Data Wrangling and Visualization

WEEK 1

Course Introduction

PROF. JACK REILLY

Welcome!

- This is **Data Wrangling and Visualization**
- I am your instructor, Jack Reilly
  - I'm a professor here at Maxwell in the Public Administration and International Affairs department
  - I hold a PhD in Political Science, with an emphasis in American politics & policy and quantitative methods
  - I do research on American politics, political behavior, and policy preferences, and used to be a stats consultant
- When not academic-ing, I am all about the outdoors of upstate New York

Now you!

- Name
  - *(and pronouns, if you wish)*
- Program of study
- Year
- Academic and career interests
- Fun fact *(if you wish)*

Our little secret

- First time this class has been offered in PAIA
  - You are my test subjects!
  - *I'm quite curious as to what you are all looking to get out of it*
- One more time around:
  - What interests you about this class?
  - What prior experience do you have with data? Statistics? R? programming?
  - Why have you taken it?
  - What are you hoping to get out of it?

Agenda

Welcome

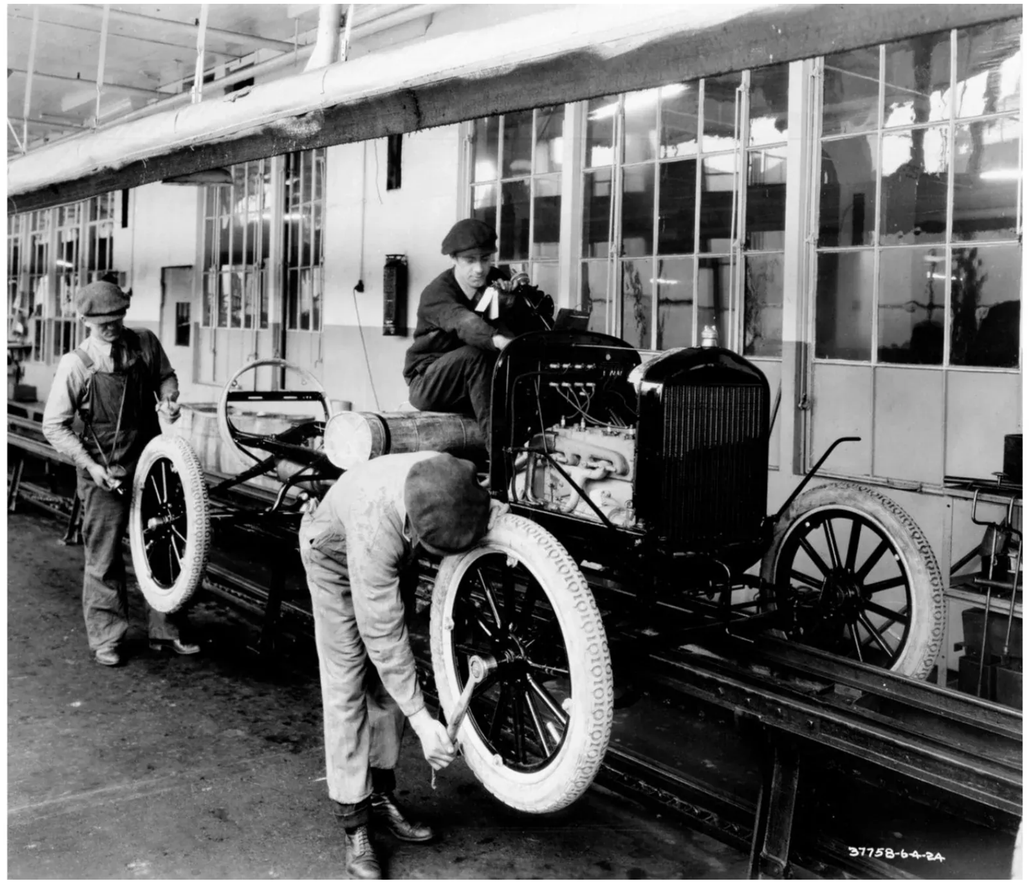
Structure and Goals of the Course

An Introduction to R

What's the purpose of this class?

I'M TRYING TO SOLVE SOME RECURRING PROBLEMS

- Almost all fields and careers are now, to a first approximation, becoming computational



Motivation (for the course)

I'M TRYING TO SOLVE SOME RECURRING PROBLEMS

- Almost all fields and careers are now, to a first approximation, computational
  - Social science fields, too!
- **But, in non-engineering fields, we often don't train people in software or computation**
  - We just expect them to pick it up along the way

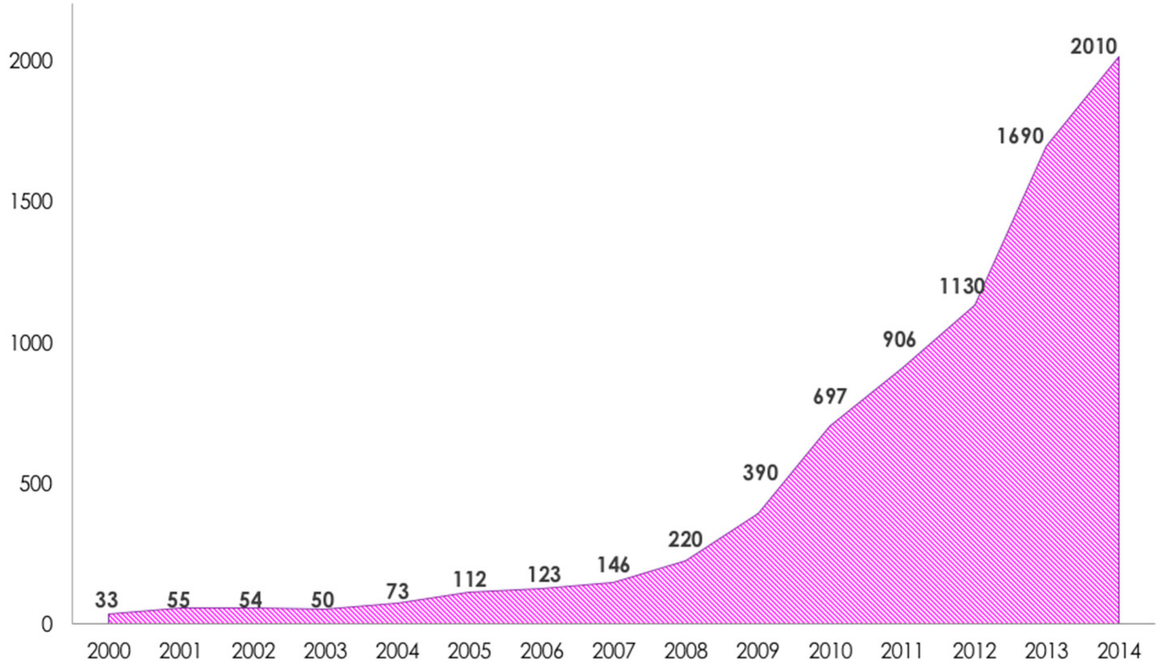


Figure 1. Number of papers published in the period 2000–2014 and containing an explicit reference to "computational social science" in the title (source: Google Scholar).

Motivation (for the course)

I'M TRYING TO SOLVE SOME RECURRING PROBLEMS

- A common "solution" is to require Introduction to Statistics
  - but, Statistics ≠ Computation
  - This is the solution for the ***quantitative*** revolution
  - We are now in the ***computational*** revolution
- Statistics: means, medians, modes; distributions, probability, linear models, estimates . . .
- Computation: data management, wrangling, and workflow, file and data storage, scripting and programming, knowledge of hardware, system design
  - Anecdotally: strangely, computational knowledge has actually *decreased* among students in the last 10 years

Motivation (for the course)

I'M TRYING TO SOLVE SOME RECURRING PROBLEMS

- Furthermore:
  - Data management & access
  - Data wrangling
  - Data visualization
  - Scripting & programming
- Usually take up **90% of the time** it takes you to do a quantitative data project
  - Statistical analysis takes up the **final 10%**
- **This class tries to help you with that 90% that Intro Statistics overlooks**

Motivation (for you)

I'M TRYING TO SOLVE SOME OTHER RECURRING PROBLEMS, TOO

- Data classes get a bad rep.
  - Programming classes get a worse rep.
    - Mathematics classes get an even worse rep.
- My goal is for it to **not be this way.**

I promise that you can succeed in this class.

Important Skills to Remember

- Everyone - yes, everyone - gets (lots of) errors.

> It's easy when you start out programming to get really frustrated and think, "Oh it's me, I'm really stupid," or, "I'm not made out to program." But, that is absolutely not the case. Everyone gets frustrated. I still get frustrated occasionally when writing R code. It's just a natural part of programming. So, it happens to everyone and gets less and less over time. Don't blame yourself. Just take a break, do something fun, and then come back and try again later."

- Hadley Wickham, lead programmer of RStudio *(and several of the packages you'll use in this class)*

Important Skills to Remember

- Everyone - yes, everyone - gets errors.
- Computers are very **very** fast, and very **very** stupid.

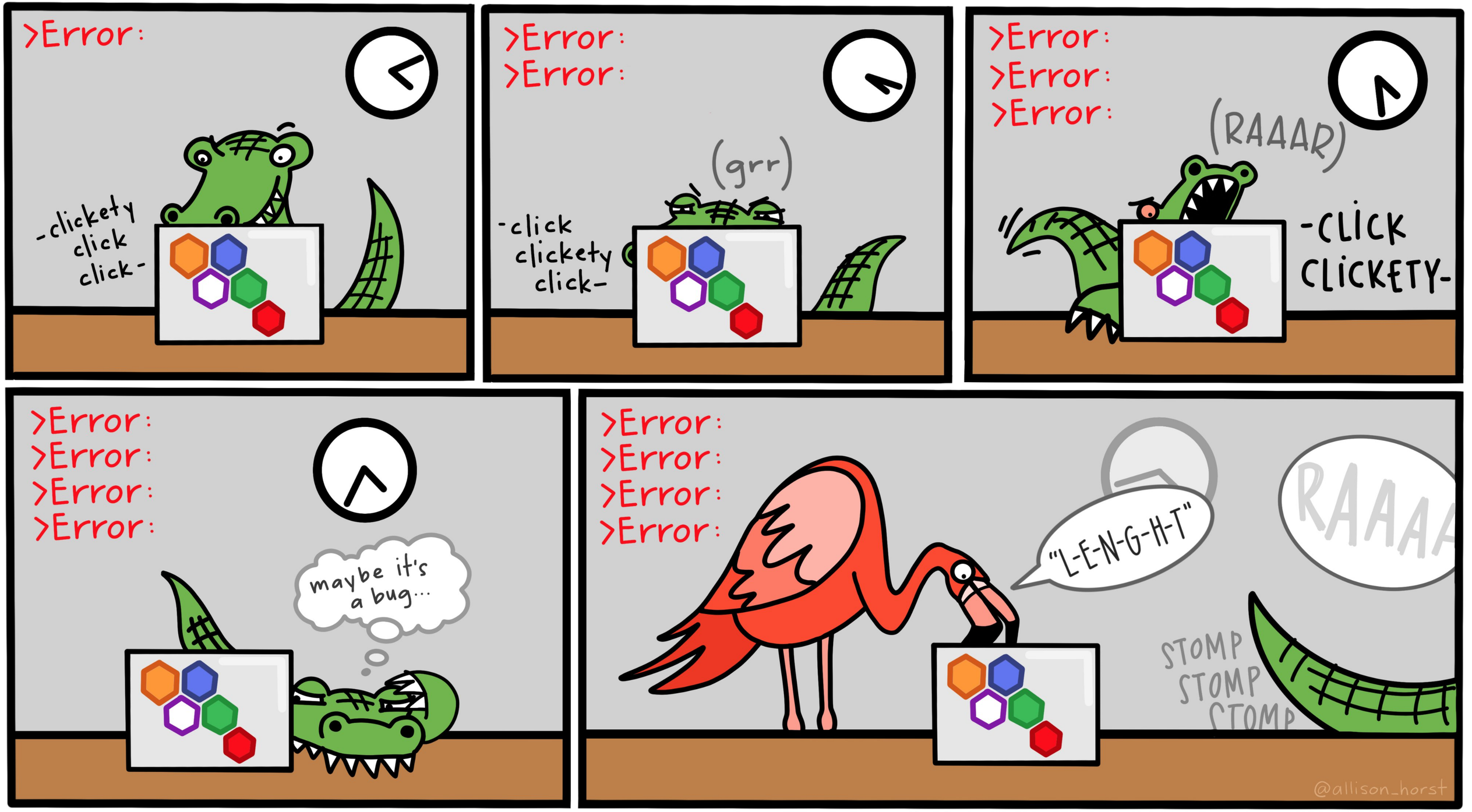
- A line of code that R might be able to run:

```
data <- read.csv(“supercalifragilisticexpialidocious.csv”)
```

- A line of code that R will throw up its hands and say “huh/” with an ugly red error:

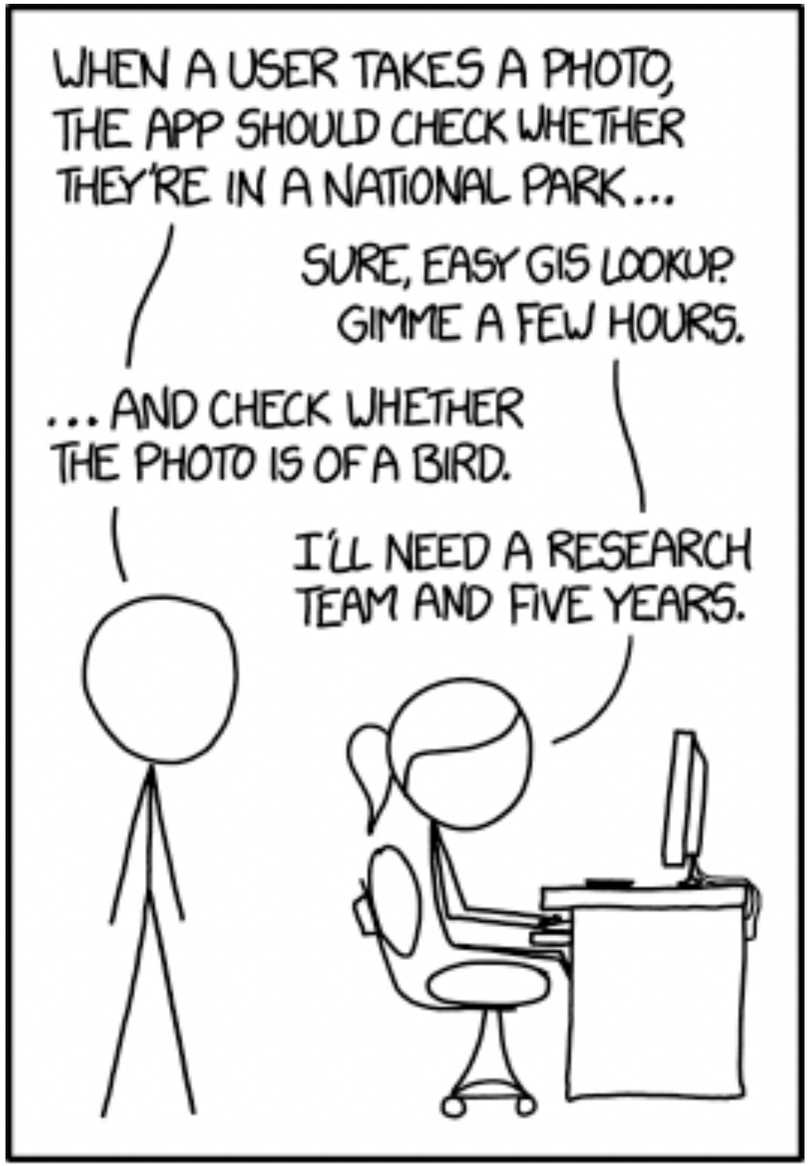
```
data <- read.cvs(“supercalifragilisticexpialidocious.csv”)
```

- Do you see the critical difference?


>Error:
-clickety click click-
>Error:
>Error:
(grr)
-click clickety click-
>Error:
>Error:
>Error:
(RAAAR)
-CLICK CLICKETY-
>Error:
>Error:
>Error:
>Error:
maybe it's a bug...
>Error:
>Error:
>Error:
>Error:
"L-E-N-G-H-T"
RAAAA
STOMP STOMP STOMP
@allison_horst

Important Skills to Remember

- Everyone - yes, everyone - gets errors.
- Computers are very **very** fast, and very **very** stupid.
- It’s actually quite hard to know ahead of time what is very difficult and what is very easy.



https://xkcd.com/1425

Important Skills to Remember

- Everyone - yes, everyone - gets errors.
- Computers are very **very** fast, and very **very** stupid.
- It’s actually quite hard to know ahead of time what is very difficult and what is very easy.

This is a **learning process** - give yourself time, patience, and just stick to it. It’ll come.

Course Structure

- Class is really about three things:
  - Data Management
  - Statistical Programming (in R)
  - Data Visualization
- At an *introductory* level
  - I'm not presuming you know **anything** (yet)

Syllabus

jacklreilly.github.io/dwv_f25

Data Wrangling and Visualization

PAI 400/600 - Syracuse University

AUTHOR
Prof. Jack Reilly

PUBLISHED
F2025

Hello and welcome to the course website for PAI 400/600, “Data Wrangling and Visualization.” I am your instructor, Professor Jack Reilly. You can find more about me on my website, but in short:

Syllabus Highlights

- Overall Course Structure:
  - We meet **twice** a week.
    - Tuesday: lecture/core topic
    - Thursday: workshop
  - There is an assignment due **every** week, by class on Thursday
    - We go over it in class!
- After the first week or two, we'll settle into a cadence
  - At the beginning, liftoff is a little complicated

Schedule

- Two major sequential units:
  - **Fundamentals** (tools, basic data manipulation, scripting and programming, basic visualization, etc)
  - **Types of Data** (higher level visualization, network data, geographic data, etc)
- Sprinkled throughout:
  - *Philosophy* of data analysis
  - Ordering your data life

Books

- Two required:
  - **Data Management:** Weidmann, Nils. Data Management for Social Scientists. Open access: https://doi.org/10.1017/9781108990424
  - **Data Visualization:** Healy, Kieran. Data Visualization: A Practical Introduction. Open access: https://socviz.co
- A **programming** book is recommended:
  - Braun & Murdoch, A First Course in Statistical Programming, 3rd Edition
  - Freeman & Ross, Programming Skills for Data Science
  - Hadley Wickham, Garrett Grolemund, and Mine Çetinkaya-Rundel, R for Data Science: Import, Tidy, Transform, Visualize, and Model Data, 2nd ed. https://r4ds.hadley.nz

Computation & Technicals

- You'll need a computer (not a tablet) - Mac, Windows, Linux are all fine
  - We'll be installing lots of (free) software to use!
- Main course resources:
  - Website: jacklreilly.github.io/dwv_f25
  - Blackboard: http://blackboard.syr.edu/
  - Others as announced (esp. data sources)

Requirements

- Regular course participation and attendance (10%)
- Weekly Assignments (30%)
  - Largely an *effort based* grading system - you do the work, you get the check
- Core Exam (30%)
  - Points based. In class and take-home
- Final Project (30%)
  - Points based

Note: graduate students have higher expectations in the final project

Odds and Ends

- How to contact me: email is best (jlreilly@syr.edu) - I'll get back to you by the end of the next day (at latest!) and usually faster
- Office hours:
  - 11-Noon (in person) Tuesday and Thursday, drop in
  - 1-3*ish* (over zoom) Friday, by appointment
- Course community: we'll spend lots of time together working through problems; much of this class will not be lecture
  - Feel free to help each other on work! (There's a reason homework sets are mostly effort/check based)
  - Remember, though: it is your job to make sure you're actually learning

AI & LLMs

- Who knows what an LLM is? How would you explain it?

AI & LLMs

- Who knows what an LLM is? How would you explain it?
  - Here's one definition: a **computational bullshit machine.**
  - New Oxford American Dictionary:

bull·shit | 'bo͝ol,SHit | *vulgar slang*

noun

stupid or untrue talk or writing; nonsense.

verb (**bullshits, bullshitting, bullshitted**) *[with object]*

talk nonsense to (someone), typically to be misleading or deceptive.

ORIGIN

early 20th century: from bull[3] + shit.

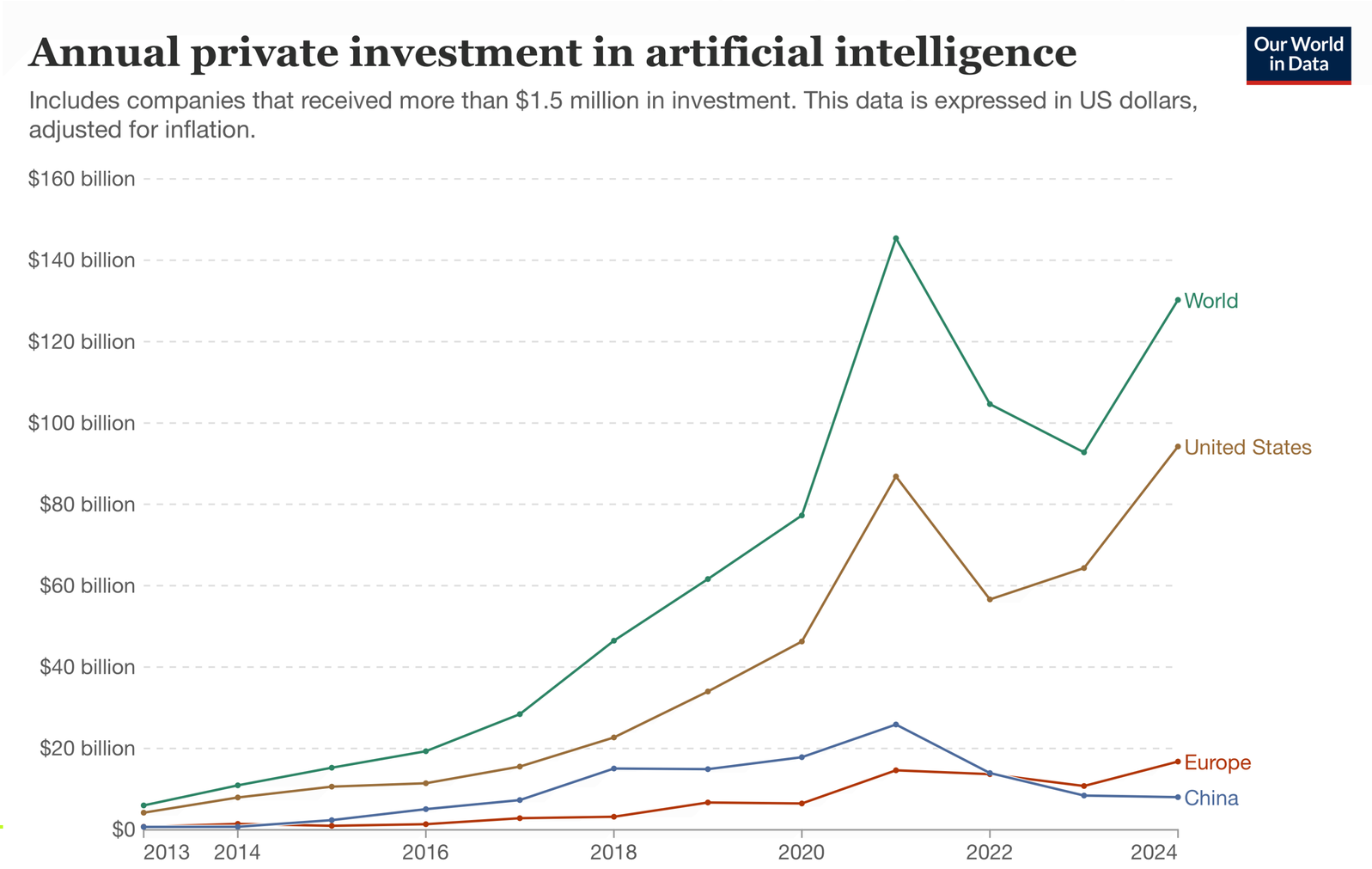
We can do better

- Harry Frankfurt, 2005: **bullshit** is “**language produced without regard for truth”**
  - Note: this does ***not*** mean that AI produced language may not accidentally be true!
- Hicks, Humphries, and Slater, 2024: “The problem here isn’t that large language models hallucinate, lie, or misrepresent the world in some way. It’s that they are not designed to represent the world at all; instead, they are designed to convey convincing lines of text.”
  - A large language model is a **probabilistic model** whose primary function is to produce the next token – usually a word – that is most likely to make sense, based upon some prompt. It does not care if this token represents truth, reality, or a functional line of code

FRANKFURT, 2005. “ON BULLSHIT”
HICKS, HUMPHRIES, AND SLATER, 2024. “CHATGPT IS BULLSHIT.”

So what do we do with these things?

- Well, basically, we invest in and use them like they're going out of style



So how should we use them?

- Formal course policy: you can use them in everything that's not a test. (with attribution)
- **However**, some advice:
  - **It's a really bad idea to rely on them too much right now.**
    - You're at the beginning of trying to understand data workflows, programming, and the like.
    - Many of your assignments are created to be simple *enough* for you to figure out on your own, but this also means ChatGPT (Claude, etc) can do them without too much trouble.
  - **Learning how to do the basics is important for being able to use tools like AI effectively in the future.**
    - You need to know something about writing code in order to use them correctly

AI & LLMs

- So, why the double-talk? Why not just ban them?
  - Because, basically: learning to use AI effectively is important.
    - They can be *super useful* if you use them correctly (ask them questions on how to do things, give them error messages and ask for troubleshooting tips, etc)
    - They can also be super *damaging* to your learning if you use them poorly (asking them to do the assignment for you)

Now, it's R Time!