

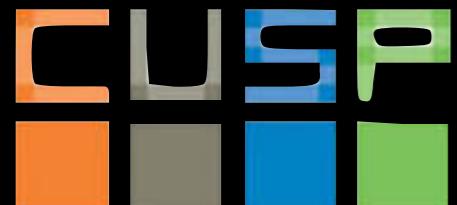
Urban Informatics

Fall 2018

dr. federica bianco fbianco@nyu.edu

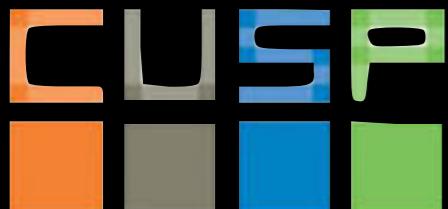


@fedhere



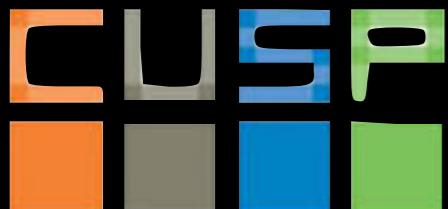
Summary:

- **Epistemological concepts:**
falsifiability, law of parsimony,
- **Good scientific practice:**
reproducibility of research



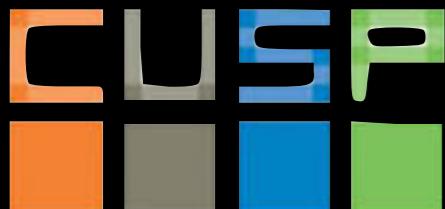
Summary:

- **Epistemological concepts:**
falsifiability, law of parsimony,
- **Good scientific practice:**
reproducibility of research
- **Gathering parsing data, API:**
data munging or wrangling, data jujitsu



Summary:

- **Epistemological concepts:**
falsifiability, law of parsimony,
- **Good scientific practice:**
reproducibility of research
- **Gathering parsing data, API:**
data munging or wrangling, data jujitsu
 - types of data
 - reporting data for reproducibility
 - cleaning and tabulating data
 - reading data from CSV files
 - reading data from JSON files
 - reading data from the CUSP dataHub
 - reading data from APIs



- IDEA
- dataset
 - define ideal data
 - figure out best data available
 - figure out if you can get new data
 - obtain data (including policy issues + technical issues)
- data handling
 - joining databases
 - formatting data
- exploratory data analysis
 - machine learning (clustering? dimensionality reduction?)
- statistics
 - models (regression)
 - prediction
 - validation (simulations)
- interpretation
- presentation
 - visualization
 - write a paper!



- IDEA
- dataset
- Define ideal data

PROBLEM IDENTIFICATION

FORMULATING HYPOTHESIS

DATA IDENTIFICATION

- figure out best data available
- figure out if you can get new data

- obtain data (including policy issues + technical issues)
- data handling
 - joining databases
 - formatting data
- exploratory data analysis
 - machine learning (clustering? dimensionality reduction?)
- statistics
 - models (regression)
 - prediction
 - validation (simulations)
- interpretation
- presentation
 - visualization
 - write a paper!



- IDEA
 - dataset
- ## PROBLEM IDENTIFICATION

FORMULATING HYPOTHESIS

- define ideal data
- figure out best data available
- figure out if you can get new data
- obtain data (including policy issues + technical issues)

DATA IDENTIFICATION

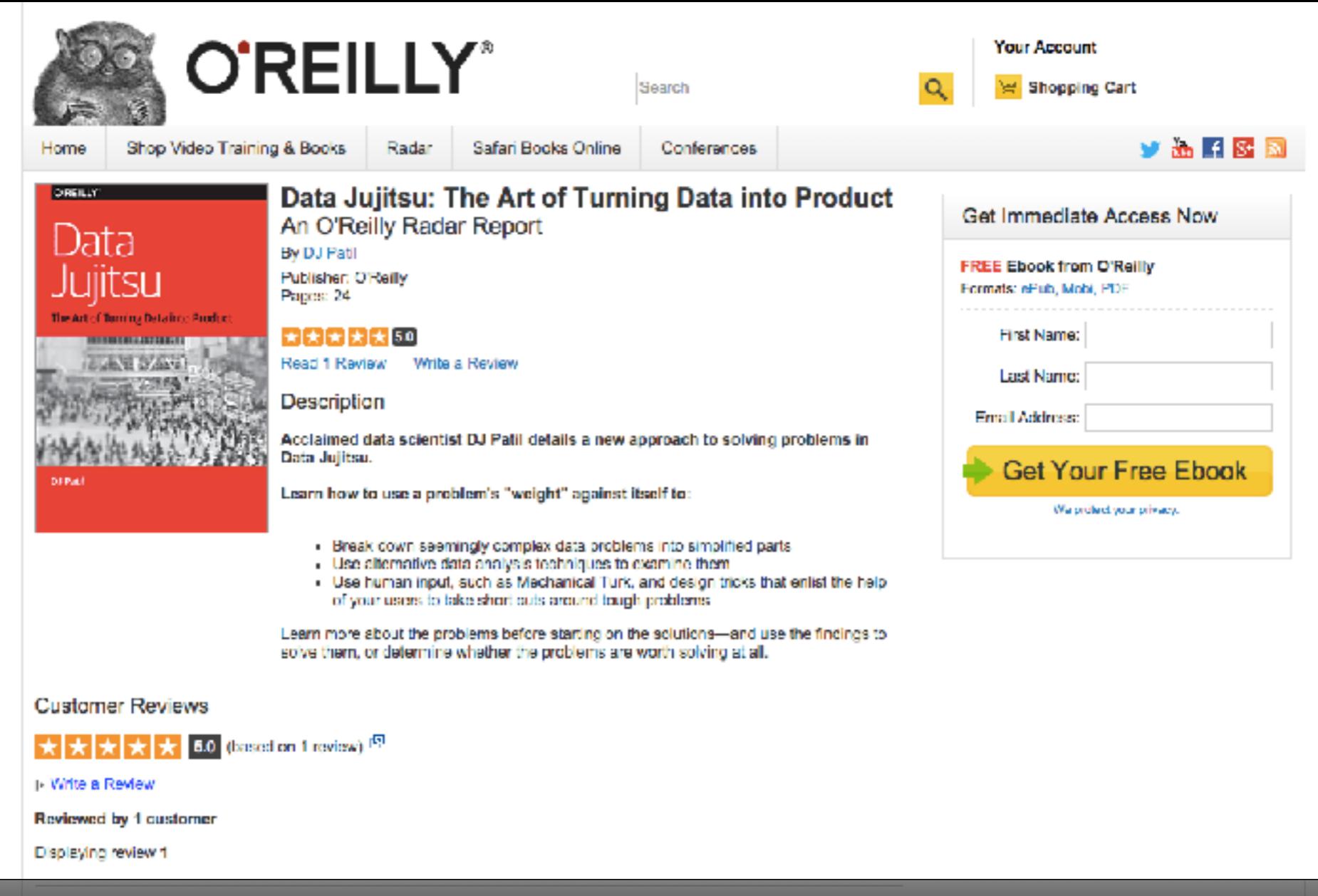
data handling

- joining databases
- formatting data

DATA PREPARATION

- exploratory data analysis
 - machine learning (clustering? dimensionality reduction?)
- statistics
 - models (regression)
 - prediction
 - validation (simulations)
- interpretation
- presentation
 - visualization
 - write a paper!





The screenshot shows the O'Reilly website product page for "Data Jujitsu: The Art of Turning Data into Product".

Product Information:

- Title:** Data Jujitsu: The Art of Turning Data into Product
- Author:** An O'Reilly Radar Report
- By:** DJ Patil
- Publisher:** O'Reilly
- Pages:** 24

Reviews: ★★★★☆ 5.0 (based on 1 review) [Read 1 Review](#) [Write a Review](#)

Description:

Acclaimed data scientist DJ Patil details a new approach to solving problems in Data Jujitsu.

Learn how to use a problem's "weight" against itself to:

- Break down seemingly complex data problems into simplified parts
- Use alternative data analysis techniques to examine them
- Use human input, such as Mechanical Turk, and design tricks that enlist the help of your users to take short cuts around tough problems

Learn more about the problems before starting on the solutions—and use the findings to solve them, or determine whether the problems are worth solving at all.

Customer Reviews:

★★★★★ 5.0 (based on 1 review) [Read 1 Review](#) [Write a Review](#)

Reviewed by 1 customer

Displaying review 1

Get Immediate Access Now:

FREE Ebook from O'Reilly
Formats: ePub, Mobi, PDF

First Name: Last Name: Email Address:

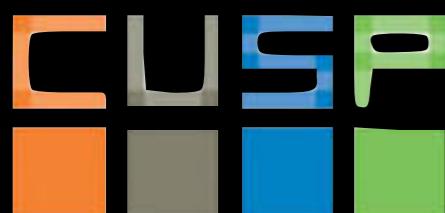
Get Your Free Ebook

We protect your privacy.

CUSP SWIGVIA SWIGVIA is a trademark of CUSP

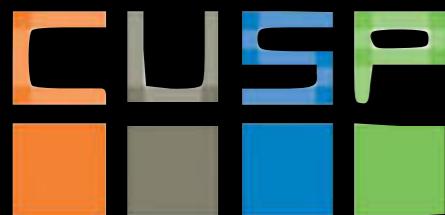
[...]data products are unique in that they are often extremely difficult, and seemingly intractable for small teams with limited funds. Yet, they get solved every day.

How? Are the people who solve them superhuman data scientists who can come up with better ideas in five minutes than most people can in a lifetime? Are they magicians of applied math who can cobble together millions of lines of code for high-performance machine learning in a few hours? No.
(continue in the next page...)



(...continued)

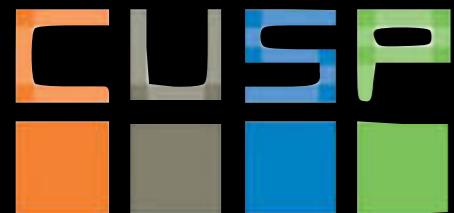
Many of them are incredibly smart, but meeting big problems head-on usually isn't the winning approach. There's a method to solving data problems that avoids the big, heavyweight solution, and instead, concentrates on building something quickly and iterating. Smart data scientists don't just solve big, hard problems; they also have an instinct for making big problems small.



Data are messy

...

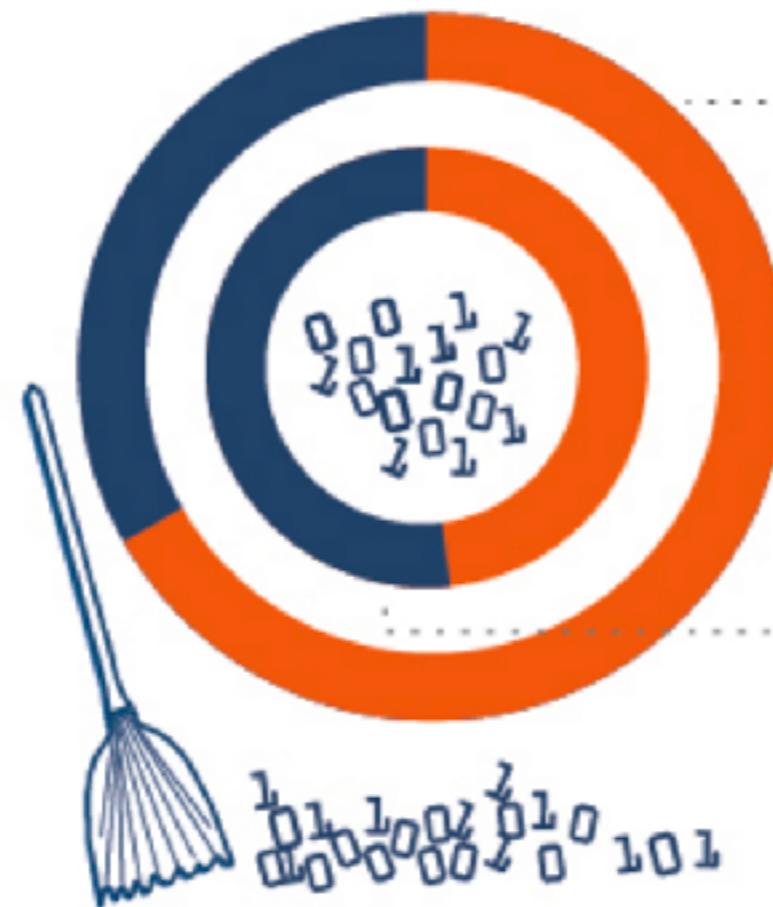
(and you have to clean the mess!)



II: Data Wrangling

WHAT ARE THEIR BIGGEST CHALLENGES?

Dirty data is the #1 hurdle for data scientists.



66.7%

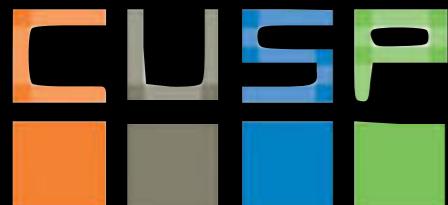
of data scientists say cleaning and organizing data is their most time-consuming task.

52.3%

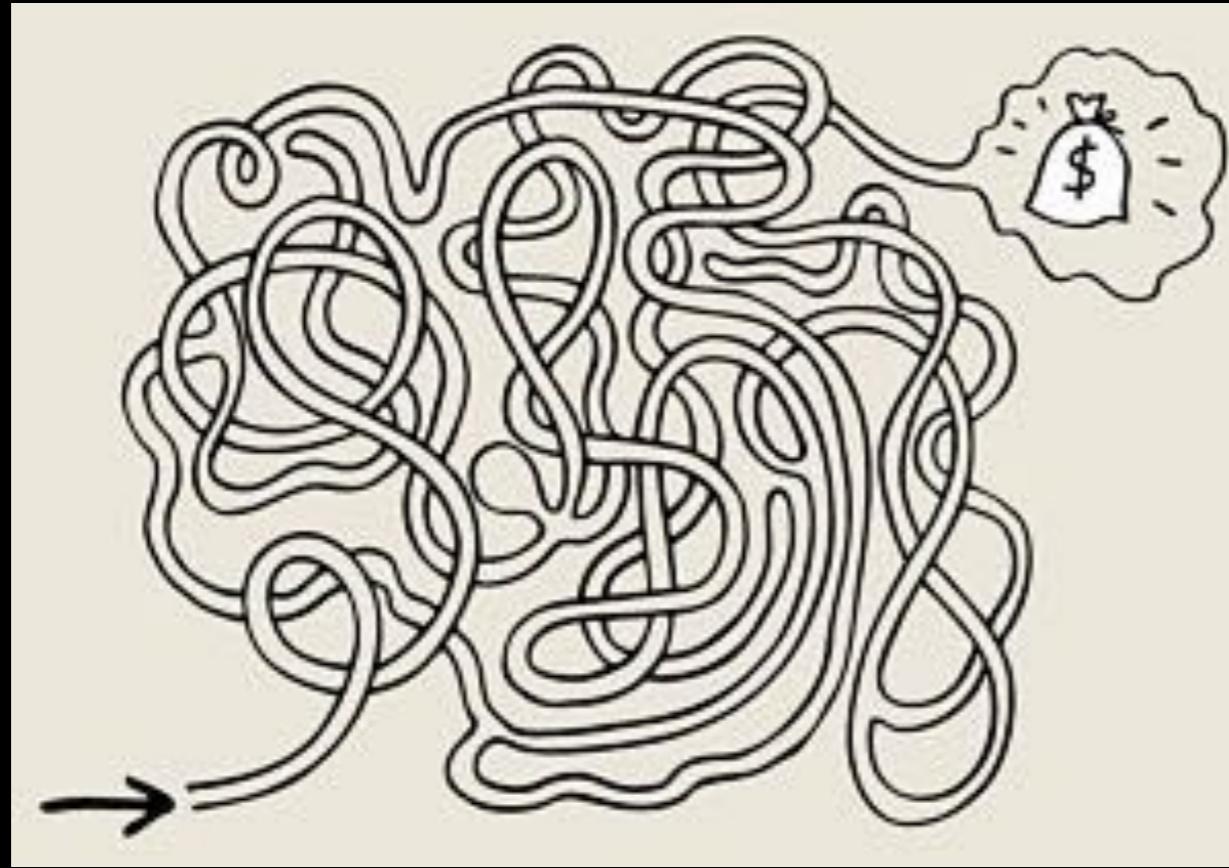
of data scientists cite poor quality data as their biggest daily obstacle.

2015 survey of data scientists

<https://www.crowdflower.com/the-data-behind-todays-data-scientists-an-infographic/>
II: Data Wrangling



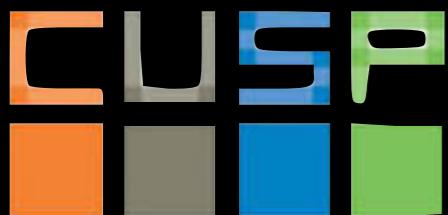
noisy data in



tidy data out



(urban) science!



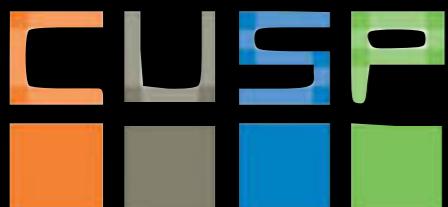
II: Data Wrangling

Reproducible research means:

code raw data

A Reproducible Research: MatLab Code

This paper implements the spirit of Reproducible Research, a publication protocol initiated by John Claerbout [Claerbout 1990] and developed by others at Stanford and elsewhere. The underlying idea is that the most effective way of publishing research is to include everything necessary to reproduce all of the results presented in the paper. In addition to all relevant mathematical equations and the reasoning justifying them, full implementation of this protocol requires that the data files and computer programs used to prepare all figures and tables are included. Cogent arguments for Reproducible Research, an overview of its development history, and honest assessment of its successes and failures, are eloquently described in [Donoho et al. (2008)].

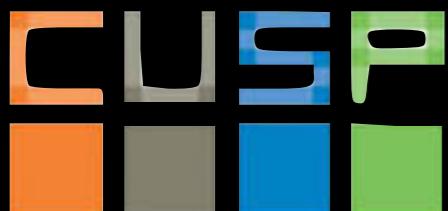


Reproducible research means:

code raw data

privacy concerns

in order for your research to be reproducible
the data you use must be accessible BUT
THAT IS NOT ALWAYS POSSIBLE:
CUSP has access to data that has restricted
access. Share your data when possible!



Reproducible research means:

code raw data

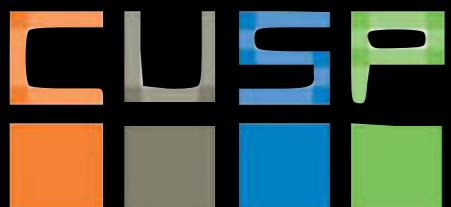
Remove sensitive data

Some day you or a collaborator may accidentally commit sensitive data, such as a password or SSH key, into a Git repository. Although you can remove the file from the latest commit with `git rm`, the file will still exist in the repository's history. Fortunately, there are other tools that can entirely remove unwanted files from a repository's history. This article will explain how to use two of them: `git filter-branch` and the [BFG Repo-Cleaner](#).

Danger: Once you have pushed a commit to GitHub, you should consider any data it contains to be compromised. If you committed a password, change it! If you committed a key, generate a new one.

This article tells you how to make commits with sensitive data unreachable from any branches or tags in your GitHub repository. However, it's important to note that those commits may still be accessible in any clones or forks of your repository, directly via their SHA-1 hashes in cached views on GitHub, and through any pull requests that reference them. You can't do anything about existing clones or forks of your repository, but you can permanently remove all of your repository's cached views and pull requests on GitHub by contacting [GitHub support](#).

<https://help.github.com/articles/remove-sensitive-data/>



Reproducible research means:

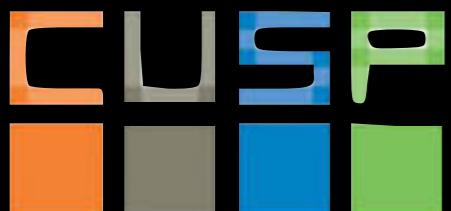
code

raw data

[https://github.com/fedhere/PUI2018_fb55/tree/master/HW3_fb55/
HW1_instructions.ipynb](https://github.com/fedhere/PUI2018_fb55/tree/master/HW3_fb55/HW1_instructions.ipynb)



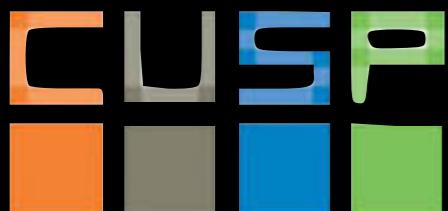
<https://help.github.com/articles/remove-sensitive-data/>



Reproducible research:

How to share your data

- Share/Reference the source of your raw data
- Share the “tidy” data
- Share the code used to process the data at each step



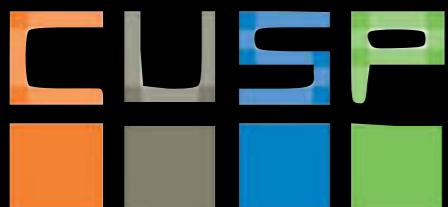
Reproducible research:

How to share your data

- Share/Reference the source of your raw data
- Share the “tidy” data
- Share the code used to process the data at each step

example:

[https://github.com/jakevdp/PythonicPerambulations/blob/master/
content/downloads/notebooks/ProntoData.ipynb](https://github.com/jakevdp/PythonicPerambulations/blob/master/content/downloads/notebooks/ProntoData.ipynb)



Reproducible research:

How to share your data

- Share/Reference the code used at each step
- Share the “tidy” data
- Share the code used at each step

example:

<https://github.com/jakevdp/PythonicPerambulations/blob/master/content/downloads/notebooks/ProntoData.ipynb>

Downloading Pronto's Data

We'll start by downloading the data (available on [Pronto's Website](#)) which you can do by uncommenting the following shell commands (the exclamation mark here is a special IPython syntax to run a shell command). The total download is about 70MB, and the unzipped files are around 900MB.

```
In [1]: # !curl -O https://s3.amazonaws.com/pronto-data/open_data_year_one.zip  
# !unzip open_data_year_one.zip
```

Next we need some standard Python package imports:

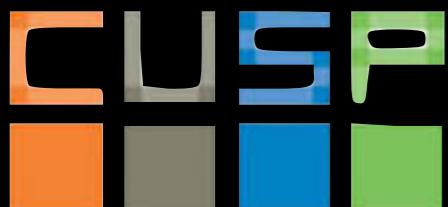
```
In [2]: %matplotlib inline  
import matplotlib.pyplot as plt  
import pandas as pd  
import numpy as np  
import seaborn as sns; sns.set()
```

And now we load the trip data with Pandas:

```
In [3]: trips = pd.read_csv('2015_trip_data.csv',  
                        parse_dates=['starttime', 'stoptime'],  
                        infer_datetime_format=True)  
trips.head()
```

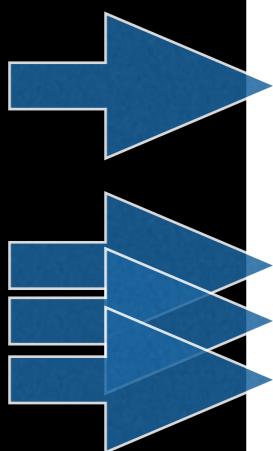
	trip_id	starttime	stoptime	bikeid	tripduration	from_station_name	to_station_name	fi
0	431	2014-10-13 10:31:00	2014-10-13 10:48:00	SEA00298	985.935	2nd Ave & Spring St	Occidental Park / Occidental Ave S & S Washington	C
1	432	2014-10-13 10:32:00	2014-10-13 10:48:00	SEA00195	926.375	2nd Ave & Spring St	Occidental Park / Occidental Ave S & S Washington	C
2	433	2014-10-13 10:33:00	2014-10-13 10:48:00	SEA00486	883.831	2nd Ave & Spring St	Occidental Park / Occidental Ave S & S Washington	C
3	434	2014-10-13 10:34:00	2014-10-13 10:48:00	SEA00333	865.937	2nd Ave & Spring St	Occidental Park / Occidental Ave S & S Washington	C
4	435	2014-10-13 10:34:00	2014-10-13 10:49:00	SEA00202	923.923	2nd Ave & Spring St	Occidental Park / Occidental Ave S & S Washington	C

Each row of this trip dataset is a single ride by a single person, and the data contains over 140,000 rows!



Reproducible research:

How to share your data



fedhere / Ulnotebooks

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Branch: master → Ulnotebooks / dataWrangling / Create new file Upload files Find file History

Bianco Federica add puidata Latest commit 5dcc922 12 minutes ago

...

PandasDataWrangling-Chap7.ipynb	moved notebooks from Lab2 repo	2 years ago
README.md	Update README.md	a year ago
RetrieveDataIntoPUIDATA.ipynb	removing tail	a year ago
aSimplePythonScript.py	aSimplePythonScript.py	2 years ago
aSimplePythonThatWritesToC...	adding script that writes to csv file	2 years ago
acquiringData.ipynb	updates...	a year ago
readingData.ipynb	updates...	a year ago
setupPUIDATA.ipynb	add puldata	12 minutes ago
twitterJson.py	adding twitter parser to json file	2 years ago

README.md

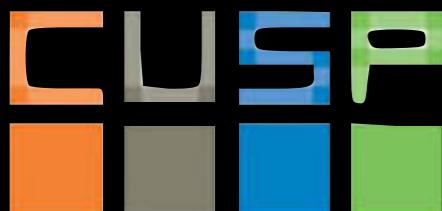
Notebooks for lecture 2 PUI 2016, 2017

Topic: Data Wrangling

Slides

Practical lecture Goals

- learn how to read csv into pandas data frame
- learn basic data frame manipulations
- learn how to read json
- learn how to get data from API
- learn how to write a python script that runs from the terminal



Reproducible research means:

Data are noisy

- Measurement errors
- Extraction errors
- Data entry errors



Make sure you do not make them noisier...

Typical data wrangling issues

Licensing issues/Privacy (some data unavailable)

Missing required field

Primary key violation (2 people - same social security number)

Parsing text into fields (separator issues)

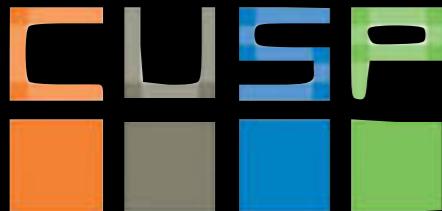
Naming conventions: NYC vs New York

Different representations (2 vs Two)

Fields corrupted (too long get truncated)

Formatting issues – especially dates

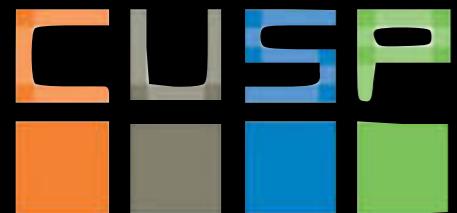
Redundant Records (exact match or other)



<http://www.cs.duke.edu> compsci216

II: Data Wrangling

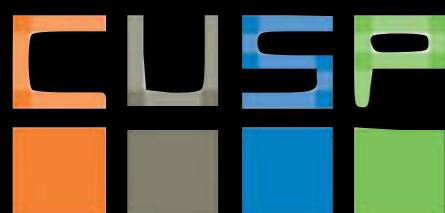
data ingestion



Types of Data Files:

- CSV comma separated values (also TSV - tab)
- JSON: corresponds to a Python data dictionary
- XML: similar to HTML

support nested structures



Types of Data Files:

- CSV comma separated values (also TSV - tab)

The screenshot shows a data visualization interface for the 'DHS Daily Report' dataset on NYC OpenData. The dataset tracks daily counts of families and individuals residing in shelter systems. The main view displays a table with columns: Date of Census, Total Adults in Shelter, Total Children in Shelter, and Total Individuals in Shelter. The data spans from September 11, 2016, to September 16, 2016. A modal window titled 'Export' is open, listing options: OData, Print, Download (selected), and CSV. The 'Download' option is described as 'Download a copy of this dataset in a static format'. The table has 7 rows of data.

Date of Census	Total Adults in Shelter	Total Children in Shelter	Total Individuals in Shelter
09/16/2016	37933	22347	60280
09/15/2016	37859	22294	60153
09/14/2016	37845	22243	60088
09/13/2016	37976	22264	60240
09/12/2016	38001	22308	60309
09/11/2016	37912	22257	60169

< Previous Next >

Showing Rows 1-50 out of 1,814

Manage More Views Filter Visualize Export Discuss Embed About

Find in this Dataset

Export

OData

Print

Download

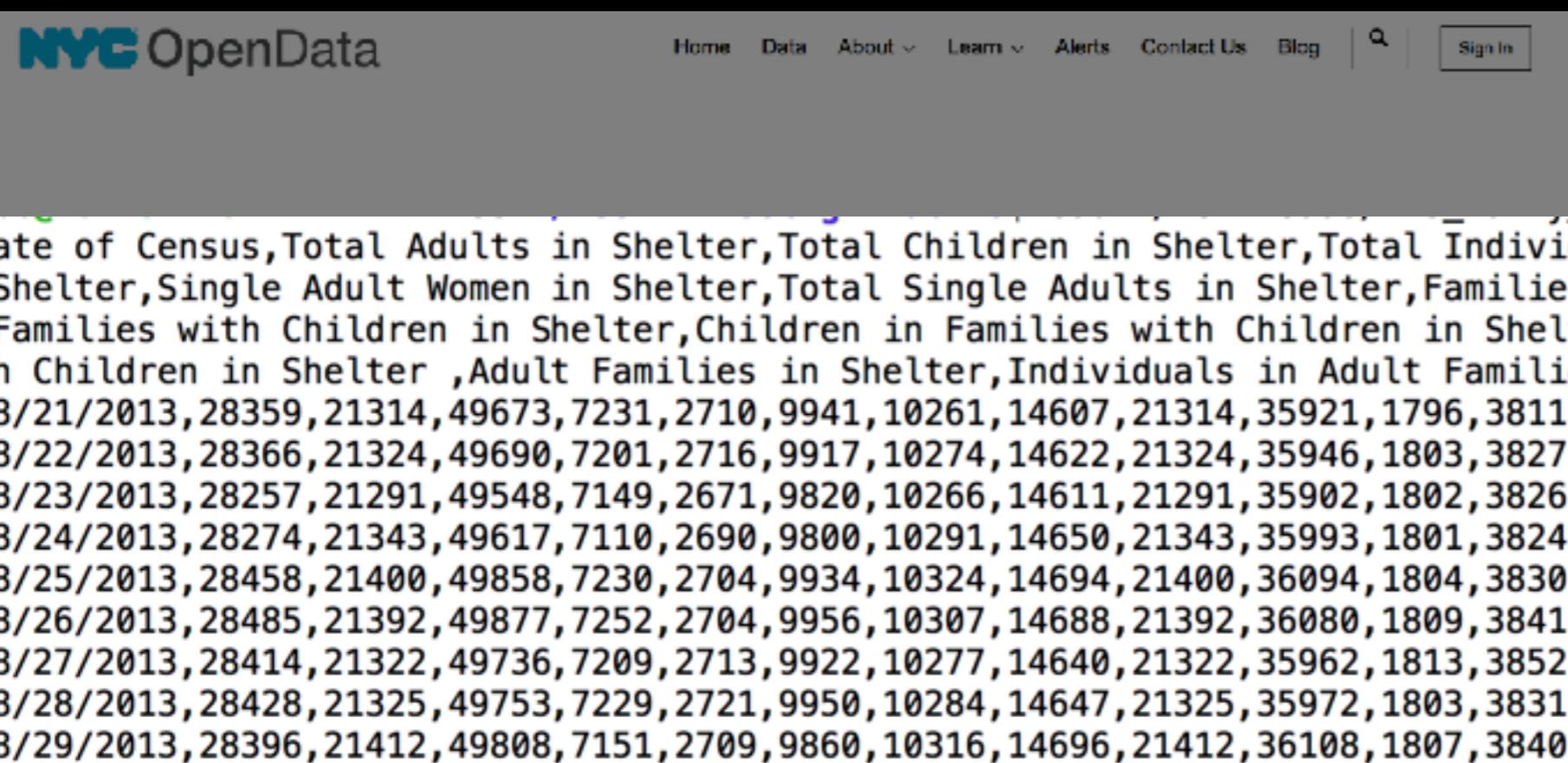
Download a copy of this dataset in a static format

CSV

CSV for Excel

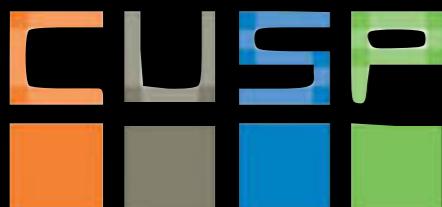
Types of Data Files:

- CSV comma separated values (also TSV - tab)



The screenshot shows a table from the NYC OpenData website. The columns are labeled with various shelter statistics: Date of Census, Total Adults in Shelter, Total Children in Shelter, Total Individuals in Shelter, Single Adult Men in Shelter, Single Adult Women in Shelter, Total Single Adults in Shelter, Families with Children in Shelter, Adults in Families with Children in Shelter, Children in Families with Children in Shelter, Total Individuals in Families with Children in Shelter, Adult Families in Shelter, and Individuals in Adult Families in Shelter. The data is presented as a series of rows, each corresponding to a specific date from August 21, 2013, to August 29, 2013.

Date of Census	Total Adults in Shelter	Total Children in Shelter	Total Individuals in Shelter	Single Adult Men in Shelter	Single Adult Women in Shelter	Total Single Adults in Shelter	Families with Children in Shelter	Adults in Families with Children in Shelter	Children in Families with Children in Shelter	Total Individuals in Families with Children in Shelter	Adult Families in Shelter	Individuals in Adult Families in Shelter
08/21/2013	28359	21314	49673	7231	2710	9941	10261	14607	21314	35921	1796	3811
08/22/2013	28366	21324	49690	7201	2716	9917	10274	14622	21324	35946	1803	3827
08/23/2013	28257	21291	49548	7149	2671	9820	10266	14611	21291	35902	1802	3826
08/24/2013	28274	21343	49617	7110	2690	9800	10291	14650	21343	35993	1801	3824
08/25/2013	28458	21400	49858	7230	2704	9934	10324	14694	21400	36094	1804	3830
08/26/2013	28485	21392	49877	7252	2704	9956	10307	14688	21392	36080	1809	3841
08/27/2013	28414	21322	49736	7209	2713	9922	10277	14640	21322	35962	1813	3852
08/28/2013	28428	21325	49753	7229	2721	9950	10284	14647	21325	35972	1803	3831
08/29/2013	28396	21412	49808	7151	2709	9860	10316	14696	21412	36108	1807	3840



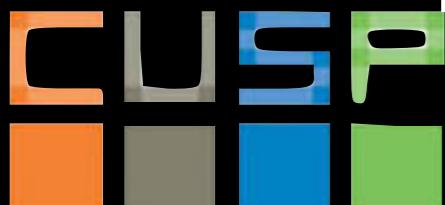
Types of Data Files:

- CSV comma separated values (also TSV - tab)

NYC OpenData

```
#print the first 8 lines out  
dhs.head(8)
```

	Date of Census	Total Adults in Shelter	Total Children in Shelter	Total Individuals in Shelter	Single Adult Men in Shelter	Single Adult Women in Shelter	Total Single Adults in Shelter	Families with Children in Shelter	Adults in Families with Children in Shelter	Children in Families with Children in Shelter	Total Individuals in Families with Children in Shelter	Adult Families in Shelter	Indivi in Famil Si
0	08/21/2013	28359	21314	49673	7231	2710	9941	10261	14607	21314	35921	1796	
1	08/22/2013	28366	21324	49690	7201	2716	9917	10274	14622	21324	35946	1803	
2	08/23/2013	28257	21291	49548	7149	2671	9820	10266	14611	21291	35902	1802	
3	08/24/2013	28274	21343	49617	7110	2690	9800	10291	14650	21343	35993	1801	
4	08/25/2013	28458	21400	49858	7230	2704	9934	10324	14694	21400	36094	1804	
5	08/26/2013	28485	21392	49877	7252	2704	9956	10307	14688	21392	36080	1809	
6	08/27/2013	28414	21322	49736	7209	2713	9922	10277	14640	21322	35962	1813	
7	08/28/2013	28428	21325	49753	7229	2721	9950	10284	14647	21325	35972	1803	



https://github.com/fedhere/PUI2018_fb55/blob/master/HW3_fb55/Assignment2_example.ipynb

II: Data Wrangling

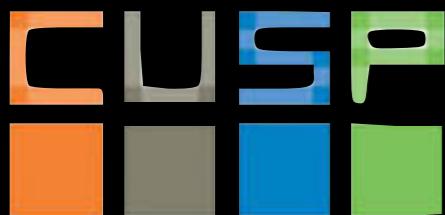
Reproducible research means:

code raw data

Raw data formats: API, Json, binary, paper...

Tidy data: tabulated data, binary, csv, tsv, ascii,
Excel (ugh!) or (object oriented) Json,...

	var 1	var 2
obs 1	7.4	9.2
obs 2	NaN	13.1



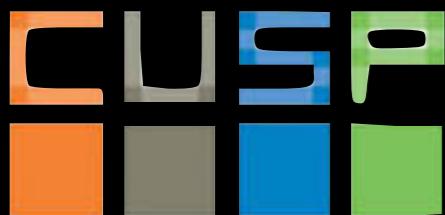
Reproducible research means:

code raw data

Raw data formats: API, Json, binary, paper...

Tidy data: tabulated data, binary, csv, tsv, ascii,
Excel (ugh!) or (object oriented) Json,...

	var 1	var 2
obs 1	7.4	9.2
obs 2	NaN	13.1



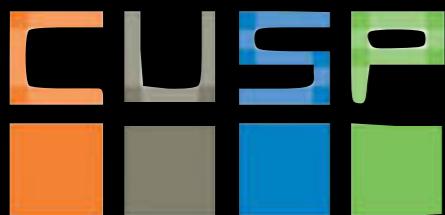
Reproducible research means:

code raw data

Raw data formats: API, Json, binary, paper...

Tidy data: tabulated data, binary, csv, tsv, ascii,
Excel (ugh!) or (object oriented) Json,...

	var 1	var 2	link
obs 1	7.4	9.2	URL 1
obs 2	NaN	13.1	URL 1



Reproducible research means:

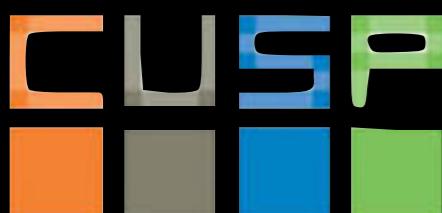
code raw data

Raw data formats: API, Json, binary, paper...

Tidy data: tabulated data, binary, csv, tsv, ascii,
Excel (ugh!) or (object oriented) Json,...

	var 1	var 2	link
obs 1	7.4	9.2	URL 1
obs 2	Nan	13.1	URL 1

never use
fake values
for missing
data!



Reproducible research means:

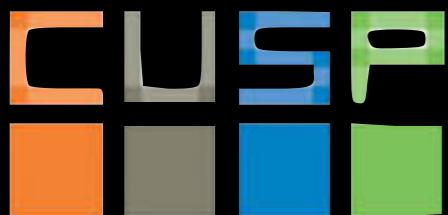
code raw data

Raw data formats: API, Json, binary, paper...

Tidy data: tabulated data, binary, csv, tsv, ascii, Excel (ugh!) or (object oriented) Json,...

Report separately :

- **Details about variables:** e.g. date: “date when test was performed”
- **Units:** remember the Mars Rover that was lost at sea cause scientists got Metric and Imperial units mixed up?!?!



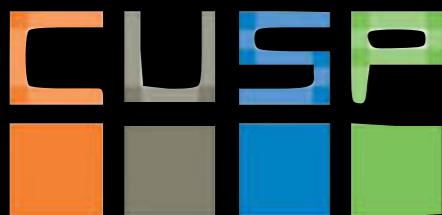
Reproducible research means:

code raw data

Raw data formats: API, Json, binary, paper...

Tidy data: tabulated data, binary, csv, tsv, ascii, Excel (ugh!) or (object oriented) Json,...

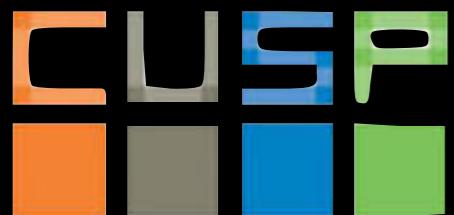
	speed (m/s)	distance (m)	link (accessed date)
obs 1	7.4	9.2	URL 1
obs 2	NaN	13.1	URL 1





loading data on a computer

[https://github.com/fedhere/UInotebooks/blob/master/
dataWrangling/readingData.ipynb](https://github.com/fedhere/UInotebooks/blob/master/dataWrangling/readingData.ipynb)



II: Data Wrangling



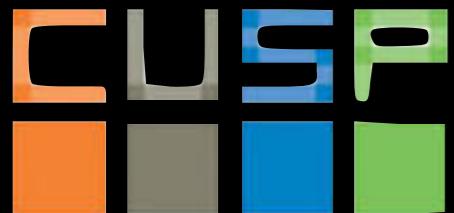
Python for Data Analysis (Pandas manual) Chapter 7

[https://github.com/fedhere/UInotebooks/blob/master/dataWrangling/
PandasDataWrangling-Chap7.ipynb](https://github.com/fedhere/UInotebooks/blob/master/dataWrangling/PandasDataWrangling-Chap7.ipynb)



II: Data Wrangling

data types



Introduction to Statistics

Online Edition

Primary author and editor: David M. Lanel

Introduction to Statistics

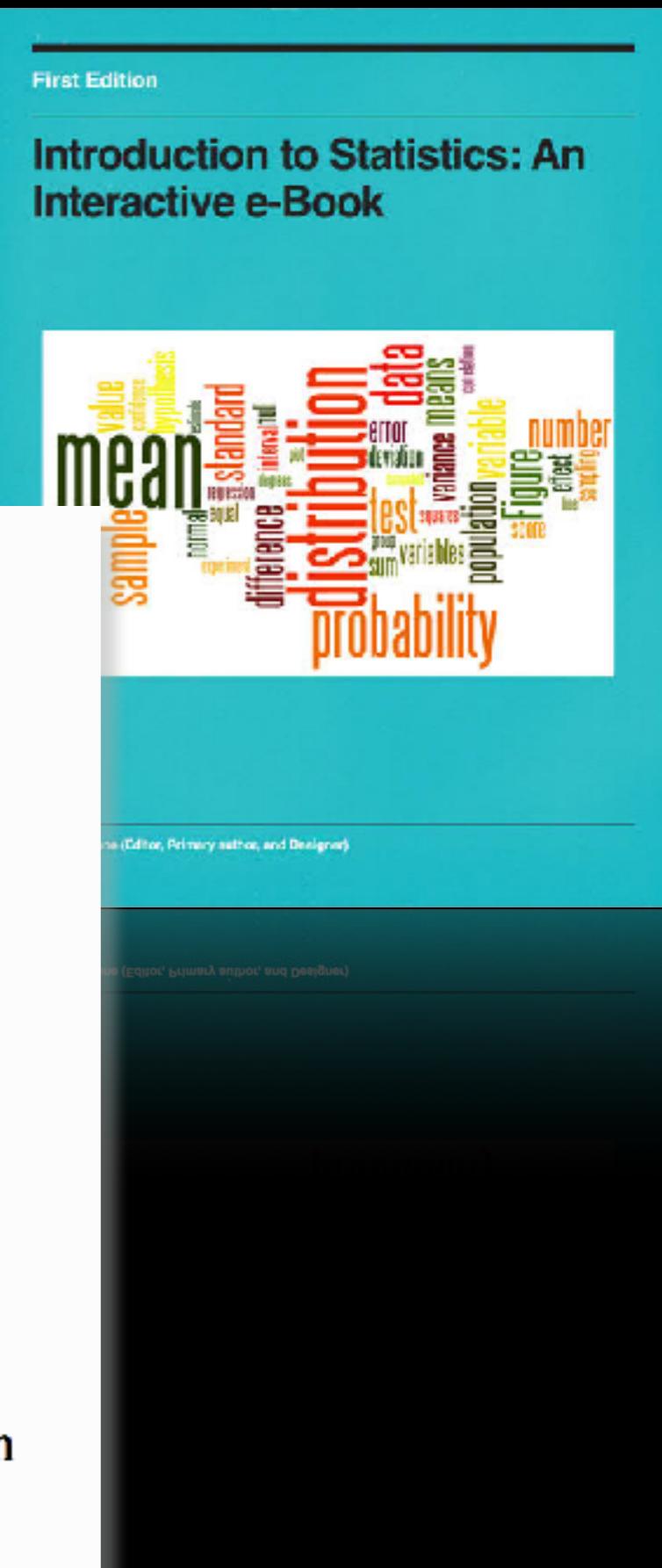
Online Edition

Primary author and editor:
David M. Lane¹

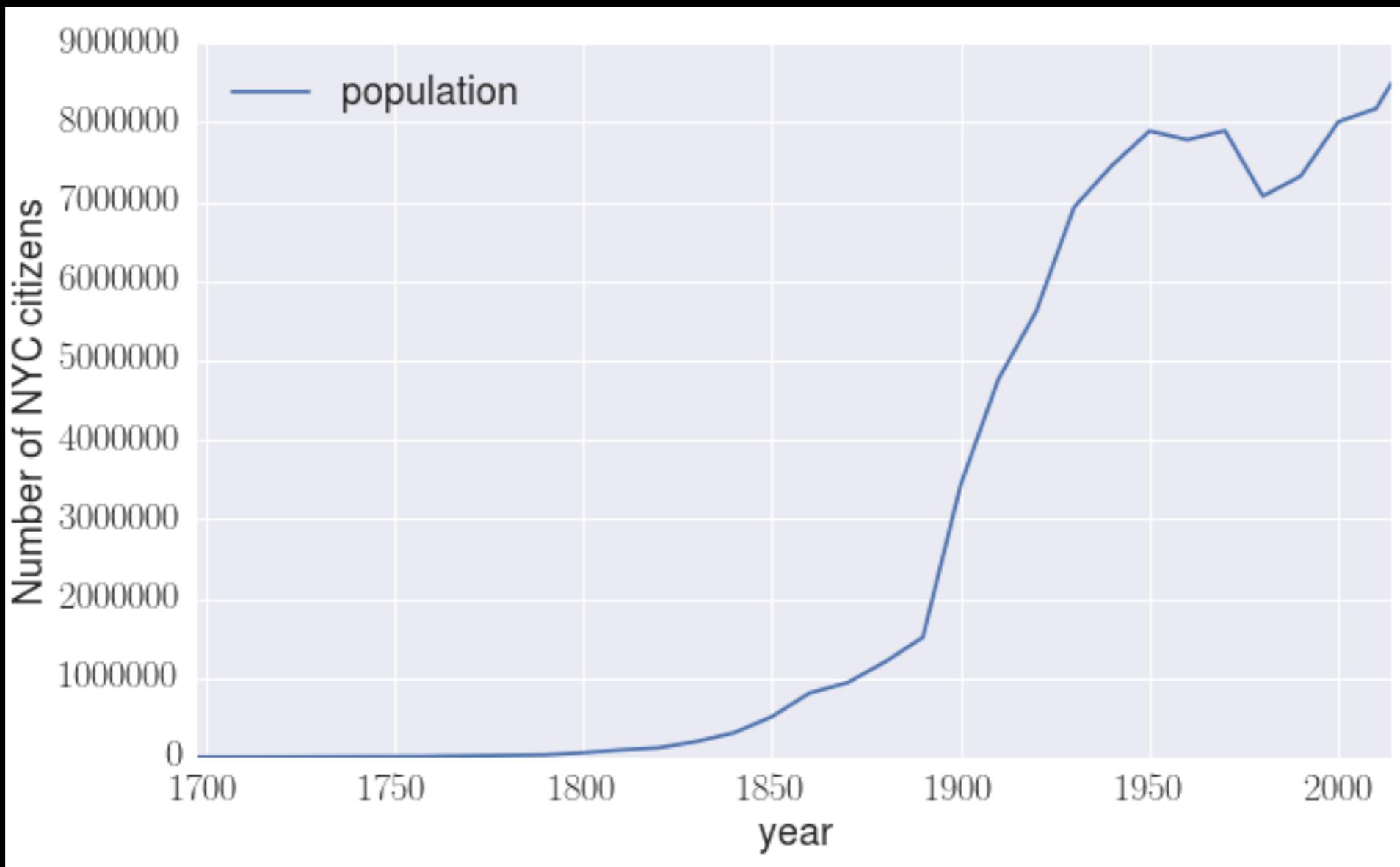
Other authors:

David Scott¹, Mikki Hebl¹, Rudy Guerra¹, Daniel Osherson¹, and Heidi Zimmer²

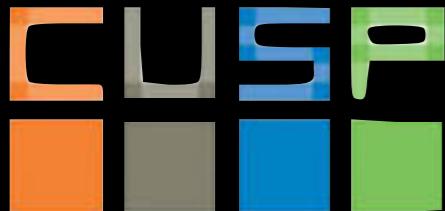
¹Rice University; ²University of Houston, Downtown Campus



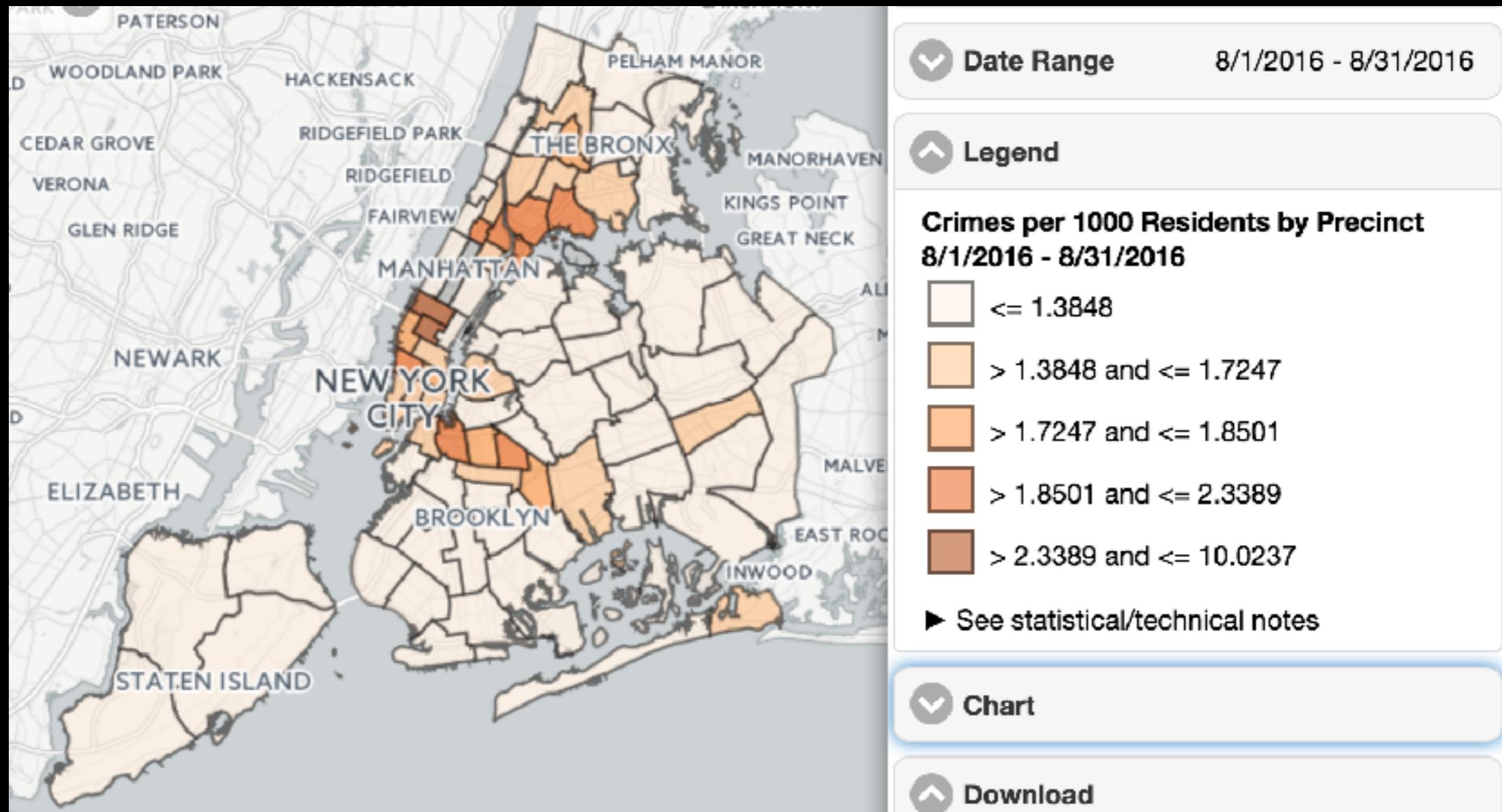
Dependent vs Independent Variable



<http://flowingdata.com/2016/09/12/watch-bacteria-evolve-resistance-to-antibiotic/>



Dependent vs Independent Variable

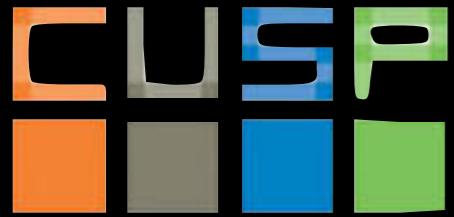


<https://maps.nyc.gov/crime/>

Dependent vs Independent Variable

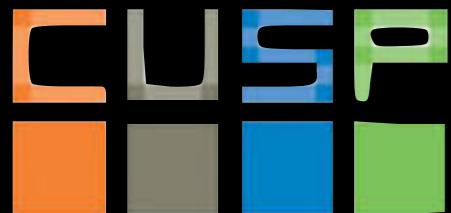


<http://flowingdata.com/2016/09/12/watch-bacteria-evolve-resistance-to-antibiotic/>



Dependent vs Independent Variable

Levels of an independent variable:
the number of experimental conditions.
e.g.: control test and experiment are 2 levels



Types of Data: *Qualitative* variables

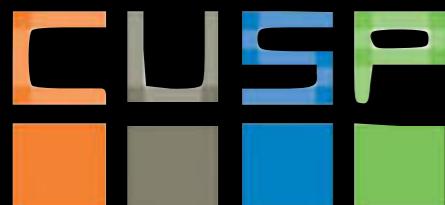
- No ordering

UrbanScience e.g. precinct, state, gender
Also called *Nominal, Categorical*

Types of Data: *Quantitative* variables

- Ordering is meaningful

Time, Distance, Age, Length, Intensity, Satisfaction



Types of Data: *Quantitative* variables

- Continuous: _____
- Discrete: • • • • • • • • • • • • • • •

Types of Data: *Quantitative* variables

- **Continuous:** distance to the closest park
- **Discrete:** *any* countable, e.g. number of crimes

Discrete data may be:

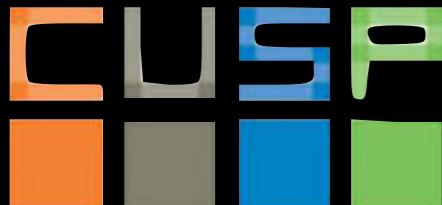
- **Counts:** number of bacteria at time t in section A
- **Ordinal:** survey response Good/Fair/Poor

Continuous data may be:

- **Continuous Ordinal:** Earthquakes (not linear scale)
- **Interval:** F temperature - interval size preserved
- **Ratio:** Car speed - 0 is naturally defined

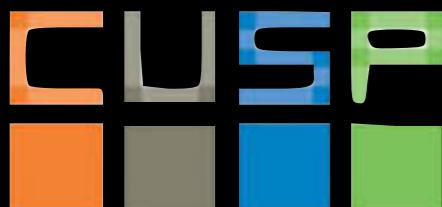
Data may also be:

- **Censored:** age > 90
- **Missing:** “Prefer not to answer” (NA / NaN)



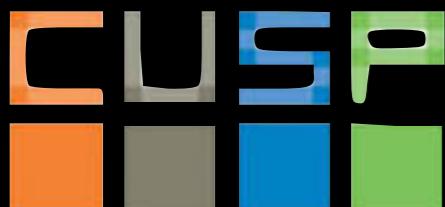
Data Definitions

- **Data:** observations that have been collected
- **Population:** the complete body of subjects we want to infer about
- **Sample:** the subset of the population we actually studied
- **Census:** collection of data from the entire population
- **Parameter:** numerical value describing an attribute of the *population*
- **Statistics:** numerical value describing an attribute of the *sample*



Data Definitions

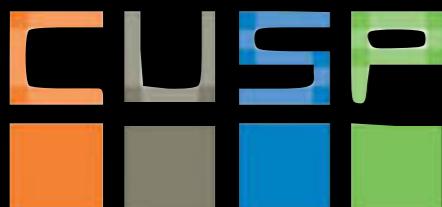
The analysis of our _____
showed that for our 10 _____ the mean height is 6.4 ft.
The standard deviation of the _____ means is 0.5 ft.
From this _____ we infer for the _____ a mean
height _____ $6.4 +/ - 5$ ft



Data Definitions

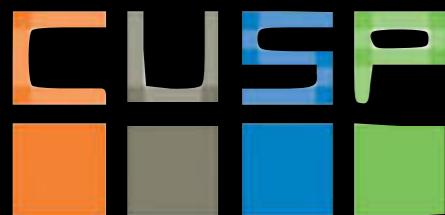
The analysis of our _____ showed that for our 10 _____ the mean height is 6.4 ft. The standard deviation of the _____ means is 0.5 ft. From this _____ we infer for the _____ a mean height _____ 6.4 ± 0.5 ft

sample parameter data population statistics



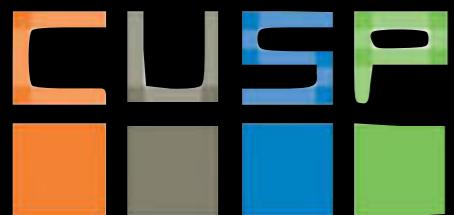
Data Definitions

The analysis of our **data** showed that for our 10 **samples** the mean height is 6.4 ft. The standard deviation of the **sample** means is 0.5 ft. From this **statistic** we infer for the **population** a mean height **parameter** 6.4 ± 0.5 ft





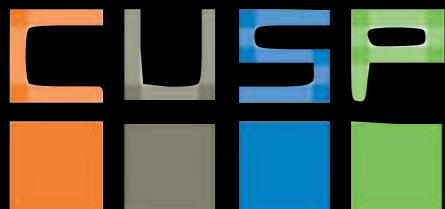
basic data imputation and
manipulation Lab:
FormattingTables.ipynb



II: Data Wrangling

How to acquire data

- Downloadable files
- API *Application Program Interface*
- Databases (MySQL, MongoDB...
Prof. Huy Vo guest lecture)



How to acquire data

- **Downloadable files**

PROS: easy to do
reproducibility: you can provide the data

CONS: store the data locally
(bad if large)

- **API Application Program Interface**

PROS: works directly with the source

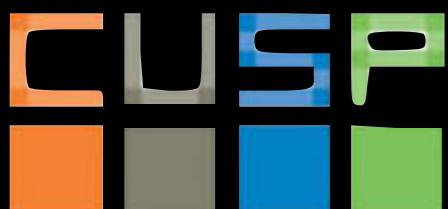
CONS: store the data locally
reproducibility: relies on data source to be permanent & API stable and accessible

- **Databases (MySQL, MongoDB...)**

Prof. Huy Vo guest lecture)

PROS: fast and flexible
can query only data of interest
reproducibility : can host remotely & provide access

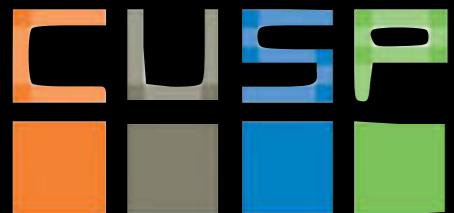
CONS: must build a database





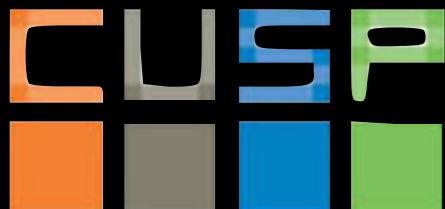
Downloading files directly
in your notebook

[https://github.com/fedhere/UInotebooks/blob/master/
dataWrangling/acquiringData.ipynb](https://github.com/fedhere/UInotebooks/blob/master/dataWrangling/acquiringData.ipynb)



Types of Data Files:

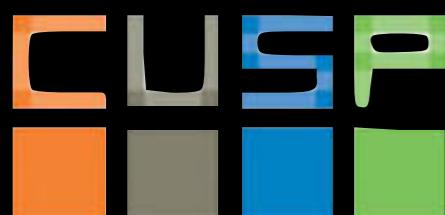
- CSV comma separated values (also TSV - tab)
- JSON: corresponds to a Python data dictionary
- XML: symilar to HTML



Types of Data Files:

- CSV comma separated values (also TSV - tab)
- JSON: corresponds to a Python data dictionary
- XML: similar to HTML

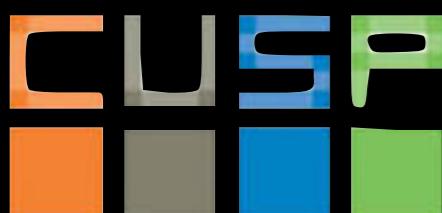
support nested structures



Types of Data Files:

- CSV comma separated values (also TSV - tab)
- JSON: corresponds to a Python data dictionary
- XML: similar to HTML

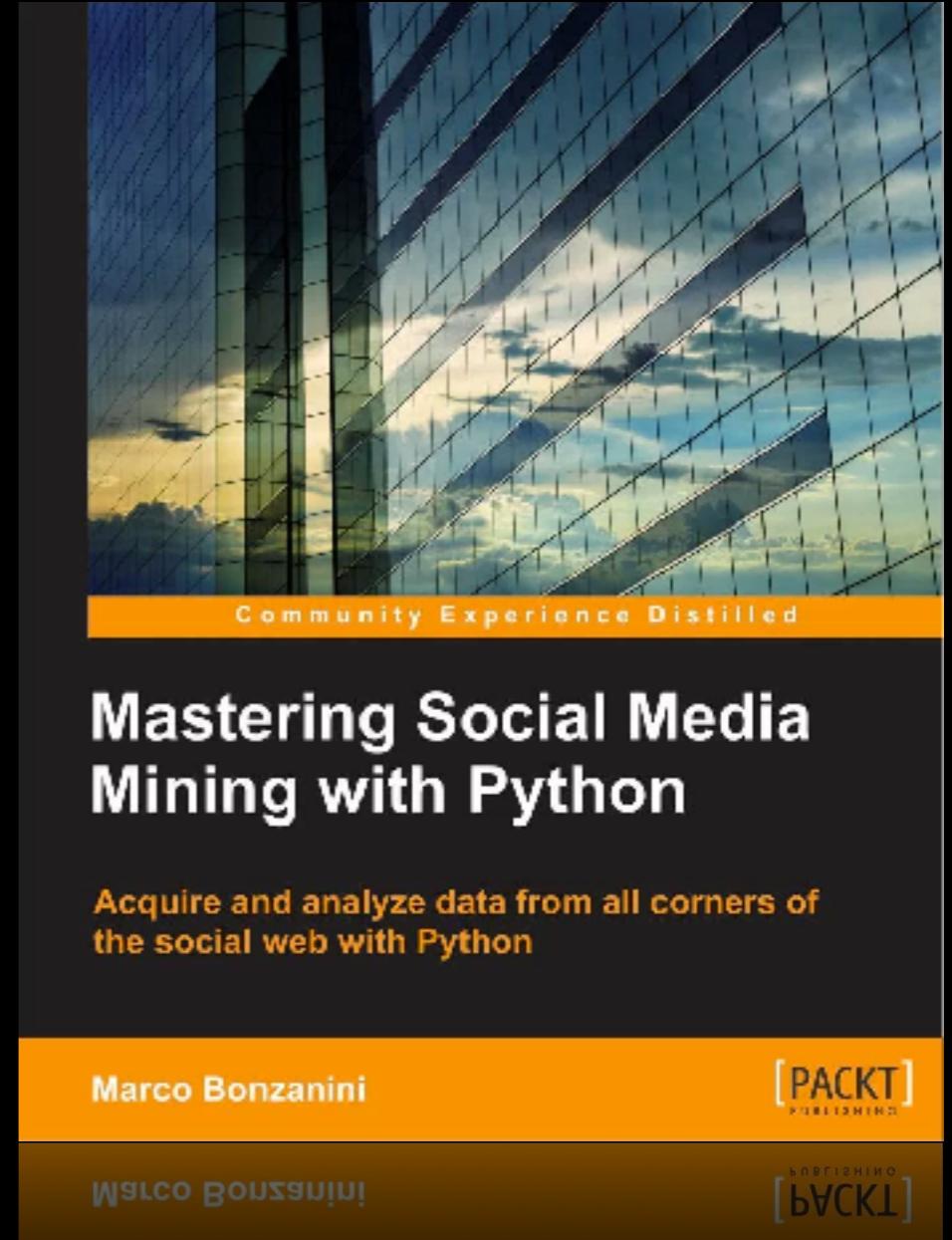
support nested structures



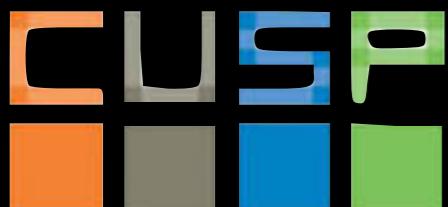
[https://github.com/fedhere/
PUI2016_fb55/blob/master/Lab2_fb55/
twitterJson.py](https://github.com/fedhere/PUI2016_fb55/blob/master/Lab2_fb55/twitterJson.py)



download an
online, nested
JSON file



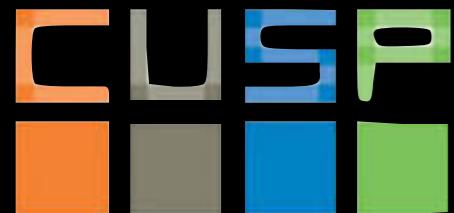
[https://marcobonzanini.com/
2015/03/02/mining-twitter-data-
with-python-part-1/](https://marcobonzanini.com/2015/03/02/mining-twitter-data-with-python-part-1/)



II: Data Wrangling

Access data through API

<http://openweathermap.org/api>

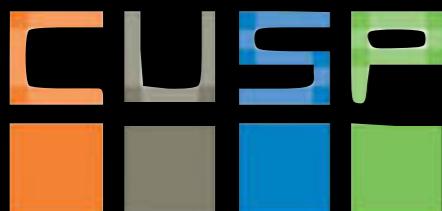




API scraping and JSON manipulation lab

Key Concepts:

- importance of reproducibility
 - data ethics and data curation
 - types of variables
 - CSV, JSON file formats properties
-
- DATA Facility data access
 - API data access
 - JSON data manipulation
 - working with Pandas Dataframes



Resources:

<https://github.com/fedhere/UInotebooks/blob/master/dataWrangling>

D.Lane 2014

Introduction to Statistics Chapters 1.6-1.8 & 6.1-6.3

<http://onlinestatbook.com/>

Mohit Sharma 2016

CUSP UCSL tutorials Python Dictionaries

<https://sharmamohit.com/tutorials/ucsl/Python-Dictionaries/>

Allen B. Downey 2012

Thinking Python (free online) Appendix A ASSIGNED READING

<http://greenteapress.com/thinkpython/html/thinkpython021.html>

Wes McKinney 2013

Pandas for Data Analysis Chapter 7

<http://tinyurl.com/hydko7p>

