# Web Scraping with Python

Cornelia Ilin, PhD

Department of Ag & Applied Economics

UW-Madison

Week 3 - Summer 2019

# What is Web Scraping?

- Also known as Web Data Extraction
- A technique used to extract large amount of data from webpage source code
- Data is extracted and saved to a local file/database in your computer

# What do we need to know

- **(1)** Basic knowledge of text-based **mark-up languages** (HTML and XHTML)
- **(2)** Good command of R and/or Python **libraries for web scraping** (I am being picky here, other languages should work too)
- **(3)** Good command of R and/or Python libraries for **data manipulation** and **cleaning**
- **(4) Data visualization**

# (1) Text-based languages

- HTML and XHTML

- XHTML extends HTML by providing well-formatted pages (maybe other things too?)

- Good resources:
  - https://www.w3schools.com/html/html_intro.asp
  - https://www.makeuseof.com/tag/5-steps-understanding-basic-html-code/

# (1) Getting the HTML (source code)

- **Windows**
  - Right click on the webpage and select “View Source”
- **Mac**
  - Right click on the webpage and select “Show Page Source”

➢ **Two examples (what is the difference?)**

- https://corneliailin.github.io/aae875_summer2019/
- www.windy.com

# (1) Getting the HTML (source code)

- **Windows**
  - Right click on the webpage and select “Inspect”
- **Mac**
  - Right click on the webpage and select “Inspect Element”

➢ **Two examples**

- https://corneliailin.github.io/aae875_summer2019/
- www.windy.com

# (1) Getting the HTML (source code)

- **Windows**
  - Right click on the webpage and select “Inspect”
- **Mac**
  - Right click on the webpage and select “Inspect Element”

➢ **Two examples**

- https://corneliailin.github.io/aae875_summer2019/
- www.windy.com

# (2) Python libraries for web scraping

- To get the HTML file: urllib.request library
  - https://docs.python.org/3/library/urllib.request.html

- To extract data from the HTML file (parse the HTML): BeautifulSoup library
  - https://www.crummy.com/software/BeautifulSoup/bs4/doc/

# (2) Python libraries for web scraping

- BeautifulSoup has specific functions that help you extract title, tags, classes etc.
- Tags categorize different elements of the document (responsible for page layout)
- Tag examples: <a> for hyperlinks; <table> for tables; <tr> for table rows

# (3) Python lib for data manipulation and cleaning

- **First:** what is the data structure we want?
  - Examples: lists, arrays, dataframe?

- **Second**: If dataframe, what is the best Python module to manipulate objects?
  - pandas module (https://pandas.pydata.org/pandas-docs/version/0.22/index.html#module-pandas)

# (4) Python libraries for data visualization

- matplotlib library
  - https://matplotlib.org/
- seaborn library
  - https://seaborn.pydata.org/

# Sources (accessed July 19, 2019)


[1] https://www.datacamp.com/community/tutorials/web-scraping-using-python

[2] https://www.w3schools.com/html/html_intro.asp

[3] https://www.makeuseof.com/tag/5-steps-understanding-basic-html-code/

[4] https://corneliailin.github.io/aae875_summer2019/

[5] www.windy.com

[6] https://docs.python.org/3/library/urllib.request.html

# Sources – cont’d (accessed July 19, 2019)


[7] https://www.crummy.com/software/BeautifulSoup/bs4/doc/

[8] https://pandas.pydata.org/pandas-docs/version/0.22/index.html#module-pandas

[9] https://matplotlib.org/

[10] https://seaborn.pydata.org/