \quad \varepsilon_i \sim N(0, \sigma^2)$$

$$P(Y | \theta) = N(Y | X\alpha, \sigma^2) = \frac{1}{2\sqrt{\pi}\sigma} e^{\left(-\frac{(X\alpha - Y)^T (X\alpha - Y)}{2\sigma}\right)}$$

Need to maximize the log of the likelihood function to solve for the parameter(s)

Linear models: least squares estimation

We write our model as

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

So, if we're estimating α with $\hat{\alpha}$, we would have

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

where $\hat{\varepsilon}$ is now a “residual error” or “observed prediction error”,

$$\hat{\varepsilon} = Y - \hat{Y}$$

Residual Sum of Squares, RSS (or Sum of Squared Errors, SSE) would then be

$$RSS = \sum_{i=1}^n e_i^2 = (Y - \hat{Y})^T (Y - \hat{Y}) = (Y - X\hat{\alpha})^T (Y - X\hat{\alpha})$$

Which we would minimize with respect to $\hat{\alpha}$ by differentiating and solving for $\hat{\alpha}$

$$\hat{\alpha} = (X^T X)^{-1} X^T Y$$

Greatest Hits of Regression Results (normal iid errors)

Linear Models

$$Y = X\alpha + \varepsilon \quad \text{regression model}$$

$$\hat{\alpha} = (X^T X)^{-1} X^T Y \quad \text{the MLE and OLS estimator of } \alpha$$

$$\hat{Y} = X\hat{\alpha} \quad \text{the fitted or predicted values}$$

$$\hat{Y} = X(X^T X)^{-1} X^T Y = HY \quad \text{where } H = X(X^T X)^{-1} X^T \text{ is called the "hat matrix"}$$

$$\hat{\varepsilon} = Y - \hat{Y} = Y - X\hat{\alpha} \quad \text{the residuals (note NOT the same as the errors } \varepsilon)$$

$$\hat{\sigma}^2 = \frac{1}{n-p} \hat{\varepsilon}^T \hat{\varepsilon} \quad \text{the estimated error variance (} p \text{ is the dimension of } \alpha)$$

$$\hat{V}(\hat{\alpha}) = \hat{\sigma}^2 (X^T X)^{-1} \quad \text{the estimated covariance matrix of } \hat{\alpha}$$

estimated standard errors for the estimated regression coefficients -- $\widehat{se}(\hat{\alpha}_j)$ --
are obtained by taking the square root of the diagonal elements of $\hat{V}(\hat{\alpha})$

Testing the significance of the parameter estimates in linear models

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

How test $H_0 : \alpha_j = 0$?

With a t-statistic. Under H_0 , we have (at least approximately) that:

$$\frac{\hat{\alpha}_j}{\widehat{se}(\hat{\alpha}_j)} \sim t_{n-p}$$

so a p-value is obtained by computing a tail probability for the observed value of $\hat{\alpha}_j$ from a t_{n-p} distribution.

$\hat{\sigma}^2 = \frac{1}{n-p} \hat{\varepsilon}^T \hat{\varepsilon}$ the estimated error variance (p is the dimension of α)

$\hat{V}(\hat{\alpha}) = \hat{\sigma}^2 (X^T X)^{-1}$ the estimated covariance matrix of $\hat{\alpha}$

estimated standard errors for the estimated regression coefficients -- $\widehat{se}(\hat{\alpha}_j)$ --

are obtained by taking the square root of the diagonal elements of $\hat{V}(\hat{\alpha})$

Investigating a *quantitative* covariate
in our example dataset

```

> ## recode() is from add-on package 'car'

> prDes$age <-
+   recode(prDes$devStage,
+         "'E16'=-2; 'P2'=2; 'P6'=6; 'P10'=10; '4_weeks'=28",
+         as.factor.result = FALSE)

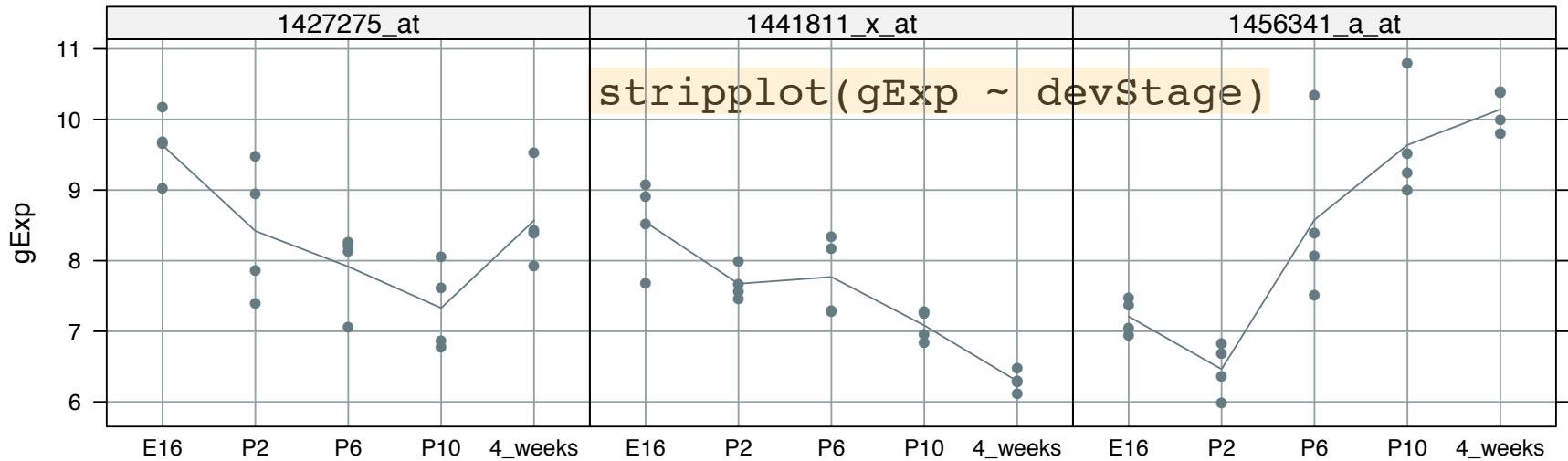
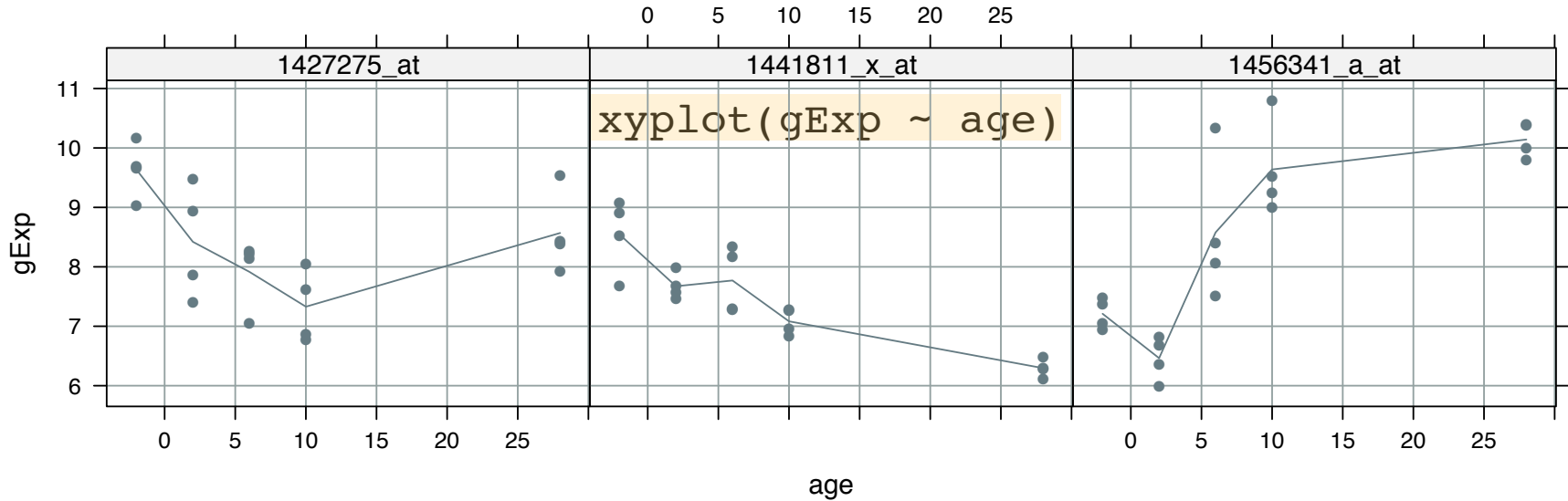
> peek(prDes)
      sample devStage gType age
Sample_22    22      E16   wt  -2
Sample_16    16      E16 NrlKO -2
Sample_5      5       P2 NrlKO  2
Sample_31    31       P6   wt   6
Sample_15    15      P10 NrlKO 10
Sample_36    36  4_weeks   wt  28
Sample_2      2  4_weeks NrlKO 28

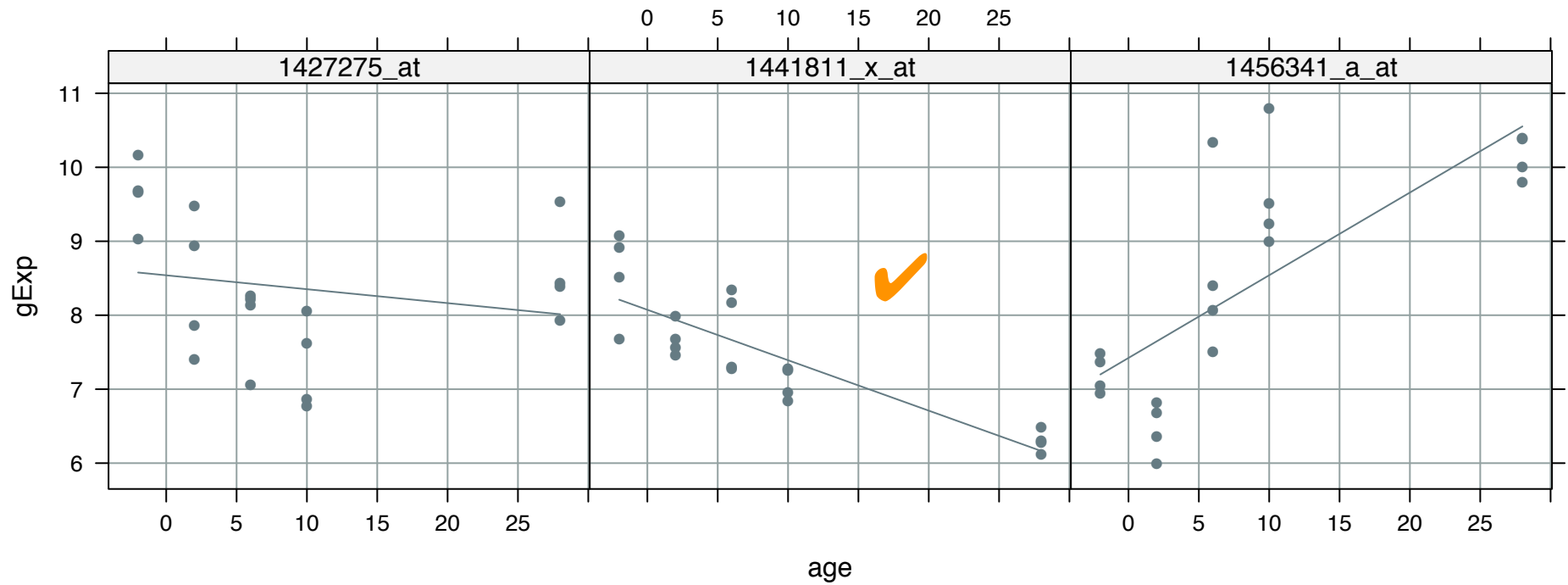
> str(prDes)
'data.frame':  39 obs. of  4 variables:
 $ sample  : num  20 21 22 23 16 17 6 24 25 26 ...
 $ devStage: Factor w/ 5 levels "E16","P2","P6",...: 1 1 1 1 1 1 1 2 2 2 ...
 $ gType   : Factor w/ 2 levels "wt","NrlKO": 1 1 1 1 2 2 2 1 1 1 ...
 $ age     : num  -2 -2 -2 -2 -2 -2 -2 2 2 2 ...

```

meet our new quantitative covariate or predictor ...
age, which is a new version of the factor devStage

for starters, let's just work with wild type data for 3 example probesets





linear looks reasonable for 1, but
not the other two

Remember: $Y=f(x)= a_0 + b x$

- The nature of the regression function $f(x; \alpha)$ is one of the defining characteristics of a regression model
 - f linear in $\alpha \Rightarrow$ linear model
 - f not linear in $\alpha \Rightarrow$ nonlinear model

nonlinear parametric regression

$$Y = \frac{1}{1 + e^{(\phi - x)/\zeta}} + \varepsilon$$

Linear in α means that the model is made up of terms that are the product of a parameter and a predictor (or a transformation of a predictor like $\log(x)$, x^2 or $1/x$)

Try:

<http://blog.minitab.com/blog/adventures-in-statistics-2/linear-or-nonlinear-regression-that-is-the-question>

simple linear regression (a linear model)

$$Y = \alpha_0 + \alpha_1 x + \varepsilon$$

What we just did.



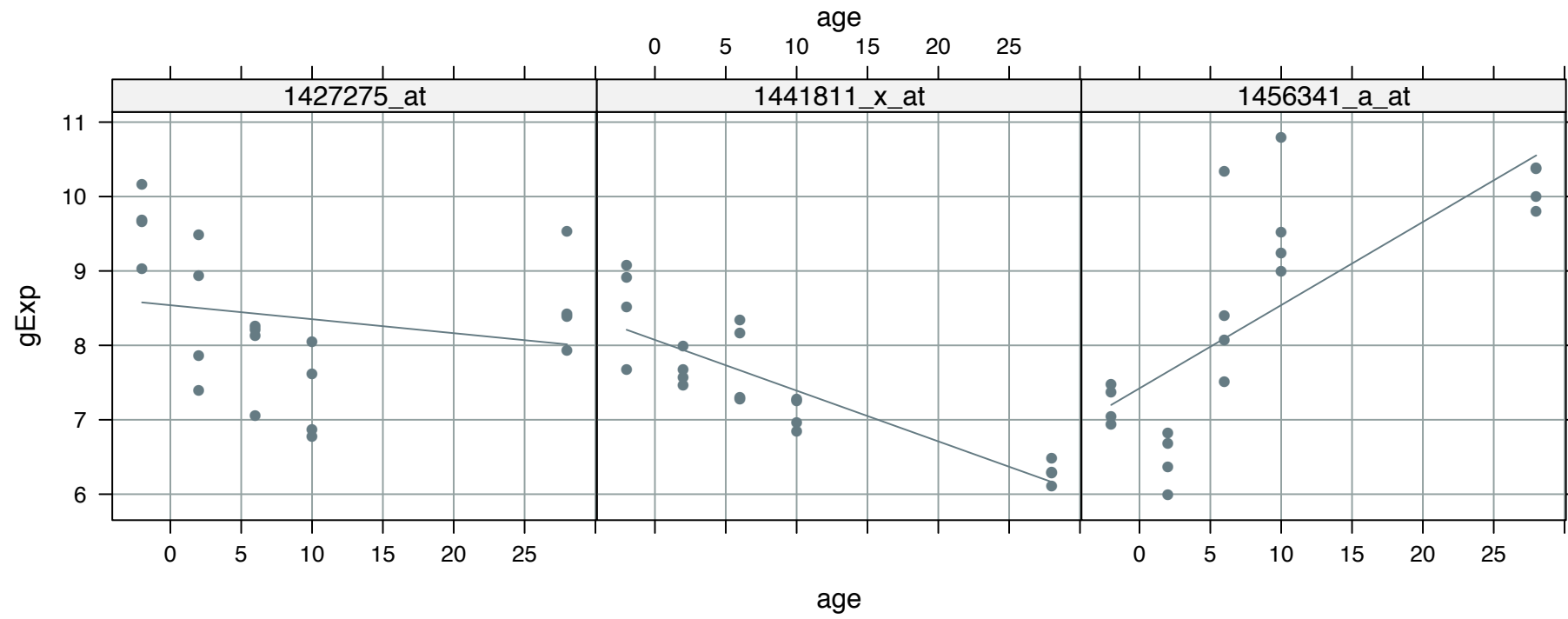
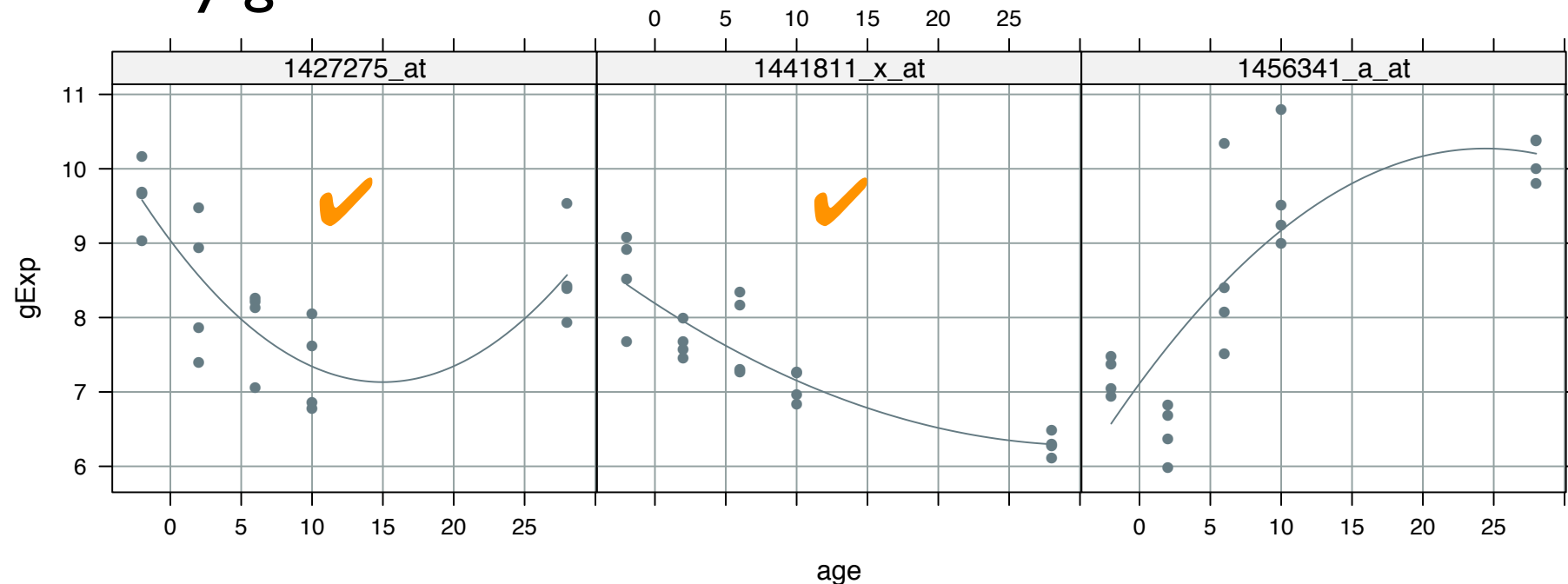
polynomial regression (also a linear model)

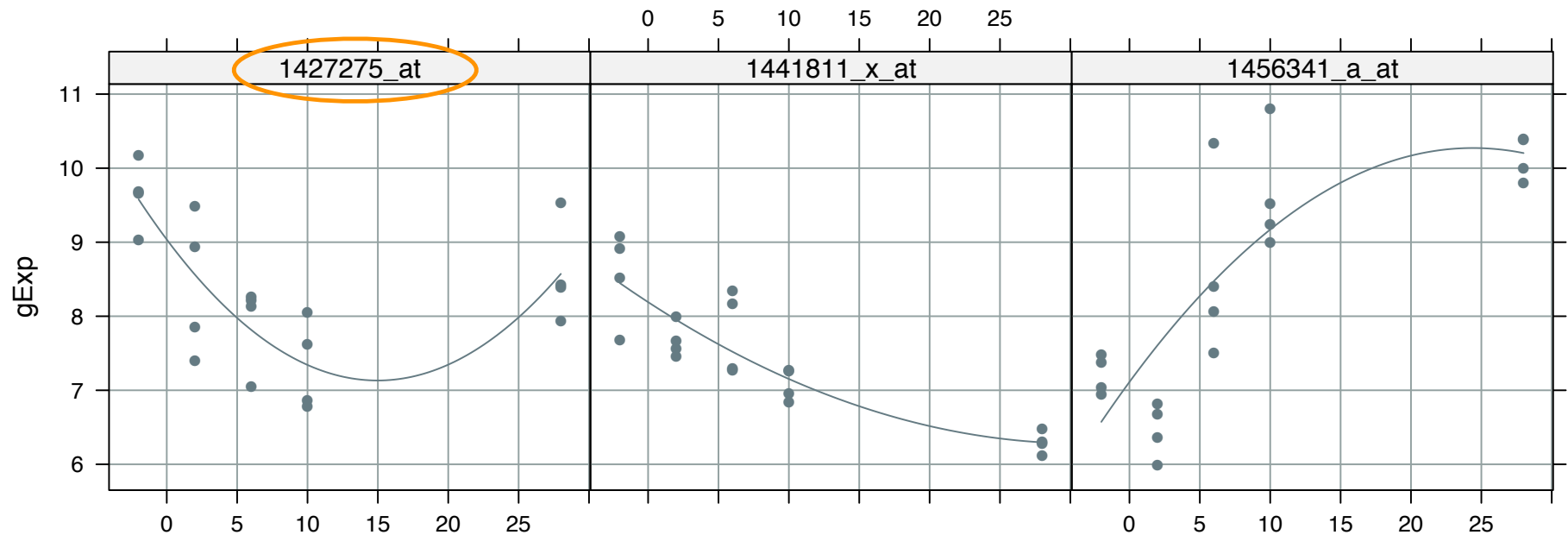
$$Y = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \varepsilon$$

What we're
about to do.



fairly good fit for 2 of 3 now!





```
> summary(quadFits[["1427275_at"]])
```

age

Call:

```
lm(formula = gExp ~ age + I(age^2), data = z)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.16275	-0.55506	0.09503	0.40804	0.95803

Coefficients:

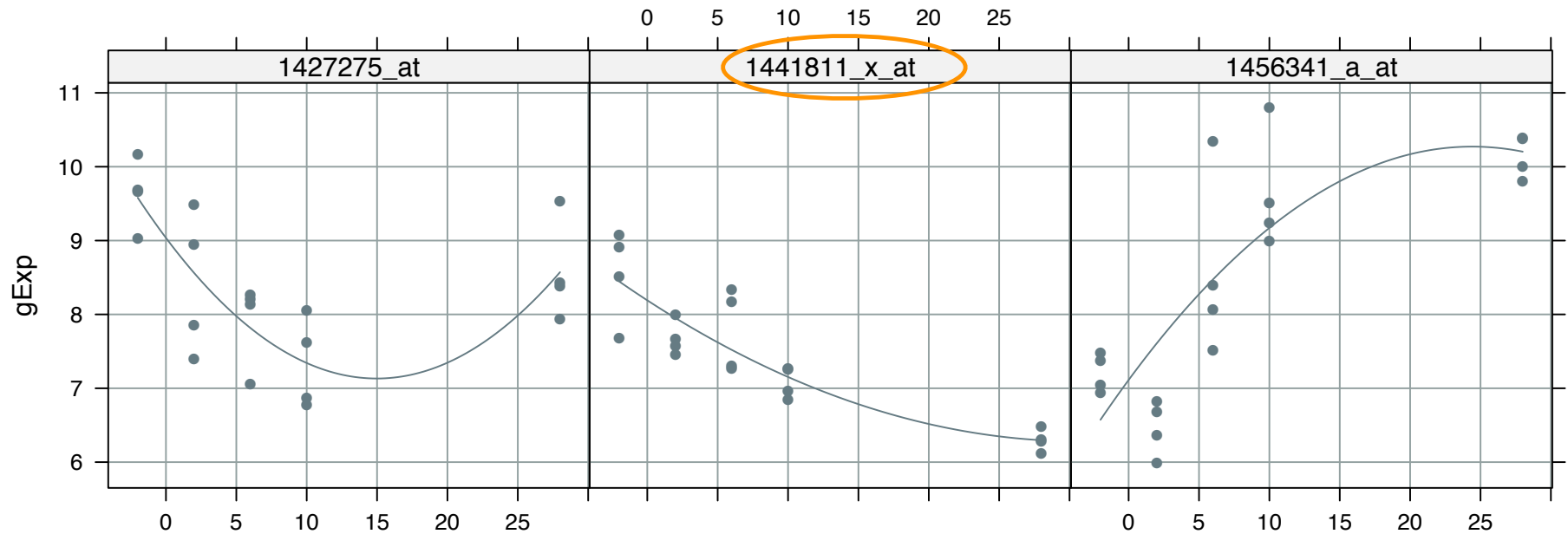
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	9.036401	0.212313	42.562	< 2e-16 ***
age	-0.254305	0.048125	-5.284	6.07e-05 ***
I(age^2)	0.008490	0.001661	5.110	8.71e-05 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6444 on 17 degrees of freedom

Multiple R-squared: 0.6218, Adjusted R-squared: 0.5773

F-statistic: 13.98 on 2 and 17 DF, p-value: 0.0002572



```
> summary(quadFits[["1441811_x_at"]])
```

age

Call:

```
lm(formula = gExp ~ age + I(age^2), data = z)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.76946	-0.25477	-0.00589	0.13662	0.82202

Coefficients:

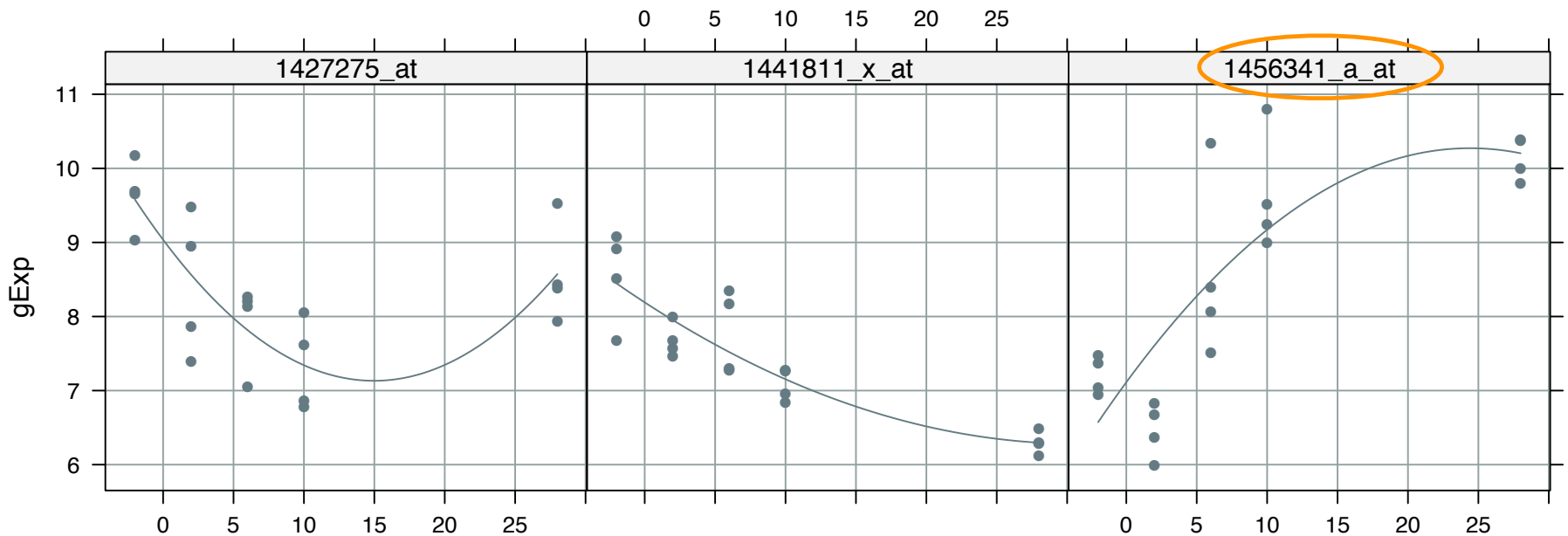
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	8.190766	0.140969	58.103	< 2e-16 ***
age	-0.123836	0.031953	-3.876	0.00121 **
I(age^2)	0.002006	0.001103	1.819	0.08660 .

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4279 on 17 degrees of freedom

Multiple R-squared: 0.774, Adjusted R-squared: 0.7475

F-statistic: 29.12 on 2 and 17 DF, p-value: 3.23e-06



```
> summary(quadFits[["1456341_a_at"]])
```

age

Call:

```
lm(formula = gExp ~ age + I(age^2), data = z)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.6211	-0.5010	-0.0050	0.3955	1.8651

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	7.112481	0.310922	22.875	3.3e-14 ***
age	0.258892	0.070477	3.673	0.00188 **
I(age^2)	-0.005303	0.002433	-2.180	0.04363 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9437 on 17 degrees of freedom

Multiple R-squared: 0.6737, Adjusted R-squared: 0.6353

F-statistic: 17.55 on 2 and 17 DF, p-value: 7.337e-05

F tests in regression

Remember this?

small model is nested within big, e.g., it's a special case where some parameters are equal to zero

model	example	# params = DF	RSS
small	$\text{lm}(y \sim \text{gType} + \text{devStage})$	$p_{\text{small}} = 6$	$\text{RSS}_{\text{small}}$
big	$\text{lm}(y \sim \text{gType} * \text{devStage})$	$p_{\text{big}} = 10$	RSS_{big}

$$y_{ijk} = \theta + \tau_j + \beta_k + (\tau\beta)_{jk} + \varepsilon_{ijk} \text{ "big"}$$

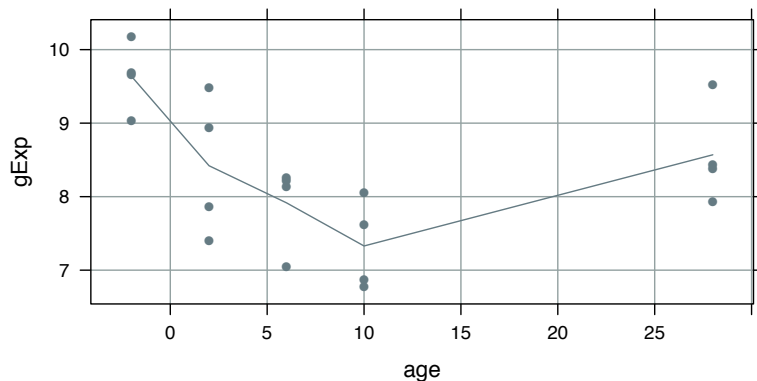
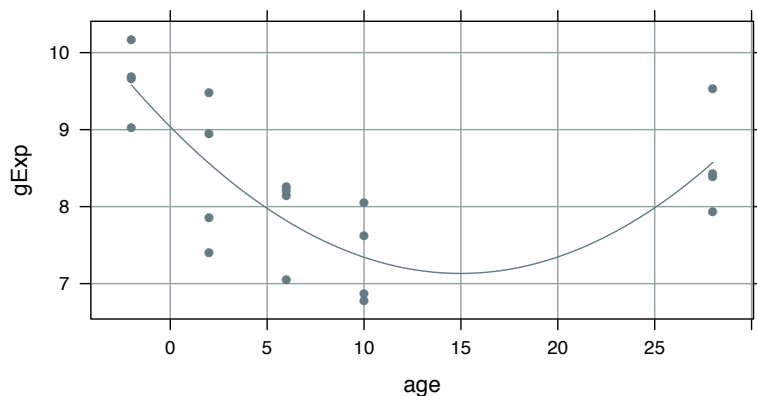
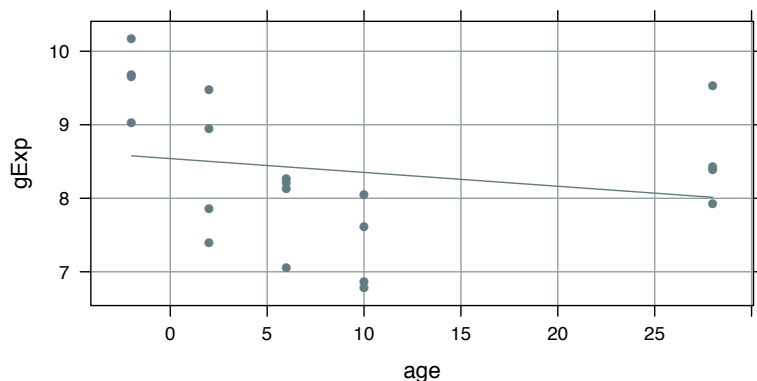
$$y_{ijk} = \theta + \tau_j + \beta_k + (\cancel{\tau\beta})_{jk} + \varepsilon_{ijk} \text{ "small"}$$

by definition:

$$p_{\text{small}} < p_{\text{big}}$$

$$\text{RSS}_{\text{small}} \geq \text{RSS}_{\text{big}}$$

$$F = \frac{\left(\frac{\text{RSS}_{\text{small}} - \text{RSS}_{\text{big}}}{p_{\text{big}} - p_{\text{small}}} \right)}{\frac{\text{RSS}_{\text{big}}}{n - p_{\text{big}}}} \sim_{H_0} F_{(p_{\text{big}} - p_{\text{small}}, n - p_{\text{big}})}$$



```
> (jGene <- luckyGenes[1])
[1] "1427275_at"
```

small

big

```
> anova(linFits[[jGene]], quadFits[[jGene]])
```

Analysis of Variance Table

Model 1: $\text{gExp} \sim \text{age}$

Model 2: $\text{gExp} \sim \text{age} + \text{I}(\text{age}^2)$

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	18	17.9021				
2	17	7.0591	1	10.843	26.113	8.71e-05 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

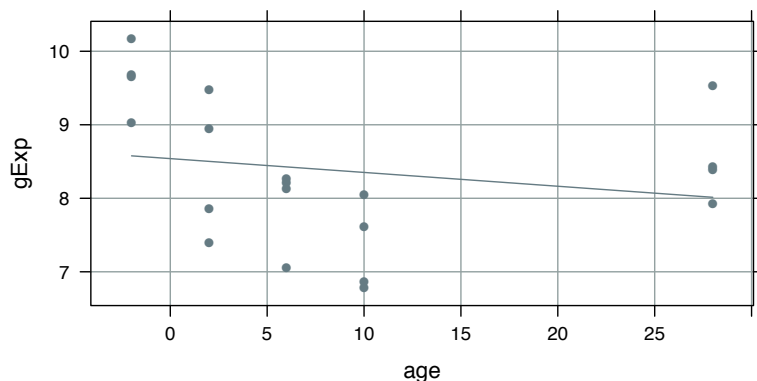
```
> AIC(linFits[[jGene]], quadFits[[jGene]], factFits[[jGene]])
```

	df	AIC
linFits[[jGene]]	3	60.54129
quadFits[[jGene]]	4	43.92930
factFits[[jGene]]	6	47.54810

it's "worth it" to go from linear to quadratic here

but hard to justify going from quadratic to one-way ANOVA

possible links to read more about using AIC to compare non-nested models: [stackexchange](#) and [Wikipedia](#)



```
> (jGene <- luckyGenes[1])
[1] "1427275_at"
```

small

big

```
> anova(linFits[[jGene]], quadFits[[jGene]])
```

Analysis of Variance Table

Model 1: gExp ~ age

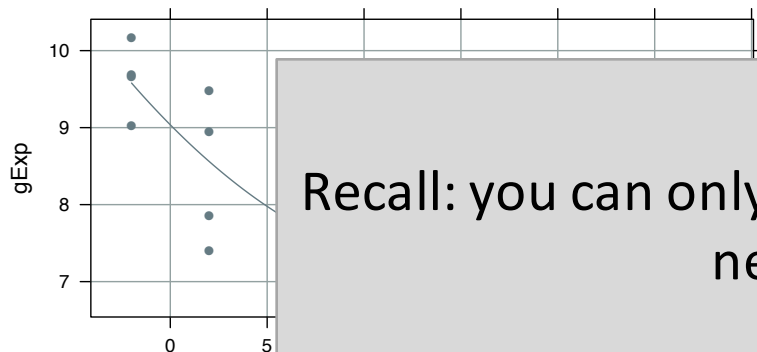
Model 2: gExp ~ age + I(age^2)

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	18	1.7	0.021			

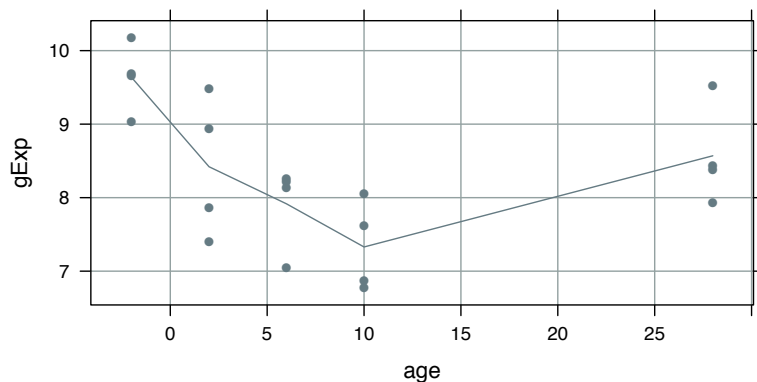
0.05 ***

0.05 '.' 0.1 ' ' 1

```
factFits[[jGene]])
```



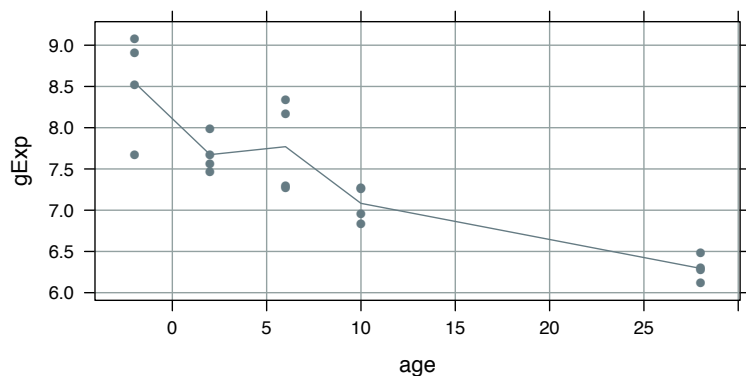
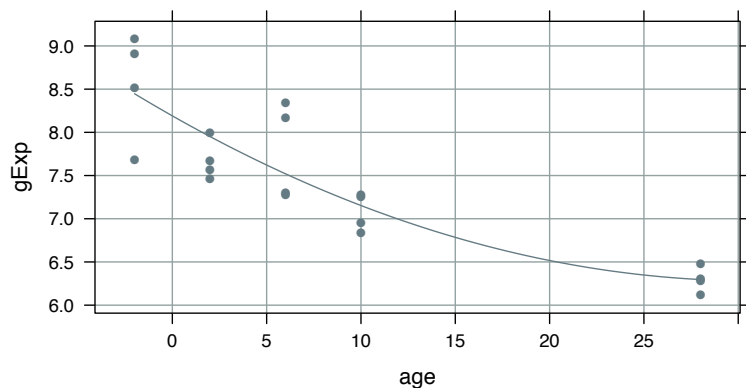
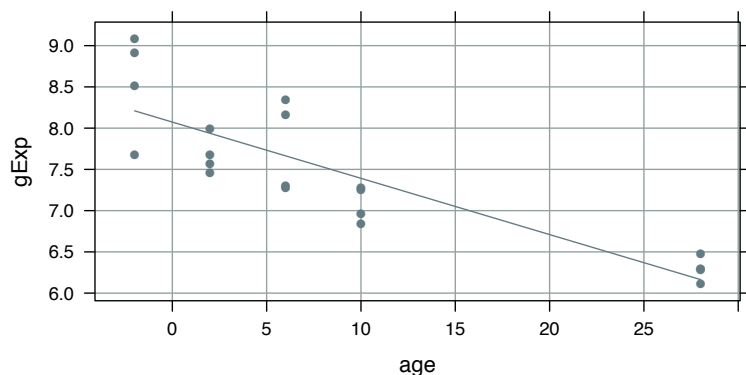
Recall: you can only use the F test when you have nested models



it's "worth it" to go from linear to quadratic here

but hard to justify going from quadratic to one-way ANOVA

possible links to read more about using AIC to compare non-nested models: [stackexchange](#) and [Wikipedia](#)



```
> (jGene <- luckyGenes[3])
```

```
[1] "1441811_x_at"
```

small

big

```
> anova(linFits[[jGene]], quadFits[[jGene]])
```

Analysis of Variance Table

Model 1: gExp ~ age

Model 2: gExp ~ age + I(age^2)

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	18	3.7176				
2	17	3.1120	1	0.60559	3.3081	0.0866 .

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> AIC(linFits[[jGene]], quadFits[[jGene]], factFits[[jGene]])
```

	df	AIC
linFits[[jGene]]	3	29.10466
quadFits[[jGene]]	4	27.54851
factFits[[jGene]]	6	27.12587

meh

not clear it's “worth it” to go from linear to quadratic here

even less payoff to go from quadratic to one-way ANOVA

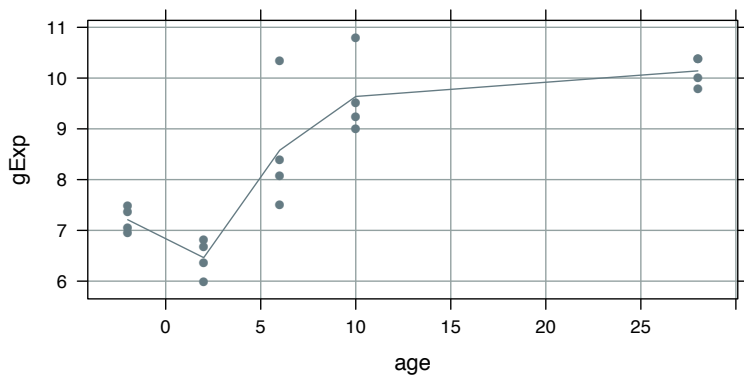
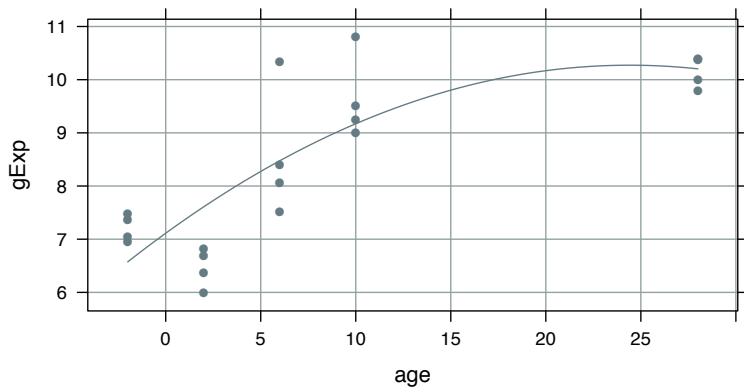
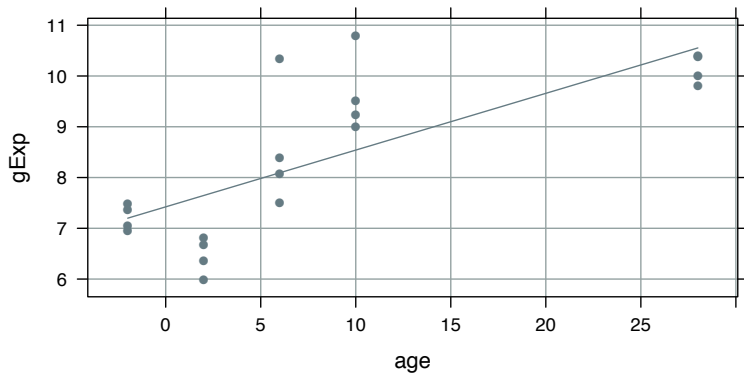
Occam's Razor and the KISS principle → stick w/ simple linear model

Occam's razor

- Occam's razor (also Ockham's razor; Latin: *lex parsimoniae* "law of parsimony")
- A problem-solving principle attributed to William of Ockham (c. 1287–1347), who was an English Franciscan friar, scholastic philosopher and theologian.
- The principle can be interpreted as stating *Among competing hypotheses, the one with the fewest assumptions should be selected.*
- Roughly speaking: do not make things more complicated than they need to be
- “Principle of parsimony”



By self-created (Moscarlop) - Own work, CC BY-SA 3.0,
<https://commons.wikimedia.org/w/index.php?curid=5523066>



```
> (jGene <- luckyGenes[2])
```

```
[1] "1456341_a_at"
```

small

big

```
> anova(linFits[[jGene]], quadFits[[jGene]])
```

Analysis of Variance Table

Model 1: gExp ~ age

Model 2: gExp ~ age + I(age^2)

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	18	19.370				
2	17	15.139	1	4.2308	4.7509	0.04363 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> AIC(linFits[[jGene]], quadFits[[jGene]], factFits[[jGene]])
```

	df	AIC
linFits[[jGene]]	3	62.11743
quadFits[[jGene]]	4	59.18864
factFits[[jGene]]	6	48.70210

it's probably “worth it” to go from linear to quadratic here (?)

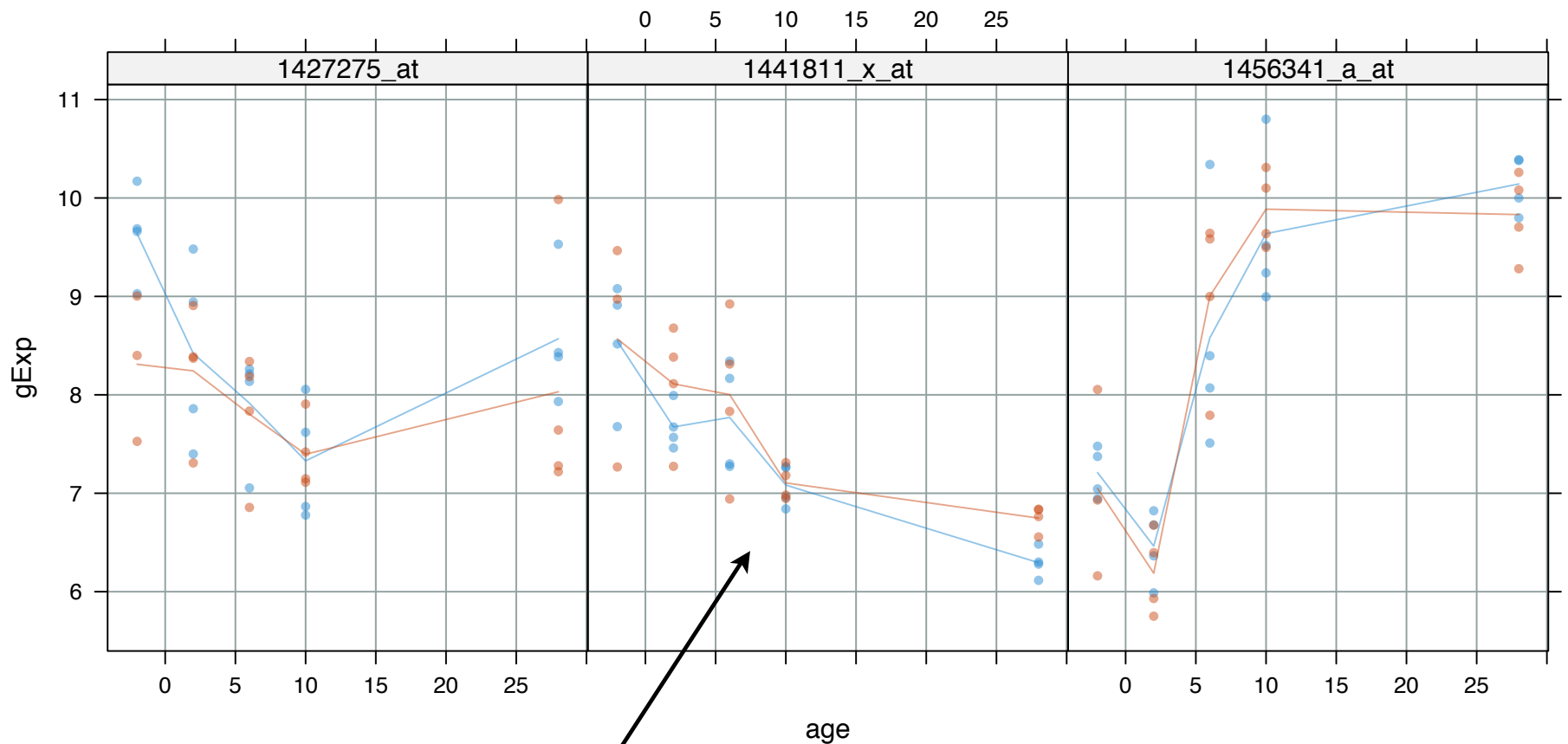
going from quadratic to one-way ANOVA seems justified

increase the complexity ...

| quantitative covariate: age

AND | categorical covariate:

genotype = wt vs. Nrl knockout



let's focus on this one for a model with just intercept and slope, possibly different for wt and Nr1KO

$$y_{ij} = \alpha_{0,wt} + \tau_{0,j} + (\alpha_{1,wt} + \tau_{1,j})age_i + \varepsilon_{ij}$$

where $j \in \{wt, Nr1KO\}$

$i = 1, 2, \dots, n_j$

$$\tau_{0,wt} = \tau_{1,wt} \equiv 0$$

```
> jFit <- lm(gExp ~ gType * age, jDat)
> summary(jFit)
```

Call:

```
lm(formula = gExp ~ gType * age, data = jDat)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.05383	-0.41194	-0.02491	0.31295	1.14417

Coefficients:

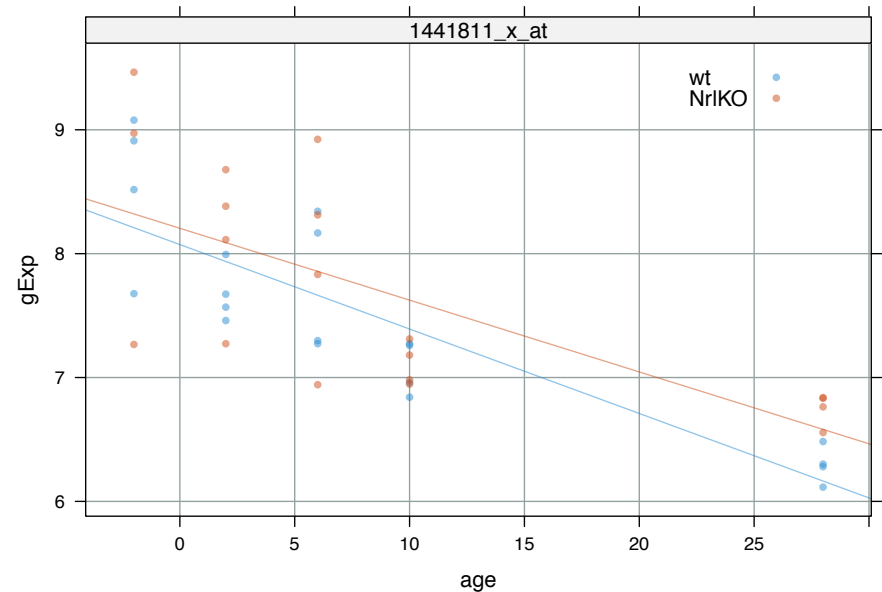
	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	8.07337	0.16552	48.776	< 2e-16	***
gTypeNr1KO	0.13148	0.24070	0.546	0.588	
age	-0.06818	0.01215	-5.612	2.51e-06	***
gTypeNr1KO:age	0.01019	0.01744	0.584	0.563	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5651 on 35 degrees of freedom

Multiple R-squared: 0.607, Adjusted R-squared: 0.5733

F-statistic: 18.02 on 3 and 35 DF, p-value: 3.047e-07



The intercept for the knockouts is:

$$\alpha_{0,wt} + \tau_{0,\Delta Nr1}$$

and the slope for knockouts is:

$$\alpha_{1,wt} + \tau_{1,\Delta Nr1}$$

as always, different parametrizations are possible!

$$y_{ij} = \alpha_{0,j} + \alpha_{1,j}age_i + \varepsilon_{ij}$$

where $j \in \{wt, NrlKO\}$

$i = 1, 2, \dots, n_j$

```
> jFitAlt <- lm(gExp ~ gType/age - 1, jDat)
> summary(jFitAlt)
```

```
Call:
lm(formula = gExp ~ gType/age - 1, data = jDat)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.05383	-0.41194	-0.02491	0.31295	1.14417

Coefficients:

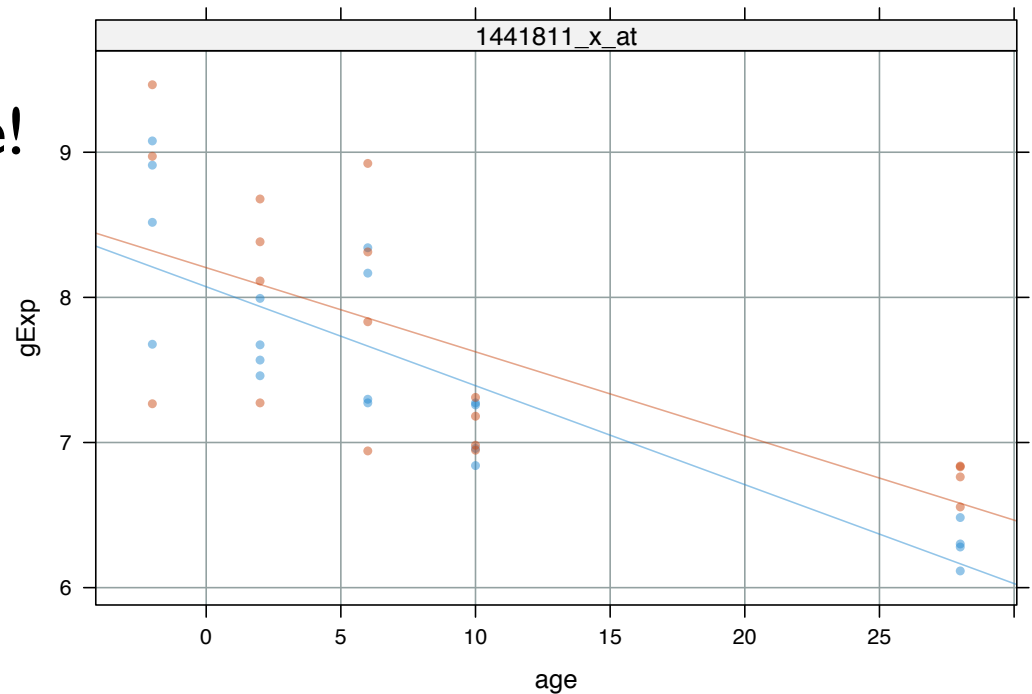
	Estimate	Std. Error	t value	Pr(> t)	
gTypewt	8.07337	0.16552	48.776	< 2e-16	***
gTypeNrlKO	8.20485	0.17476	46.949	< 2e-16	***
gTypewt:age	-0.06818	0.01215	-5.612	2.51e-06	***
gTypeNrlKO:age	-0.05799	0.01251	-4.636	4.80e-05	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5651 on 35 degrees of freedom

Multiple R-squared: 0.9951, Adjusted R-squared: 0.9945

F-statistic: 1761 on 4 and 35 DF, p-value: < 2.2e-16



(intercept, slope) for wild type:

$(\alpha_{0,wt}, \alpha_{1,wt})$

(intercept, slope) for the knockouts:

$(\alpha_{0,\Delta Nrl}, \alpha_{1,\Delta Nrl})$

as always, you can switch between parametrizations via multiplication by an appropriate contrast matrix!

$$y_{ij} = \alpha_{0,wt} + \tau_{0,j} + (\alpha_{1,wt} + \tau_{1,j})age_i + \varepsilon_{ij}$$

where $j \in \{wt, NrlKO\}$

$i = 1, 2, \dots, n_j$

$$\tau_{0,wt} = \tau_{1,wt} \equiv 0$$



$$y_{ij} = \alpha_{0,j} + \alpha_{1,j}age_i + \varepsilon_{ij}$$

where $j \in \{wt, NrlKO\}$

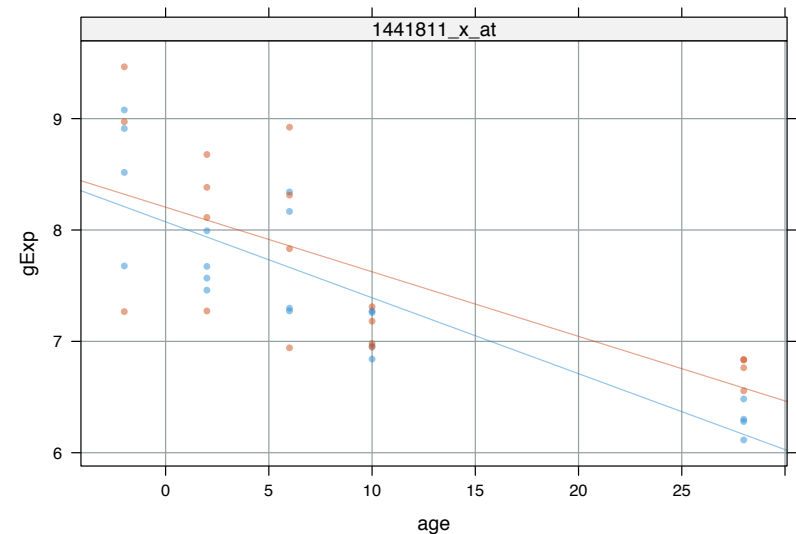
$i = 1, 2, \dots, n_j$

```
> (contMat <- rbind(c(1, 0, 0, 0),
+                   c(1, 1, 0, 0),
+                   c(0, 0, 1, 0),
+                   c(0, 0, 1, 1)))
```

	[,1]	[,2]	[,3]	[,4]
[1,]	1	0	0	0
[2,]	1	1	0	0
[3,]	0	0	1	0
[4,]	0	0	1	1

```
> cbind(coefDefault = coef(jFit),
+       coefAlt = coef(jFitAlt),
+       matrixResult = as.vector(contMat %*% coef(jFit)))
```

	coefDefault	coefAlt	matrixResult
(Intercept)	8.07337352	8.07337352	8.07337352
gTypeNrlKO	0.13147574	8.20484926	8.20484926
age	-0.06817881	-0.06817881	-0.06817881
gTypeNrlKO:age	0.01018928	-0.05798953	-0.05798953



as always, you can assess the relevance of several terms at once -- such as everything involving genotype -- with an F test

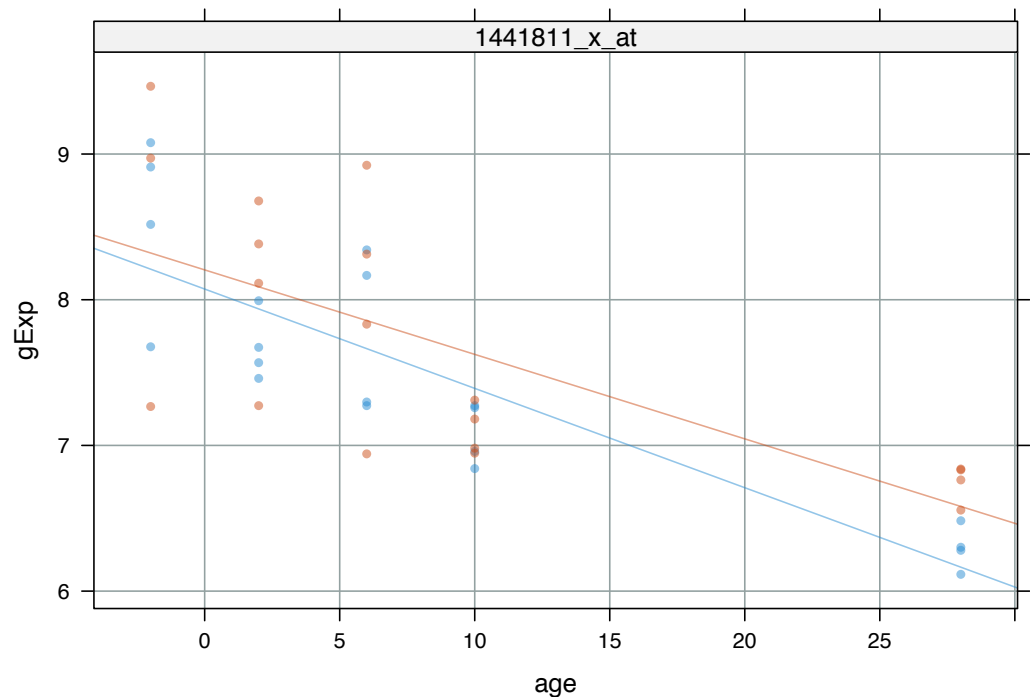
it's not clear that genotype affects the intercept or the slope

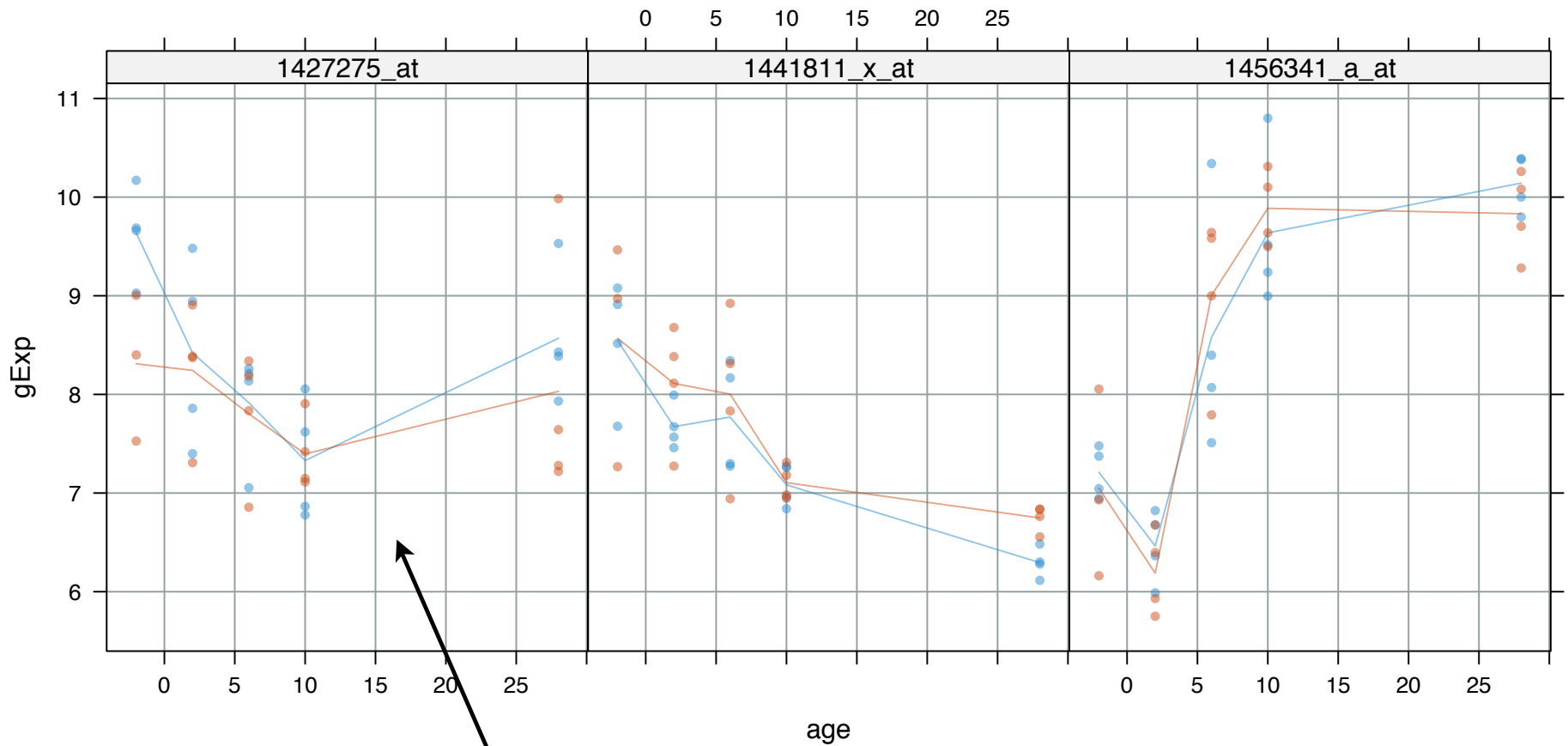
```
small      big
> anova(lm(gExp ~ age, jDat), jFit)
```

Analysis of Variance Table

```
Model 1: gExp ~ age
Model 2: gExp ~ gType * age
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	37	11.774				
2	35	11.176	2	0.59807	0.9365	0.4016





let's focus here for a model including a quadratic age term

$$y_{ij} = \alpha_{0,wt} + \tau_{0,j} + (\alpha_{1,wt} + \tau_{1,j})age_i + (\alpha_{2,wt} + \tau_{2,j})age_i^2 + \varepsilon_{ij}$$

where $j \in \{wt, NrlKO\}$

$i = 1, 2, \dots, n_j$

$$\tau_{0,wt} = \tau_{1,wt} = \tau_{2,wt} \equiv 0$$

```
> summary(jFit)
```

Call:

```
lm(formula = gExp ~ gType * (age + I(age^2)), data = jDat)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.16275	-0.55816	0.08203	0.42020	1.96803

Coefficients:

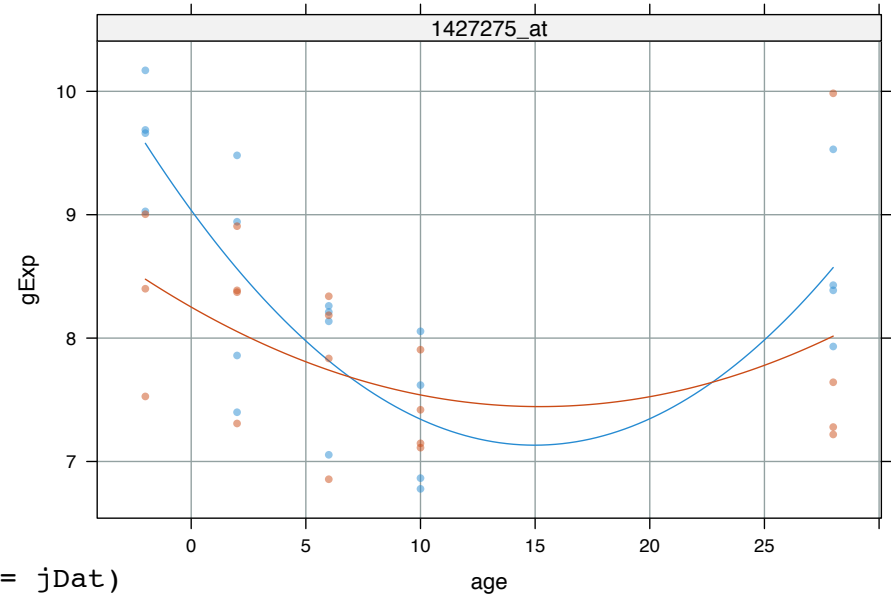
	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	9.036401	0.234853	38.477	< 2e-16	***
gTypeNrlKO	-0.784969	0.350249	-2.241	0.0319	*
age	-0.254305	0.053234	-4.777	3.55e-05	***
I(age^2)	0.008490	0.001838	4.620	5.63e-05	***
gTypeNrlKO:age	0.148195	0.078232	1.894	0.0670	.
gTypeNrlKO:I(age^2)	-0.005001	0.002673	-1.871	0.0702	.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7128 on 33 degrees of freedom

Multiple R-squared: 0.4755, Adjusted R-squared: 0.3961

F-statistic: 5.984 on 5 and 33 DF, p-value: 0.0004804



as always, you can assess the relevance of several terms at once -- such as everything involving genotype -- with an F test

borderline evidence that genotype affects something about the parabola (location or shape)

small

```
> anova(lm(gExp ~ age + I(age^2), jDat),  
+       lm(gExp ~ gType * (age + I(age^2)), jDat))
```

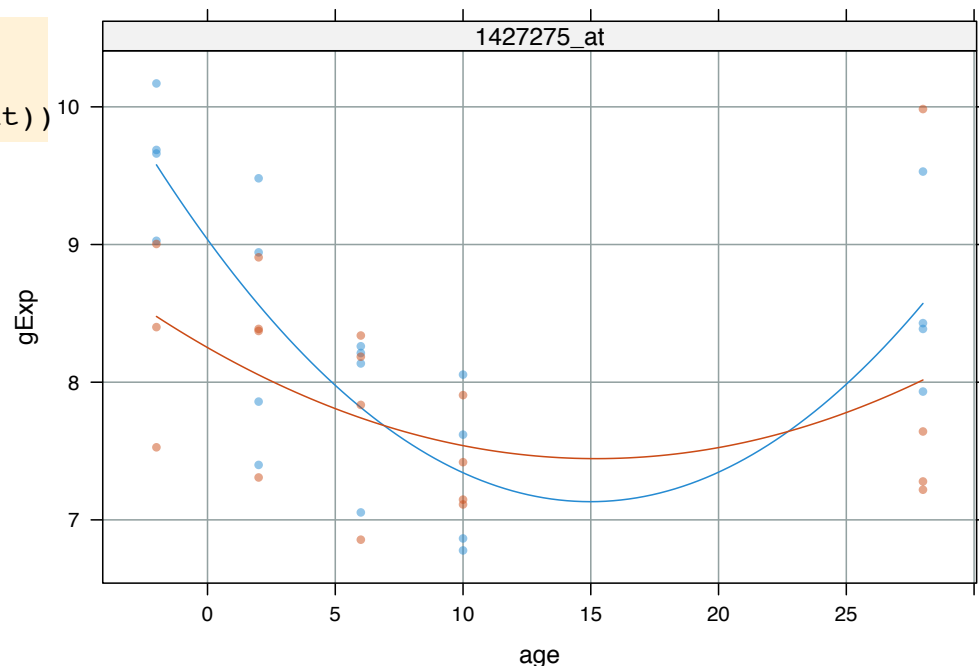
big

Analysis of Variance Table

Model 1: $\text{gExp} \sim \text{age} + \text{I}(\text{age}^2)$

Model 2: $\text{gExp} \sim \text{gType} * (\text{age} + \text{I}(\text{age}^2))$

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	36	20.081				
2	33	16.767	3	3.3144	2.1744	0.1097



linear model framework is extremely general!

one extreme (simple): two-sample common variance t-test

another extreme (flexible): a polynomial, potentially different for each level of some factor

dichotomous variable? OK!

categorical variable? OK!

quantitative variable? OK!

various combinations of the above? OK!

don't be afraid to build models with more than 1 covariate

don't be intimidated by all the “contrast” talk

`lm(yMat ~ x)`

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

$$\begin{bmatrix} y_{11} & \dots & y_{1G} \\ y_{21} & & y_{2G} \\ \vdots & & \\ y_{n1} & & y_{nG} \end{bmatrix} = X \begin{bmatrix} \alpha_1 & \dots & \alpha_G \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} & \dots & \varepsilon_{1G} \\ \varepsilon_{21} & & \varepsilon_{2G} \\ \vdots & & \\ \varepsilon_{n1} & & \varepsilon_{nG} \end{bmatrix}$$

built-in function `lm()` can do “multivariate regression” = many dependent vars (“responses”)
aka “multivariate multiple regression”

From `lm()` documentation:

If response is a matrix a linear model is fitted separately by least-squares to each column of the matrix.

`lm` returns an object of class “lm” or for multiple responses of class `c("mlm", "lm")`.

Industrial scale model fitting is good because things like this are not recomputed 30K times unnecessarily*

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

$\hat{\alpha} = (X^T X)^{-1} X^T Y$ the MLE and OLS estimator of α

$\hat{\sigma}^2 = \frac{1}{n-p} \hat{\varepsilon}^T \hat{\varepsilon}$ the estimated error variance

$\hat{V}(\hat{\alpha}) = \hat{\sigma}^2 (X^T X)^{-1}$ the estimated covariance matrix of $\hat{\alpha}$

How test $H_0 : \alpha_j = 0$?

With a t-statistic. Under H_0 , we have (at least approximately) that:

$$\frac{\hat{\alpha}_j}{\widehat{se}(\hat{\alpha}_j)} \sim t_{n-p}$$

so a p-value is obtained by computing a tail probability for the observed value of $\hat{\alpha}_j$ from a t_{n-p} distribution.

* under the hood, `lm()` is doing something more clever and numerically stable than this

I have fit **all** the models we've considered to all ~30K probesets.

Let's examine some of the results *en masse*.

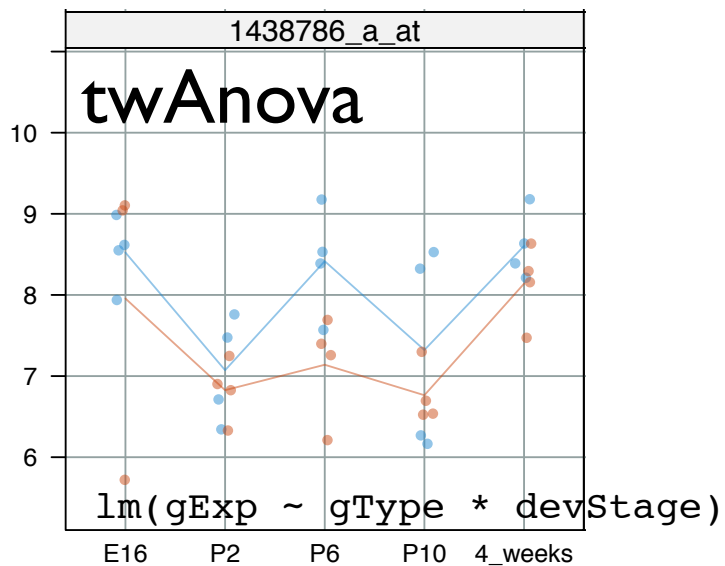
Let this drive home the point that ...

- background variability
- intercepts
- Nrl knockout effects
- devStage effects
- age effects, both linear and quadratic
- and interactions of all the above

differ for each gene.

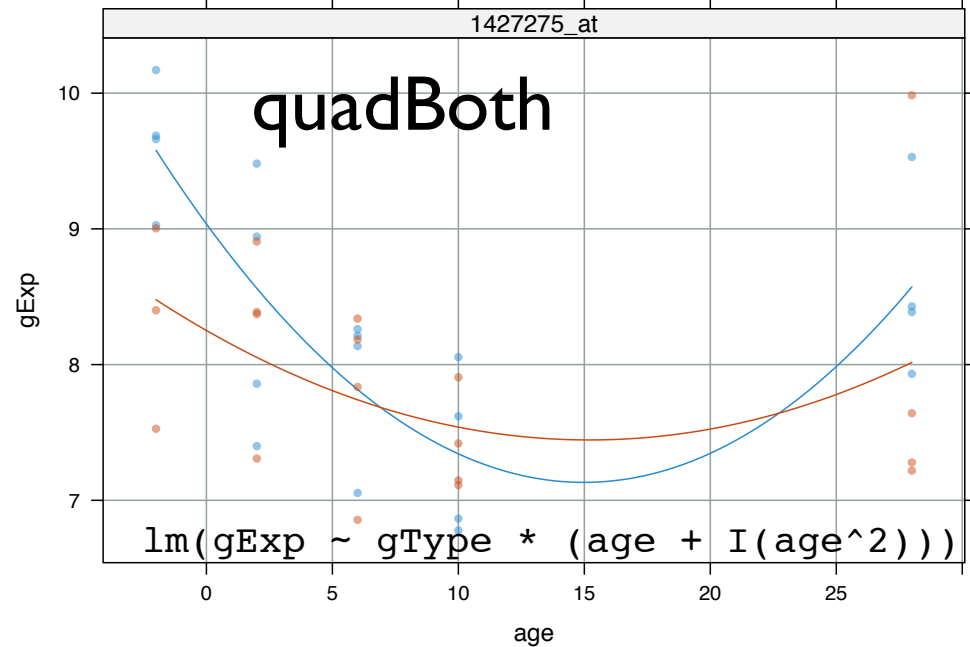
1438786_a_at

twAnova



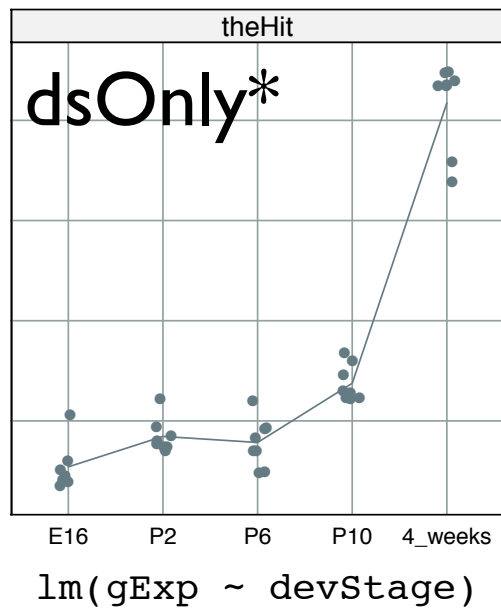
1427275_at

quadBoth



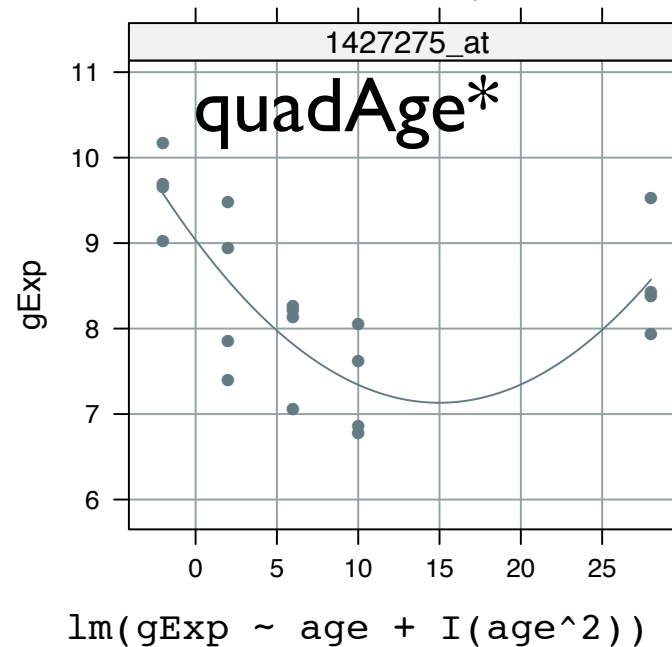
theHit

dsOnly*

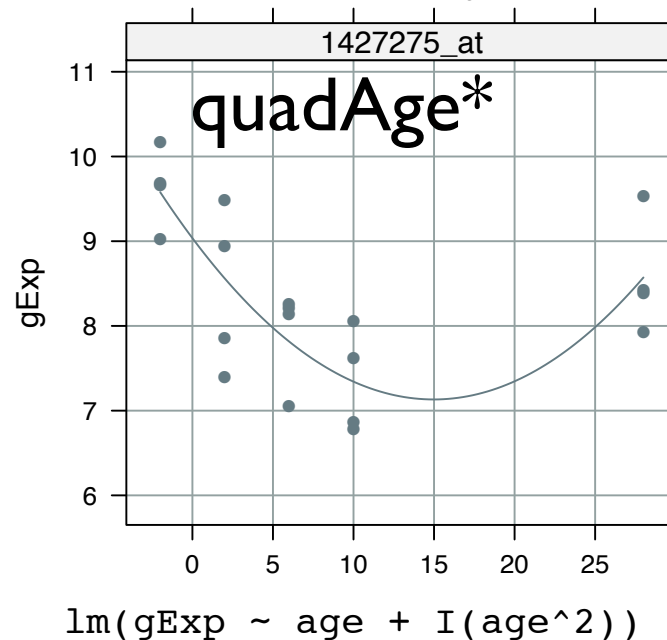
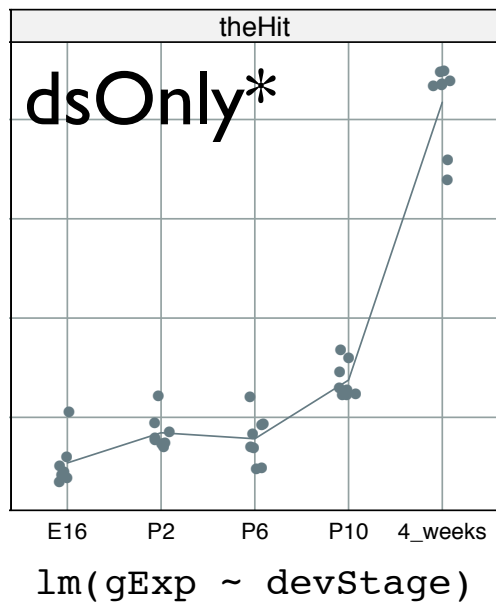
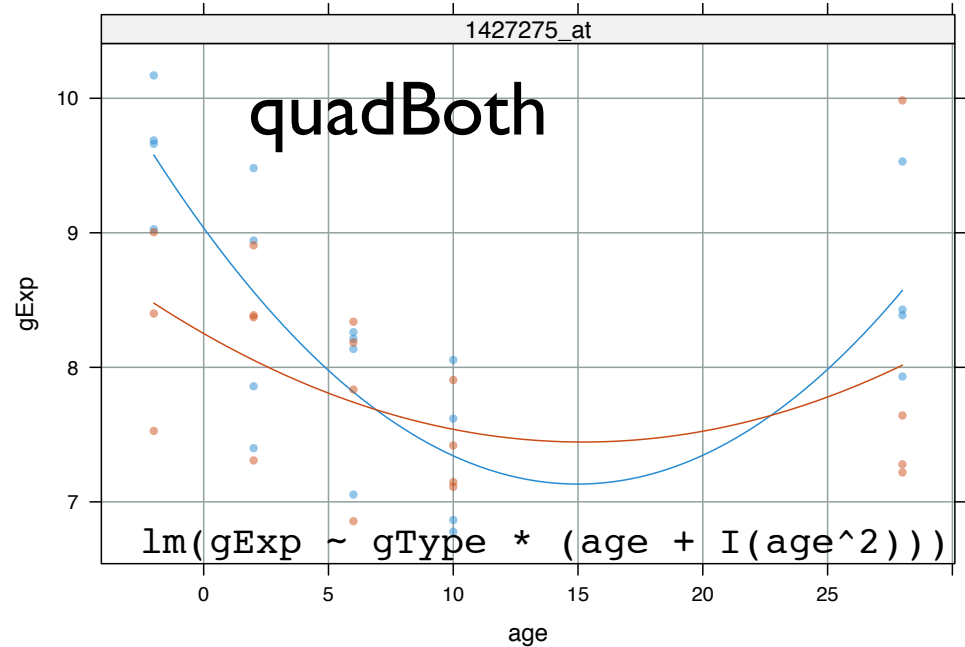
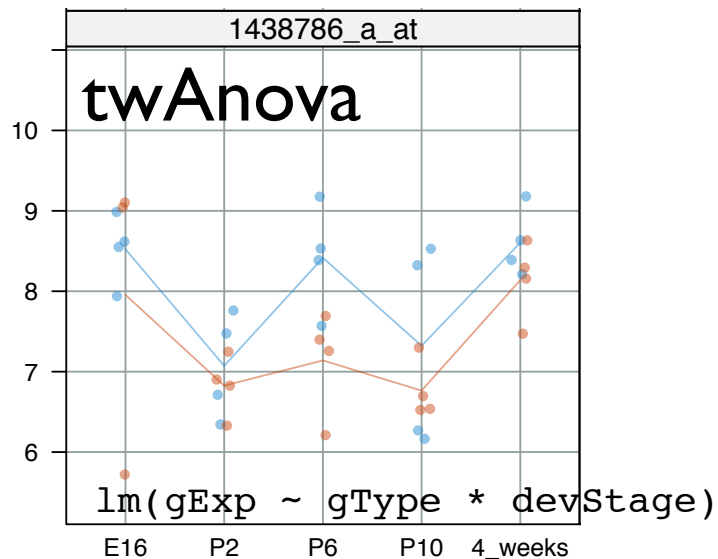


1427275_at

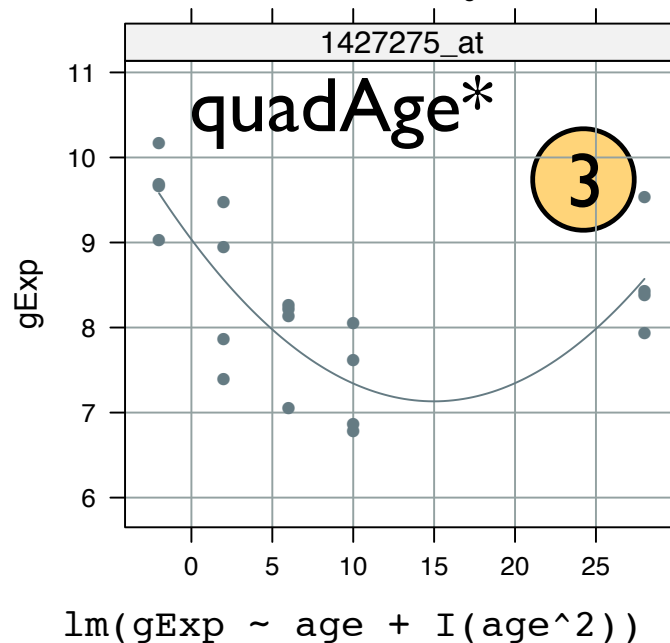
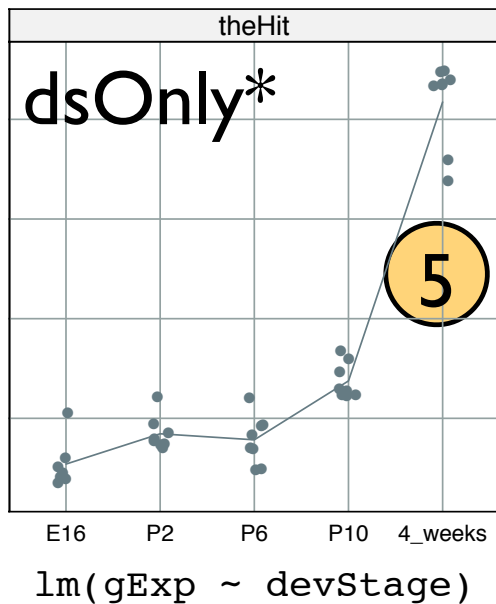
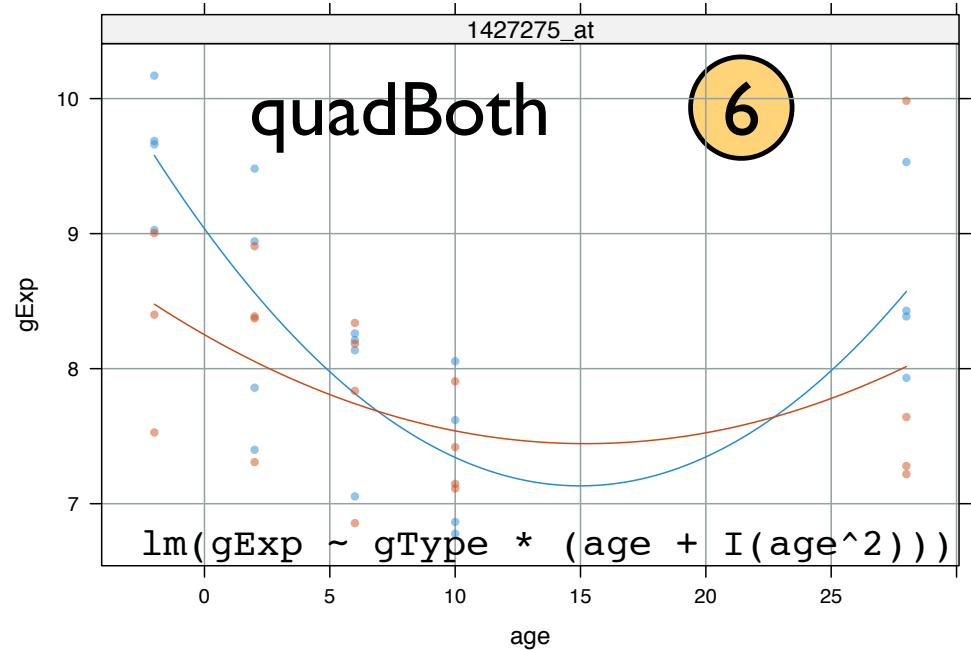
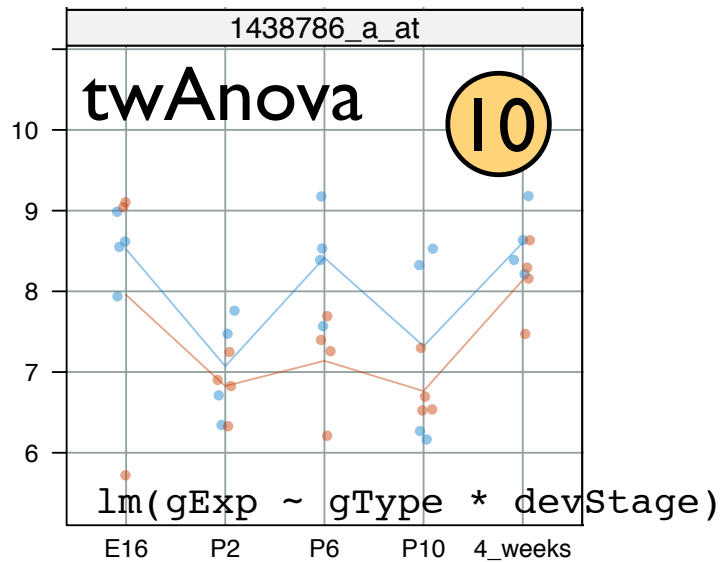
quadAge*



* Figures slightly misleading. Model is fit to all the data, wild type and Nrl knockout, but gType is not used as a covariate.



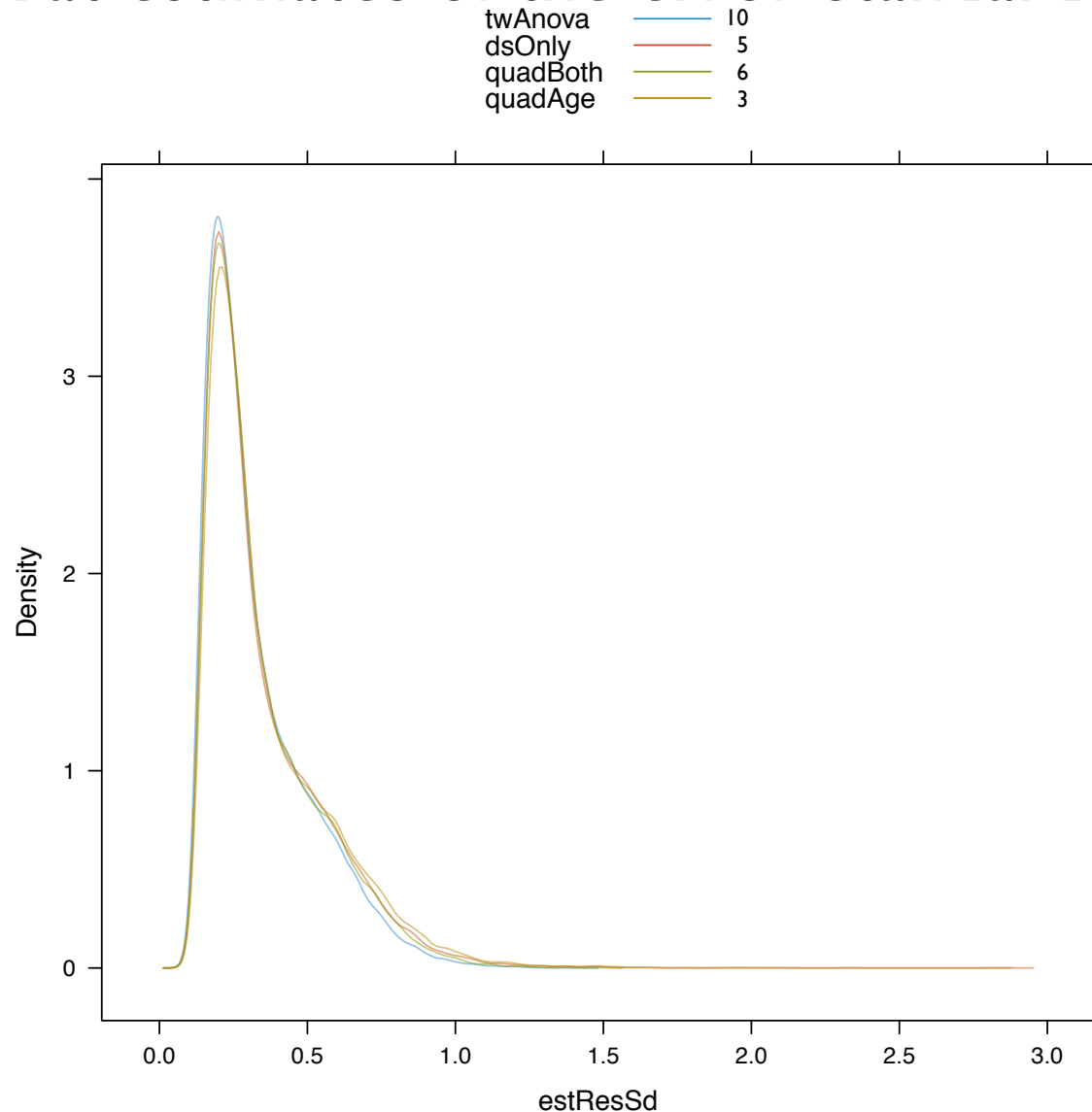
How “big” are these models? How many parameters are we using to specify the mean structure?



How “big” are these models? How many parameters are we using to specify the mean structure?

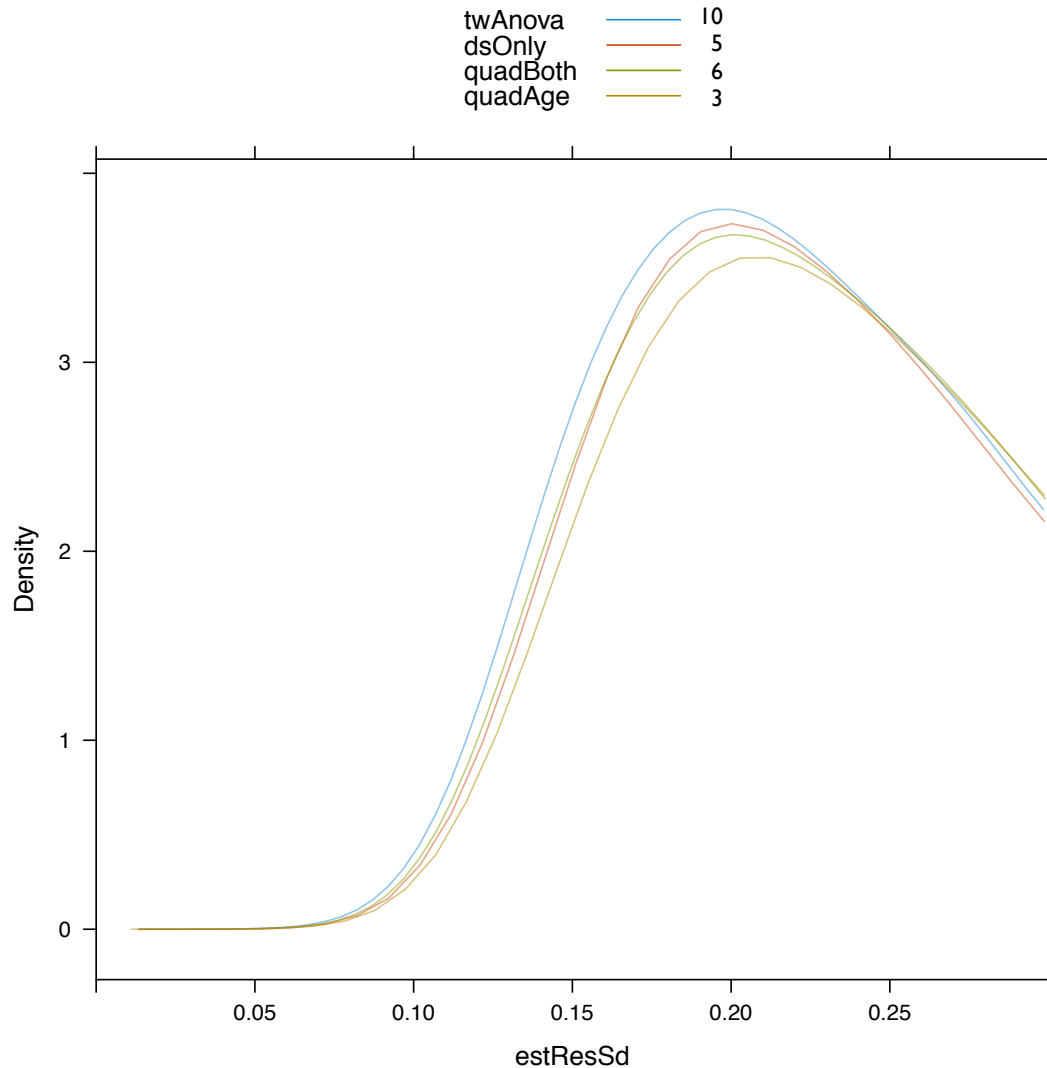
$$y_i = f(x_i; \alpha) + \varepsilon_i, \text{var}(\varepsilon) = \sigma^2$$

Let's look at estimates of the error standard deviation.



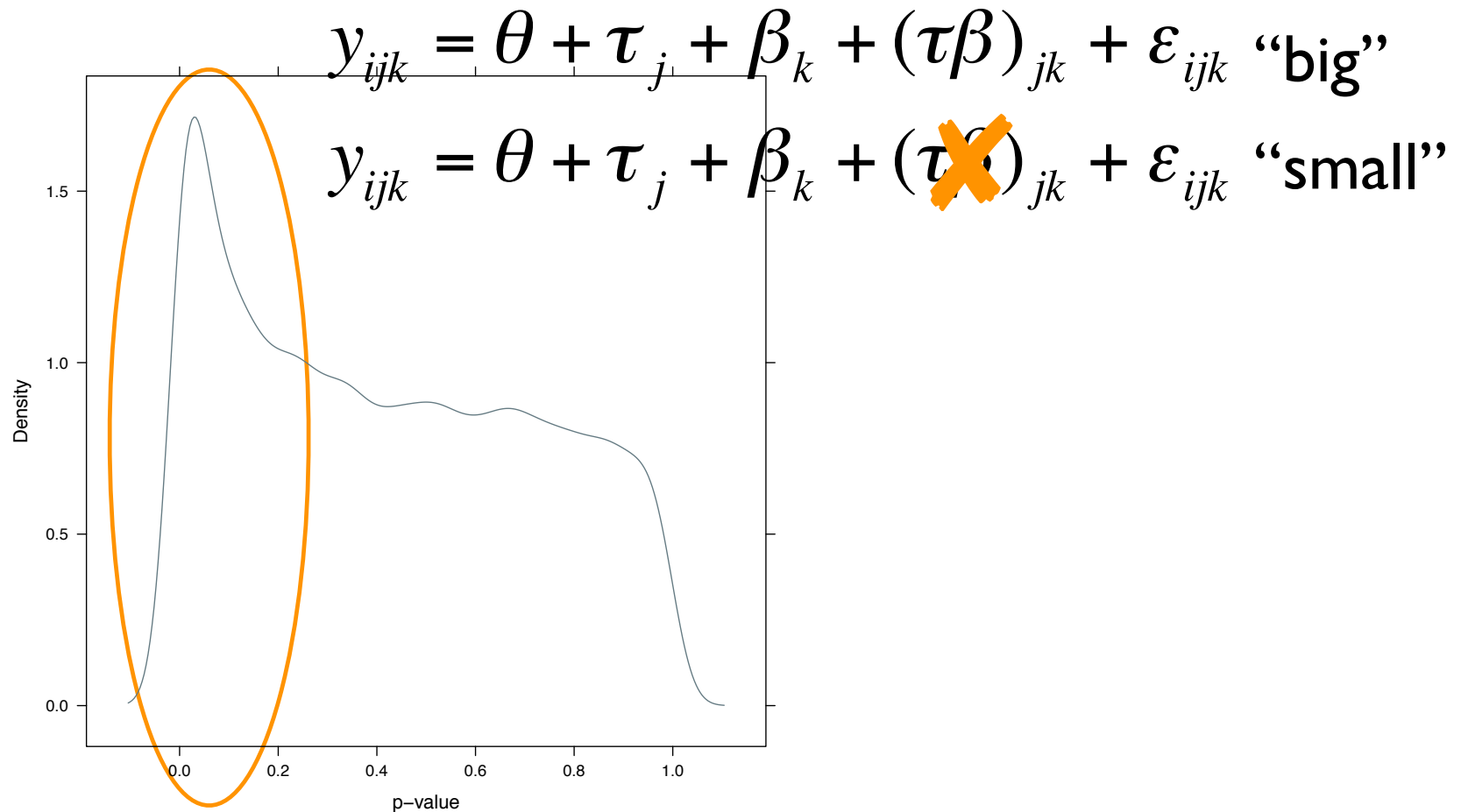
$$y_i = f(x_i; \alpha) + \varepsilon_i, \text{var}(\varepsilon) = \sigma^2$$

Let's look at estimates of the error standard deviation.



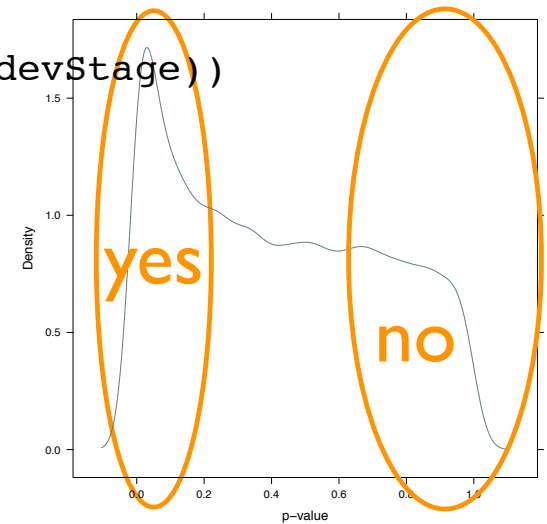
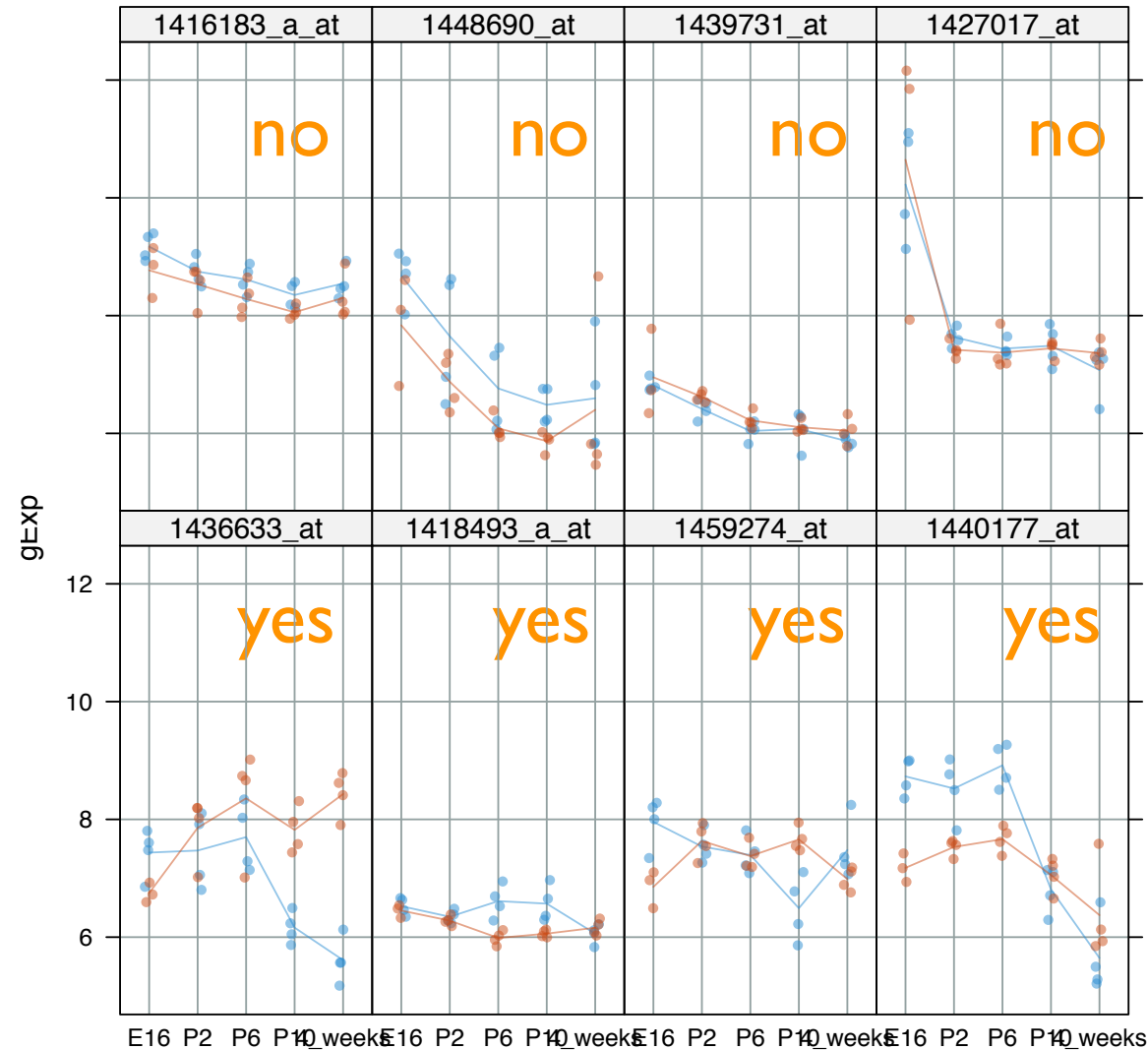
In the two-way ANOVA model, is there evidence for gType * devStage interaction? YES.

```
## this code is fictional but conveys the point  
anova(lm(gExp ~ gType * devStage), lm(gExp ~ gType + devStage))  
## inspecting the p-values from these F tests
```



```
## this code is fictional but conveys the point
anova(lm(gExp ~ gType * devStage), lm(gExp ~ gType + devStage))
## inspecting the p-values from these F tests
```

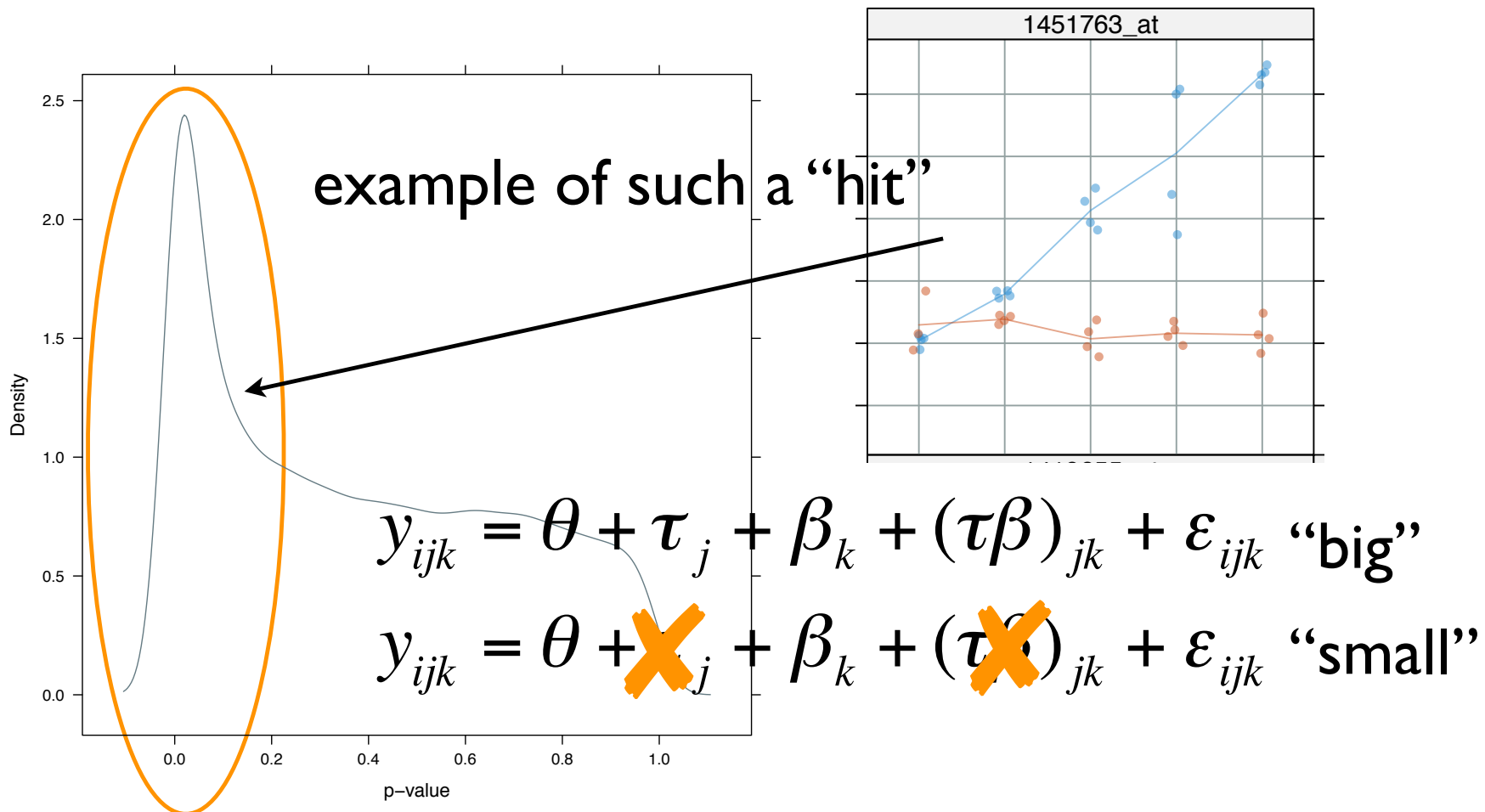
wt •
Nr1KO •



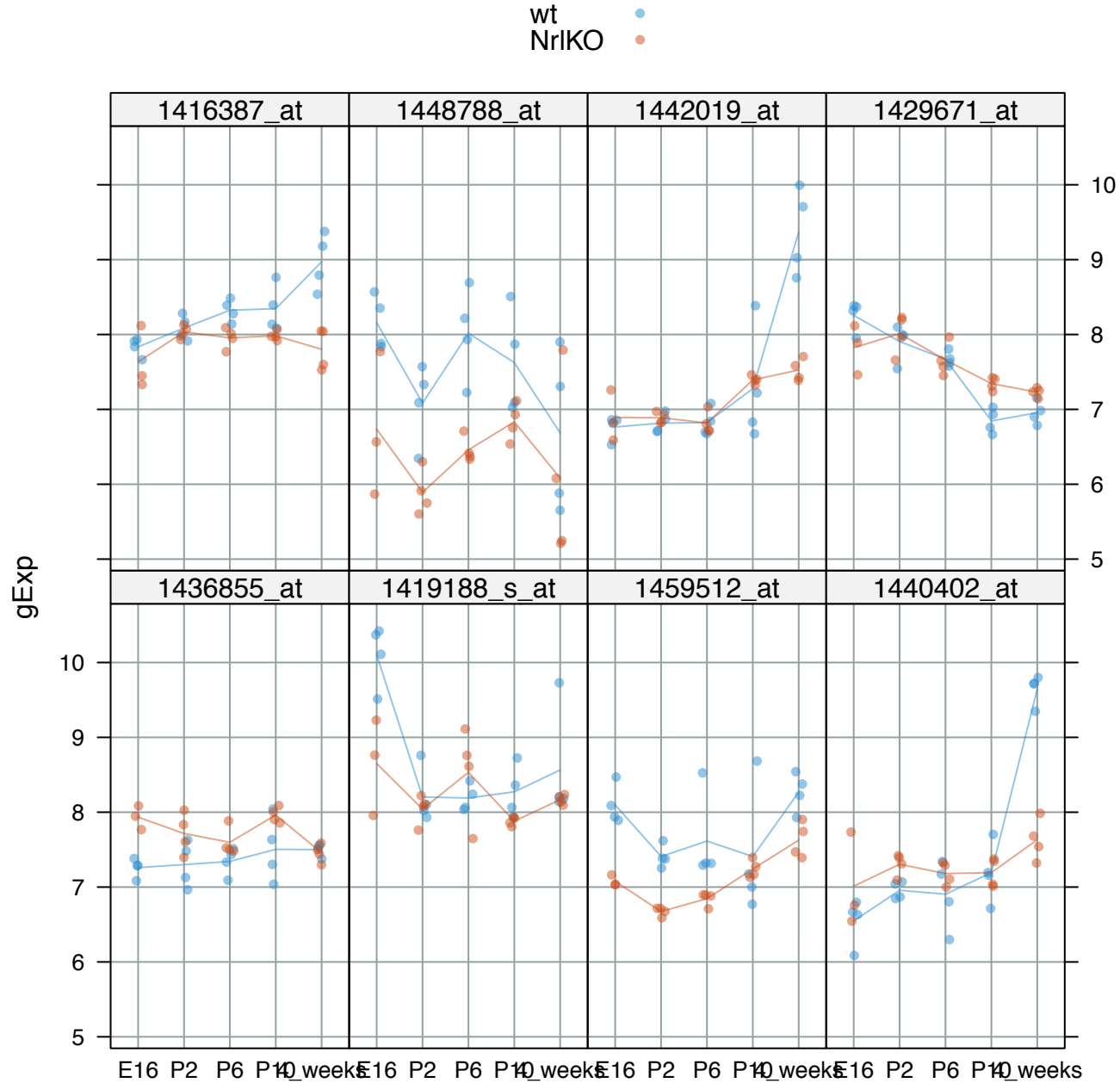
interaction?

In the two-way ANOVA model, is there evidence that genotype matters? YES.

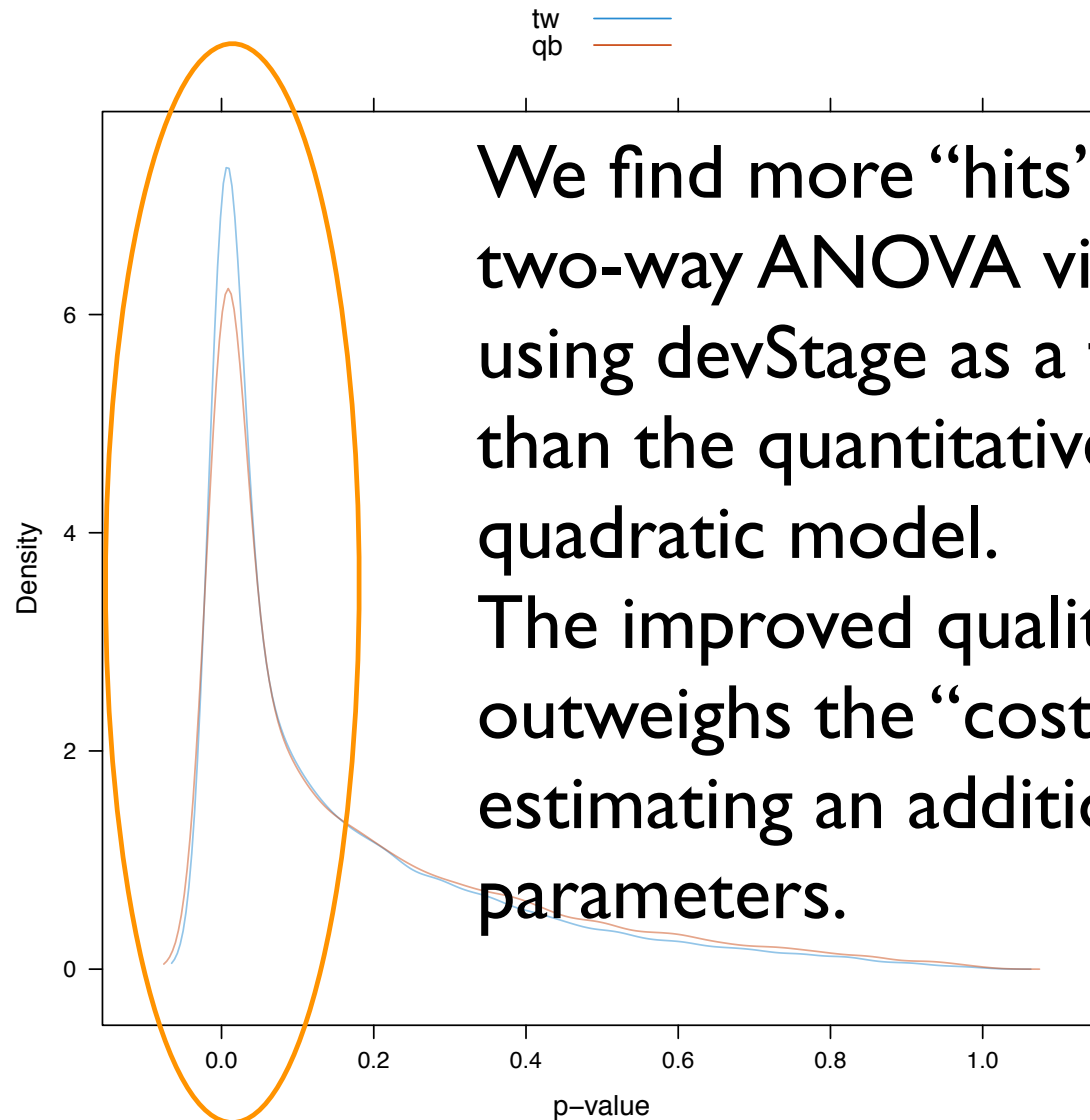
```
## this code is fictional but conveys the point  
anova(lm(gExp ~ gType * devStage), lm(gExp ~ devStage))  
## inspecting the p-values from these F tests
```



more “gType” hits within the ANOVA models



Looking at evidence of any differential expression at all (overall F test) in the two-way ANOVA model vs. the quadratic.



We find more “hits” with the two-way ANOVA viewpoint, i.e. using devStage as a factor rather than the quantitative age and a quadratic model. The improved quality of fit outweighs the “cost” of estimating an additional 4 parameters.

where to next? ... **Next lecture:**

in many studies, the # replicates is small relative to #
params being estimated

can lead to crazy small estimates of error variance
which leads to crazy large test statistics
which leads to crazy small p-values
which leads to “hits” where the observed phenomenon is
rather subtle

which leads to people saying the platform and/or analysis
method and/or analyst is bad

moderating the variance estimates can be very helpful -->
limma!

where to next? ... following 'next lecture:

multiple testing, large scale inference

analysis of high-throughput data results in thousands of “gene-wise” hypothesis tests

often, “gene-wise” analysis is relatively simple

BUT a recurring and thorny issue is how to handle thousands of p-values, each for a separate hypothesis test

how to guard against crazy # false positives?

which error rate is more relevant ... rate at which null genes are ‘discovered’ or rate at which ‘discoveries’ are null?