Week 12

- Getting Data

Getting Data

Outcomes for this evening

By the end of today's class, you should be able to do the following:

- Describe how your computer accesses content on the Internet
- Scrape content from websites using Python libraries
- Explain the moral and ethical issues of scraping
- Demonstrate use of APIs
- Extract content from documents (PDF, DOCX, RTF) and from webpages (HTML using BeautifulSoup)

- Science
- Getting Data using Computers
- Scraping Data
- APIs
- Activity: Negotiating for Data
- Get Data from Documents
- Fake Data
- Homework

"Data Science" without Science

- Getting data and using Python does not make you a Data Scientist

Steps commonly observed in practice:

- data gathering
- cleaning data
- data analysis
- data visualization

--> That doesn't mean you are applying a scientific process

What is Science?

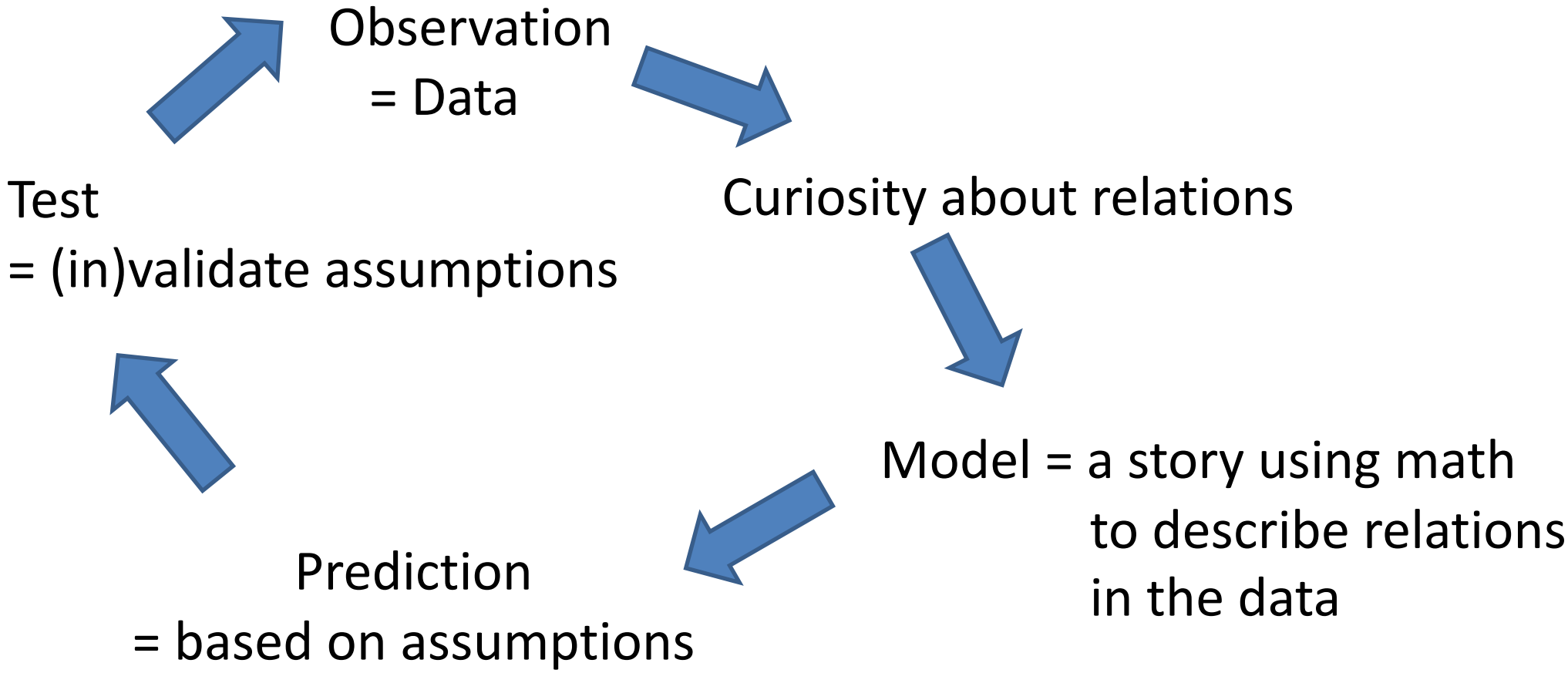
Science is a systematic enterprise that builds and organizes knowledge in the form of testable explanations and predictions about the universe.

https://en.wikipedia.org/wiki/Science

What is your perception of the process of Science?

Activity:

How to Science



Make a claim, Test the claim

A claim should be falsifiable. The test should be statistically meaningful.

In practice, strictness depends on intended outcome.

Good enough to make money:

- Altering what color the button on your ecommerce website is to maximize sales
- Recommending a movie to viewers

Critical to safety or health of humans:

- Determining which dosage of a drug is efficacious for medical treatments
- Guidance for an autonomous vehicle

Hard task in Science: Reproducibility

Someone else, following only your documentation, should be able to get an outcome that validates your claim.

An investment without immediate benefit.

Who's impacted by reproducibility: your future self; others.

Example reasons why reproducibility fails

- The outcome was specific to the data used
- Assumptions were wrong
- Software implementation has a bug
- Documentation provided is insufficient to enable reproducibility

Activity: Coding

- Homework Review

- ~~Science~~
- Final Project
- Getting Data using Computers
- Scraping Data
- APIs
- Activity: Negotiating for Data
- Get Data from Documents
- Fake Data
- Homework

You pick the data source for your project

from a list I provide, or you provide a suggestion in your proposal.

- Data discovery is outside scope of course, so I provide you data sources

- You should pick a data source you have some background knowledge in

--> Enables use of a model and generation of hypotheses

What I care about for the project

Challenge: determine relevant scope appropriate to your skills while still pushing your experience

Report: A description of research methods can be called “coherent” and “complete” when readers understand the process well enough to replicate it themselves. [citation]

Analysis

- Applies methods learned in class
- Uses Python 3 in Jupyter notebook
- Demonstrates process of data science, from gathering data to telling story through characterization
- Visualization: Intuitive, relevant

Example proposal content

The source URL for the data is http://pinkmonkeys.edu/. There is no cost to accessing this data. [Does access require creation of an account?]

Accessing this data does not violate any laws. This data does not appear to have been previously analyzed based on a Google search.

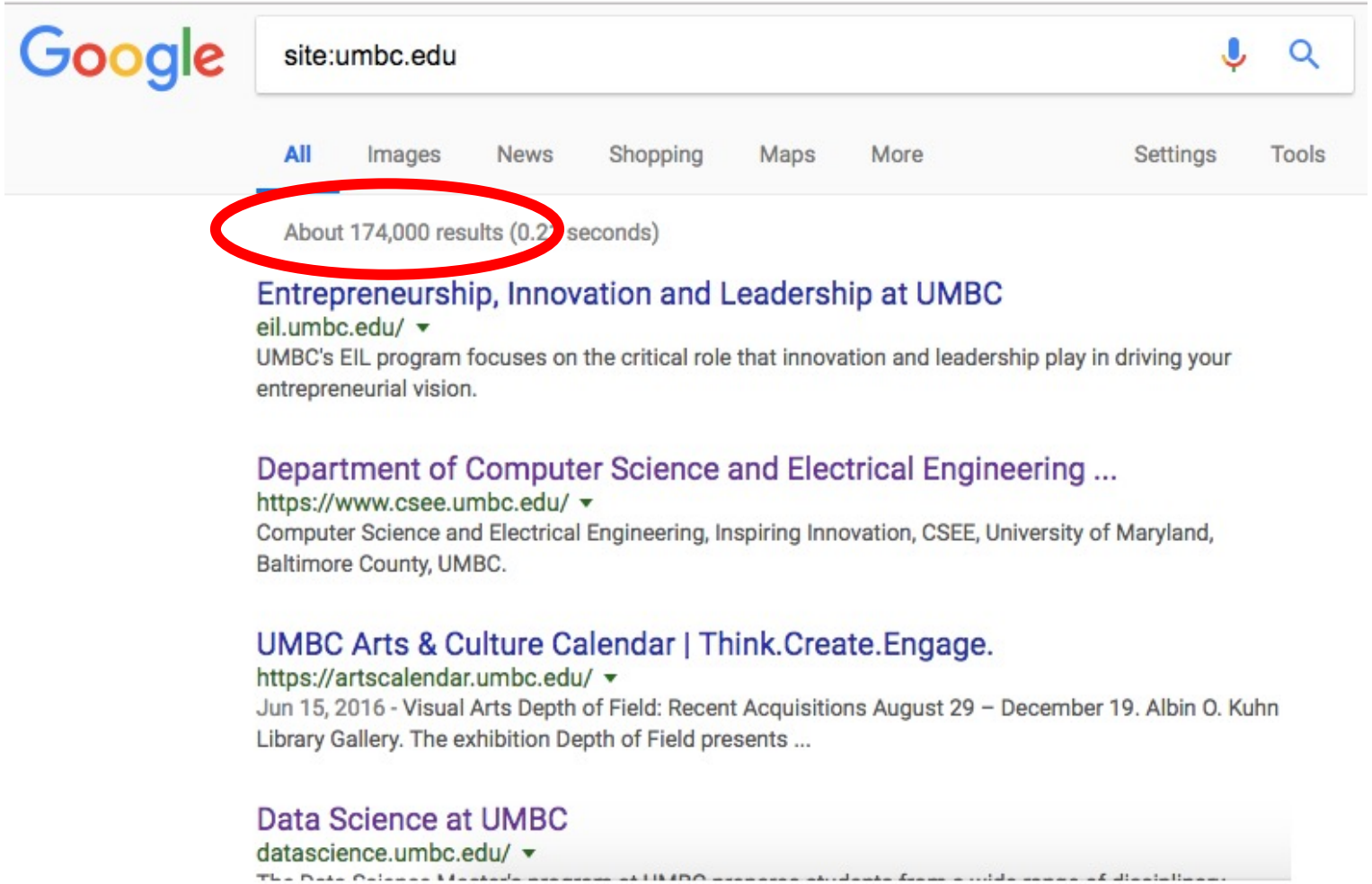
A preliminary survey of the data indicates there are 10,000 rows, 20 columns, and the file is 500 kB.

- ~~Science~~
- ~~Final Project~~
- Getting Data using Computers
- Scraping Data
- APIs
- Activity: Negotiating for Data
- Get Data from Documents
- Fake Data
- Homework

Online digital content

Normally presented via a GUI (web browser) for human interaction

I want all the data



Google has indexed trillions of web pages

Typical data sources

- Web GUI – unstructured text, electronic format
- Databases (SQL, HBase, Accumulo)
- Semi-structured text and mixed media, i.e. Word/PDF documents
- API (REST, SOAP)
- Electronic documents (Word, Powerpoint, Excel)
- Books, papers (hard copy) – unstructured text

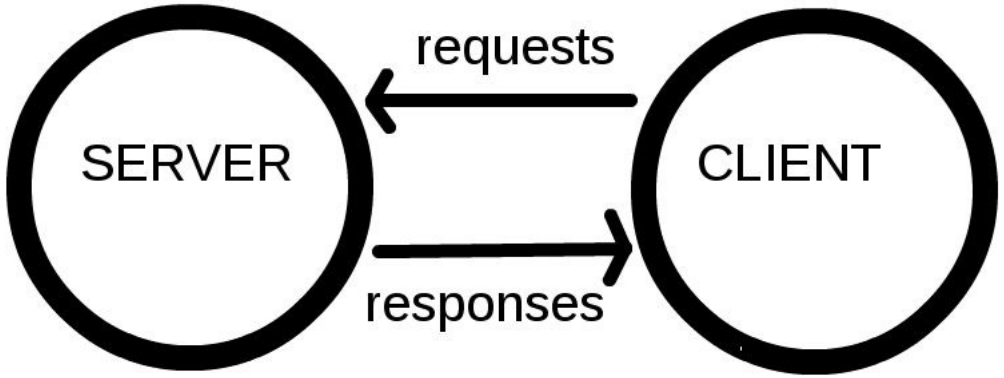
A well-rounded data scientist can extract data from any source

How to computer: Internet version

1. Type in web address
2. Get Content

If that's all you know,
then that's all you can do

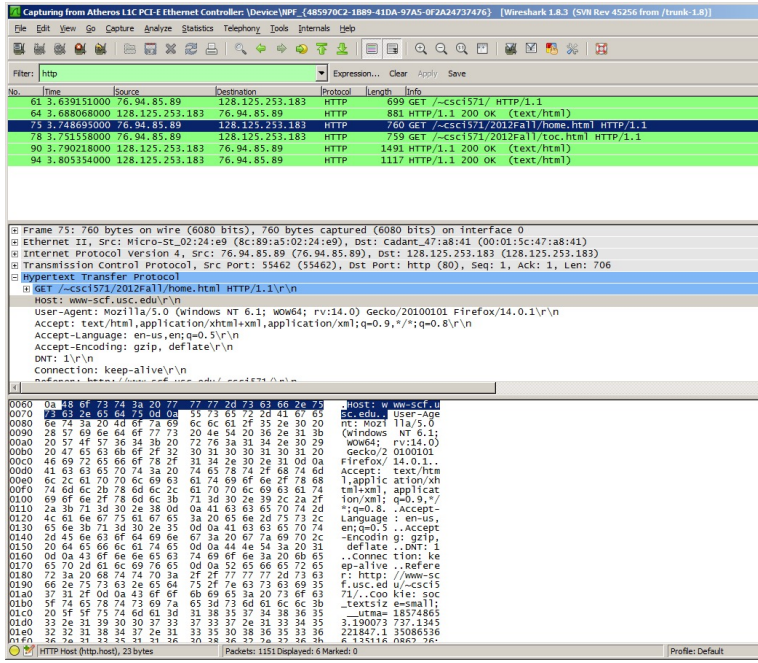
New Knowledge means New Vocabulary



In your role as a data scientist, understanding this jargon enables discussion of processes relevant to getting data

Packets contain information

- Text
- HTML: Hyper Text Markup Language
- CSS: Cascading Style Sheets
- JavaScript
- Other media, i.e. sound, video, images



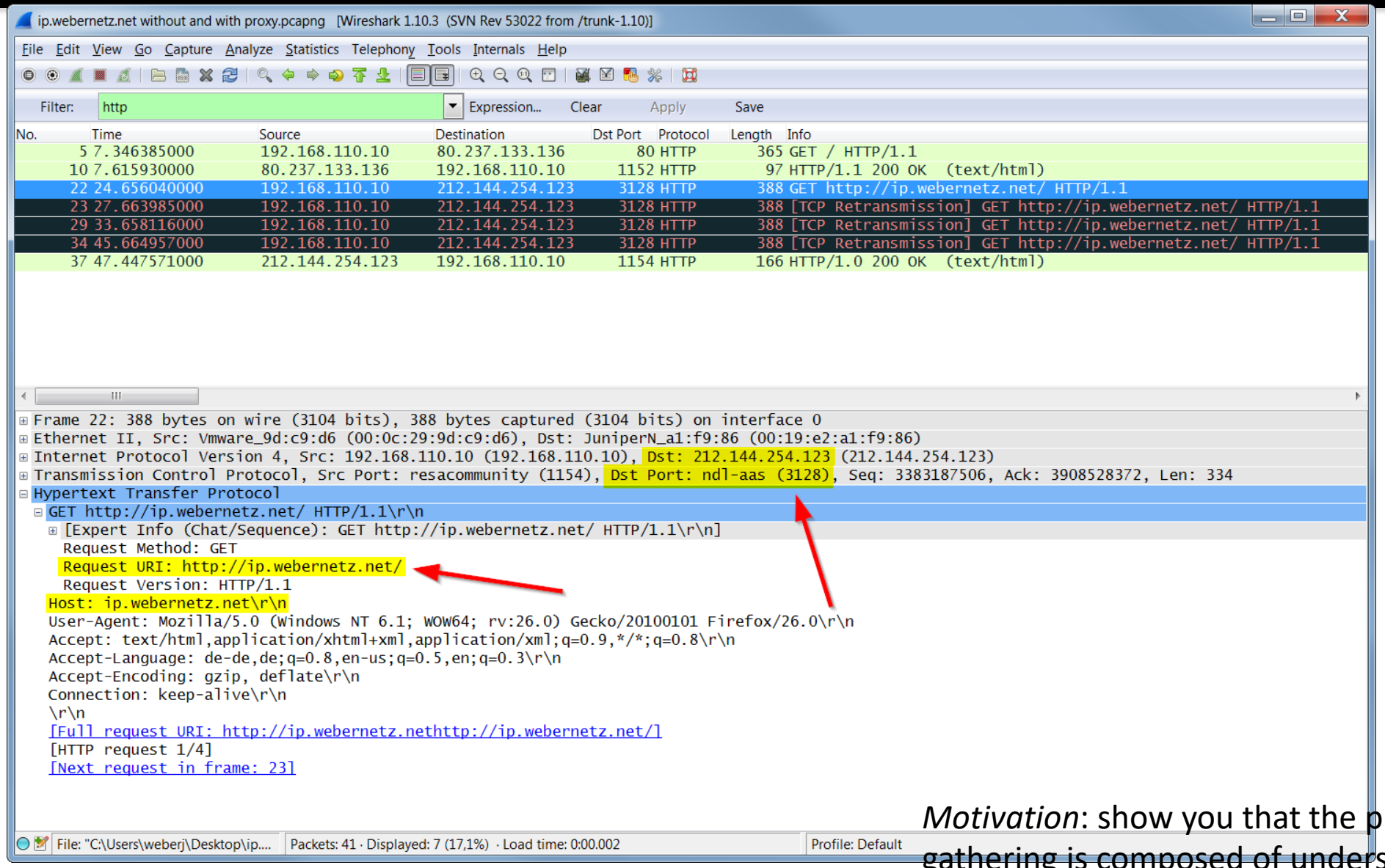
As a data scientist, this is likely a layer of knowledge deeper than most of you will care about

Process of getting a webpage

1. Client computer is connected to Internet and has an address
2. Person on client computer enters URL in web browser
3. URL is resolved to address of server
4. Using HTTP, browser sends a GET request to the server asking for the file https://mydomain.com/web-server.htm
5. Server sends content to client computer
6. Client computer renders the HTML content as a web page

Disclaimer: not describing ports, IP, DNS, TCP/IP, cookies, certs

Why this matters to you, the data scientist: when the process fails, your ability to troubleshoot depends on understanding the process



Motivation: show you that the process of data gathering is composed of understandable steps

Using computers to do boring repetitive work

There are many software tools to automate getting data on the web

- Curl
- Wget
- Requests
- Scrapy

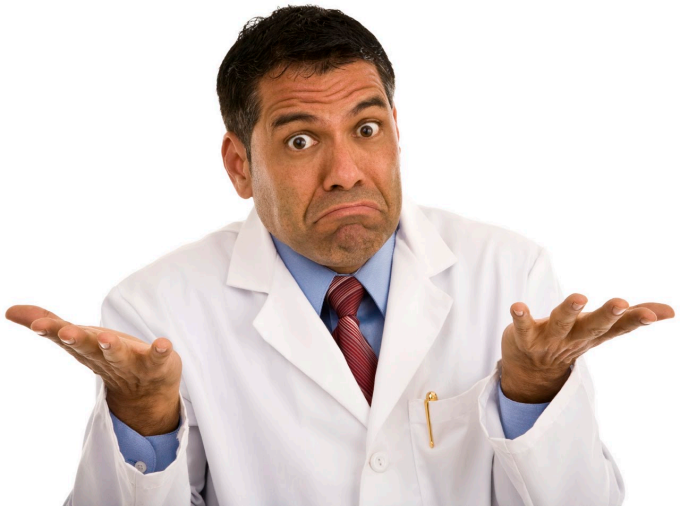
I will focus on text content

- ~~Science~~
- ~~Final Project~~
- ~~Getting Data using Computers~~
- Scraping Data
- APIs
- Activity: Negotiating for Data
- Get Data from Documents
- Fake Data
- Homework

"I didn't know that was illegal"

- Ignorance of the law is no defense to criminal charges

- Mistakes happen, but that doesn't mean you're not liable



I am not a lawyer
This is not legal advice

Legality of Content Scraping

Web Scraping = automated collection of content

- Few sites make explicit statements about scraping

Example: https://www.50states.com/terms.htm

- Web Crawlers
  - consume server resources (CPU, memory, power)
  - consume bandwidth
  - Access parts of sites not commonly accessed by humans

Instructions for crawlers: Robots.txt

http://www.robotstxt.org/robotstxt.html

- Based on your user agent string:

https://www.whatismybrowser.com/detect/what-is-my-user-agent

- Example robots.txt files:
  - https://cs.stanford.edu/robots.txt
  - https://slashdot.org/robots.txt

Activity: discuss ethics of scraping content

Let's discuss legal risks and moral issues:

- At what scale does scraping become an issue?

Just a few pages of content versus an entire domain

- Does your stance change based on who you are getting content from?

Small business versus large corporation

- Does your position change based on the purpose of scraping?

For research versus for commercial interest

Example of Data Scraping

- In 2016, "a group of Danish researchers publicly released a dataset of nearly 70,000 users of the online dating site OkCupid, including usernames, age, gender, location, what kind of relationship (or sex) they're interested in, personality traits, and answers to thousands of profiling questions used by the site."

- https://www.wired.com/2016/05/okcupid-study-reveals-perils-big-data-science/

Tools for getting content: wget (22 years old!)

- Normally a command in linux
- supports HTTP, HTTPS and FTP
- able to recover from a prematurely broken transfer

Examples:

```
wget www.fogcam.org
wget -m --limit-rate=200k http://mysite.com
```

-m=mirror; rate limit to 200 kB/sec

- Can follow links in content
- Also available in Python - https://pypi.org/project/wget/

curl

- Normally a command in linux
- Supports FTP, FTPS, Gopher, HTTP, HTTPS, SCP, SFTP, TFTP, TELNET, DICT, LDAP, LDAPS, FILE, POP3, IMAP, SMB/CIFS, SMTP, RTMP and RTSP

Example:

```
curl www.fogcam.org
```

- Also available in Python as "import pycurl"
- https://curl.trillworks.com/
- http://pycurl.io/docs/latest/index.html

Python: Requests

Send HTTP using Python, get webpage back

getting_data/get_1_requests.ipynb

Python: Scrapy

- Framework for creating a web crawler
- https://doc.scrapy.org/en/latest/intro/tutorial.html

```
conda install -c conda-forge scrapy
```

- A site intended for exploration: http://toscrape.com/
  - http://quotes.toscrape.com/
  - http://books.toscrape.com/

Having data for a browser is insufficient

- Scraping tools get data from the Internet
- Data for web browsers is not suitable for analysis

Activity: view source of https://www.umbc.edu/

In Chrome, View > Developer > View Source

In Firefox, Tools > Web Developer > Page Source

Parsing HTML using beautifulsoup

- getting_data/get_2_beautifulsoup.ipynb
- getting_data/get_3_table_requests_bs4_pandas.ipynb
- getting_data/get_4_requests_bs4_blocked_by_server.ipynb
- getting_data/get_5_requests_bs4_table_violates_robots.ipynb

Summary of Tools

Getting data:

- Wget
- Curl
- Requests

Analyzing HTML:

- BeautifulSoup

For Data 601, requests + BeautifulSoup is typically relevant

- ~~Science~~
- ~~Final Project~~
- ~~Getting Data using Computers~~
- ~~Scraping Data~~
- APIs
- Activity: Negotiating for Data
- Get Data from Documents
- Fake Data
- Homework

Scraping data isn't the only path

- https://www.ssa.gov/oact/babynames/
- Year and Popularity scale for a List of the Most Popular Names

Can we scrape the data?

- https://www.ssa.gov/robots.txt

Do policies allow scraping?

- https://www.ssa.gov/agency/websitepolicies.html

Sometimes data is made available for download

- https://www.google.com/search?q=site%3Assa.gov+download+baby+names
- https://www.google.com/search?q=site%3Assa.gov+api

In this case, we find the data is available for direct download:

- https://www.ssa.gov/oact/babynames/limits.html
- https://catalog.data.gov/dataset?tags=baby-names
- https://www.ssa.gov/open/data/
- https://www.ssa.gov/data/

Example of a simple API

- http://swoogle.umbc.edu/SimService/
- http://swoogle.umbc.edu/SimService/api.html

- getting_data/get_6_requests_use_simple_API.ipynb

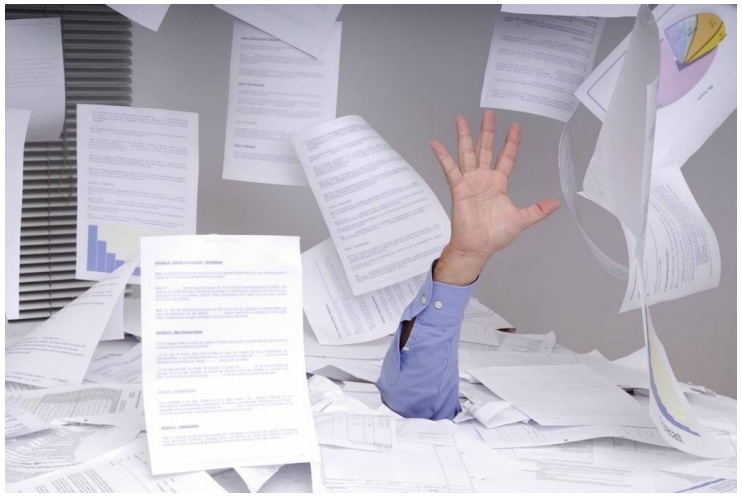
Constrain access using API keys

- https://sis.jhu.edu/api as an alternative to https://sis.jhu.edu/classes/

- getting_data/get_7_requests_use_API_with_key.ipynb

- ~~Science~~
- ~~Final Project~~
- ~~Getting Data using Computers~~
- ~~Scraping Data~~
- ~~APIs~~
- ~~Activity: Negotiating for Data~~
- Get Data from Documents
- Fake Data
- Homework

Getting data from (electronic) documents



Not all data is on the Internet

- Internal business records
- Reports
- Essays submitted by students of Data 601 section 4

getting_data/text_1_extract_text_from_docx.ipynb

--> If you can losslessly convert your document to HTML, do that

- ~~Science~~
- ~~Final Project~~
- ~~Getting Data using Computers~~
- ~~Scraping Data~~
- ~~APIs~~
- ~~Activity: Negotiating for Data~~
- ~~Get Data from Documents~~
- Fake Data
- Homework

When you don't have the data you need

Fake (synthetic) data

- What variables would you need in your model?
- What is the distribution of each variable?
- Expected frequency of measurements?
- Expected number of instances?
- Expected data format (JSON, XML, CSV, HTML, DOCX)
- Expected data type (date strings, text, numbers)

https://www.generatedata.com/ https://www.mockaroo.com/

Benefits of using fake data

- No sensitive or confidential information; privacy not at risk
- Enables you to write analysis before you have the data
- Play with visualizations
- Generate report layout
- Negotiate with stakeholders so they have something to point at

How to create fake data

- Consider the characteristics of the data
  - Name
  - Phone number
  - address
- Create data that has those characteristics

- Create the data structure you want to work with
  - List, table
  - Streaming versus static

- ~~Science~~
- ~~Final Project~~
- ~~Getting Data using Computers~~
- ~~Scraping Data~~
- ~~APIs~~
- ~~Activity: Negotiating for Data~~
- ~~Get Data from Documents~~
- ~~Fake Data~~
- Homework

Consequences of Exploration

Exploration means you don't know.

- Not all your investments get to the customer
- You will write code you discard
- Perfection is a wasted when exploring

In contrast, product development focuses on what will reach customers in order to minimize costs.

Data Science doesn't always win

Having a correct and complete solution does not mean it gets selected

Decision makers can be swayed

- By emotionally-driven stories
- On the basis of prior relationships
- To select a cheaper option.

It is vital to determine what convinces your audience before investing effort in Data Science analysis that uses code, data, math.

If your customer plays the lottery, they may not understand your numerical results