

Let them
eat cake
(first)!



@minebocek



mine-cetinkaya-rundel



cetinkaya.mine@gmail.com



bit.ly/let-eat-cake-cc





Imagine you're new to baking,
and you're in a baking class.
I'm going to present two
options for starting the class.
Which one gives you **better**
sense of the final product?

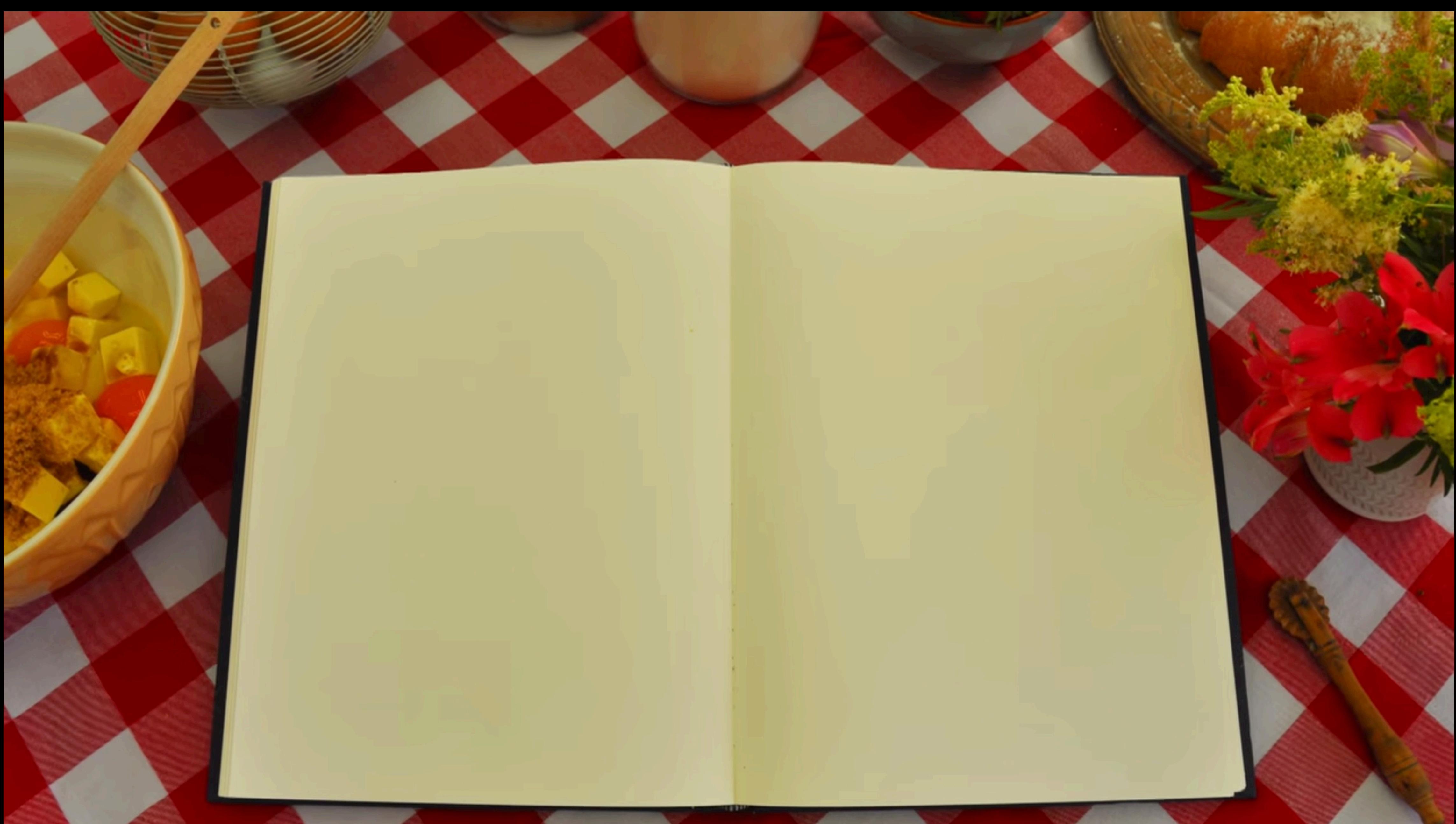
Pineapple and coconut sandwich cake

Pineapple and Coconut Sandwich Cake



Pineapple and Coconut Sandwich Cake





5 design principles

Backward design

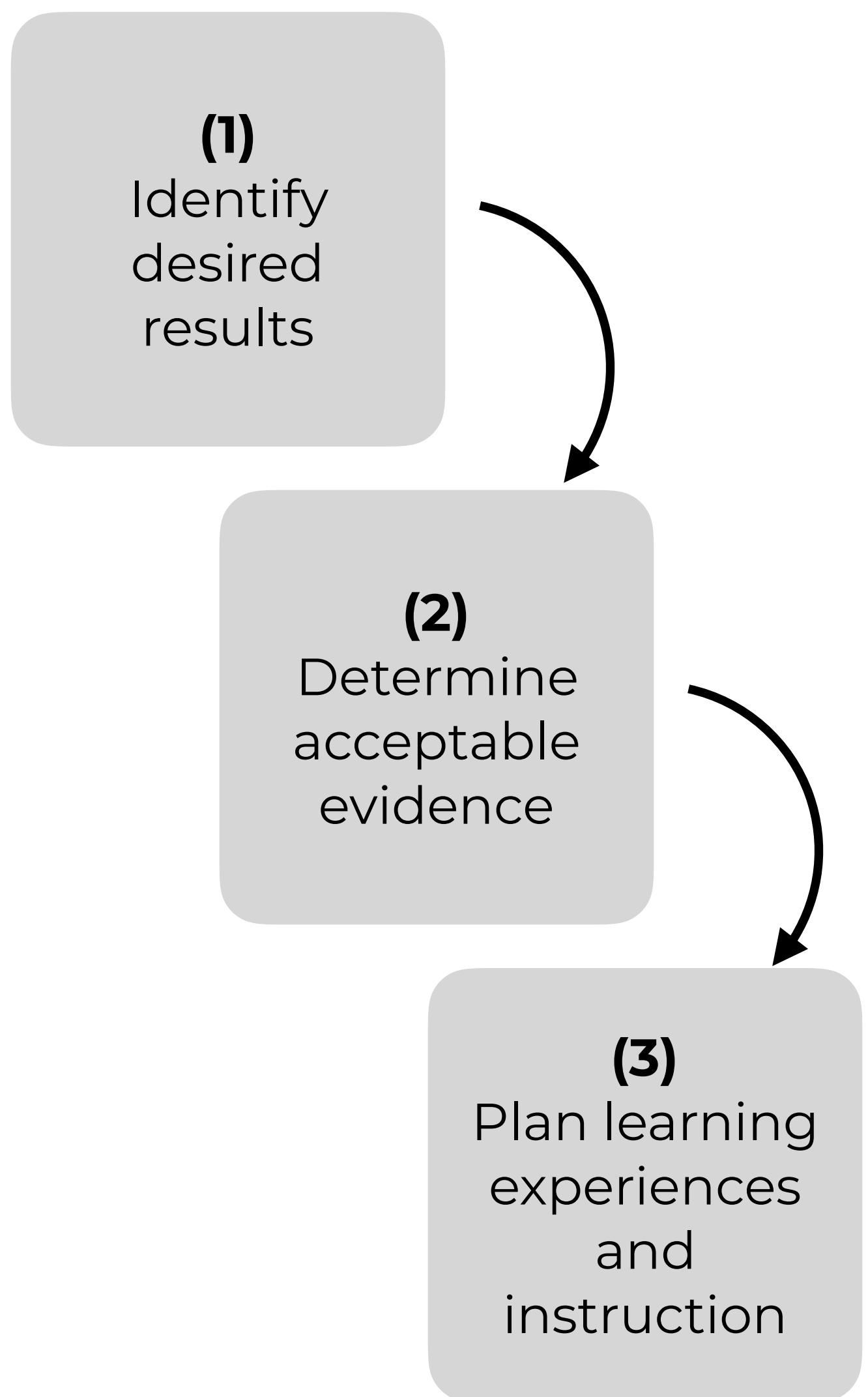
set goals for educational curriculum before choosing instructional methods + forms of assessment



analogous to travel planning - itinerary deliberately designed to meet cultural goals, not purposeless tour of all major sites in a foreign country



Wiggins, Grant P., Grant Wiggins, and Jay McTighe. *Understanding by design*. Ascd, 2005.

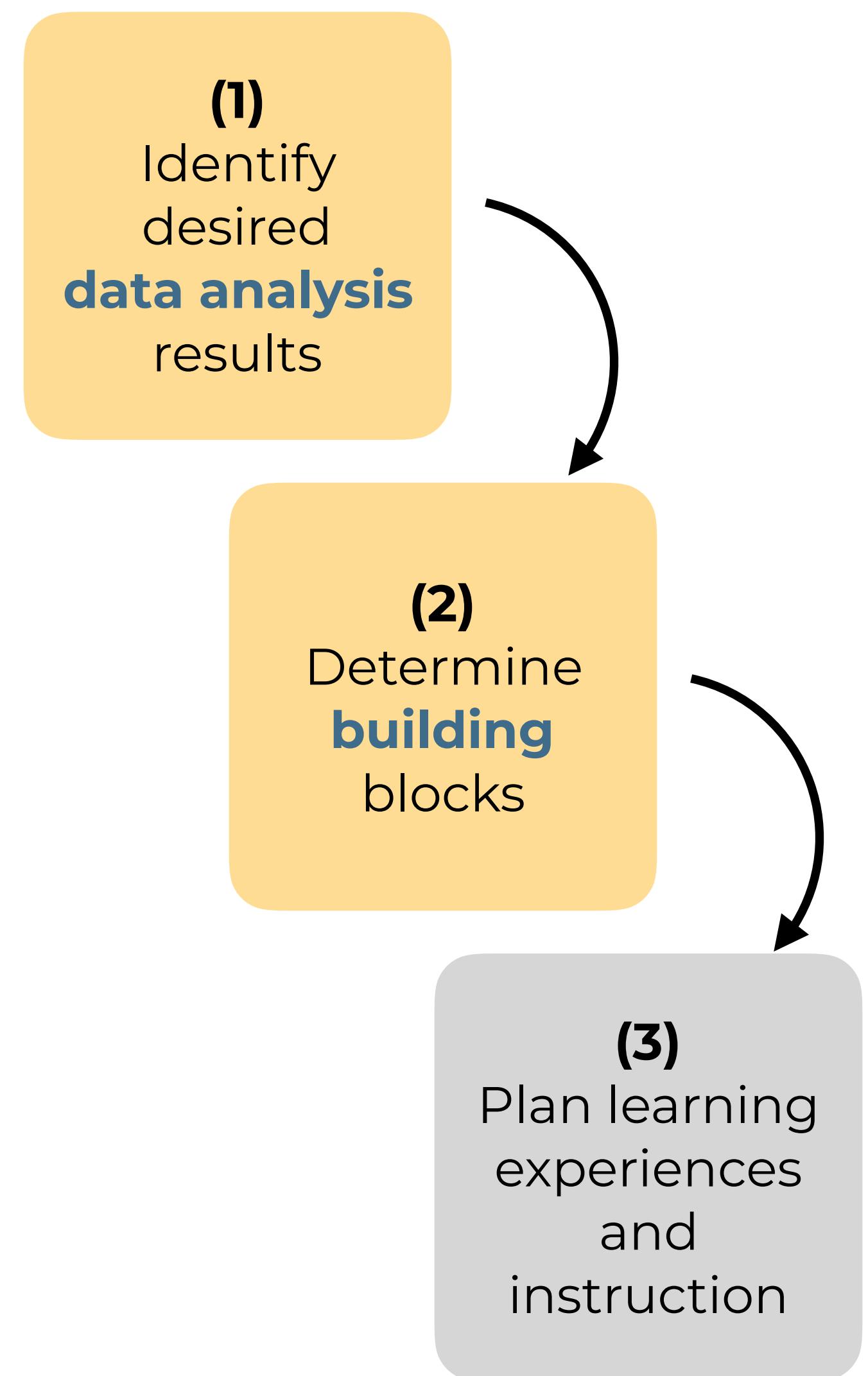


Designing backwards

students are first exposed to results and findings of a data analysis



and then learn the building blocks of the methods and techniques used along the way



Context

assumes
no
background



focuses on
EDA +
modeling &
inference +
modern
computing



uses R as the
statistical
programming
language



requires
reproducibility



emphasizes
collaboration +
effective
communi-
cation



GAISE 2016

1. Teach statistical thinking.

a. Teach statistics as an investigative process of problem-solving and decision-making.

Students should not leave their introductory statistics course with the mistaken impression that statistics consists of an unrelated collection of formulas and methods. Rather, students should understand that statistics is a problem-solving and decision-making *process* that is fundamental to scientific inquiry and essential for making sound decisions.

b. Give students experience with multivariable thinking.

We live in a complex world in which the answer to a question often depends on many factors. Students will encounter such situations within their own fields of study and everyday lives. We must prepare our students to answer challenging questions that require them to investigate and explore relationships among many variables. Doing so will help them to appreciate the value of statistical thinking and methods.

2. Focus on conceptual understanding.

3. Integrate real data with a context and a purpose.

4. Foster active learning.

5. Use technology to explore concepts and analyze data.

6. Use assessments to improve and evaluate student learning.

GAISE 2016, http://www.amstat.org/asa/files/pdfs/GAISE/GaiseCollege_Full.pdf.

① NOT a commonly used subset of tests and intervals and produce them with hand calculations

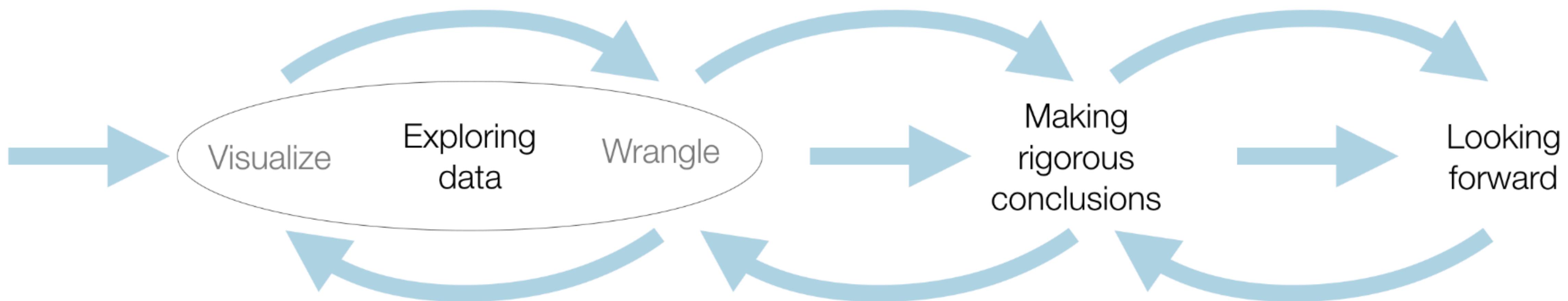
② Multivariate analysis requires the use of computing

③ NOT use technology that is only applicable in the intro course or that doesn't follow good science principles

④ Data analysis isn't just inference and modeling, it's also data importing, cleaning, preparation, exploration, and visualization

Intro to Data Science

Duke University & soon University of Edinburgh



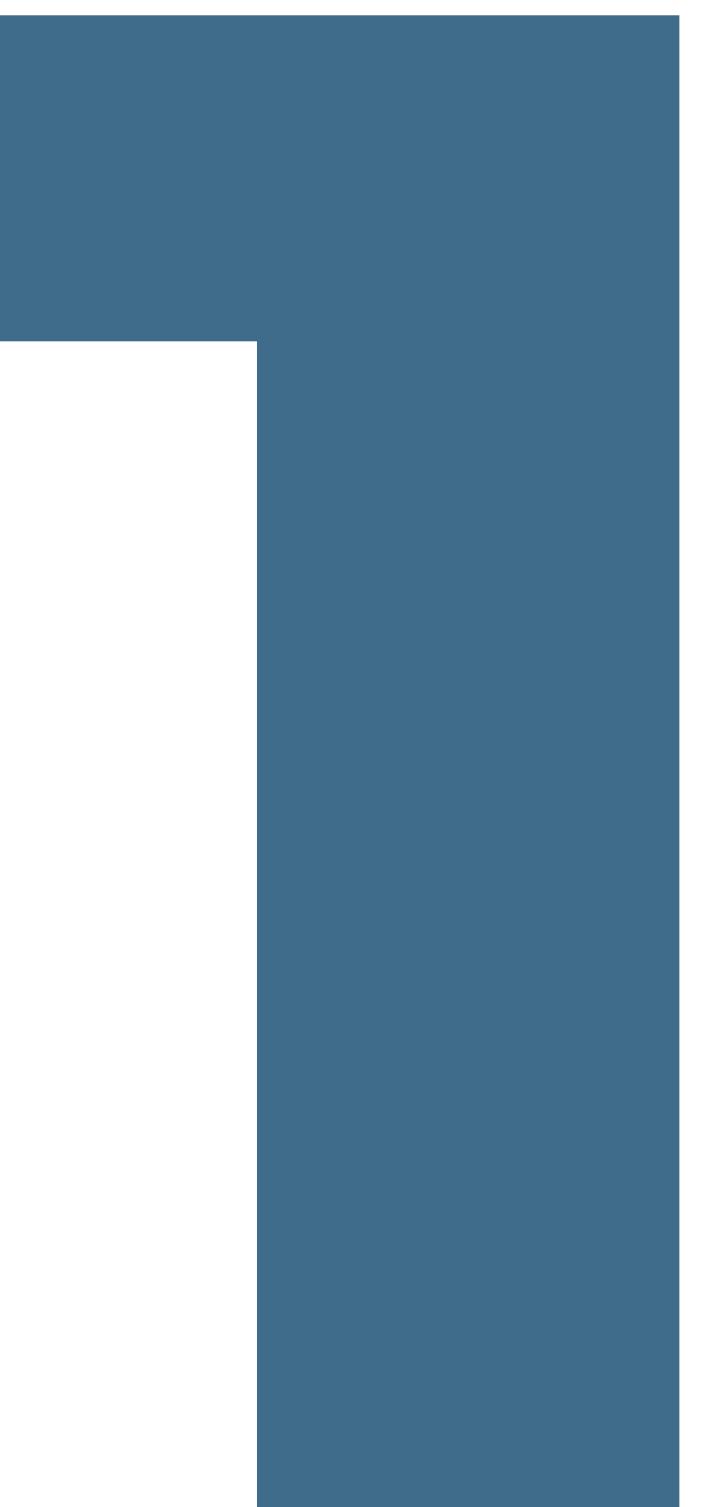
Fundamentals of
data & data viz,
confounding variables,
Simpson's paradox
+
R / RStudio,
R Markdown, simple git

Tidy data, data frames vs.
summary tables,
recoding and transforming,
web scraping and iteration
+
collaboration on GitHub

Building & selecting
models, visualizing
interactions, prediction &
validation, inference via
simulation

Data science ethics,
interactive viz & reporting,
text analysis,
Bayesian inference
+
communication,
dissemination

**start
with
cake**





Which of the following is more likely to be **motivating** for a wide range of students?

- ❑ Declare the following variables
- ❑ Then, determine the class of each variable

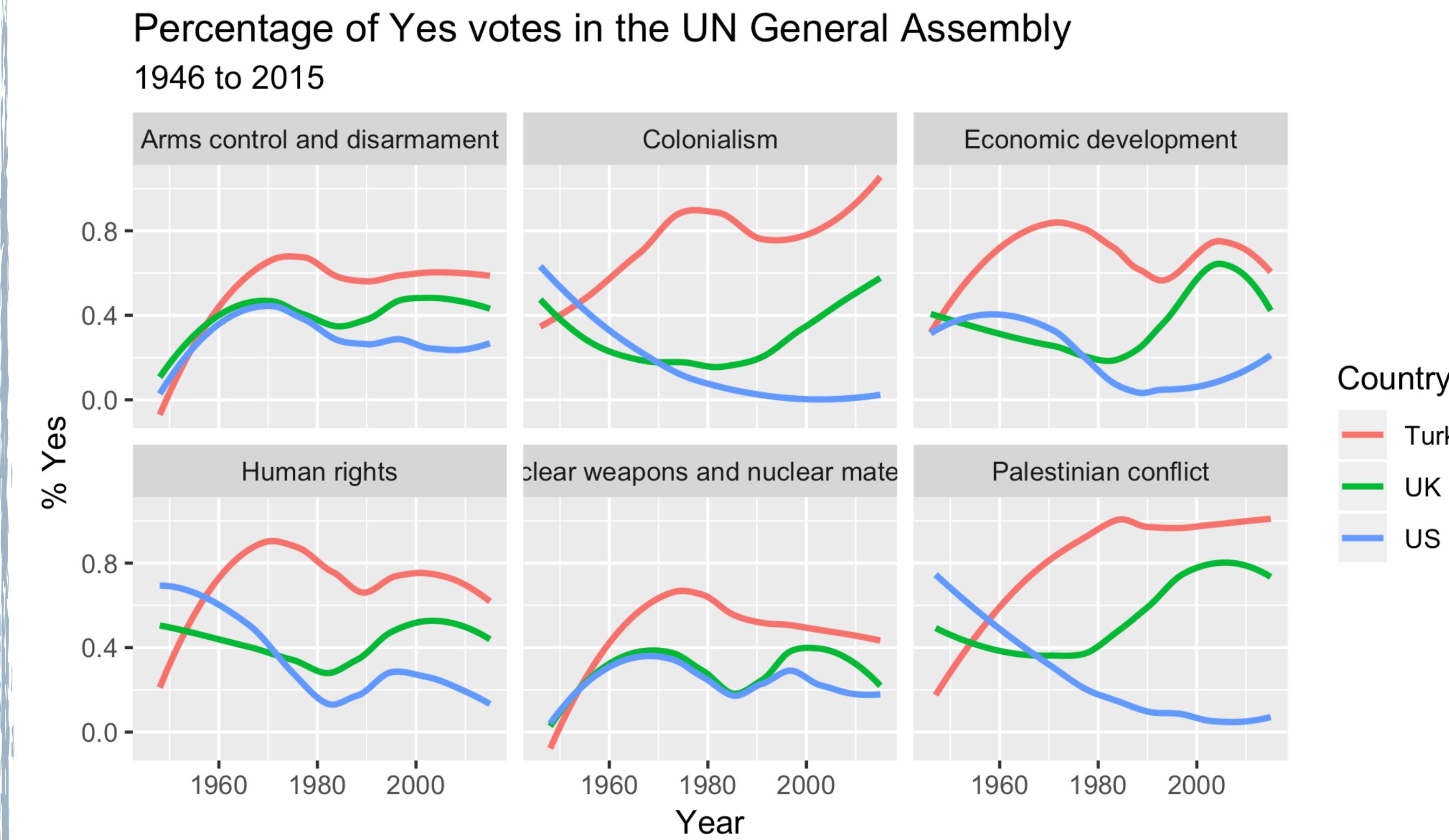
```
# Declare variables
x <- 8
y <- "monkey"
z <- FALSE

# Check class of x
class(x)
#> [1] "numeric"

# Check class of y
class(y)
#> [1] "character"

# Check class of z
class(z)
#> [1] "logical"
```

- ❑ Open today's demo project
- ❑ Knit the document and discuss the results with your neighbor



- ❑ Then, change Turkey to a different country, and plot again

with great examples,
comes a great amount of code...

but let's focus on the task at hand...

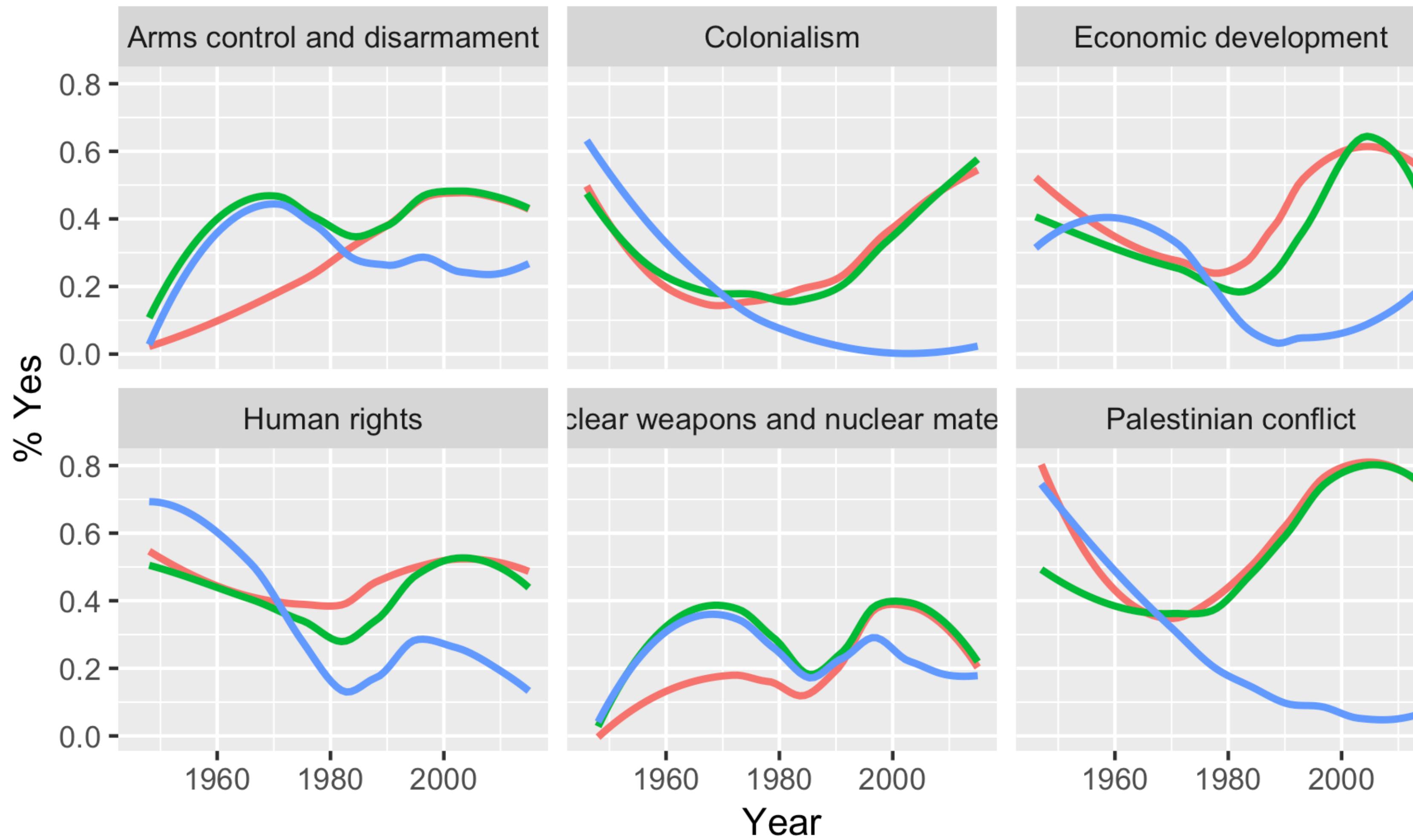
- ❑ Open today's demo project
- ❑ Knit the document and discuss the results with your neighbor
- ❑ Then, change Turkey to a different country, and plot again

```
un_votes %>%  
  filter(country %in% c("UK & NI", "US", "Turkey")) %>%  
  inner_join(un_roll_calls, by = "rcid") %>%  
  inner_join(un_roll_call_issues, by = "rcid") %>%  
  group_by(country, year = year(date), issue) %>%  
  summarize(  
    votes = n(),  
    percent_yes = mean(vote == "yes")  
) %>%  
  filter(votes > 5) %>% # only use records where there are more than 5 votes  
  ggplot(mapping = aes(x = year, y = percent_yes, color = country)) +  
  geom_smooth(method = "loess", se = FALSE) +  
  facet_wrap(~ issue) +  
  labs(  
    title = "Percentage of Yes votes in the UN General Assembly",  
    subtitle = "1946 to 2015",  
    y = "% Yes",  
    x = "Year",  
    color = "Country"  
)
```

```
un_votes %>%
  filter(country %in% c("UK & NI", "US", "Turkey")) %>%
  inner_join(un_roll_calls, by = "rcid") %>%
  inner_join(un_roll_call_issues, by = "rcid") %>%
  group_by(country, year = year(date), issue) %>%
  summarize(
    votes = n(),
    percent_yes = mean(vote == "yes")
  ) %>%
  filter(votes > 5) %>% # only use records where there are more than 5 votes
  ggplot(mapping = aes(x = year, y = percent_yes, color = country)) +
  geom_smooth(method = "loess", se = FALSE) +
  facet_wrap(~ issue) +
  labs(
    title = "Percentage of Yes votes in the UN General Assembly",
    subtitle = "1946 to 2015",
    y = "% Yes",
    x = "Year",
    color = "Country"
  )
```

```
un_votes %>%
  filter(country %in% c("UK & NI", "US", "France")) %>%
  inner_join(un_roll_calls, by = "rcid") %>%
  inner_join(un_roll_call_issues, by = "rcid") %>%
  group_by(country, year = year(date), issue) %>%
  summarize(
    votes = n(),
    percent_yes = mean(vote == "yes")
  ) %>%
  filter(votes > 5) %>% # only use records where there are more than 5 votes
  ggplot(mapping = aes(x = year, y = percent_yes, color = country)) +
  geom_smooth(method = "loess", se = FALSE) +
  facet_wrap(~ issue) +
  labs(
    title = "Percentage of Yes votes in the UN General Assembly",
    subtitle = "1946 to 2015",
    y = "% Yes",
    x = "Year",
    color = "Country"
  )
```

Percentage of Yes votes in the UN General Assembly 1946 to 2015



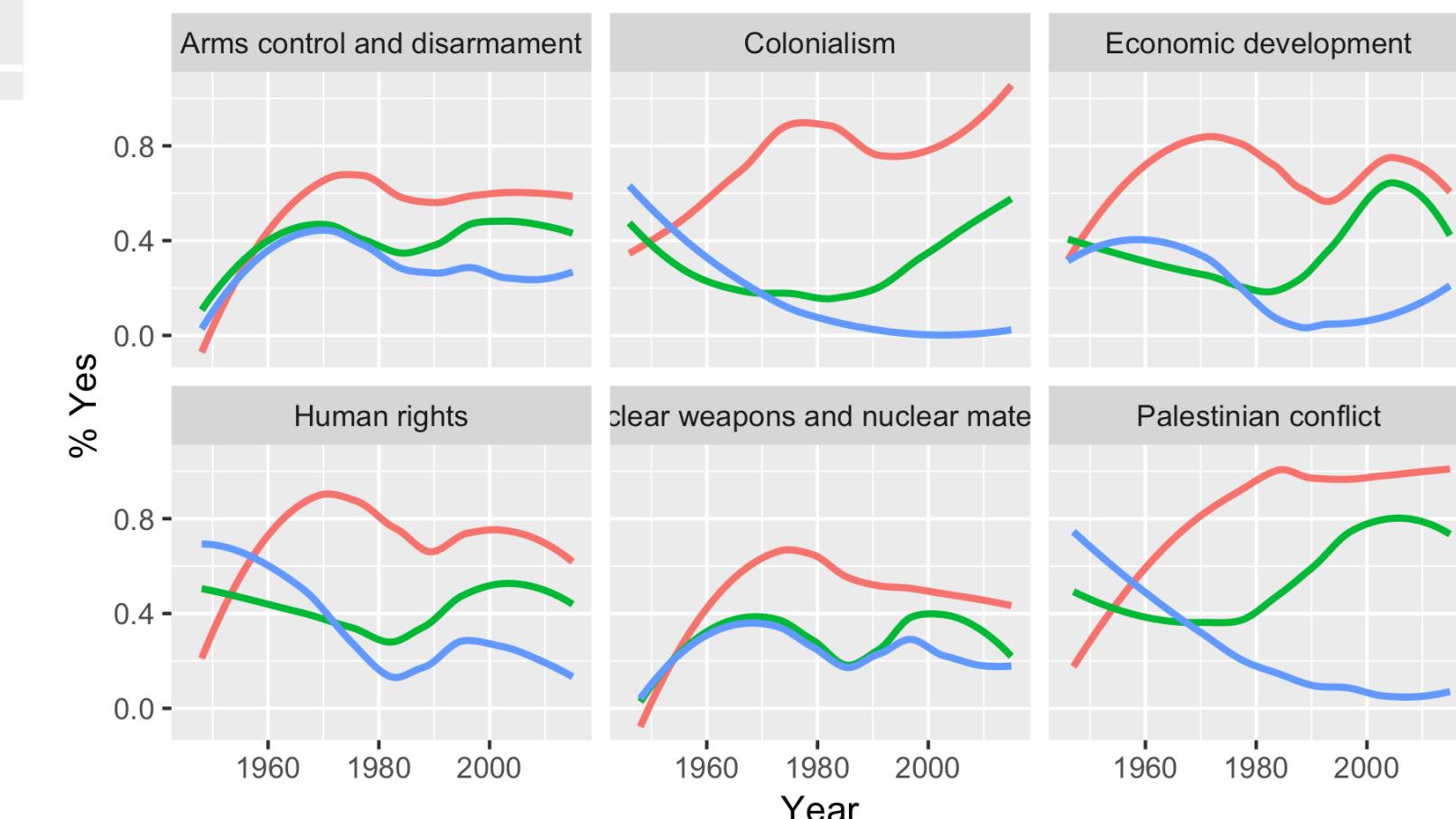
Country

- France
- UK & NI
- US

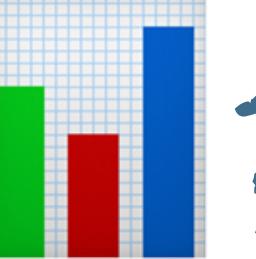
Country

- Turkey
- UK & NI
- US

Percentage of Yes votes in the UN General Assembly
1946 to 2015



bit.ly/let-eat-cake-cc

why  = ?

more likely for
students to have
intuition
coming in



easier for
students
to catch their
own mistakes





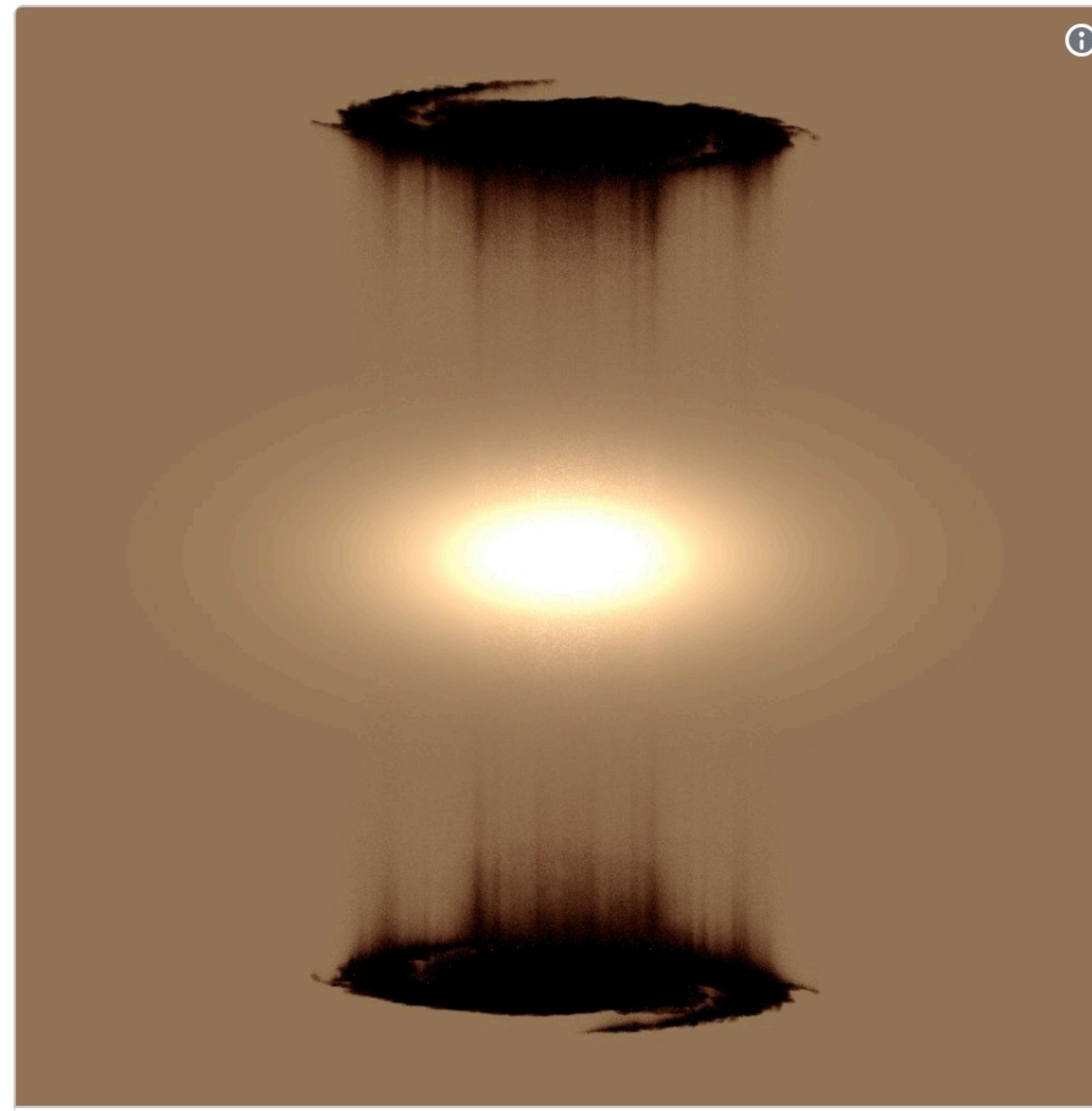
accidental aRt
@accidental_aRt

When data visualization goes beautifully wrong. Brought to you by [@ErikaMudrak](#) & [@kara_woo](#).

Submit your art at: [accidental-aRt.tumblr.com](#)

Joined October 2013

[Tweet to accidental aRt](#)



Judy Schmidt
@SpaceGeck

sometimes if you put a negative where it shouldn't be you accidentally open a portal to unknown space

63 7:34 PM - Dec 27, 2018

My niece just found the color markers.
[tumblr.co/Z7M53q2eUJdrx](#)

5 12:40 PM - Dec 7, 2018



accidental-art
My niece just found the color markers.
[accidental-art.tumblr.com](#)

why  = ?

more likely for
students to have
intuition
coming in



easier for
students
to catch their
own mistakes



who doesn't
like a good
piece of ~~cake~~
visualization?



ex: Introduction to R for Data Science

Microsoft Professional Program Certificate in Data Science

Course Syllabus

Section 1: Introduction to Basics

Take your first steps with R. Discover the basic data types in R and assign your first variable.

Section 2: Vectors

Analyze gambling behaviour using vectors. Create, name and select elements from vectors.

Section 3: Matrices

Learn how to work with matrices in R. Do basic computations with them and demonstrate your knowledge by analyzing the Star Wars box office figures.

Section 4: Factors

R stores categorical data in factors. Learn how to create, subset and compare categorical data.

Section 5: Data Frames

When working R, you'll probably deal with Data Frames all the time. Therefore, you need to know how to create one, select the most interesting parts of it, and order them.

Section 6: Lists

Lists allow you to store components of different types. Section 6 will show you how to deal with lists.

Section 7: Basic Graphics

Discover R's packages to do graphics and create your own data visualizations.

ex: Data Science Specialization

Johns Hopkins University

JOHNS HOPKINS Data Science Specialization

About How It Works **Courses** Instructors Enrollment Options FAQ

Enroll Starts Sep 27

1
COURSE

The Data Scientist's Toolbox

★★★★★ 4.5 16,022 ratings • 3,325 reviews

In this course you will get an introduction to the main tools and ideas in the data scientist's toolbox. The course gives an overview of the data, questions, and tools that data analysts and data scientists work with. There are two components to this course. The first is a c... [MORE](#)

2
COURSE

R Programming

★★★★★ 4.6 12,076 ratings • 2,558 reviews

In this course you will learn how to program in R and how to use R for effective data analysis. You will learn how to install and configure software necessary for a statistical programming environment and describe generic programming language concepts as they are i... [MORE](#)

3
COURSE

Getting and Cleaning Data

★★★★★ 4.6 5,178 ratings • 829 reviews

Before you can work with data you have to get some. This course will cover the basic ways that data can be obtained. The course will cover obtaining data from the web, from APIs, from databases and from colleagues in various formats. It will also cover the basics of data ... [MORE](#)

4
COURSE

Exploratory Data Analysis

★★★★★ 4.7 3,957 ratings • 591 reviews

This course covers the essential exploratory techniques for summarizing data. These techniques are typically applied before formal modeling commences and can help inform the development of more complex statistical models. Exploratory techniques are also important for eliminating or sharpening potential hypotheses about the world that can be addressed by the data. We will cover in detail the plotting systems in R as well as some of the basic principles of constructing data graphics. We will also cover some of the common multivariate statistical techniques used to visualize high-dimensional data. [LESS](#)

1
SECTION

25 hours to complete

Week 1: Background, Getting Started, and Nuts & Bolts

This week covers the basics to get you started up with R. The Background Materials lesson contains information about course mechanics and some videos on i... [MORE](#)

28 videos (Total 129 min), 9 readings, 8 quizzes [SEE ALL](#)

2
SECTION

12 hours to complete

Week 2: Programming with R

Welcome to Week 2 of R Programming. This week, we take the gloves off, and the lectures cover key topics like control structures and functions. We also intr... [MORE](#)

13 videos (Total 91 min), 3 readings, 5 quizzes [SEE ALL](#)

3
SECTION

10 hours to complete

Week 3: Loop Functions and Debugging

We have now entered the third week of R Programming, which also marks the halfway point. The lectures this week cover loop functions and the debuggi... [MORE](#)

8 videos (Total 61 min), 2 readings, 4 quizzes [SEE ALL](#)

4
SECTION

11 hours to complete

Week 4: Simulation & Profiling

This week covers how to simulate data in R, which serves as the basis for doing simulation studies. We also cover the profiler in R which lets you collect det... [MORE](#)

6 videos (Total 42 min), 4 readings, 5 quizzes [SEE ALL](#)

cherish
day
one





Which of the following is more likely to be **welcoming** for a wide range of students?

- ❑ Install R
- ❑ Install RStudio
- ❑ Install the following packages:
 - ❑ tidyverse
 - ❑ rmarkdown
 - ❑ ...
- ❑ Load these packages
- ❑ Install git

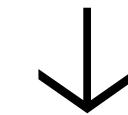
- ❑ Go to rstudio.cloud (or some other server based solution)
- ❑ Log in with your ID & pass

```
> hello R!
```

method of delivery,
and medium of interaction matters



R Studio®



Data
Analysis
References
Appendix

UN Votes

Mine Çetinkaya-Rundel

2018-09-26

Let's take a look at the voting history of countries in the United Nations General Assembly. We will be using data from the `unvotes` package. Additionally, we will make use of the `tidyverse` and `lubridate` packages for the analysis, and the `DT` package for interactive display of tabular output.

Data

The `unvotes` package provides three datasets we can work with: `un_roll_calls`, `un_roll_call_issues`, and `un_votes`. Each of these datasets contains a variable called `roid`, the roll call id, which can be used as a unique identifier to join them with each other.

- The `un_votes` dataset provides information on the voting history of the United Nations General Assembly. It contains one row for each country-vote pair.

`un_votes`

```
## # A tibble: 738,764 x 4
##   roid country      country_code vote
##   <int> <chr>        <chr>     <fct>
## 1 3 United States of America US      yes
## 2 3 Canada          CA      no
## 3 3 Cuba            CU      yes
## 4 3 Haiti           HT      yes
## 5 3 Dominican Republic DO      yes
## 6 3 Mexico          MX      yes
## 7 3 Guatemala       GT      yes
## 8 3 Honduras        HN      yes
## 9 3 El Salvador     SV      yes
## 10 3 Nicaragua       NI     yes
## # ... with 738,754 more rows
```

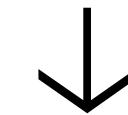
- The `un_roll_calls` dataset contains information on each roll call vote of the United Nations General Assembly.

`un_roll_calls`

```
## # A tibble: 5,429 x 9
##   roid session importantvote date      unres amend para short descr
##   <int> <dbl> <dbl> <date>    <chr> <dbl> <dbl> <dbl> <dbl>
## 1 3     1     1     0 1946-01-01 8/1/66    1     0 AMEN_ TO ADD_
## 2 4     1     1     0 1946-01-02 8/1/79    0     0 SECU_ TO ADD_
## 3 5     1     1     0 1946-01-04 8/1/98    0     0 VOTI_ "TO AD_
```



R Studio®



Data
Analysis
References
Appendix

UN Votes

Mine Çetinkaya-Rundel

2018-09-26

Let's take a look at the voting history of countries in the United Nations General Assembly. We will be using data from the `unvotes` package. Additionally, we will make use of the `tidyverse` and `lubridate` packages for the analysis, and the `DT` package for interactive display of tabular output.

Data

The `unvotes` package provides three datasets we can work with: `un_roll_calls`, `un_roll_call_issues`, and `un_votes`. Each of these datasets contains a variable called `roid`, the roll call id, which can be used as a unique identifier to join them with each other.

- The `un_votes` dataset provides information on the voting history of the United Nations General Assembly. It contains one row for each country-vote pair.

```
un_votes
## # A tibble: 738,764 x 4
##   roid country      country_code vote
##   <int> <chr>        <chr>     <fct>
## 1 3 United States of America US      yes
## 2 3 Canada          CA      no
## 3 3 Cuba            CU      yes
## 4 3 Haiti           HT      yes
## 5 3 Dominican Republic DO      yes
## 6 3 Mexico          MX      yes
## 7 3 Guatemala       GT      yes
## 8 3 Honduras        HN      yes
## 9 3 El Salvador      SV      yes
## 10 3 Nicaragua        NI     yes
## # ... with 738,754 more rows
```

- The `un_roll_calls` dataset contains information on each roll call vote of the United Nations General Assembly.

```
un_roll_calls
## # A tibble: 5,429 x 9
##   roid session importantvote date      unres amend para short descr
##   <int> <dbl> <dbl> <date>    <chr> <dbl> <dbl> <dbl> <chr>
## 1 3     1     1     0 1946-01-01 8/1/66    1     0 AMEN_ TO ADD_
## 2 4     1     1     0 1946-01-02 8/1/79    0     0 SECU_ TO ADD_
## 3 5     1     1     0 1946-01-04 8/1/98    0     0 VOTI_ "TO AD_
```

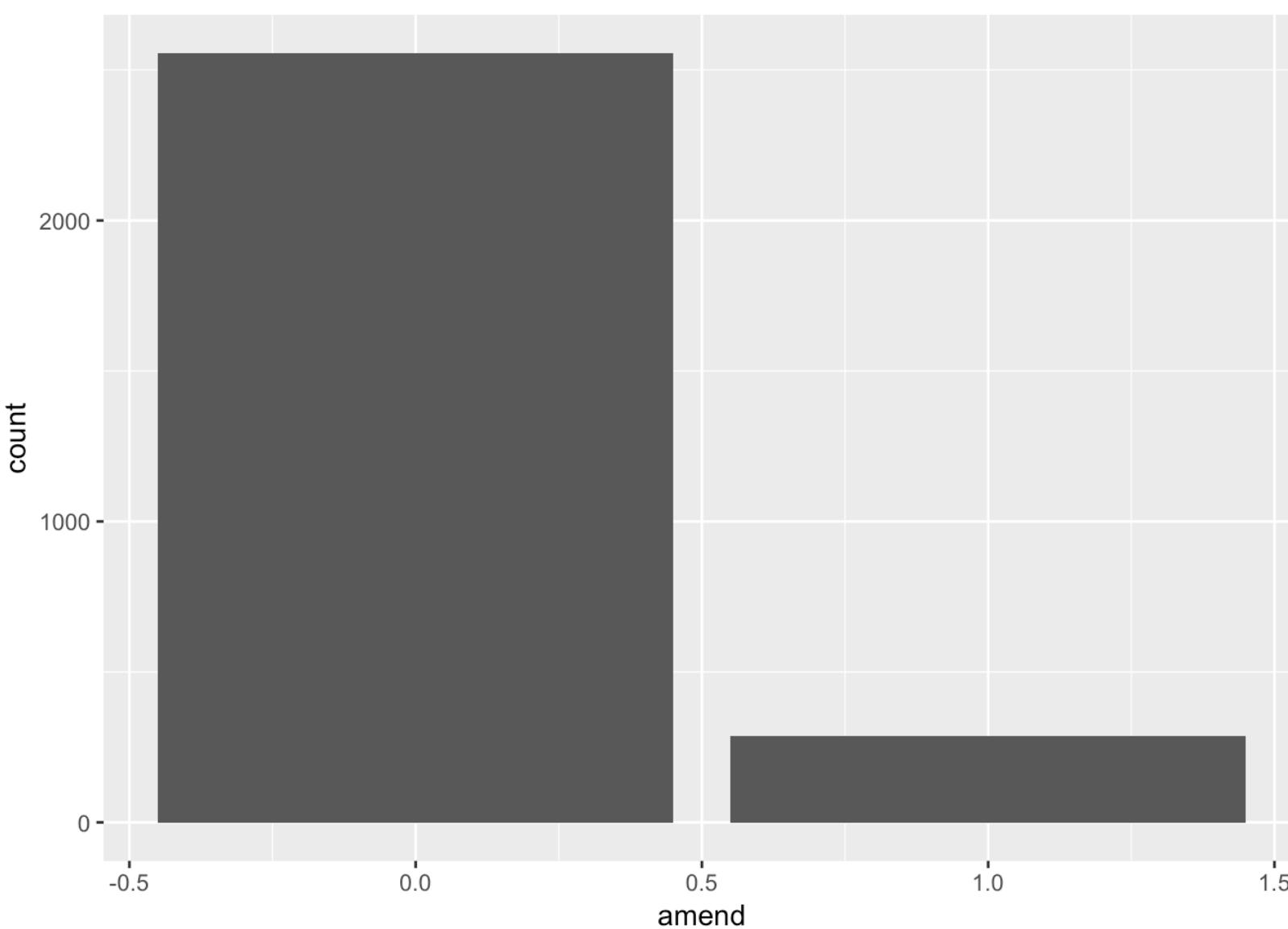
**skip
baby
steps**

73

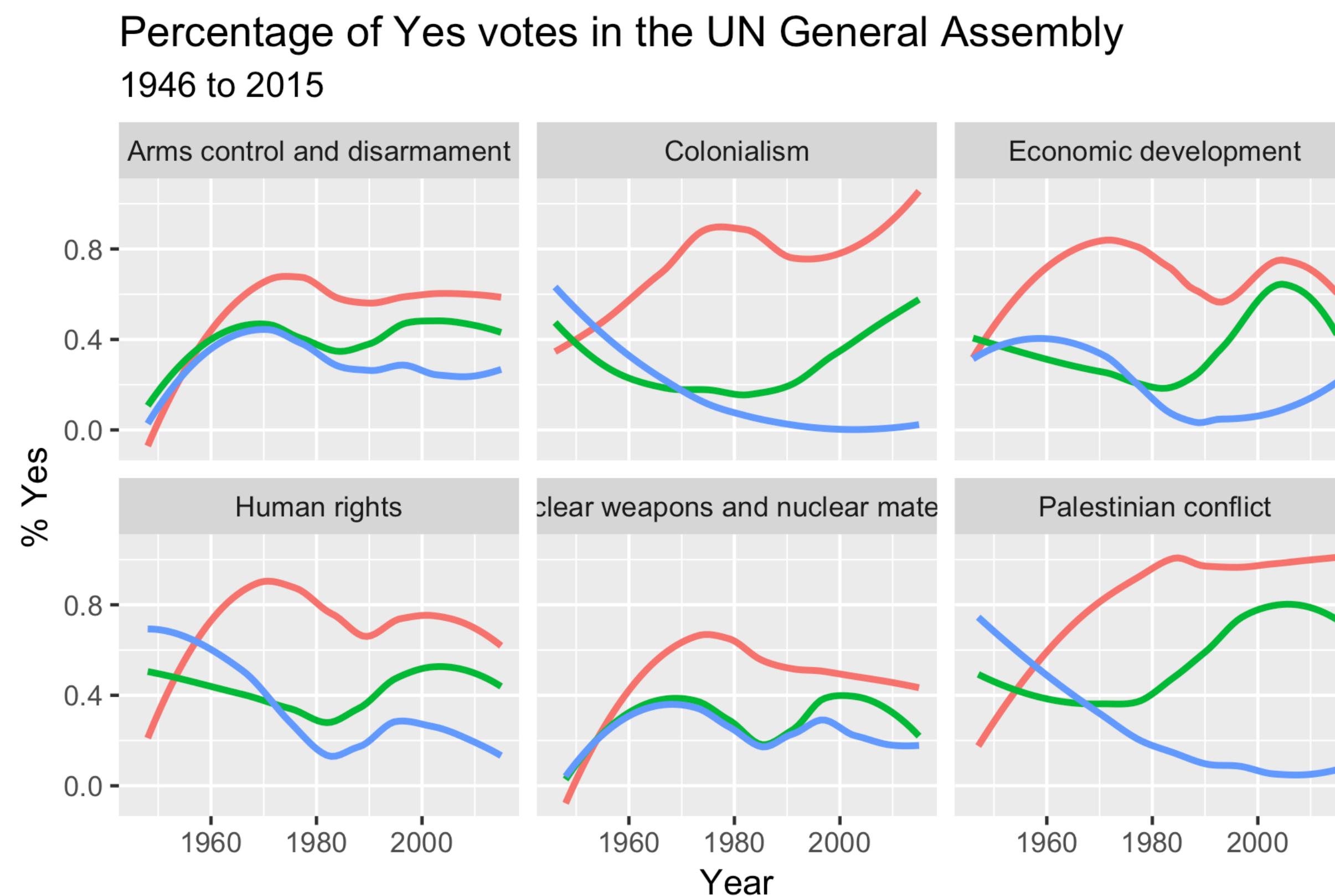


Which of the following is more likely to **inspire** students to want to learn more?

Create a visualization displaying whether the vote was on an amendment.



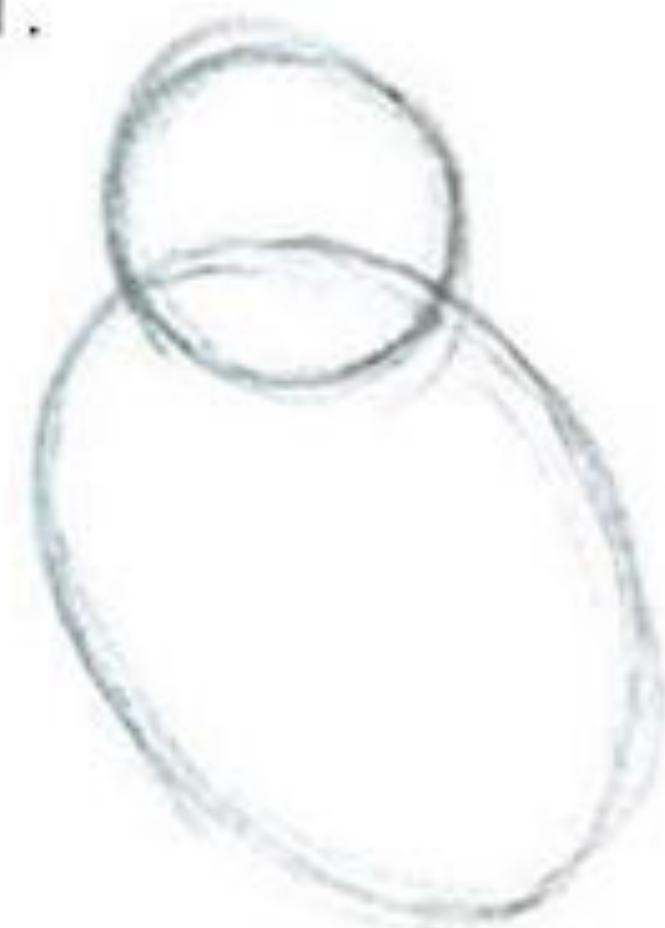
Create a visualization displaying how US, UK, and Turkey voted over the years on issues of arms control and disarmament, colonialism, economic development, human rights, nuclear weapons, and Palestinian conflict.



non-trivial examples can be motivating,
but need to avoid !

How to draw an owl

1.



2.



1. Draw some circles

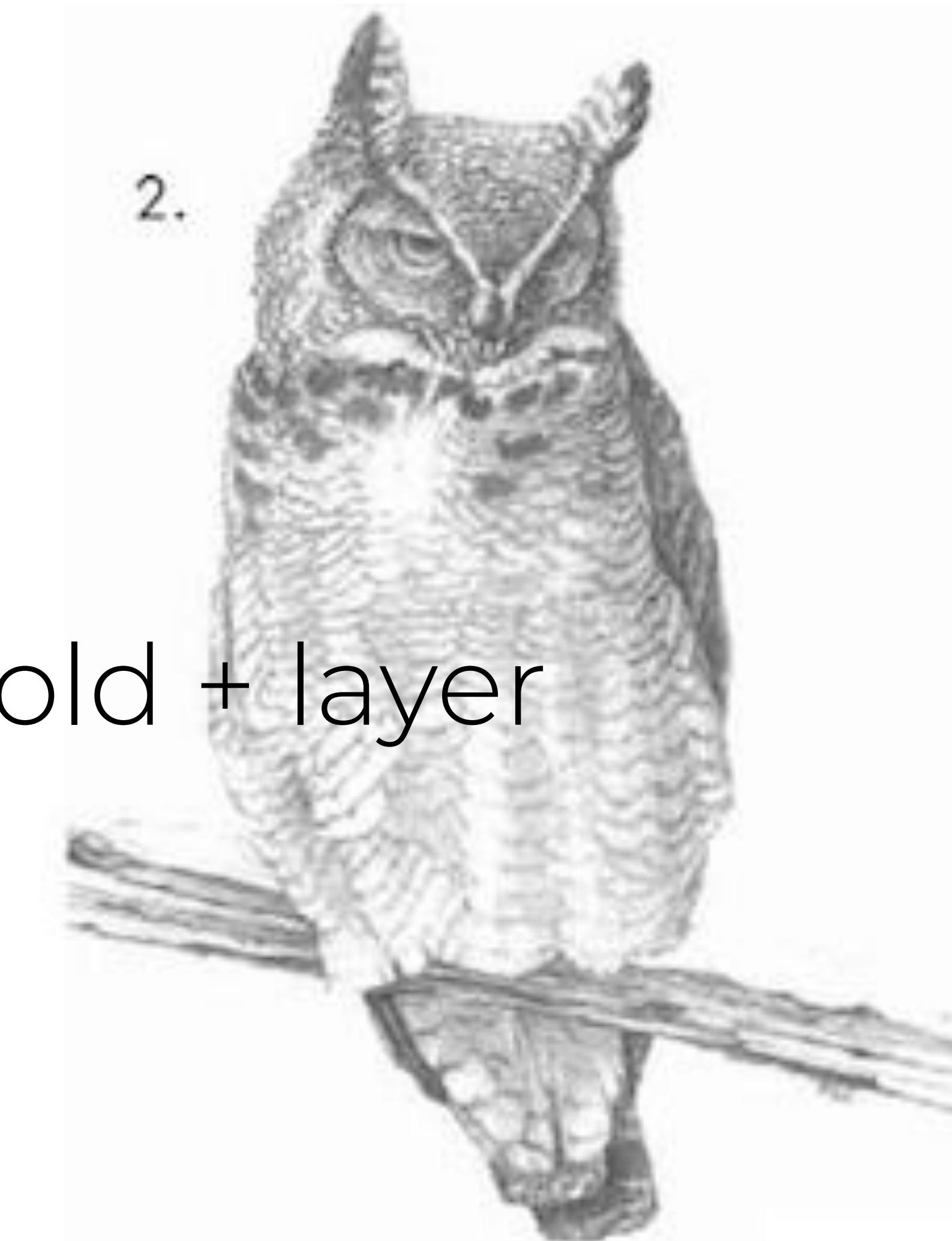
2. Draw the rest of the  owl

How to draw an owl

1.



2.

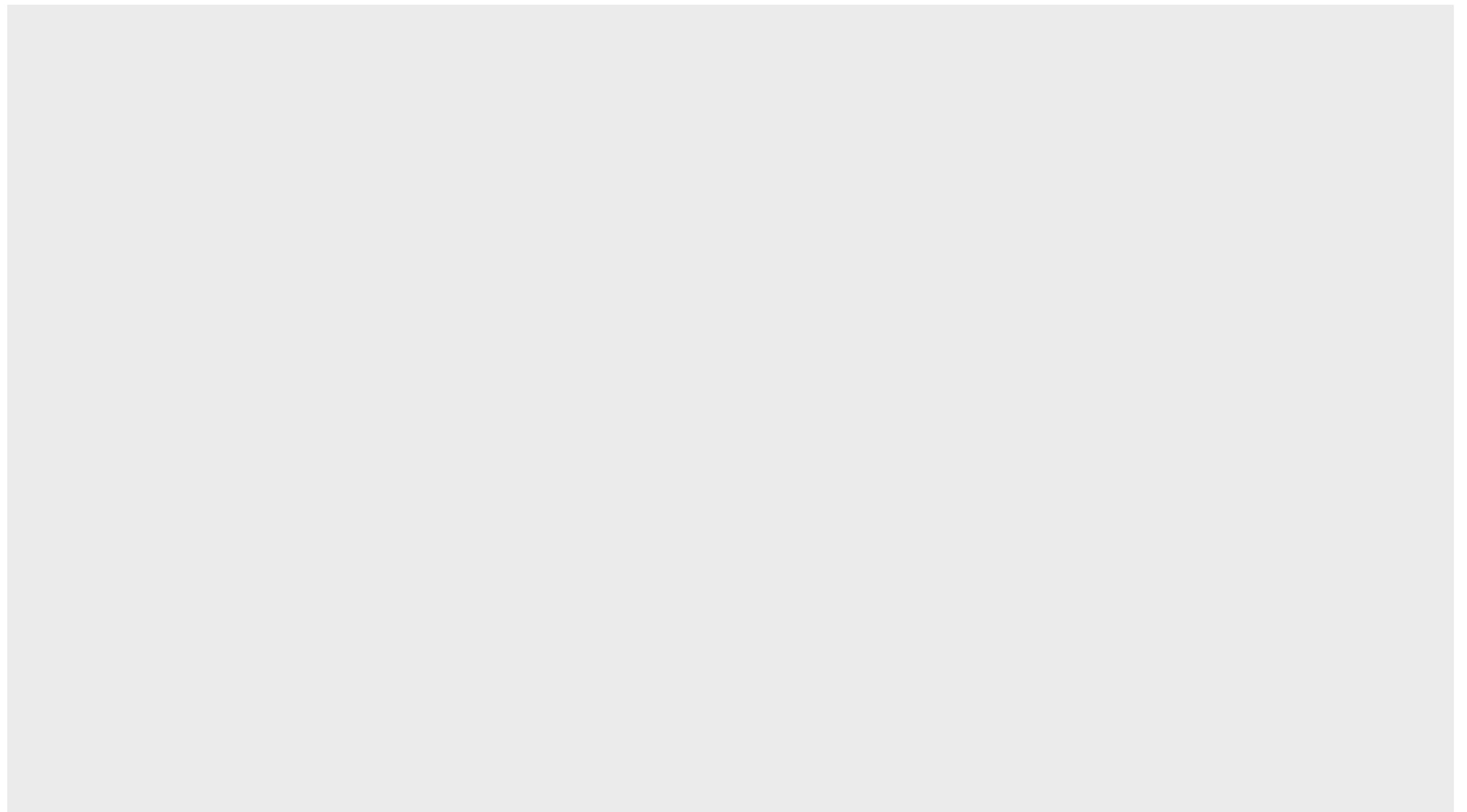


scaffold + layer

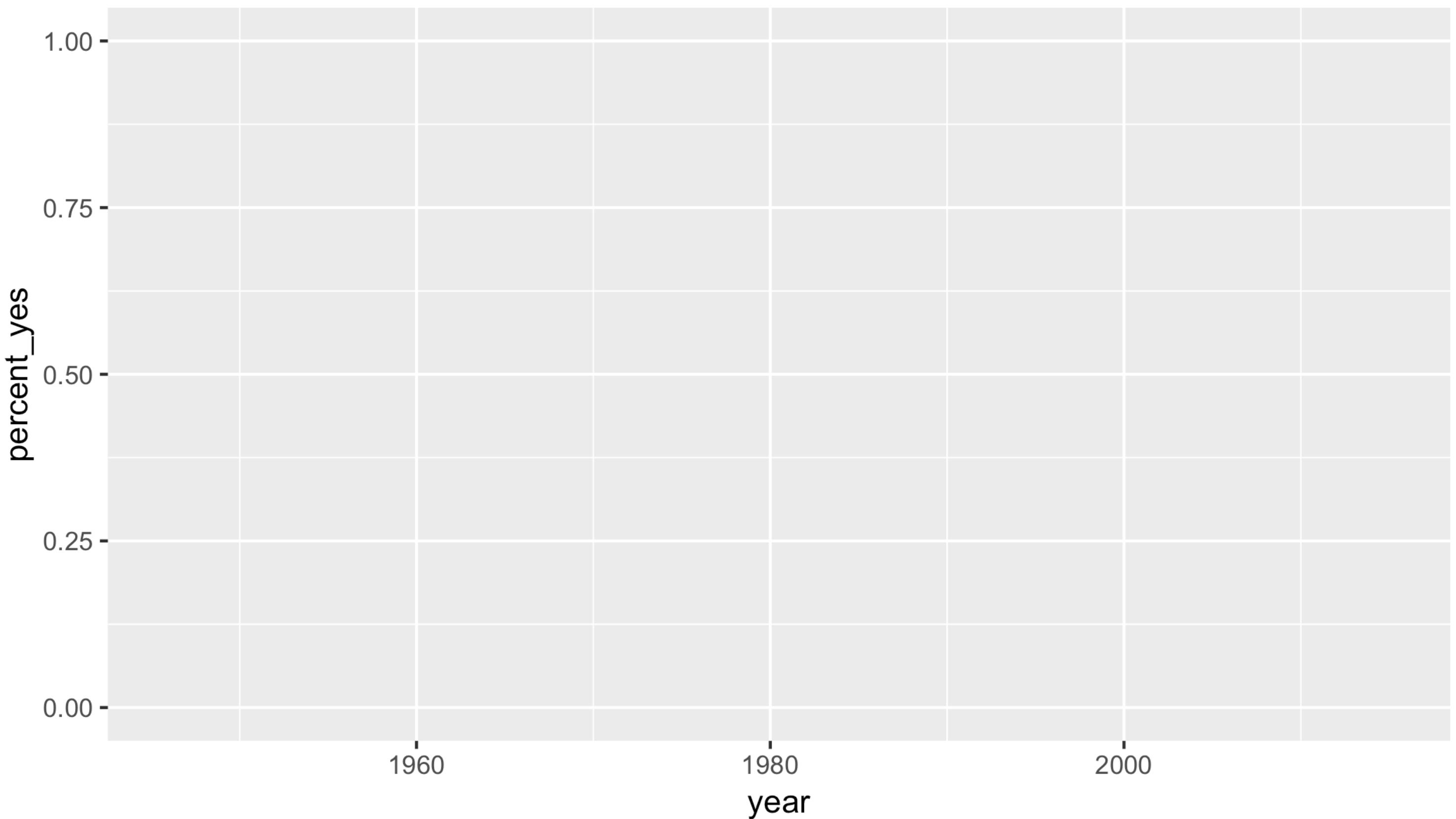
1. Draw some circles

2. Draw the rest of the @#\$% owl

```
ggplot(data = un_votes_joined)
```



```
ggplot(data = un_votes_joined,  
       mapping = aes(x = year, y = percent_yes))
```



```
ggplot(data = un_votes_joined,  
       mapping = aes(x = year, y = percent_yes))
```

function(arguments)

often a verb

what to apply that
verb to

```
ggplot(data = un_votes_joined,  
       mapping = aes(x = year, y = percent_yes))
```

rows =
observations

	country	year	issue	votes	percent_yes
1	Turkey	1946	Colonialism	15	0.80000000
2	Turkey	1946	Economic development	7	0.57142857
3	Turkey	1947	Colonialism	9	0.22222222
4	Turkey	1947	Palestinian conflict	6	0.00000000
5	Turkey	1948	Arms control and disarmament	8	0.00000000
6	Turkey	1948	Colonialism	13	0.46153846
7	Turkey	1948	Human rights	11	0.18181818
8	Turkey	1948	Nuclear weapons and nuclear material	7	0.00000000
9	Turkey	1948	Palestinian conflict	11	0.27272727
10	Turkey	1949	Colonialism	35	0.54285714
11	Turkey	1949	Economic development	11	0.09090909
12	Turkey	1949	Palestinian conflict	17	0.23529412
13	Turkey	1950	Colonialism	14	0.64285714
14	Turkey	1952	Colonialism	12	0.08333333
15	Turkey	1952	Human rights	11	0.36363636
16	Turkey	1953	Colonialism	9	0.00000000
17	Turkey	1953	Human rights	7	0.28571429
18	Turkey	1954	Colonialism	8	0.12500000

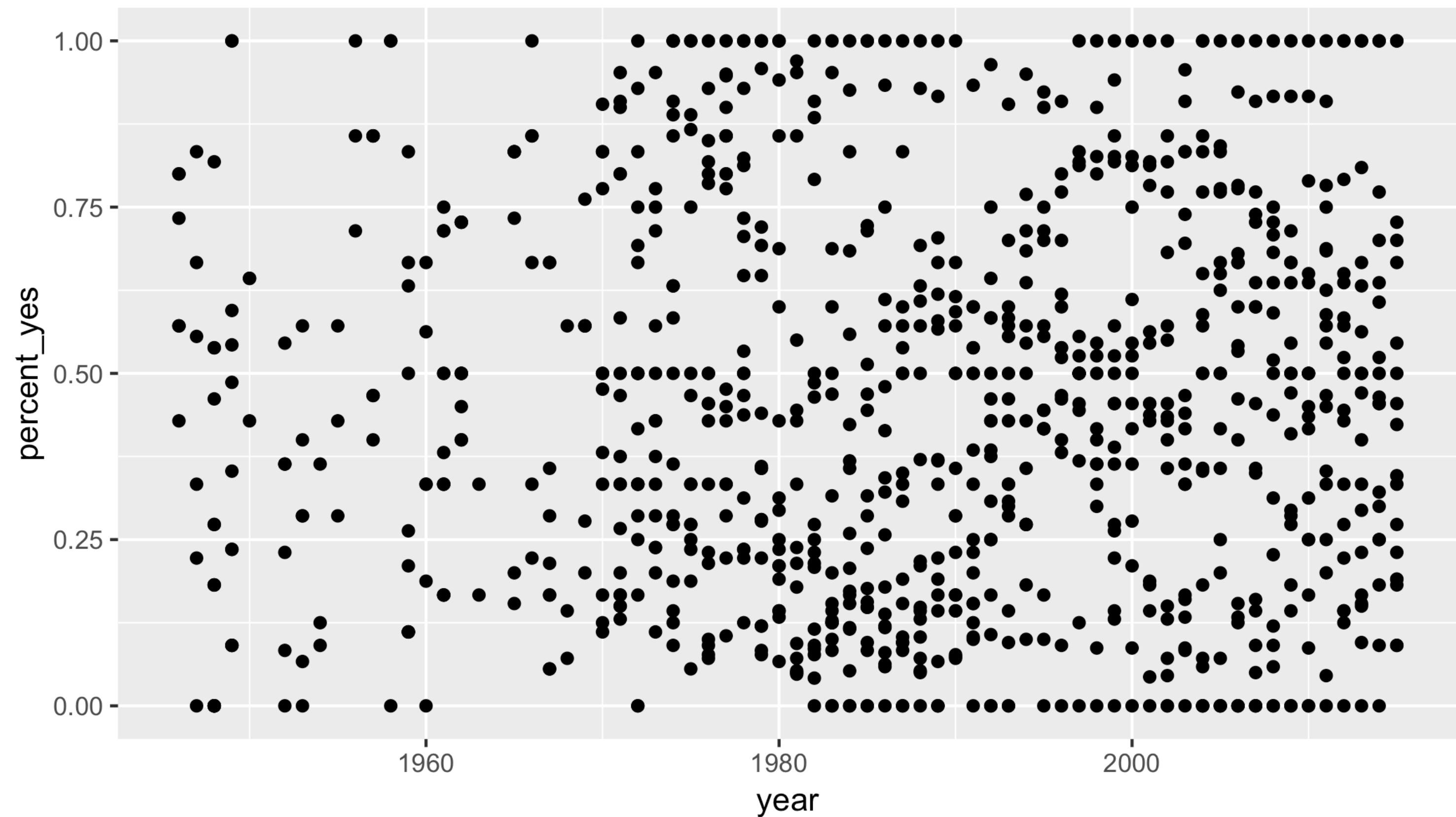
"tidy"
data frame

columns =
variables

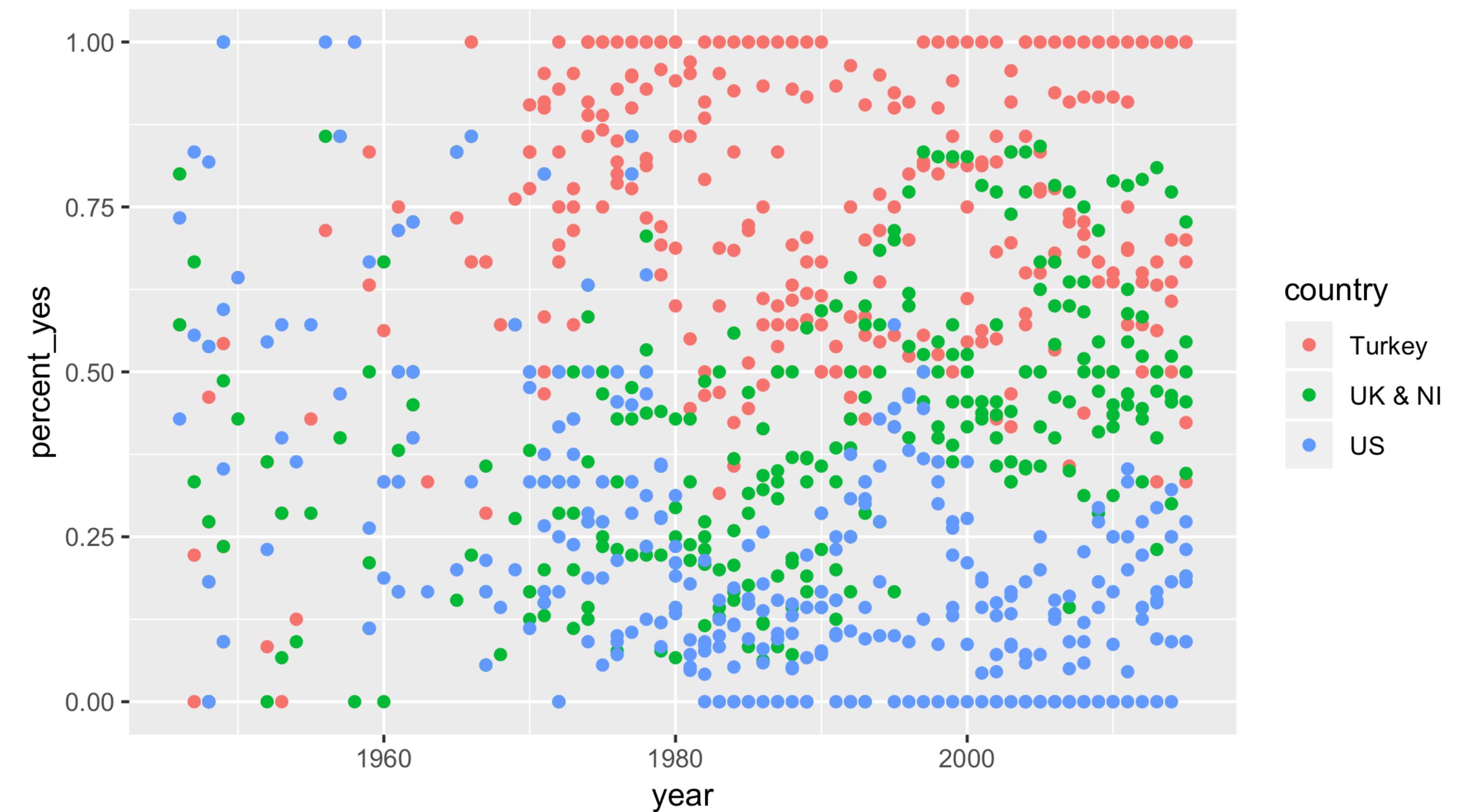


bit.ly/let-eat-cake-cc

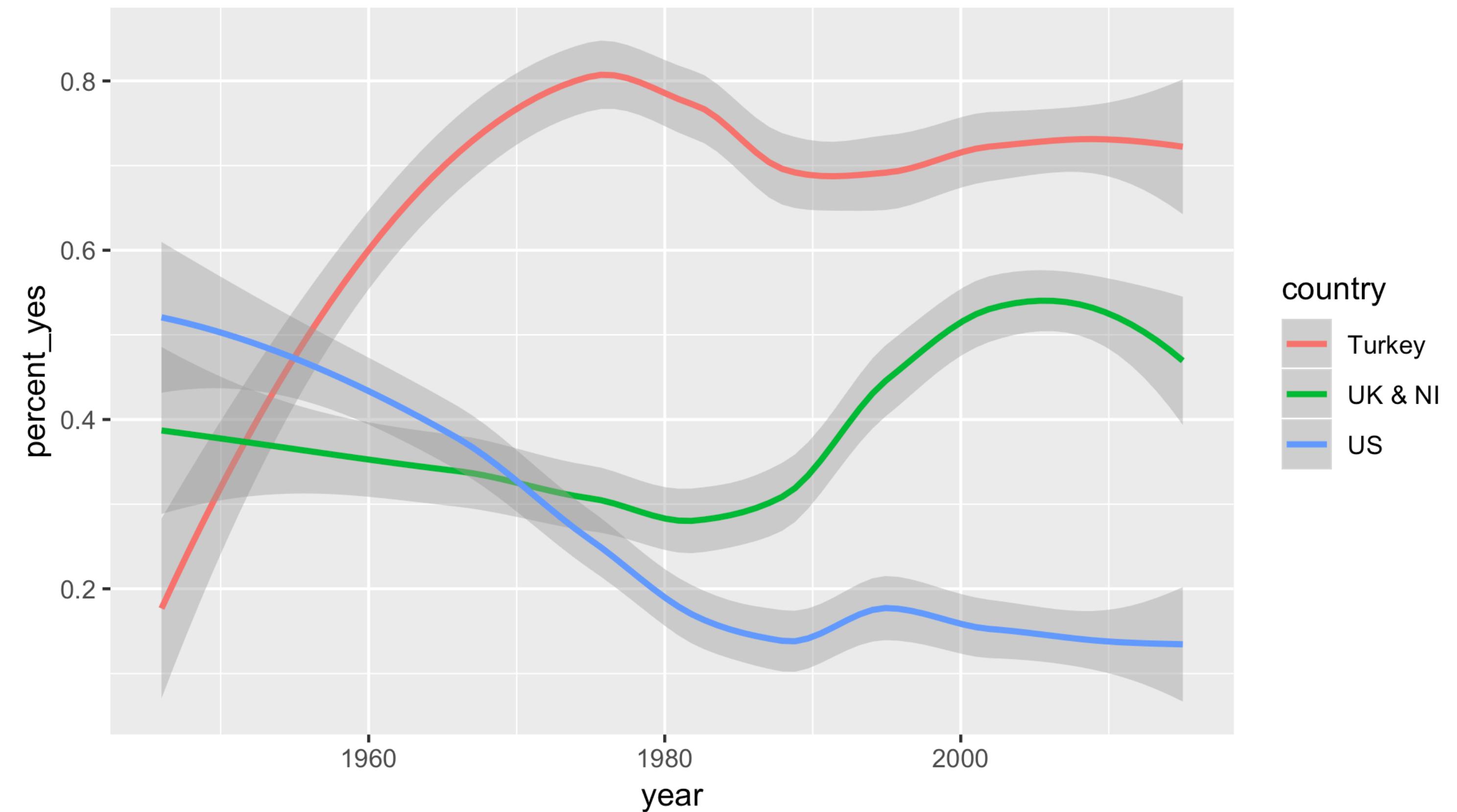
```
ggplot(data = un_votes_joined,  
       mapping = aes(x = year, y = percent_yes)) +  
  geom_point()
```



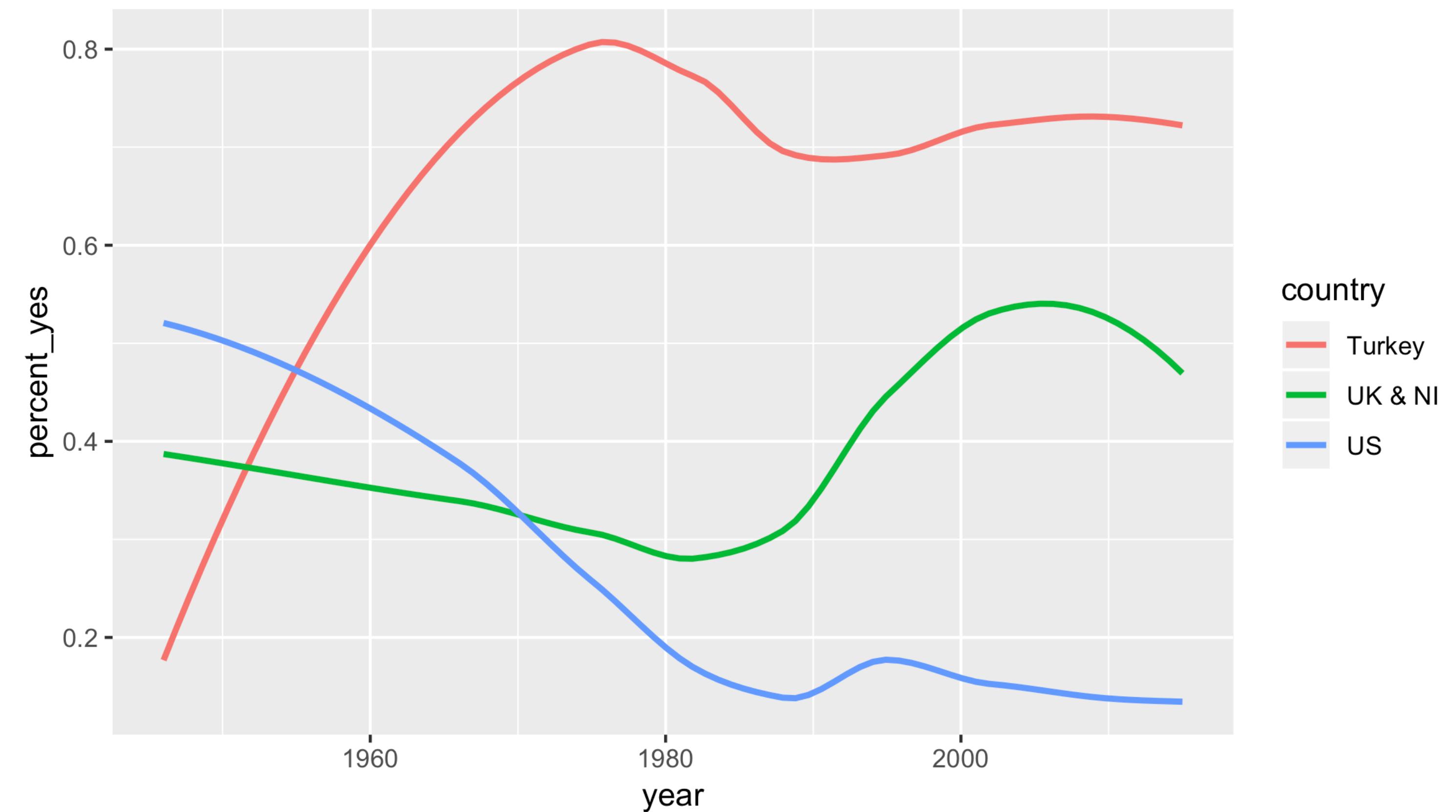
```
ggplot(data = un_votes_joined,  
       mapping = aes(x = year, y = percent_yes, color = country)) +  
geom_point()
```



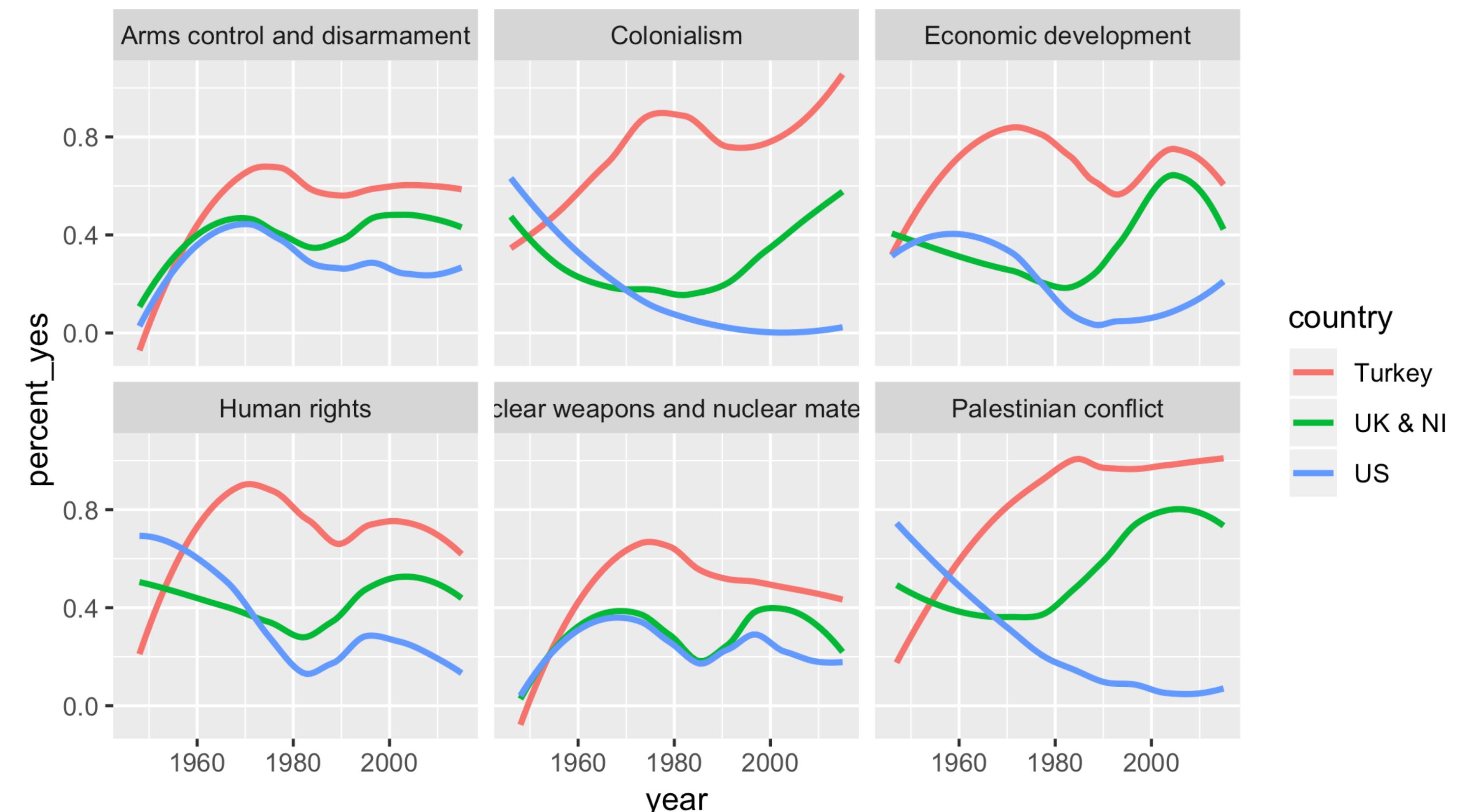
```
ggplot(data = un_votes_joined,  
       mapping = aes(x = year, y = percent_yes, color = country)) +  
  geom_smooth(method = "loess")
```



```
ggplot(data = un_votes_joined,  
       mapping = aes(x = year, y = percent_yes, color = country)) +  
  geom_smooth(method = "loess", se = FALSE)
```



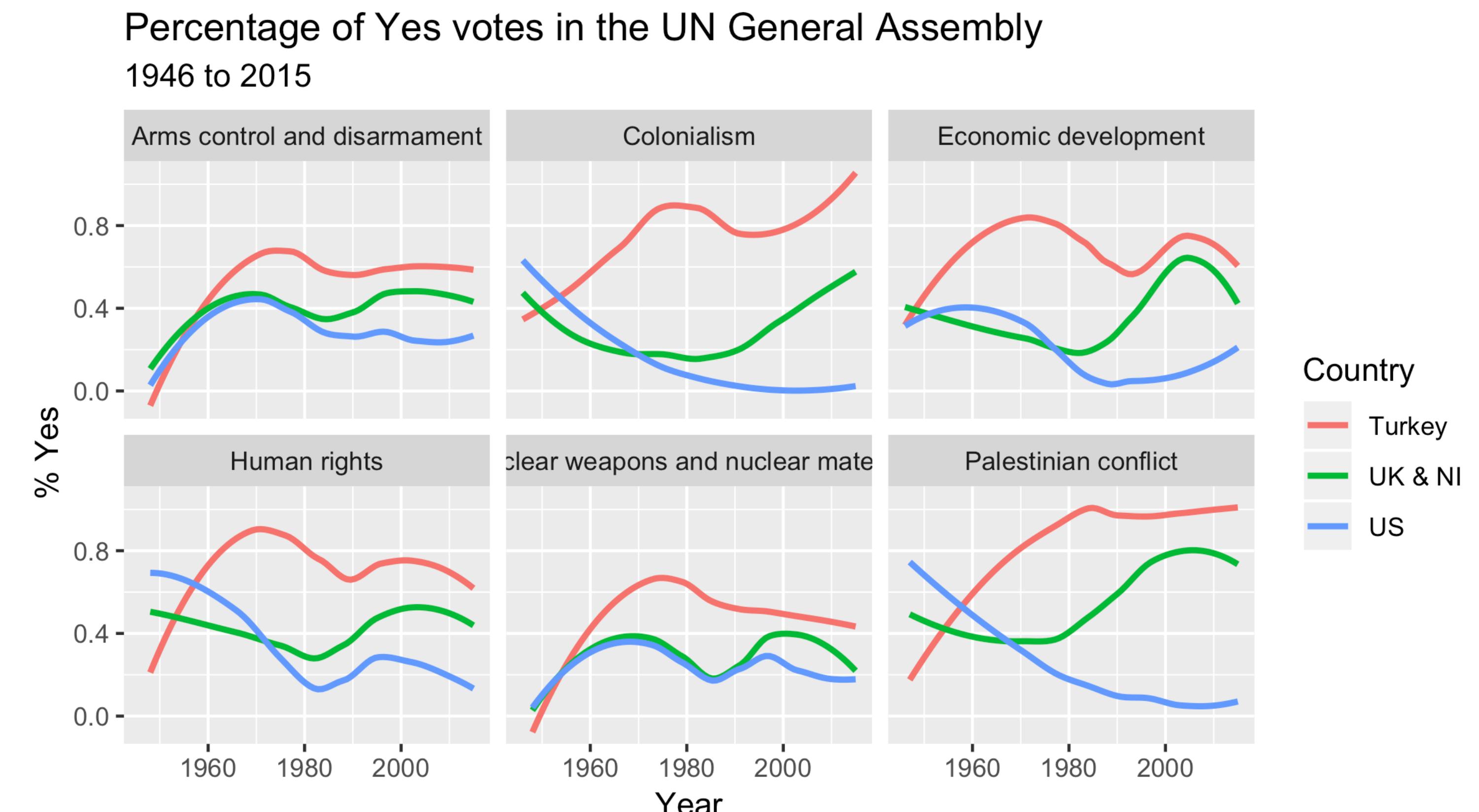
```
ggplot(data = un_votes_joined,  
       mapping = aes(x = year, y = percent_yes, color = country)) +  
  geom_smooth(method = "loess", se = FALSE) +  
  facet_wrap(~ issue)
```



```

ggplot(data = un_votes_joined,
       mapping = aes(x = year, y = percent_yes, color = country)) +
  geom_smooth(method = "loess", se = FALSE) +
  facet_wrap(~ issue) +
  labs(
    title = "Percentage of 'Yes' votes in the UN General Assembly",
    subtitle = "1946 to 2015",
    y = "% Yes",
    x = "Year",
    color = "Country"
)

```



hide
the
veggies

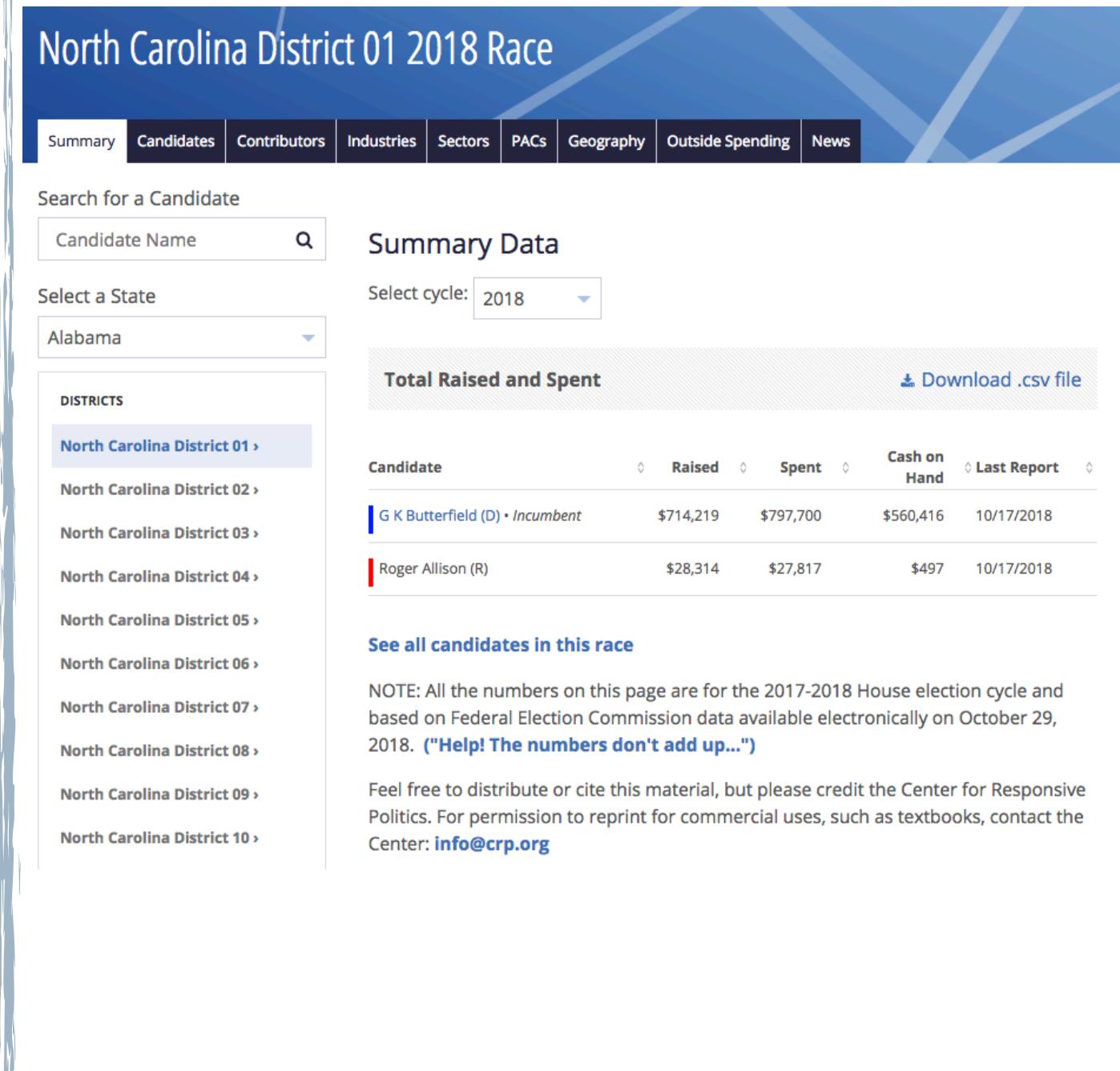




Which of the following is more likely to be **interesting** for a wide range of students?

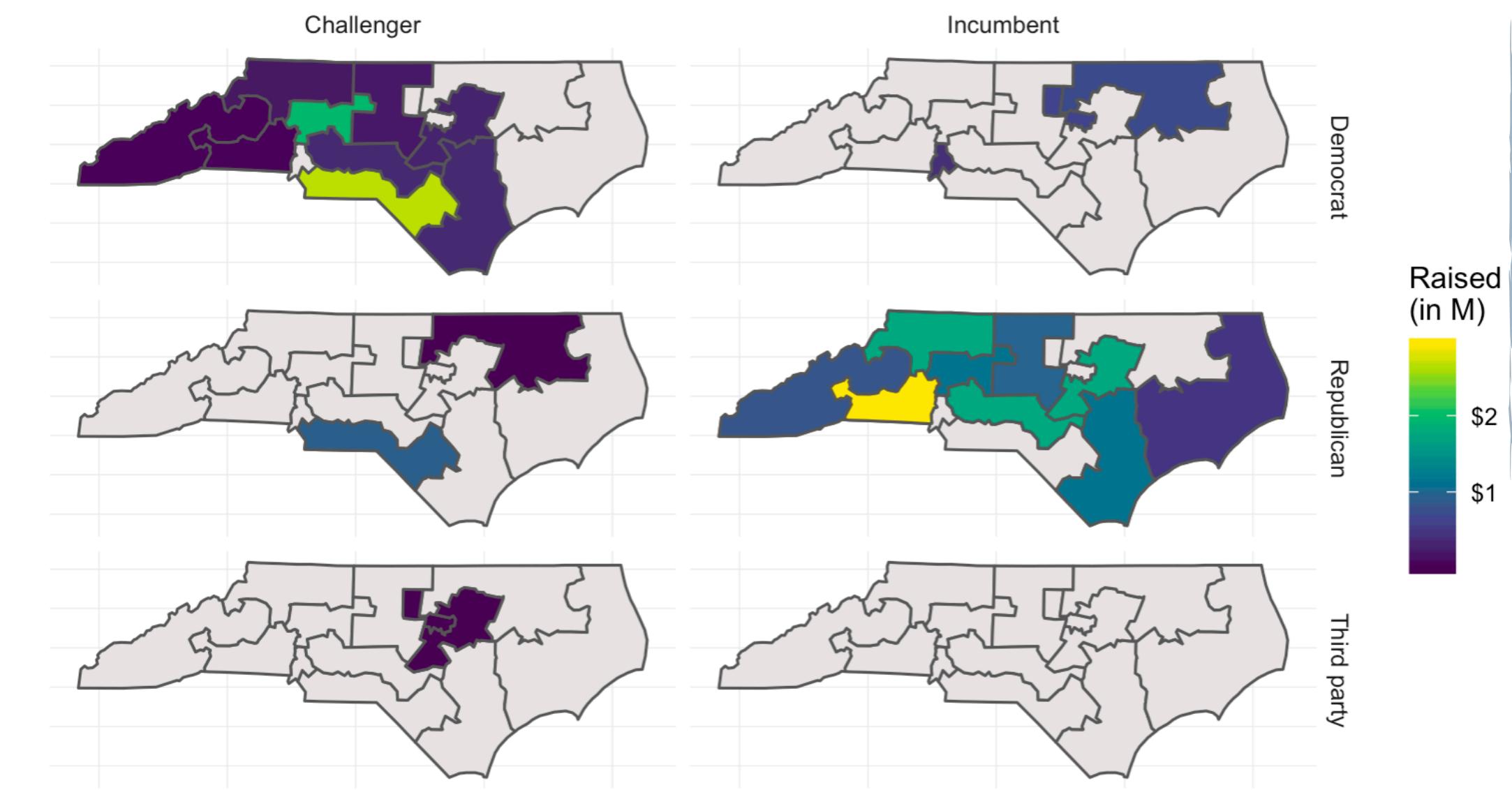
- Topic: Web scraping
- Tools:
 - **rvest**
 - regular expressions

- Today we start with this:



- and end with this:

Political contributions for 2018 NC Congressional Races
as of 9/30/2018



- and do so in a way that is easy to replicate for another state

students will encounter lots of new challenges along the way — let that happen, and then provide a solution

- **Lesson:** Web scraping essentials
for turning a structured table into
a data frame in R.

- **Lesson:** Web scraping essentials for turning a structured table into a data frame in R.

- **Ex 1:** Scrape the table off the web and save as a data frame.

Candidate	Raised	Spent	Cash on Hand	Last Report
G K Butterfield (D) • Incumbent	\$714,219	\$797,700	\$560,416	10/17/2018
Roger Allison (R)	\$28,314	\$27,817	\$497	10/17/2018



#	candidate_info	raised	spent	cash_on_hand	last_report	race
1	G K Butterfield (D) • Incumbent	714219	797700	560416	2018-10-17	North Carolina District 01
2	Roger Allison (R)	28314	27817	497	2018-10-17	North Carolina District 01

- **Lesson:** Web scraping essentials for turning a structured table into a data frame in R.

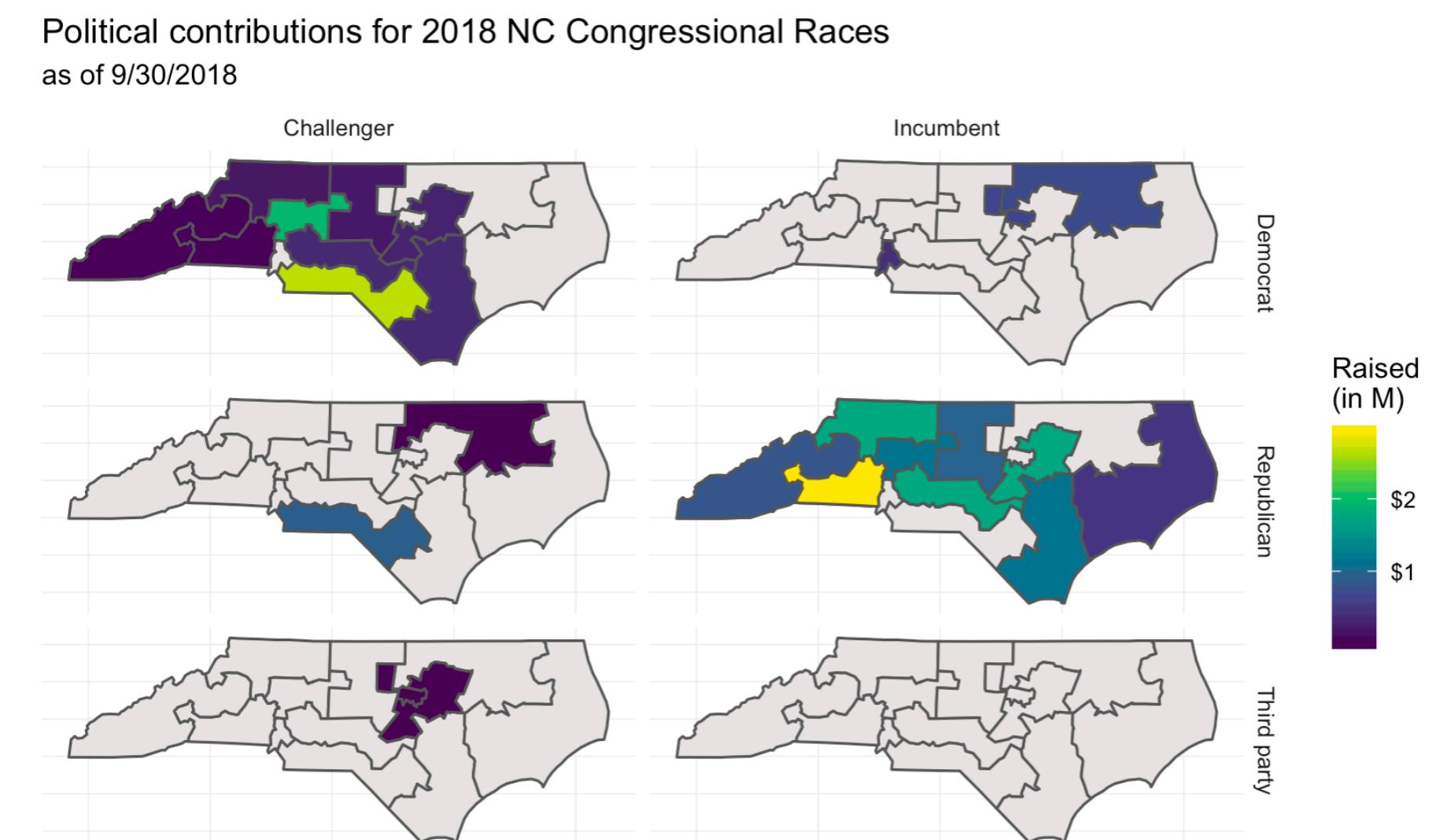
- **Ex 1:** Scrape the table off the web and save as a data frame.

Candidate	Raised	Spent	Cash on Hand	Last Report
G K Butterfield (D) • Incumbent	\$714,219	\$797,700	\$560,416	10/17/2018
Roger Allison (R)	\$28,314	\$27,817	\$497	10/17/2018



#	candidate_info	raised	spent	cash_on_hand	last_report	race
1	G K Butterfield (D) • Incumbent	714219	797700	560416	2018-10-17	North Carolina District 01
2	Roger Allison (R)	28314	27817	497	2018-10-17	North Carolina District 01

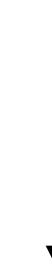
- **Ex 2:** What other information do we need represented as variables in the data to obtain the desired facets?



- **Lesson:** Web scraping essentials for turning a structured table into a data frame in R.

- **Ex 1:** Scrape the table off the web and save as a data frame.

Candidate	Raised	Spent	Cash on Hand	Last Report
G K Butterfield (D) • Incumbent	\$714,219	\$797,700	\$560,416	10/17/2018
Roger Allison (R)	\$28,314	\$27,817	\$497	10/17/2018



candidate_info	raised	spent	cash_on_hand	last_report	race
1 G K Butterfield (D) • Incumbent	714219	797700	560416	2018-10-17	North Carolina District 01
2 Roger Allison (R)	28314	27817	497	2018-10-17	North Carolina District 01

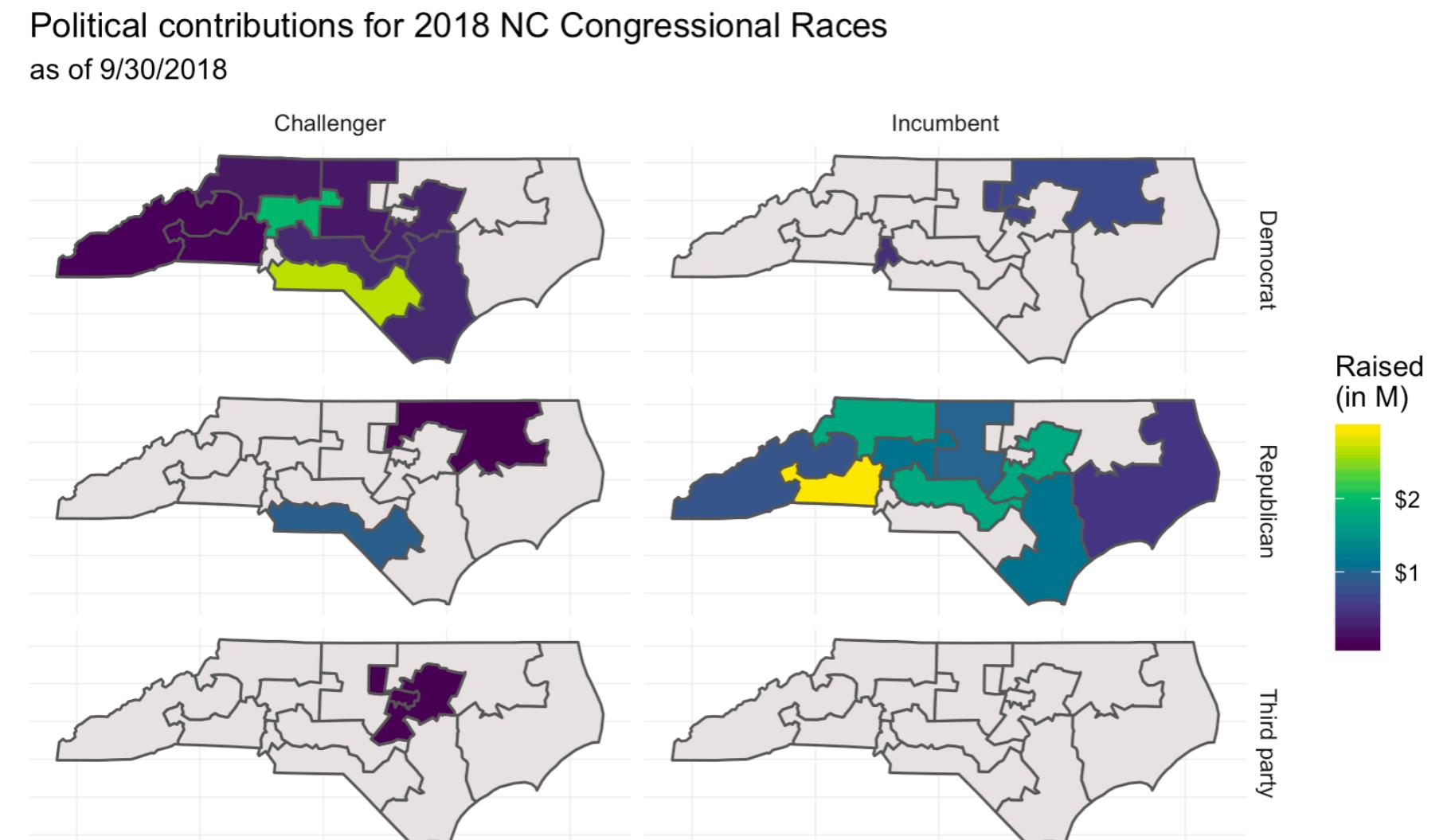
- **Lesson:** “Just enough” string parsing and regular expressions to go from

candidate_info
1 G K Butterfield (D) • Incumbent
2 Roger Allison (R)

to

candidate_name	party	status
G K Butterfield	Democrat	Incumbent
Roger Allison	Republican	Challenger

- **Ex 2:** What other information do we need represented as variables in the data to obtain the desired facets?



leverage
the
ecosystem



- Estimate the difference between the average evaluation score of male and female faculty.

	score	rank	ethnicity	gender	bty_avg
	<dbl>	<chr>	<chr>	<chr>	<dbl>
1	4.7	tenure track	minority	female	5
2	4.1	tenure track	minority	female	5
3	3.9	tenure track	minority	female	5
4	4.8	tenure track	minority	female	5
5	4.6	tenured	not minority	male	3
6	4.3	tenured	not minority	male	3
7	2.8	tenured	not minority	male	3
8	4.1	tenured	not minority	male	3.33
9	3.4	tenured	not minority	male	3.33
10	4.5	tenured	not minority	female	3.17
...
463	4.1	tenure track	minority	female	5.33

```
t.test(evals$score ~ evals$gender)

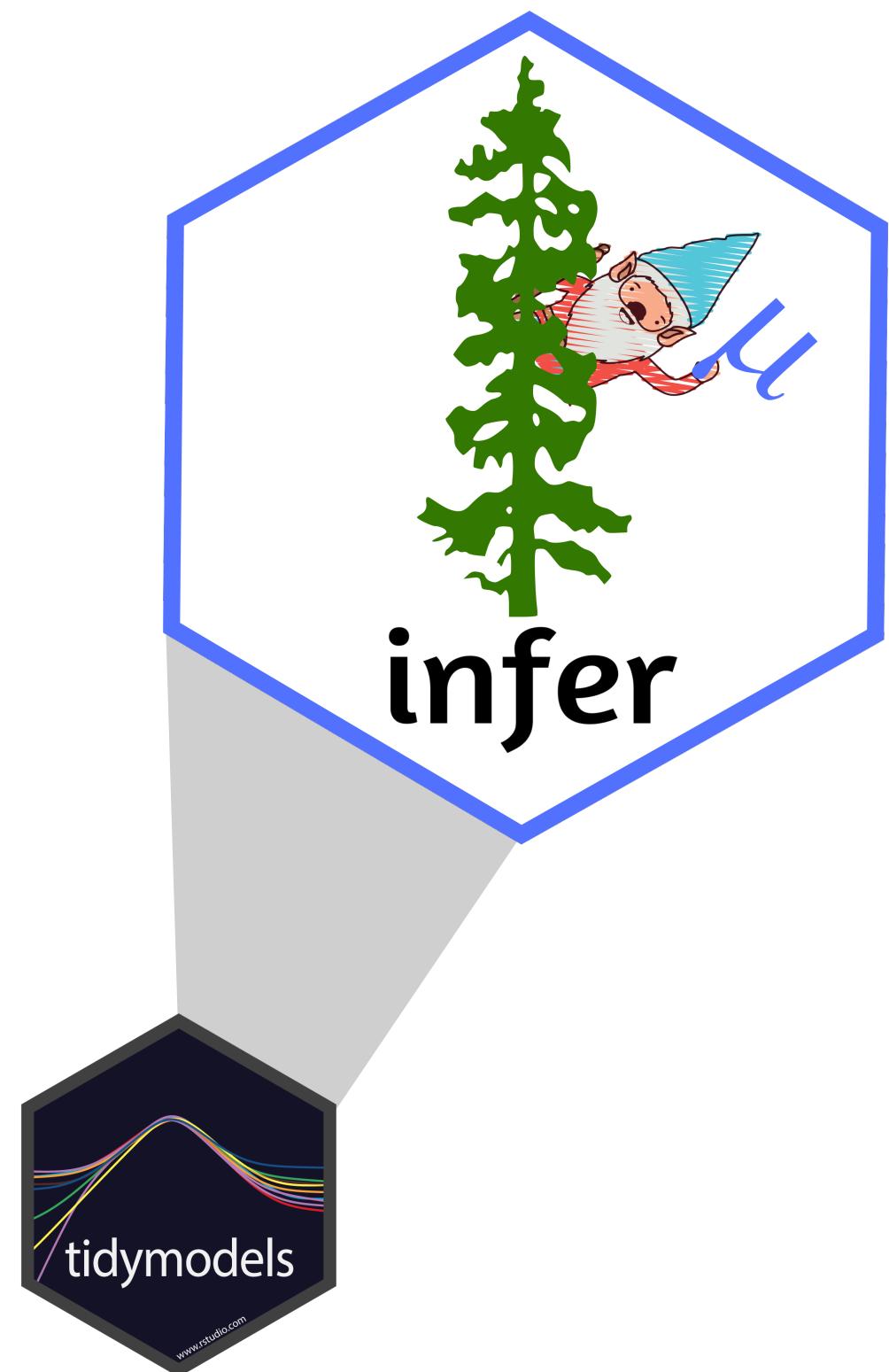
# Welch Two Sample t-test

# data: evals$score by evals$gender
# t = -2.7507, df = 398.7, p-value = 0.006218
# alternative hypothesis: true difference in
# means is not equal to 0
# 95 percent confidence interval:
# -0.24264375 -0.04037194
# sample estimates:
# mean in group female    mean in group male
#                 4.092821                  4.234328
```

```
library(tidyverse)
library(infer)

evals %>%
  specify(score ~ gender) %>%
  generate(reps = 15000,
            type = "bootstrap") %>%
  calculate(stat = "diff in means",
            order = c("male", "female")) %>%
  summarise(
    l = quantile(stat, 0.025),
    u = quantile(stat, 0.975)
  )

#      l      u
# 0.0410 0.243
```



infer

The objective of this package is to perform statistical inference using an expressive statistical grammar that coheres with the `tidyverse` design framework.

Now part of the `tidymodels` suite of modeling packages.

```
library(tidyverse)  
library(infer)
```

start with data

```
evals %>%
```

```
library(tidyverse)  
library(infer)
```

```
evals %>%  
  specify(score ~ gender)
```

specify the model

```
library(tidyverse)  
library(infer)
```

```
evals %>%  
  specify(score ~ gender) %>%  
  generate(reps = 15000, type = "bootstrap")
```

generate bootstrap samples

```
library(tidyverse)  
library(infer)
```

```
evals %>%  
  specify(score ~ gender) %>%  
  generate(reps = 15000, type = "bootstrap") %>%  
  calculate(stat = "diff in means", order = c("male", "female"))
```

calculate sample statistics

```
library(tidyverse)
library(infer)

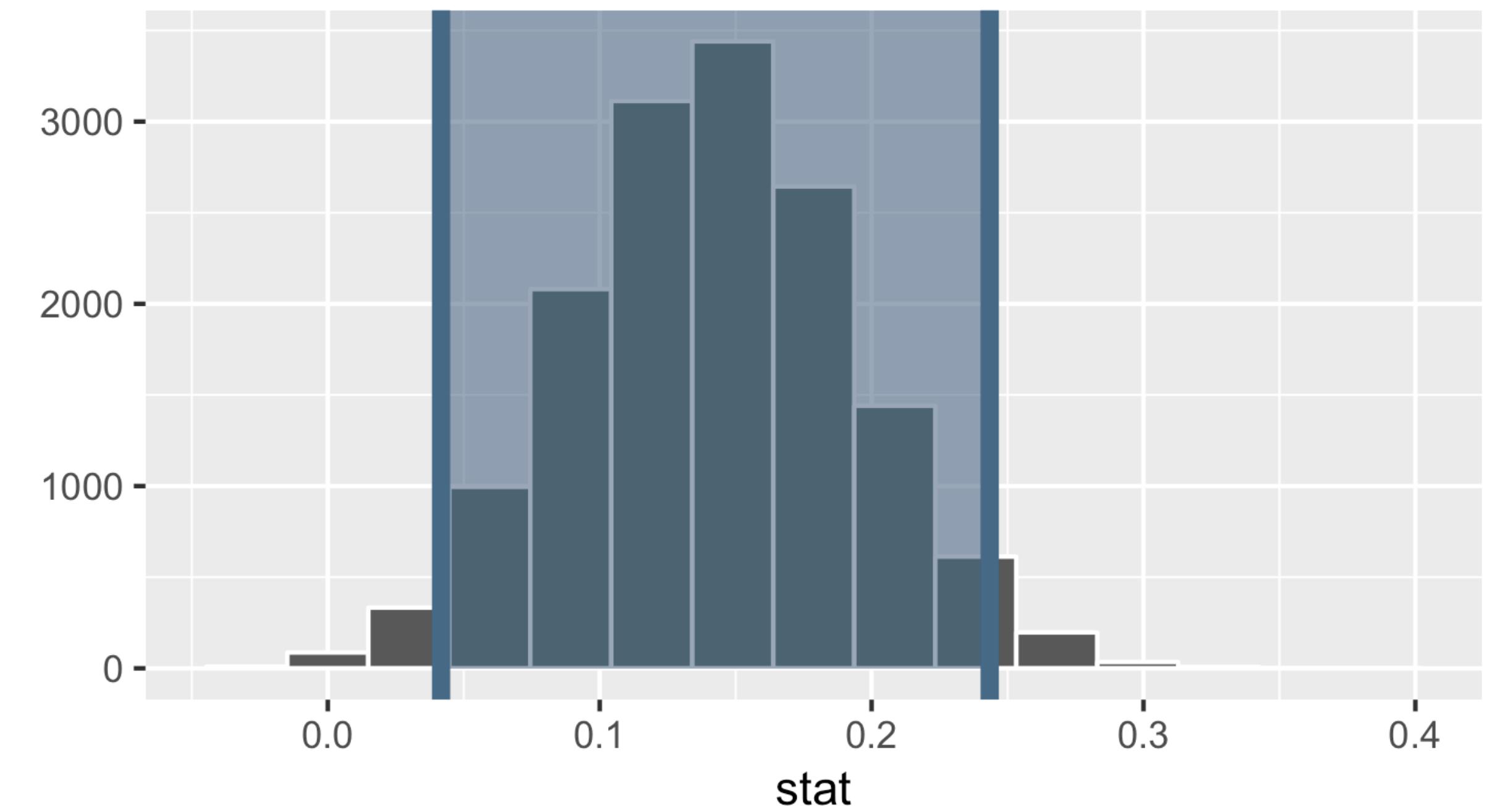
evals %>%
  specify(score ~ gender) %>%
  generate(reps = 15000, type = "bootstrap") %>%
  calculate(stat = "diff in means", order = c("male", "female")) %>%
  summarise(l = quantile(stat, 0.025), u = quantile(stat, 0.975))
```

summarise CI bounds

```
library(tidyverse)
library(infer)

evals %>%
  specify(score ~ gender) %>%
  generate(reps = 15000, type = "bootstrap") %>%
  calculate(stat = "diff in means", order = c("male", "female")) %>%
  summarise(l = quantile(stat, 0.025), u = quantile(stat, 0.975))
```

```
#      l     u
# 0.0410 0.243
```

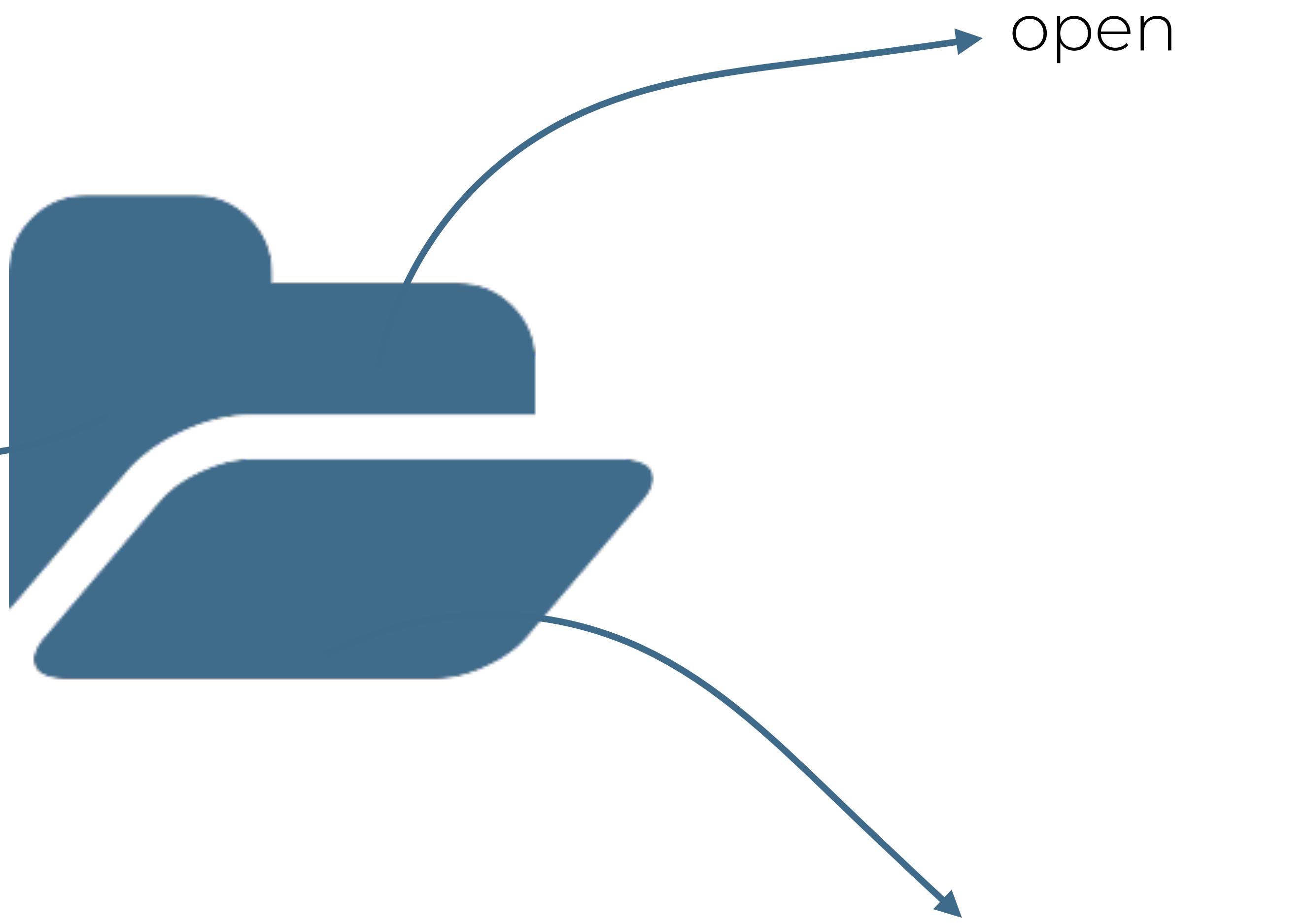


- 1 start with cake
- 2 skip baby steps
- 3 cherish day one
- 4 hide the veggies
- 5 leverage the ecosystem

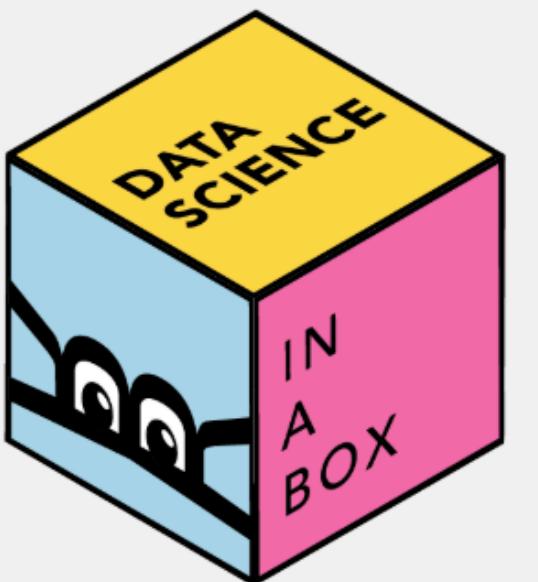
tl;dr

3 goals

scalable



validated

 Search...

Hello #dsbox

Overview

Philosophy

Topics

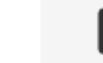
Tech stack

Community

Course content

Infrastructure

Pedagogy

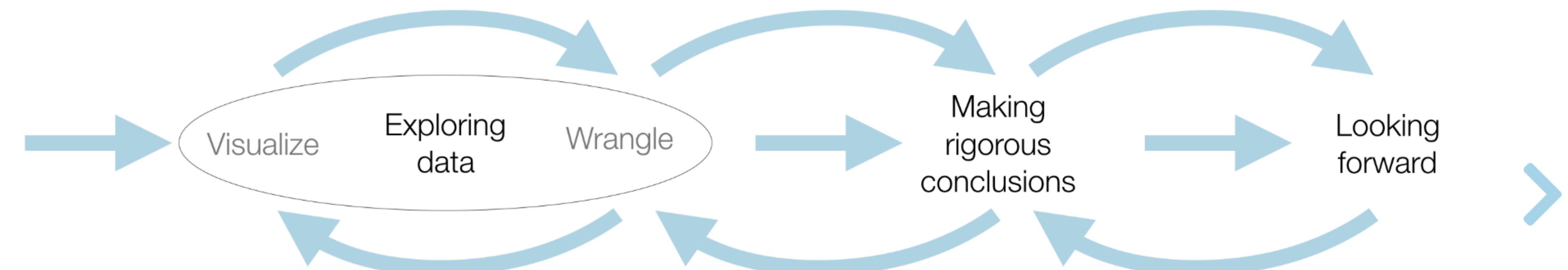


Data Science in a Box > Hello #dsbox > Topics



Topics

The course content is organized in three units:



Unit 1 - Exploring data: This unit focuses on data visualization and data wrangling. Specifically we cover fundamentals of data and data visualization, confounding variables, and Simpson's paradox as well as the concept of tidy data, data import, data cleaning, and data curation. We end the unit with web scraping and introduce the idea of iteration in preparation for the next unit. Also in this unit students are introduced to the toolkit: R, RStudio, R Markdown, Git, GitHub, etc.

Unit 2 - Making rigorous conclusions: In this part we introduce modeling and statistical inference for making data based conclusions. We discuss building, interpreting, and selecting models, visualizing interaction effects, and prediction and model validity. Statistical inference is introduced from a simulation based perspective, and the Central Limit Theorem is discussed very briefly to lay the foundation for future coursework in statistics.

Unit 3 - Looking forward: In the last unit we present a series of modules such as interactive reporting and visualization with Shiny, text analysis, and Bayesian inference. These are independent modules that instructors can choose to include in their introductory data science curriculum depending on how much time they have left in the semester.

open

Retrospective study of 205 open ended student projects

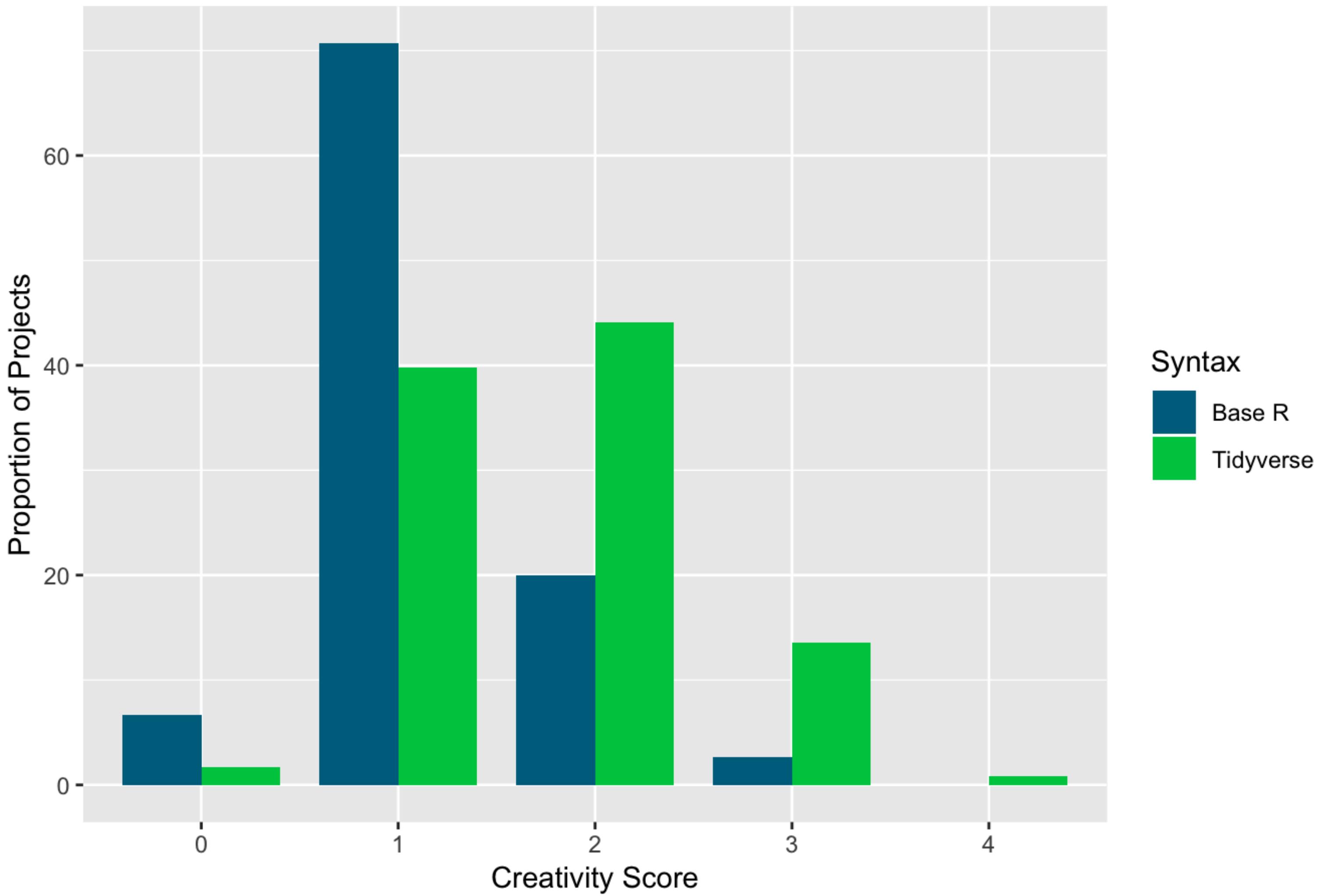
- on **creativity, depth** and the complexity of **multivariate visualizations**
- compared across students who learned R using **base R** syntax vs. **tidyverse**

validated

Creativity:

1. Creation of new variable(s) based on existing variables
2. Transformation of existing variables
3. Existence of a subgroup analysis
4. Use of a subset of the dataset for all steps of the project

Tidyverse Syntax Projects Score Higher on the Creativity Metric on Average

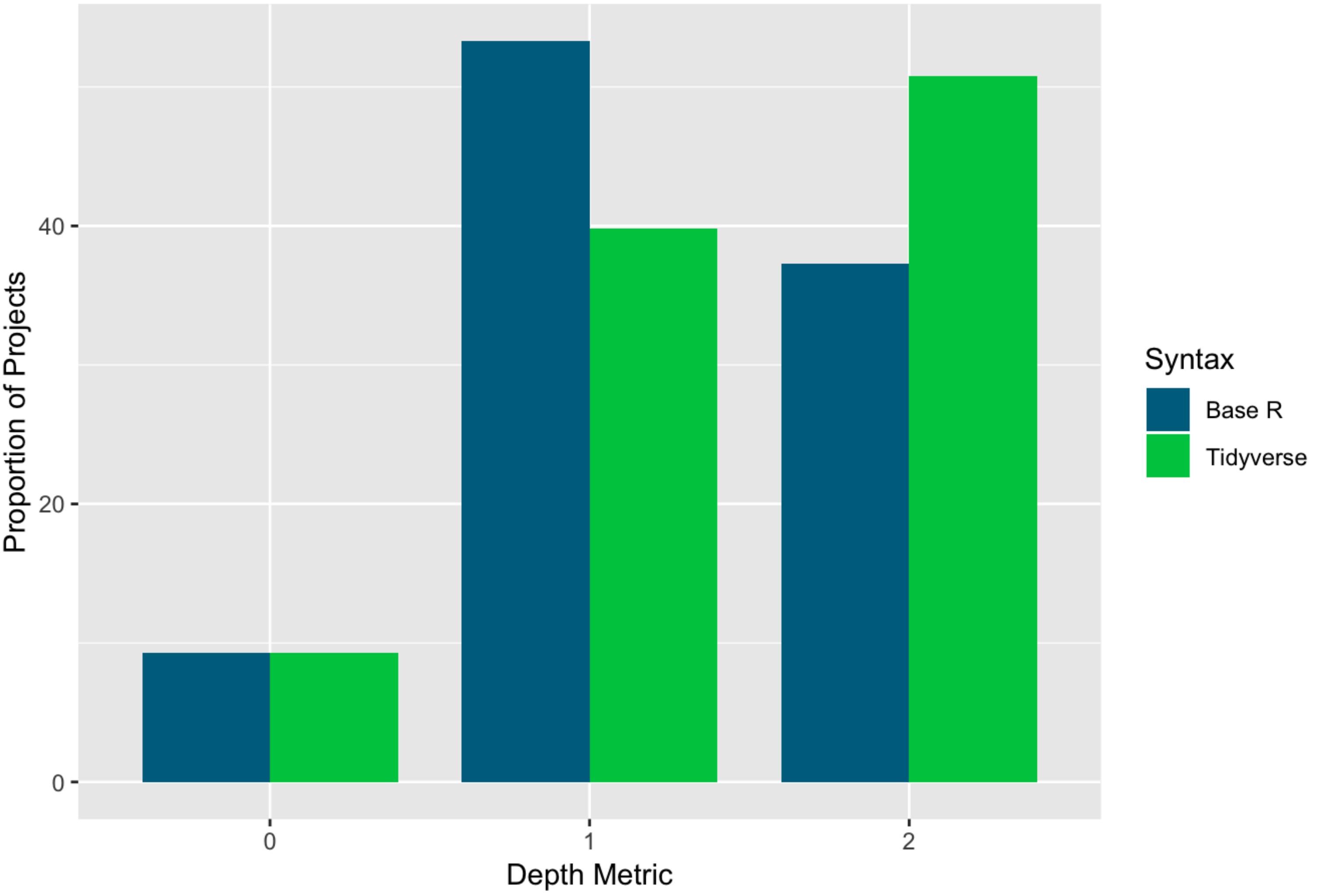


validated

Depth:

1. Presence of consistent theme throughout the project
2. Use of relevant data

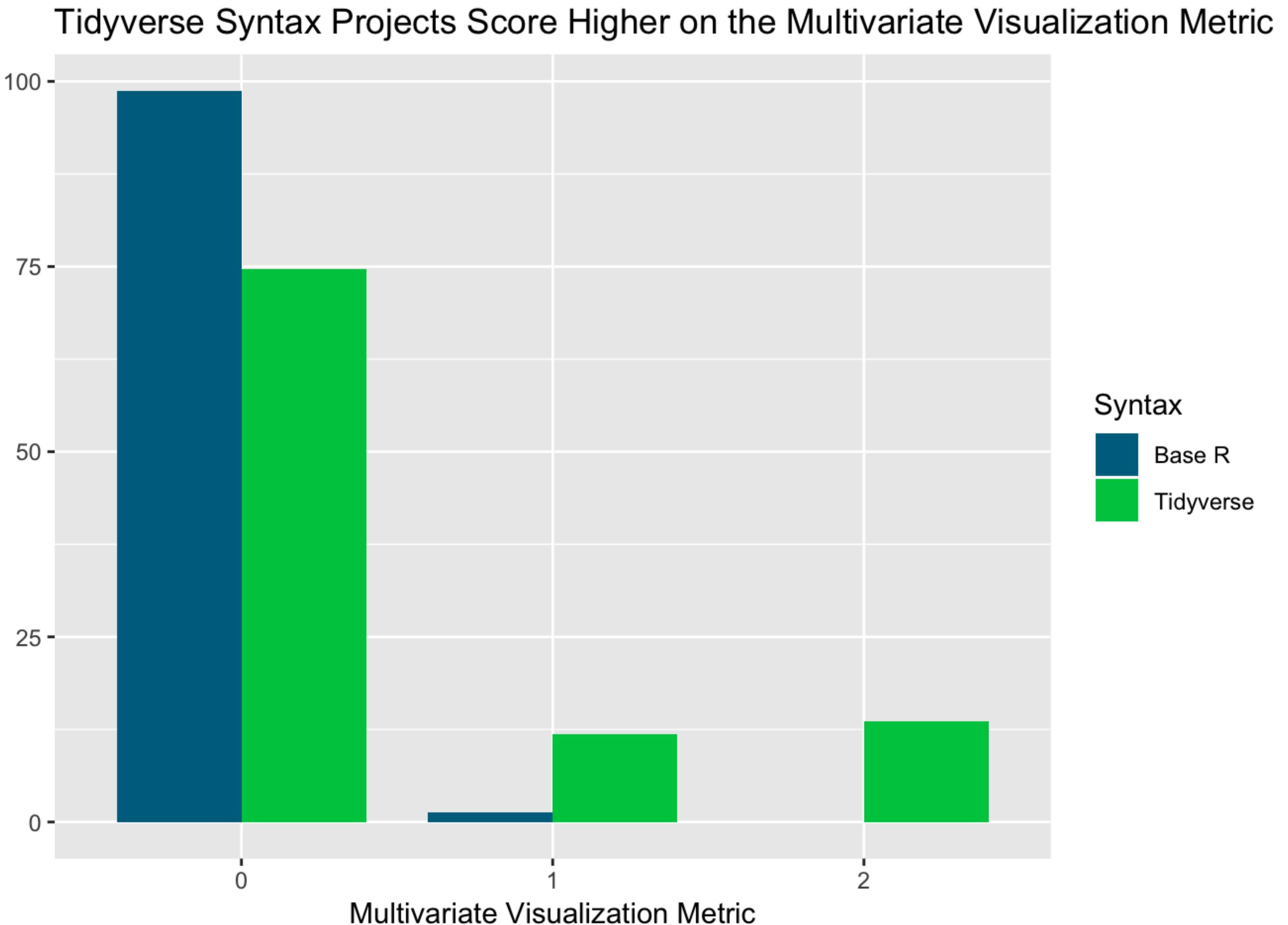
Tidyverse Syntax Projects Score Higher on the Depth Metric on Average



validated

Multivariate visualizations:

1. Presence of a visualization with 3+ variables
2. Interpretation of the multivariate visualization





1. Formative assessments
2. Automated feedback
3. Peer review

Let them eat cake (first)!*

↳ bit.ly/let-eat-cake-cc

</> bit.ly/repo-eat-cake

* You can tell
them all about the
ingredients later!



@minebocek



mine-cetinkaya-rundel



cetinkaya.mine@gmail.com

