Course Introduction

Data Wrangling and Visualization

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 moonlight as a stats consultant for tech companies
- For fun: when I was a young person, I played basketball, ultimate (frisbee), and hiked a lot. Now, as a not-quite-young-so person, I play disc golf, still hike a lot, and advise the coaching staff of the New York Knicks from the comfort of my couch

Now you!

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

And one more time around

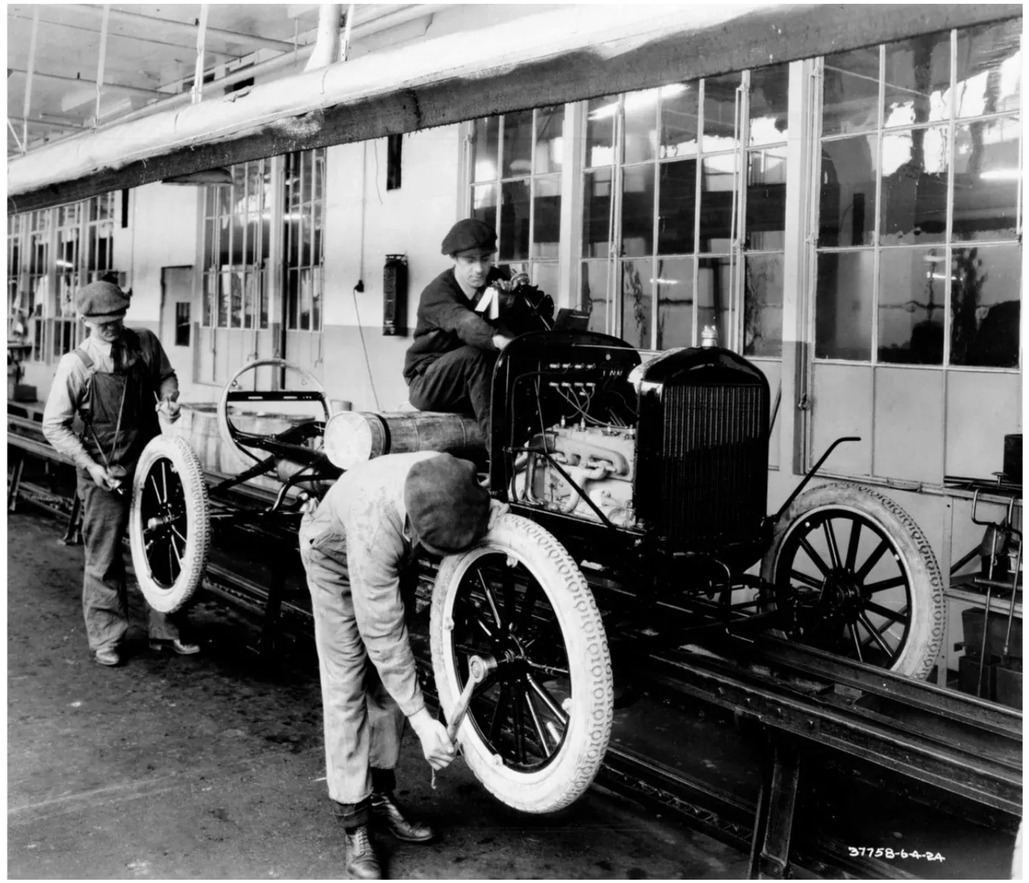
- What interests you about this class?
- What prior experience do you have with data? Statistics? R? programming?
- Why have you taken it?
- What are you hoping to get out of it?

Agenda

- **Welcome**
- Structure and Goals of the Course
- An Introduction to R

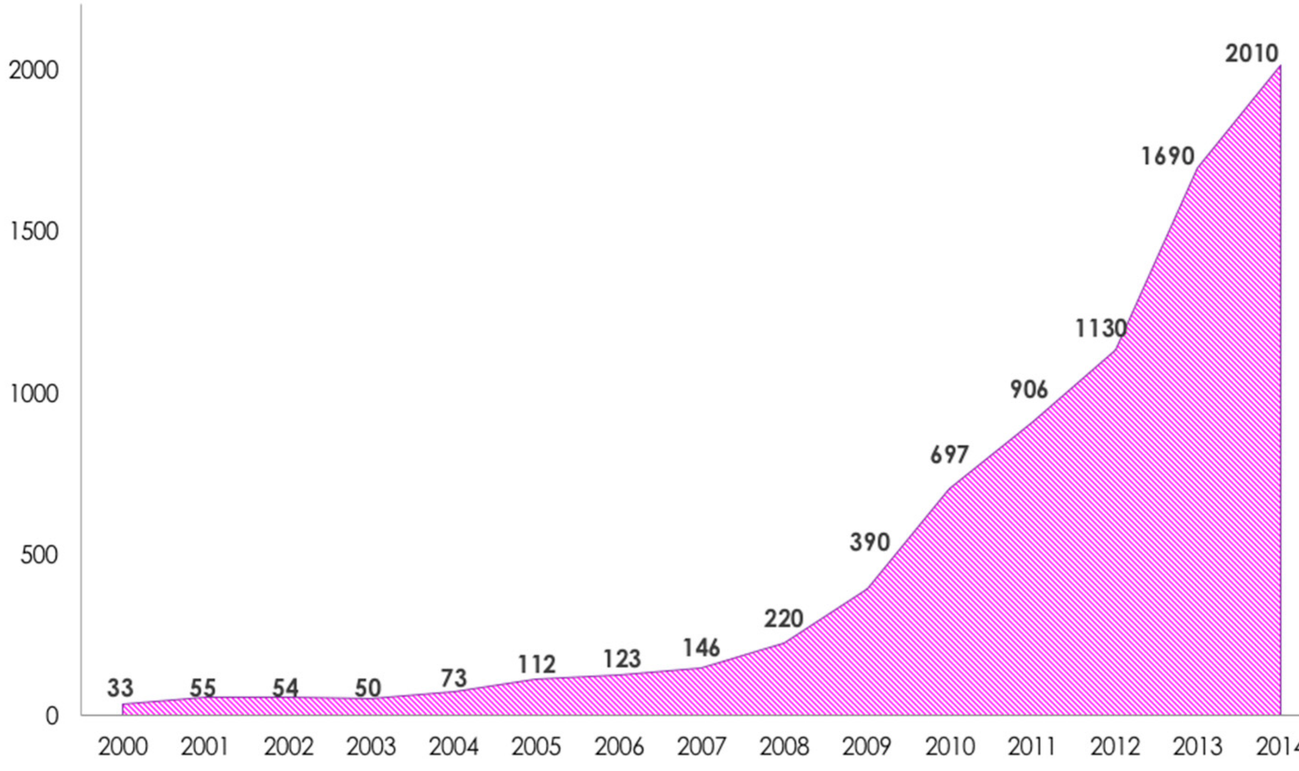
What's the purpose of this class? I'm trying to solve some recurring problems

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



Motivation (for the course)

- 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



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)

- A common “solution” is to require Introduction to Statistics. **But, Statistics ≠ Computation**
  - Statistics 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)

- 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 as well
- 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) coding 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. *If Hadley Wickham can get errors, you can get errors. It's OK.*

- Computers are very **very** fast, and very **very** stupid.

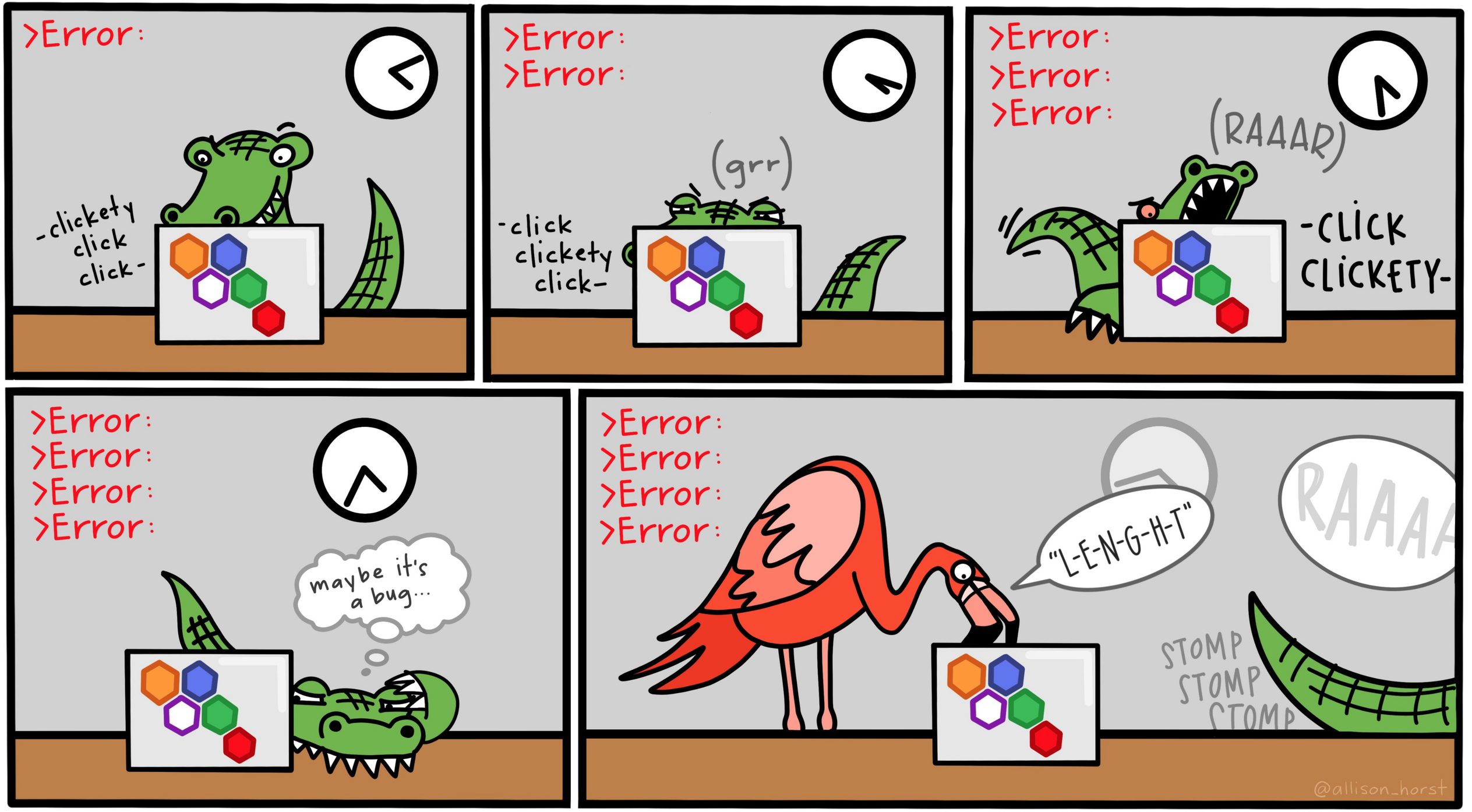
- A line of code that R might be able to run:

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

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

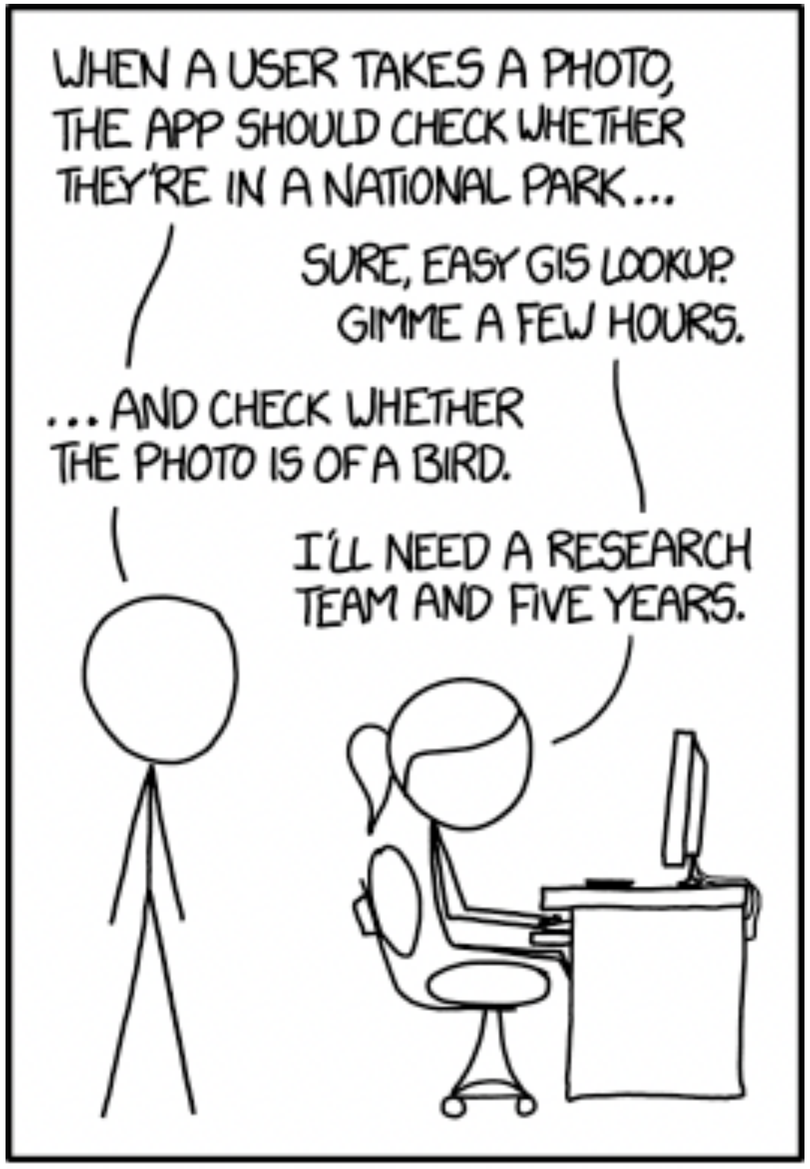
```
data <- read.cvs(“supercalifragilisticexpialidocious.csv”)
```

- 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.



IN CS, IT CAN BE HARD TO EXPLAIN THE DIFFERENCE BETWEEN THE EASY AND THE VIRTUALLY IMPOSSIBLE.

https://xkcd.com/1425

Important Skills to Remember

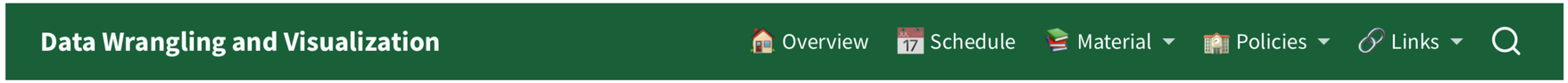
- Everyone - yes, everyone - gets errors.
- Computers are very **very** fast, and very **very** stupid.
- It's actually quite hard to know ahead of time what is very difficult and what is very easy.

This is a **learning process** - give yourself time, patience, and just stick to it. It will 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_f26



Data Wrangling and Visualization

Learn how to use quantitative data to inform decisions in public administration, public policy, and international affairs.

PAI 741 • Fall 2026
Maxwell School of Citizenship & Public Affairs
Syracuse University

Instructor

- Dr. Jack Reilly
- Eggers Hall 225
- jlreilly@syr.edu

Course details

- Tuesdays & Thursdays
- 11 AM -12:20 PM
- Eggers 225B

Please browse the course pages for course overview, schedule, course content, problem sets, course and campus policies, and guidelines for initially thinking about use of AI in this class. If you would prefer to have the syllabus as a single page, go here. And note you can download any page on the website you like as a pdf or formatted document.

I look forward to seeing you in class.

The website is built with the same tools you're going to learn in the course!

Syllabus Highlights

- Overall Course Structure:
  - We meet **twice** a week.
    - Tuesday: lectures & core topics
    - Thursday: R workshops (assignment coverage, new material, and work time)
  - There is an assignment due **every** week on Thursday
  - There are also optional **code review** sessions available for students, with the professor (sign up at jacklreilly.github.io/appointments or email)

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

- Three 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
  - **Data Programming:** Hadley Wickham, Garrett Grolemund, and Mine Çetinkaya-Rundel, *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data, 2nd ed*. Open access: https://r4ds.hadley.nz
- A **deeper programming** book is optional, but not required:
  - Braun & Murdoch, A First Course in Statistical Programming, 3rd Edition
  - Freeman & Ross, Programming Skills for Data Science

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_f26
  - Blackboard: http://blackboard.syr.edu/
  - Course data drive

Work Requirements and Grading

- Regular course participation and attendance (10%)
- Weekly Assignments (30%)
- Core Exam (15%)
- Practicums (30%)
- Final Project (15%)

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:
  - Fridays (formally 11-12 and 3-4, but . . . I'm around)
  - After class for short periods
  - by appointment
- Course community:
  - Feel free to help each other on problem sets and projects! (Not practicums/exams)
  - 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?
  - Here's one definition: a **computational bullshit machine.**

bull·shit | ˈbo͝olˌSHit | *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

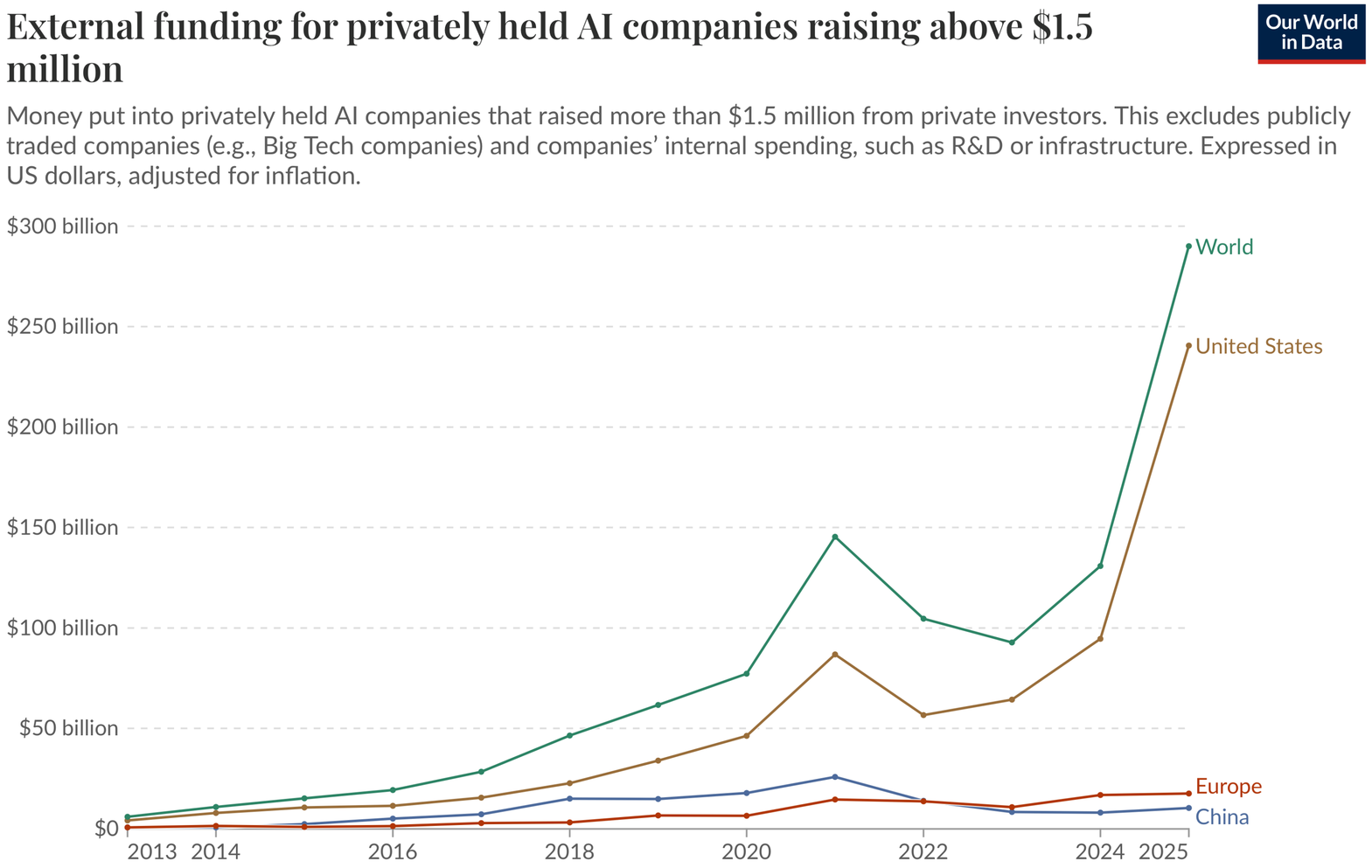
early 20th century: from bull[3] + shit.

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

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 the core exam. (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 Claude (ChatGPT, 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 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)

Bonus

- LLMs “*native language*” is plain text . . . just like the tools in this class
  - While they are developing capabilities elsewhere, working in plain text opens up wide capabilities in LLMs
  - Word, Excel, and PowerPoint . . . fluent, but like a second language

Now, **it's** **R** **Time!**