AUTOMATED DATA COLLECTION WITH PYTHON

Ivan Hernandez, Ph.D
GOALS FOR THE SESSION

› Discuss the Growing Interest in **Big Data**

› Introduce Automated Data Collection **Methods**

› Describe the **Process** of Automating Data Collection

› Present how to use **Python** to Extract Data from the Web
DATA TODAY
CHANGING PERSPECTIVES ON DATA

› Data driven decisions being emphasized

› Age of Big Data
  › Larger
  › More Frequent
  › More Varied

› Where to access this data?


SOURCES OF BIG DATA

Finance Trends

Orange County family says they were kicked off Delta flight after argument
Los Angeles Times - 1 hour ago
An Orange County couple says they were kicked off a Los Angeles-bound Delta flight last month after airline staff insisted their 2-year-old son could not sit by himself, even though the family had already paid for the seat.
Family, booted from Delta flight and threatened with jail after refusing to...Washington Post
California family kicked off Delta flight after refusing to give up...Fox News
New York Daily News - Business Insider

Employee Information

Knowledge Bases

Big data

This word is a decorate collection of data. For the best, see Big Data.

By now a key data tool, or an engine of computing, big data processing applications refer to datasets deemed too large for them to handle. To handle, big data requires powerful computing infrastructure, algorithms to sift through the sampling, and analysis to make sense out of the results. The sheer amount of data available today is revolutionizing business models and operations in much the same way that the digital revolution did in the mid-20th century. The amount of data that can be captured and analyzed across industries is overwhelming. As a result, businesses have the opportunity to use big data to understand their customers better and make better business decisions. The challenge is to find the right use cases and to implement them effectively.

Social Media

Sports

News
Web-based data can facilitate market intelligence

Provides the following knowledge benefits

- Pricing analysis
- Competitive intelligence
- Events
- Product data
- Popularity
- Reputation
COLLECTING BIG DATA

› How to collect this available data?

› **Human collection method:**
  › Sit in front of a computer
  › Go to a website of interest
  › Copy the relevant data
  › Paste into a common file
  › Repeat 1,000,000 times for other data and other websites
COLLECTING BIG DATA

• Limitations of Human Collection:

  • Menial
  • Mental Demands
  • Inaccuracy
  • Cost
  • Scalability
Automated Data collection is about being able to translate what you would do as a human collecting the data to what your computer can do.

Goal: Give a computer a set of instructions to follow

- First do this
- Then do that
- Finally do this

Let the computer carry-out those instructions, and you come back to a completed project.

How do you talk to a computer?
HOW TO TALK TO A COMPUTER

› We can tell a computer what to do using programming languages:
  › Python
  › C
  › Java
  › PHP
  › R

› To tell a computer what to do using a programming language requires:

  › Understanding how a computer sees things

  › Understanding what the functions that are available
THINKING LIKE A COMPUTER

› Automating requires you to consider what are the capabilities and limitations of a computer.

This Title is in an H1 tag

This text is inside a div tag, whose class is equal to box1
This text is inside a div tag, whose class is equal to box2
This text is inside a span tag, whose class is equal to box3
This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional Content: This content is not inside any tag

This is what you see

This is what your computer sees
Know the functions/instruction that are available from the programming language

Automated Data collection is about being able to translate what **you would do** as a human collecting the data to corresponding steps of what your **computer can do**

**Example: Download the Main Headline from the New York Times**

**What you would do:**
- Go to the New York Times website
- Look at the text in the main heading
- Copy that headline with the mouse
- Open a text file called “data.txt”
- Paste the copied text in the file
- Save it

**What your computer can do:**
- `page = requests.get(“http://nyt.com”).text`
- `headline = page.find(“h1”)`
- `text = headline.text`
- `file = open(“data.txt”,”wb”)`
- `file.write(text)`
- `file.close()`
You have to think about everything you would do, and how your computer can do it.

First, think how would YOU download the latest stock prices for Apple?

- I would go to Google Finance (https://www.google.com/finance)
- I would type in “Apple” at the search bar
- I would look for the bold number
- I would copy the price
- I would open a text file
- I would paste the price into the file
- I would save the file and close it
Next, think about how can you have your COMPUTER do those same steps:

- It would be hard to have a computer type in a search box, so I have to think of a way for it to access a stock another way - THINK ABOUT WHAT A COMPUTER CAN DO
- Notice that the url for Apple’s stock price page is:
  - https://www.google.com/finance?q=APPL
  - The stock name always comes after “q=”

- If I know the stock name, I can tell a computer to go to that page
- I can tell a computer to look for text tagged as bold
- I can tell a computer to save the bold text as a variable called “price”
- I can tell the computer to open a file
- I can tell the computer to write the stock price variable in the file
- I can tell the computer to save and close the file
FOUR STEPS OF AUTOMATED DATA COLLECTION

› Four Steps to Automatically Collecting Data (Scraping)
  › Download the HTML source of a page
  › Extract the content from the HTML
  › Save the content
  › Repeat the process on a different Page

› Each of those steps has specific commands in Python (and R) associated with it

› Successfully collecting data requires chaining those commands together
STEP 1: DOWNLOAD THE HTML SOURCE

- Download the HTML source of a page
  - Python command:
    ```
    import urllib
    page = urllib.urlopen("https://www.google.com/finance?q=APPL")
    ```
  - R Command
    ```
    library(RCurl)
    page <- getURL("https://www.google.com/finance?q=APPL")
    ```
STEP 2: EXTRACTING THE CONTENT

› Extract the content

We’ll get to this part in a minute...
**STEP 3: SAVE THE CONTENT**

- **Save the Content**
  - **Python command:**
    ```python
textfile = open("data.txt", "a")
textfile.write(content)
textfile.close()
```  
  - **R Command**
    ```r
write(content,"data.txt",append=TRUE)
```
STEP 4: REPEAT THE PROCESS

Repeat the Processes

Python command:

```python
stocks = ["AAPL", "GOOGL", "MSFT"]
for stock in stocks:
    *** extract content ***
```

R Command

```r
stocks <- c("AAPL", "GOOGL", "MSFT")
for (stock in stocks){
    *** extract content ***
}
```
The hardest part of automated data collection is extracting the content. Code must be customized to your particular situation. It depends on:
- How much content is needed (one thing or many?)
- The structure of the HTML (is it bold?, is it a heading?, is it italicized?)
- The kind of content (is it text?, is it a url?, is it an image?)

We will go over the major cases/situations that you could have.
EXTRACTING CONTENT FROM WEB SITES
THE STRUCTURE OF A WEBSITE

› Extracting content from a website requires understanding how websites are written

› Websites are written in HTML
  › Text is formatted by putting it in between “tags”, which describe the way it should be displayed in a browser
  › Typically each tag has an opening tag and a closing tag, which isolate the specific text to be formatted

› Example:
  › <h1>Hello</h1>
  › <i>Hello</i>
  › <strong>Hello</strong>
To view the raw HTML of a website (i.e., the source), you can

- Chrome/Firefox/Opera/Internet Explorer: Ctrl + U
- Safari: Command + Option + U

```html
<html>
  <head>
    <title>Example Domain</title>
  </head>
  <body>
    <h1>Example Domain</h1>
    <p>
      "This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission."
    </p>
    <p>
      <a href="http://www.iana.org/domains/example">More information...</a>
    </p>
  </body>
</html>
```

The HTML source of http://example.com
You can also right-click on a specific part of a website and select "Inspect" to more easily examine a specific part of the HTML.
This Title is in an H1 tag

This text is inside a div tag, whose class is equal to box1
This text is inside a div tag, whose class is equal to box2
This text is inside a span tag, whose class is equal to box3
This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional Content: This content is not inside any tag
This Title is in an H1 tag

This text is inside a div tag, whose class is equal to box1

This text is inside a div tag, whose class is equal to box2

This text is inside a span tag, whose class is equal to box3

This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional Content: This content is not inside any tag

The main heading is inside of an <h1> tag
READING THE STRUCTURE OF A WEBSITE

This Title is in an H1 tag

This text is inside a div tag, whose class is equal to box1

This text is inside a div tag, whose class is equal to box2

This text is inside a span tag, whose class is equal to box3

This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional Content: This content is not inside any tag

The second line is inside a <div> tag with a class equal to “box1”
This Title is in an H1 tag

This text is inside a div tag, whose class is equal to box1

This text is inside a div tag, whose class is equal to box2

This text is inside a span tag, whose class is equal to box3

This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional Content: This content is not inside any tag

The third line is inside a <div> tag with a class equal to “box2”
READING THE STRUCTURE OF A WEBSITE

This Title is in an H1 tag

This text is inside a div tag, whose class is equal to box1

This text is inside a div tag, whose class is equal to box2

This text is inside a span tag, whose class is equal to box3

This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional Content: This content is not inside any tag

The fourth line is inside a <span> tag with a class equal to “box3”
READING THE STRUCTURE OF A WEBSITE

This Title is in an H1 tag

This text is inside a div tag, whose class is equal to box1

This text is inside a div tag, whose class is equal to box2

This text is inside a span tag, whose class is equal to box3

This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional Content: This content is not inside any tag

The fourth line is inside a <p> tag with an id equal to “box4”
READING THE STRUCTURE OF A WEBSITE

This Title is in an H1 tag

This text is inside a div tag, whose class is equal to box1
This text is inside a div tag, whose class is equal to box2
This text is inside a span tag, whose class is equal to box3
This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional Content: This content is not inside any tag

The fifth line is inside an <a> tag with an href that directs to google.com
Reading the structure of a website

This title is in an H1 tag

This text is inside a div tag, whose class is equal to box1

This text is inside a div tag, whose class is equal to box2

This text is inside a span tag, whose class is equal to box3

This text is inside a p tag, whose id is equal to box4

This is a link to Google

Additional content: This content is not inside any tag

The fifth line is NOT inside any tags
Extracting Content from a Web Page

When you have the HTML source of a website, you need to examine where in the source is the content you want to extract.

- What are its closest tags?
- Are those tags unique to the content?
- Does the tag have an id or class name?
- Does some specific word or character always precede the content of interest?

When you know the answers to the above questions, you direct Python to extract the content based on the identifying information.
DEMONSTRATION OF DATA EXTRACTION
Walkthrough of How to Extract Web Page Content With Python
SUMMARY
There’s a growing interest in the benefits of “Big Data”

The internet provides a vast source of data

Data can be collected from the internet at scale through automation

Automated data collection involves thinking of the steps a human would take when collecting the data, and translating those steps to procedures a computer can understand.

Using the urllib and BeautifulSoup libraries, Python provides a method for automating data collection from the internet.