👩‍💻 building interactive tutorials in R

🔗 bit.ly/teach-r-online-mats

dr. mine çetinkaya-rundel
dr. colin rundel

Activity

while we wait to get started...

- Go to INSERT LINK
- Start working through the tutorial
- Feel free to make mistakes and test out the feedback
- If you get to the very end, follow the instructions (but if they seem a bit opaque, don't fret, we'll say more about "submission" later...)

you...

- know and teach R
- are familiar with R Markdown
- are interested in providing automated feedback
- might be interested in automated marking

why

auto feedback

Nudging

students towards the right answer,
especially in formative assessments

Sample question:

Suppose 10 means from a simulated sampling distribution is stored in a vector called `means`.

```
means
```

```
##  [1] -1.21  0.28  1.08 -2.35  0.43  0.51 -0.57 -0.55 -0.56 -0.89
```

What is the value of the first mean?

Sample answer:

```
mean[1]
```

```
## Error in mean[1]: object of type 'closure' is not subsettable
```

Nudging

Not all feedback is useful, at least not for beginners...

Providing helpful feedback can help them nudge them towards success:

```
mean[1]
```

```
## x `mean` is a function and a function doesn't have elements that can be subsetted with square brackets.

## ℹ `means` is the vector of sample means calculated earlier.
```

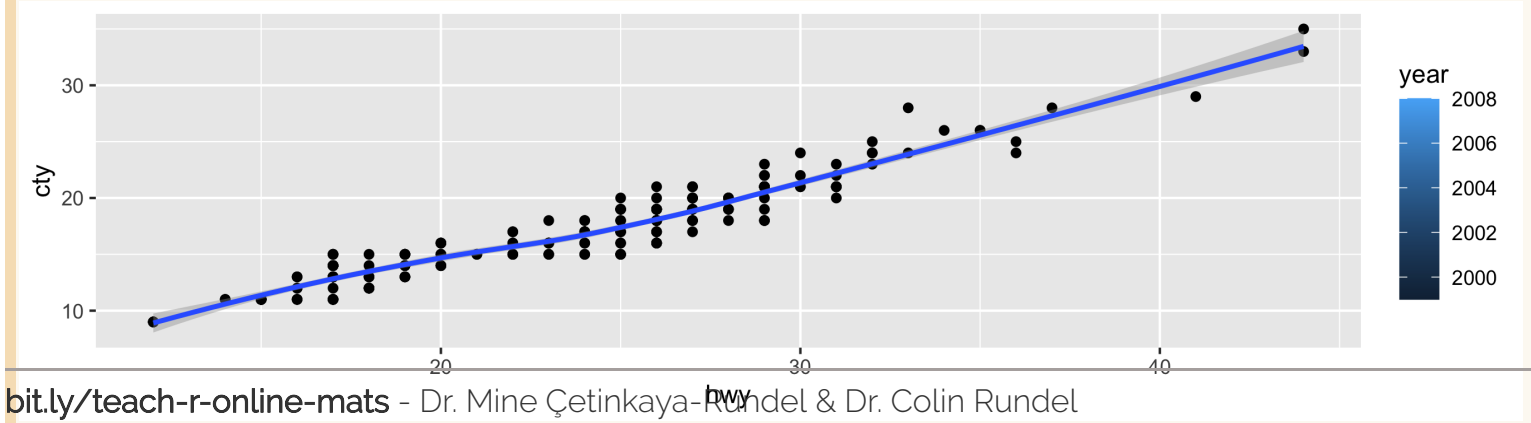
Sample question:

Visualise the relationship between city and highway mileage of cars from the `mpg` dataset, conditional on year of manufacture.

Sample answer:

There is a strong, positive, linear relationship between the city and highway mileage of cars. Year does not seem to be related to either variable.

```
ggplot(mpg,aes(x = hwy, y = cty, fill=year)) +geom_point()+geom_smooth()
```



Sample feedback:

- You mention a linear relationship, however your plot uses a loess fit to visualise the relationship between city and highway mileage. Also, the plot displays the uncertainty around the fit, but you haven't addressed it in your narrative.
- Year should be mapped to the `color` aesthetic, not `fill`.
- Plot styling: Use informative axis labels, noting units of measurement. Also, give an informative title to your plot.
- Code styling: Use consistent spacing around operators (e.g `=`) and line breaks after `+` in each layer of your ggplot.

👉

Nudging

students towards the right answer, especially in formative assessments

\+

Scaling

up efficiency of grading faster than (human) resources

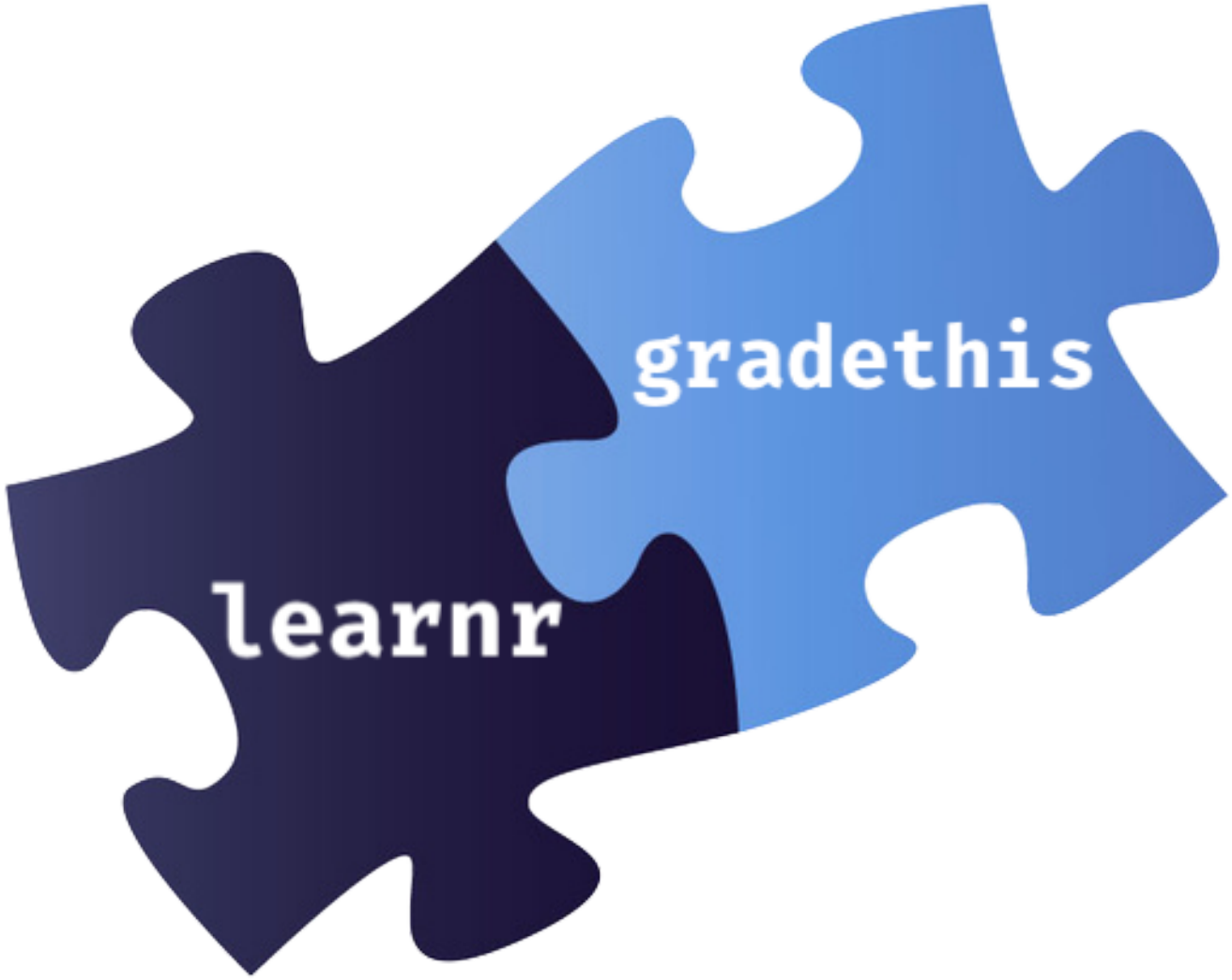
Scaling

Our courses are growing, and that's a good thing, right?

- Students turning in their work as R Markdown documents makes collecting submissions including code and narrative straightforward.
- Providing feedback on both the code and narrative is not scalable unless (human) resources dedicated to your course grow proportionally with enrolments.

how

auto feedback


gradethis
learnr

learnr

- learnr is an R package that makes it easy to create interactive tutorials from R Markdown documents.
- Tutorials can include:
  - Narrative, figures, illustrations, and equations
  - Code exercises (R code chunks that users can edit and execute directly)
  - Multiple choice questions
  - Videos (YouTube, Vimeo)
  - Interactive Shiny components
- learnr is on CRAN

```
install.packages("learnr")
```

Lego Sales

- Introduction
- Data
- Counting frequencies
- Discretizing variables
- Grouped summaries
- Wrap Up

Start Over

progress bar

Most common age group

Count the number of customers in each age group and display the counts in descending order, from most common to least common age group.

narrative

Which age group is the most common? Write code in the chunk below to figure it out!

Code | Start Over | Solution | Run Code | Submit Answer

```
1 lego_sales %>%
2   ___(___)
3
```

code exercise

Now, based on your findings, answer the following question:

Which age group is the most common in the dataset?

- ○ 36 - 50
- ○ 26 - 35
- ○ 51 and over
- ○ 18 and under
- ○ 19 - 25

Submit Answer

multiple choice question

Continue

 gradethis

- Companion to the learnr package, **gradethis** provides multiple methods to grade learnr exercises:
  - `grade_code()`: Grade code against a solution
  - `grade_conditions()`: Grade all specified conditions
  - `grade_result()`: Grade result of exercise code
- gradethis is not yet on CRAN

```
devtools::install_github("rstudio/gradethis")
```

demo

[tutorial]

[code]

learnr

YAML

Start with a YAML, just like in R Markdown:

```
---
title: "Lego Sales"
output:
  learnr::tutorial:
    progressive: true
    allow_skip: true
    css: "css/font-size.css"
runtime: shiny_prerendered
---
```

- `runtime: shiny_prerendered`
- `progressive: true` for "Continue" buttons between subsections
- `alow_skip: true` to allow skipping exercises

Customization

- You can change the style of your learnr tutorial
- You might, at a minimum, implement a couple customizations for accessibility:
  - Increase font size in the narrative, using a CSS file that lives in a directory called `css/` and loaded in the YAML with

```
css: "css/font-size.css"
```

- Increase font size in code boxes, using a JS file that lives in a directory called `js/` and loaded with

```
<script language="JavaScript" src="js/exercise-font-size.js"></script>
```

Narrative

- R Markdown style section and subsection headings with ##, ###, etc.
- Text, figures, illustrations, and equations.
- Videos: supported services include YouTube and Vimeo

```
### Learning goals

- Create frequency tables and arrange them in ascending / descending order
- Convert numerical variables into ordinal variables by grouping ranges
- Calculate summary statistics for groups in your data

### Getting help

If you have any questions about the assignment, please post them on [Piazza](https://piazza.com)!
```

Multiple choice questions

```
quiz(
  question("What number is the letter A in the alphabet?",
    answer("8"),
    answer("14"),
    answer("1", correct = TRUE),
    answer("23"),
    incorrect = "See [here](https://en.wikipedia.org/wiki/English_alphabet) and try again.",
    allow_retry = TRUE
  ),

  question("Where are you right now? (select ALL that apply)",
    answer("Planet Earth", correct = TRUE),
    answer("Pluto"),
    answer("At a computing device", correct = TRUE),
    answer("In the Milky Way", correct = TRUE),
    incorrect = paste0("Incorrect. You're on Earth, ",
                       "in the Milky Way, at a computer.")
  )
)
```

Text entry questions

```
question_text(
  "Please enter the word 'C0rrect' below:",
  answer("correct", message = "Don't forget to capitalize"),
  answer("c0rrect", message = "Don't forget to capitalize"),
  answer("Correct", message = "Is it really an 'o'?"),
  answer("C0rrect ", message = "Make sure you do not have a trailing space"),
  answer("C0rrect", correct = TRUE),
  allow_retry = TRUE,
  trim = FALSE
)
```

Your turn!

We strongly recommend one person in each group to share their screen.

- Go to bit.ly/teach-r-online-cloud
- Log in with Google or GitHub, or create a new account
- Once you join our workspace, start the assignment titled `Palmer penguins`
- Open `penguins.Rmd` and click on **Run Document**
- Read the instructions under `Multiple choice questions`
- See help for `?question()` and `?quiz()` and don't hesitate to call for help!

10:00

Code exercises - rendered

Most common subtheme

Among the most common theme of Lego sets purchased, what is the most common subtheme?

Code | Start Over | Hints | Run Code | Submit Answer

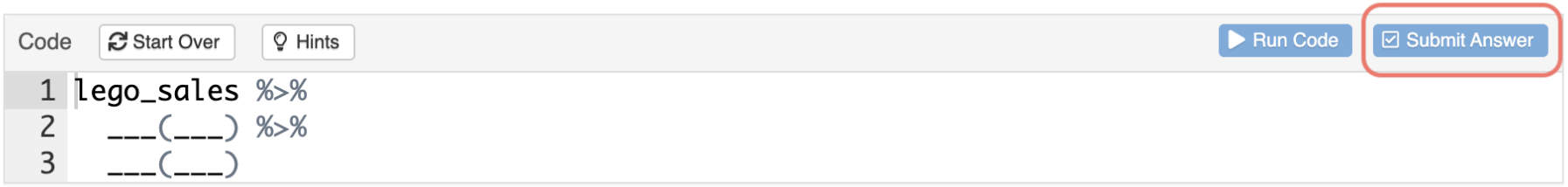
```
1 lego_sales %>%
2   ___(___) %>%
3   ___(___)
```

Code exercises - code

Code exercises - solution

Most common subtheme

Among the most common theme of Lego sets purchased, what is the most common subtheme?




````
```{r most-common-subtheme-solution}
lego_sales %>%
  filter(theme == "Star Wars") %>%
  count(subtheme, sort = TRUE)
```
````

gradethis

Most common subtheme

Among the most common theme of Lego sets purchased, what is the most common subtheme?

Code | Start Over | Hints | Run Code | Submit Answer

```
1 lego_sales %>%
2   filter(theme == "Star Wars") %>%
3   count(theme)
```

theme <chr>	n <int>
Star Wars	75

1 row

Did you count themes instead of subthemes? Don't give up now, try it one more time.

Most common subtheme

Among the most common theme of Lego sets purchased, what is the most common subtheme?

Code | Start Over | Hints | Run Code | Submit Answer

```
1 lego_sales %>%
2   filter(theme == "Star Wars") %>%
3   count(subtheme)
```

subtheme <chr>	n <int>
Battlefront	7
Buildable Figures	11
Episode III	6
Episode IV	2
Episode V	10
MicroFighters	10
Original Content	7
Rebels	3
Seasonal	3
The Force Awakens	15

1-10 of 11 rows Previous 1 2 Next

Did you forget to sort the counts in descending order? Give it another try.

Most common subtheme

Among the most common theme of Lego sets purchased, what is the most common subtheme?

Code | Start Over | Hints | Run Code | Submit Answer

```
1 lego_sales %>%
2   filter(theme == "Gear") %>%
3   count(subtheme)
```

subtheme <chr>	n <int>
Digital media	1
Digital Media	2
Key Chains/City	3
Key Chains/Friends	6
Key Chains/Miscellaneous	6
Key Chains/Ninjago	5
Playmats	2
Role-Play toys	7
Stationery	7
Video Games/3DS	2

1-10 of 14 rows Previous 1 2 Next

Not quite. Take a peek at the hint! Please try again.

Checking the result

- Use a code chunk with the same label, suffixed with `-check`
- `.result` refers to the resulting output
- Think about ways things can go wrong and write test cases for them
- Write a "catch all" test case for everything else

````
```{r most-common-subtheme-check}
grade_result(
  pass_if(~ identical(as.character(.result[1,1]), "The Force Awakens"), "You have successfully counted subthemes and sorted the counts in descending order."),
  fail_if(~ identical(as.character(.result[1,1]), "Battlefront"), "Did you forget to sort the counts in descending order?"),
  fail_if(~ identical(as.character(.result[1,1]), "Ultimate Collector Series"), "Did you accidentally sort the counts in ascending order?"),
  fail_if(~ identical(as.character(.result[1,1]), "Star Wars"), "Did you count themes instead of subthemes?"),
fail_if(~ TRUE, "Not quite. Take a peek at the hint!")
)
```
````

Your turn!

We strongly recommend one person in each group to share their screen.

- Go to bit.ly/teach-r-online-cloud
- In the assignment titled `Palmer penguins`, open `penguins.Rmd` and click on **Run Document**
- Read the instructions under `Code exercises`
- Write more hints and code checking tests

15:00

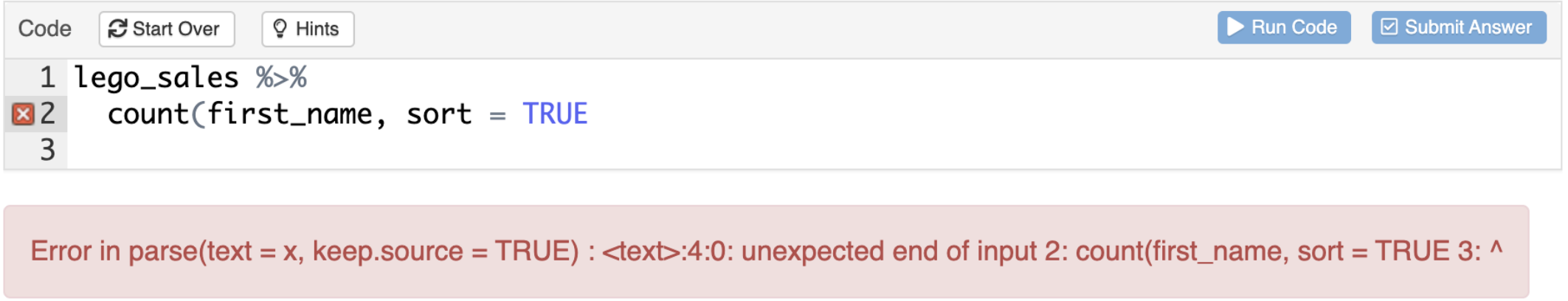
Other checking options

- `grade_code()`: Grade code against a solution
- `grade_conditions()`: Grade all specified conditions
- See `gradethis::gradethis_demo()` for a walk through of how each of these options work

Known challenges

- Currently learnr gives a warning if the exercise code chunk produces an invisible result, e.g. code chunk makes an assignment instead of outputting a result -- we think this will be fixed soon [PR].

- If code in the exercise chunk is invalid, you might get the R error instead:



sharing

Sharing with students

- You could share the R Markdown document (and all accompanying files) but that's probably not what you want to do...
- Deploy on
  - shinyapps.io
  - RStudio Connect (free for academic use, requires setup)
- Road less travelled: distribute as a package
- See the publishing instructions on the learnr website for step-by-step instructions

recording data

Recording attempts

- A "good enough" solution for formative exercises: embed a Google/Microsoft/etc. Form at the end and ask students to "submit" their work.

- This only records that the student reached the end of the tutorial and not how (or even if) they answered any of the questions or exercises.

- Tip: Add a free-text question to the form asking students to reflect on the exercises they just completed - you can then analyse the free-text data to gain insights into what students are struggling with -- "minute paper".

`[example]`

Recording solutions

- The `learnrhash` package builds on the previous method by providing a way for students to submit their answers by generating a text "hash" which can be copy and pasted into the web form.

```
devtools::install_github("rundel/learnrhash")
```

- This package is being used to create the submission tool at the end of the sample tutorial.

See also the **submitr** package by Danny Kaplan for a different approach to recording event data in learnr tutorials.

demo

[tutorial]

[code]

Submissions

Since I've just submitted, my details and my hash will now be stored in a Google Sheet - if you would like to play around with this data (Instructor mode) you can access it here:

https://bit.ly/tutorial-data

We don't have time today to demo reading in and decoding these data, but we will include code for the whole process in the learnrhash package repository in the next week, using data from your submissions.

closing thoughts

Best practices for automated feedback

- Measure twice, cut once (verify the correctness of your tests) 📏📏✂️
- Use rounding & type coercion to write robust tests 💪
- Don't give automated feedback on everything, asking narrative questions that can't be auto checked but gets the student thinking and writing has pedagogical benefits 🙈
- Consider peer feedback where automated feedback is not feasible (e.g. interpretation, narrative) but scalability is an issue 👯

Q: What is an approachable way to get started?

Build a tutorial where students build develop their analysis in exercise code chunks (that are not checked) and only multiple choice questions are used for assessment. `[example]`

Q: I've built simple tutorials already. How do I make the jump to code checking and providing automated feedback that is actually useful?

- Replicate `gradethis::gradethis_demo()`, then make incremental changes
- Read the Testing chapter in R Packages (Wickham and Bryan)
- Also read the Metaprogramming section in Advanced R (Wickham)

Q: Sounds great, but can it handle my class size and usage?

- First, chances are you're not using these live, but you might be...
- If so, make sure to max out your instances and instance size on shinyapps.io.
- Essential reading:
  - Publishing learnr Tutorials on shinyapps.io by Angela Li
  - Teach R with learnr: a powerful tool for remote teaching by Allison Horst

thank you!

- All materials at bit.ly/teach-r-online-mats
- Sign up for the upcoming workshop at bit.ly/teach-r-online:
  - 17 July: Teaching computing with Git and GitHub