python

Introduction to Web Scraping with Python

Dr. Vincent Grégoire
Department of Finance

March 2017

THE UNIVERSITY OF MELBOURNE

What is web scraping

- A lot of valuable information is freely available online, but not necessarily in a nice structured format.
- Web scraping is the process of collecting data that is more or less structured from the Internet.
- It includes:
  - Crawling from page to page, following links (think Google crawler)
  - Accessing APIs to get data
  - Extracting information from resulting downloads, such as Web pages, PDF documents, Word documents, etc...

Why web scraping

- A lot of valuable information is freely available online, but not necessarily in a nice structured format.
- Nicely formatted datasets, such as CRSP, Compustat, etc. have been extensively studied, so the bar is high to answer important questions using only those.
- Interesting questions can be answered by those who manage to find the right data.

When not to use Python for web scraping

1. See if data is available another way.
2. Consider if another tool is better suited for the task.
   - i.e. Chrome/Firefox extensions to download all files from a page.
3. Consider time required to write code vs time to download manually.

The tools of web scraping

Beautiful Soup

Powerful package for parsing hierarchical structured data such as HTML, XML and Word XML (.docx).

Scrapy

Package for building crawlers and automating information extraction. Works with Beautiful Soup.

urllib, requests

Package for accessing websites, submitting forms and downloading pages and files.

Selenium

Automate your web browser with Python.

Application Programming Interface (API)

- Various online services provide an API with functions a programmer can use to access some specific features or request data. Some are official and supported, some are unofficial (but people know about them.) The data available is up to the provider.

- Some services with official APIs:
  - Twitter
  - LinkedIn
  - Facebook
  - Thomson Reuters Tick History
  - Indiegogo (crowdfunding)
  - LendingClub (peer-to-peer lending)
  - Instagram
- Some services with unofficial APIs:
  - Kickstarter
  - Google Trends

Packages for specific websites

Here are a few packages designed to access the API of specific websites (note I have not tested all of them):

- Kickstarter
  - https://github.com/argaen/kickscraper
- Twitter
  - https://github.com/bear/python-twitter
- Yahoo! Finance
  - https://github.com/lukaszbanasiak/yahoo-finance
- LinkedIn
  - https://github.com/ozgur/python-linkedin
- LendingClub
  - https://github.com/jgillick/LendingClub

Examples

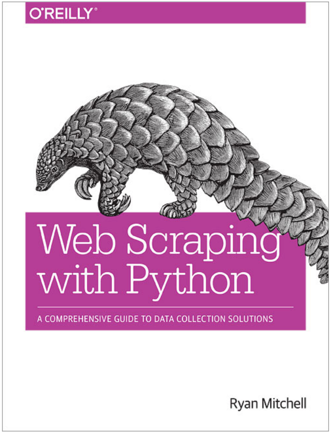
We'll see two examples today:

- The impact of Trump's tweets on the stock market
  - Use of the Twitter API
  - Use of TextBlob for sentiment analysis
  - Combination with intraday stock market data

- Downloading 10-Ks from SEC Edgar
  - Pure web scraping
  - How to automatically download files from websites

Books

For more advanced web scraping, see:



O'REILLY

Web Scraping with Python

A COMPREHENSIVE GUIDE TO DATA COLLECTION SOLUTIONS

Ryan Mitchell

Online version available via the library.