Linear Models for Analysis and Prediction

February 2-7, 2022

Common statistical tests

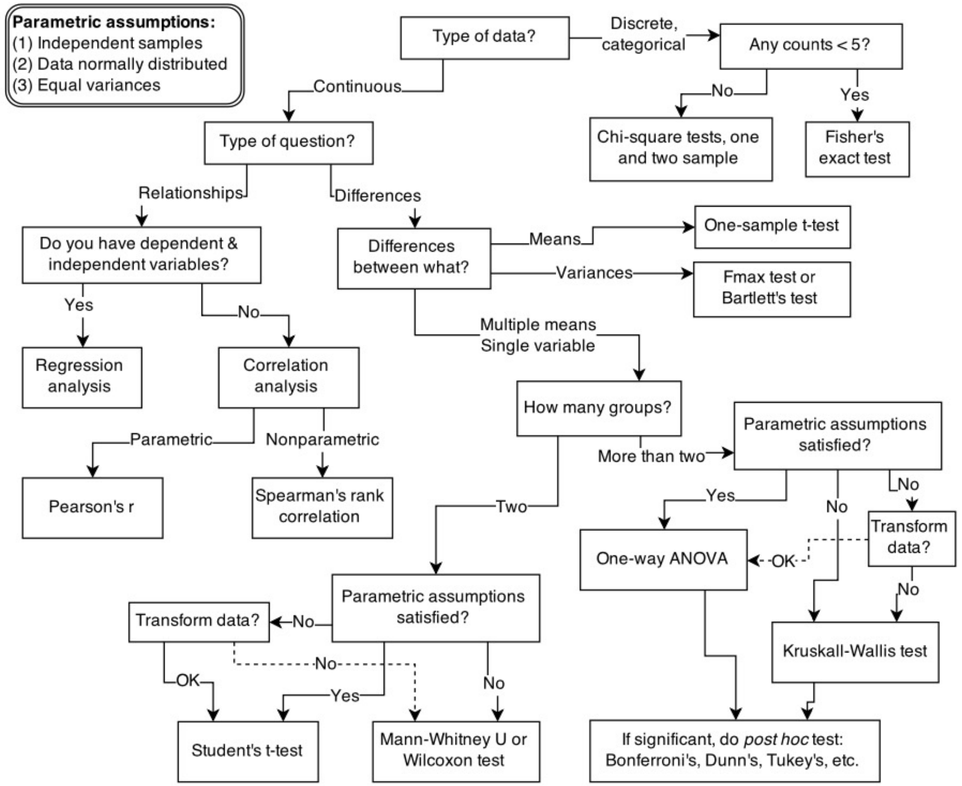
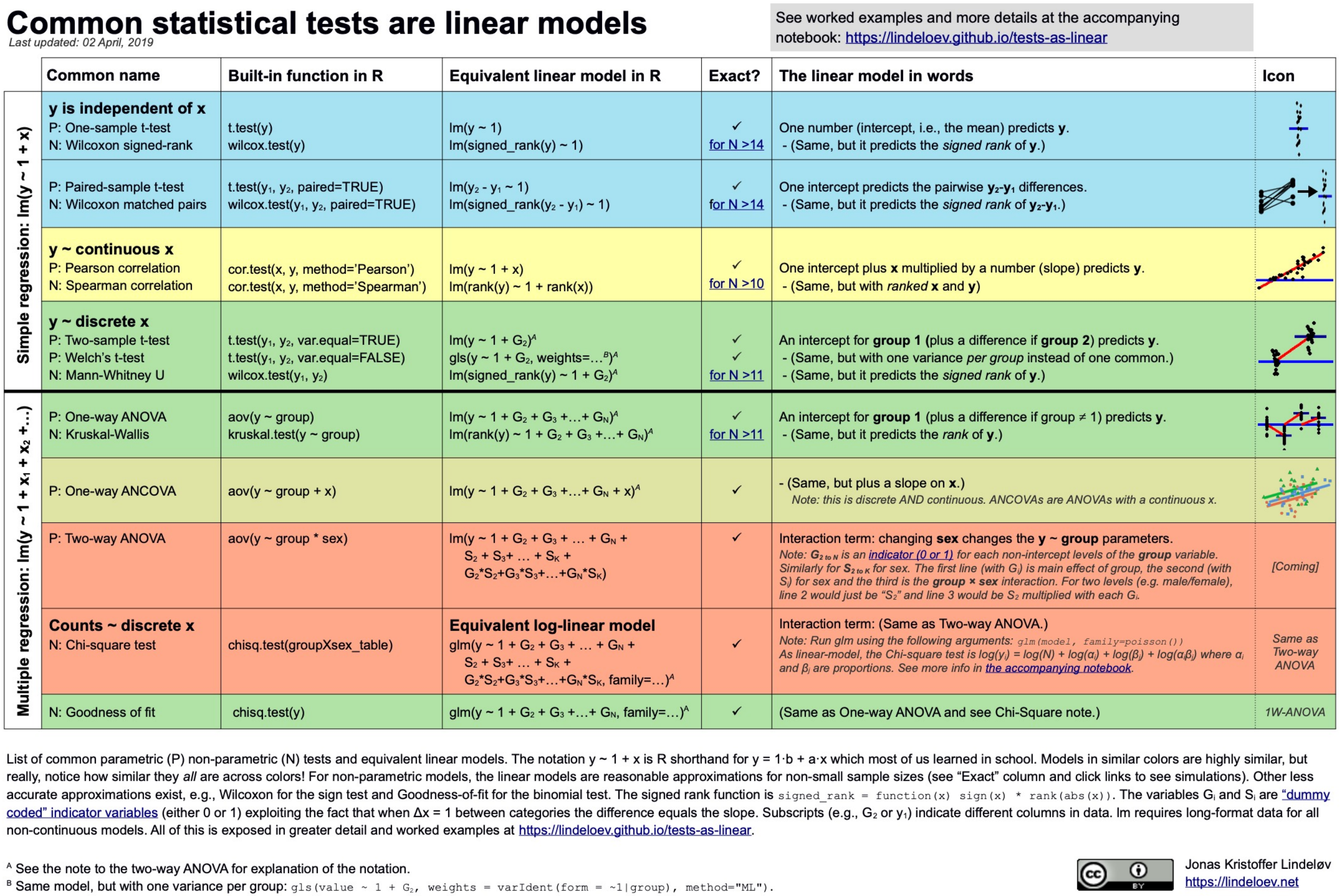


FIGURE 1.1. Example decision tree, or flowchart, for selecting an appropriate statistical procedure. Beginning at the top, the user answers a series of questions about measurement and intent, arriving eventually at the name of a procedure. Many such decision trees are possible.

McElreath (2020) *Statistical Rethinking* (2nd ed)



Common statistical tests are linear models

Last updated: 02 April, 2019

See worked examples and more details at the accompanying notebook: https://lindeloev.github.io/tests-as-linear

	Common name	Built-in function in R	Equivalent linear model in R	Exact?	The linear model in words	Icon

Simple regression: lm(y ~ 1 + x)	**y is independent of x**
	P: One-sample t-test
	N: Wilcoxon signed-rank
		t.test(y)
		wilcox.test(y)
			lm(y ~ 1)
			lm(signed_rank(y) ~ 1)
				✓
				for N >14
					One number (intercept, i.e., the mean) predicts **y**.
					- (Same, but it predicts the *signed rank* of **y**.)	

	P: Paired-sample t-test
	N: Wilcoxon matched pairs
		t.test(y_1, y_2, paired=TRUE)
		wilcox.test(y_1, y_2, paired=TRUE)
			lm(y_2 - y_1 ~ 1)
			lm(signed_rank(y_2 - y_1) ~ 1)
				✓
				for N >14
					One intercept predicts the pairwise **y_2-y_1** differences.
					- (Same, but it predicts the *signed rank* of **y_2-y_1**.)	

	y ~ continuous x
	P: Pearson correlation
	N: Spearman correlation
		cor.test(x, y, method='Pearson')
		cor.test(x, y, method='Spearman')
			lm(y ~ 1 + x)
			lm(rank(y) ~ 1 + rank(x))
				✓
				for N >10
					One intercept plus **x** multiplied by a number (slope) predicts **y**.
					- (Same, but with *ranked* **x** and **y**)	

	y ~ discrete x
	P: Two-sample t-test
	P: Welch's t-test
	N: Mann-Whitney U
		t.test(y_1, y_2, var.equal=TRUE)
		t.test(y_1, y_2, var.equal=FALSE)
		wilcox.test(y_1, y_2)
			lm(y ~ 1 + G_2)^A
			gls(y ~ 1 + G_2, weights=...^{B})^A
			lm(signed_rank(y) ~ 1 + G_2)^A
				✓
				✓
				for N >11
					An intercept for **group 1** (plus a difference if **group 2**) predicts **y**.
					- (Same, but with one variance *per group* instead of one common.)
					- (Same, but it predicts the *signed rank* of **y**.)	

Multiple regression: lm(y ~ 1 + x_1 + x_2 +...)	P: One-way ANOVA
	N: Kruskal-Wallis
		aov(y ~ group)
		kruskal.test(y ~ group)
			lm(y ~ 1 + G_2 + G_3 +...+ G_N)^A
			lm(rank(y) ~ 1 + G_2 + G_3 +...+ G_N)^A
				✓
				for N >11
					An intercept for **group 1** (plus a difference if group ≠ 1) predicts **y**.
					- (Same, but it predicts the *rank* of **y**.)	

	P: One-way ANCOVA	aov(y ~ group + x)	lm(y ~ 1 + G_2 + G_3 +...+ G_N + x)^A	✓	- (Same, but plus a slope on **x**.) *Note: this is discrete AND continuous. ANCOVAs are ANOVAs with a continuous x.*	
	P: Two-way ANOVA	aov(y ~ group * sex)	lm(y ~ 1 + G_2 + G_3 + ... + G_N + S_2 + S_3+ ... + S_K + G_2*S_2+G_3*S_3+...+G_N*S_K)	✓	Interaction term: changing **sex** changes the **y ~ group** parameters. *Note: **$G_{2 \text{ to } N}$** is an indicator (0 or 1) for each non-intercept levels of the **group** variable. Similarly for **$S_{2 \text{ to } K}$** for sex. The first line (with G_i) is main effect of group, the second (with S_i) for sex and the third is the **group × sex** interaction. For two levels (e.g. male/female), line 2 would just be "S_2" and line 3 would be S_2 multiplied with each G_i.*	*[Coming]*
	Counts ~ discrete x N: Chi-square test	chisq.test(groupXsex_table)	**Equivalent log-linear model** glm(y ~ 1 + G_2 + G_3 + ... + G_N + S_2 + S_3+ ... + S_K + G_2*S_2+G_3*S_3+...+G_N*S_K, family=...)^A	✓	Interaction term: (Same as Two-way ANOVA.) *Note: Run glm using the following arguments:* `glm(model, family=poisson())` *As linear-model, the Chi-square test is $\log(y_i) = \log(N) + \log(\alpha_i) + \log(\beta_j) + \log(\alpha_i\beta_j)$ where α_i and β_j are proportions. See more info in the accompanying notebook.*	*Same as Two-way ANOVA*
	N: Goodness of fit	chisq.test(y)	glm(y ~ 1 + G_2 + G_3 +...+ G_N, family=...)^A	✓	(Same as One-way ANOVA and see Chi-Square note.)	*1W-ANOVA*

List of common parametric (P) non-parametric (N) tests and equivalent linear models. The notation y ~ 1 + x is R shorthand for y = 1·b + a·x which most of us learned in school. Models in similar colors are highly similar, but really, notice how similar they *all* are across colors! For non-parametric models, the linear models are reasonable approximations for non-small sample sizes (see "Exact" column and click links to see simulations). Other less accurate approximations exist, e.g., Wilcoxon for the sign test and Goodness-of-fit for the binomial test. The signed rank function is `signed_rank = function(x) sign(x) * rank(abs(x))`. The variables G_i and S_i are "dummy coded" indicator variables (either 0 or 1) exploiting the fact that when Δx = 1 between categories the difference equals the slope. Subscripts (e.g., G_2 or y_1) indicate different columns in data. lm requires long-format data for all non-continuous models. All of this is exposed in greater detail and worked examples at https://lindeloev.github.io/tests-as-linear.

[A] See the note to the two-way ANOVA for explanation of the notation.

[B] Same model, but with one variance per group: `gls(value ~ 1 + G₂, weights = varIdent(form = ~1|group), method="ML")`.

Jonas Kristoffer Lindeløv
https://lindeloev.net

https://lindeloev.github.io/tests-as-linear

Regression

$$y_i = \beta_0 + \ \beta_1 x_{1i} + \varepsilon_i$$

Regression

$$y_i = \beta_0 + \ \beta_1 x_{1i} + \varepsilon_i$$

"outcome" variable
"dependent" variable
"criterion" variable

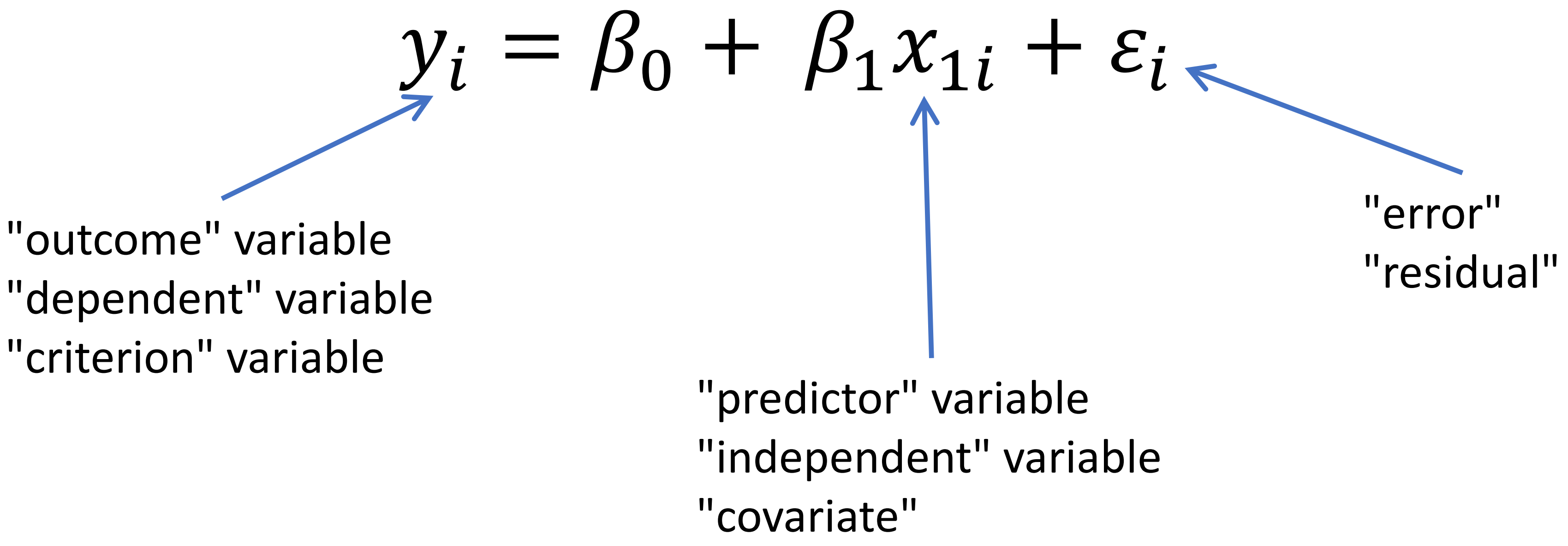
Regression

$$y_i = \beta_0 + \ \beta_1 x_{1i} + \varepsilon_i$$

"outcome" variable
"dependent" variable
"criterion" variable

"predictor" variable
"independent" variable
"covariate"

Regression



$$y_i = \beta_0 + \ \beta_1 x_{1i} + \varepsilon_i$$

"outcome" variable
"dependent" variable
"criterion" variable

"predictor" variable
"independent" variable
"covariate"

"error"
"residual"

- assume that relationship is linear and observations are independent
- assume that residuals are distributed normally with mean 0 and constant variance
- variability of error does not depend on value of x (assumption of homoscedasticity)

Regression

The choice of values for β_0 and β_1 is such that the **sum of squared** differences between the actual and predicted y values is minimized

$$y_i = \beta_0 + \beta_1 x_{1i} + \varepsilon_i$$

"outcome" variable
"dependent" variable
"criterion" variable

"predictor" variable
"independent" variable
"covariate"

"error"
"residual"

- assume that relationship is <u>linear</u> and observations are <u>independent</u>
- assume that residuals are distributed normally with mean 0 and constant variance
- variability of error does not depend on value of x (assumption of homoscedasticity)

Regression

The choice of values for β_0 and β_1 is such that the **sum of squared** differences between the actual and predicted y values is minimized

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \cdots + \varepsilon_i$$

- assume that relationship is linear and observations are independent
- assume that residuals are distributed normally with mean 0 and constant variance
- variability of error does not depend on value of x (assumption of homoscedasticity)

Regression assumptions

- Your data do not have to be normal but you are assuming that the errors/residuals are (multivariate) normal – you don't have more or less precise predictions depending on on the values of your predictors

- If you have multiple predictors, the analysis breaks down if the predictors are highly correlated with each other (multicollinearity)
  - If one is a linear combination of the others the model will just fail

- If observations are not independent, your estimates of the variance of the residuals, which impacts the precision or your parameter estimates, are biased

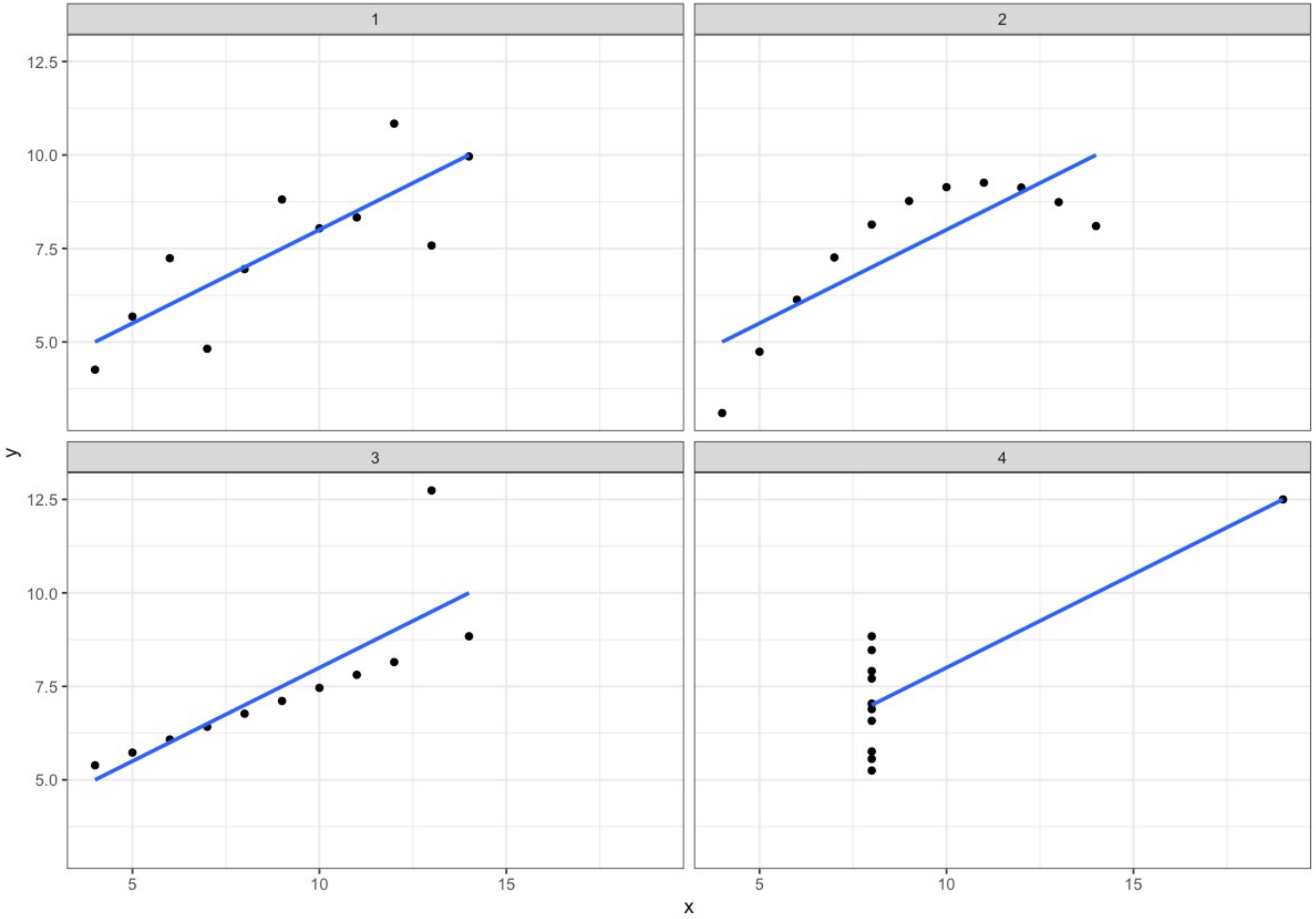
Checking assumptions

- graph data

- "Anscombe's quartet"

Checking assumptions

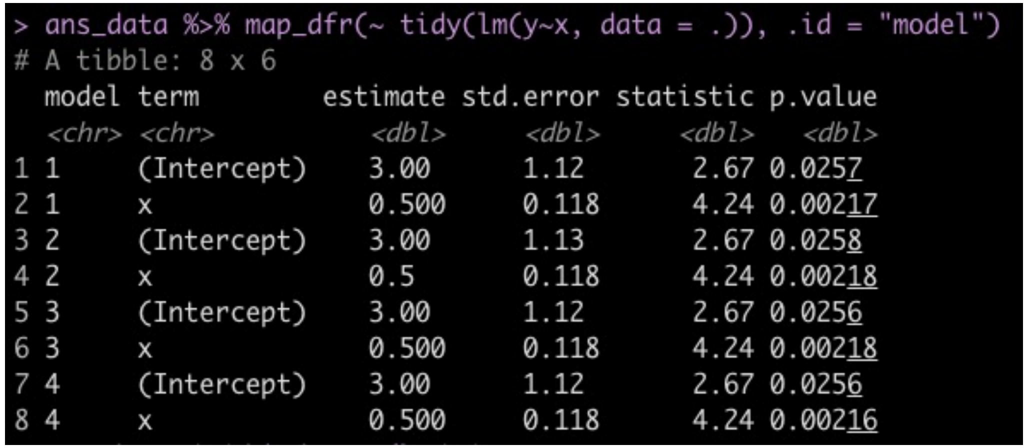
- graph data
- "Anscombe's quartet"



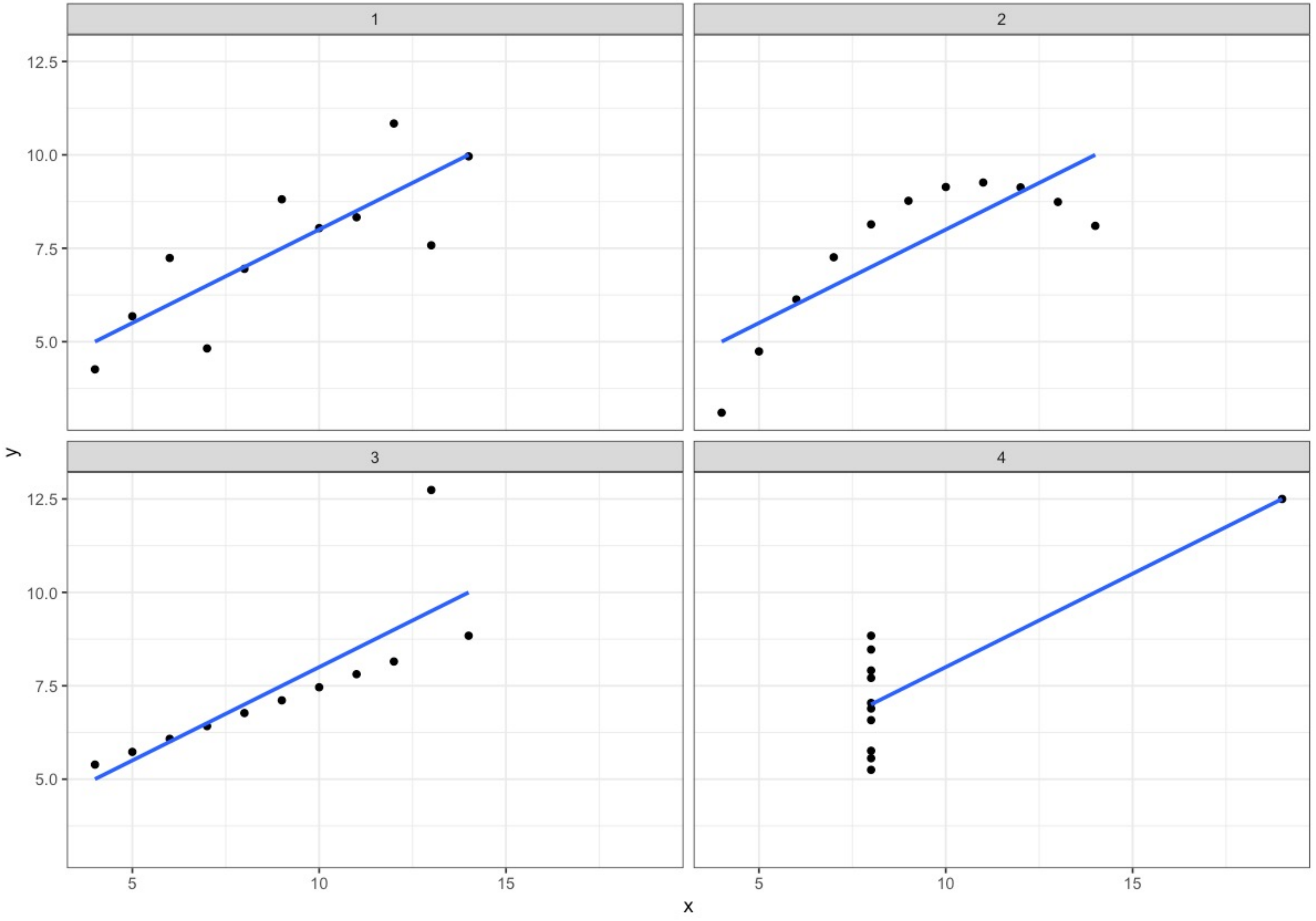
20220202_anscombe.R

Checking assumptions

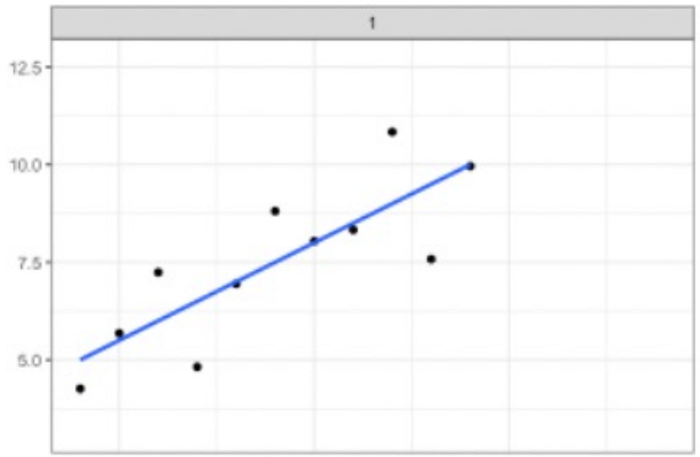
- graph data
- "Anscombe's quartet"

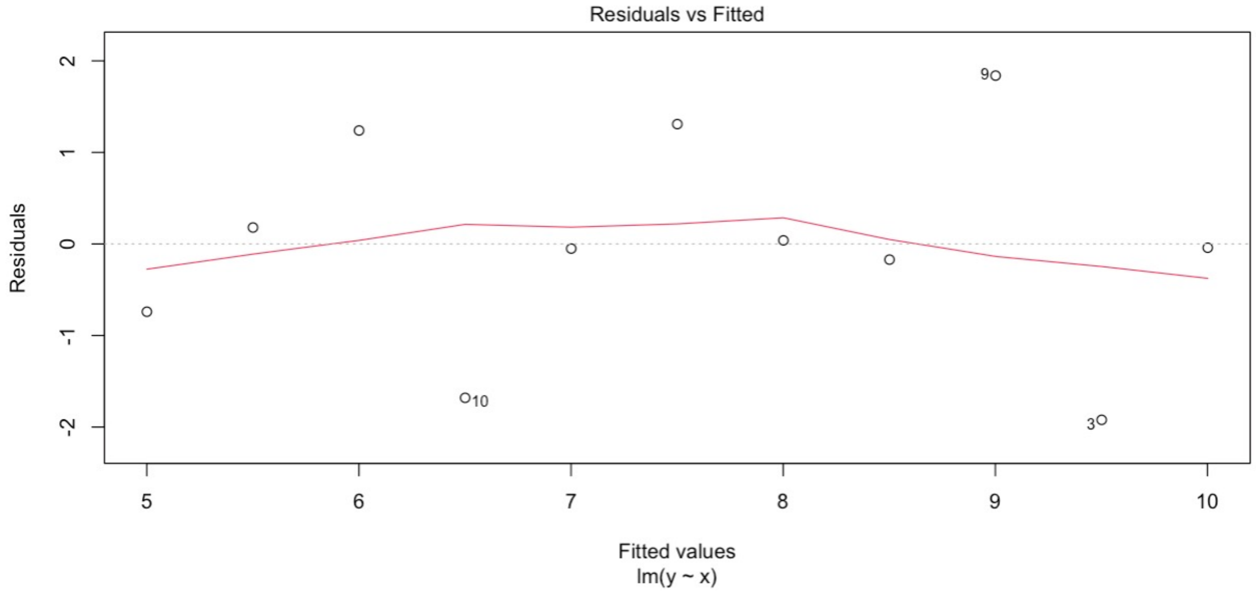


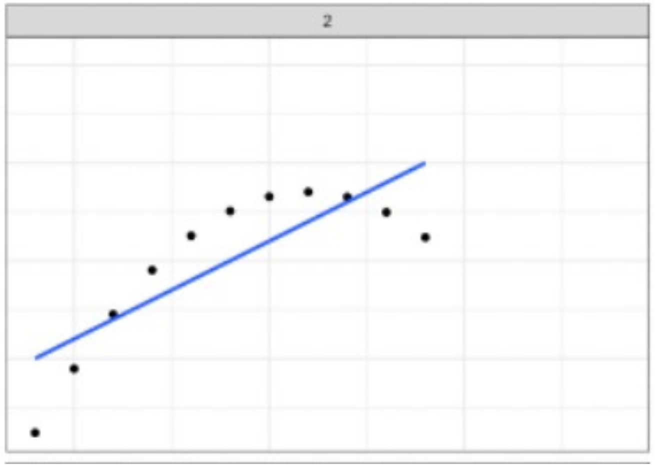
```
> ans_data %>% map_dfr(~ tidy(lm(y~x, data = .)), .id = "model")
# A tibble: 8 x 6
  model term        estimate std.error statistic p.value
  <chr> <chr>          <dbl>     <dbl>     <dbl>   <dbl>
1 1     (Intercept)    3.00      1.12       2.67 0.0257
2 1     x              0.500     0.118      4.24 0.00217
3 2     (Intercept)    3.00      1.13       2.67 0.0258
4 2     x              0.5       0.118      4.24 0.00218
5 3     (Intercept)    3.00      1.12       2.67 0.0256
6 3     x              0.500     0.118      4.24 0.00218
7 4     (Intercept)    3.00      1.12       2.67 0.0256
8 4     x              0.500     0.118      4.24 0.00216
```

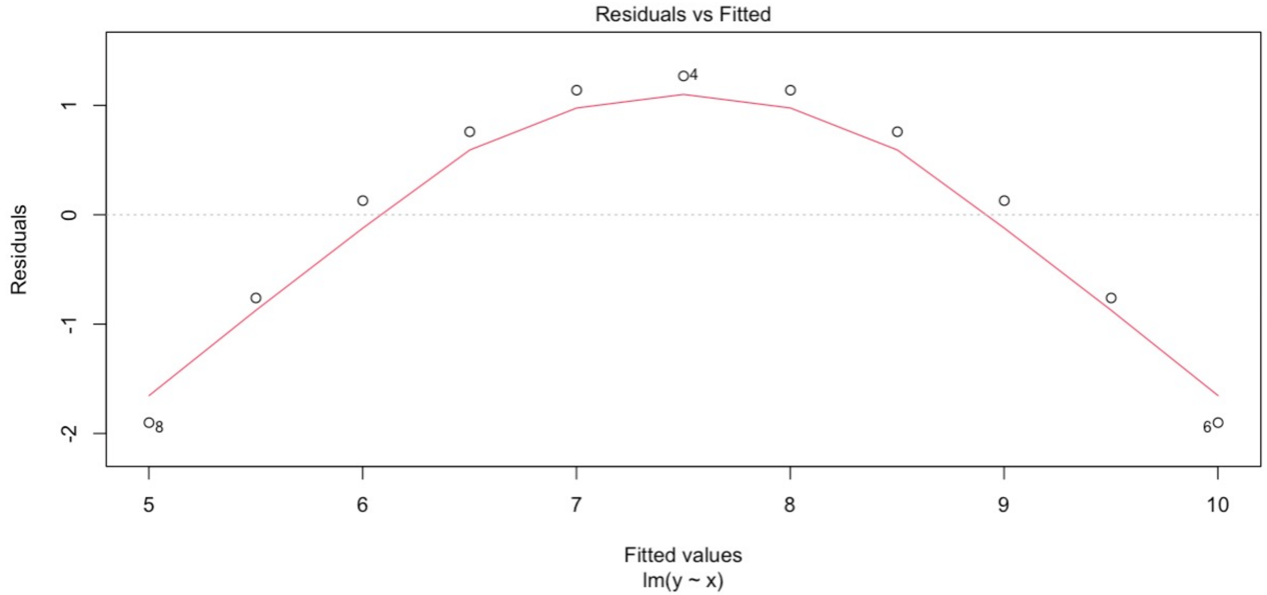


20220202_anscombe.R

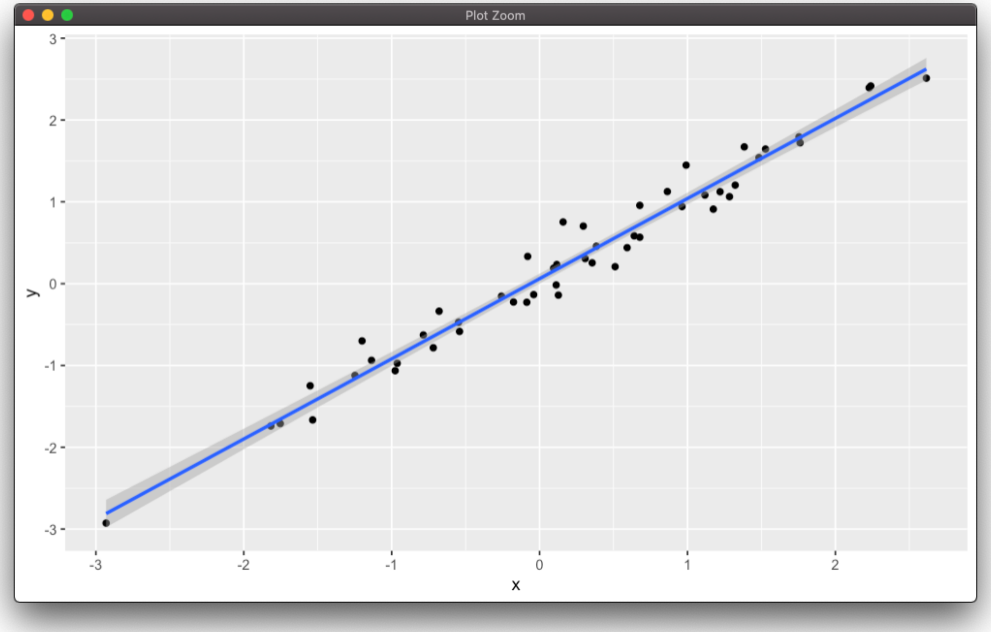


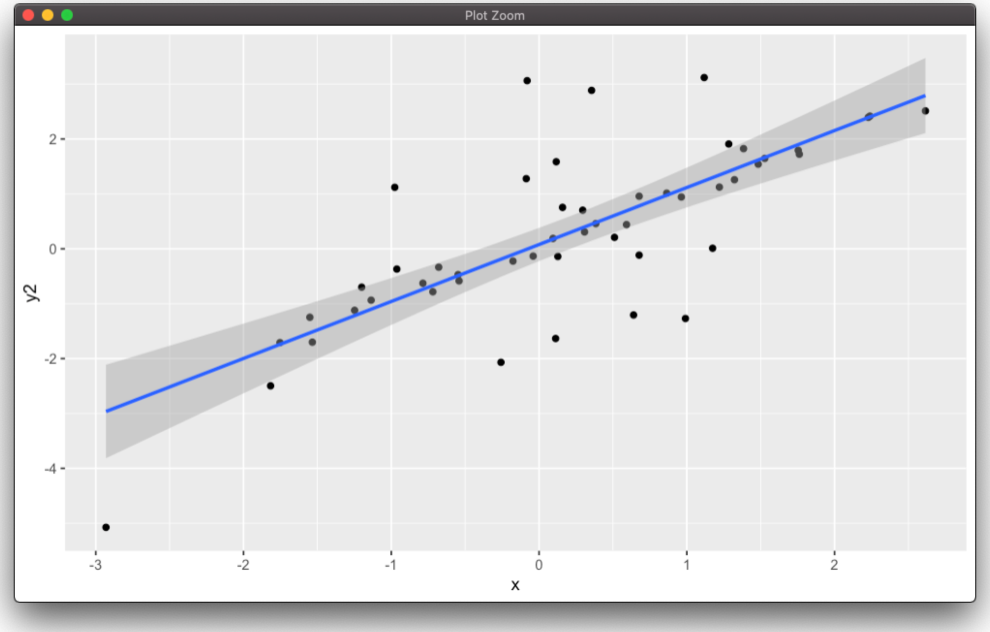


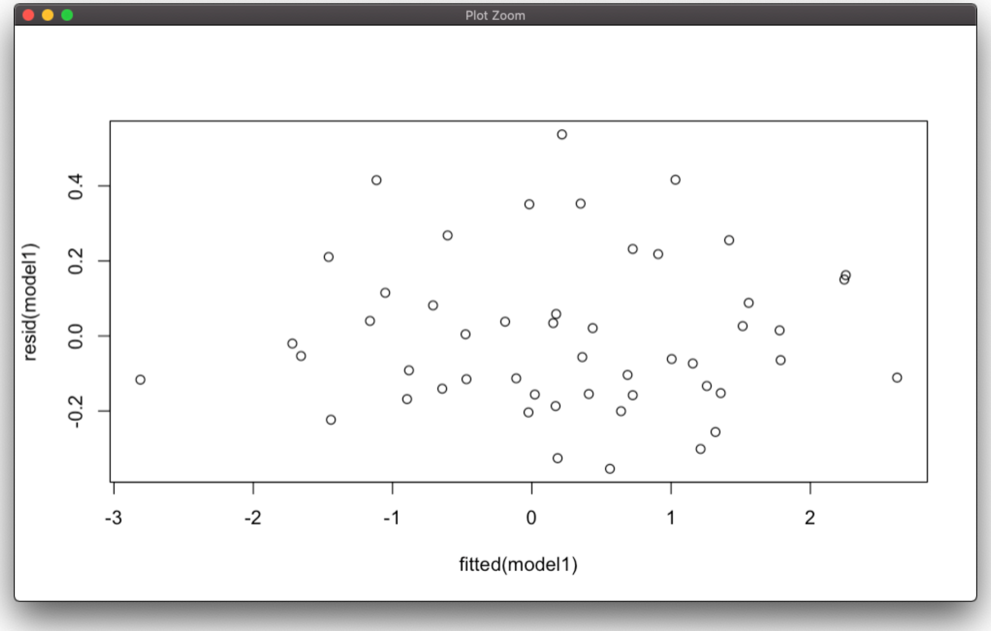


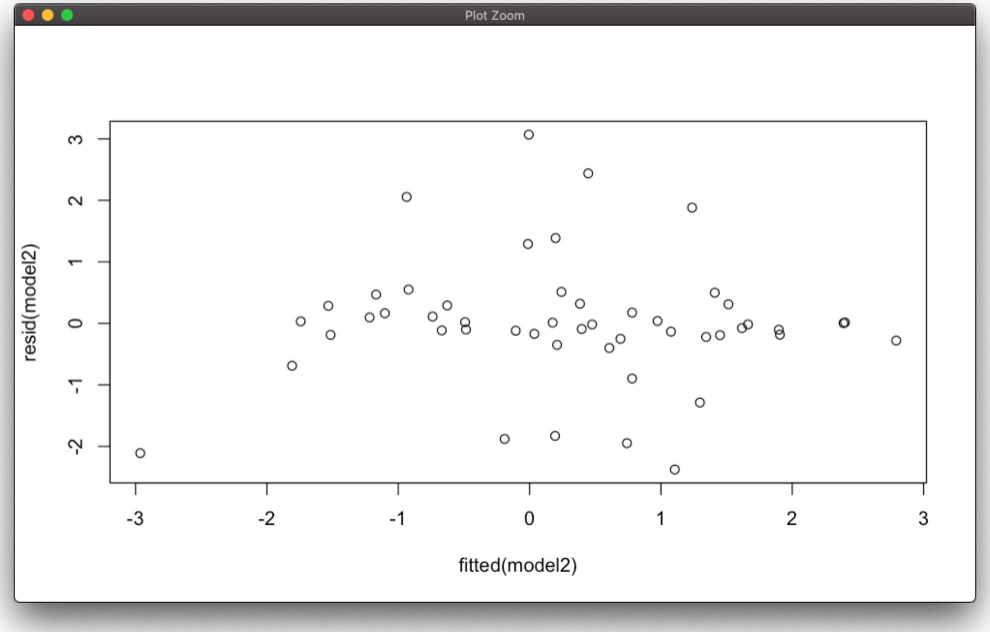


20220202_anscombe.R

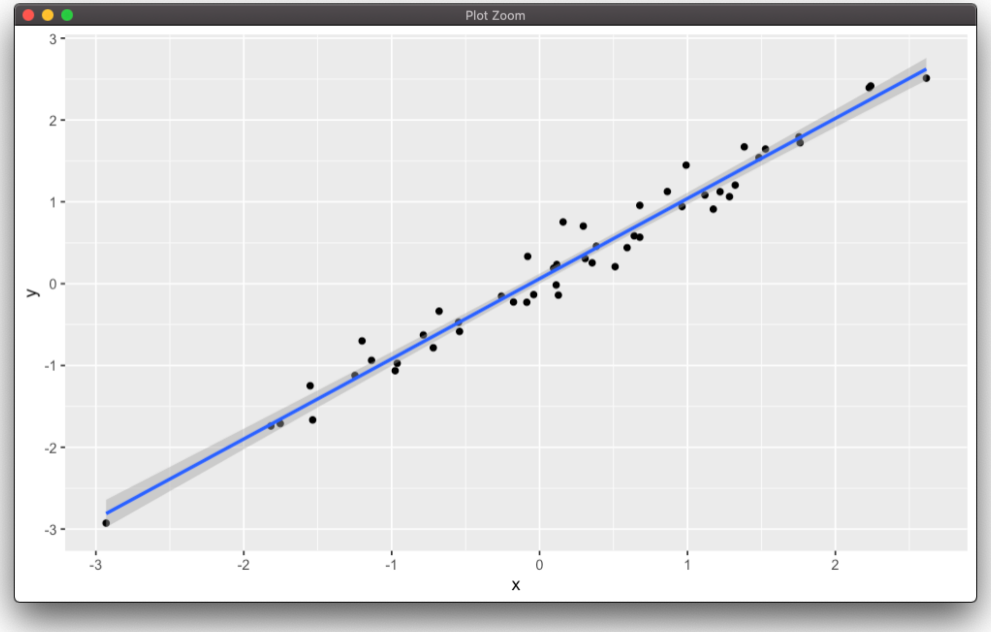


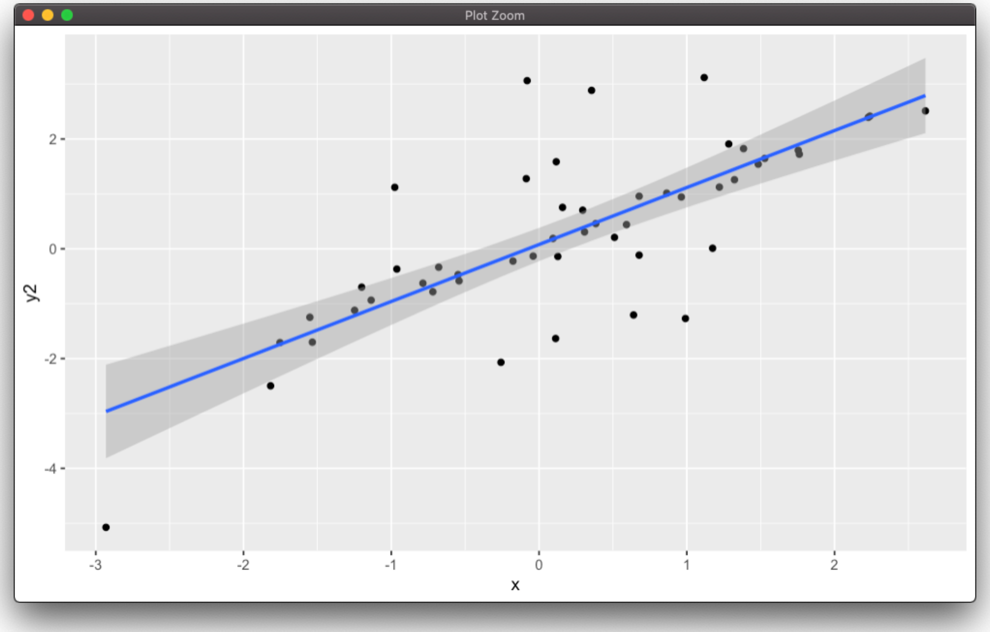


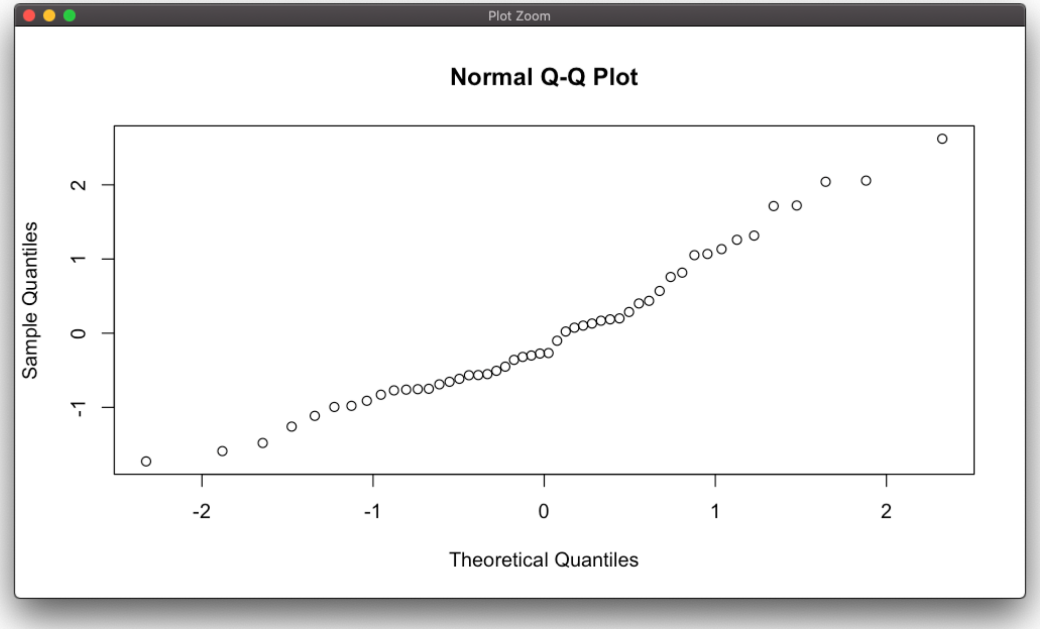


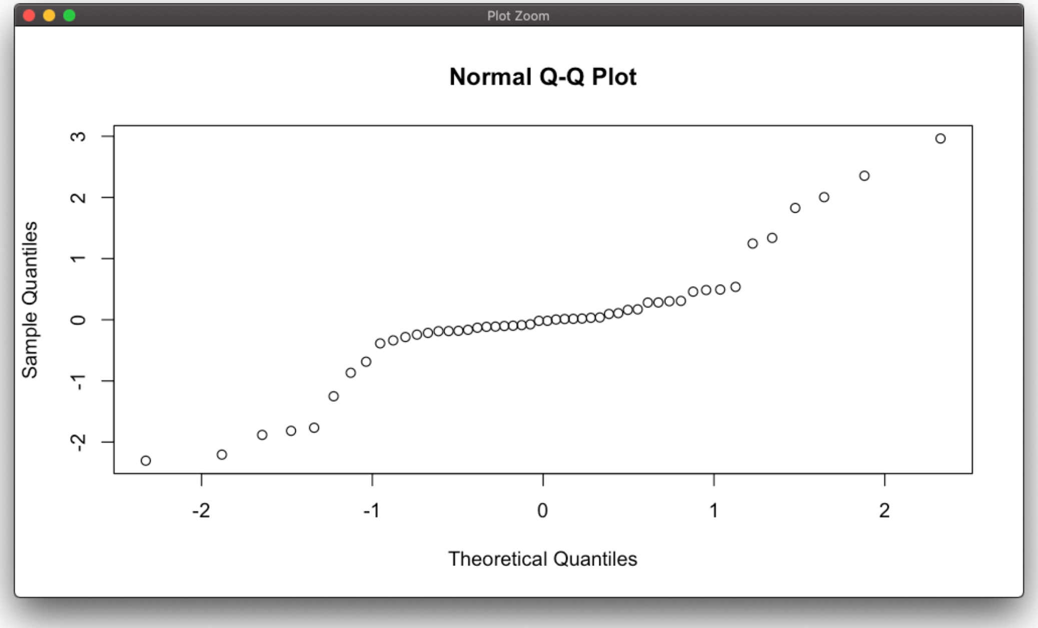


20220202_regression.R









20220202_regression.R

Analysis vs prediction

- analysis: trying to understand relationship between x variables and y variable (usually we are here)
  - what are the p-values? 😬
  - inferences about the direction of relationship: related to degree of experimental control – observational vs experimental study – were your x variables manipulated / randomly assigned?

- prediction: trying to devise an optimal strategy to predict y from a set of x variables on a **new** sample
  - variable selection / engineering
  - forward, backward, stepwise selection
  - cross-validation, consideration of range of x variables
  - <u>do not predict</u> outside of range of sample

Regression vs ANOVA

- Underlying mathematical model is essentially identical

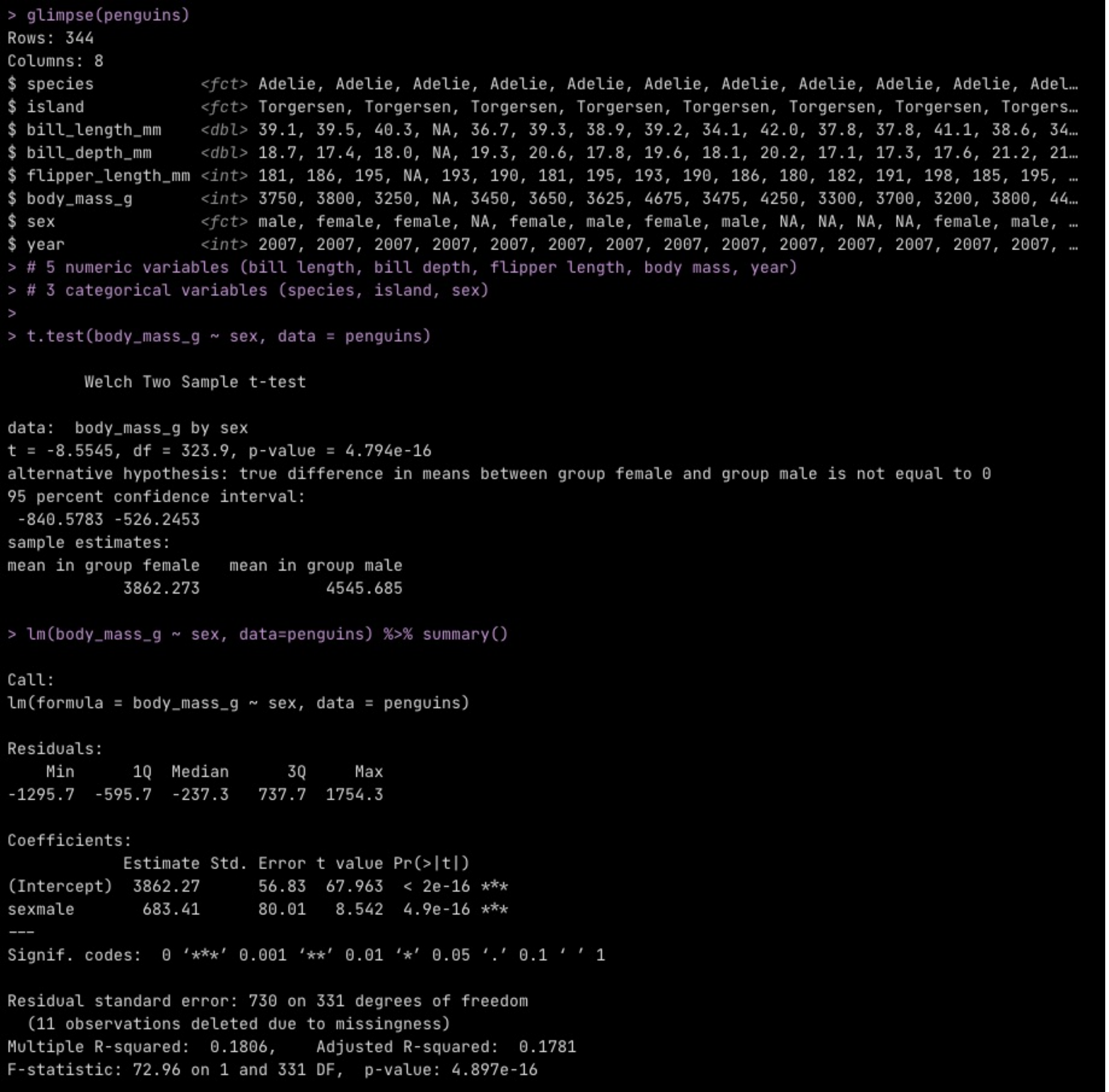
- ANOVA involves categorical predictors whereas regression typically involves continuous predictors (but can accommodate both)

- By recoding categorical predictors ("dummy variables") can compute ANOVA results in regression framework
  - eye color – blue 0, brown 1
  - eyecolor1 – blue 0, brown 0, green 1; eyecolor2 – blue 0, brown 1, green 0
  - R does this automatically when you put a character or factor variable into a model

Main effects and interactions

- Main effects move your prediction of the outcome variable per unit of the predictor (continuous) or based on category membership of the predictor (discrete)

- Interactions allow predictors to influence each other. For example, the relationship between height and IQ is different for people with brown eyes versus people with blue eyes



```
> glimpse(penguins)
Rows: 344
Columns: 8
$ species           <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adel…
$ island            <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torgersen, Torgersen, Torgersen, Torgers…
$ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34.1, 42.0, 37.8, 37.8, 41.1, 38.6, 34…
$ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18.1, 20.2, 17.1, 17.3, 17.6, 21.2, 21…
$ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, 186, 180, 182, 191, 198, 185, 195, …
$ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 3475, 4250, 3300, 3700, 3200, 3800, 44…
$ sex               <fct> male, female, female, NA, female, male, female, male, NA, NA, NA, NA, female, male, …
$ year              <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, …
> # 5 numeric variables (bill length, bill depth, flipper length, body mass, year)
> # 3 categorical variables (species, island, sex)
>
> t.test(body_mass_g ~ sex, data = penguins)

        Welch Two Sample t-test

data:  body_mass_g by sex
t = -8.5545, df = 323.9, p-value = 4.794e-16
alternative hypothesis: true difference in means between group female and group male is not equal to 0
95 percent confidence interval:
 -840.5783 -526.2453
sample estimates:
mean in group female   mean in group male
            3862.273             4545.685

> lm(body_mass_g ~ sex, data=penguins) %>% summary()

Call:
lm(formula = body_mass_g ~ sex, data = penguins)

Residuals:
    Min      1Q  Median      3Q     Max
-1295.7  -595.7  -237.3   737.7  1754.3

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  3862.27      56.83  67.963  < 2e-16 ***
sexmale       683.41      80.01   8.542  4.9e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 730 on 331 degrees of freedom
  (11 observations deleted due to missingness)
Multiple R-squared:  0.1806,    Adjusted R-squared:  0.1781
F-statistic: 72.96 on 1 and 331 DF,  p-value: 4.897e-16
```

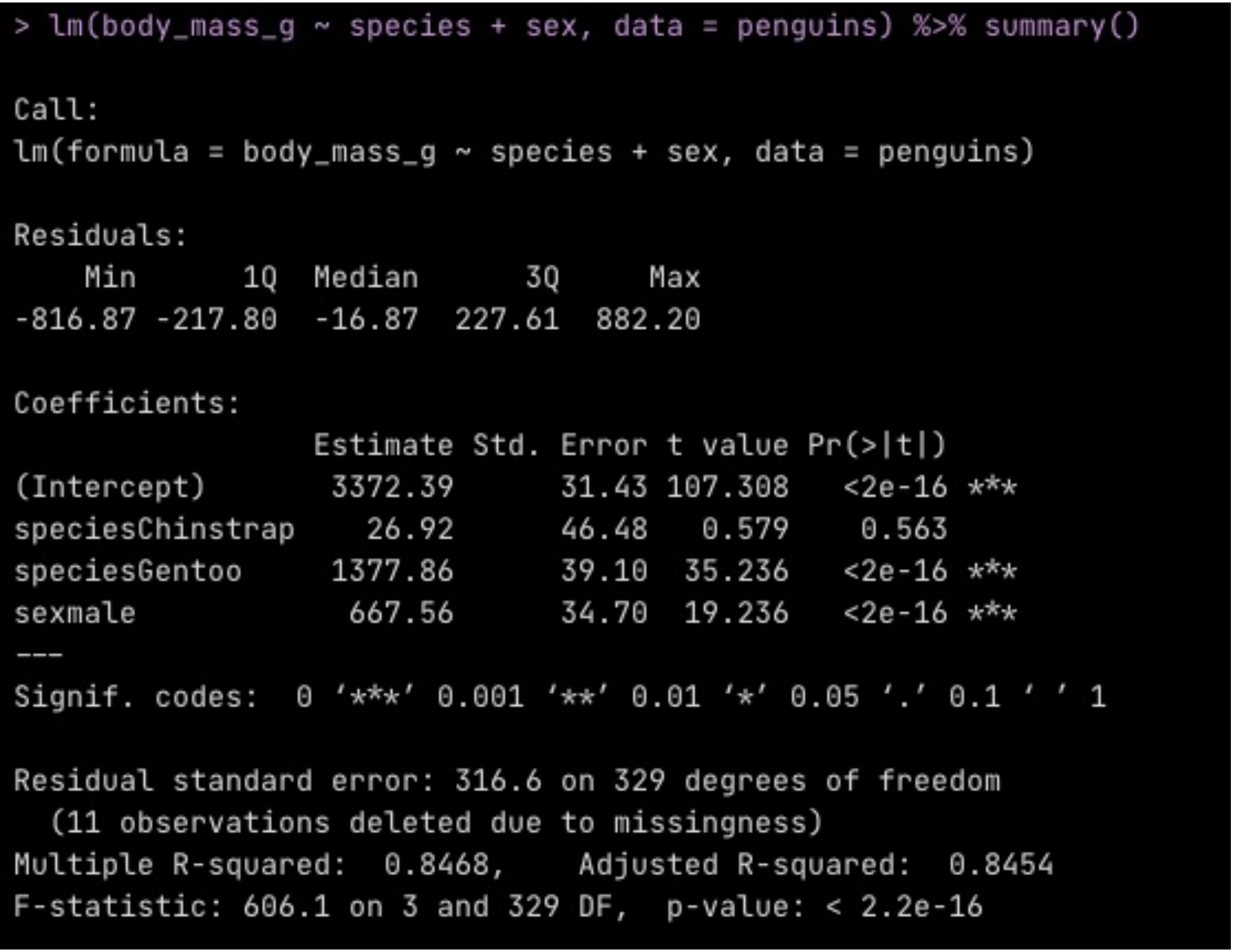
"palmerpenguins" data

t-test and lm ("linear model") use same "formula" syntax

y ~ x

lm creates a "linear model" object
we must run summary (or some other function) on to see results

20220202_linearmodels.R



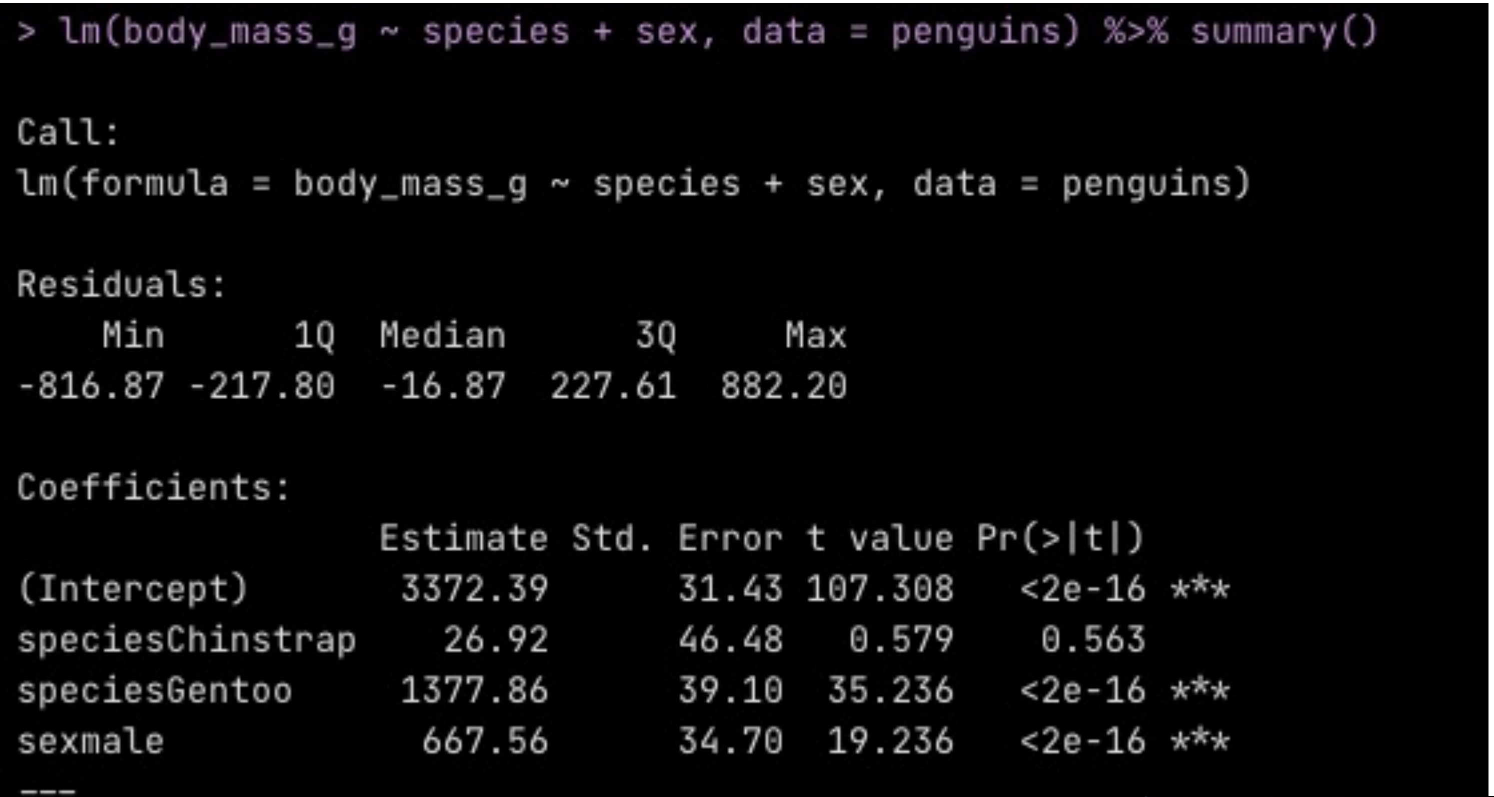
```
> lm(body_mass_g ~ species + sex, data = penguins) %>% summary()

Call:
lm(formula = body_mass_g ~ species + sex, data = penguins)

Residuals:
    Min      1Q  Median      3Q     Max
-816.87 -217.80  -16.87  227.61  882.20

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       3372.39      31.43 107.308   <2e-16 ***
speciesChinstrap    26.92      46.48   0.579    0.563
speciesGentoo     1377.86      39.10  35.236   <2e-16 ***
sexmale            667.56      34.70  19.236   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 316.6 on 329 degrees of freedom
  (11 observations deleted due to missingness)
Multiple R-squared:  0.8468,    Adjusted R-squared:  0.8454
F-statistic: 606.1 on 3 and 329 DF,  p-value: < 2.2e-16
```

categorical predictors: coefficients are relative to reference level (default is alphabetical)

intercept is average "reference" penguin
female, Adelie

if Chinstrap, 26.92 g heavier than intercept
if Gentoo, 1377.86 g heavier than intercept
if male (of any species), 667.56 g heavier

20220202_linearmodels.R

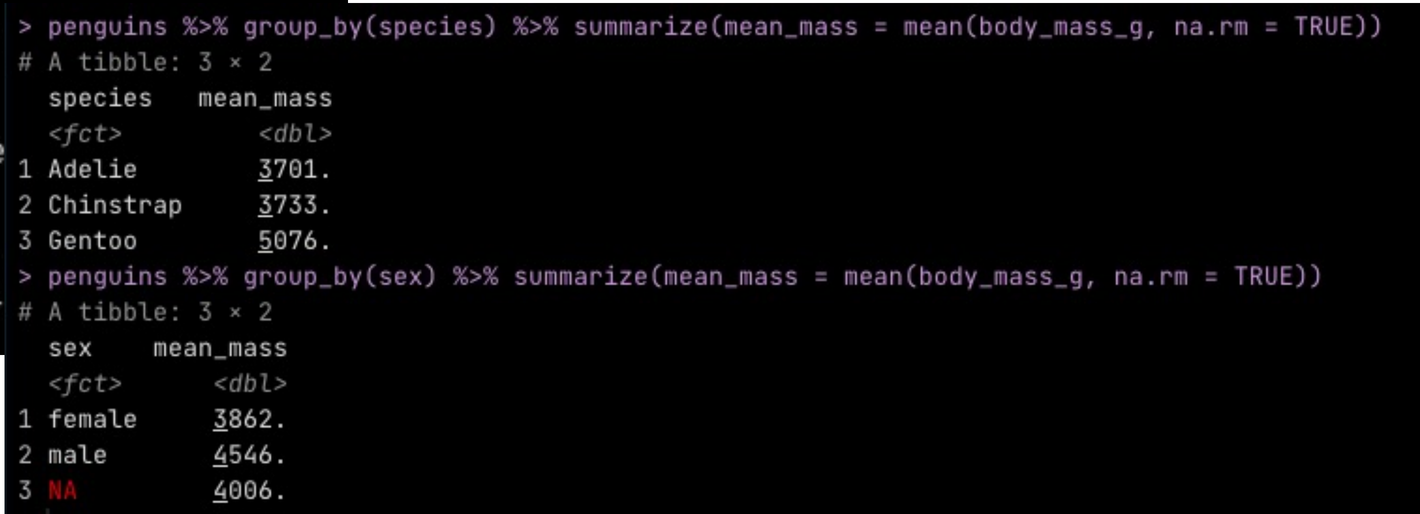


```
> lm(body_mass_g ~ species + sex, data = penguins) %>% summary()

Call:
lm(formula = body_mass_g ~ species + sex, data = penguins)

Residuals:
    Min      1Q  Median      3Q     Max 
-816.87 -217.80  -16.87  227.61  882.20 

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)       3372.39      31.43 107.308   <2e-16 ***
speciesChinstrap    26.92      46.48   0.579    0.563    
speciesGentoo     1377.86      39.10  35.236   <2e-16 ***
sexmale            667.56      34.70  19.236   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’

Residual standard error: 316.6 on 329 degrees of free
  (11 observations deleted due to missingness)
Multiple R-squared:  0.8468,	Adjusted R-squared: 
F-statistic: 606.1 on 3 and 329 DF,  p-value: < 2.2e-
```

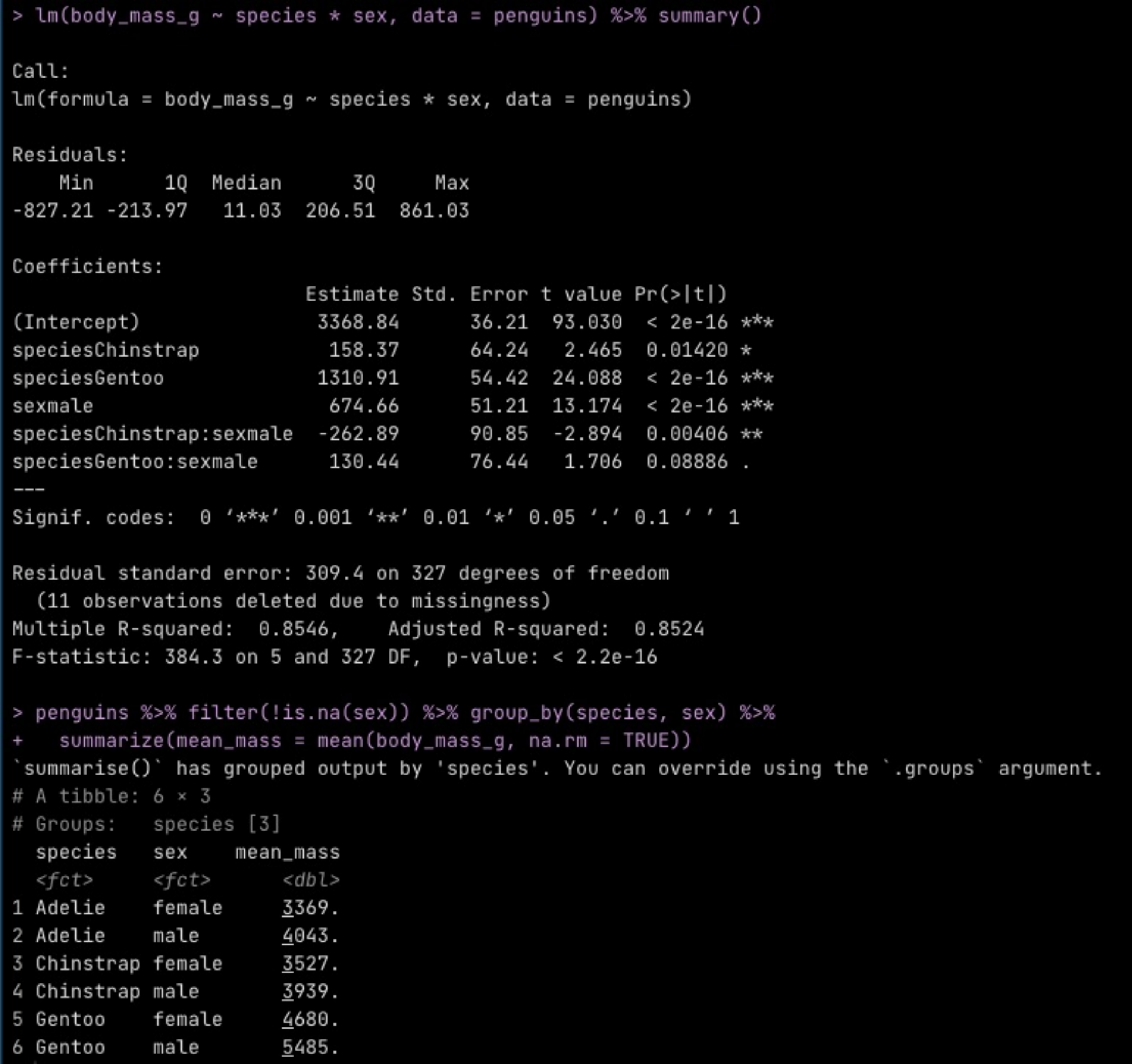
categorical predictors: coefficients are relative to reference level (default is alphabetical)

intercept is average "reference" penguin
female, Adelie

if Chinstrap, 26.92 g heavier than intercept
if Gentoo, 1377.86 g heavier than intercept
if male (of any species), 667.56 g heavier



```
> penguins %>% group_by(species) %>% summarize(mean_mass = mean(body_mass_g, na.rm = TRUE))
# A tibble: 3 × 2
  species   mean_mass
  <fct>         <dbl>
1 Adelie        3701.
2 Chinstrap     3733.
3 Gentoo        5076.
> penguins %>% group_by(sex) %>% summarize(mean_mass = mean(body_mass_g, na.rm = TRUE))
# A tibble: 3 × 2
  sex    mean_mass
  <fct>      <dbl>
1 female     3862.
2 male       4546.
3 NA         4006.
```



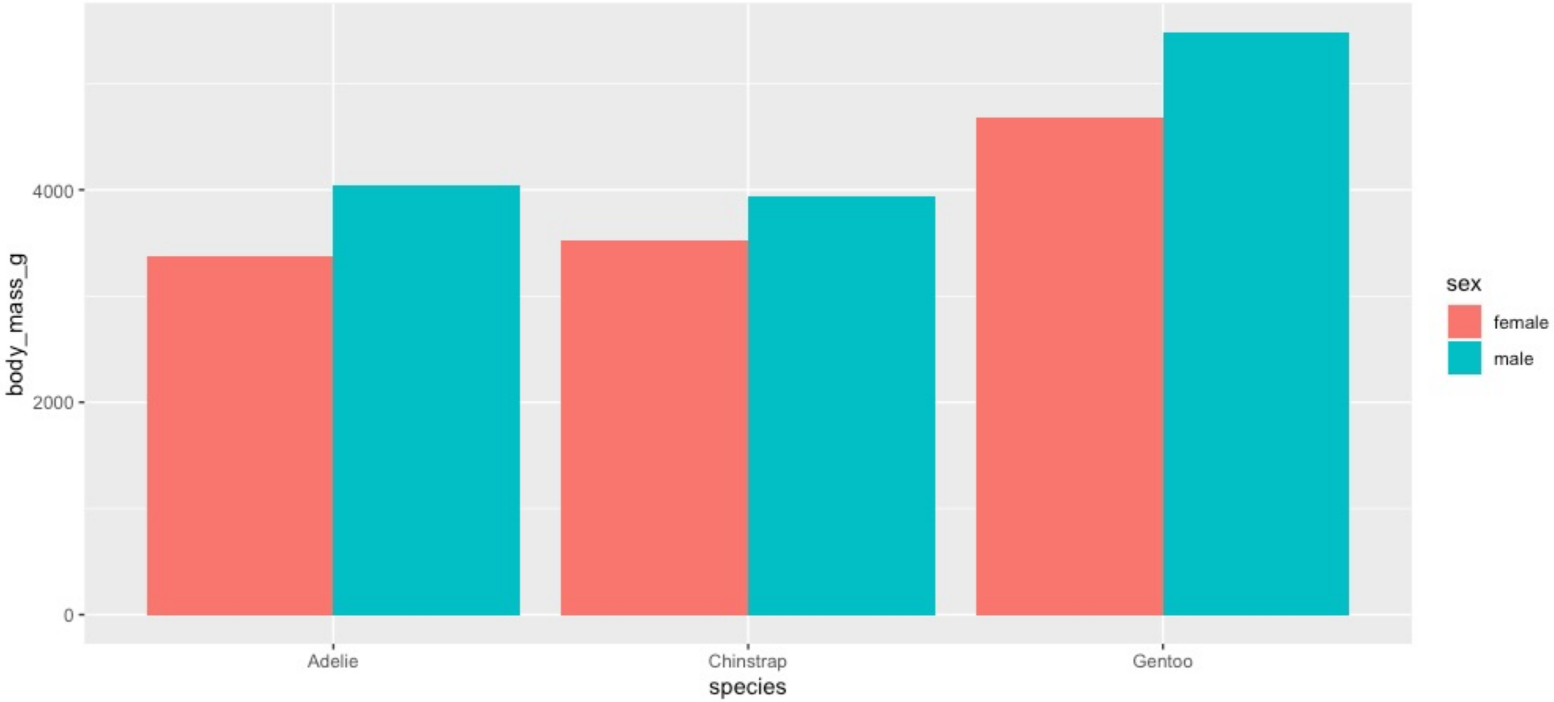
```
> lm(body_mass_g ~ species * sex, data = penguins) %>% summary()

Call:
lm(formula = body_mass_g ~ species * sex, data = penguins)

Residuals:
    Min      1Q  Median      3Q     Max 
-827.21 -213.97   11.03  206.51  861.03 

Coefficients:
                         Estimate Std. Error t value Pr(>|t|)    
(Intercept)               3368.84      36.21  93.030  < 2e-16 ***
speciesChinstrap           158.37      64.24   2.465  0.01420 *  
speciesGentoo             1310.91      54.42  24.088  < 2e-16 ***
sexmale                    674.66      51.21  13.174  < 2e-16 ***
speciesChinstrap:sexmale  -262.89      90.85  -2.894  0.00406 ** 
speciesGentoo:sexmale      130.44      76.44   1.706  0.08886 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 309.4 on 327 degrees of freedom
  (11 observations deleted due to missingness)
Multiple R-squared:  0.8546,	Adjusted R-squared:  0.8524 
F-statistic: 384.3 on 5 and 327 DF,  p-value: < 2.2e-16

> penguins %>% filter(!is.na(sex)) %>% group_by(species, sex) %>%
+   summarize(mean_mass = mean(body_mass_g, na.rm = TRUE))
`summarise()` has grouped output by 'species'. You can override using the `.groups` argument.
# A tibble: 6 × 3
# Groups:   species [3]
  species   sex    mean_mass
  <fct>     <fct>      <dbl>
1 Adelie    female     3369.
2 Adelie    male       4043.
3 Chinstrap female     3527.
4 Chinstrap male       3939.
5 Gentoo    female     4680.
6 Gentoo    male       5485.
```

include interaction term: allows additional adjustment to prediction for combination of species and sex

adjust for species, adjust for sex, then additional adjustments for males of particular species

can see that difference between males and females is different for different species (less pronounced for Chinstrap compared to other two)

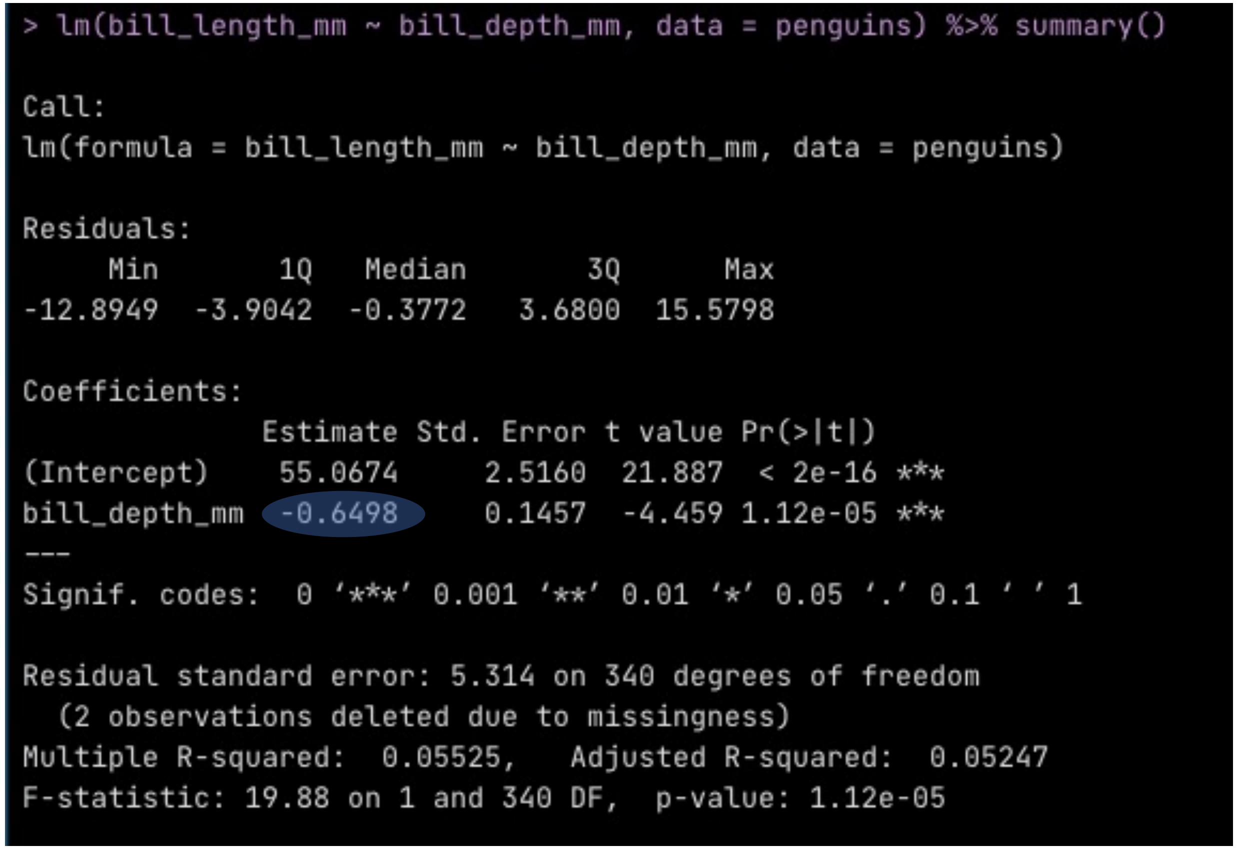
20220202_linearmodels.R



include interaction term: allows additional adjustment to prediction for combination of species and sex

adjust for species, adjust for sex, then additional adjustments for males of particular species

can see that difference between males and females is different for different species (less pronounced for Chinstrap compared to other two)

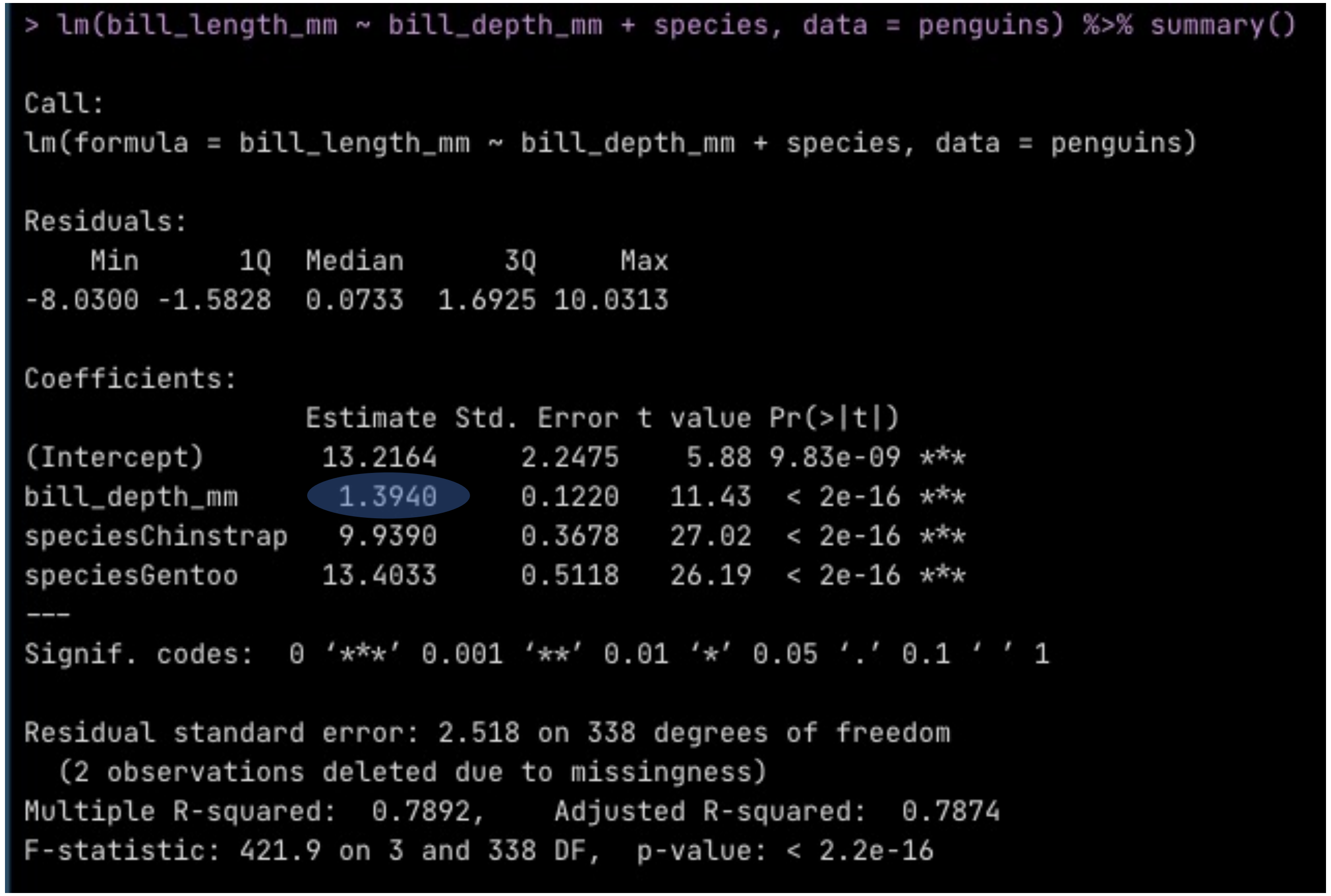


```
> lm(bill_length_mm ~ bill_depth_mm, data = penguins) %>% summary()

Call:
lm(formula = bill_length_mm ~ bill_depth_mm, data = penguins)

Residuals:
     Min       1Q   Median       3Q      Max 
-12.8949  -3.9042  -0.3772   3.6800  15.5798 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)    55.0674     2.5160  21.887  < 2e-16 ***
bill_depth_mm  -0.6498     0.1457  -4.459 1.12e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.314 on 340 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.05525,	Adjusted R-squared:  0.05247 
F-statistic: 19.88 on 1 and 340 DF,  p-value: 1.12e-05
```

this also works with continuous variables

on average, every additional mm of bill depth would be associated with 0.6498 mm less bill length

20220202_linearmodels.R



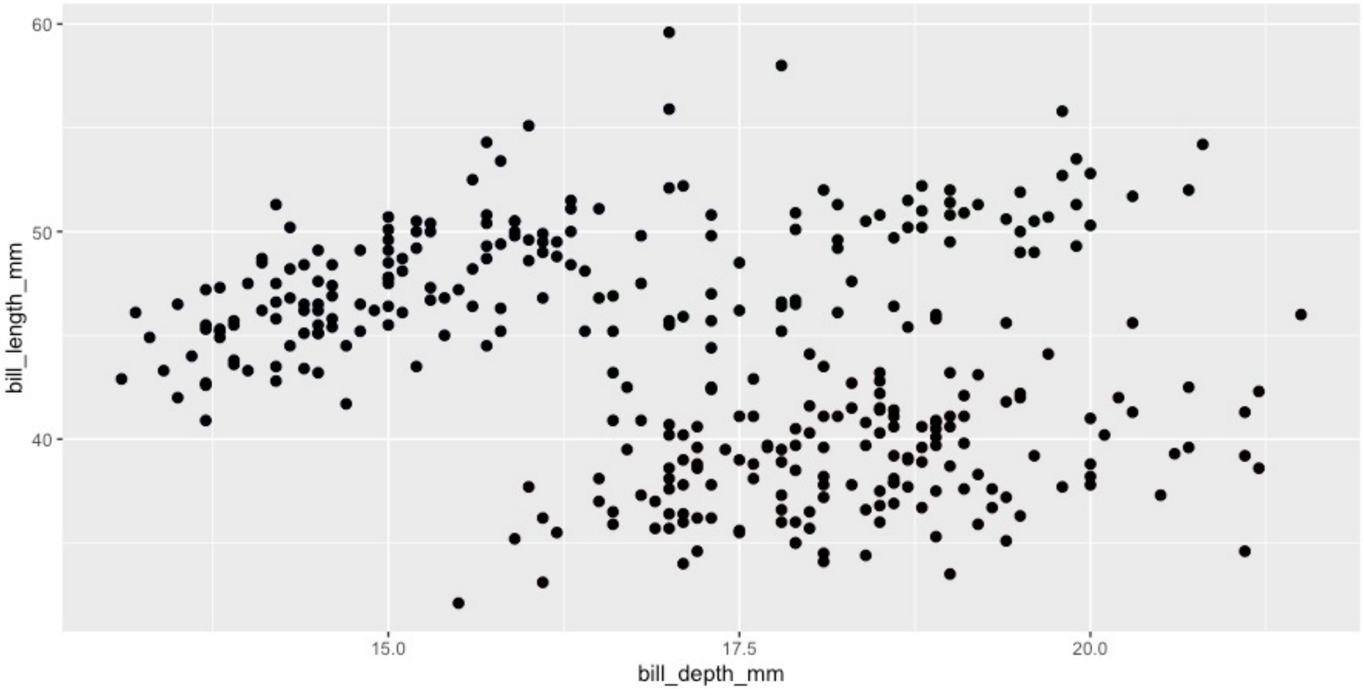
```
> lm(bill_length_mm ~ bill_depth_mm + species, data = penguins) %>% summary()

Call:
lm(formula = bill_length_mm ~ bill_depth_mm + species, data = penguins)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.0300 -1.5828  0.0733  1.6925 10.0313 

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)       13.2164     2.2475    5.88 9.83e-09 ***
bill_depth_mm      1.3940     0.1220   11.43  < 2e-16 ***
speciesChinstrap   9.9390     0.3678   27.02  < 2e-16 ***
speciesGentoo     13.4033     0.5118   26.19  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.518 on 338 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.7892,	Adjusted R-squared:  0.7874 
F-statistic: 421.9 on 3 and 338 DF,  p-value: < 2.2e-16
```

but if we add species into the model, quite a different picture emerges

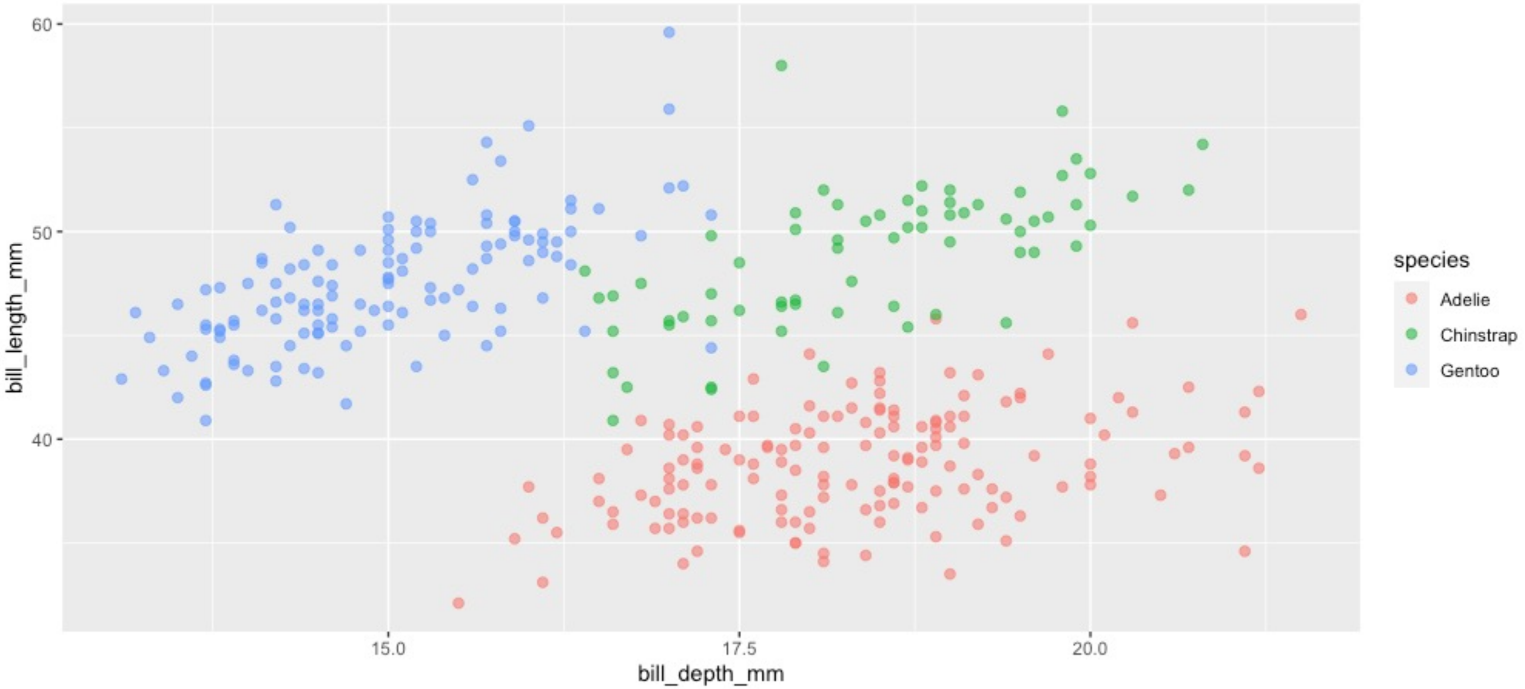
now coefficient of bill depth is positive instead of negative and there are significant effects of species. What's going on?

(we have not even added an interaction term yet!)

20220202_linearmodels.R

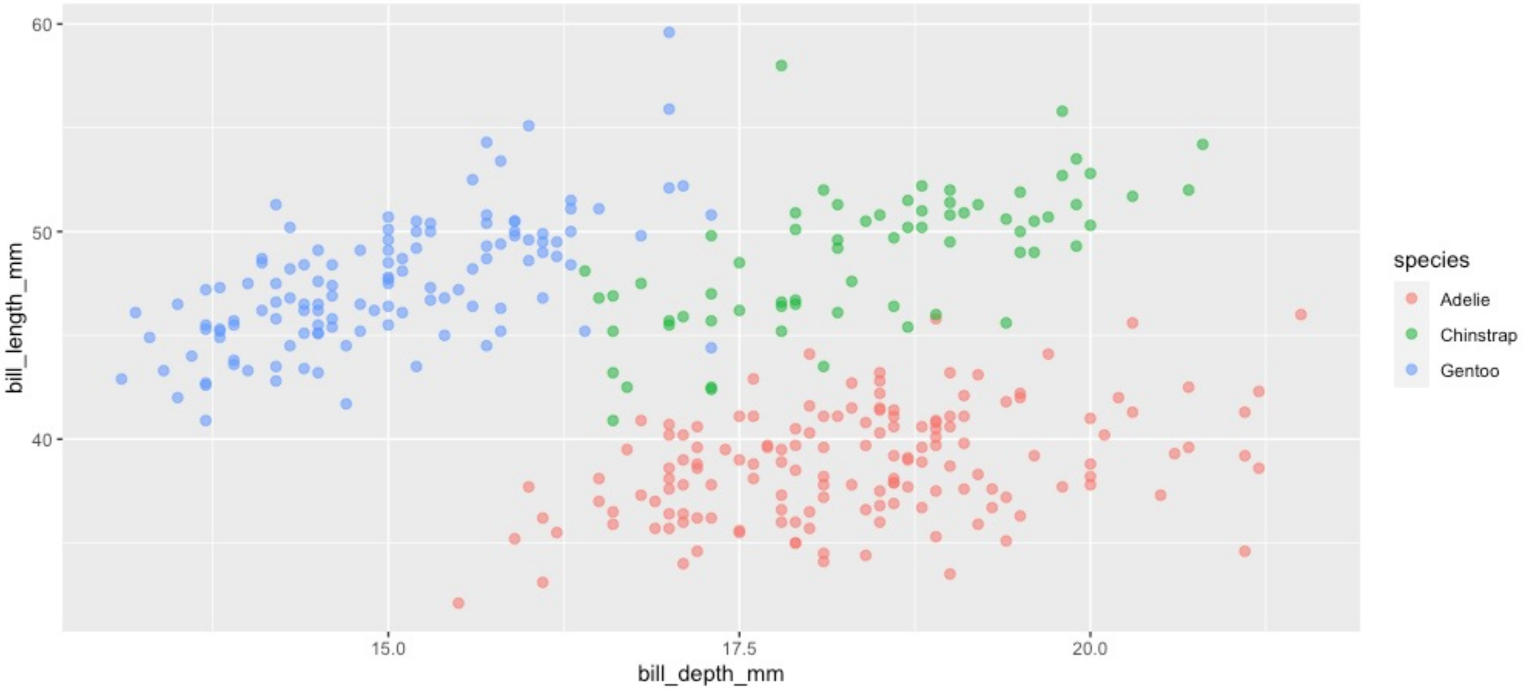


Plotting data relating bill length and depth shows clear negative relationship.



Plotting data relating bill length and depth shows clear negative relationship.

But if we look at data by species, can see that there are clusters of points for each species – substantial species differences in both bill length and depth.



Plotting data relating bill length and depth shows clear negative relationship.

But if we look at data by species, can see that there are clusters of points for each species – substantial species differences in both bill length and depth.

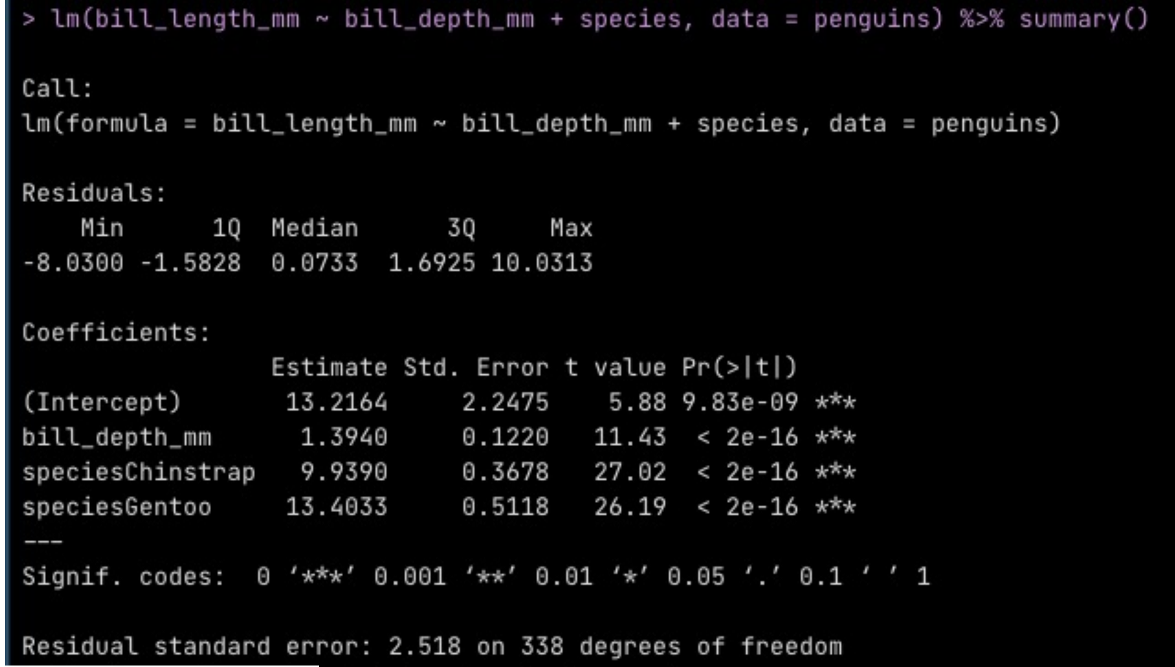
overall differences in bill depth and length between species confound the relationship between the two variables

once you adjust for species, now the relationship is positive.
"*Simpson's Paradox*"

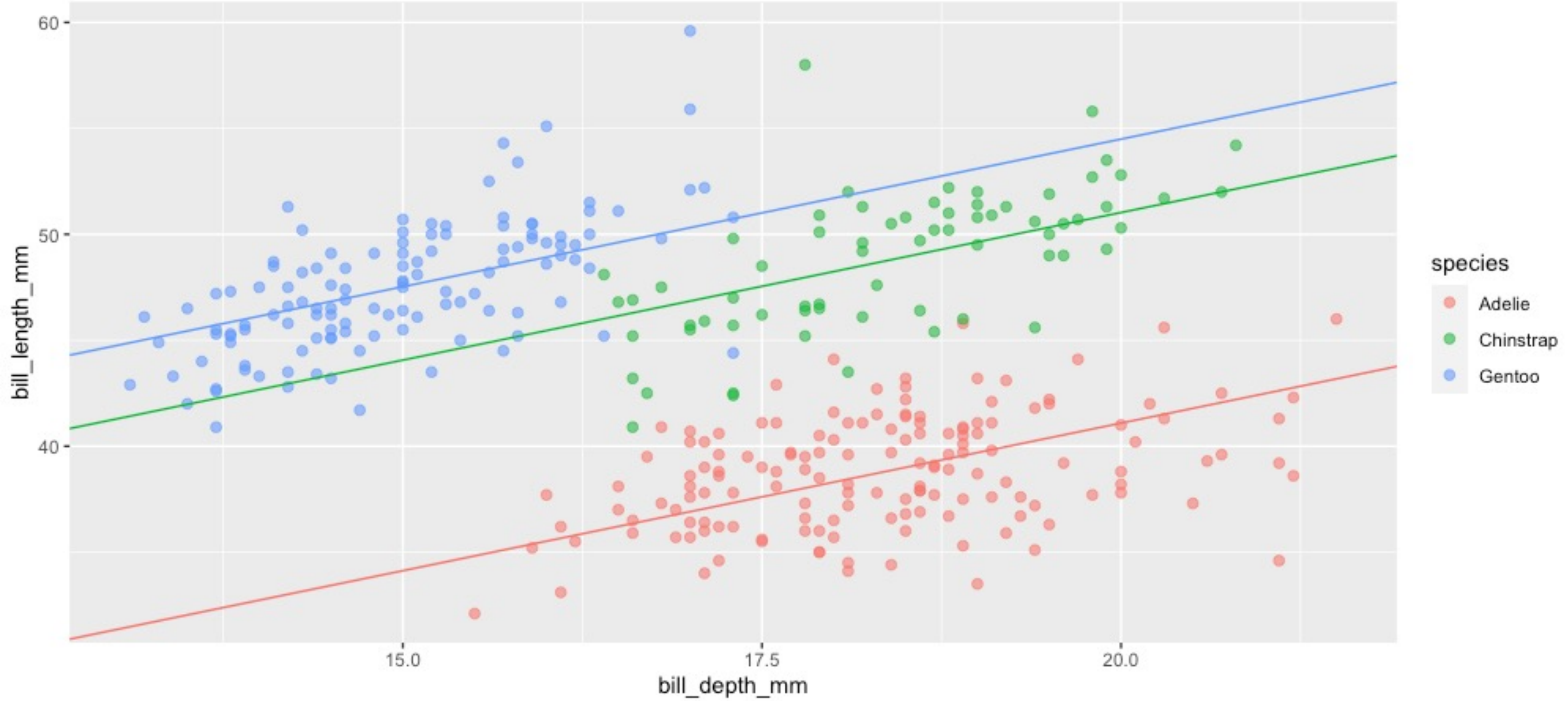
20220202_linearmodels.R

additive model – no interaction term
"+" between bill_depth_mm and species

regression lines are parallel and shifted up and down by species (adjusting y-intercept)



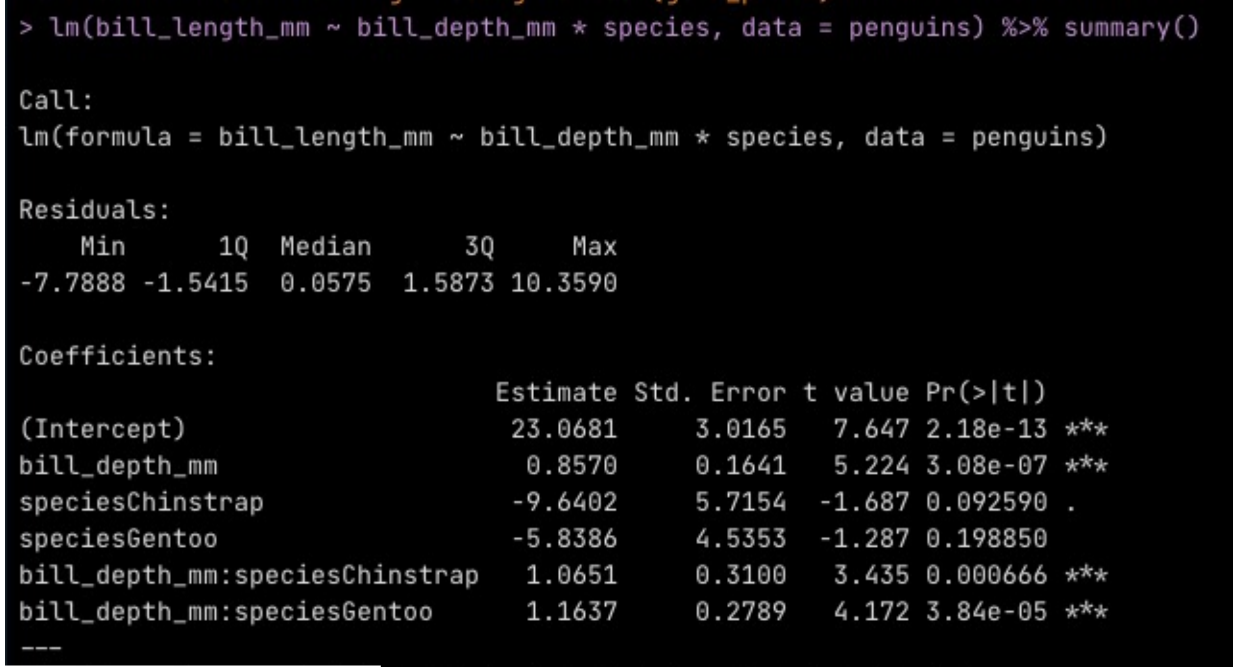
```
> lm(bill_length_mm ~ bill_depth_mm + species, data = penguins) %>% summary()

Call:
lm(formula = bill_length_mm ~ bill_depth_mm + species, data = penguins)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.0300 -1.5828  0.0733  1.6925 10.0313 

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)       13.2164     2.2475    5.88 9.83e-09 ***
bill_depth_mm      1.3940     0.1220   11.43  < 2e-16 ***
speciesChinstrap   9.9390     0.3678   27.02  < 2e-16 ***
speciesGentoo     13.4033     0.5118   26.19  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.518 on 338 degrees of freedom
                     leted due to missingness)
                      0.7892,    Adjusted R-squared:  0.7874 
                     n 3 and 338 DF,  p-value: < 2.2e-16
```



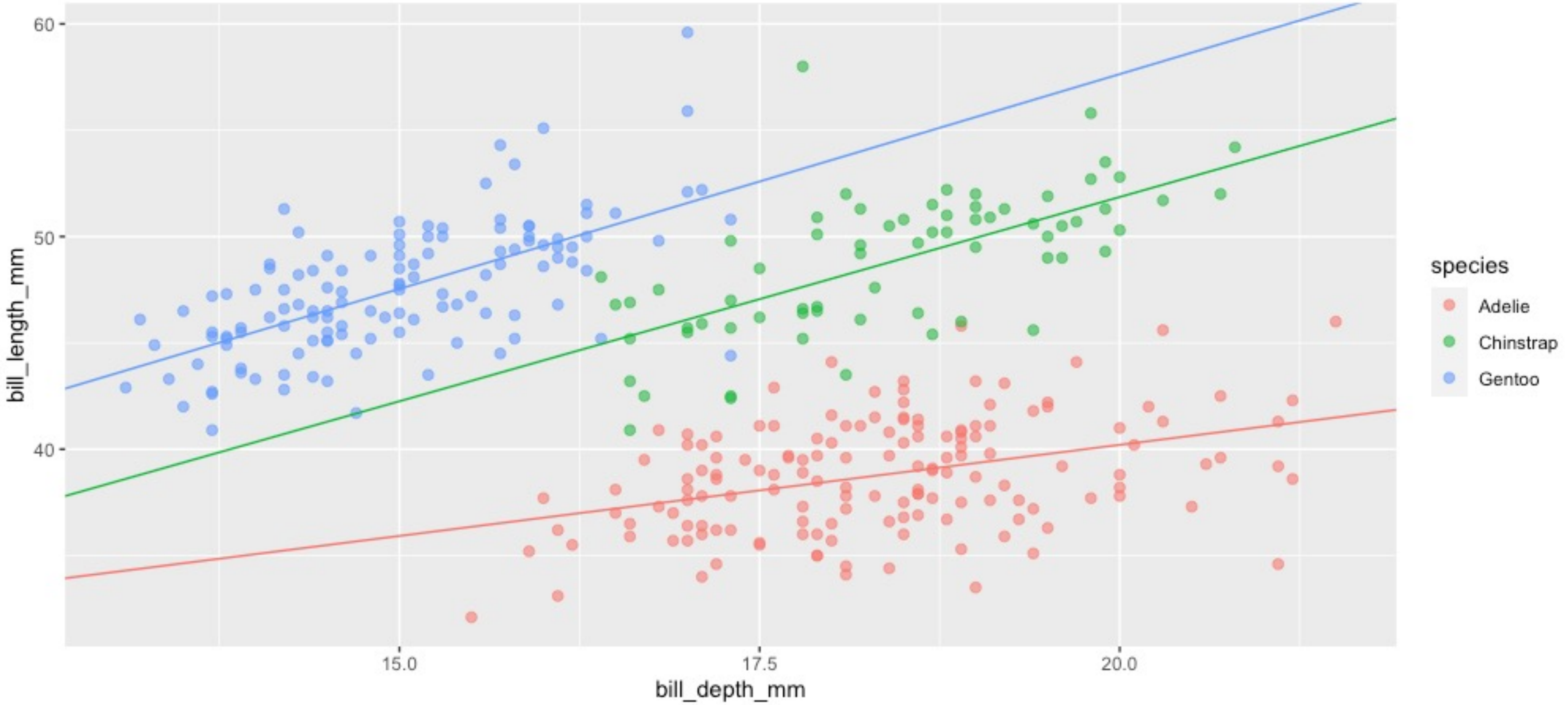
20220202_linearmodels.R

interaction term
"*" between bill_depth_mm and species

regression lines vary in slope as well as intercept: coefficients of interaction terms are modifiers to slope parameter. Can see relationship is stronger for Chinstrap and Gentoo than Adelie (each mm of bill depth predicts more increase in length)



```
> lm(bill_length_mm ~ bill_depth_mm * species, data = penguins) %>% summary()

Call:
lm(formula = bill_length_mm ~ bill_depth_mm * species, data = penguins)

Residuals:
    Min      1Q  Median      3Q     Max 
-7.7888 -1.5415  0.0575  1.5873 10.3590 

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)    
(Intercept)                    23.0681     3.0165   7.647 2.18e-13 ***
bill_depth_mm                   0.8570     0.1641   5.224 3.08e-07 ***
speciesChinstrap               -9.6402     5.7154  -1.687 0.092590 .  
speciesGentoo                  -5.8386     4.5353  -1.287 0.198850    
bill_depth_mm:speciesChinstrap  1.0651     0.3100   3.435 0.000666 ***
bill_depth_mm:speciesGentoo     1.1637     0.2789   4.172 3.84e-05 ***
---
                      0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                       2.445 on 336 degrees of freedom
                      ed due to missingness)
                      3024,     Adjusted R-squared:  0.7995 
                       and 336 DF,  p-value: < 2.2e-16
```



20220202_linearmodels.R

PAUSE

What if your observations are not independent?

- If you have observations that are not independent in some way, often you have *nested data*

- Mice from the same litter are more likely to be similar to each other than mice from different litters

- Dendritic segments from the same neuron are more likely to resemble each other than segments from different neurons

Nested data

- Average down to the level of independence
  - mean scores for mice from same litter
  - mean morphological measure for segments from same neuron
    - oops, different neurons from same mouse dependent! average again
  - This solves the independence problem but throws away data
    - you get same analysis result whether you have 1 segment from one neuron per mouse as you would if you have 10 segments from 50 neurons per mouse (although, averaging)

- Treat observations as independent
  - Inflates degrees of freedom, standard error estimates and p-values are wrong

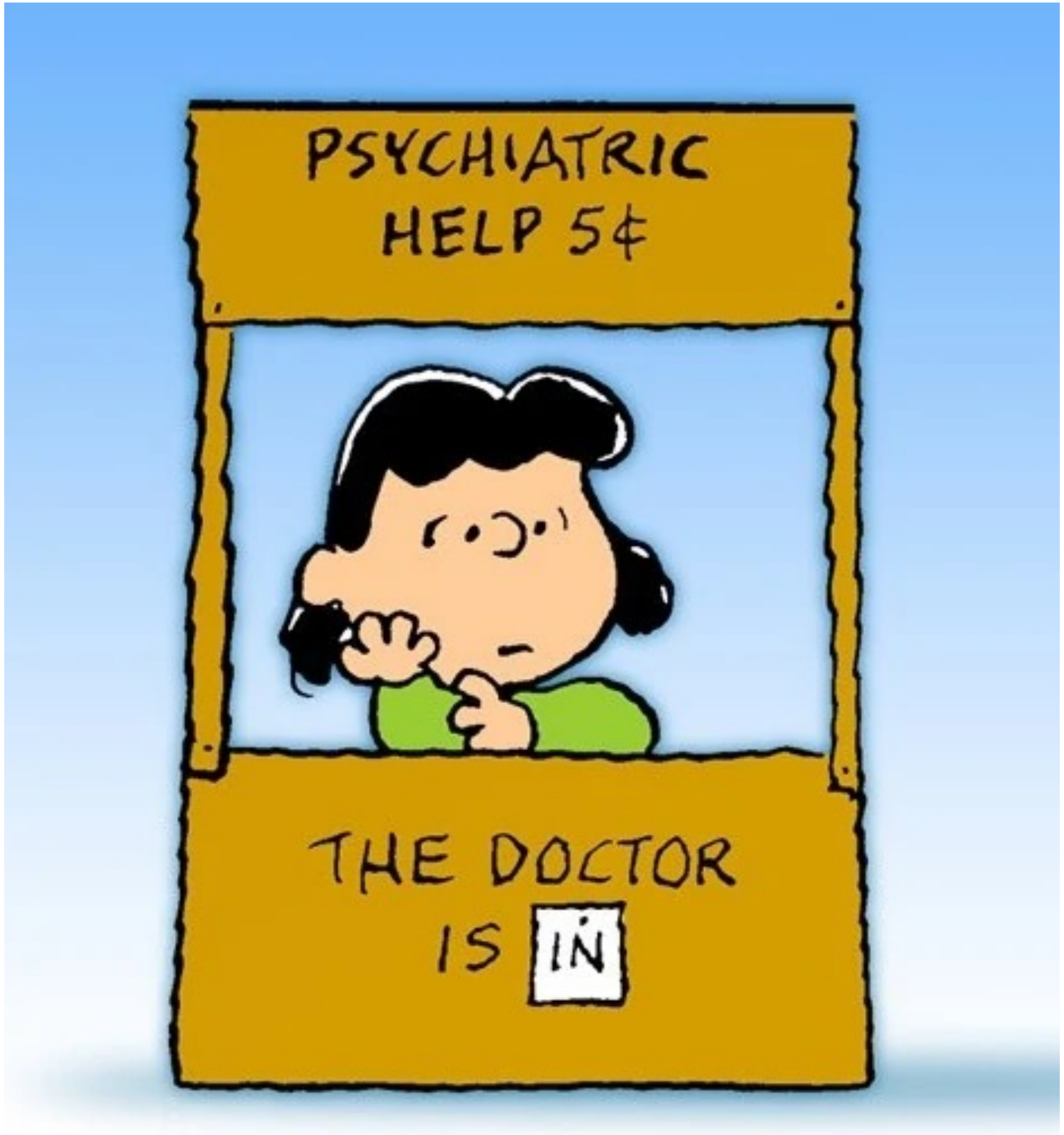
But what about repeated measures ANOVA?

- Repeated measures ANOVA is fine
- But:
- Any missing data require deletion of the entire case
- Not good when repeated observations per unit vary (3 to 7 dendrite segments per neuron: what do you do with that?)

Fine doesn't mean fine.
The scale goes: great, good, okay, not okay, I hate you, fine.

Multilevel modeling

- Multilevel modeling / linear mixed models / hierarchical linear models
- More general and flexible framework that subsumes many analysis strategies
- Can become complex quickly but is extremely useful for many kinds of neuroscience data where simpler analyses require unrealistic assumptions or over-simplify the problem


PSYCHIATRIC
HELP 5¢
THE DOCTOR
IS IN

Regression

$$y_i = \beta_0 + \underbrace{\beta_1 x_{1i} + \varepsilon_i}_{\substack{\text{decompose into effects of} \\ \text{individual and predictor}}}$$

Multilevel modeling

$$y_i = \beta_0 + \underbrace{\beta_1 x_{1i} + \varepsilon_i}_{\text{decompose into effects of individual and group/cluster}}$$

$$y_{ij} = \beta_{0j} + \beta_{1j} x_{1ij} + \varepsilon_{ij}$$

each *i*th observation is one of *j* clusters

Multilevel modeling

$$y_i = \beta_0 + \underbrace{\beta_1 x_{1i} + \varepsilon_i}_{\substack{\text{decompose into effects of} \\ \text{individual and group/cluster}}}$$

$$y_{ij} = \beta_{0j} + \beta_{1j} x_{1ij} + \varepsilon_{ij}$$

each *i*th observation is one of *j* clusters

$$\beta_{0j} = \gamma_{00} + u_{0j} \qquad \beta_{1j} = \gamma_{10}$$

grand mean

cluster residual

regression coeff of variable *within* cluster

Multilevel modeling

$$y_i = \beta_0 + \underbrace{\beta_1 x_{1i} + \varepsilon_i}_{\text{decompose into effects of individual and group/cluster}}$$

"level 1"

$$y_{ij} = \beta_{0j} + \beta_{1j} x_{1ij} + \varepsilon_{ij}$$

each i^{th} observation is one of *j* clusters

"level 2"

$$\beta_{0j} = \gamma_{00} + u_{0j}$$

grand mean

cluster residual

$$\beta_{1j} = \gamma_{10}$$

regression coeff of variable *within* cluster

Multilevel modeling

$$y_i = \beta_0 + \underbrace{\beta_1 x_{1i} + \varepsilon_i}_{\substack{\text{decompose into effects of} \\ \text{individual and group/cluster}}}$$

"level 1"

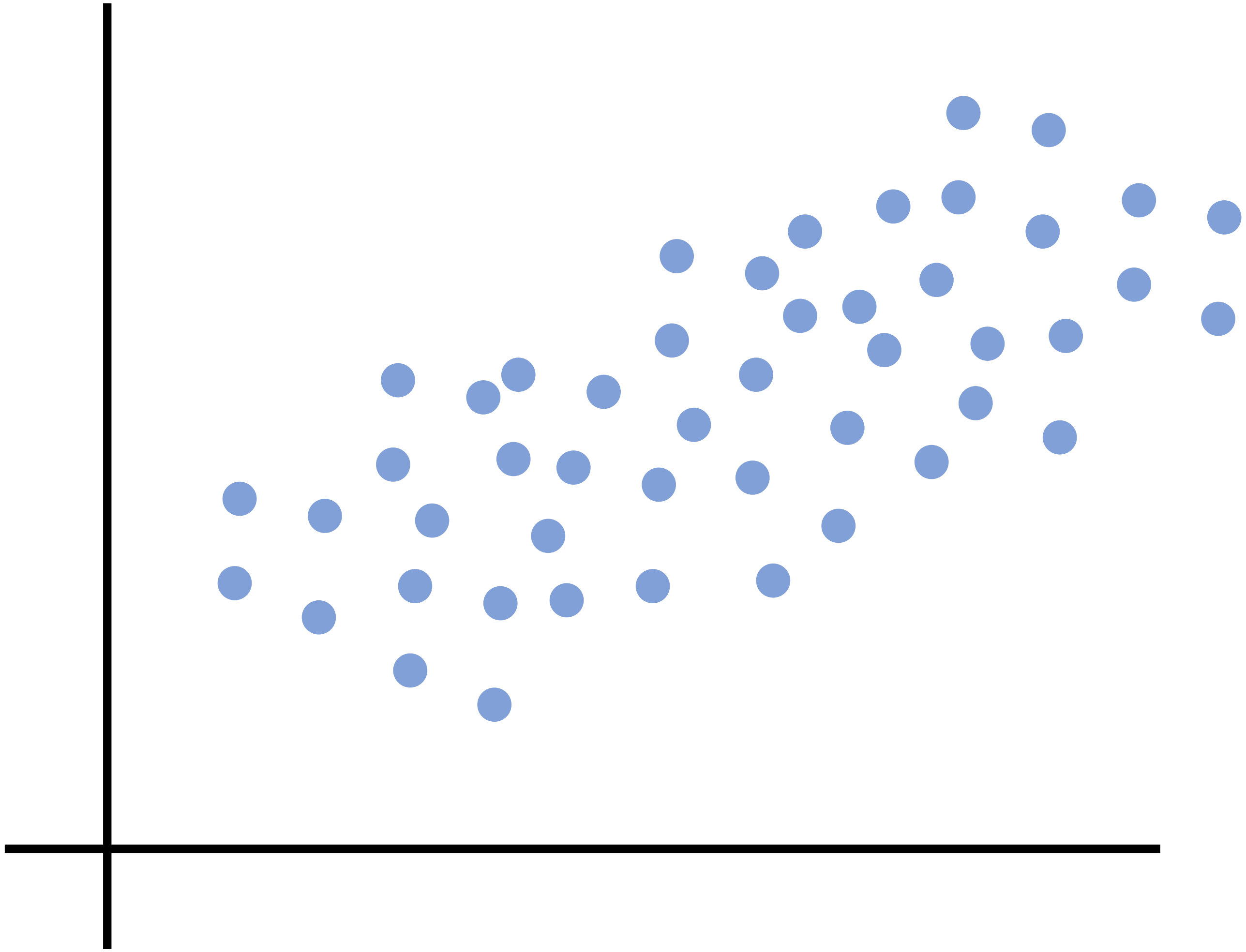
$$y_{ij} = \beta_{0j} + \beta_{1j} x_{1ij} + \varepsilon_{ij}$$

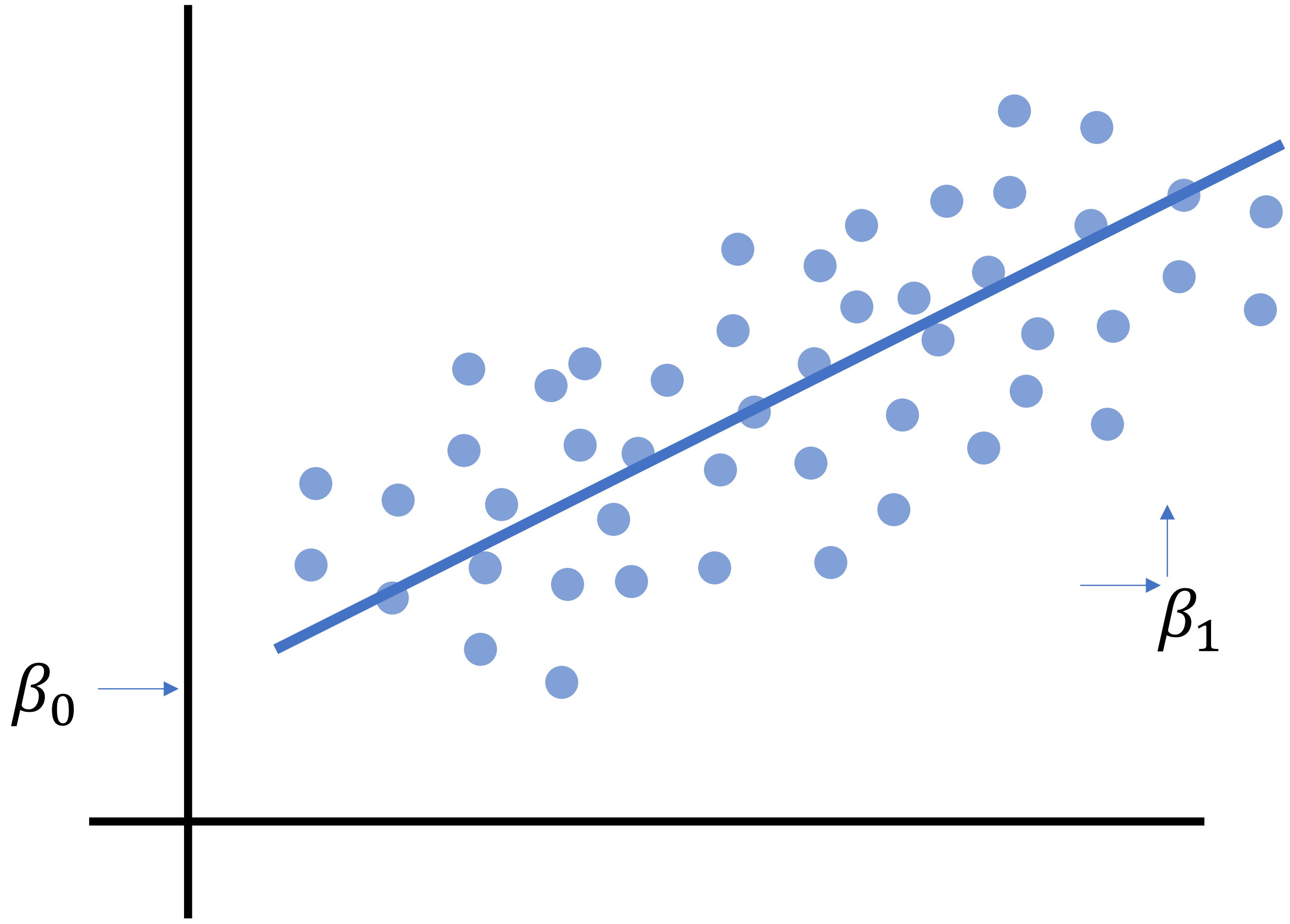
each *i*th observation is one of *j* clusters

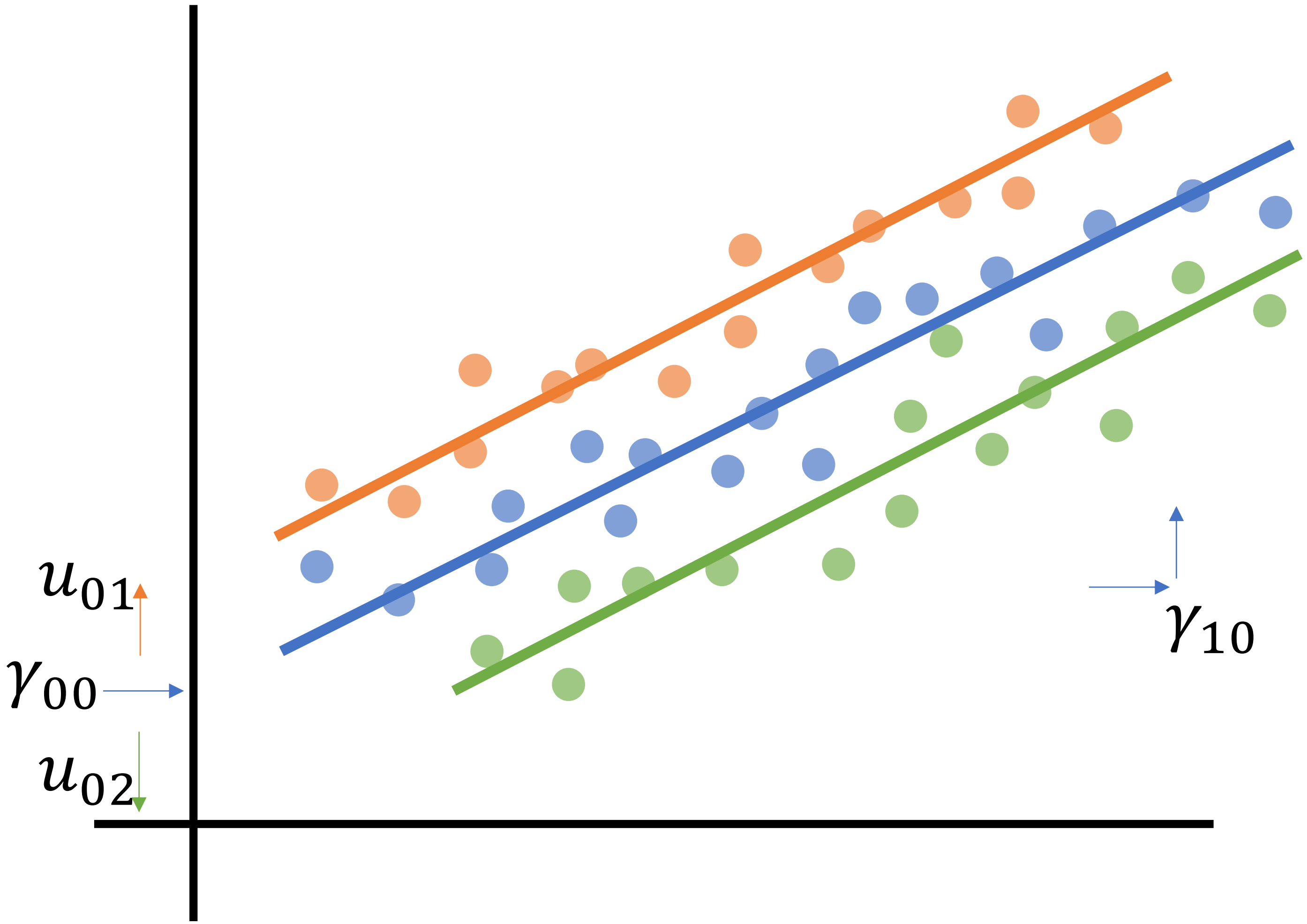
"level 2"

$$\beta_{0j} = \underset{\substack{\text{grand} \\ \text{mean}}}{\gamma_{00}} + \underset{\substack{\text{cluster} \\ \text{residual}}}{u_{0j}}$$

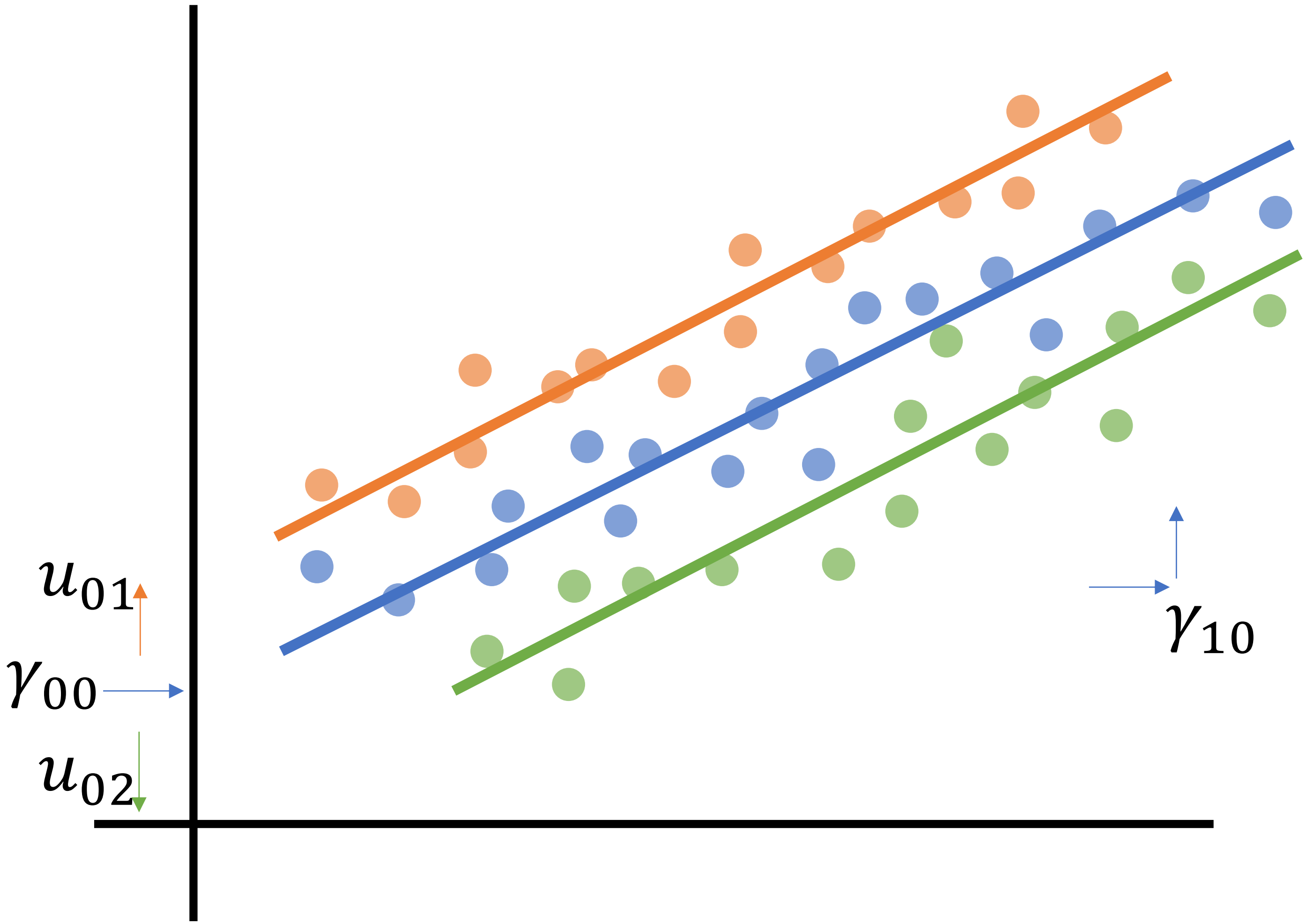
$$\beta_{1j} = \underset{\substack{\text{regression coeff} \\ \text{of variable} \\ \textit{within}\text{ cluster}}}{\gamma_{10}} + \underset{\substack{\text{regression coeff} \\ \text{may vary between} \\ \text{clusters!}}}{\gamma_{1j}}$$



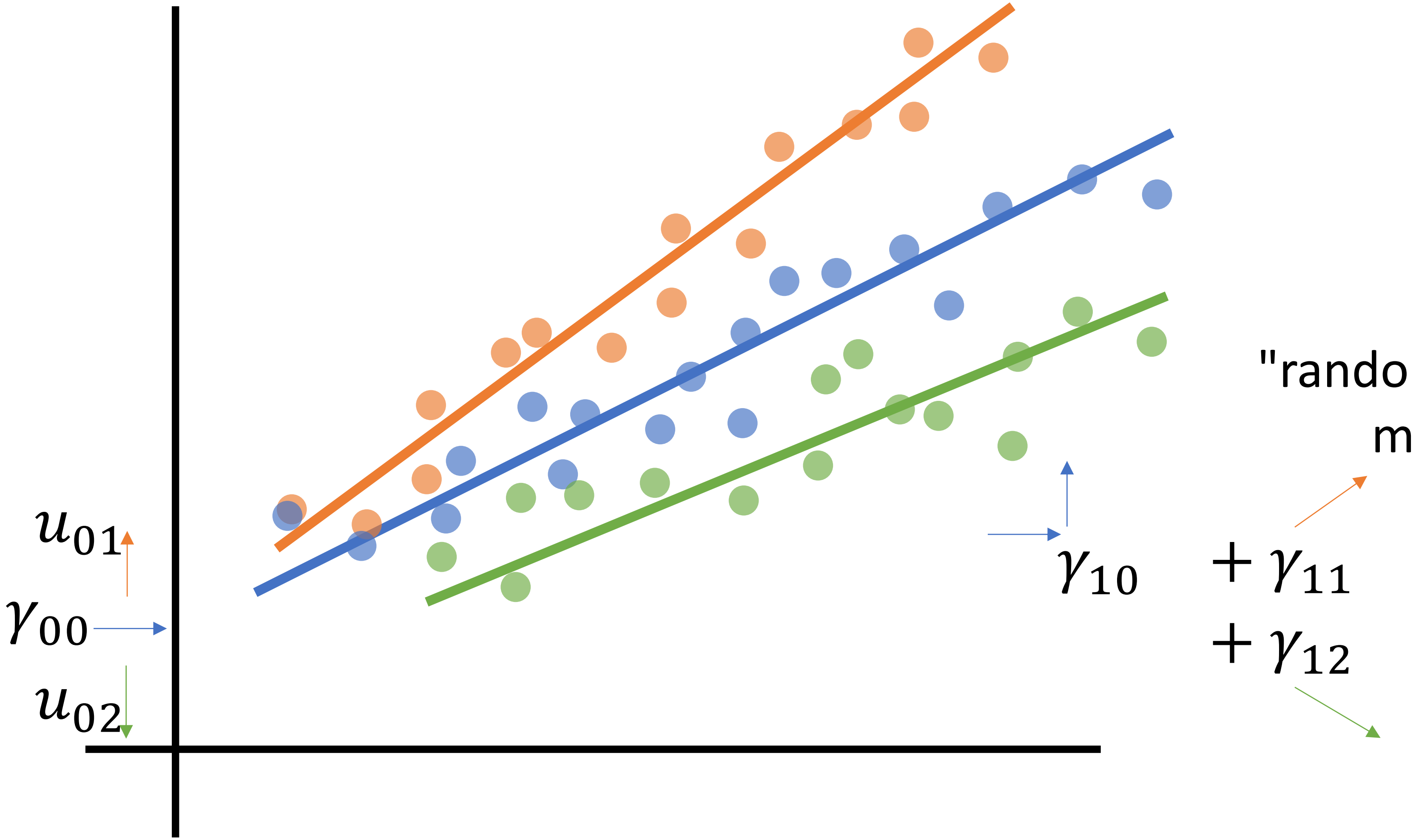

β_0
β_1


u_{01}
γ_{00}
u_{02}
γ_{10}

random intercepts
shift line by
cluster mean



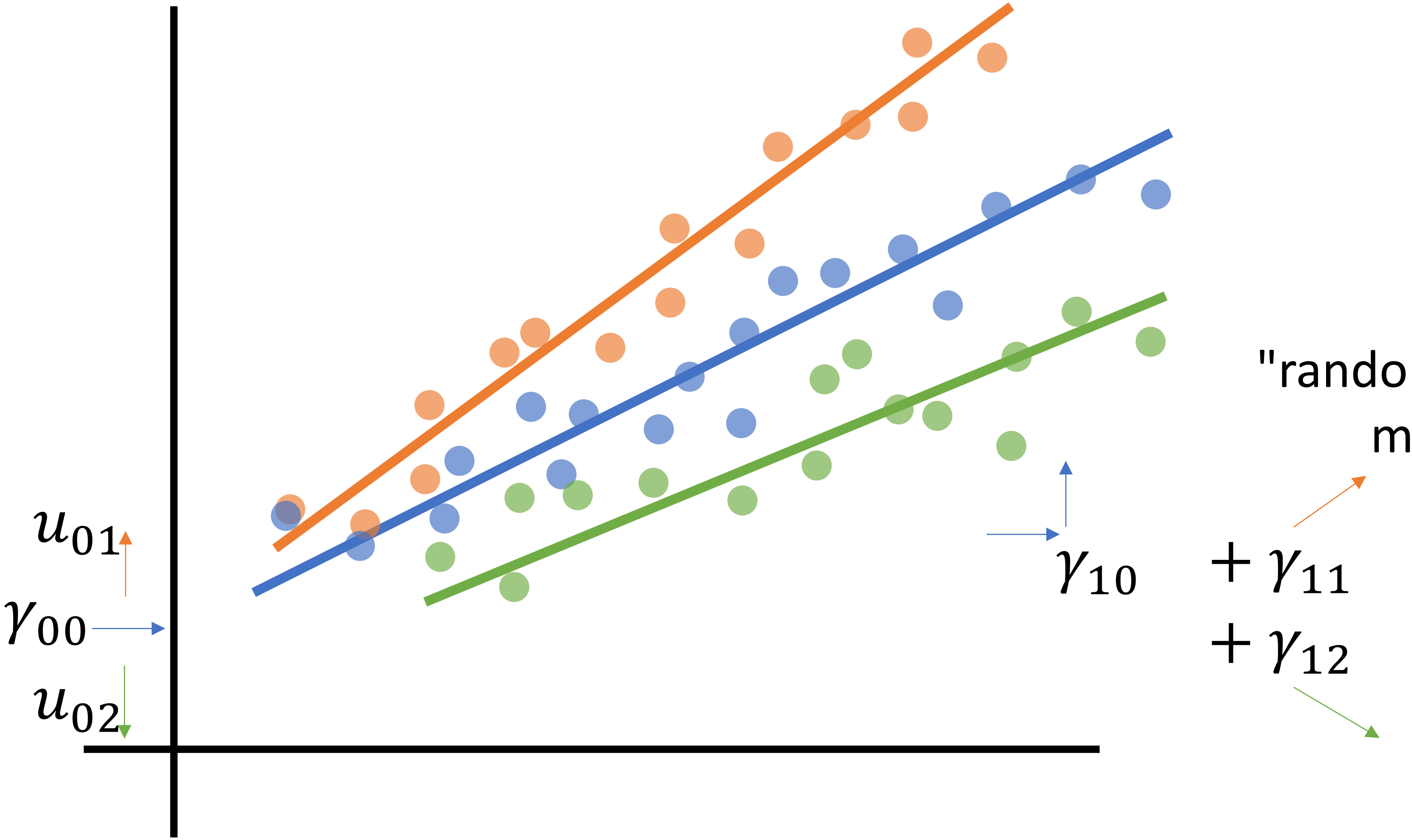
"parallel slopes"
model



random intercepts
shift line by
cluster mean

"random slopes"
model

random intercepts shift line by cluster mean



"random slopes" model

Conceptually, this is not all that different from interactions (e.g. of bill length and depth with species)
What differs in this framework is that multiple data points may come from **same unit of observation**

Fixed and random effects

- "Linear mixed models"
  - "Linear": parameters are linear (adding things together)
  - "Mixed": mix of "fixed" and "random" effects
- Fixed effects: usually of intrinsic interest
  - Can be continuous or categorical
  - Unknown constant parameters relating to outcome variable: these are what we're estimating
  - Exhaustively sampled (you have to be wild type or control)
- Random effects: usually not of intrinsic interest
  - Unobserved random variables usually assumed to be normally distributed
  - Each level not of interest: randomly sampled from larger population (mice, neurons, classrooms)
  - *Introduces dependence into errors that requires explicit modeling*

Fixed and random effects

```
lm(outcome ~ predictor1 + predictor2, data = my_data) %>% summary()
lm(outcome ~ predictor1 + predictor2, data = my_data) %>% anova()
```

Fixed and random effects

```
lm(outcome ~ predictor1 + predictor2, data = my_data) %>% summary()
lm(outcome ~ predictor1 + predictor2, data = my_data) %>% anova()
```

```
library(lme4)
lmer(outcome ~ fixed1 + fixed2 + (1|random), data = my_data) %>%
summary()
lmer(outcome ~ fixed1 + fixed2 + (1|random), data = my_data) %>%
anova()
```

"lmer" = "linear mixed effects regression"

Fixed and random effects

```
lm(outcome ~ predictor1 + predictor2, data = my_data) %>% summary()
lm(outcome ~ predictor1 + predictor2, data = my_data) %>% anova()
```

```
library(lme4)
lmer(outcome ~ fixed1 + fixed2 + (1|random), data = my_data) %>%
summary()
lmer(outcome ~ fixed1 + fixed2 + (1|random), data = my_data) %>%
anova()
```

```
lmer(outcome ~ fixed1 + fixed2 + (fixed1|random) + (fixed2|random),
data = my_data) %>% anova()
lmer(outcome ~ fixed1 * fixed2 + (fixed1*fixed2|random), data =
my_data) %>% anova()
```

Fixed and random effects

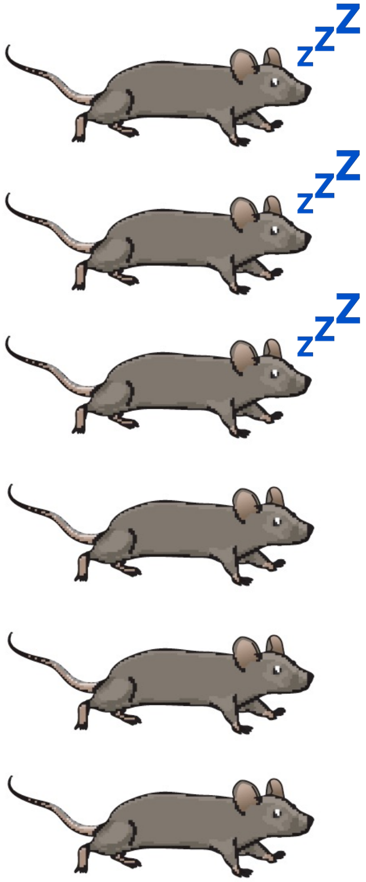
```
lm(outcome ~ predictor1 + predictor2, data = my_data) %>% summary()
lm(outcome ~ predictor1 + predictor2, data = my_data) %>% anova()
```

```
library(lme4)
lmer(outcome ~ fixed1 + fixed2 + (1|random), data = my_data) %>%
summary()
lmer(outcome ~ fixed1 + fixed2 + (1|random), data = my_data) %>%
anova()
```

```
lmer(outcome ~ fixed1 + fixed2 + (fixed1|random) + (fixed2|random),
data = my_data) %>% anova()
lmer(outcome ~ fixed1 * fixed2 + (fixed1*fixed2|random), data =
my_data) %>% anova()
```

essentially, regression for random effects; 1 = intercept

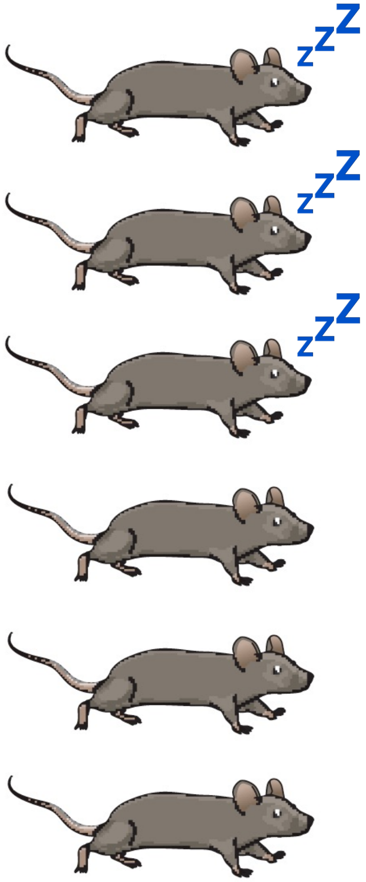
Fixed and random effects



isoflurane
N = 3 dams

control
N = 3 dams

Fixed and random effects



isoflurane
N = 3 dams

→ 4 male pups, 5 female

→ 6 male pups, 4 female

→ 3 male pups, 8 female

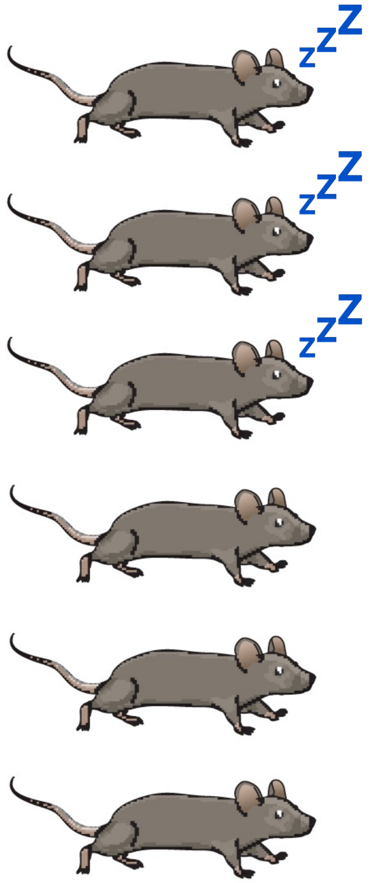
control
N = 3 dams

→ 7 male pups, 4 female

→ 3 male pups, 3 female

→ 6 male pups, 8 female

Fixed and random effects



isoflurane
N = 3 dams

→ 4 male pups, 5 female

→ 6 male pups, 4 female

→ 3 male pups, 8 female

control
N = 3 dams

→ 7 male pups, 4 female

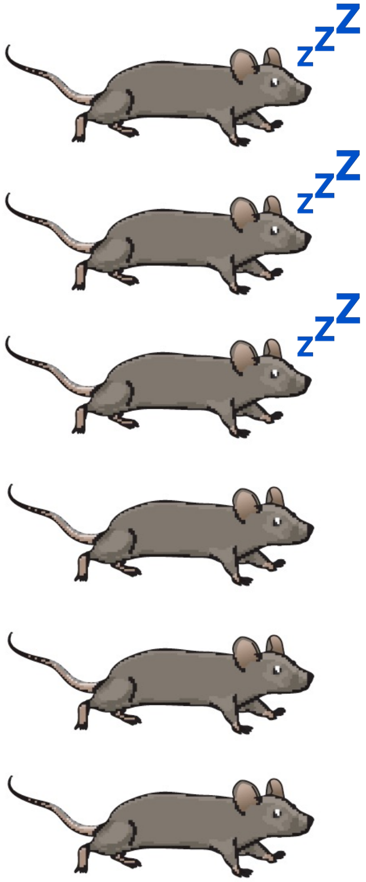
→ 3 male pups, 3 female

→ 6 male pups, 8 female

N = 30 isoflurane, 31 control?

N = 3 isoflurane, 3 control?

Fixed and random effects



isoflurane
N = 3 dams

→ 4 male pups, 5 female

→ 6 male pups, 4 female

→ 3 male pups, 8 female

control
N = 3 dams

→ 7 male pups, 4 female

→ 3 male pups, 3 female

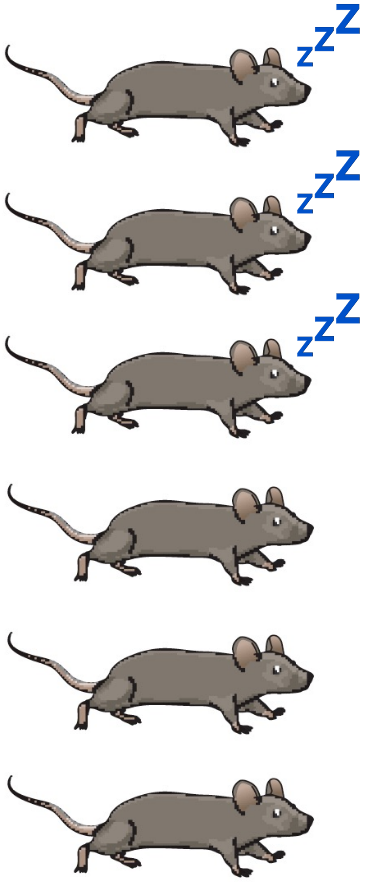
→ 6 male pups, 8 female

N = 30 isoflurane, 31 control?

N = 3 isoflurane, 3 control?

X pups from different litters not independent

Fixed and random effects



isoflurane
N = 3 dams

control
N = 3 dams

→ 4 male pups, 5 female

→ 6 male pups, 4 female

→ 3 male pups, 8 female

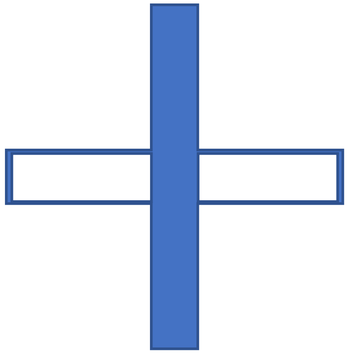
→ 7 male pups, 4 female

→ 3 male pups, 3 female

→ 6 male pups, 8 female

N = 30 isoflurane, 31 control?

N = 3 isoflurane, 3 control?

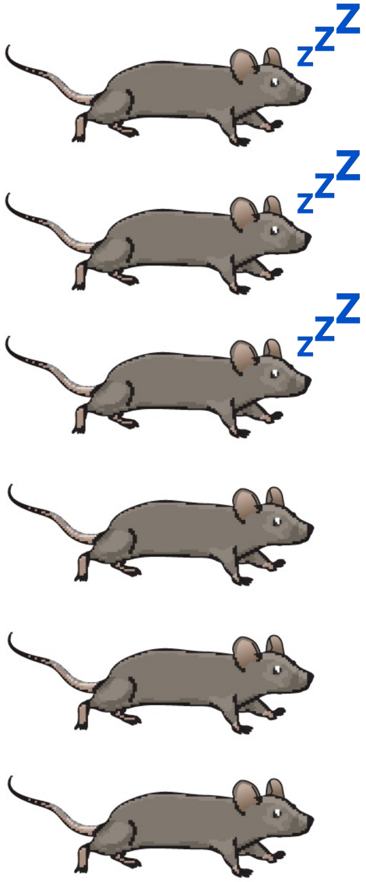


X pups from different litters not independent

NO.

```
lm(openarms ~ iso + sex + litter, data = babyrats)
lm(openarms ~ iso * sex * litter, data = babyrats)
```

Fixed and random effects



isoflurane
N = 3 dams

→ 4 male pups, 5 female

→ 6 male pups, 4 female

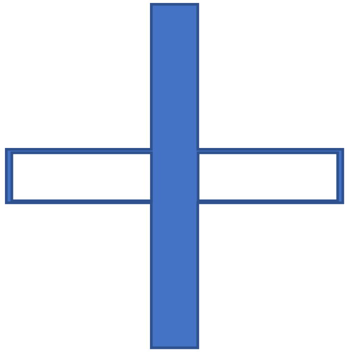
→ 3 male pups, 8 female

control
N = 3 dams

→ 7 male pups, 4 female

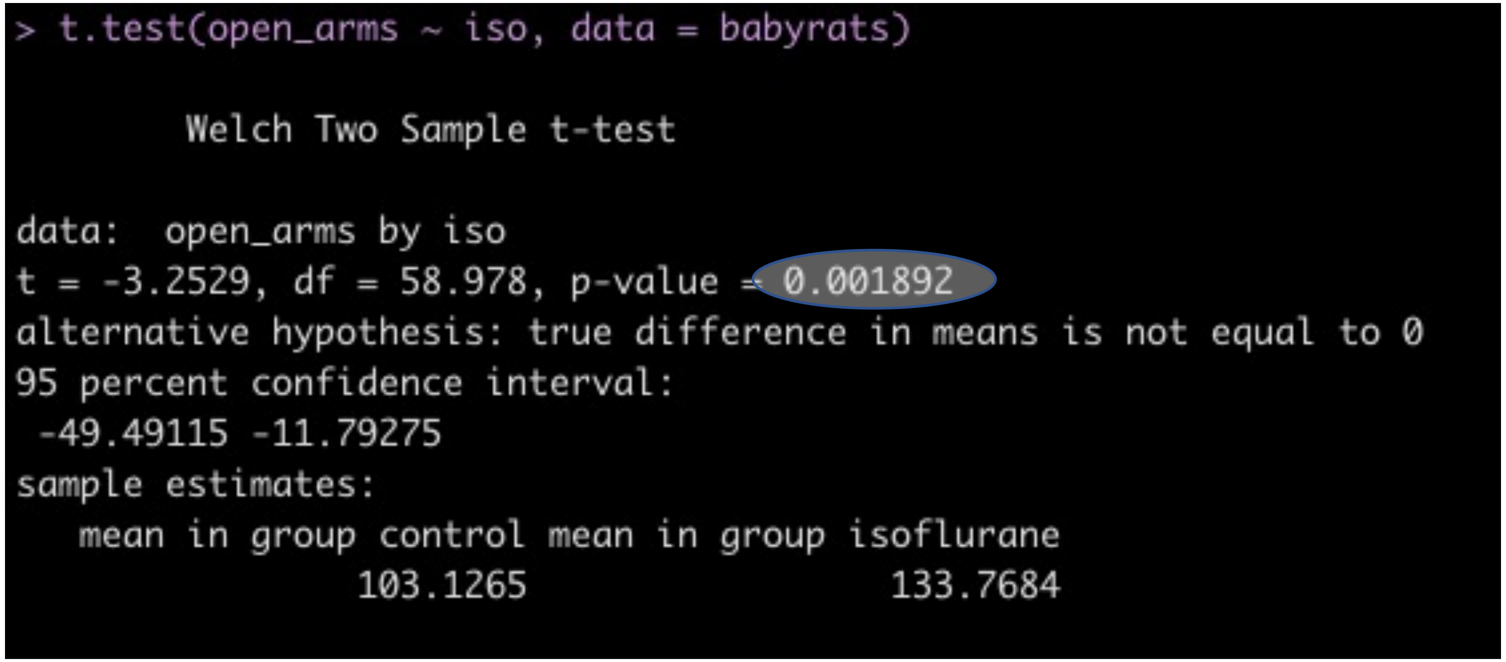
→ 3 male pups, 3 female

→ 6 male pups, 8 female



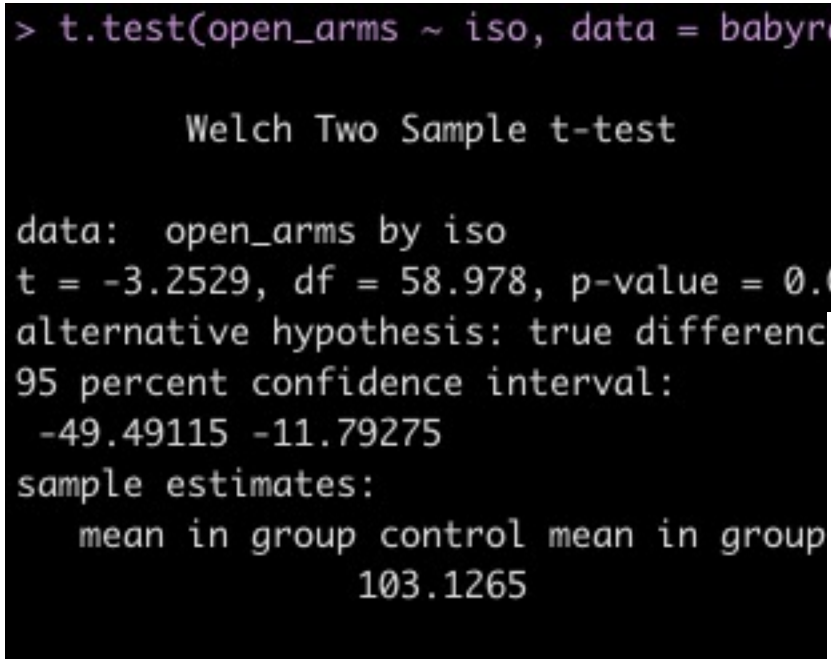
```
library(lme4)
lmer(openarms ~ iso * sex + (1|litter), data = babyrats)
```

better

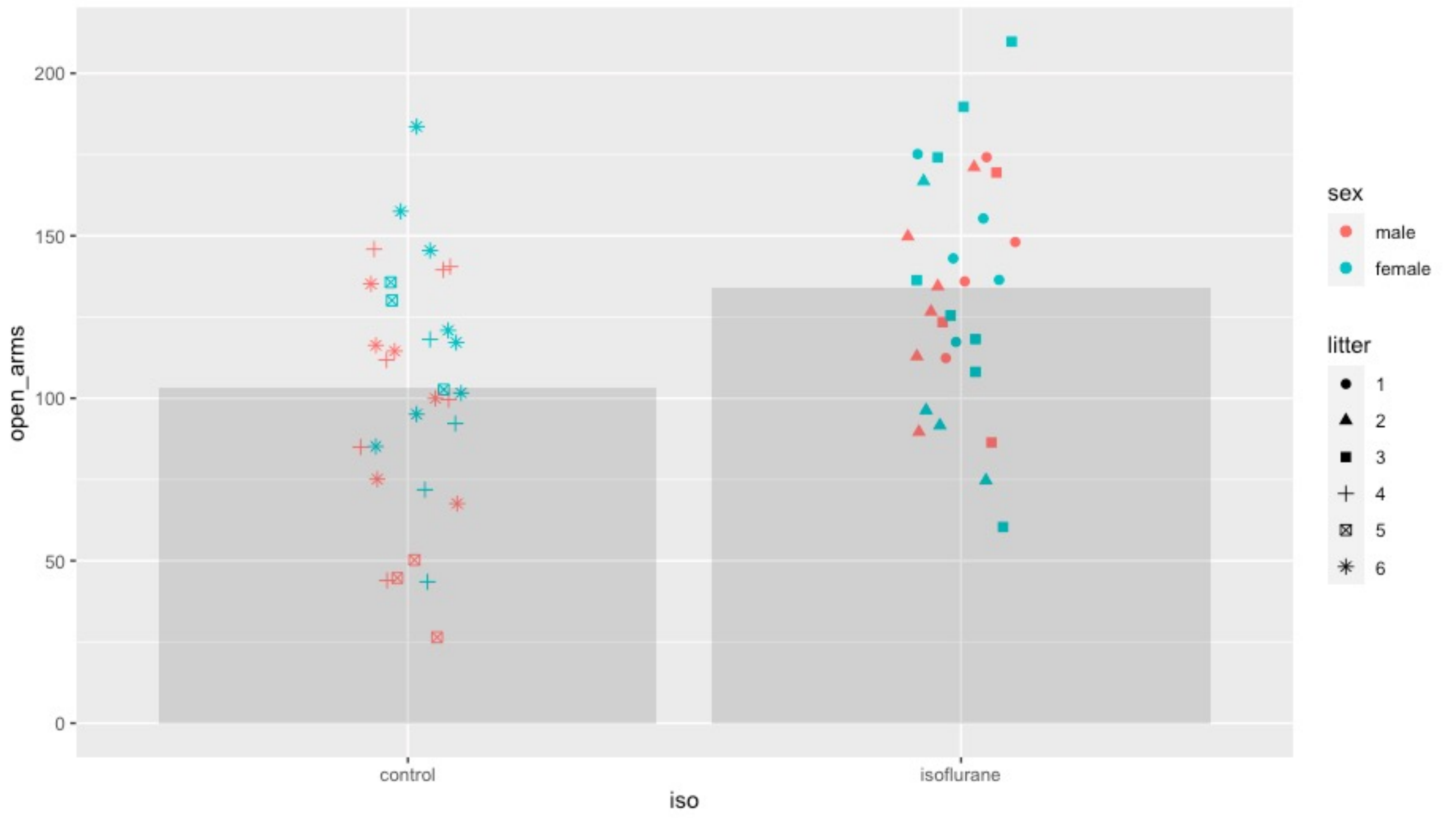


```
> t.test(open_arms ~ iso, data = babyrats)

	Welch Two Sample t-test

data:  open_arms by iso
t = -3.2529, df = 58.978, p-value = 0.001892
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -49.49115 -11.79275
sample estimates:
   mean in group control mean in group isoflurane
                103.1265                 133.7684
```

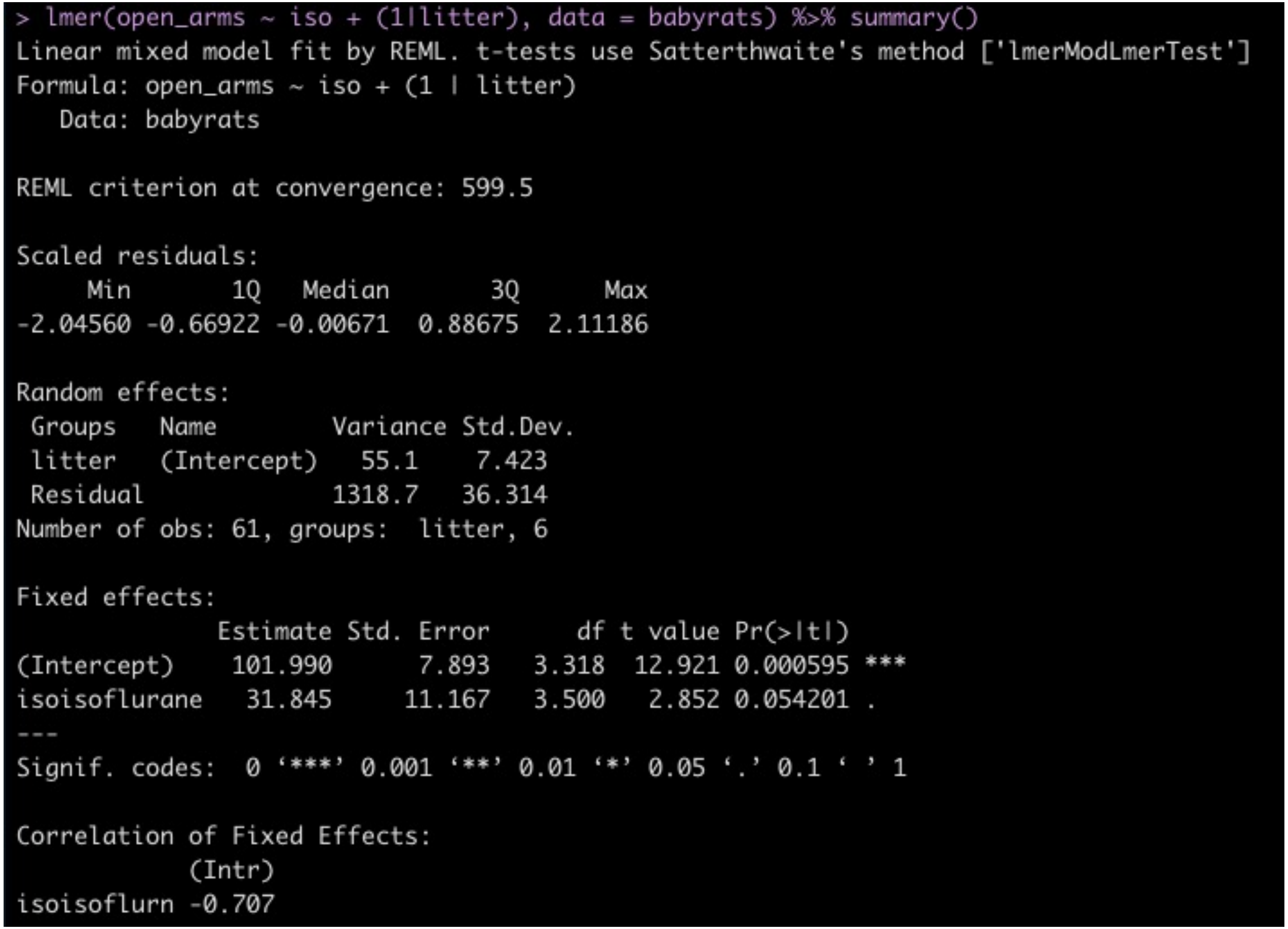
20220202_LMM_example.R



```
> t.test(open_arms ~ iso, data = babyrats)

	Welch Two Sample t-test

data:  open_arms by iso
t = -3.2529, df = 58.978, p-value = 0.001892
alternative hypothesis: true differenc
95 percent confidence interval:
 -49.49115 -11.79275
sample estimates:
   mean in group control mean in group
             103.1265
```

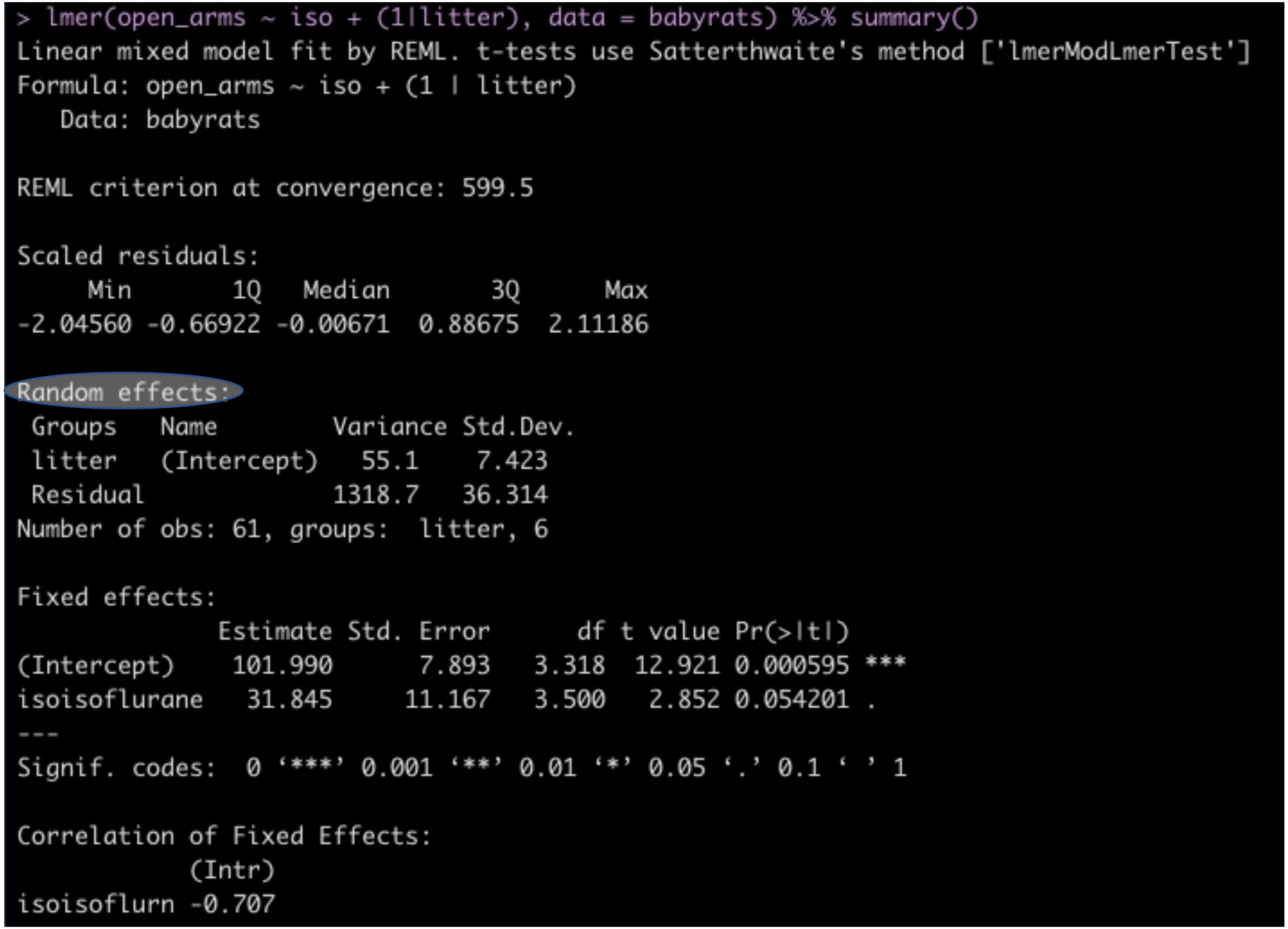


20220202_LMM_example.R



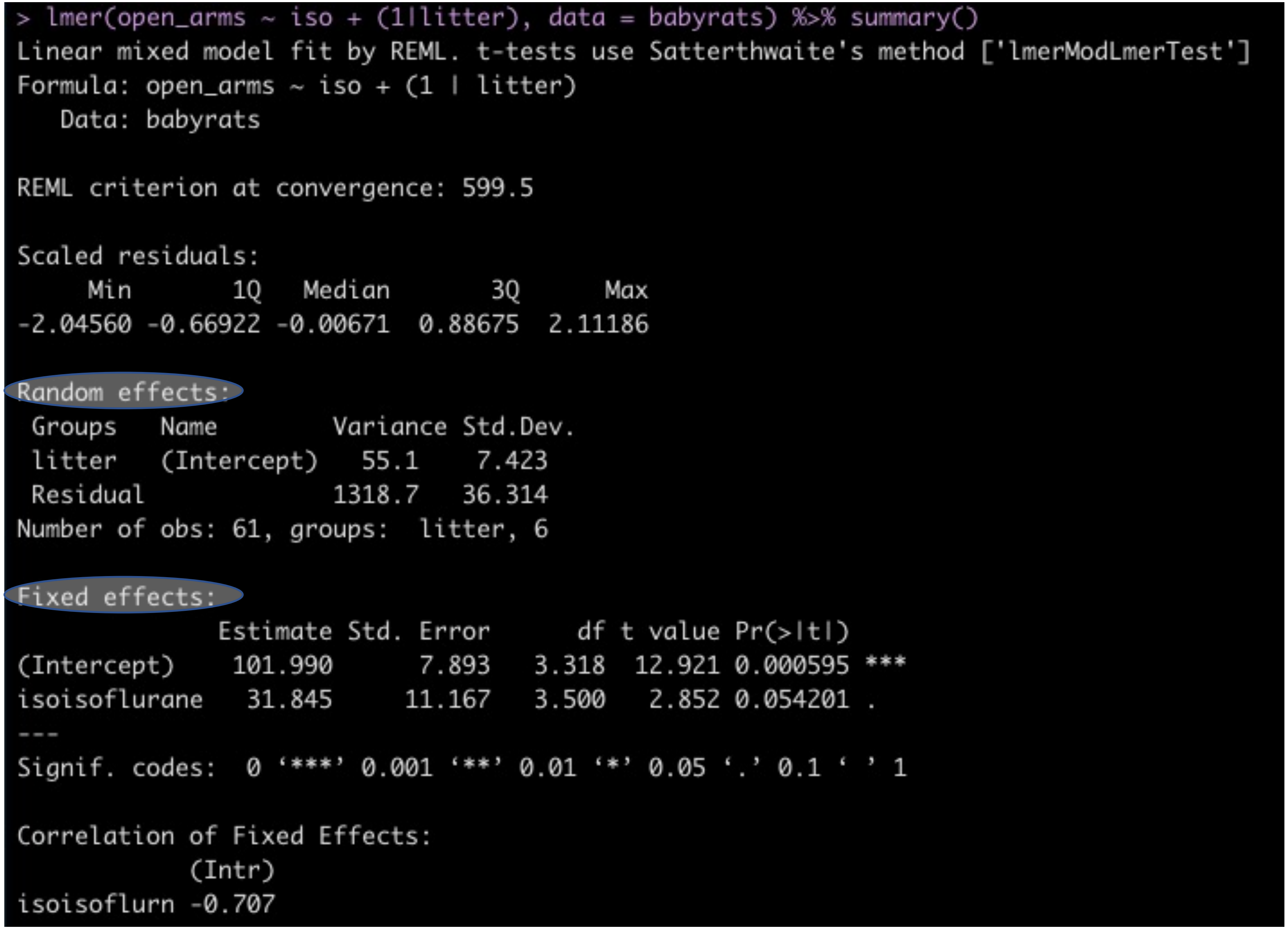
```
> lmer(open_arms ~ iso + (1|litter), data = babyrats) %>% summary()
Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: open_arms ~ iso + (1 | litter)
   Data: babyrats

REML criterion at convergence: 599.5

Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.04560 -0.66922 -0.00671  0.88675  2.11186

Random effects:
 Groups   Name        Variance Std.Dev.
 litter   (Intercept)   55.1    7.423
 Residual             1318.7   36.314
Number of obs: 61, groups:  litter, 6

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)
(Intercept)    101.990      7.893   3.318  12.921 0.000595 ***
isoisoflurane   31.845     11.167   3.500   2.852 0.054201 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr)
isoisoflurn -0.707
```

20220202_LMM_example.R



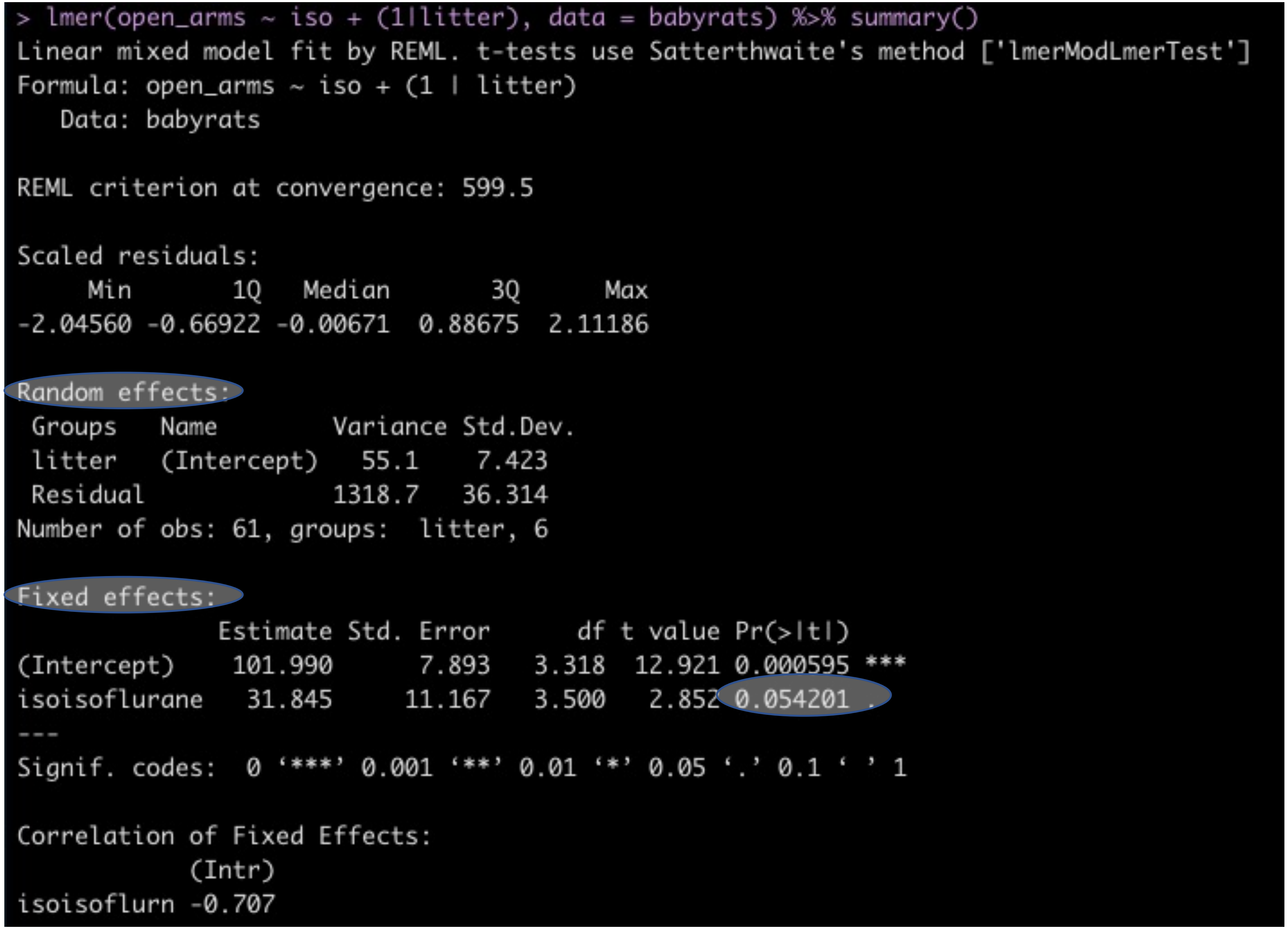
```
> lmer(open_arms ~ iso + (1|litter), data = babyrats) %>% summary()
Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: open_arms ~ iso + (1 | litter)
   Data: babyrats

REML criterion at convergence: 599.5

Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.04560 -0.66922 -0.00671  0.88675  2.11186

Random effects:
 Groups   Name        Variance Std.Dev.
 litter   (Intercept)   55.1    7.423
 Residual             1318.7   36.314
Number of obs: 61, groups:  litter, 6

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)
(Intercept)    101.990      7.893   3.318  12.921 0.000595 ***
isoisoflurane   31.845     11.167   3.500   2.852 0.054201 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr)
isoisoflurn -0.707
```

20220202_LMM_example.R



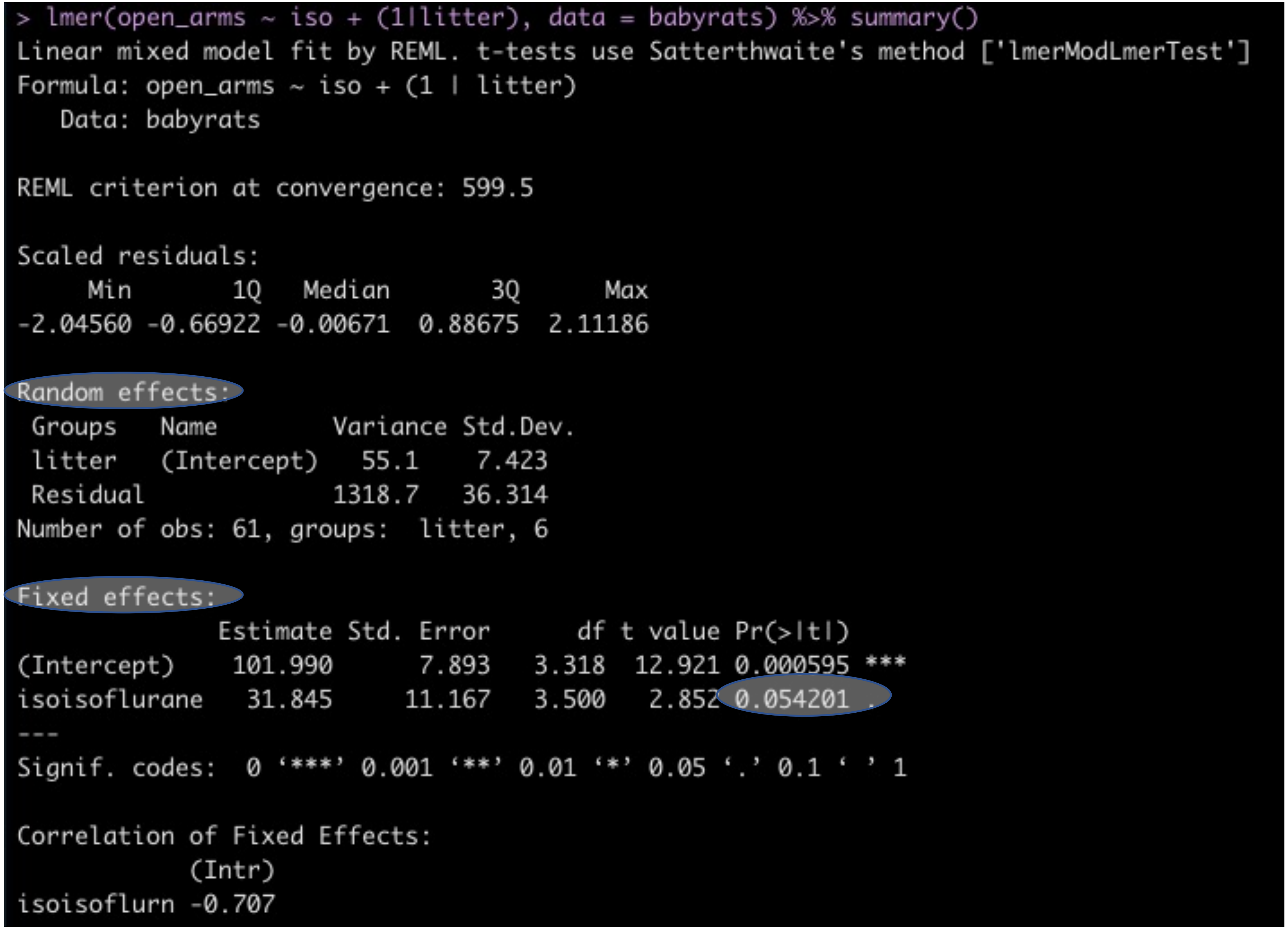
```
> lmer(open_arms ~ iso + (1|litter), data = babyrats) %>% summary()
Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: open_arms ~ iso + (1 | litter)
   Data: babyrats

REML criterion at convergence: 599.5

Scaled residuals:
     Min       1Q   Median       3Q      Max 
-2.04560 -0.66922 -0.00671  0.88675  2.11186 

Random effects:
 Groups   Name        Variance Std.Dev.
 litter   (Intercept)   55.1    7.423  
 Residual             1318.7   36.314  
Number of obs: 61, groups:  litter, 6

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)    
(Intercept)    101.990      7.893   3.318  12.921 0.000595 ***
isoisoflurane   31.845     11.167   3.500   2.852 0.054201 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr)
isoisoflurn -0.707
```

20220202_LMM_example.R



```
> lmer(open_arms ~ iso + (1|litter), data = babyrats) %>% summary()
Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: open_arms ~ iso + (1 | litter)
   Data: babyrats

REML criterion at convergence: 599.5

Scaled residuals: 
     Min       1Q   Median       3Q      Max 
-2.04560 -0.66922 -0.00671  0.88675  2.11186 

Random effects:
 Groups   Name        Variance Std.Dev.
 litter   (Intercept)   55.1    7.423  
 Residual             1318.7   36.314  
Number of obs: 61, groups:  litter, 6

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)    
(Intercept)    101.990      7.893   3.318  12.921 0.000595 ***
isoisoflurane   31.845     11.167   3.500   2.852 0.054201 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr)
isoisoflurn -0.707
```

20220202_LMM_example.R



```
> lmer(open_arms ~ iso + (1|litter), data = babyrats) %>% summary()
Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: open_arms ~ iso + (1 | litter)
   Data: babyrats

REML criterion at convergence: 599.5

Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.04560 -0.66922 -0.00671  0.88675  2.11186

Random effects:
 Groups   Name        Variance Std.Dev.
 litter   (Intercept)   55.1    7.423
 Residual             1318.7   36.314
Number of obs: 61, groups:  litter, 6

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)
(Intercept)    101.990      7.893   3.318  12.921 0.000595 ***
isoisoflurane   31.845     11.167   3.500   2.852 0.054201 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr)
isoisoflurn -0.707
```

not accounting for correlated errors / residuals in rats from same litter results in artificially small standard error estimates for between-group effect

when this is accounted for, between-group effect just misses .05 threshold

20220202_LMM_example.R

What could go wrong in LMMs? (A lot)

- Singularity problems
    - misspecified model: multicollinearity, insufficient data
    - zero variance in random effects (cannot attribute any variability to cluster)
    - similar problems to regular regression

- Estimate full model and reduce?
- Limit to effects of interest?
- Center / scale predictor variables
    - grand mean centering vs cluster centering (depends on question)

What sample size?

- "N" (number of clusters) and "n" (number of observations per cluster)
- Some theoretical work provides general guidance
- Related to intraclass correlation (ICC) - similarity of observations within clusters
  - ICC = 0 observations are as similar across clusters as within
  - ICC = 1 observations within clusters are identical
- If ICC = 0 then you just need more observations
- As ICC approaches 1 increasing number of observations per cluster becomes less effective – you need more clusters

Repeated Measures

- Similar to any other clustered data – repeated observations on same unit of analysis (mouse, neuron, ...)

- Treat repeated measure as factor or numeric?
  - expect similar change across evenly spaced units of time?
  - usually treat as discrete occasions of measurement

- Factor – coefficients will be relative to first occasion
- Can use ordered factor to estimate polynomial trends

About those p-values

- type I vs type III sums of squares
- briefly: type I are sequential (take out all the variance associated with the first effect, then calculate the variance associated with the next effect, ...) whereas type III are simultaneous (all effects account for the variability present in all the others)
- type I do not take into account different levels of effects (interactions)
- this is more of a problem when designs unbalanced (different N in different conditions)

- SPSS gives type III sums of squares
- summary(model) gives type I for lm type model
- car::**A**nova(model, type = 3) gives type III for lm type model

http://www.utstat.utoronto.ca/reid/sta442f/2009/typeSS.pdf