DSC 152:
Applied Statistical Data Analysis and Inference

Lecture #13
Pitfalls of Mixing Model Selection
with Statistical Inference

Tuesday, May 12
Spring Quarter 2026
Peter Chi

A common (flawed) workflow

from today's reading

The Annals of Statistics
2013, Vol. 41, No. 2, 802–837
DOI: 10.1214/12-AOS1077

VALID POST-SELECTION INFERENCE

BY RICHARD BERK, LAWRENCE BROWN[1], ANDREAS BUJA[1], KAI ZHANG[1] AND LINDA ZHAO[1]

University of Pennsylvania

It is common practice in statistical data analysis to perform data-driven variable selection and derive statistical inference from the resulting model. Such inference enjoys none of the guarantees that classical statistical theory provides for tests and confidence intervals when the model has been chosen a priori. We propose to produce valid "post-selection inference" by reducing

A common (flawed) workflow

from today's reading

1. Introduction: The problem with statistical inference after model selection. Classical statistical theory grants validity of statistical tests and confidence intervals assuming a wall of separation between the selection of a model and the analysis of the data being modeled. In practice, this separation rarely exists, and more often a model is "found" by a data-driven selection process. As a consequence inferential guarantees derived from classical theory are invalidated. Among model selection methods that are problematic for classical inference, *variable selection* stands out because it is regularly taught, commonly practiced and highly researched as a technology. Even though statisticians may have a general awareness that the data-driven selection of variables (predictors, covariates) must somehow affect subsequent classical inference from F- and t-based tests and confidence intervals, the practice is so pervasive that it appears in classical undergraduate textbooks on statistics such as Moore and McCabe (2003).

A common (flawed) workflow

To summarize, the (flawed) workflow is:

- Start with a dataset with one outcome variable and many covariates, one of which is your primary covariate of interest
  - Example:
    - **Outcome variable:** life expectancy
    - **Primary covariate of interest:** education level
    - **Other covariates (potential confounders):** region, SES at birth, current SES, ethnicity, family history of illnesses, dietary habits, exercise habits, etc.
  - Your question of interest is whether there is a relationship between education level and life expectancy

- Perform model selection to determine which set of covariates should be included in the "best model" (keeping education level in no matter what)

- From this "best model," use the p-value corresponding to education level to answer the original question of interest

So what's wrong with that?

Unfortunately, if we do the procedure described on the previous slide, our hypothesis testing will be invalid!

Recall: what do we mean by invalid?

Example

To illustrate the point, we will investigate the following simple example:

Just one potential confounder

Suppose that we are considering the model:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$$

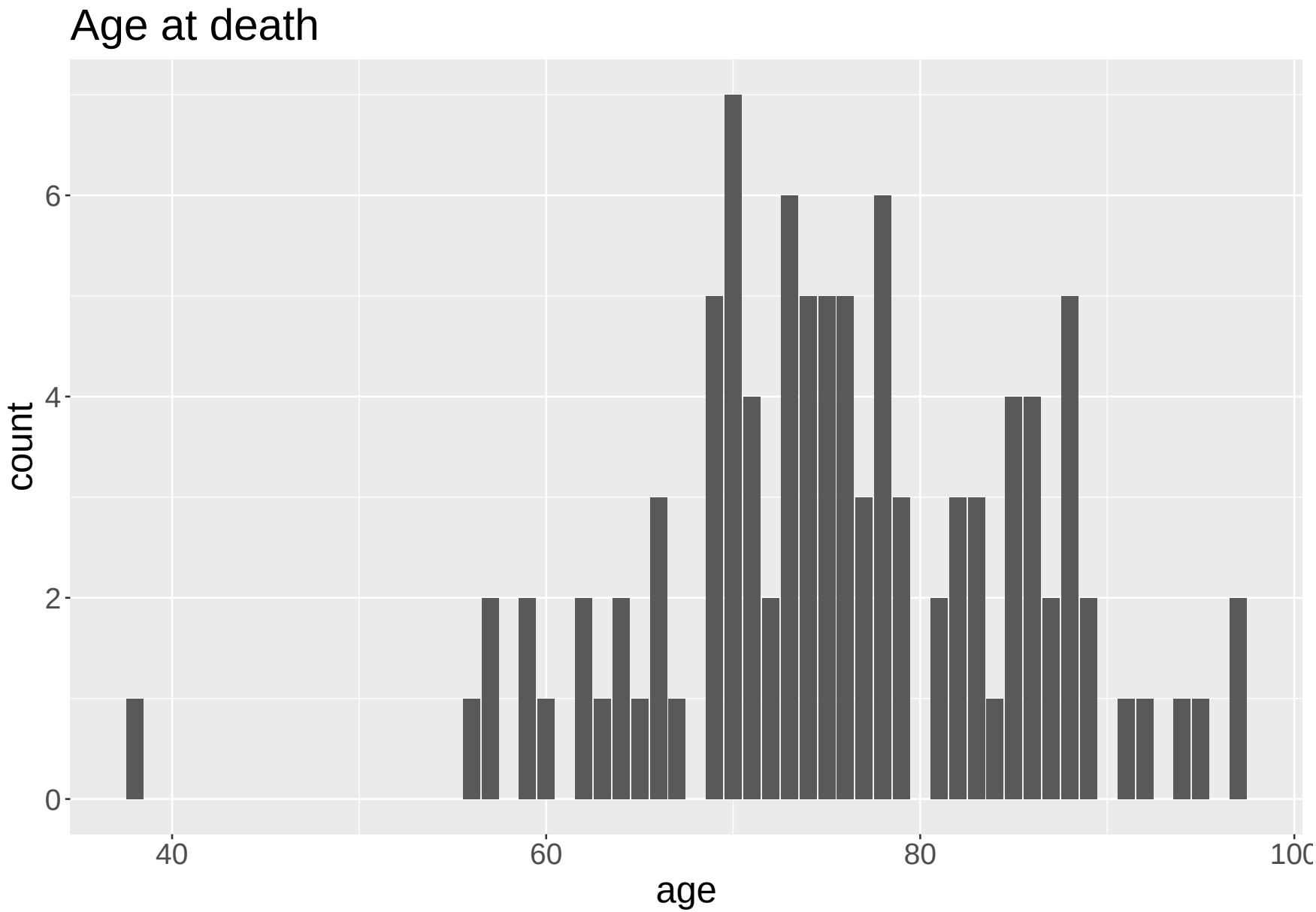
where x_1 is the primary covariate of interest, and x_2 is a potential confounding variable.

For example:

- let x_1 be the education level (in years of education)
- let x_2 be the amount that they typically exercised (in hours per week)
- let y be their age at death (in years)

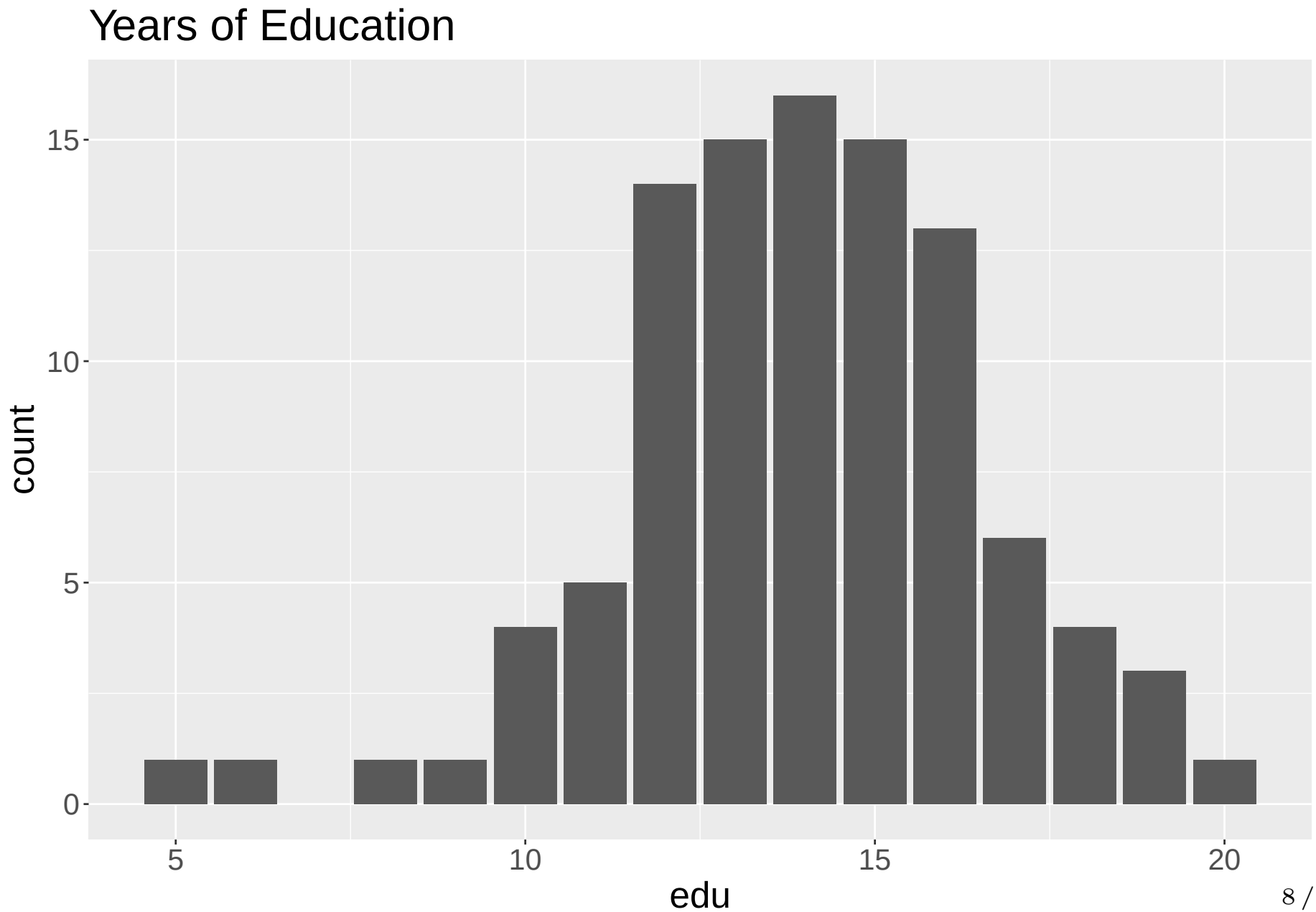
Example

These are simulated data...



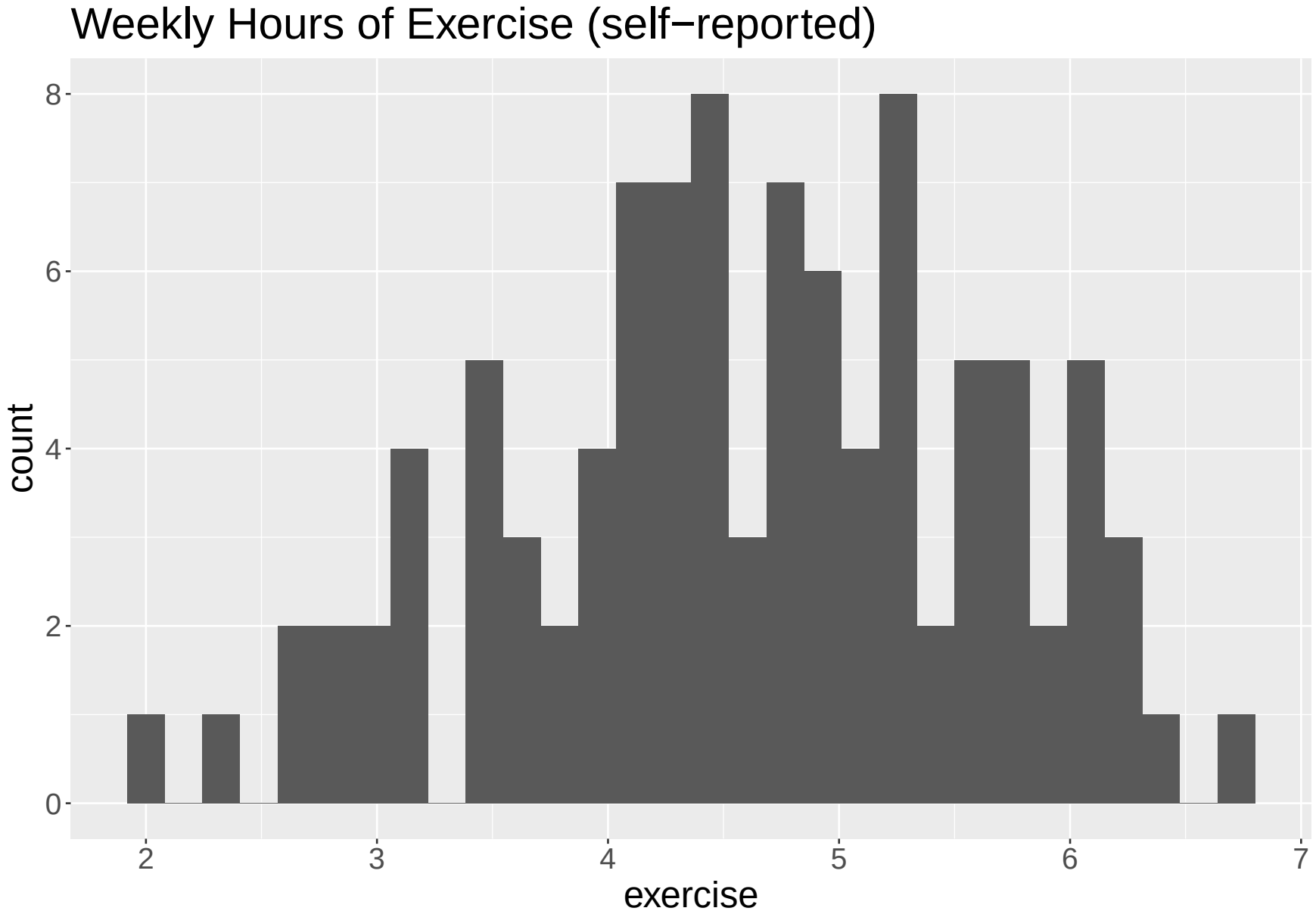
Example

These are simulated data...

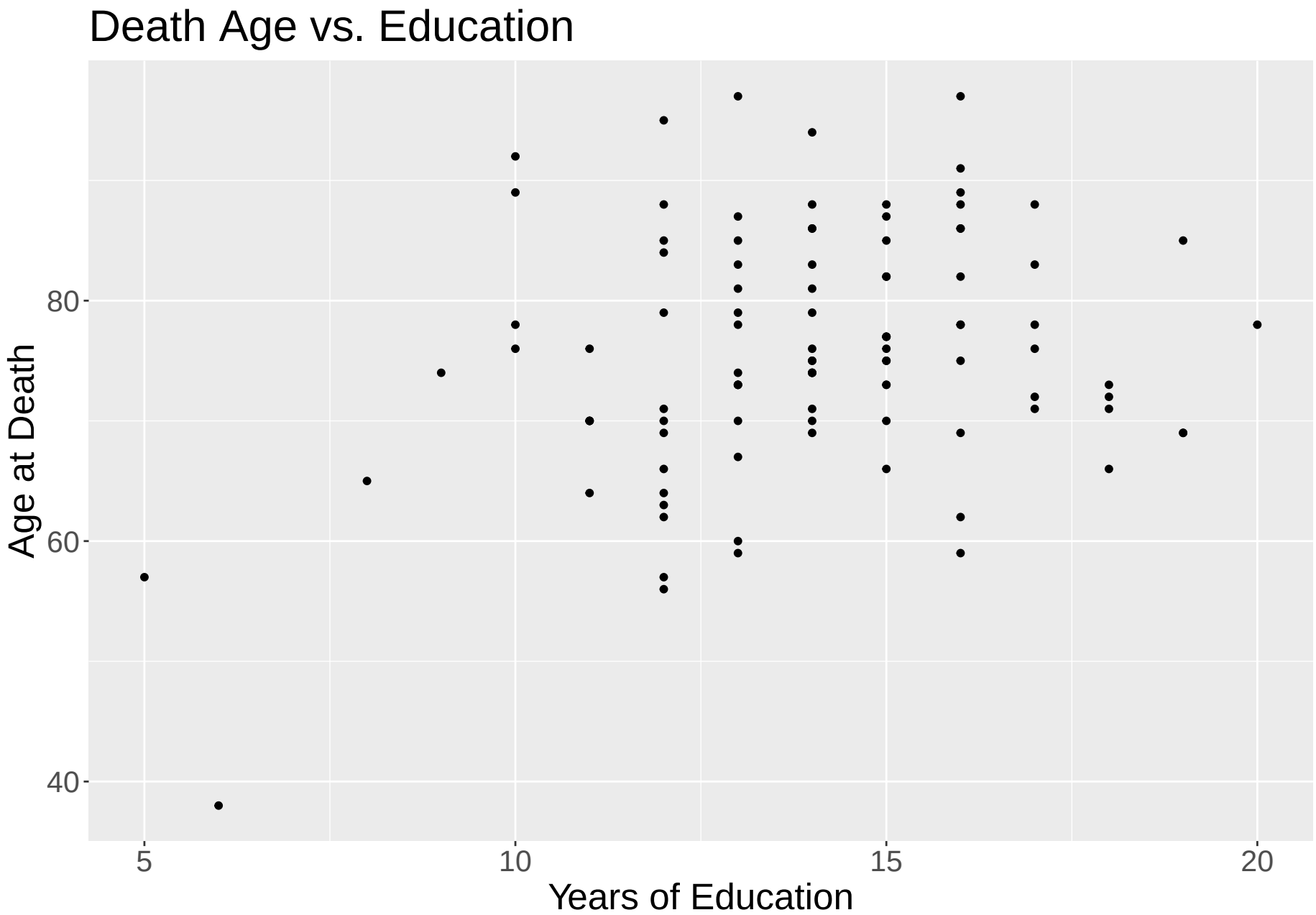


Example

These are simulated data...



Example


Death Age vs. Education
Age at Death
80
60
40
5
10
15
20
Years of Education

Example

```
summary(lm(age ~ edu + exercise, data=life_df))
```

```
##
## Call:
## lm(formula = age ~ edu + exercise, data = life_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.665  -6.377  -1.146   6.834  23.386
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  61.4936     5.2873  11.630   <2e-16 ***
## edu           0.5100     0.6681   0.763    0.447
## exercise      1.5554     1.7248   0.902    0.369
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.726 on 97 degrees of freedom
## Multiple R-squared:  0.07765,    Adjusted R-squared:  0.05863
## F-statistic: 4.083 on 2 and 97 DF,  p-value: 0.01984
```

Example

The flawed workflow would take the output on the previous slide and decide whether to keep `exercise` in the final model based on whether `exercise` is statistically significant in the model.

In this case, the p-value for `exercise` is 0.369, so we would remove `exercise` from the model.

Then, we run the model with just `edu` and get its p-value...

Example

```r
summary(lm(age ~ edu, data=life_df))
```

```
##
## Call:
## lm(formula = age ~ edu, data = life_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -29.7394  -6.8231  -0.8409   6.4598  22.1895
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  61.6785     5.2783  11.685  < 2e-16 ***
## edu           1.0102     0.3722   2.714  0.00785 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.717 on 98 degrees of freedom
## Multiple R-squared:  0.06992,    Adjusted R-squared:  0.06043
## F-statistic: 7.367 on 1 and 98 DF,  p-value: 0.007851
```

Example

But was this the right decision??

In reality, the data were simulated under:

- An association between exercise and age of death
- An association between exercise and education level
- No direct association between education level and age of death

So this workflow resulted in a Type I Error!

And it wouldn't have happened if we kept `exercise` in the model!!

Example

What happens on average?

Two Questions:

- If we repeat the simulation many times (10,000), what is the overall Type I Error rate on the test for education level?
- And if we just used the full model each time, would our Type I Error rate be ok?

Example

```r
count1 <- 0
count2 <- 0
reps <- 10000

for(i in 1:reps){
  # Simulate data
  x1 <- round(rnorm(100, mean=14, sd=2.5), 0) # integer values for years of education
  x2 <- round(rnorm(100, mean=(x1/3), sd=0.5), 2) # two decimals for exercise
  x2 <- ifelse(x2 > 0, x2, 0) # no negative values of exercise
  y <- round(70 + x2 + rnorm(100, sd=10), 0) # integer values for age

  # Run model
  model <- lm(y ~ x1+x2)

  # Check whether the p-value for exercise is significant
  if(summary(model)$coefficient[3,4] < 0.05){
    pval1 <- summary(model)$coefficient[2,4] # if it is, keep it in the model
  } else {
    model2 <- lm(y ~ x1)
    pval1 <- summary(model2)$coefficient[2,4] # if it's not, take it out
  }

  # Count how frequently the education p-value is significant
  if(pval1 < 0.05){
    count1 <- count1 + 1
  }

  # Also count how many times significant from the full model every time
  if(summary(model)$coefficient[2,4] < 0.05){
    count2 <- count2 + 1
  }
}
```

Example

Type I Error rate from using full model always

```
count2 / reps
```

```
## [1] 0.0521
```

Example

Type I Error rate from using full model always

```
count2 / reps
```

```
## [1] 0.0521
```

Type I Error rate when doing model selection followed by inference

```
count1 / reps
```

```
## [1] 0.1476
```

P.S. This is a big problem!!

Your Turn #1

In the previous example, there was a true causal pathway through the confounder of "exercise." It makes sense that we would have a problem in this case. (Why?)

But, is there still a problem when the secondary covariate does NOT actually have any association with the outcome variable? Here we will explore that.

Simulation study:

- Simulate a vector `x1` of size 100 where each observation follows a $N(0, 1)$ distribution
- Simulate a vector `x2` of size 100 where each x_{i2} observation follows a $N(x_{i1}, \sigma = 0.5)$ distribution; that is, each value of the `x2` vector is centered at the corresponding `x1` value, with some noise

Your Turn #1

Simulation study (continued):

- Simulate a vector `y` that has no relationship with `x1` or `x2`, and each observation just follows a $N(0, 1)$ distribution.

- Run the full model with `x1` and `x2` as covariates, then check whether the p-value for `x2` is less than 0.05
  - If it is, keep it in the model and grab the p-value for `x1`
  - If it is not, remove it from the model, run the linear model again without it, and grab the p-value for `x1` from that model
  - Increment a counter each time the p-value for `x1` is less than 0.05 under this procedure

- Increment a separate counter each time the full model itself gives a p-value of less than 0.05 for `x1`

- Put this in a loop, and compare the Type I Error rates for the model selection approach vs. just using the full model every time

What should we actually do?

As we have shown, the model selection $\rightarrow$ statistical inference workflow is NOT A GOOD IDEA!

So, what should we do instead?

A proper statistical inference workflow consists of the following:

- A presentation of summary statistics from the data, in the form of a table and usually one or more appropriate graphical representations

- The statistical inference must be done using a model that is pre-specified, either based on scientific rationale, prior studies, or a combination of both. This constitutes the "primary analysis."

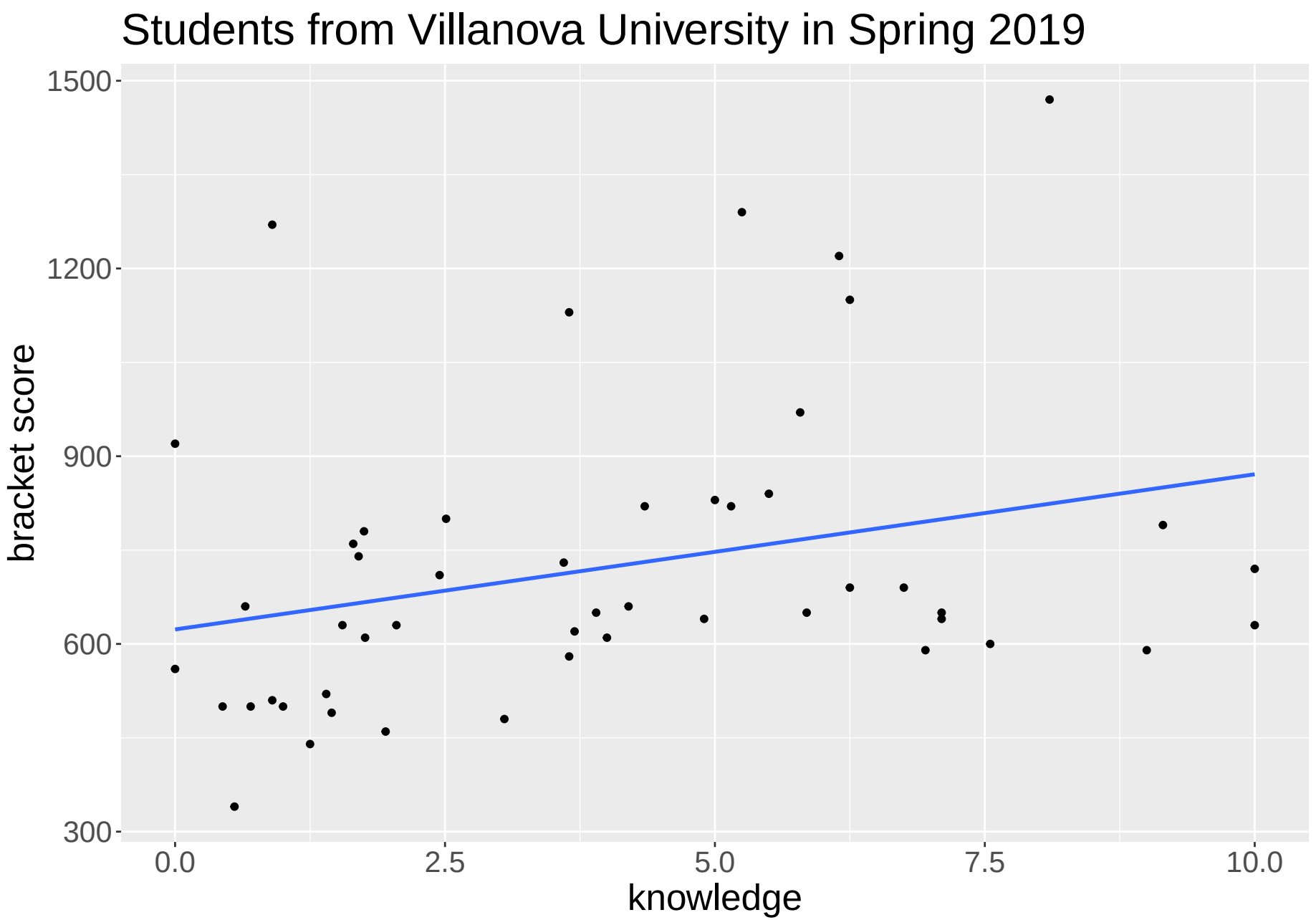
What should we actually do?

A proper statistical inference workflow consists of the following:

Once the primary analysis is done, THEN secondary analyses should be performed. Secondary analyses consist of:

- Model diagnostics (as discussed in Lecture #10)
- Sensitivity analyses: what WOULD have happened if we had fit a different model? Here you can do any model building procedure to find potential "better" models.
- Possibly a permutation test (particularly if model diagnostics give you any cause for concern that can't be fixed by model building)

Example: March Madness (again)



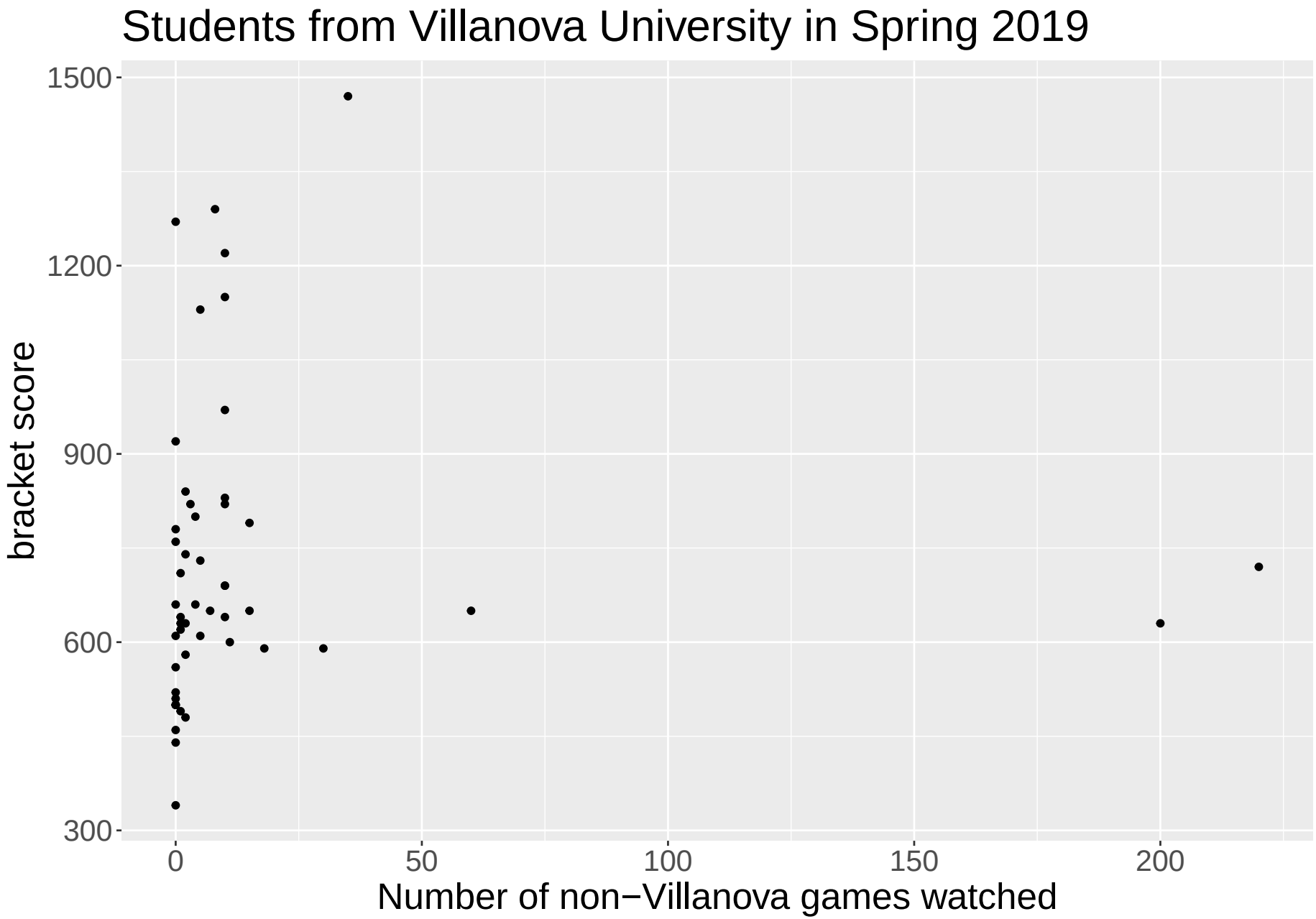
Example: March Madness (again)

Now let us consider the full dataset, which can be found under Lecture #7 on the course website. It has the following columns:

columns	description
nova_games	# of Villanova games watched
other_games	# of non-Villanova games watched
name1_player	# of teams for which you can name 1 player
name2_players	# of teams for which you can name 2 players
name_coach	# of teams for which you can name the head coach
identify_strat	how frequently can you identify strategies on the court
used_knowledge	to what degree did you use your knowledge in your picks
bracket_score	final score on the bracket
knowledge	overall knowledge score from 0 to 10

Suppose that our primary interest is actually in whether there is a relationship between `other_games` and `bracket_score`.

Example: March Madness (again)



Example: March Madness (again)

Your Turn #2

Let's walk through these components of the statistical inference workflow:

- Performing a primary statistical analysis using a pre-specified model

- Sensitivity analysis: backward selection using p-values
  - Note: this is one of the simplest model building procedures possible; there are many other possibilities that you may have learned about in other classes. You are welcome to use them in this class but I will not cover them.

Example: March Madness (again)

Your Turn #2

For the primary analysis, let's suppose that we think that all of the other covariates (except `knowledge`) should be included in the model.

We would do this because we have scientific rationale for thinking that this is the right thing to do.

Example: March Madness (again)

Your Turn #2

```
model_all <- lm(bracket_score ~ nova_games+other_games+name1_player+name2_players+name_coach+
                  identify_strat+used_knowledge, data=march_madness)
summary(model_all)
```

```
##
## Call:
## lm(formula = bracket_score ~ nova_games + other_games + name1_player +
##     name2_players + name_coach + identify_strat + used_knowledge,
##     data = march_madness)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -317.54 -139.82  -21.16   77.85  614.10
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              429.458    147.379   2.914  0.00588 **
## nova_games                 5.668      4.489   1.263  0.21417
## other_games               -1.859      1.952  -0.952  0.34687
## name1_player               3.546     14.877   0.238  0.81286
## name2_players            -13.892     21.364  -0.650  0.51936
## name_coach                20.375     15.261   1.335  0.18958
## identify_stratNever      196.571    164.550   1.195  0.23946
## identify_stratRarely     128.553    157.337   0.817  0.41886
## identify_stratSometimes  208.339    155.380   1.341  0.18773
## identify_stratUsually     40.090    139.458   0.287  0.77527
## used_knowledge             1.764      1.548   1.140  0.26134
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 228.1 on 39 degrees of freedom
## Multiple R-squared:  0.2694, Adjusted R-squared:  0.08206
```

Example: March Madness (again)

Your Turn #2

Primary Analysis

The p-value for `other_games` from this primary analysis is 0.3468729 so we would fail to reject H_0 at $\alpha = 0.05$.

Sensitivity Analysis

Now, to do backward selection using the p-values, we remove covariates one-by-one, in order of the highest p-value. We stop when all p-values are less than 0.05.

Note that `identify_strat` is a categorical variable with 4 dummy variables, so these need to be either all in the model or all removed. This means that it requires a partial F-test!

For the sake of simplicity, let us assume that it's just not going to be in the final model and remove it first (otherwise, we would need to do the partial F-test on every step, which will get somewhat painful).

Example: March Madness (again)

Your Turn #2

The entirety of Your Turn #2 is:

1. Do the primary analysis and report its conclusions (shown on the previous slides)

2. Do backward selection with p-values, showing the model summary at each step, and then report what the final model is under this procedure and the p-value for the primary covariate of interest in that model, along with brief comments on what you observed.

Recap and Looking Ahead

Recap

- Model Selection → Inference is not a good workflow!
- Model Selection is a fine thing to do if you are doing a purely Machine Learning endeavour, or as part of Secondary Analyses of a Statistical Inference Workflow

Looking Ahead

Transformations

Today's Daily Check

Both Your Turns