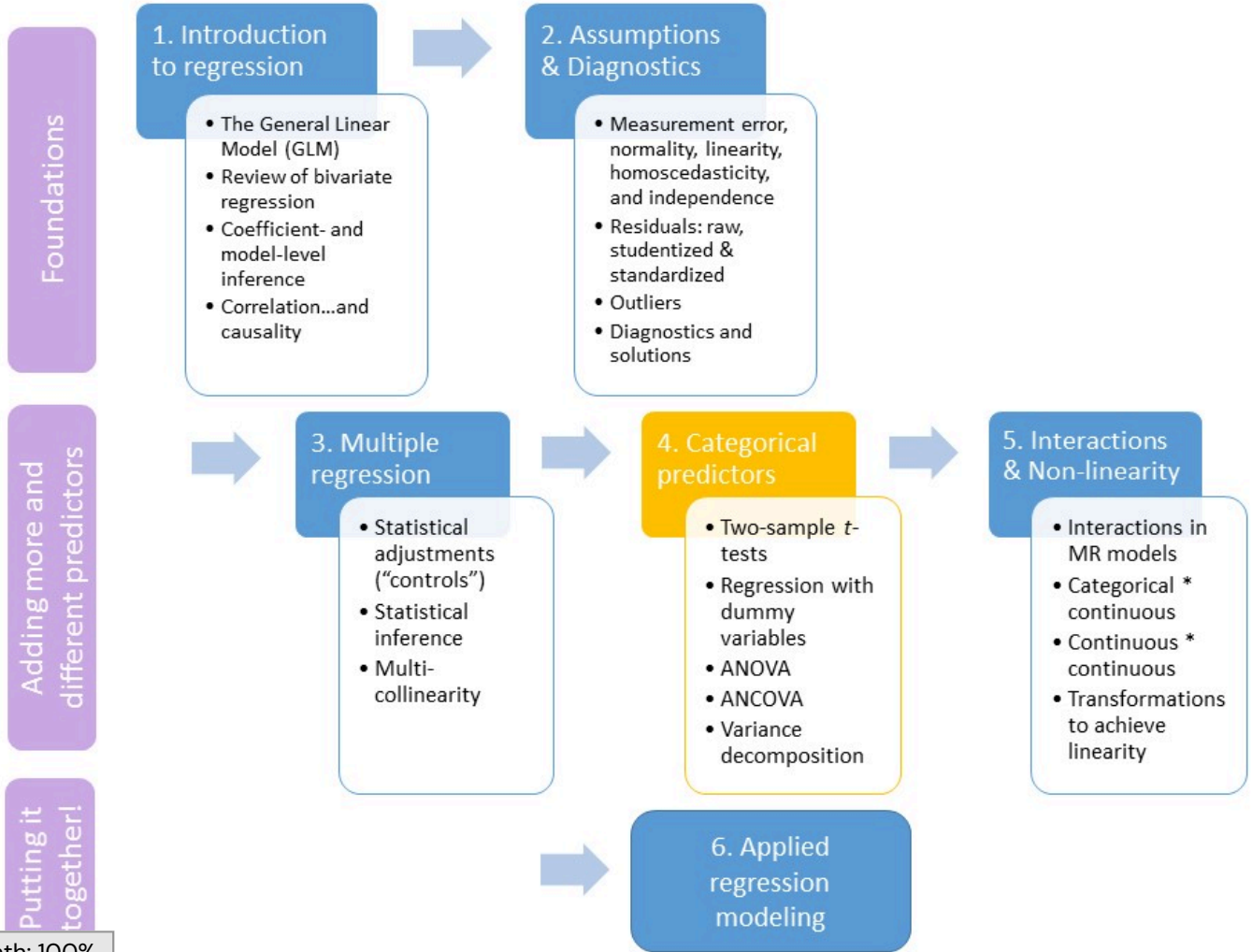
Categorical predictors and ANOVA/ANCOVA

EDUC 643: Unit 4

David D. Liebowitz

Processing math: 100%

Roadmap



Processing math: 100%

Unit goals

- Describe the relationship between dichotomous and polychotomous variables and convert variables between these forms, as necessary
- Conduct a two-sample t-test
- Describe the relationship between a two-sample t-test and regressing a continuous outcome on a dichotomous predictor
- Estimate a regression with one dummy variable as a predictor and interpret the results (including when the reference category changes)
- Estimate a multiple regression model with several continuous and dummy variables and interpret the results
- Estimate an ANOVA model and interpret the within- and between-group variance
    - Do the same for an ANCOVA model, adjusting for additional continuous predictors
- Describe the similarities and differences of Ordinary-Least Squares regression analysis and ANOVA/ANCOVA, and when one would prefer one approach to another
- Describe potential Type I error problems that arise from multiple group comparisons and potential solutions to these problems, including theory, pre-registration, ANOVA and *post-hoc* corrections
- Describe the relationship between different modeling approaches with the General Linear Model family

Processing math: 100%

Categorical variables

Processing math: 100%

Categorical variables

So far, we have only looked at General Linear Models (and their associated OLS regression estimating equations) involving continuous predictors. But what about categorical predictors?

What are categorical predictors?

- Categorical predictors are ***predictors in statistical models whose values denote categories***. Of course, this begs the question...

Processing math: 100%

Categorical predictors

Important distinctions and conventions:

Nominal predictors

- These have *unordered* values
- E.g., gender, religion, political party

Ordinal predictors

- These have *ordered* values
- E.g., grade, developmental stage, education level (?)

Another important distinction: dichotomies (only 2 categories) vs. polychotomies (>2 categories)

Processing math: 100%

Our (new!) motivating question

A team of researchers based at the University of Oregon aimed to understand the effects of the COVID-19 pandemic on students' early literacy skills.[1]

Ann Swindells Professor in Special Education Gina Biancarosa, former UO doctoral students David Fainstein, Chris Ives, and Dave Furjanic, along with CTL Research Manager Patrick Kennedy, used data from assessments of 471,456 students across 1,684 schools on the Dynamic Indicators of Basic Early Literacy Skills (DIBELS) to analyze the extent to which students' Oral Reading Fluency (ORF) scores differed across four waves of DIBELS assessment prior-to and during the pandemic.

Their study is published in *The Elementary School Journal*.

[1] For various reasons, the pandemic is a "lousy natural experiment" for examining the effects of a particular policy response (e.g, virtual schooling). However, it is quite possible to seek to understand its global effects via just the type of analysis Furjanic et al. conducted.

Processing math: 100%

Our data

```
str(dibels)
```

```
## 'data.frame':    5396 obs. of  18 variables:
##  $ sch_deid     : int  10001 10001 10001 10002 10002 10002 10003 10003 10003
##  $ grade        : int  1 2 3 1 2 3 1 2 3 1 ...
##  $ y1_boy_mean  : num  30.6 71.3 102.9 34.1 79.5 ...
##  $ y1_moy_mean  : num  51.8 105.7 132.6 62.3 118.1 ...
##  $ y2_boy_mean  : num  26 71.1 90.5 32.6 68.1 ...
##  $ y2_moy_mean  : num  46.5 97.8 111.2 50.9 98.6 ...
##  $ st           : chr  "AL" "AL" "AL" "AL" ...
##  $ school_magnet: chr  "No" "No" "No" "No" ...
##  $ school_titlei: chr  "Title I targeted assistance school" "Title I targete
##  $ tr_ts        : int  100 105 107 96 82 78 74 92 73 124 ...
##  $ school_enroll: int  312 312 312 256 256 256 239 239 239 418 ...
##  $ frpl_prop    : num  0.1423 0.0423 0.0423 0.1492 0.0492 ...
##  $ pre          : num  41.2 88.5 117.8 48.2 98.8 ...
##  $ post         : num  36.2 84.4 100.9 41.7 83.3 ...
##  $ asian_prop   : num  0.06 0.1238 0.0841 0.1042 0.061 ...
##  $ black_prop   : num  0.14 0.0857 0.1402 0.125 0.2439 ...
##  $ hisp_prop    : num  0.09 0.0571 0.0561 0.0938 0.061 ...
##  $ white_prop   : num  0.61 0.686 0.654 0.583 0.573 ...
```

Processing math: 100%

How similar is our data to Furjanic?

```
# How many unique schools are represented?
length(unique(dibels$sch_deid))
```

```
## [1] 1527
```

```
# How many total students contribute test-scores?
sum(dibels$tr_ts)
```

```
## [1] 396188
```

Processing math: 100%

Mean comparison

```
mean(dibels$pre)
```

```
## [1] 75.12461
```

```
mean(dibels$post)
```

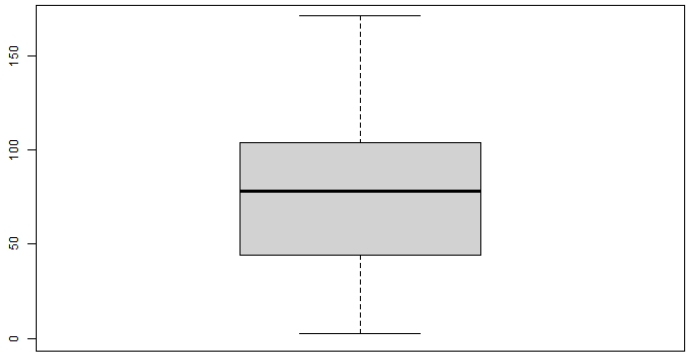
```
## [1] 70.67498
```

Means are 4.5 words per-minute apart.

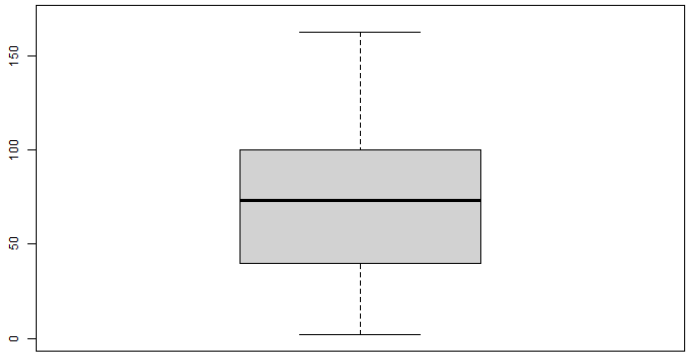
Processing math: 100%

Understanding the distributions

Pre-pandemic



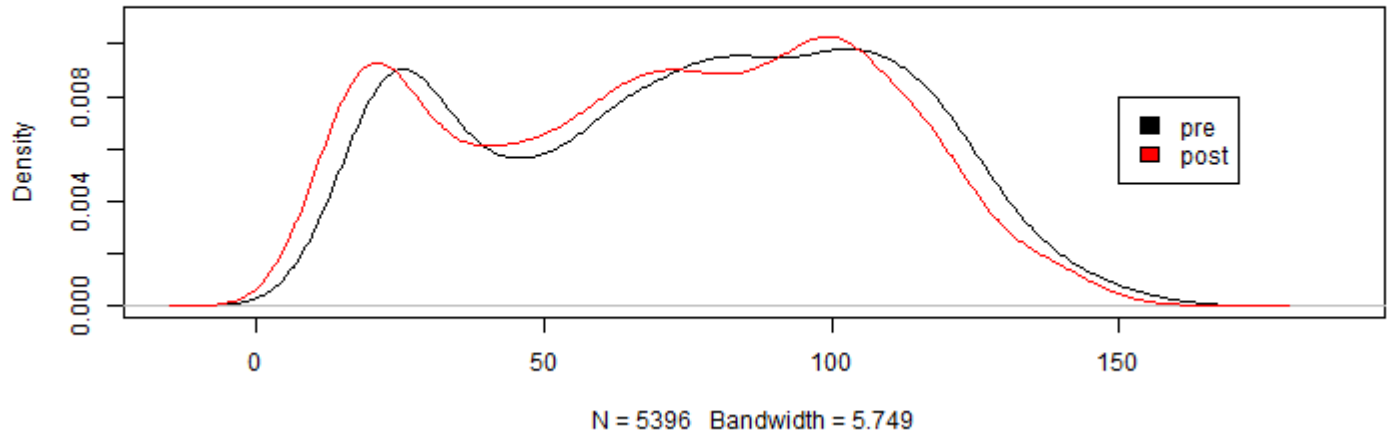
Post-pandemic onset



Processing math: 100%

Understanding the distributions

```
plot(density(dibels$pre), main=" ", sub=NULL, ylim=range(0,0.011))
lines(density(dibels$post), col="red")
legend(150, .008, legend=c("pre", "post"), fill=c("black", "red"))
```



But, as you may by now have anticipated, we are interested in knowing how likely we are to have gotten such a difference by *idiosyncrasy of sampling* from a population of school-grades in which there was no difference. Fortunately, we have just such a tool in our toolbox already. What will the (...) distribution of means of repeatedly drawn samples from a given population be?

Processing math: 100%

Two-sample t-test

```
t.test(dibels$pre, dibels$post)
```

```
##
##     Welch Two Sample t-test
##
## data:  dibels$pre and dibels$post
## t = 6.4962, df = 10790, p-value = 8.602e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  3.106978 5.792286
## sample estimates:
## mean of x mean of y
##  75.12461  70.67498
```

```
# I'm allowing in this t-test for the possibility that my sample in
# each group is of different sizes and has different variance.
# These assumptions affect the precision of my estimates. In some
# settings, particularly experimental ones, I can impose stricter
# assumptions and get more precise estimates.
```

Processing math: 100%

Our old friend

We can now answer a lingering question from last term, and avoid having to make some torturous assumptions about what the "true" population mean is:

```
t.test(who$life_expectancy[who$status=="Developing"],
       who$life_expectancy[who$status=="Developed"])
```

```
##
##     Welch Two Sample t-test
##
## data:  who$life_expectancy[who$status == "Developing"] and who$life_expectar
## t = -12.854, df = 103.88, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -12.78858  -9.36995
## sample estimates:
## mean of x mean of y
##  69.70199  80.78125
```

```
# The square braces [] allow me to subset my data
# by the boolean operations within them
```

Processing math: 100%

Waves of data

Show 9 entries Search:

	sch_deid	grade	y1_boy_mean	y1_moy_mean	pre	post
1	10001	1	30.6	51.8	41.2	36.2
2	10001	2	71.3	105.7	88.5	84.4
3	10001	3	102.9	132.6	117.8	100.9
4	10002	1	34.1	62.3	48.2	41.7
5	10002	2	79.5	118.1	98.8	83.3
6	10002	3	95.7	127.4	111.5	105.8
7	10003	1	39.4	61.0	50.2	36.5
8	10003	2	86.3	122.3	104.3	88.6
9	10003	3	103.9	134.4	119.2	116.3

Showing 1 to 9 of 5,396 entries

Previous 1 2 3 4 5 … 600 Next

Processing math: 100%

Waves of data

Show 6 entries Search:

	sch_deid	grade	y1_boy_mean	y1_moy_mean	pre	post
1	10001	1	30.6	51.8	41.2	36.2
2	10001	2	71.3	105.7	88.5	84.4
3	10001	3	102.9	132.6	117.8	100.9
4	10002	1	34.1	62.3	48.2	41.7
5	10002	2	79.5	118.1	98.8	83.3
6	10002	3	95.7	127.4	111.5	105.8

Showing 1 to 6 of 5,396 entries

Previous 1 2 3 4 5 ... 900 Next

I actually have the same outcome stored across multiple variables. What is it? This is a classic example of a phenomenon you will come to know and hate: the curse of wide and long data structures.

Processing math: 100%

Wide and long data

Various types of analyses will necessitate different data structures:

Long to wide

Tidy data

Country	Year	Mobile
A	1990	1
B	2000	80
A	2000	100
B	1990	2

Wide data

Country	1990	2000
A	1	100
B	2	80

Wide to long

Wide data

Country	1990	2000
A	1	100
B	2	80

Tidy/long data

Country	Year	Mobile
A	1990	1
B	2000	80
A	2000	100
B	1990	2

Understanding exactly how to do this will take repeated time and practice, and you will nearly always need to look up and remind yourself how to do it. Bookmark and familiarize yourself with this vignette: https://tidyr.tidyverse.org/articles/pivot.html!

You don't need to be able to do this for assignments in this class, but I have the code for how to do so at the end of the slides.

Processing math: 100%

Long DIBELS

Show 9 entries Search:

	sch_deid	grade	period	mean_orf	pre	post
1	10001	1	y1_boy	30.6	41.2	36.2
2	10001	1	y1_moy	51.8	41.2	36.2
3	10001	1	y2_boy	26.0	41.2	36.2
4	10001	1	y2_moy	46.5	41.2	36.2
5	10001	2	y1_boy	71.3	88.5	84.4
6	10001	2	y1_moy	105.7	88.5	84.4
7	10001	2	y2_boy	71.1	88.5	84.4
8	10001	2	y2_moy	97.8	88.5	84.4
9	10001	3	y1_boy	102.9	117.8	100.9

Showing 1 to 9 of 21,584 entries

Previous 1 2 3 4 5 ... 2,399 Next

Processing math: 100%

Dummy coding

The mean values for pre-pandemic and post-pandemic onset are no longer helpful:

```
dibels_long <- select(dibels_long, -c(pre, post))
```

But it will be helpful for us to be able to designate which observations refer to a time period before the pandemic, and which refer to a time period post-onset:

```
dibels_long <- mutate(dibels_long,
                      post = ifelse(period=="y1_boy" |
                                    period=="y1_moy", 0, 1))
```

Processing math: 100%

Dummy variables

Dummy (or indicator variables) distinguish between categories, but offer no meaningful quantitative information ***on their own***.

By convention, the variable name corresponds to the category given by the value==1, e.g.:

$$post = 1 \text{ if after pandemic onset}$$

$$post = 0 \text{ if pre-pandemic}$$

The category given the value 0 is called the reference category.

Good data management practice: call the categorical variable the value implied by its substantive meaning when equal to 1 (i.e., "*post*" rather than "*pandemic*"; "*treat*" rather than "*condition*") so that you are clear on what 0 and 1 represent.

Processing math: 100%

Polychotomies

In fact, dummy coding will prove essential for categorical variables with more than two categories as well, especially those that are nominal (i.e., unordered):

```
table(dibels_long$school_titlei)
```

```
##
##                                                                 Missing
##                                                                     752
##                                                     Not a Title I school
##                                                                    2752
## Title I schoolwide eligible-Title I targeted assistance program
##                                                                    1124
##                       Title I schoolwide eligible school-No program
##                                                                     120
##                                         Title I schoolwide school
##                                                                   14712
##             Title I targeted assistance eligible school-No program
##                                                                     284
##                                 Title I targeted assistance school
##                                                                    1840
```

Processing math: 100%

Polychotomies

Here we have seven different levels of a school's Title I status. We can probably simplify these, but we need to be able to represent them using numerical values, when these levels don't inherently have a numerical structure. So... we use **dummy coding**. First, let's simplify the categories:

```
dibels_long <- dibels_long %>%
      mutate(title1 = case_when(school_titlei=="Missing" ~ "Missing",
                                school_titlei=="Not a Title I school" ~
                                school_titlei=="Title I schoolwide eligi
                                school_titlei=="Title I schoolwide eligi
                                school_titlei=="Title I schoolwide schoo
                                school_titlei=="Title I targeted assista
                                school_titlei=="Title I targeted assista
table(dibels_long$title1, exclude=NULL)
```

```
##
##             Missing       Not Title I Title I schoolwide   Title I targeted
##                 752              2752              15956               2124
```

Processing math: 100%

Dummy coding

The most common process for representing categorical variables in regression is dummy coding.

- Dummy coding essentially creates a new (dummy-coded) variable for each level.

School Status	D1	D2	D3
Not Title I	0	0	0
Title I schoolwide	1	0	0
Title I targeted	0	1	0
Missing	0	0	1

- One group becomes the reference group (in this case "Not Title I").
- The dummy-coded variables are then coded "1" for their corresponding level, and 0 for all other levels.

Processing math: 100%

Dummy coding

In a sample dataset, we could conceive of the dummy coding scheme like this:

School	Title I status	D1 (Schoolwide)	D2 (Targeted)	D3 (Missing)
10001	Not Title I	0	0	0
10002	Title I schoolwide	1	0	0
10003	Title I targeted	0	1	0
10004	Missing	0	0	1
10005	Title I schoolwide	1	0	0

Since "Not Title I" is our reference, we don't create a new column for it (it's implied by 0s in all other groups).

Hence, for K categories in our original variable, we have $K - 1$ dummy-coded variables.

Processing math: 100%

Dummies in R

From our polychotomous categorical variable, we can hand-create dummies:

```
dibels_long <- dibels_long %>%
  mutate(title1_school = ifelse(title1=="Title I schoolwide", 1, 0)) %>%
  mutate(title1_target = ifelse(title1=="Title I targeted", 1, 0)) %>%
  mutate(title1_miss = ifelse(title1=="Missing", 1, 0))
```

But, R is actually really smart, so the most straightforward way is to turn our original variable into a factor and then let R automatically convert it into a series of dummies when we need them:

```
dibels_long$title1 <- factor(dibels_long$title1)
```

Processing math: 100%

Categorical predictors in regression

Processing math: 100%

Categorical predictors in regression

In our standard multiple regression model, we have noted that we've made several important assumptions about our outcome (Y_i) and residuals (ε_i):

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_k X_{ki} + \varepsilon_i$$

but, we haven't made any particular assumptions about the form of the Xs. In fact, regression models can easily accommodate categorical variables (both dichotomous and polychotomous)!

Processing math: 100%

Pre/post in regression

We can now estimate whether there was a difference in ORF scores pre- and post-pandemic onset in regression:

```
fit1 <- lm(mean_orf ~ post, data=dibels_long)
summary(fit1)
```

```
##
## Call:
## lm(formula = mean_orf ~ post, data = dibels_long)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -74.306 -33.112   1.504  28.995 133.325
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  75.1246     0.3665 204.958   <2e-16 ***
## post         -4.4496     0.5184  -8.584   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 38.08 on 21582 degrees of freedom
## Multiple R-squared:  0.003403,   Adjusted R-squared:  0.003356
```

Processing math: 100%

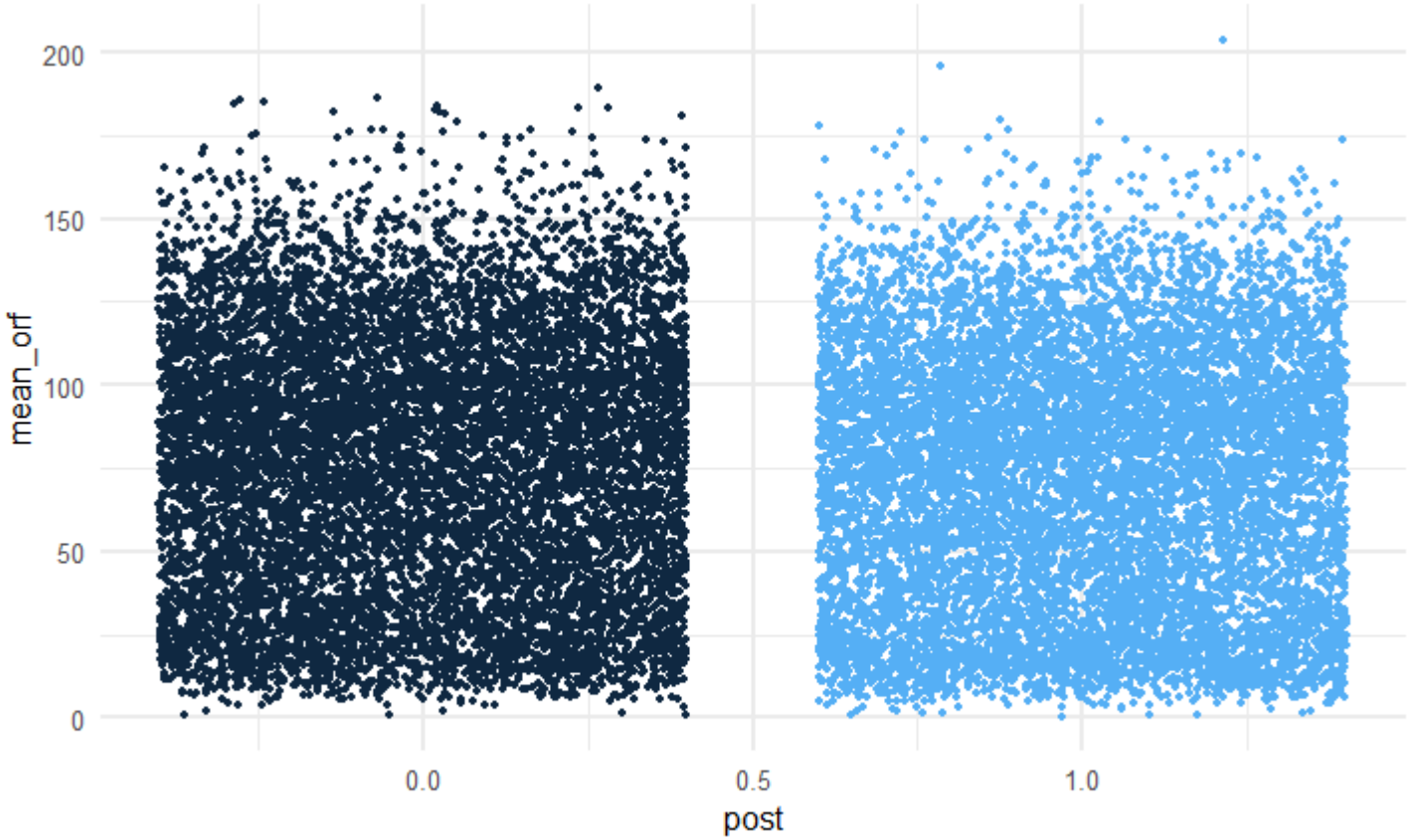
Pre/post in regression

```
##
## Call:
## lm(formula = mean_orf ~ post, data = dibels_long)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -74.306 -33.112   1.504  28.995 133.325
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  75.1246     0.3665 204.958   <2e-16 ***
## post         -4.4496     0.5184  -8.584   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 38.08 on 21582 degrees of freedom
## Multiple R-squared:  0.003403,    Adjusted R-squared:  0.003356
## F-statistic: 73.69 on 1 and 21582 DF,  p-value: < 2.2e-16
```

Our point estimate is identical to our original two-sample t-test, though our inference has changed slightly.

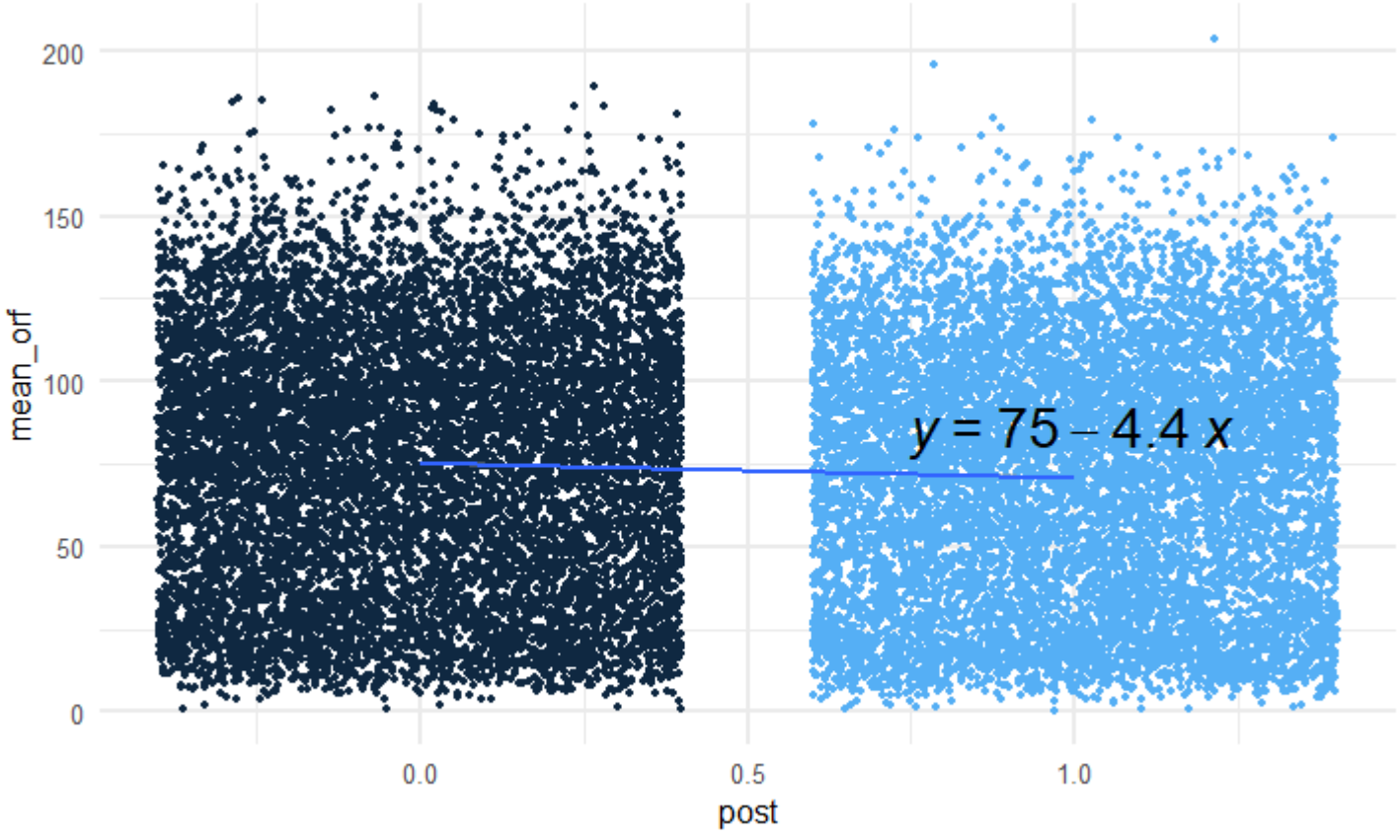
Processing math: 100%

Regression w. categorical predictors



Processing math: 100%

Regression w. categorical predictors



***y*-intercept**: estimated value of *Y* when dichotomous predictor=0

ted difference in *Y* between categories of predictor

Processing math: 100%

Reference category

What happens if we change the reference category?

```
# Create a new dummy variable called "pre", coded
# as one if test is prior to pandemic onset
dibels_long <- mutate(dibels_long,
                      pre = ifelse(period=="y1_boy" |
                                   period=="y1_moy", 1, 0))

# Fit the model
fit2 <- lm(mean_orf ~ pre, data=dibels_long)
```

Processing math: 100%

Reference category

What happens if we change the reference category?

```
...
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  70.6750     0.3665 192.819   <2e-16 ***
## pre           4.4496     0.5184   8.584   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 38.08 on 21582 degrees of freedom
## Multiple R-squared:  0.003403,    Adjusted R-squared:  0.003356
## F-statistic: 73.69 on 1 and 21582 DF,  p-value: < 2.2e-16
...
```

- Sign of slope is reversed
- Y-intercept is value of new reference category
- SE and inference remain exact same
- Full model statistics are the same

Processing math: 100%

What about the waves?

Up until now, we've focused on a simple comparison of pre- and post-pandemic onset scores. But this glosses over the facts that:

- Students typically improve substantially over the course of the year (we're lumping these time points together)
- We aren't able to capture the dynamic ways in which performance may have evolved over the early parts of the pandemic

We can use our multiple wave collection (now captured in our categorical polychotomous variable ***period***) to address this.

```
table(dibels_long$period, exclude=NULL)
```

```
##
## y1_boy y1_moy y2_boy y2_moy
##   5396   5396   5396   5396
```

Processing math: 100%

Categorical predictors

Nominal predictors

- These have *unordered* values
- E.g., gender, religion, political party, state of residence
- NEVER include a nominal predictor directly in a regression model
    - You end up with the problem of "country-ness" as a predictor

Ordinal predictors

- These have *ordered* values
- E.g., grade, developmental stage, education level (?)
- CAN include an ordinal predictor directly in regression, but make sure this is what you want!
    - Should you convert a political view scale (1=progressive, 2=liberal, 3=moderate, 4=conservative, 5=right-wing) to a series of dummies?
    - What about education (1=HS dropout, 2=HS grad, 3=some college, 4=college grad)?

Processing math: 100%

Polychotomies in regression

In a regression model, categorical predictors are typically entered in their dummy-coded format:

$$Y = \beta_0 + \beta_1 D_1 + \beta_2 D_2 + \beta_3 D_3 + \ldots + \varepsilon$$

In our four-wave ORF regression, we can think of the equation like this:[1]

$$MEAN\_ORF_j = \beta_0 + \beta_1 Y1\_MOY_j + \beta_2 Y2\_BOY_j + \beta_3 Y2\_MOY_j + \varepsilon_j$$

where did *Y1_BOY* (year 1, beginning of year) go?

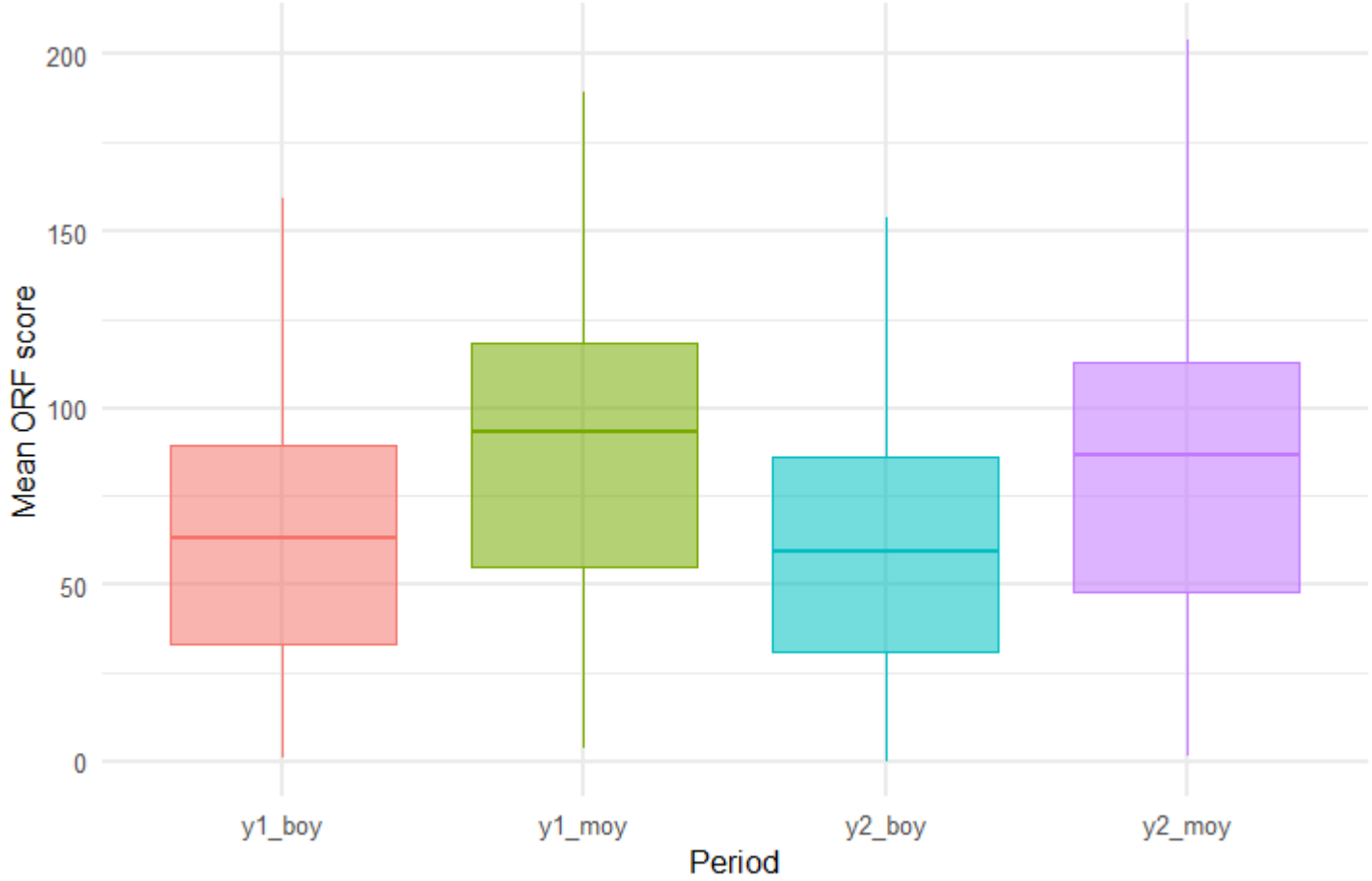
How should we interpret each of the coefficients?

How should we interpret the intercept?

[1] Following convention, I'm subscripting this equation with "*j*" because our data does not represent individual observations (which we typically subscript with "*i*"), but mean values, aggregated at the school-grade level.

Processing math: 100%

Why does this work?



Processing math: 100%

Interpreting coefficients

```
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   62.3436     0.4921 126.701  < 2e-16 ***
periody1_moy  25.5620     0.6959  36.734  < 2e-16 ***
periody2_boy  -2.7914     0.6959  -4.011 6.06e-05 ***
periody2_moy  19.4541     0.6959  27.957  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 36.15 on 21580 degrees of freedom
...
```

On average, when measured in Fall 2019, grades in schools had a mean ORF score of 62.3

Processing math: 100%

Interpreting coefficients

```
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   62.3436     0.4921 126.701  < 2e-16 ***
periody1_moy  25.5620     0.6959  36.734  < 2e-16 ***
periody2_boy  -2.7914     0.6959  -4.011 6.06e-05 ***
periody2_moy  19.4541     0.6959  27.957  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 36.15 on 21580 degrees of freedom
...
```

On average, when measured in Fall 2019, grades in schools had a mean ORF score of 62.3

On average, when measured in Winter 2020, grades in schools had a mean ORF score of 87.9 (62.3 + 25.6)

Processing math: 100%

Interpreting coefficients

```
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   62.3436     0.4921 126.701  < 2e-16 ***
periody1_moy  25.5620     0.6959  36.734  < 2e-16 ***
periody2_boy  -2.7914     0.6959  -4.011 6.06e-05 ***
periody2_moy  19.4541     0.6959  27.957  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 36.15 on 21580 degrees of freedom
...
```

On average, when measured in Fall 2019, grades in schools had a mean ORF score of 62.3

On average, when measured in Winter 2020, grades in schools had a mean ORF score of 87.9 (62.3 + 25.6)

On average, when measured in Fall 2020, grades in schools had a mean ORF score of 59.6 (62.3 + (-2.8))

Processing math: 100%

Interpreting coefficients

```
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   62.3436     0.4921 126.701  < 2e-16 ***
periody1_moy  25.5620     0.6959  36.734  < 2e-16 ***
periody2_boy  -2.7914     0.6959  -4.011 6.06e-05 ***
periody2_moy  19.4541     0.6959  27.957  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 36.15 on 21580 degrees of freedom
...
```

On average, when measured in Fall 2019, grades in schools had a mean ORF score of 62.3

On average, when measured in Winter 2020, grades in schools had a mean ORF score of 87.9 (62.3 + 25.6)

On average, when measured in Fall 2020, grades in schools had a mean ORF score of 59.6 (62.3 + (-2.8))

On average, when measured in Winter 2021, grades in schools had a mean ORF score of 81.79 (62.3 + 19.5)

Processing math: 100%

Interpreting coefficient significance

Coefficient significance tests still test the null hypothesis $\beta_k = 0$, but **we are testing against the reference group** implicit in our intercept.

```
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   62.3436     0.4921 126.701  < 2e-16 ***
periody1_moy  25.5620     0.6959  36.734  < 2e-16 ***
periody2_boy  -2.7914     0.6959  -4.011 6.06e-05 ***
periody2_moy  19.4541     0.6959  27.957  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

...
```

Which DIBELS test wave differs significantly from our reference group: "Y1_BOY"?

So, this is just a comparison of means, or a series of independent-sample t-tests!

Processing math: 100%

Changing reference category

If we change the model's reference category with a polychotomous variable, we ***will*** change the parameter estimates and associated tests. Each refers to the estimated mean difference for that group and the reference category. There can be significant variation from one group (e.g., time period) to another, but not all groups are different from each other.

```
# I can specify directly in my call which group to serve as reference
summary(lm(mean_orf ~ relevel(period, ref="y2_boy"), data=dibels_long))
```

```
...
## Coefficients:
##                                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)                               59.5522     0.4921 121.028  < 2e-16 *
## relevel(period, ref = "y2_boy")y1_boy      2.7914     0.6959   4.011 6.06e-05 *
## relevel(period, ref = "y2_boy")y1_moy     28.3534     0.6959  40.745  < 2e-16 *
## relevel(period, ref = "y2_boy")y2_moy     22.2455     0.6959  31.968  < 2e-16 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.15 on 21580 degrees of freedom
## Multiple R-squared:  0.1021,    Adjusted R-squared:  0.1019
## F-statistic: 817.7 on 3 and 21580 DF,  p-value: < 2.2e-16
...
```

Processing math: 100%

Prediction with categorical variables

Using the coefficients from our output, we have the following fitted regression equation:

$$\hat{MEAN\_ORF}_j = 62.3 + 25.6(Y1\_MOY_j) + (-2.8)(Y2\_BOY_j) + 19.6(Y2\_MOY_j)$$

What is the predicted ORF for school grades in the middle of the 2020-21 school year?

$$\hat{MEAN\_ORF}_j = 62.3 + 25.6(0) + (-2.8)(0) + 19.6(1) = 62.3 + 19.6 = 81.9$$

For dummy coded variables, we just add the appropriate effects for the group we are interested in, or omit them if they are in our reference group.

Processing math: 100%

So many tests

Reference group	Y1_BOY	Y1_MOY	Y2_BOY	Y2_MOY
Y1_BOY	.	1	2	3
Y1_MOY		.	4	5
Y2_BOY			.	6
Y2_MOY				.

DANGER: we're back in the land of multiple hypothesis testing, and we may be inadvertently committing Type I error!

Processing math: 100%

Dangers of multiple hypothesis tests

If your goal is to find a "statistically significant" result, you will detect such a relationship 1 out of 20 times (on average).

Imagine rolling a die. What is the probability you roll a 1? $1/6 = 0.167$

Now, roll it twice, what is the probability **at least** one of your rolls is a 1?
$1 - (5/6 * 5/6) = 0.306$

If you conduct enough tests, you'll detect a relationship eventually.

Processing math: 100%

Multiple tests in the wild

- How many tests?

$$\frac{(n \text{ categories})(n \text{ categories} - 1)}{2}$$

- ~ 80 counties, so 3160 tests

Table I.4.2(1/2) **Comparing countries' and economies' performance in mathematics**

Statistically significantly **above** the OECD average
Not statistically significantly different from the OECD average
Statistically significantly **below** the OECD average

Mean score	Comparison country/economy	Countries and economies whose mean score is not statistically significantly different from the comparison country's/economy's score
591	B-S-J-Z (China)	
569	Singapore	
558	Macao (China)	Hong Kong (China)[1]
551	Hong Kong (China)[1]	Macao (China)
531	Chinese Taipei	Japan, Korea
527	Japan	Chinese Taipei, Korea, Estonia
526	Korea	Chinese Taipei, Japan, Estonia, Netherlands[1]
523	Estonia	Japan, Korea, Netherlands[1]
519	Netherlands[1]	Korea, Estonia, Poland, Switzerland
516	Poland	Netherlands,[1] Switzerland, Canada
515	Switzerland	Netherlands,[1] Poland, Canada, Denmark
512	Canada	Poland, Switzerland, Denmark, Slovenia, Belgium, Finland
509	Denmark	Switzerland, Canada, Slovenia, Belgium, Finland
509	Slovenia	Canada, Denmark, Belgium, Finland
508	Belgium	Canada, Denmark, Slovenia, Finland, Sweden, United Kingdom
507	Finland	Canada, Denmark, Slovenia, Belgium, Sweden, United Kingdom
502	Sweden	Belgium, Finland, United Kingdom, Norway, Germany, Ireland, Czech Republic, Austria, Latvia
502	United Kingdom	Belgium, Finland, Sweden, Norway, Germany, Ireland, Czech Republic, Austria, Latvia, France
501	Norway	Sweden, United Kingdom, Germany, Ireland, Czech Republic, Austria, Latvia, France, Iceland
500	Germany	Sweden, United Kingdom, Norway, Ireland, Czech Republic, Austria, Latvia, France, Iceland, New Zealand
500	Ireland	Sweden, United Kingdom, Norway, Germany, Czech Republic, Austria, Latvia, France, Iceland, New Zealand
499	Czech Republic	Sweden, United Kingdom, Norway, Germany, Ireland, Austria, Latvia, France, Iceland, New Zealand, Portugal[1]
499	Austria	Sweden, United Kingdom, Norway, Germany, Ireland, Czech Republic, Latvia, France, Iceland, New Zealand, Portugal[1]
496	Latvia	Sweden, United Kingdom, Norway, Germany, Ireland, Czech Republic, Austria, France, Iceland, New Zealand, Portugal,[1] Australia
495	France	United Kingdom, Norway, Germany, Ireland, Czech Republic, Austria, Latvia, Iceland, New Zealand, Portugal,[1] Australia
495	Iceland	Norway, Germany, Ireland, Czech Republic, Austria, Latvia, France, New Zealand, Portugal,[1] Australia
494	New Zealand	Germany, Ireland, Czech Republic, Austria, Latvia, France, Iceland, Portugal,[1] Australia
492	Portugal[1]	Czech Republic, Austria, Latvia, France, Iceland, New Zealand, Australia, Russia, Italy, Slovak Republic
491	Australia	Latvia, France, Iceland, New Zealand, Portugal,[1] Russia, Italy, Slovak Republic
488	Russia	Portugal,[1] Australia, Italy, Slovak Republic, Luxembourg, Spain, Lithuania, Hungary
487	Italy	Portugal,[1] Australia, Russia, Slovak Republic, Luxembourg, Spain, Lithuania, Hungary, United States[1]
486	Slovak Republic	Portugal,[1] Australia, Russia, Italy, Luxembourg, Spain, Lithuania, Hungary, United States[1]
483	Luxembourg	Russia, Italy, Slovak Republic, Spain, Lithuania, Hungary, United States[1]
481	Spain	Russia, Italy, Slovak Republic, Luxembourg, Lithuania, Hungary, United States[1]
481	Lithuania	Russia, Italy, Slovak Republic, Luxembourg, Spain, Hungary, United States[1]
481	Hungary	Russia, Italy, Slovak Republic, Luxembourg, Spain, Lithuania, United States[1]
478	United States[1]	Italy, Slovak Republic, Luxembourg, Spain, Lithuania, Hungary, Belarus, Malta
472	Belarus	United States,[1] Malta
472	Malta	United States,[1] Belarus
464	Croatia	Israel
463	Israel	Croatia
454	Turkey	Ukraine, Greece, Cyprus, Serbia
453	Ukraine	Turkey, Greece, Cyprus, Serbia
451	Greece	Turkey, Ukraine, Cyprus, Serbia
451	Cyprus	Turkey, Ukraine, Greece, Serbia
448	Serbia	Turkey, Ukraine, Greece, Cyprus, Malaysia
440	Malaysia	Serbia, Albania, Bulgaria, United Arab Emirates, Romania
437	Albania	Malaysia, Bulgaria, United Arab Emirates, Romania
436	Bulgaria	Malaysia, Albania, United Arab Emirates, Brunei Darussalam, Romania, Montenegro
435	United Arab Emirates	Malaysia, Albania, Bulgaria, Romania
430	Brunei Darussalam	Bulgaria, Romania, Montenegro
430	Romania	Malaysia, Albania, Bulgaria, United Arab Emirates, Brunei Darussalam, Montenegro, Kazakhstan, Moldova, Baku (Azerbaijan), Thailand
430	Montenegro	Bulgaria, Brunei Darussalam, Romania

Processing math: 100%

One fix

Instead of using $\alpha = 0.05$ for each individual test, use $\alpha = 0.05$ for the **family of tests** when we examine multiple contrasts to test a single hypothesis.

Bonferroni method

Take a given α-threshold and "split it" across the entire family of tests. Assuming $\alpha = 0.05$:

- For 2 tests, conduct each at 0.025 level;
- For 3 tests, conduct each at 0.0167 level;
  etc. ...

Use this new threshold to identify the critical t-statistic given the number of degrees of freedom. For the PISA example this would be p=0.05/3160=0.000016

Other approaches exist! Bonferroni is an extremely conservative one--beware!

As tests increase, so do critical t-values:

# tests	# new α	t-statistic (df = ∞)
1	0.0500	1.96
2	0.0250	2.24
3	0.0167	2.39
4	0.0125	2.50
5	0.0100	2.58
6	0.0083	2.64
10	0.0050	2.81
20	0.0025	3.02
50	0.0010	3.29
100	0.0005	3.48

Processing math: 100%

Bonferroni correction in R

```
pairwise.t.test(dibels_long$mean_orf, dibels_long$period,
                p.adjust.method = "bonferroni")
```

```
##
##     Pairwise comparisons using t tests with pooled SD
##
## data:  dibels_long$mean_orf and dibels_long$period
##
##         y1_boy  y1_moy  y2_boy
## y1_moy < 2e-16 -       -
## y2_boy 0.00036 < 2e-16 -
## y2_moy < 2e-16 < 2e-16 < 2e-16
##
## P value adjustment method: bonferroni
```

We can see that our inference has become slightly weaker for our *Y1_BOY* vs. *Y2_BOY* comparison, though still smaller than most traditional thresholds. Note that the others have become weaker too, but they were so small to begin with that we don't see this change.

Processing math: 100%

Another potential solution

Is there another way we could conceive of the assessment data collection wave variable that is *not* categorical?

```
dibels_long$time <- as.numeric(dibels_long$period)
summary(lm(mean_orf ~ time, dibels_long))
```

```
...
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  65.3975     0.6335  103.24   <2e-16 ***
## time          3.0009     0.2313   12.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.99 on 21582 degrees of freedom
## Multiple R-squared:  0.007738,    Adjusted R-squared:  0.007692
## F-statistic: 168.3 on 1 and 21582 DF,  p-value: < 2.2e-16
...
```

But perhaps, there are more satisfying ways to address this challenge!

Processing math: 100%

ANOVA

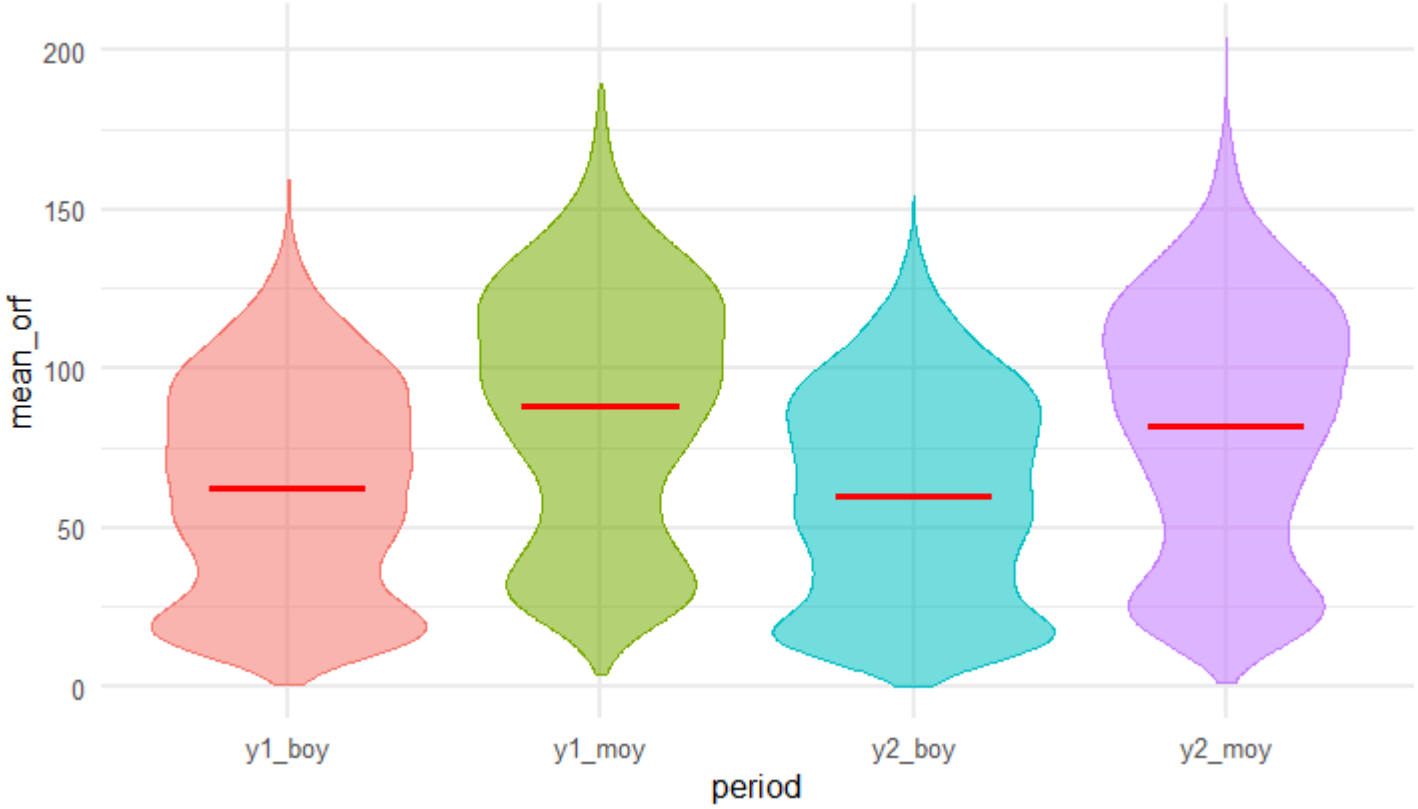
Processing math: 100%

ANOVA

- Analysis of variance (ANOVA) is a special case of the general linear model
- The primary goal of ANOVA is a comparison of means across different groups
    - $H_0 : \mu_1 = \mu_2 = \mu_3 \ldots \mu_K$
- Although regression frameworks are more the norm across most disciplines, the ANOVA approach can be especially useful for:
    - Exploring and comparing the within- and between-group variation in the outcome
    - Simultaneously testing the main effects of categorical variables *(and avoiding some of the problems of multiple hypothesis testing)*

Processing math: 100%

Within and between

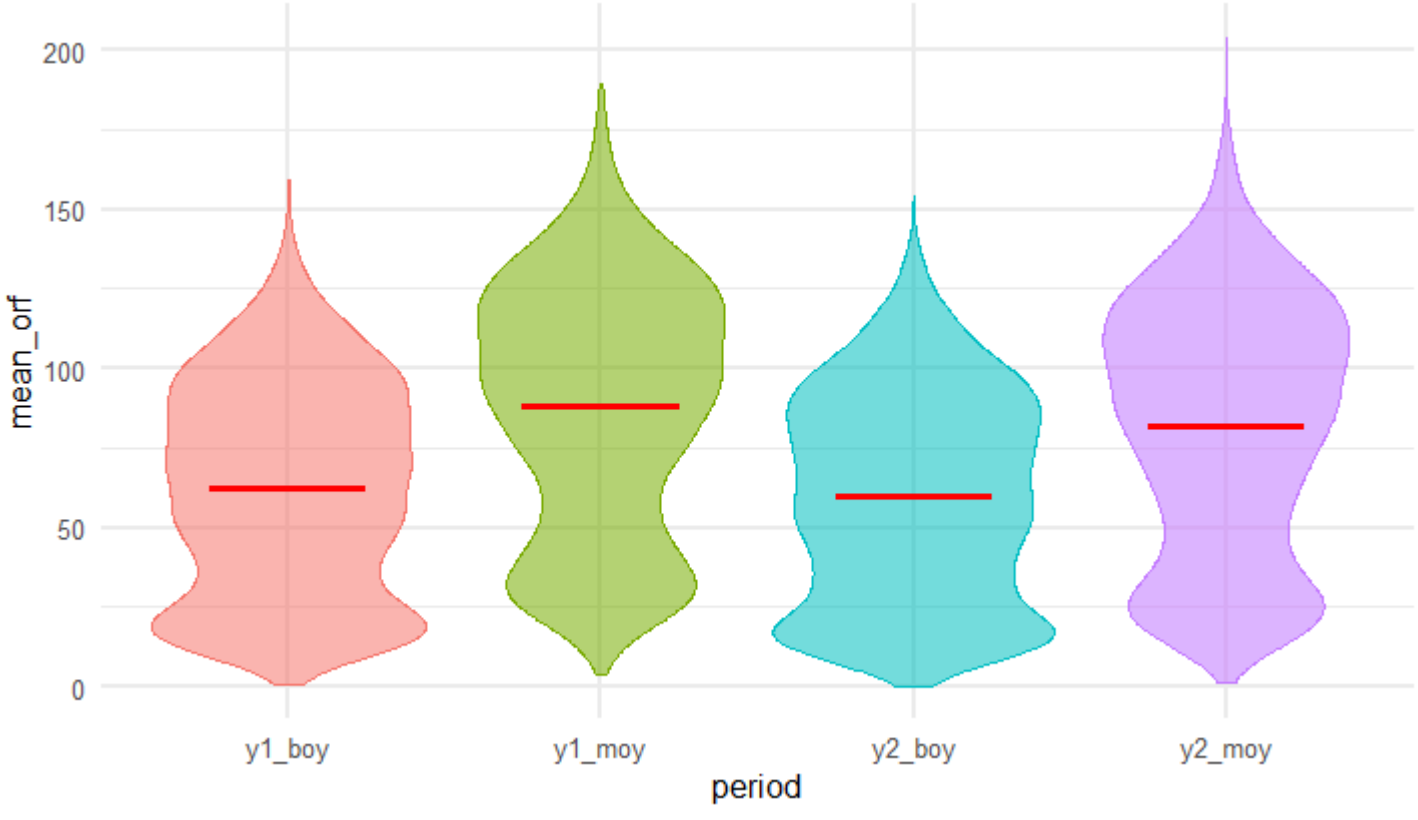


Can you describe the variability within- and between-test periods?

Check out the podcast Within & Between on quant methods and unpacking the hidden curriculum of academia hosted by Jessica Logan and Sara Hart (takes a developmental science perspective).

Processing math: 100%

Within and between



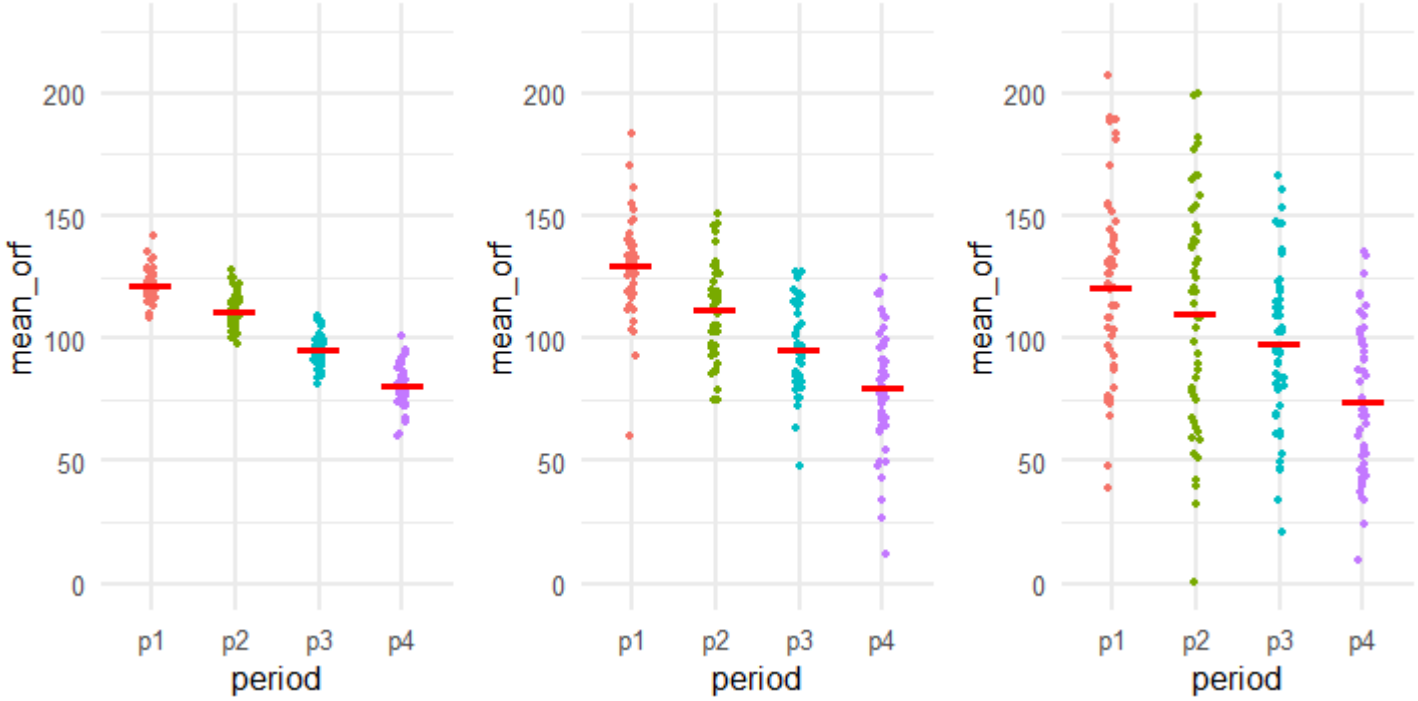
Questions we might want to answer about group differences:

1. Are observed differences between groups "real"?
2. In what context can we place these differences to evaluate their magnitude?

Processing math: 100%

Within and between

Let's imagine a slightly simpler example. Imagine three different data sets with a four-level categorical predictor and across each data set, the mean value of each category was the same.



How important are the differences in these group means across each dataset? Within-group variation provides important context for evaluating magnitude of between-group variation!

Processing math: 100%

Partitioning variance

In regression, we partition our total variance SS_{total} into our SS_{model} and SS_{residual} :

SS_{model} = Deviation of observed value from the predicted value $(Y_i - \hat{Y}_i)$

SS_{residual} = Deviation of predicted value from the grand mean $(\hat{Y} - \bar{Y}_i)$

In ANOVA, we apply a similar but slightly different conceptual process.

Processing math: 100%

Partioning variance in ANOVA

In ANOVA, we separate variance into between-group and within-group variance:

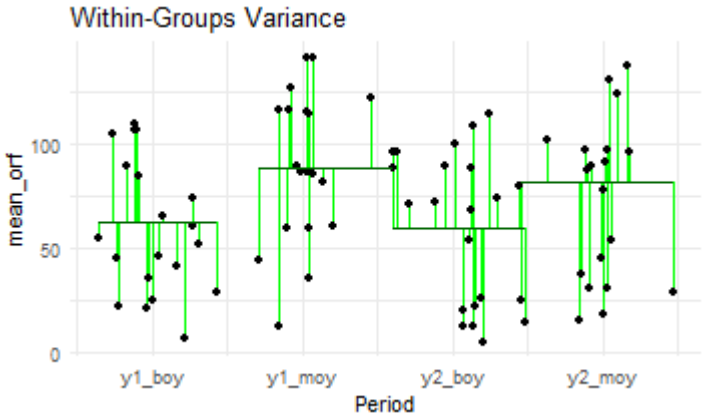
SS_{within} = Deviation of observed value from its group mean $(Y_{ik} - \bar{Y}_k)$

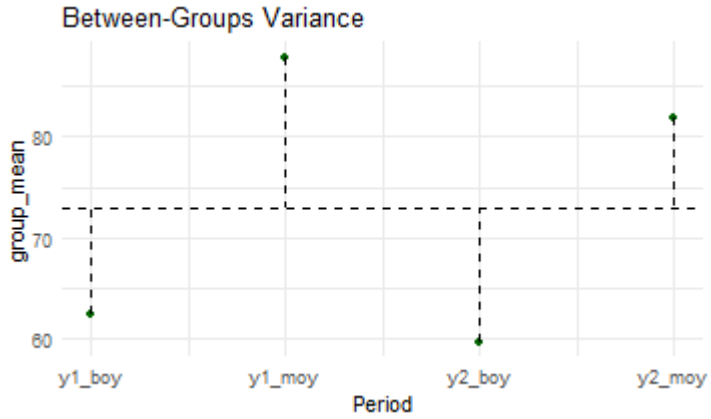
SS_{between} = Deviation of group mean from the grand mean $(\bar{Y}_k - \bar{Y})$

$$SS_{\text{total}} = SS_{\text{within}} + SS_{\text{between}}$$

Processing math: 100%

Visualized variance partition





We can represent the residual variance around the group means (here on just a random selection of 20 observations from each period). Just like the error term in our regression model, it is all the remaining variance our predictor (***period***) can't explain.

In addition to each individual observation's deviation from its group mean, each group's mean also deviates from the grand mean of Oral Reading Fluency.

Processing math: 100%

ANOVA test statistic

When we conduct an ANOVA we are testing the significance of an F-statistic using the following formula:

$$F = \frac{MS_{\text{between}}}{MS_{\text{within}}}$$

The mean squares (MS) of between- and within-group variance is just the Sum of the Squares (SS) for each group, divided by its degrees of freedom (df):

$$MS_w = \frac{SS_w}{df_w} \qquad MS_b = \frac{SS_b}{df_b}$$

$$df_w = N - G \qquad df_b = G - 1$$

where w subscripts within, b subscripts between, N is the number of observations and G the number of groups.

Processing math: 100%

ANOVA significance test

The null hypothesis of an ANOVA is about the ratio of between- to within-group variance.

Essentially, when we state $H_0 : \mu_1 = \mu_2 = \mu_3 \ldots \mu_K$, we are asking if the mean square variance of the group means around the grand mean is greater than the mean square variance of observations around their group mean. If the between-group variance were much larger than the within-group variance, then the F-statistic would exceed 1.

$$F = \frac{MS_{\text{between}}}{MS_{\text{within}}} = \frac{4.3}{1.5} = 2.87$$

If the between-group variance is equal to or smaller than the within-group variance, then our F-statistic will be ≤ 1.

$$F = \frac{MS_{\text{between}}}{MS_{\text{within}}} = \frac{0.2}{1.5} = 0.13$$

Processing math: 100%

Calculating the F-statistic

Let's find our F-statistic for our ***period*** variable.

Within-Group (Residual) Variance MS_{Within}

```
# total n - number of groups (4)
nrow(dibels_long)
```

```
## [1] 21584
```

```
df_within <- 21584 - 4

sum((dibels_long$mean_orf - dibels_long$group_mean)^2) / df_within
```

```
## [1] 1306.462
```

Processing math: 100%

Calculating the F-statistic

Let's find our F-statistic for our ***period*** variable.

Between-Group Variance MS_{Between}

```
# number of groups (4) - 1
df_btw <- 4-1

sum((mean(dibels_long$mean_orf) - dibels_long$group_mean)^2) / df_btw
```

```
## [1] 1068298
```

Processing math: 100%

Calculating the F-statistic

$$MS_{\text{Between}} = 1,068,298$$

$$MS_{\text{Within}} = 1,306$$

$$F = \frac{MS_{\text{Between}}}{MS_{\text{Within}}} = \frac{1,068,298}{1306} = 817.99$$

Our F-statistic is 818. Now that we see how it is calculated, let's fit an ANOVA in R to review the output and make an inference (note that we could now also consult an F-statistic lookup table to get the same info!).

Processing math: 100%

ANOVA in R

Because ANOVA is just a particular method of analyzing variance in GLMs, we can wrap the `anova` command around our `lm` model fit.

```
fit3 <- lm(mean_orf ~ period, dibels_long)

anova(fit3)
```

```
## Analysis of Variance Table
##
## Response: mean_orf
##               Df   Sum Sq Mean Sq F value    Pr(>F)
## period         3  3204894 1068298   817.7 < 2.2e-16 ***
## Residuals  21580 28193461    1306
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can see all the information we calculated earlier. With a *p*-value $< 2.2 \times 10^{-16}$, our *F*-statistic is highly unlikely to be a product of a population in which the population means across the four waves of ORF administration were equal. Therefore we reject the null hypothesis and conclude that, on average in the population, the mean ORF scores differed significantly across the waves of assessment administration.

Processing math: 100%

Variance decomposition

- In addition to using ANOVA to "batch test" group differences, as we've seen it can be a useful tool to decompose the variance of your outcome into between and within group variation
- We can, in fact, extend this analysis to understand how much of the variation in an individual's outcome occurs across **different** groups. For example:
  - What proportion of the variation in child outcomes occurs within classrooms, compared to schools, compared to neighborhoods?
  - Are differences in school funding greater between schools, between districts or between states?
- You can explore more about these topics in our HLM sequence (EDLD 628/629)

Processing math: 100%

ANOVA vs. regression

- Both are implementations of the General Linear Model
- A regression with dummy indicator variables is statistically identical to ANOVA
- The F-test in a regression model represents a test of the model's variance against the residual
- In ANOVA, we can have one or more F-tests where we "batch test" a group of coefficients
    - This can help avoid Type I errors (rejecting the null when it is in fact true)
    - ANOVA doesn't tell you anything about the magnitude of the difference...which seems important?
- Learning regression is the more general approach, of which ANOVA is a special implementation; by learning regression you have a more flexible tool kit

Processing math: 100%

Presenting our results

```
modelsummary(list(fit1, fit3),
             stars=T,
             vcov = "robust",
             gof_omit = "Adj.|AIC|BIC|Log|RMSE|RSE|Std.Err",
             coef_rename = c("post" = "Post-Pandemic Onset",
                             "periody1_moy" = "Winter 2020",
                             "periody2_boy" = "Fall 2020",
                             "periody2_moy" = "Winter 2021"))
```

Processing math: 100%

Table 1. Estimates of grade-level average Oral Reading Fluency (ORF) score across waves of DIBELS administration, 2019-2021

	(1)	(2)
(Intercept)	75.125***	62.344***
	(0.369)	(0.449)
Post-Pandemic Onset	-4.450***	
	(0.518)	
Winter 2020		25.562***
		(0.696)
Fall 2020		-2.791***
		(0.631)
Winter 2021		19.454***
		(0.700)
Num.Obs.	21584	21584
R2	0.003	0.102

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

coefficients and heteroscedastic-robust standard errors in parentheses. Each observation is a school-grade-test value.

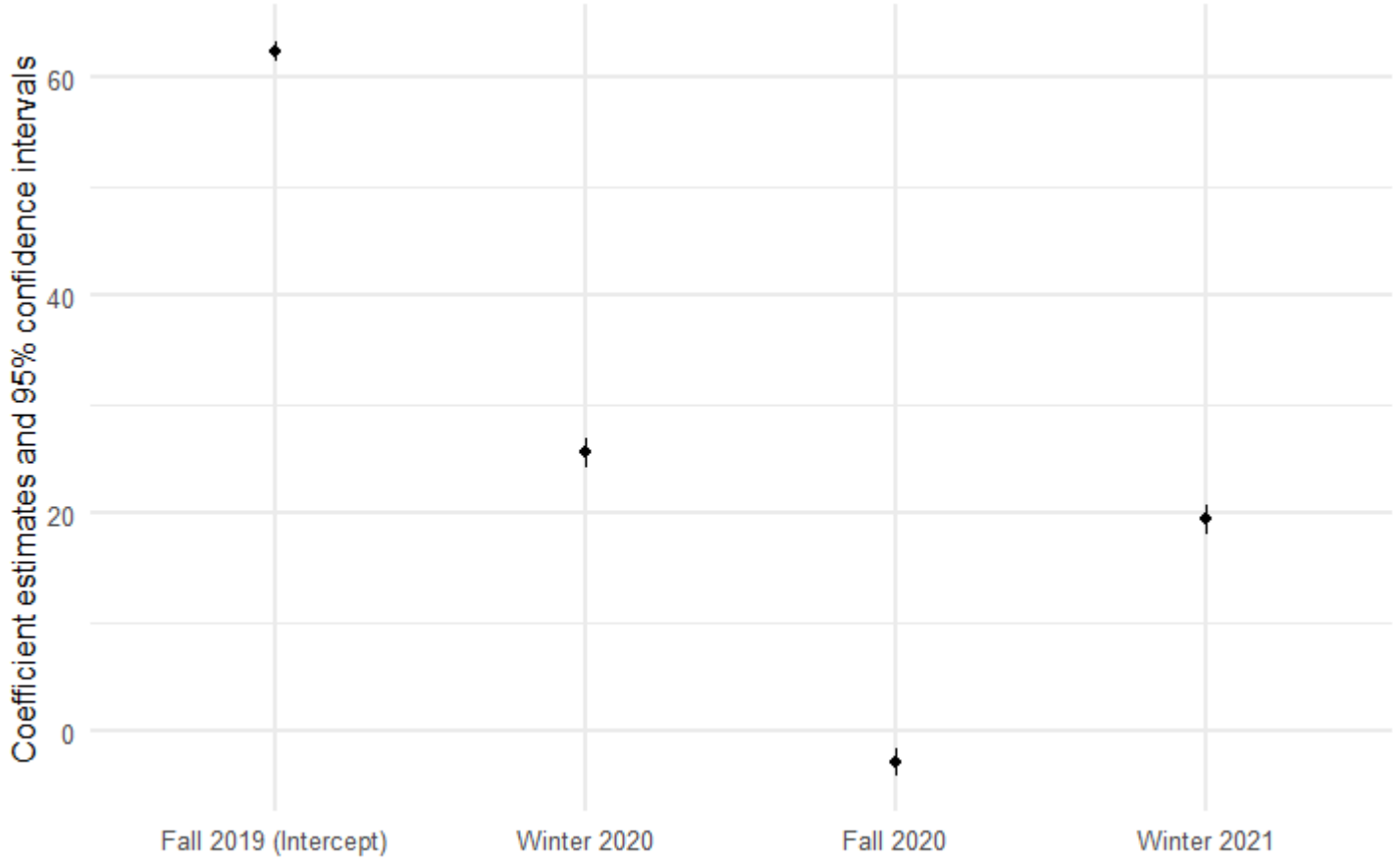
Processing math: 100%

Visualizing results

```
coef_plot <- modelplot(fit3,
               coef_rename = c("(Intercept)" = "Fall 2019 (Intercept)",
                               "periody1_moy" = "Winter 2020",
                               "periody2_boy" = "Fall 2020",
                               "periody2_moy" = "Winter 2021"),
               vcov = "robust") +
               coord_flip() +
               theme_minimal(base_size = 16)
```

Processing math: 100%

Visualizing results



Processing math: 100%

Multiple regression with categorical variables

Processing math: 100%

Mawwrr predictors

Now that we've learned the basic concept of multiple regression, it's a fairly simple task to add additional covariates (either continuous or categorical) to our equation.

What theoretically justified covariates might be sensible to include? How would we make such a determination?

Preliminarily, let's look at two: *GRADE* and *SCHOOL_ENROLL*:

$$MEAN\_ORF_j = \beta_0 + \beta_1 Y1\_MOY_j + \beta_2 Y2\_BOY_j + \beta_3 Y2\_MOY_j + \beta_4 GRADE_j + \beta_5 SCHOOL\_ENROLL_j + \ldots + \varepsilon_j$$

Processing math: 100%

Mawwrr predictors

$$MEAN\_ORF_j = \beta_0 + \beta_1 Y1\_MOY_j + \beta_2 Y2\_BOY_j + \beta_3 Y2\_MOY_j + \beta_4 GRADE_j + \beta_5 SCHOOL\_ENROLL_j + \ldots + \varepsilon_j$$

Before fitting any models, **can we interpret what each of these coefficients will now represent?**

To what do I need to attend when specifying the predictor *GRADE* in my model?

Processing math: 100%

MR with categoricals

```
fit4<-lm(mean_orf ~ period + as.factor(grade) +
                    school_enroll, data=dibels_long)
summary(fit4)
```

```
...
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        16.0746471  0.3715338  43.266  < 2e-16 ***
## periody1_moy       25.5620256  0.3324093  76.899  < 2e-16 ***
## periody2_boy       -2.7913676  0.3324093  -8.397  < 2e-16 ***
## periody2_moy       19.4541290  0.3324093  58.525  < 2e-16 ***
## as.factor(grade)2 38.9543683  0.3267633 119.213  < 2e-16 ***
## as.factor(grade)3 60.2924720  0.3572314 168.777  < 2e-16 ***
## as.factor(grade)4 81.9691911  0.3804983 215.426  < 2e-16 ***
## as.factor(grade)5 82.7431382  0.3913853 211.411  < 2e-16 ***
## school_enroll      0.0025891  0.0007059   3.668 0.000245 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.27 on 21575 degrees of freedom
## Multiple R-squared:  0.7952,    Adjusted R-squared:  0.7951
## F-statistic: 1.047e+04 on 8 and 21575 DF,  p-value: < 2.2e-16
```

Processing math: 100%

```
##
## Call:
## lm(formula = mean_orf ~ period + as.factor(grade) + school_enroll,
##     data = dibels_long)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -98.509 -10.805  -0.484  10.386  86.013
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        16.0746471  0.3715338  43.266  < 2e-16 ***
## periody1_moy       25.5620256  0.3324093  76.899  < 2e-16 ***
## periody2_boy       -2.7913676  0.3324093  -8.397  < 2e-16 ***
## periody2_moy       19.4541290  0.3324093  58.525  < 2e-16 ***
## as.factor(grade)2 38.9543683  0.3267633 119.213  < 2e-16 ***
## as.factor(grade)3 60.2924720  0.3572314 168.777  < 2e-16 ***
## as.factor(grade)4 81.9691911  0.3804983 215.426  < 2e-16 ***
## as.factor(grade)5 82.7431382  0.3913853 211.411  < 2e-16 ***
## school_enroll       0.0025891  0.0007059   3.668 0.000245 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.27 on 21575 degrees of freedom
## Multiple R-squared:  0.7952,    Adjusted R-squared:  0.7951
## stic: 1.047e+04 on 8 and 21575 DF,  p-value: < 2.2e-16
```

Processing math: 100%

Show what you know

Use the adjusted means to show your findings when your question predictor is categorical. Set all predictors to their sample means or to the value of the category and then compute the predicted value of your outcome at each level of your categorical question predictor:

```
mean(dibels_long$school_enroll)
```

```
## [1] 336.9754
```

```
prop.table(table(dibels_long$grade))
```

```
##
##         1         2         3         4         5
## 0.2657524 0.2524092 0.1864344 0.1540030 0.1414010
```

Processing math: 100%

Show what you know

Use the adjusted means to show your findings when your question predictor is categorical. Set all predictors to their sample means or to the value of the category and then compute the predicted value of your outcome at each level of your categorical question predictor:

For Y1_MOY:

$$\hat{MEAN\_ORF_j} = 16.06 + 25.56(1) + (-2.79)(0) + 19.45(0) + 38.95(0.25) + 60.29(0.19) + 81.96(0.15) + 82.74$$

For Y2_BOY:

$$\hat{MEAN\_ORF_j} = 16.06 + 25.56(0) + (-2.79)(1) + 19.45(0) + 38.95(0.25) + 60.29(0.19) + 81.96(0.15) + 82.74$$

For Y2_MOY:

$$\hat{MEAN\_ORF_j} = 16.06 + 25.56(0) + (-2.79)(0) + 19.45(1) + 38.95(0.25) + 60.29(0.19) + 81.96(0.15) + 82.74$$

For Y1_BOY:

Processing math: 100%

$$\hat{MEAN\_ORF} = 16.06 + 25.56(0) + (-2.79)(0) + 19.45(0) + 38.95(0.25) + 60.29(0.19) + 81.96(0.15) + 82.74$$

Present in simple tabular format

Table 1.

Mean Oral Reading Fluency score across different administrations of the DIBELS 8

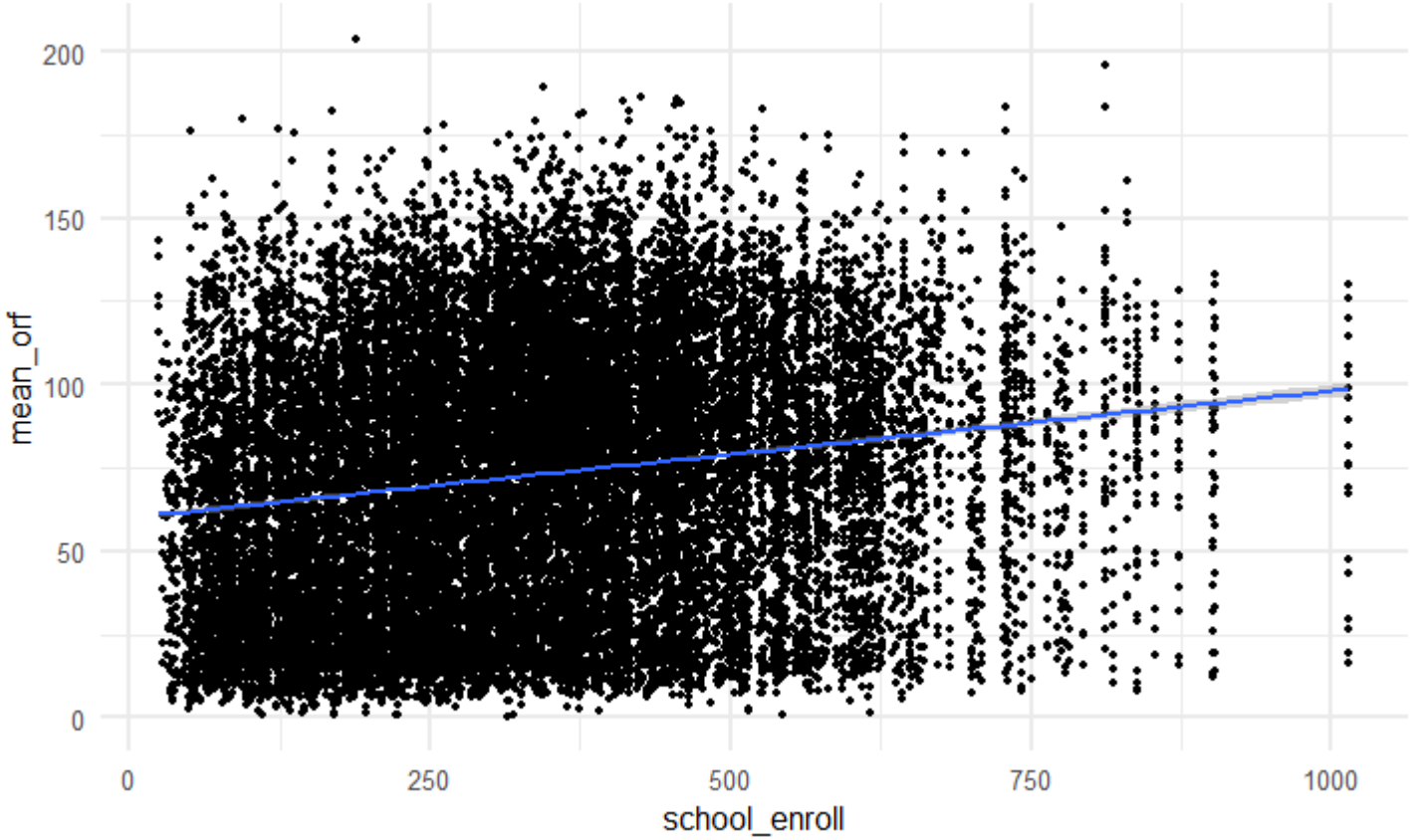
Wave	Unadjusted	Adjusted
Fall 2019	62.3	62.1
Winter 2020	87.9	87.7
Fall 2020	59.6	62.1
Winter 2021	81.8	81.6

*Adjusted mean = adjusting for grade and school size.

Processing math: 100%

An aside

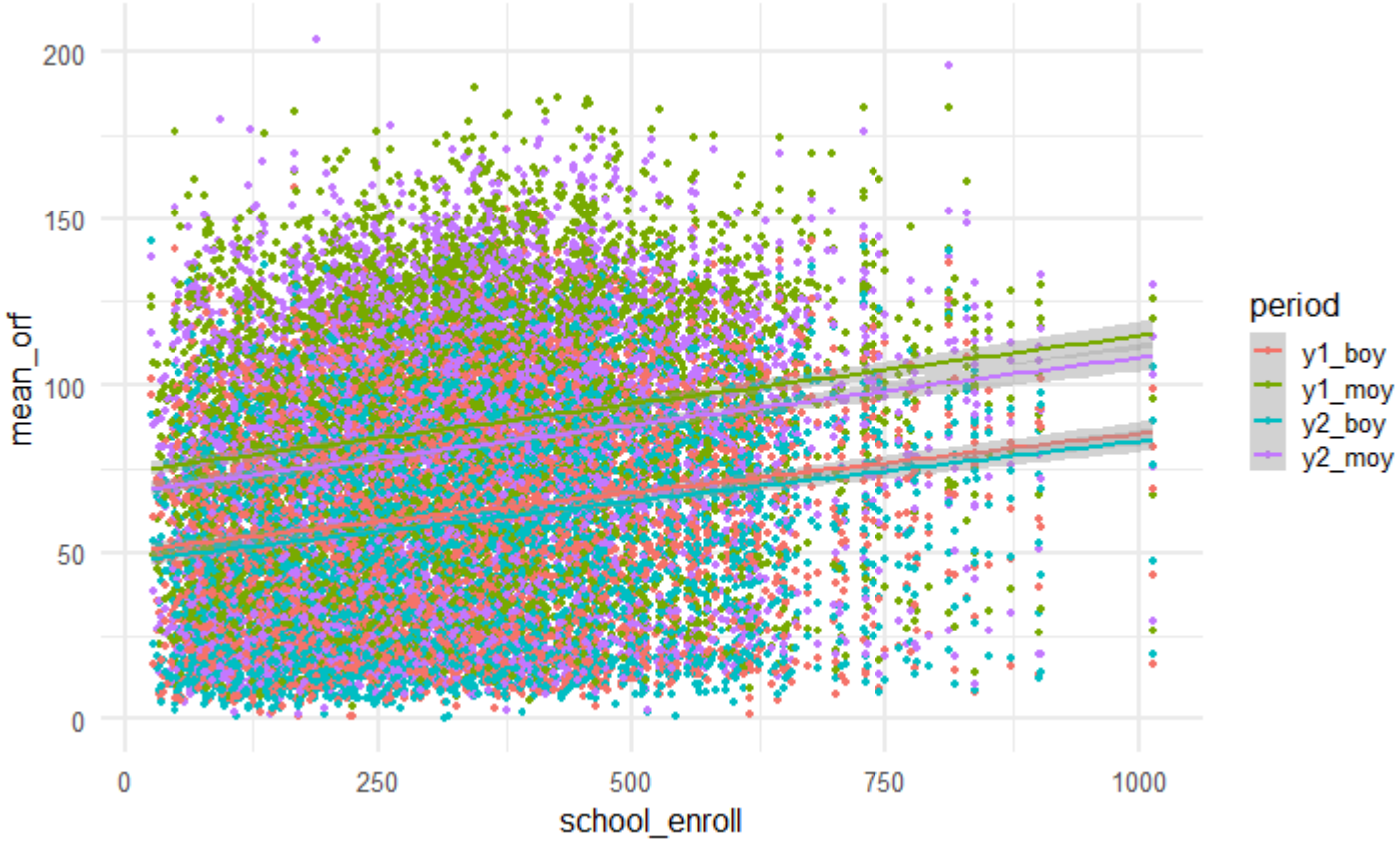
If our question predictor were continuous, and we wanted to adjust for a categorical, how might we do so? With just one other predictor?



Processing math: 100%

An aside

If our question predictor were continuous, and we wanted to adjust for a categorical, how might we do so? With just one other predictor?



Processing math: 100%

le regression assumption is being relaxed here?

An aside

If our question predictor were continuous, and we wanted to adjust for a categorical, how might we do so? With multiple predictors?

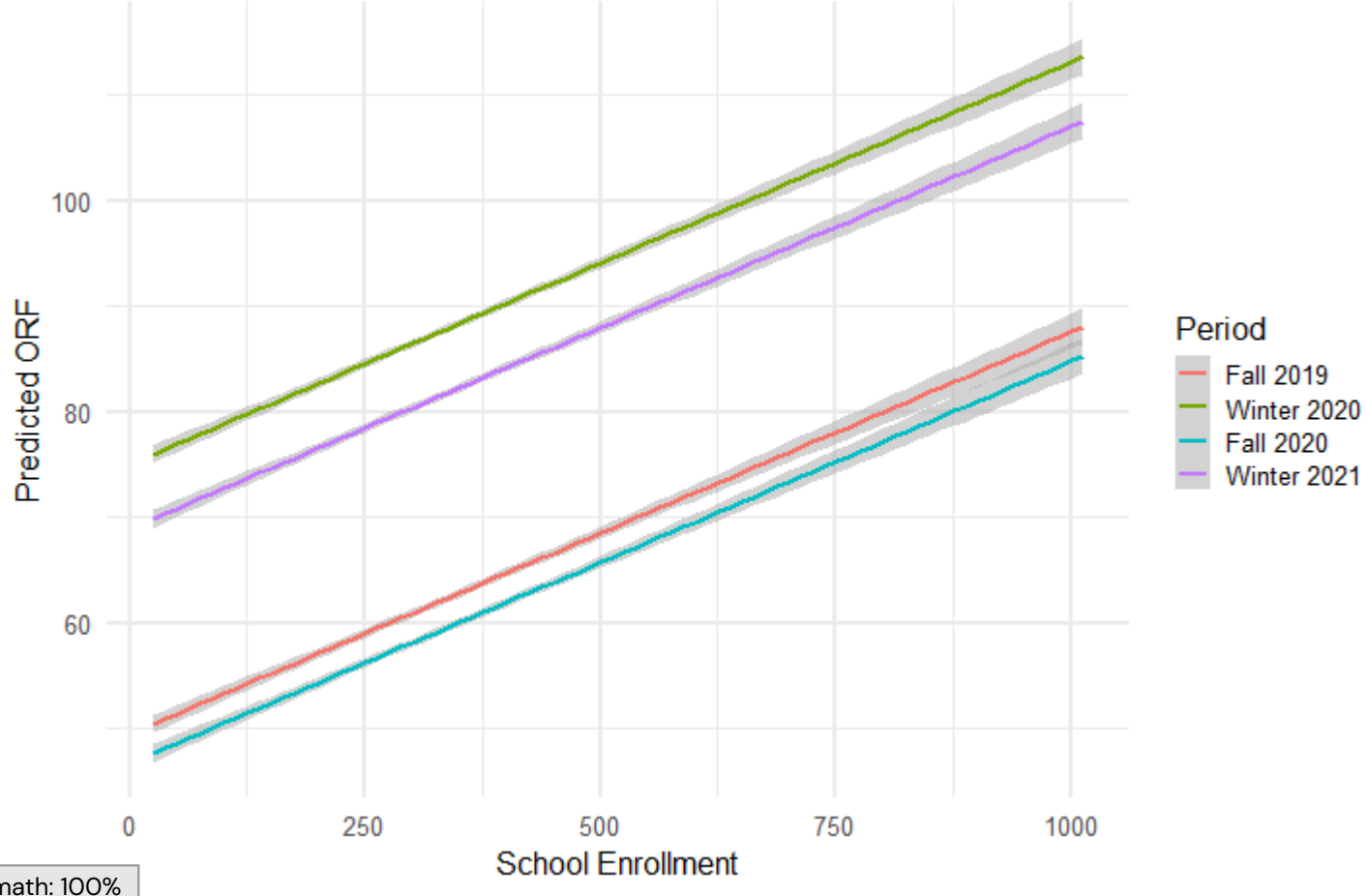
```
df3 <- margins::margins(fit4,
            at = list(period=c("y1_boy", "y1_moy",
                               "y2_boy", "y2_moy")))

# Use prototypical values in resulting dataset to show results
proto <-ggplot(data=df3, aes(x=school_enroll, y=fitted,color=period)) +
   geom_smooth(method='lm') +
   xlab("School Enrollment") + ylab("Predicted ORF") +
   scale_color_discrete(name = "Period",
                        breaks=c("y1_boy", "y1_moy",
                                 "y2_boy", "y2_moy"),
                        labels=c("Fall 2019","Winter 2020",
                                  "Fall 2020", "Winter 2021")) +
   theme_minimal(base_size=16)
```

Processing math: 100%

An aside

If our question predictor were continuous, and we wanted to adjust for a categorical, how might we do so? With multiple predictors?



Processing math: 100%

Alongside previous results

```
modelsummary(list(fit3, fit4),
             stars=T,
             escape=F,
             vcov = "robust",
             gof_omit = "Adj.|AIC|BIC|Log|RMSE|RSE|Std.Err",
             coef_rename = c("periody1_moy" = "Winter 2020",
                             "periody2_boy" = "Fall 2020",
                             "periody2_moy" = "Winter 2021",
                             "as.factor(grade)2" = "2nd Grade",
                             "as.factor(grade)3" = "3rd Grade",
                             "as.factor(grade)4" = "4th Grade",
                             "as.factor(grade)5" = "5th Grade",
                             "school_enroll" = "School Enrollment (#)"))
```

Processing math: 100%

Alongside previous results

	(1)	(2)
(Intercept)	62.344***	16.075***
	(0.449)	(0.326)
Winter 2020	25.562***	25.562***
	(0.696)	(0.332)
Fall 2020	−2.791***	−2.791***
	(0.631)	(0.311)
Winter 2021	19.454***	19.454***
	(0.700)	(0.330)
2nd Grade		38.954***
		(0.297)
3rd Grade		60.292***
		(0.345)
4th Grade		81.969***
		(0.406)

Processing math: 100%

Alternative format

	(1)	(2)
(Intercept)	62.344***	16.075***
	(0.449)	(0.326)
Winter 2020	25.562***	25.562***
	(0.696)	(0.332)
Fall 2020	-2.791***	-2.791***
	(0.631)	(0.311)
Winter 2021	19.454***	19.454***
	(0.700)	(0.330)
Covariates?	**No**	**Yes**
Num.Obs.	21584	21584
R2	0.102	0.795
+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001		
Cells report coefficients and heteroscedastic-robust standard errors in parentheses. Each observation is a school-grade-test value. Covariates include grade-level and total school enrollment.		

Processing math: 100%

Putting into words

Processing math: 100%

ANCOVA

Processing math: 100%

ANCOVA

- Analysis of covariance (ANCOVA) is an extension of ANOVA and multiple regression
- It is also a part of the broader family of General Linear Models
- The model relates categorical predictors to a continuous outcome, adjusting for the effects of other covariates
    - Note: you may see in some (older) sources the statement that ANCOVA models can only adjust for the effects of other continuous covariates. This is not true as long as you are careful to specify your categorical covariates as dummy indicators
- The null hypothesis is still the same as ANOVA ($\mu_1 = \mu_2 = \mu_K$).

Processing math: 100%

ANCOVA results

We can examine whether there are differences in the ORF scores by when students sat for the test, while adjusting for students' grade level and their school's size

```
anova(fit4)
```

```
## Analysis of Variance Table
##
## Response: mean_orf
##                     Df   Sum Sq Mean Sq   F value    Pr(>F)
## period               3  3204894 1068298  3583.474 < 2.2e-16 ***
## as.factor(grade)     4 21757555 5439389 18245.759 < 2.2e-16 ***
## school_enroll        1     4010    4010    13.452 0.0002453 ***
## Residuals        21575  6431895     298
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- As before, we reject the null and conclude that there is a difference, on average in the population, between waves of the ORF administration, *adjusting for the effects of students' grade and school size*
- However, our F-statistic is now **MUCH** bigger
- We've dramatically shrunk the RSS (28,193,461 vs. 6,431,752)

Processing math: 100%

ANOVA v. ANCOVA

Let's contrast an ANOVA with an ANCOVA test:

```
anova(fit3, fit4)
```

```
## Analysis of Variance Table
##
## Model 1: mean_orf ~ period
## Model 2: mean_orf ~ period + as.factor(grade) + school_enroll
##   Res.Df      RSS Df Sum of Sq     F    Pr(>F)
## 1  21580 28193461
## 2  21575  6431895  5  21761565 14599 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- In our basic ANOVA, our residual SS was 28,193,461 (it still is now). Now we see the residual SS for our ANCOVA fit is 6,431,752, or meaningfully (and statistically significantly) smaller
- Our variance has been "reorganized" with the addition of **school_enroll**.
- We can compare the two model fits with a new F-statistic that assesses whether one explains more of the variance ("is a better fit") than the other...it is.

Processing math: 100%

ANCOVA v. MR

We can also compare our ANCOVA output to our regression output and see our dummy-coded, "unbatched" analysis:

```
...
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        16.0746471  0.3715338  43.266  < 2e-16 ***
## periody1_moy       25.5620256  0.3324093  76.899  < 2e-16 ***
## periody2_boy       -2.7913676  0.3324093  -8.397  < 2e-16 ***
## periody2_moy       19.4541290  0.3324093  58.525  < 2e-16 ***
## as.factor(grade)2  38.9543683  0.3267633 119.213  < 2e-16 ***
## as.factor(grade)3  60.2924720  0.3572314 168.777  < 2e-16 ***
## as.factor(grade)4  81.9691911  0.3804983 215.426  < 2e-16 ***
## as.factor(grade)5  82.7431382  0.3913853 211.411  < 2e-16 ***
## school_enroll       0.0025891  0.0007059   3.668 0.000245 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.27 on 21575 degrees of freedom
## Multiple R-squared:  0.7952,    Adjusted R-squared:  0.7951
## F-statistic: 1.047e+04 on 8 and 21575 DF,  p-value: < 2.2e-16
...
```

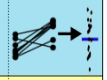
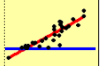
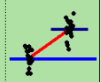
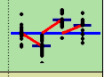
Processing math: 100%

Common statistical tests are linear models

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 $\mathbf{y_2}$-$\mathbf{y_1}$ differences. - (Same, but it predicts the *signed rank* of $\mathbf{y_2}$-$\mathbf{y_1}$.)	
	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: $G_{2\,to\,N}$ is an indicator (0 or 1) for each non-intercept levels of the **group** variable. Similarly for $S_{2\,to\,K}$ 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")`.

CC BY
Jonas Kristoffer Lindeløv
https://lindeloev.net

Processing math: 100%

It was the GLM the whole time...

Processing math: 100%

Synthesis and wrap-up

Processing math: 100%

Putting categorical predictors together

1. Regression models can easily include dichotomous and polychotomous predictors
    - Can be used for either nominal or ordinal predictors with sensible planning around dummy variables and the omitted reference category
2. All assumptions are about Y at particular values of X (or Xs)—no assumptions about the distribution of the predictors
3. The same toolkit we've developed for continuous predictors can be used for dichotomous and polychotomous predictors (including hypothesis tests, correlations and plots)
4. Be aware that when you introduce many categorical predictors you are implicitly engaging in multiple hypothesis testing
    - ANOVA/ANCOVA can help you address this, but be careful not to focus on just interpreting p-values
5. ANOVA/ANCOVA are just special cases of multiple regression
    - Can be useful to avoid problems of multiple hypothesis testing and understanding within- and between-variation
    - Can tell you little to nothing about the magnitude of group differences

Processing math: 100%

Goals for the unit

- Describe the relationship between dichotomous and polychotomous variables and convert variables between these forms, as necessary
- Conduct a two-sample t-test
- Describe the relationship between a two-sample t-test and regressing a continuous outcome on a dichotomous predictor
- Estimate a regression with one dummy variable as a predictor and interpret the results (including when the reference category changes)
- Estimate a multiple regression model with several continuous and dummy variables and interpret the results
- Estimate an ANOVA model and interpret the within- and between-group variance
    - Do the same for an ANCOVA model, adjusting for additional continuous predictors
- Describe the similarities and differences of Ordinary-Least Squares regression analysis and ANOVA/ANCOVA, and when one would prefer one approach to another
- Describe potential Type I error problems that arise from multiple group comparisons and potential solutions to these problems, including theory, pre-registration, ANOVA and *post-hoc* corrections
- Describe the relationship between different modeling approaches with the General Linear Model family

Processing math: 100%

To-Dos

Reading:

- **Finish by Feb. 18**: LSWR Chapter 14 and 16.6

Assignment 2:

- Due Feb. 14, 11:59pm

Assignment 3:

- Due Feb. 24, 11:59pm

Processing math: 100%

Pivot longer

```
dibels_long <- dibels %>%
                  pivot_longer(
                    cols = c("y1_boy_mean", "y1_moy_mean",
                             "y2_boy_mean", "y2_moy_mean"),
                    names_to = "period",
                    names_pattern = "(.*)_mean",
                    values_to = "mean_orf")
dibels_long$period <- factor(dibels_long$period)
```

Processing math: 100%