Data Analysis: From Data Crawling to Visualization

An Introduction to APIs for Social Scientists and How To (not) Display Data

Andreas Küpfer
Technical University of Darmstadt

andreas.kuepfer@tu-darmstadt.de
andreaskuepfer.github.io
@ankuepfer

Guest Lecture
National University of Kyiv-Mohyla Academy
March 15, 2023

TECHNISCHE UNIVERSITÄT DARMSTADT

About me

- PhD Student in the area of Computational Social Science (Technical University of Darmstadt)
- M.Sc. in Data Science (University of Mannheim)
- Interested in the study of text, video and audio and their interplay using machine learning methods (Data sources: Social Media, Political Speeches or Political Advertisements)

TECHNISCHE UNIVERSITÄT DARMSTADT

What about you?

- Working with data?
- Coding Experience?

Don’t worry… you don’t need anything of the above to follow along.

TECHNISCHE UNIVERSITÄT DARMSTADT

Outline

Why Analyzing Data?

Let's start with a Tweet

A single tweet can be analyzed quite intuitively:

- Author? Date? Language? Length?
- User mentions? Hashtags? Emotions?
- Sentiment?
- User interactions?

Let's scale this up

- Zelenskyy has almost 3000 tweets

Can you still analyze this?

Let's scale this up

- Zelenskyy has almost 3000 tweets
- He is followed by more than 7 million users

Can you still analyze this?

Let's scale this up

- Zelenskyy has almost 3000 tweets
- He is followed by more than 7 million users
- Each follower (and non-follower) interacts with Zelenskyy's tweets

Can you still analyze this?

Let's scale this up

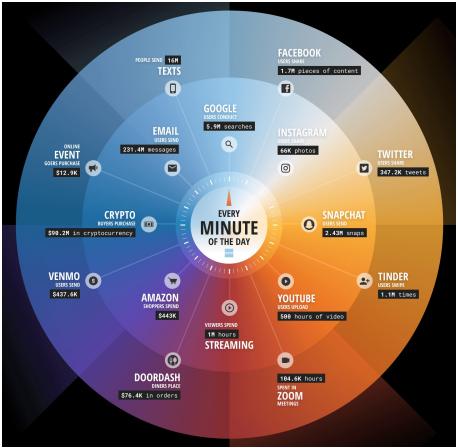
- Zelenskyy has almost 3000 tweets
- He is followed by more than 7 million users
- Each follower (and non-follower) interacts with Zelenskyy's tweets
- Each follower posts own tweets

Can you still analyze this?

Let's scale this up

- This example shows the environment of a single user
- There are over 368 million active users on Twitter
- Most of them are way less active and connected — But: some are even larger
- This is just Twitter! (And we did not consider micro interactions such as clicking on Ads, Scrolling through Twitter, etc. yet)

We're drowning in data



Source: Domo *Data Never Sleeps*

Different Data Sources and How To Crawl Them

Where to get Data?

There are three main approaches:

1. Collecting data by hand
2. Use Application Programming Interfaces (APIs)
3. Crawling data with automated scripts

1) Collecting Data by Hand

- You could simply download News Articles, Social Media Posts, etc. by hand and store it locally

  Disadvantage: Time consuming and often not feasible

- Sometimes Companies offer files containing (aggregated) subsets of their data

  Disadvantage: You cannot decide which part of data should be retrieved and in most cases the data is not offered in realtime

2) Use Application Programming Interfaces (APIs)

- APIs let us retrieve large amounts of data of a platform in realtime
- They are offered and maintained by many companies (such as Twitter or Google)

  Disadvantage: You have to comply with the platforms policy and the platform can control how much and which data you can retrieve

3) Crawling Data with Automated Scripts

- Flexible frameworks are capable of crawling websites without (or very limited) APIs

  Disadvantage: This often is a legal gray area as companies try to impede such crawling — However, for scientific purposes, this may states an option

APIs: An Overview

In this lecture, we focus on APIs. Why?

- They are a great trade-off between collection data by hand and implementing a custom crawling system
  - APIs are often well-documented by the community and/or companies
  - Only basic programming skills required (R or Python)
  - There are out of the box wrapper packages available for most of the APIs

APIs: An Overview

There are many APIs. Let's have a look at some of them[1]:

1. Google News
2. Internet Archive
3. TikTok
4. Youtube
5. Twitter

1. The following slides are mainly based (exception: TikTok API) on the great overview of *APIs for Social Scientists: A collaborative Review* by researchers of the University of Mannheim

1) Google News

With the News API (formerly Google News API), you can get article snippets and news headlines, both up to four years old and real-time, from over 80'000 news sources worldwide.

- **Prerequisites**: API Key request via newsapi.org
- Free version is limited to only 100 requests/day as well as a trunction of the retrived news articles — Business access for $449 per month gives for full access
- **Example use case**:
  - Chrisinger et al. (2021) used the News API to collect and analyze a large dataset of newspaper articles for their study on the discourse on food stamps in the US
  - They collected 13'987 newspaper articles using keyword queries, and ran a structural topic model

2) Internet Archive

The Internet Archive is a non-profit organization that creates and provides a digital library of internet sites and other digital artifacts, including books, audio recordings, videos, images, and software programs. The library currently stores over 588 billion web pages spanning over 25 years. The API provides access to the mementos stored in the Internet Archive.

- **Prerequisites**: No prerequisites!
- The API is freely available without and restrictions or authentication and implemented in the R package archiveRetriever
- **Example use case**:
  - Gavras (2022) accessed newspaper articles from a total of 86 online newspapers from 29 different countries across Europe through the Internet Archive
  - They used this data to research European media discourse

3) TikTok

There are many (unofficial) TikTok APIs — TrakTok as one of them allows you to download videos, comments, user accounts and more. (since Feb 21, 2023 there is also an official API for non-profit researchers in US)

- **Prerequisites**: Depending on the data to be retrieved either a logged in user account or no account at all
- Keep in mind, that such unofficial APIs might stop working if the platform (i.e. TikTok) changes its own data routines
- **Example use case**:
  - Guinaudeau et al. (2021) identified 11'546 TikTok accounts that primarily post about politics
  - Compared to YouTube, TikTok has a higher percentage of video uploads, more viral content, and less dependence on account followers/subscribers for viewership

4) Youtube

The YouTube APIs can be used to access YouTube data such as videos, playlists and comments.

- **Prerequisites**: A google account to log into Google Cloud Platform where you can retrieve access to the APIs
- The YouTube APIs are free of charge and many functions are implmenented in the R package tuber
- **Example use case**:
  - Obadimu (2019) collected comments from eight YouTube channels, either pro- or anti-NATO, using the YouTube Data API
  - They assigned five types of toxicity scores to analyze hateful comments and used word clouds to quantify the count of words in the comments
  - The final dataset included 1'424 pro-NATO videos with 8'276 comments and 3'461 anti-NATO videos with 46'464 comments

5) Twitter (1/2)

- Until recently, Twitter was a role model in terms of free access to data for scientific purposes (up to 10 Mio. tweets per month free of charge)
- But thanks to Elon Musk, this seemed to change as Twitter released a paid version of the API now with a very limited access and suspended the Academic Track

5) Twitter (2/2)

XXX TODO XXX Update with current situation XXX

- **Prerequisites**: API Key request via developer.twitter.com/en
- Free version is limited to only ...
- **Example use case**:
  - Chrisinger et al. (2021) used the News API to collect and analyze a large dataset of newspaper articles for their study on the discourse on food stamps in the US
  - They collected 13,987 newspaper articles using keyword queries, and ran a structural topic model

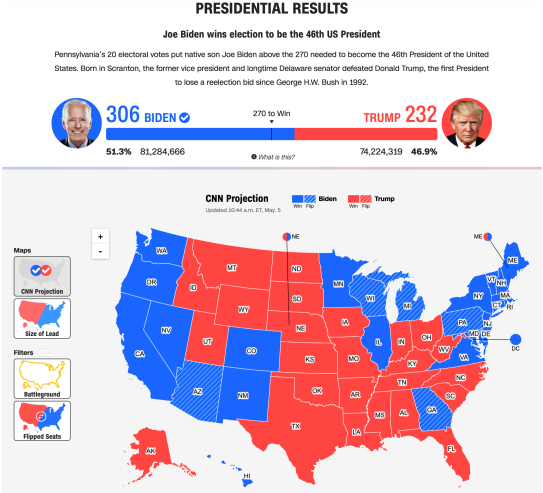
Hands-On: Analyzing the Content Tweets

Tweets from Zelenskyy as a Use Case (Part II)

- Our dataset: All tweets from Zelenskyy gathered via the retired Academic Access API of Twitter
- Using the programming language R to analyze how Zelenskyy uses country flags in his tweets
- Sentiment analysis to connect negative and positive sentiment to specific countries mentioned
- Follow me to the Google Colab Notebook containing all the code necessary to follow along (*Google Account required to execute code*)

Why Visualizing Data?

‘A Picture Is Worth a Thousand Words’



Source: CNN News

Why should we Visualize Data?[2]

- Data visualization allows us to easily understand complex information and patterns
- It helps us to spot trends, clusters, and outliers that may not be apparent in raw data
- Graphics can reveal data features that statistics and models may miss
- Visualization raises questions that stimulate research and suggest new ideas

2. The following three slides are based on *Unwin, A. (2020). Why Is Data Visualization Important? What Is Important in Data Visualization? Harvard Data Science Review, 2(1). https://doi.org/10.1162/99608f92.8ae4d525*

Why should we Visualize Data?

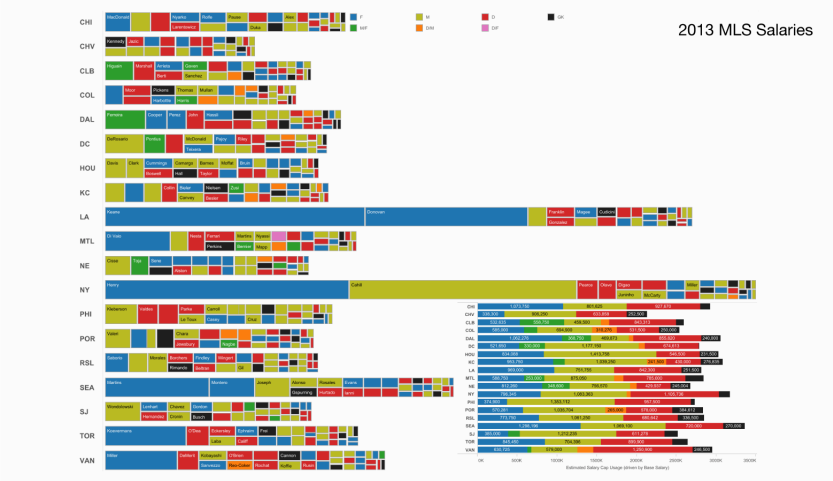
- It is essential for exploratory data analysis and data mining to check data quality
- Visualization helps analysts become familiar with the structure and features of the data before them
- It is useful for data cleaning and identifying unusual groups or patterns
- Evaluation of modeling output is easier when visualizations are used to present results

Why should we Visualize Data?

- Dynamic and interactive graphics are in an exciting stage of development and have much to add
- Graphics on their own are insufficient, they are part of a whole
- The potential synergy of text and graphics can be appreciated by talking through your own graphics, explaining them to others
- When it comes to graphics you have not drawn yourself, the same kinds of questions are still relevant, although they may be more difficult to answer

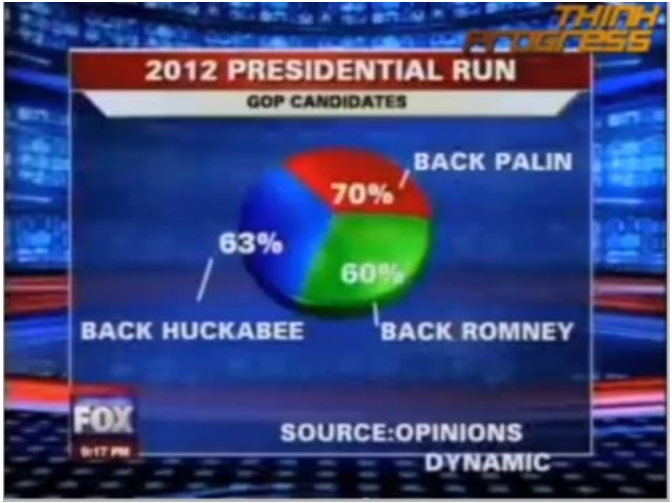
How To Display Data Badly

Some examples...



Source: Toptal UX

Some examples...



Source: Livingqlikview Data Viz

How To Display Data Badly[3]

Wainer, Howard (1984):

Methods for displaying data badly have been developing for many years, and a wide variety of interesting and inventive schemes have emerged. Presented here is a synthesis yielding the 12 most powerful techniques that seem to underlie many of the realizations found in practice. These 12 (the dirty dozen) are identified and illustrated.

3. The following slides are based on *Wainer, H. (1984). How to Display Data Badly. The American Statistician, 38(2), 137–147. https://doi.org/10.2307/2683253* and *Bissantz Blog*

How To Display Data Badly: Rule 1

Show as few data as possible (Minimize the data density)

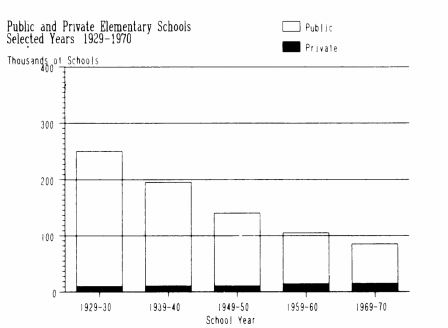
If a graphic threatens to look empty because it contains only a few data values, fill the rest with decorative elements that do not contribute to further clarification of the matter.

Source: Bissantz Blog

How To Display Data Badly: Rule 2

Hide what data you do show

There are several effective ways to do this: Take a striking grid and print the data lightly over it. Or: Slavishly adhere to the rule that zero lines should not be cut off, so that the variation in the data disappears even when it would be interesting.



Source: Wainer, Howard (1984)

How To Display Data Badly: Rule 3

Ignore the visual metaphor altogether

Change the scale on two graphs that you will then compare with each other. Additionally, give the two graphs a different meaning by swapping the meaning of the lines.

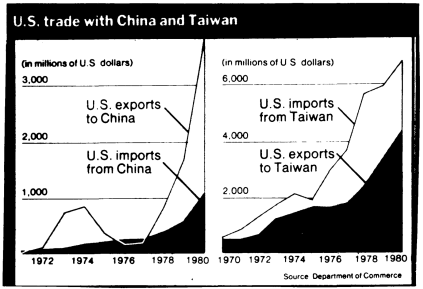
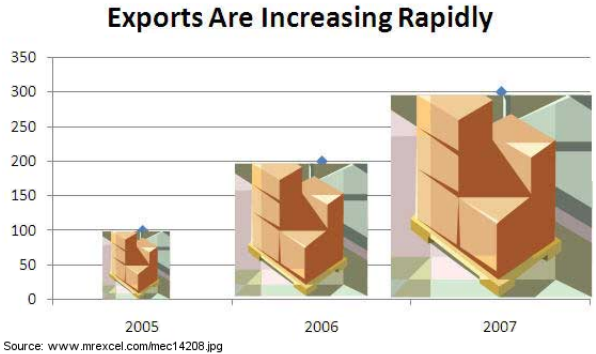


Figure 7. Reversing the metaphor in mid-graph while changing scales on both axes (© June 14, 1981, The New York Times).

Source: Wainer, Howard (1984)

Only order matters

Use length as a criterion of order, but confuse the reader's eye by showing areas. This will even square the most harmless difference.



Source: Bissantz Blog

How To Display Data Badly: Rule 5

Graph data out of context

This works particularly well with time series. A painfully strong drop in a value? Start with the time point after that. Only a slight fluctuation: enlarge the scale, compress the x-axis.

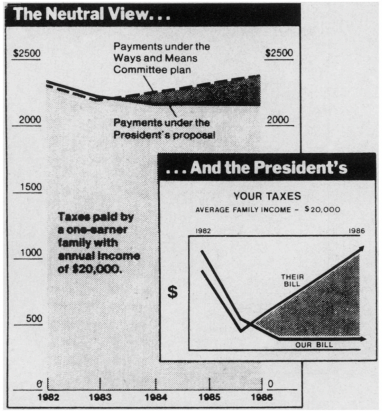


Figure 11. The White House showing neither scale nor context (© 1981, The New York Times, *reprinted with permission).*

Source: Wainer, Howard (1984)

How To Display Data Badly: Rule 6

Change scales in mid-axis

Take two series of different magnitudes, scale each one individually and combine them into one graph.

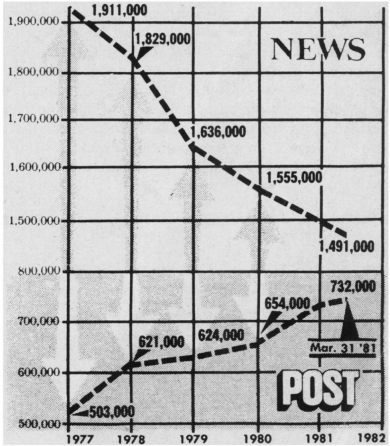


Figure 12. Changing scale in mid-axis to make large differences small (© *1981,* New York Post).

Source: Wainer, Howard (1984)

How To Display Data Badly: Rule 7

Emphasize the trivial (ignore the important)

Splitting the development of salary differences by level of education and gender into two graphs. One shows the development by education level for women, and the other shows it for men. This can cleverly conceal the gender pay gap.

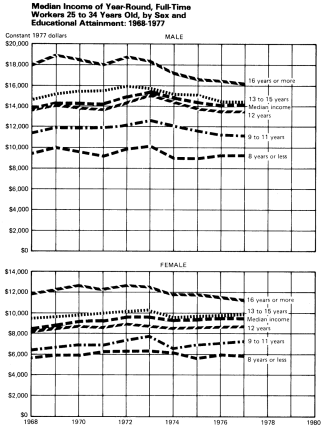


Figure 15. Emphasizing the trivial: Hiding the main effect of sex differences in income through the vertical placement of plots (from SI3).

Source: Wainer, Howard (1984)

Jiggle the baseline

Add small values to large ones and present them as stacked. Even the most intelligent observer will miss how the smaller values have changed.

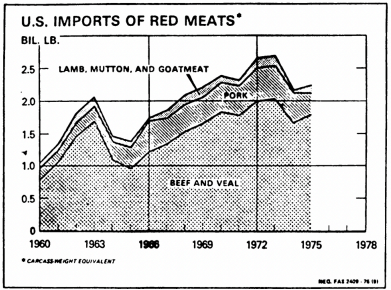


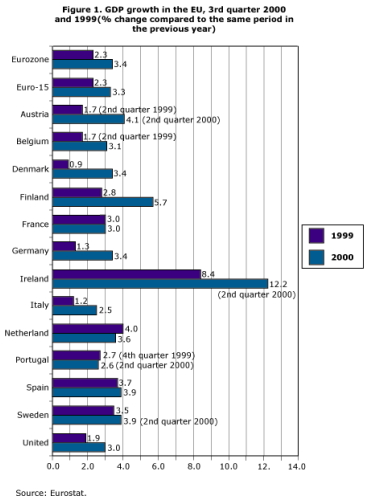
Figure 17. Jiggling the baseline makes comparisons more difficult (from Handbook of Agricultural Charts).

Source: Wainer, Howard (1984)

How To Display Data Badly: Rule 9

Austria first

Generally, sort alphabetically and not by an analytical criterion. Defend this by pointing out that it is easier to find a specific entry that way.

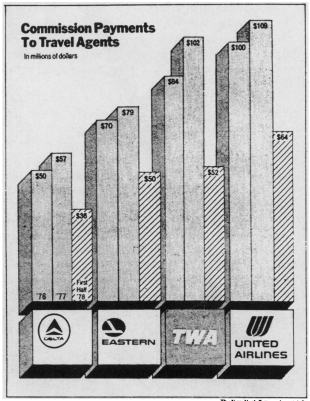


Source: Bissantz Blog

How To Display Data Badly: Rule 10

Label (a) illegibly, (b) incompletely, (c) incorrectly, and (d) ambiguously

Fair cuts lower commission payments to travel agents: Change the meaning of your data by labeling data points with a tiny label



Complex web of discount fares and airlines' telephone delays are raising travel agents' overhead, offsetting revenue gains from higher volume.

Source: Wainer, Howard (1984)

How To Display Data Badly: Rule 11

More is murkier: (a) More decimal places and (b) more dimensions

Use as many decimal places as R provides and avoid rounding.

Table 1. Optimal Selection From a Finite Sequence With Sampling Cost

	b/c = 10.0		*100.0*		*1,000.0*	
N	r^*	$(G_N(r^*) - a)/c$	r^*	$(G_N(r^*) - a)/c$	r^*	$(G_N(r^*) - a)/c$
3	2	.20000	2	2.22500	2	22.47499
4	2	.26333	2	2.88833	2	29.13832
5	2	.32333	3	3.54167	3	35.79166
6	3	.38267	3	4.23767	3	42.78764
7	3	.44600	3	4.90100	3	49.45097
8	3	.50743	4	5.57650	4	56.33005
9	3	.56743	4	6.26025	4	63.20129
10	4	.62948	4	6.92358	4	69.86462

NOTE: $g(Xs + r - 1) = bR(Xs + r - 1) + a$, if $S = s$, and $g(Xs + r - 1) = 0$, otherwise.
Source: Dhariyal and Dudewicz (1981).

Source: Wainer, Howard (1984)

How To Display Data Badly: Rule 12

If it has been done well in the past, think of another way to do it.

Many visualizations that are still useful today have existed for hundreds of years. In this case, do not use a line graph with one line per data type, but rather use these clear bar plots.

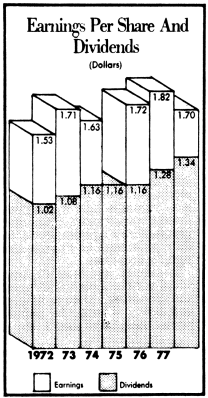


Figure 23. An extra dimension confuses even the grapher (© 1979, The Washington Post).

Source: Wainer, Howard (1984)

Data Visualization using R and ‘ggplot2‘

Let's Learn How to Visualize Data!

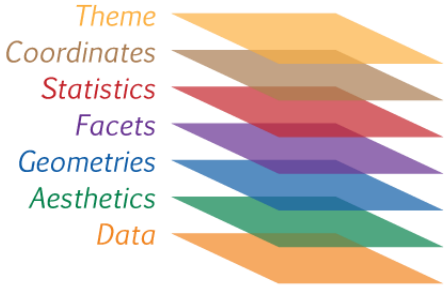
Source: Illustration by allison_horst

What is ‘ggplot2‘?

- ‘ggplot2‘ is a data visualization package for the R programming language that is widely used for creating high-quality graphics
- It is built on the Grammar of Graphics, which is a framework for constructing graphics that allows users to specify the components of a plot and how they should be mapped to the data
- ‘ggplot2‘ provides a wide range of customizable plotting options and supports a variety of plot types, including scatter plots, line plots, bar charts, histograms, and more
- It is designed to make it easy for users to create complex visualizations and to customize their plots using a consistent set of syntax and principles

Structure of ‘ggplot2‘

- Implements Wilkinson’s The .gold[Grammar of Graphics (1999)]
- Each plot is a unique combination of data, aesthetic mappings, geometric objects, etc.
- Individual elements are called layers



Source: Holiday notes2 - Grammar of graphics

‘ggplot2‘: Example in Theory (1/3)

- Suppose we have a dataset of political candidates’ campaign spending and their corresponding vote share in a hypothetical election
- We want to create a scatter plot that shows the relationship between campaign spending and vote share

‘ggplot2‘: Example in Theory (2/3)

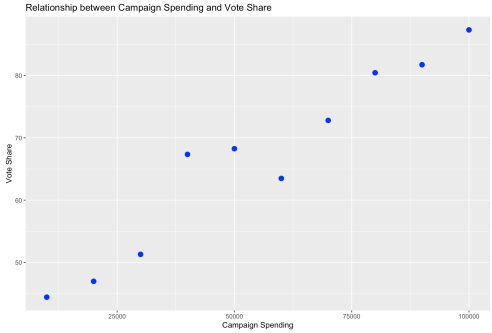
- To create this plot using ‘ggplot2‘ and the Grammar of Graphics, we would first load the dataset into R
- Next, we would use the ‘ggplot()‘ function in ggplot2 to create the plot
- We would specify the data source and aesthetic mappings, which are the campaign spending variable mapped to the x-axis and the vote share variable mapped to the y-axis
- We would also add a title and labels for the x-axis and y-axis
- After specifying the data and aesthetic mappings, we would add a layer of points to represent each candidate’s data point on the plot

‘ggplot2‘: Example in Theory (3/3)

- We could customize the points by changing their color, size, and shape to make the plot more visually appealing
- Finally, we could add additional layers or formatting options to the plot, such as a trend line or confidence intervals, to analyze the relationship between campaign spending and vote share in more detail

‘ggplot2‘: Example in Code (3/3)

```r
# Load the data
data <- read.csv("campaign_spending.csv")
# Create the plot
ggplot(data,
       aes(x = campaign_spending, y = vote_share)) +
  geom_point(color = "blue", size = 3) +
  ggtitle("Relationship between Campaign Spending and Vote Share") +
  xlab("Campaign Spending") +
  ylab("Vote Share")
```



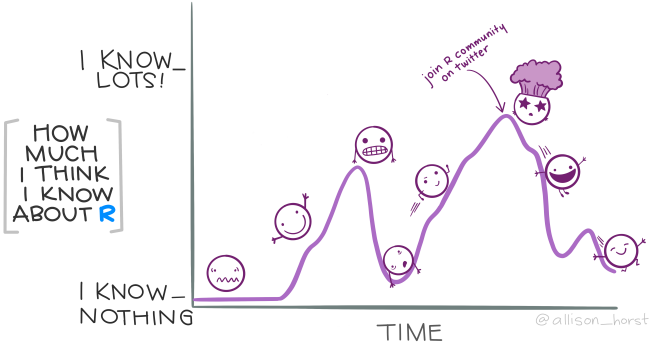
Hands-On: Visualizing the Content of Tweets

Tweets from Zelenskyy as a Use Case

- Our dataset: All tweets from Zelenskyy gathered via the retired Academic Access API of Twitter
- We preprocessed our data to have the number of flag mentions per tweet as well as the sentiment of a tweet
- Follow me to the Google Colab Notebook containing all the code necessary to follow along (*Google Account required to execute code*)

Outlook

Don’t give up!


I KNOW_ LOTS!
HOW MUCH I THINK I KNOW ABOUT R
I KNOW_ NOTHING
join R community on twitter
TIME
@allison_horst

Some Helpful Resources

There are many learning offerings freely available on the web. Below you find some recommendations:

- DataQuest interactive tutorials: Introduction to Data Analysis in R
- R for Data Science by Hadley Wickham and Garrett Grolemund (2022)
- How to learn R? by Ozlem Tuncel (2022)