Tidyverse Tools in R for Data Science and Statistical Inference

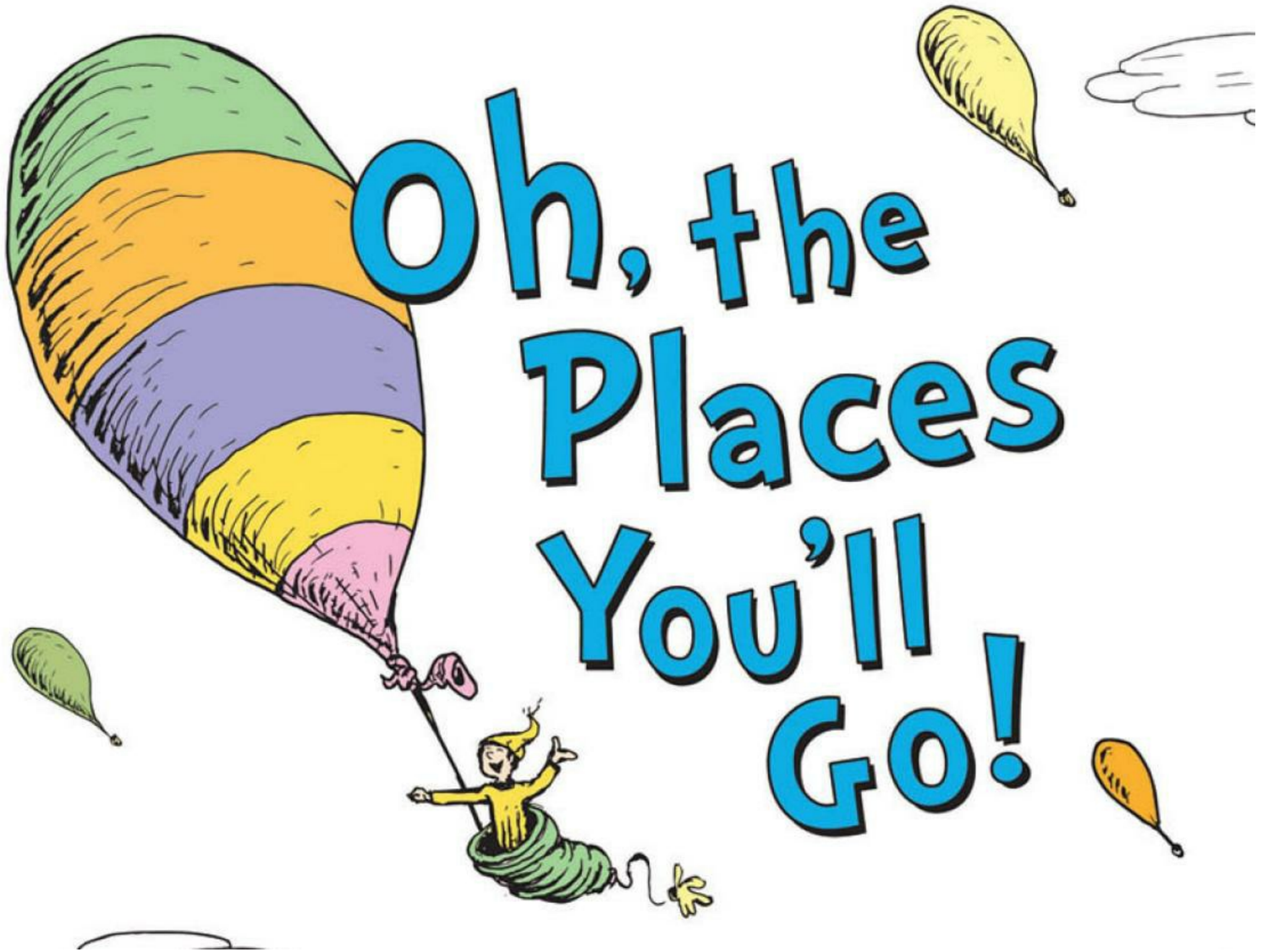
Dr. Chester Ismay
Senior Curriculum Lead at DataCamp

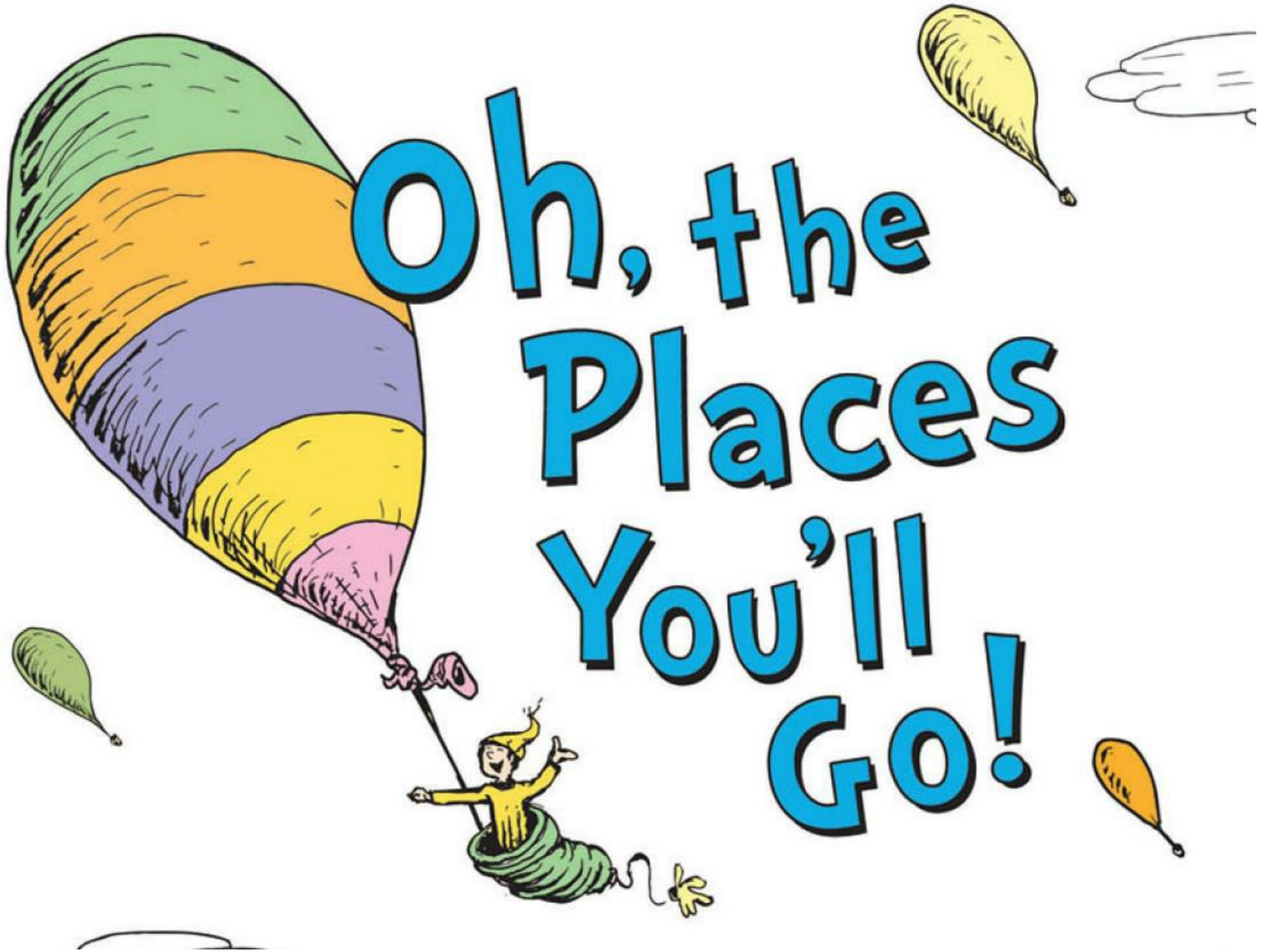
ismayc
@old_man_chester
chester@datacamp.com

2018-06-20
Journal Club June - OCHIN

Slides available at http://bit.ly/ochin-ismay
PDF slides at http://bit.ly/ochin-ismay-pdf

DataCamp


Oh, the Places You'll Go!



with the tidyverse!!!

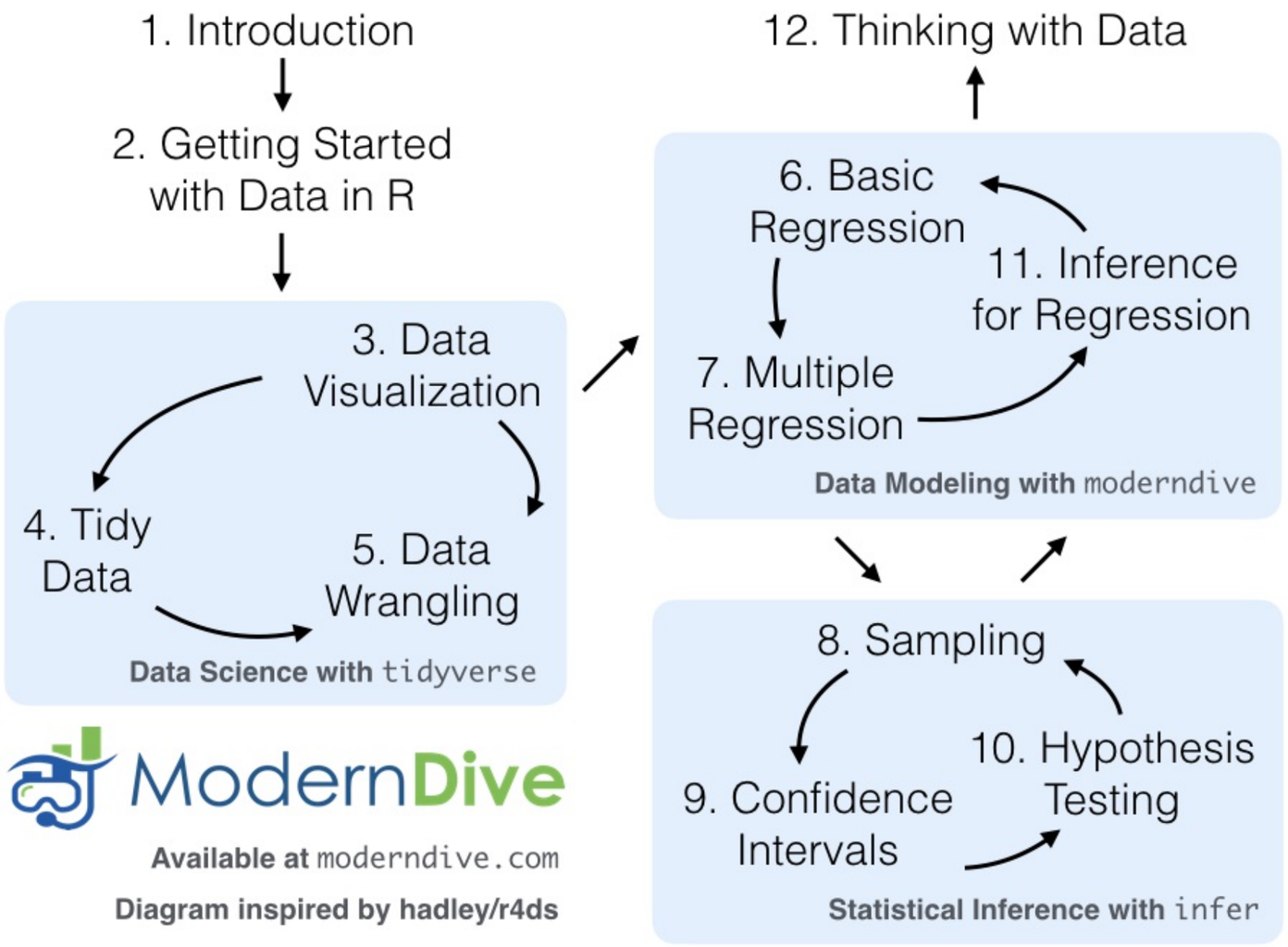
Freely available information

ModernDive

A Modern Dive into Data with R

- Webpage: http://moderndive.com
- Developmental version at https://moderndive.netlify.com
- GitHub Repo
- Please signup for our mailing list!

Slides available at http://bit.ly/ochin-ismay


1. Introduction
2. Getting Started with Data in R
3. Data Visualization
4. Tidy Data
5. Data Wrangling
Data Science with tidyverse
6. Basic Regression
7. Multiple Regression
11. Inference for Regression
Data Modeling with moderndive
8. Sampling
9. Confidence Intervals
10. Hypothesis Testing
Statistical Inference with infer
12. Thinking with Data
ModernDive
Available at moderndive.com
Diagram inspired by hadley/r4ds

Designed for the novice / Nice for the practioner

For much of this book, we will assume that you are using R via RStudio. First time users often confuse the two. At its simplest:

- R is like a car's engine
- RStudio is like a car's dashboard

R: Engine	RStudio: Dashboard
	

Slides available at http://bit.ly/ochin-ismay

Designed for the novice / Nice for the practioner

A good analogy for R packages is they are like apps you can download onto a mobile phone:

R: A new phone	R Packages: Apps you can download
	 

Slides available at http://bit.ly/ochin-ismay

R Data Types

Vector/variable

- Type of vector (`int`, `num` or `dbl`, `chr`, `lgl`, `date`)

R Data Types

Vector/variable

- Type of vector (`int`, `num` or `dbl`, `chr`, `lgl`, `date`)

Data frame

- Vectors of (potentially) different types
- Each vector has the same number of rows

The bare minimum needed for understanding today

```
library(tibble)
library(lubridate)
ex1 <- data_frame(
    vec1 = c(1980, 1990, 2000, 2010),
    vec2 = c(1L, 2L, 3L, 4L),
    vec3 = c("low", "low", "high", "high"),
    vec4 = c(TRUE, FALSE, FALSE, FALSE),
    vec5 = ymd(c("2017-05-23", "1776/07/04",
                 "1983-05/31", "1908/04-01"))
  )
ex1
```

Slides available at http://bit.ly/ochin-ismay

The bare minimum needed for understanding today

```
library(tibble)
library(lubridate)
ex1 <- data_frame(
    vec1 = c(1980, 1990, 2000, 2010),
    vec2 = c(1L, 2L, 3L, 4L),
    vec3 = c("low", "low", "high", "high"),
    vec4 = c(TRUE, FALSE, FALSE, FALSE),
    vec5 = ymd(c("2017-05-23", "1776/07/04",
                 "1983-05/31", "1908/04-01"))
  )
ex1
```

```
# A tibble: 4 x 5
   vec1  vec2 vec3  vec4  vec5
  <dbl> <int> <chr> <lgl> <date>
1  1980     1 low   TRUE  2017-05-23
2  1990     2 low   FALSE 1776-07-04
3  2000     3 high  FALSE 1983-05-31
4  2010     4 high  FALSE 1908-04-01
```

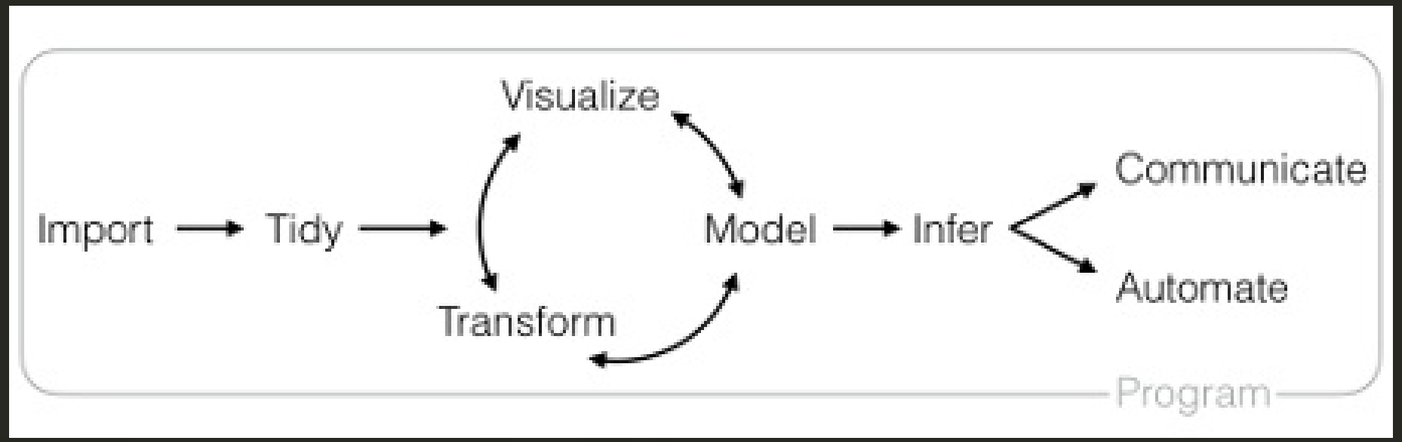
Slides available at http://bit.ly/ochin-ismay

Welcome to the tidyverse!

The `tidyverse` is a collection of R packages that share common philosophies and are designed to work together.

Slides available at http://bit.ly/ochin-ismay

The workflow model



Modified from image by Hadley Wickham

Slides available at http://bit.ly/ochin-ismay

Table of Contents

Motivating example for today

- Great British Bakeoff data collected by Alison Hill

Slides available at http://bit.ly/ochin-ismay

Return to Table of Contents

Data Importing

Read from CSV file

From the internet

```
bakeoff <- read_csv("http://bit.ly/bakeoff-csv")
```

Locally from your computer

```
library(readr)
bakeoff <- read_csv("data/bakeoff.csv")
```

```
rmarkdown::paged_table(bakeoff)
```

series <int>	episode <int>	baker <chr>
1	1	Annetha
1	1	David
1	1	Edd
1	1	Jasminder
1	1	Jonathan
1	1	Louise
1	1	Miranda
1	1	Ruth
1	1	Lea
1	1	Mark
1	2	David
1	2	Edd
1	2	Jasminder
1	2	Jonathan
1	2	Miranda
1	2	Ruth

Other packages for reading in data

- Read in SPSS, SAS, and STATA data files

Other packages for reading in data

- Read in SPSS, SAS, and STATA data files

- Read in Excel data files

Data Transforming/Wrangling

Get summary information quickly and readably

```
bakeoff %>%
  filter(!is.na(us_season)) %>%
  count(series, episode)
```

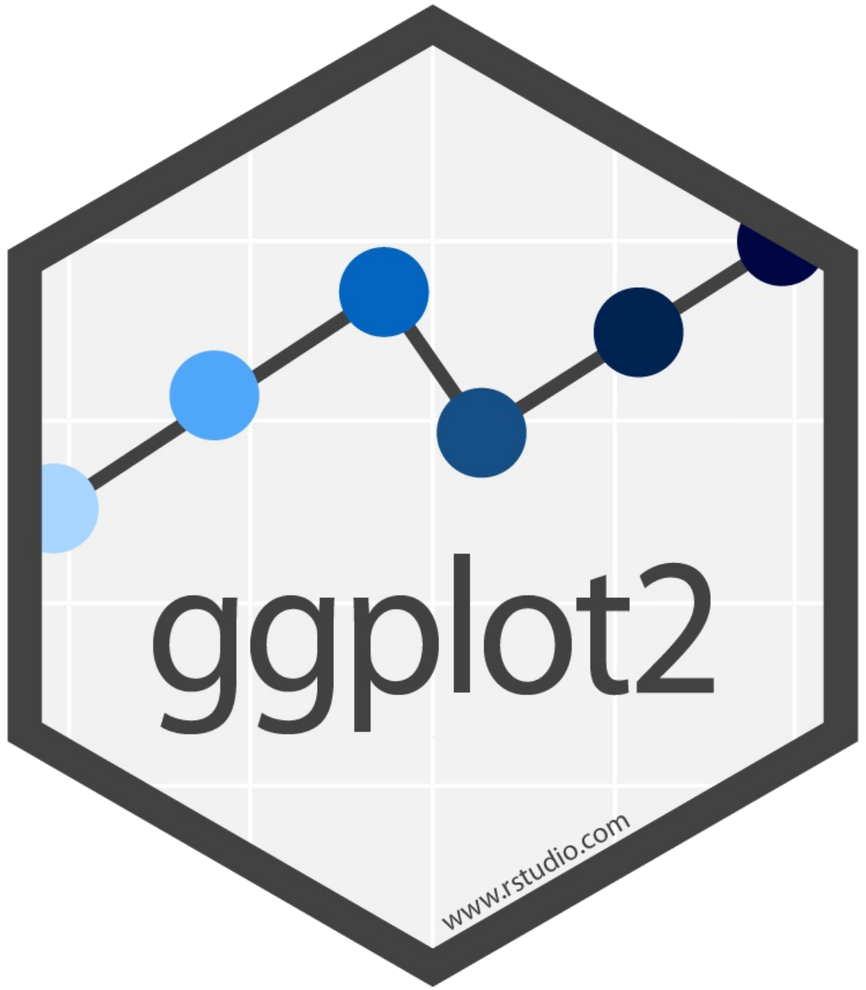
```
# A tibble: 40 x 3
   series episode     n
    <int>   <int> <int>
 1      4       1    13
 2      4       2    12
 3      4       3    11
 4      4       4     9
 5      4       5     8
 6      4       6     7
 7      4       7     6
 8      4       8     5
 9      4       9     4
10      4      10     3
# ... with 30 more rows
```

Get summary information quickly and readably

```
bakers_season4_episodes <- bakeoff %>%
  filter(!is.na(us_season), series == 4) %>%
  group_by(episode) %>%
  summarize(count = n())
bakers_season4_episodes
```

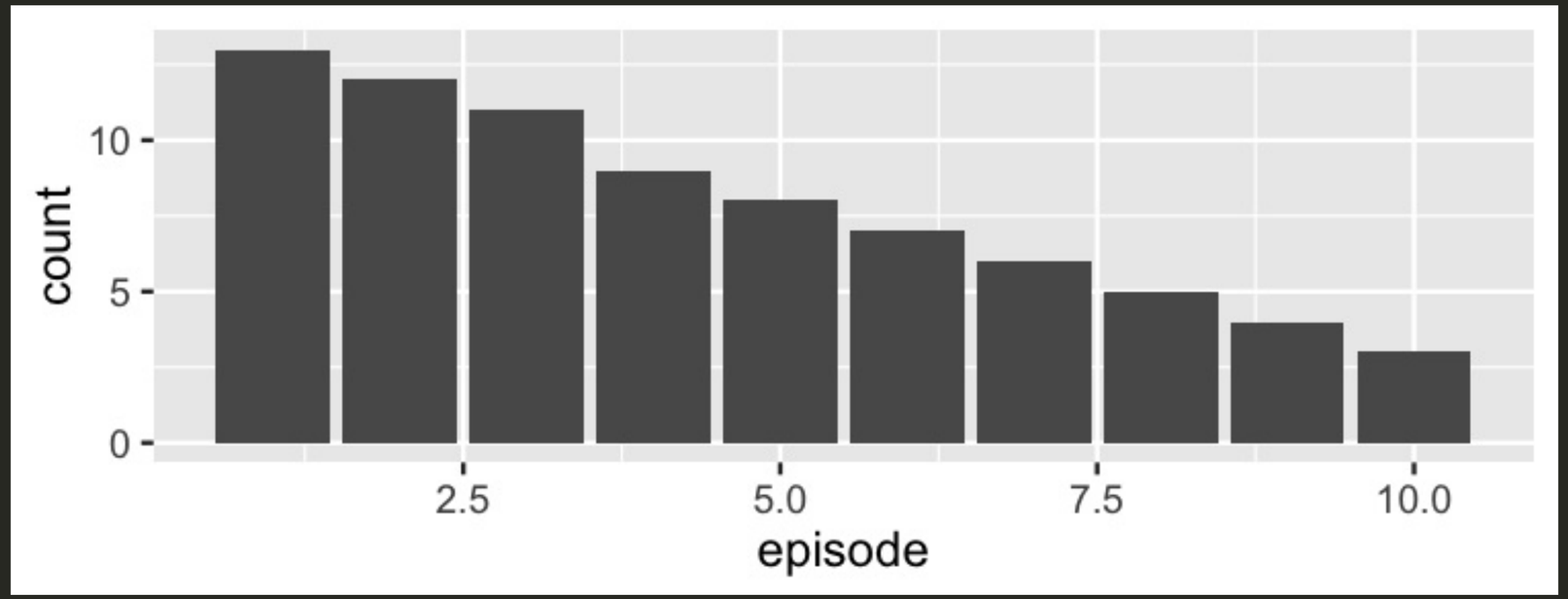
```
# A tibble: 10 x 2
   episode count
     <int> <int>
 1       1    13
 2       2    12
 3       3    11
 4       4     9
 5       5     8
 6       6     7
 7       7     6
 8       8     5
 9       9     4
10      10     3
```

Data Visualization



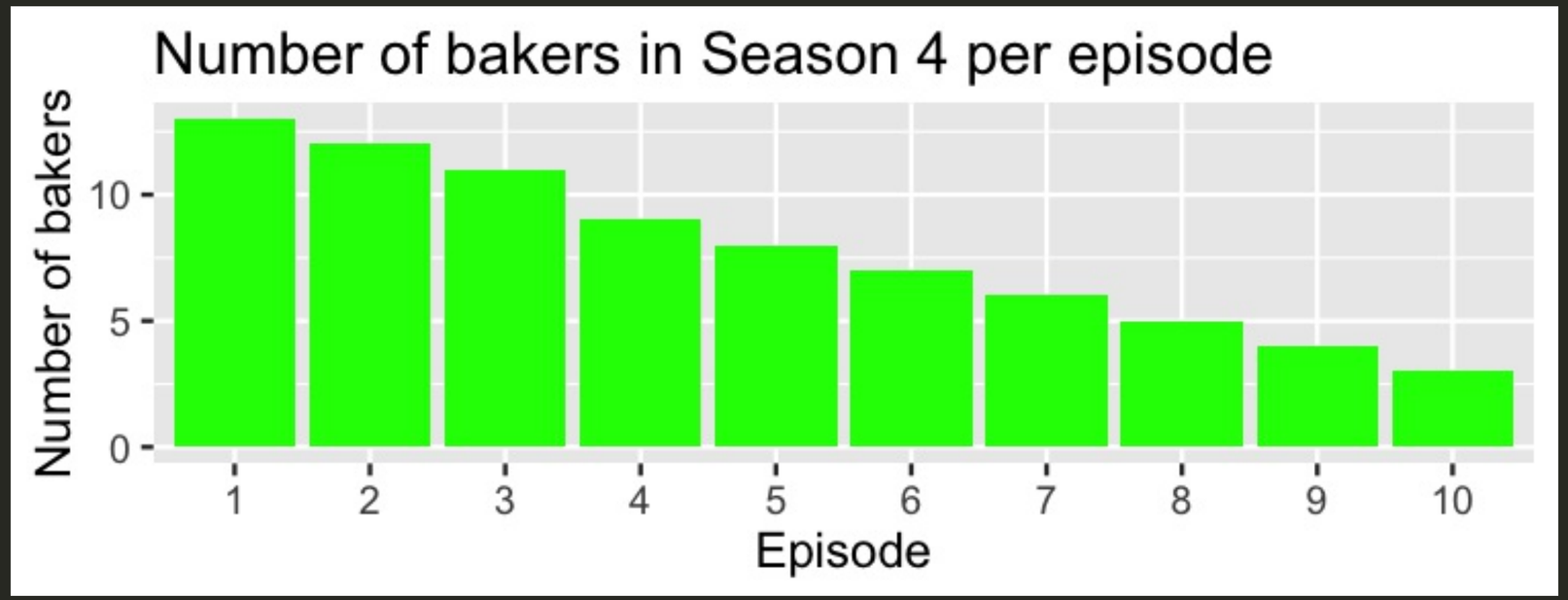
Visualize our last result

```
ggplot(data = bakers_season4_episodes,
       mapping = aes(x = episode, y = count)) +
         geom_col()
```

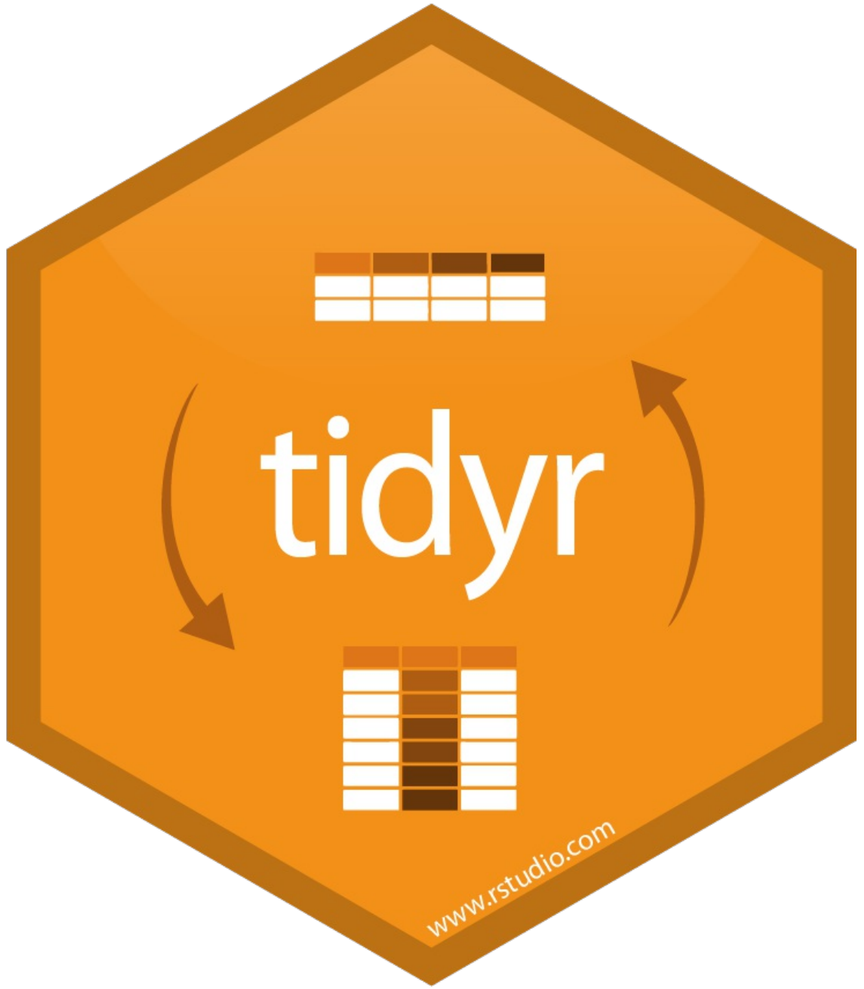


Tweak it

```
ggplot(data = bakers_season4_episodes,
       mapping = aes(x = factor(episode), y = count)) +
  geom_col(fill = "green") +
  labs(x = "Episode", y = "Number of bakers",
       title = "Number of bakers in Season 4 per episode")
```



Data Tidying



"Messy" data

```
ratings <- read_csv("http://bit.ly/ratings-csv",
                    col_types = cols(series = col_factor(levels = NULL)))
ratings
```

```
# A tibble: 8 x 11
  series    e1    e2    e3    e4    e5    e6    e7    e8    e9
  <fct>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 1       2.24  3     3     2.6   3.03  2.75 NA    NA    NA
2 2       3.1   3.53  3.82  3.6   3.83  4.25  4.42  5.06 NA
3 3       3.85  4.6   4.53  4.71  4.61  4.82  5.1   5.35  5.7
4 4       6.6   6.65  7.17  6.82  6.95  7.32  7.76  7.41  7.41
5 5       8.51  8.79  9.28 10.2   9.95 10.1  10.3   9.02 10.7
6 6      11.6  11.6  12.0  12.4  12.4  12    12.4  11.1  12.6
7 7      13.6  13.4  13.0  13.3  13.1  13.1  13.4  13.3  13.4
8 8       9.46  9.23  8.68  8.55  8.61  8.61  9.01  8.95  9.03
# ... with 1 more variable: e10 <dbl>
```

- What if we'd like to plot average 7 day viewership for each episode?

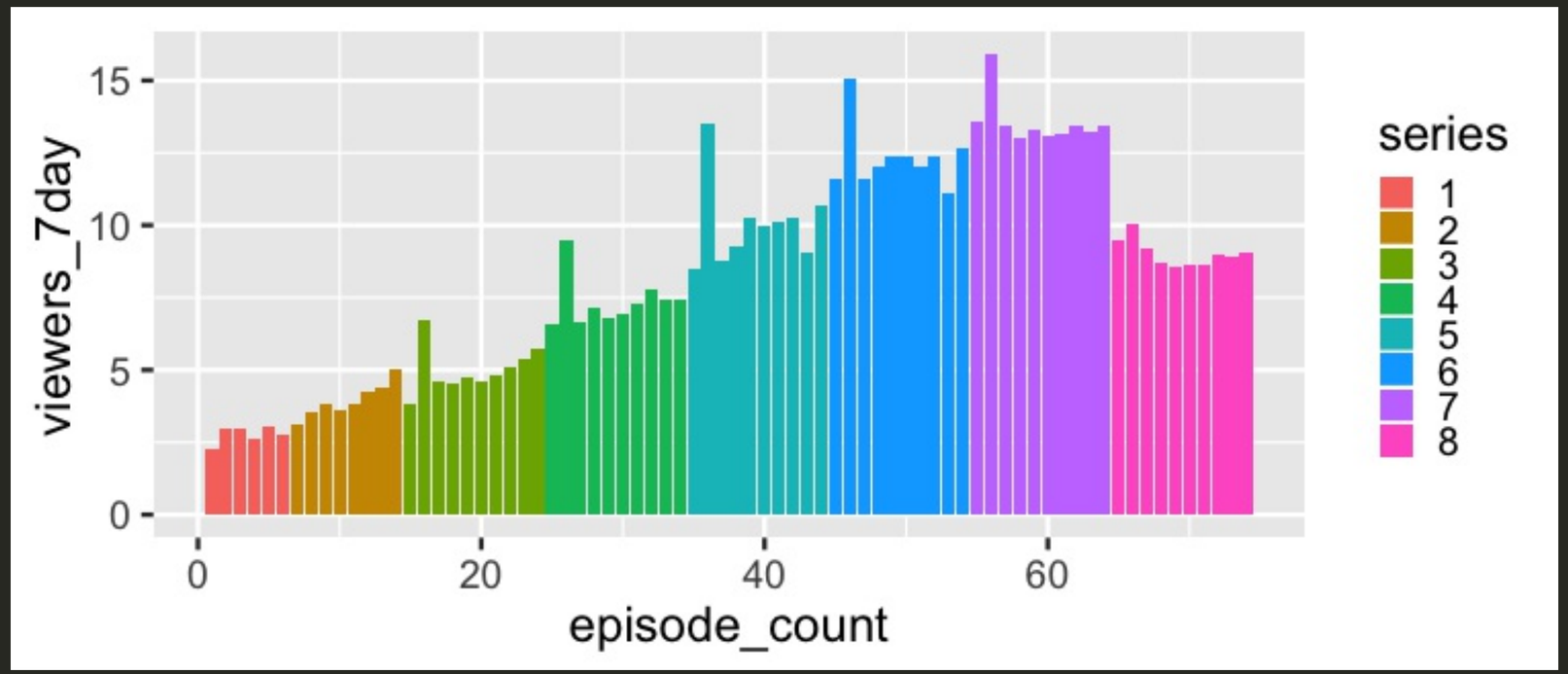
Some tidying and cleaning

```
# Gather and count episodes
tidy_ratings <- ratings %>%
    gather(key = "episode", value = "viewers_7day", -series,
           na.rm = TRUE) %>%
    arrange(series, episode) %>%
    mutate(episode_count = row_number())
tidy_ratings
```

```
# A tibble: 74 x 4
   series episode viewers_7day episode_count
   <fct>  <chr>          <dbl>         <int>
 1 1      e1              2.24             1
 2 1      e2              3                2
 3 1      e3              3                3
 4 1      e4              2.6              4
 5 1      e5              3.03             5
 6 1      e6              2.75             6
 7 2      e1              3.1              7
 8 2      e2              3.53             8
 9 2      e3              3.82             9
10 2      e4              3.6             10
# ... with 64 more rows
```

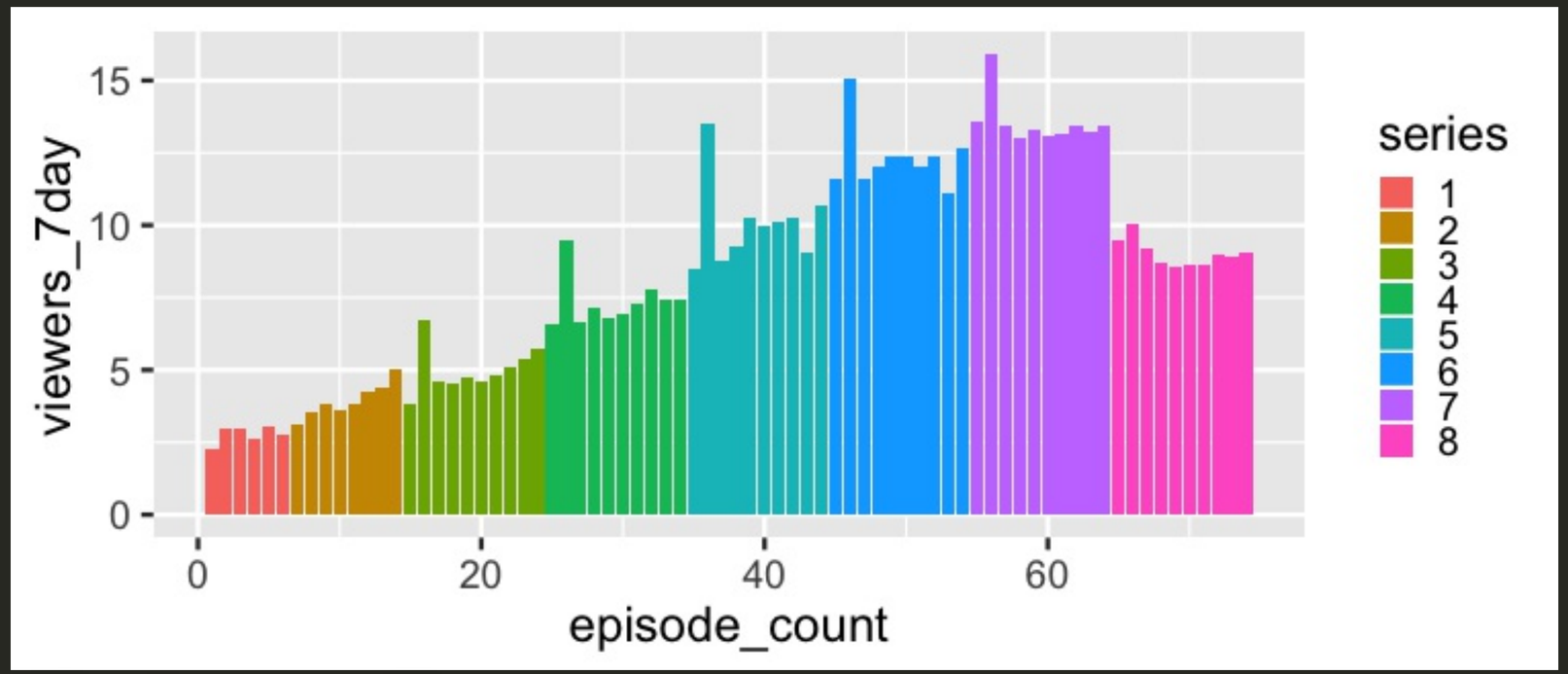
Plot viewers by episode and series

```
ggplot(tidy_ratings, aes(x = episode_count, y = viewers_7day,
                         fill = series)) + geom_col()
```



Plot viewers by episode and series

```
ggplot(tidy_ratings, aes(x = episode_count, y = viewers_7day,
                         fill = series)) + geom_col()
```



What happened?

Data Modeling

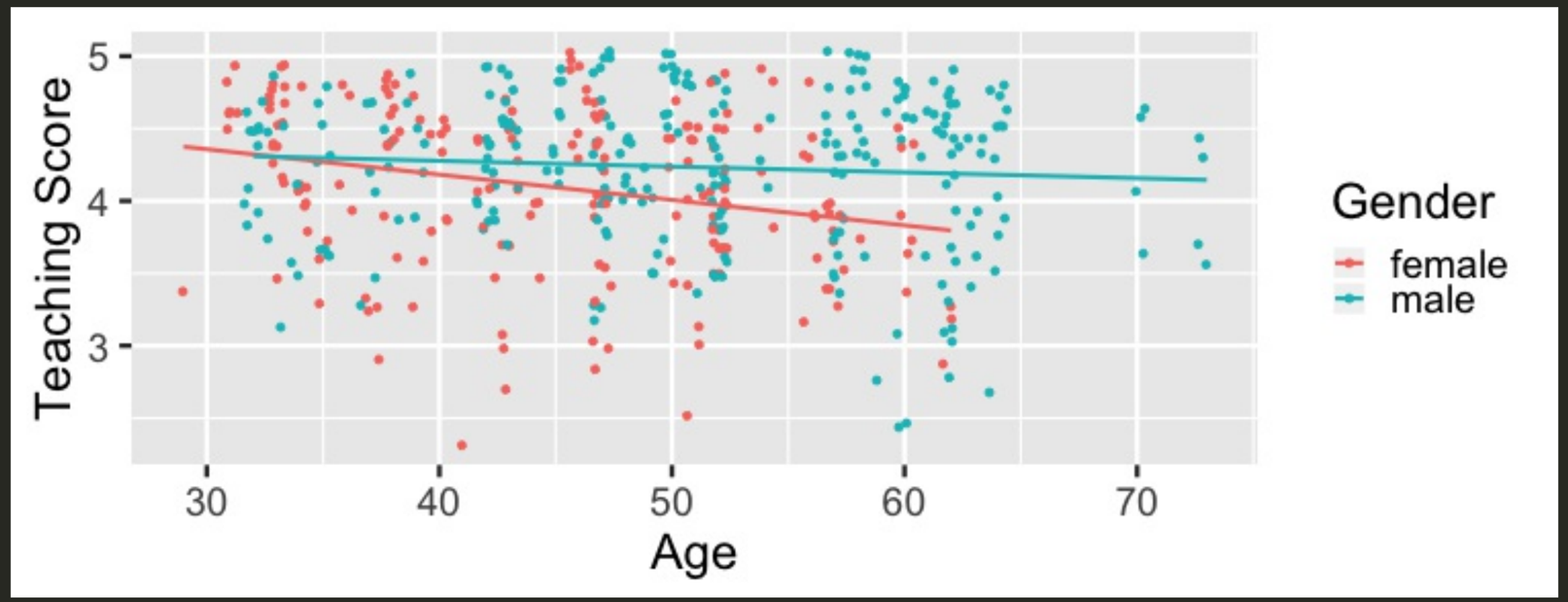
Get the data - More information on ModernDive

```
download.file("http://bit.ly/evals-rds", destfile = "data/evals.rds")
( evals <- read_rds("data/evals.rds") )
```

```
# A tibble: 463 x 4
   score bty_avg   age gender
   <dbl>   <dbl> <int> <fct>
 1   4.7    5       36 female
 2   4.1    5       36 female
 3   3.9    5       36 female
 4   4.8    5       36 female
 5   4.6    3       59 male
 6   4.3    3       59 male
 7   2.8    3       59 male
 8   4.1    3.33    51 male
 9   3.4    3.33    51 male
10   4.5    3.17    40 female
# ... with 453 more rows
```

Visualize the fits

```
ggplot(evals, aes(x = age, y = score, col = gender)) +
  geom_jitter() +
  labs(x = "Age", y = "Teaching Score", color = "Gender") +
  geom_smooth(method = "lm", se = FALSE)
```



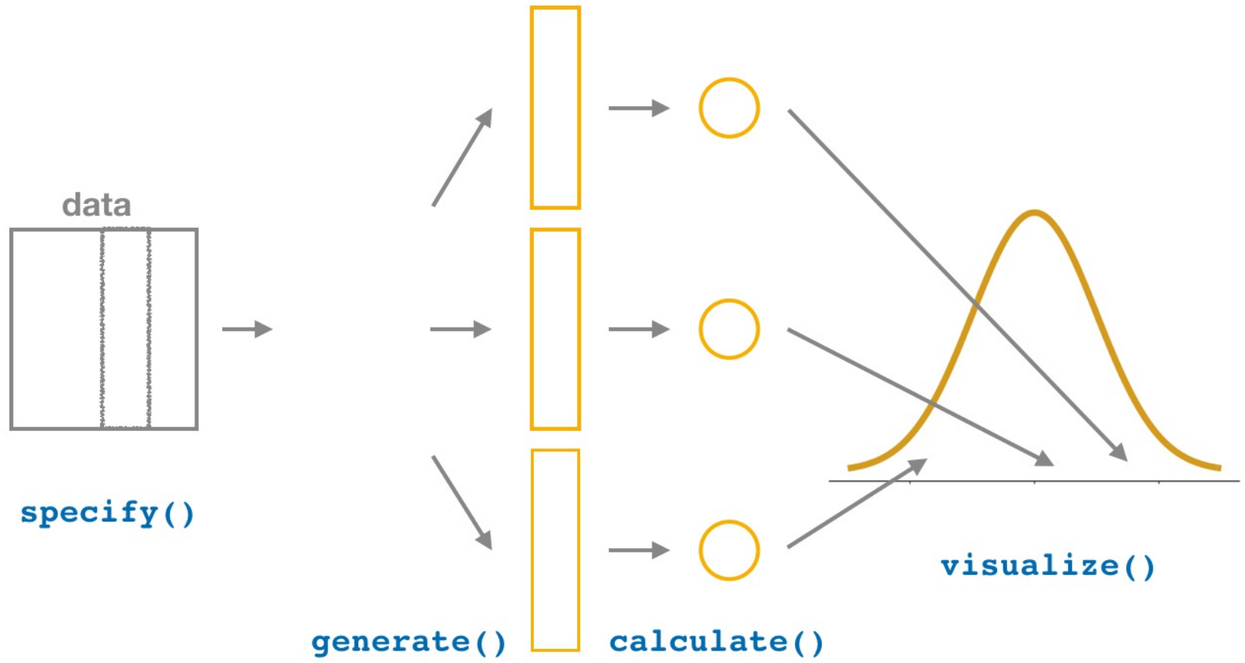
Analyze the model

```
library(moderndive)
score_model_interaction <- lm(score ~ age * gender, data = evals)
get_regression_table(score_model_interaction)
```

```
# A tibble: 4 x 7
  term   estimate std_error statistic p_value conf_low conf_high
  <chr>     <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
1 inter…    4.88      0.205     23.8    0        4.48      5.29
2 age      -0.018     0.004     -3.92   0       -0.026    -0.009
3 gende…   -0.446     0.265     -1.68   0.094   -0.968     0.076
4 age:g…    0.014     0.006      2.45   0.015    0.003     0.024
```

Statistical Inference

Infer pipeline



Mythbusters yawning example

```
library(infer)
mythbusters_yawn <- read_rds(url("http://bit.ly/mythbusters_yawn-rds"))
mythbusters_yawn %>%
  specify(formula = yawn ~ group, success = "yes") %>%
  generate(reps = 1000) %>%
  calculate(stat = "diff in props", order = c("seed", "control")) %>%
  get_ci()
```

```
# A tibble: 1 x 2
  `2.5%` `97.5%`
   <dbl>   <dbl>
1 -0.201   0.301
```

Communicating with Data / Automating the Process

The magic

- These slides were made with R Markdown
- If I wasn't using R Markdown, what would I do if there was a mistake in `bakeoff.csv` and I needed to update a report based on it?

Resources

- ModernDive
- DataCamp
- R for Data Science
- Portland R User Group
- R-Ladies PDX
- RStudio cheatsheets
- #rstats on Twitter

To run the code in these slides

Make sure you have the (current) up-to-date R, RStudio, and R packages

- Beginner's Guide on ModernDive.com

- R (at least version 3.4.4 or higher)
- RStudio (at least version 1.1.442 or higher)
- Run this in the RStudio Console

```
pkgs <- c("tidyverse", "moderndive", "remotes")

install.packages(pkgs)
remotes::install_github("andrewpbray/infer", ref = "p_value")
```

Thanks for attending! Contact me: Email or Twitter

- Special thanks to
  - Albert Y. Kim
  - Andrew Bray
  - Alison Hill
- Slides created via the R package xaringan by Yihui Xie
- Slides' source code at