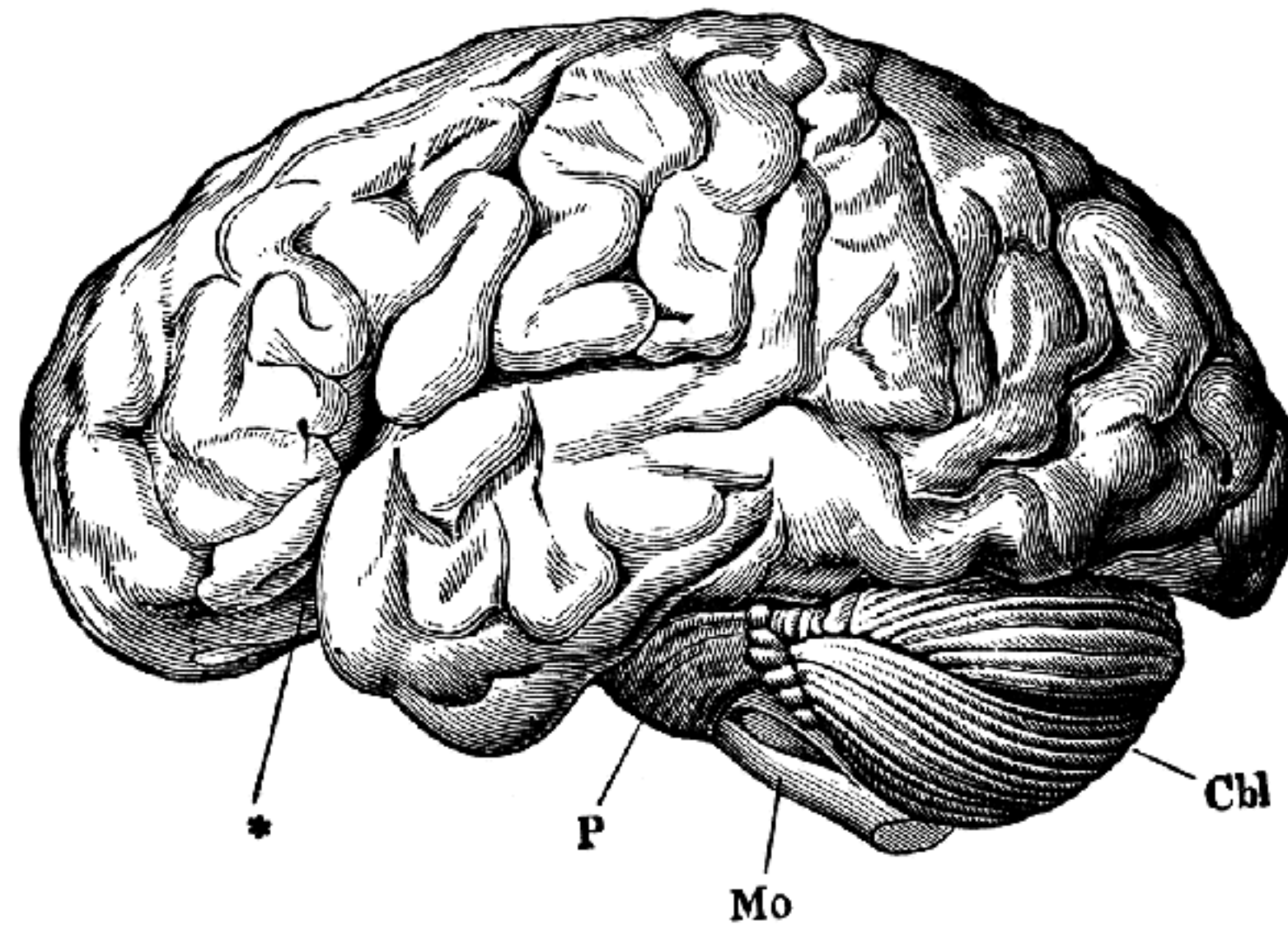


Make Interactive Web Tutorials with **learnr** and R



Garrett Grolemond

Educator, RStudio

March 2018

Video at
vimeo.com/261761874

Garrett Grolmund

Educator, RStudio

March 2018



learnr



Uses

1. Flipped classrooms
2. Lab-like supplements
3. Self-learning
4. Learn by doing
5. Package tutorials
6. Workshop follow ups

To install

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

rstudio.github.io/learnr

The screenshot shows a web browser window with the URL `https://rstudio.github.io/learnr/index.html`. The page has a blue header with navigation links: **learnr**, Home, Exercises, Questions, Publishing, Formats, and Examples. A sidebar on the left contains a menu with the following items: Overview (selected), Getting Started, Tutorial Types, Exercises, Questions, Videos, Shiny Components, External Resources, Preserving Work, and Publishing. The main content area is titled "Interactive Tutorials for R Overview".

The **Overview** section states: "The **learnr** package makes it easy to turn any [R Markdown](#) document into an interactive tutorial. Tutorials consist of content along with interactive components for checking and reinforcing understanding. Tutorials can include any or all of the following:

1. Narrative, figures, illustrations, and equations.
2. Code exercises (R code chunks that users can edit and execute directly).
3. Quiz questions.
4. Videos (supported services include YouTube and Vimeo).
5. Interactive Shiny components.

Tutorials automatically preserve work done within them, so if a user works on a few exercises or questions and returns to the tutorial later they can pick up right where they left off.

Examples

Here are some simple examples of tutorials created with the **learnr** package:

Data basics

Tibbles

What is a tibble?

How many rows as a specific type of data frame that you will encounter in the wild.

The `flights` data frame in the `tidyverse` package is an example of a tibble. `flights` describes every flight that departed from New York City in 2013. The data comes from the US Bureau of Transportation Statistics, and is documented in [this page](#).

Use the code chunk below to print the contents of `flights`.

```
library(tidyverse)
flights
```

```
# A tibble: 331,174 x 14
#   year month   day dep_time sched_dep_time dep_delay arr_time
#   <int> <int> <int> <time>         <time>         <dbl>   <time>
#1  2013     1     1   5:00      5:00             0  5:40
#2  2013     1     1   5:00      5:00             0  5:40
#3  2013     1     1   5:00      5:00             0  5:40
#4  2013     1     1   5:00      5:00             0  5:40
#5  2013     1     1   5:00      5:00             0  5:40
#6  2013     1     1   5:00      5:00             0  5:40
#7  2013     1     1   5:00      5:00             0  5:40
#8  2013     1     1   5:00      5:00             0  5:40
#9  2013     1     1   5:00      5:00             0  5:40
#10 2013     1     1   5:00      5:00             0  5:40
# ... with 331,164 more rows, and 3 more variables: actual_dep_time, actual_delay, actual_arr_time
```

Filter observations

Filter rows with `filter()`

`filter()` allows you to subset observations based on their values. The first argument is the name of the data frame. The second and subsequent arguments are the expressions that filter the data frame. For example, we can select all flights on January 1st with:

```
flights %>% filter(month == 1, day == 1)
```

Code

```
flights %>% filter(month == 1, day == 1)
```

Boolean operations

Missing values

Exercises

year	month	day	dep_time	sched_dep_time	dep_delay	arr_time
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40
2013	1	1	5:00	5:00	0	5:40

Summarize Tables

Summarize groups with `summarize()`

Combining multiple operations

Useful summary functions

Count

Exercises

Combining multiple operations

Multiple steps

Imagine that we want to explore the relationship between the distance and average delay for each destination in `flights`. Along what you know about `dplyr`, you might write code like this:

```
flights %>% group_by(dest) %>%
  summarise(
    dist = mean(distance, na.rm = TRUE),
    delay = mean(delay, na.rm = TRUE)
  )
```

Applied data = `delay`, `mapping` = `aes(x = dist, y = delay)` + `geom_point(mapping = aes(x = dist, y = delay))` + `geom_smooth(mapping = aes(x = dist, y = delay))`

Where next?



Challenges

1. Code checking

learnr



Exercise Checking

The **learnr** package doesn't directly include features for checking exercise inputs however it does include lower-level hooks that enable other packages to provide tools for exercise checking. You can provide an external function for exercise checking by setting the `exercise.checker` knitr chunk option within the `setup` chunk of your tutorial, then adding a “-check” chunk for any exercise you want to check. For example:

checkr

The screenshot shows a web browser window displaying the GitHub repository page for 'checkr' by user 'dtkaplan'. The browser's address bar shows the URL 'https://github.com/dtkaplan/checkr/blob/master/README.md'. The repository page includes a header with the repository name 'dtkaplan / checkr' and statistics: 1 Watch, 1 Star, and 0 Forks. Below this is a navigation bar with tabs for 'Code', 'Issues 3', 'Pull requests 0', 'Projects 0', 'Wiki', and 'Insights'. The main content area shows the 'README.md' file for the 'master' branch. It includes a commit by 'dtkaplan' with the message 'Post conference commit' and hash '884b0f1' from February 7. The file statistics show '19 lines (11 sloc)' and '1.1 KB'. At the bottom, there is a 'build unknown' status and the title 'checkr: Checking student answers in an R-tutorial system'.

checkr/README.md at master

GitHub, Inc. [US] | <https://github.com/dtkaplan/checkr/blob/master/README.md>

dtkaplan / checkr

Watch 1 Star 1 Fork 0

Code Issues 3 Pull requests 0 Projects 0 Wiki Insights

Branch: master checkr / README.md Find file Copy path

dtkaplan Post conference commit 884b0f1 on Feb 7

1 contributor

19 lines (11 sloc) | 1.1 KB Raw Blame History

build unknown

checkr: Checking student answers in an R-tutorial system

Challenges

1. Code checking
2. Recording Events

learnr



Recording Events

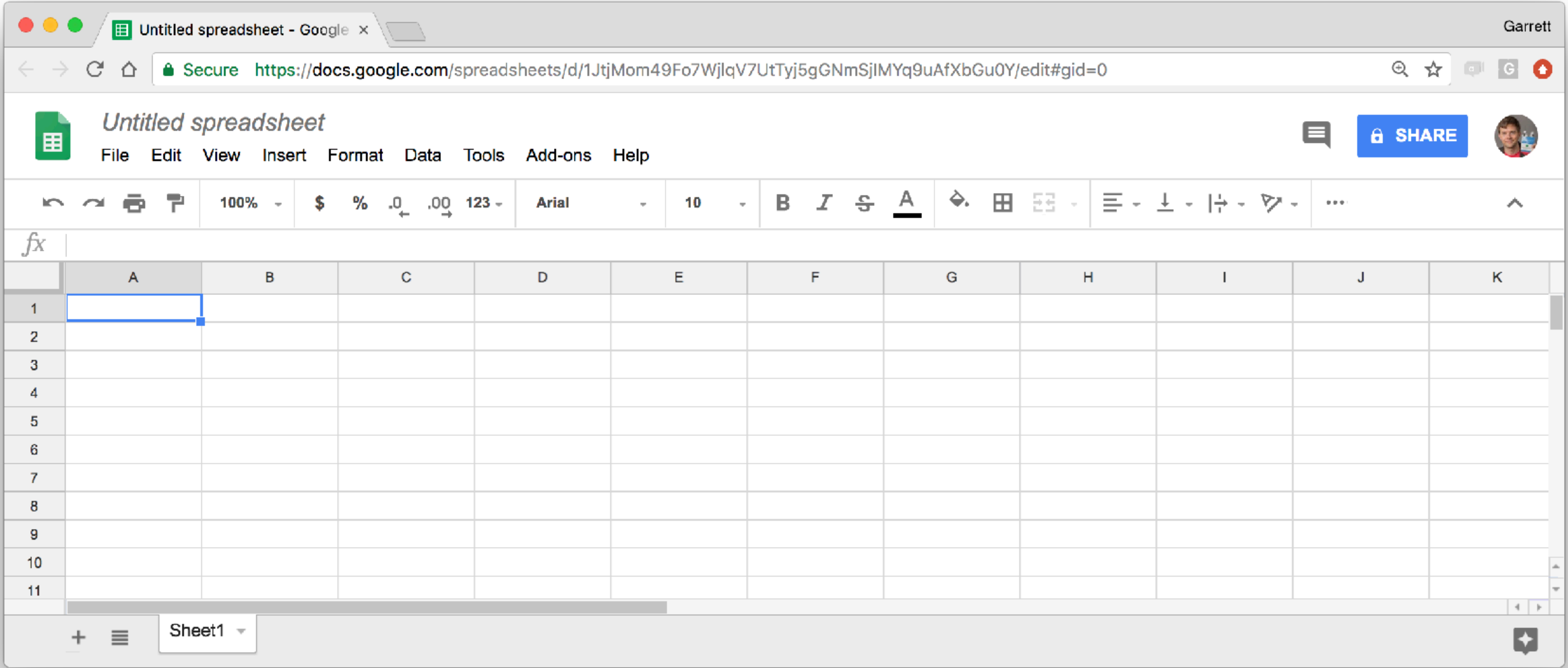
As part of deploying a tutorial you might want to record the various user events which occur within tutorials (e.g. exercise and question submissions, requests for hints/solutions, etc.).

simple checking

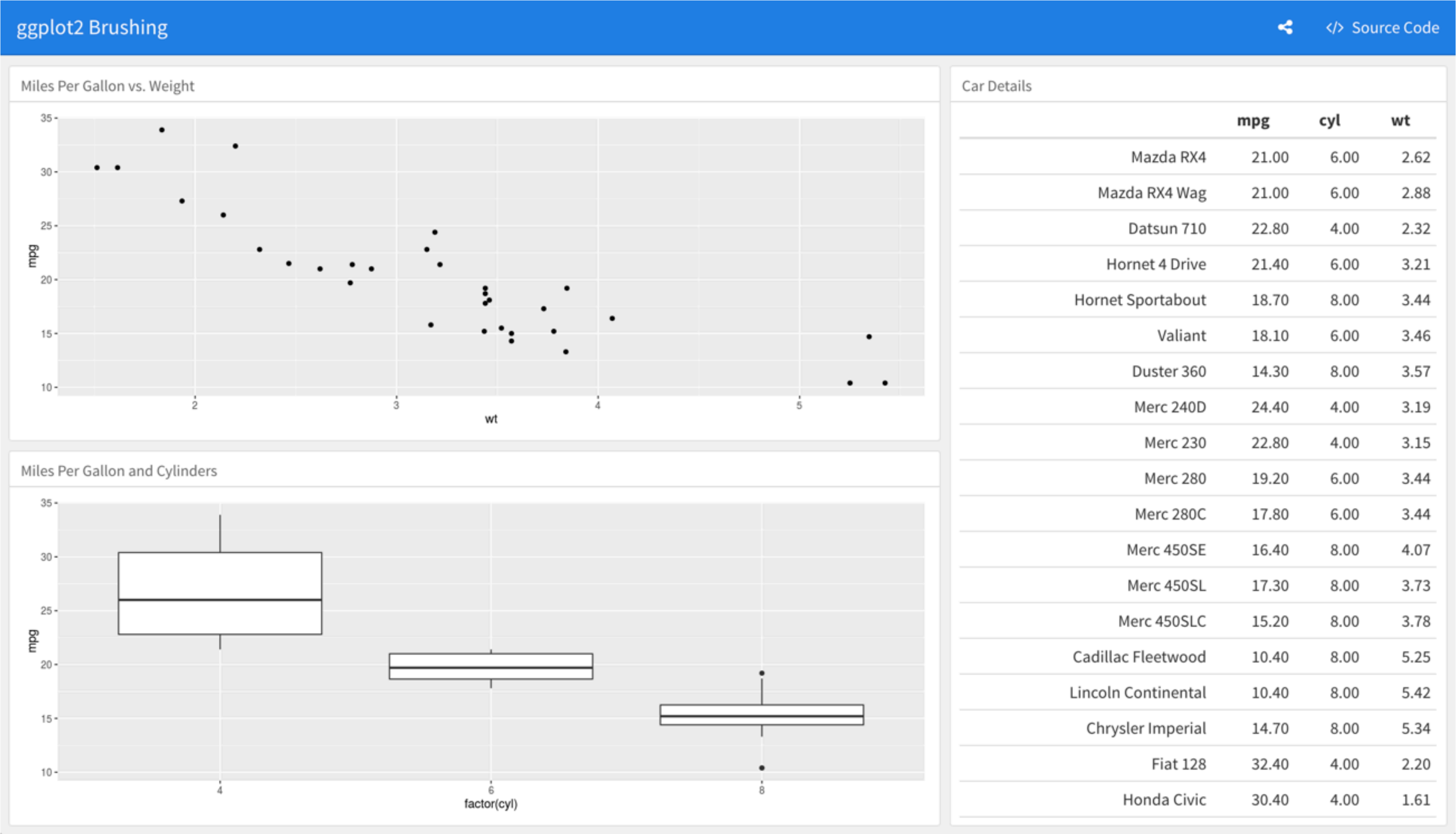
```
> user <- quote(map(lists, mean))  
> solution <- quote(map(lists, mean, na.rm = TRUE))  
> strict_check(user, solution)
```

I expected your code to include `na.rm = TRUE`. You may have referred to it in a different way, or left out an important argument name. Please try again.

Record to Googlesheets



Display events in dashboard



Run all solution chunks

```
```{r boxplot-solution}  
boxplot(Petal.Length ~ Species, data = iris)
```
```

github.com/rstudio-education/grader

The screenshot shows the GitHub repository page for `rstudio-education/grader`. The browser tabs include "Interactive Tutorials for R", "RStudio Cloud", and the current tab "rstudio-education/grader: Tool". The address bar shows the URL `https://github.com/rstudio-education/grader`. The repository page header includes the GitHub logo, "This repository", a search bar, and navigation links for "Pull requests", "Issues", "Marketplace", and "Explore". The repository name "rstudio-education / grader" is displayed, along with buttons for "Unwatch" (3), "Star" (0), and "Fork" (0). Below the header, tabs for "Code", "Issues" (0), "Pull requests" (0), "Projects" (0), "Wiki", "Insights", and "Settings" are visible. The main content area shows the repository description "Tools for teachers to use with learnr" and an "Add topics" link. A summary bar indicates "2 commits", "1 branch", "0 releases", "1 contributor", and the "MIT" license. Below this, a bar shows the current branch "master" and a "New pull request" button, along with buttons for "Create new file", "Upload files", "Find file", and a green "Clone or download" button. The commit history section shows a single commit by "garrettgman" titled "initial commit" with the hash "aa243bc" and a timestamp of "5 hours ago". A table lists the files in the repository, all of which were added in the initial commit 5 hours ago:

| File | Commit | Time |
|---------------|----------------|-------------|
| R | initial commit | 5 hours ago |
| tests | initial commit | 5 hours ago |
| .Rbuildignore | initial commit | 5 hours ago |
| .gitignore | initial commit | 5 hours ago |
| DESCRIPTION | initial commit | 5 hours ago |
| LICENSE | Initial commit | 5 hours ago |
| NAMESPACE | initial commit | 5 hours ago |
| README.md | Initial commit | 5 hours ago |

Below the file list, the "README.md" file is selected, showing the repository name "grader" in a large font.

Thank you

rstudio.github.io/learnr/

rmarkdown.rstudio.com

rstudio.cloud

github.com/rstudio-education/grader

github.com/rstudio-education/primers