

# Applied Data Analysis (CS401)



Lecture 2  
Handling data  
28 Sep 2022

**EPFL**

**Robert West**



# Announcements

- **Register** your teams (4 people) [here](#) by Fri 7 Oct
  - Each team member must individually complete the form!
- **Project milestone P1** to be released this Fri, due Fri 14 Oct
- **First quiz** (previously called “Q $\theta$ ”, now “Q1”\*) to be held in this Friday’s lab session
  - Test run, to make sure everything works smoothly
  - Won’t count towards grade
- **Friday’s lab session:**
  - Intro to Pandas (very important for Homework H1 and rest of course)

\* This way, Q $i$  is always about material from week  $i$ .

# Feedback

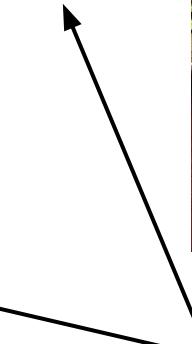
Give us feedback on this lecture here:

<https://go.epfl.ch/ada2022-lec2-feedback>

Feedback form available for each lecture and lab session

- What did you (not) like about this lecture?
- What was (not) well explained?
- On what would you like more (fewer) details?
- Is it nicer to follow the lecture online or offline?
- ...

# Cooking with data



Part 2:  
Data sources

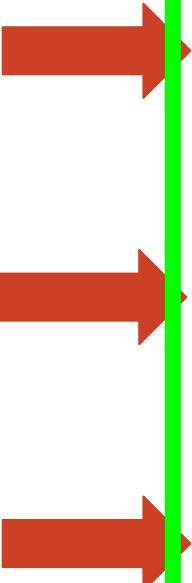
Part 1:  
Data models

Part 3:  
Data wrangling

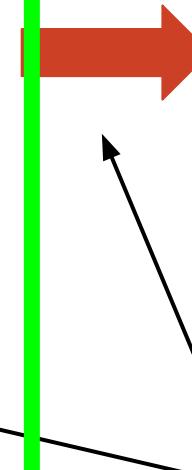
# Cooking with data



Part 2:  
Data sources



Part 1:  
**Data models**



Part 3:  
Data wrangling



WIKIPEDIA  
The Free Encyclopedia

Article Talk

Not logged in Talk Contributions Create account Log in

Read Edit View history

Search Wikipedia



## Data model

From Wikipedia, the free encyclopedia

A **data model** (or **datamodel**)<sup>[1][2][3][4][5]</sup> is an **abstract model** that organizes elements of **data** and standardizes how they relate to one another and to the properties of real-world entities. For instance, a data model may specify that the data element representing a car be composed of a number of other elements which, in turn, represent the color and size of the car and define its owner.

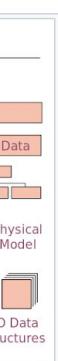
The term **data model** can refer to two distinct but closely related concepts. Sometimes it refers to an abstract formalization of the objects and relationships found in a particular application domain: for example the customers, products, and orders found in a manufacturing organization. At other times it refers to the set of concepts used in defining such formalizations: for example concepts such as entities, attributes, relations, or

Bob's definition: A data model specifies how you think about the world

4 topics

- 4.1 Data architecture
- 4.2 Data modeling
- 4.3 Data properties

functional specification to aid a computer software make-or-buy decision. The figure is an example of the interaction between process and data models.<sup>[6]</sup>



ed on  
int. A  
ed, and  
of  
ation of a

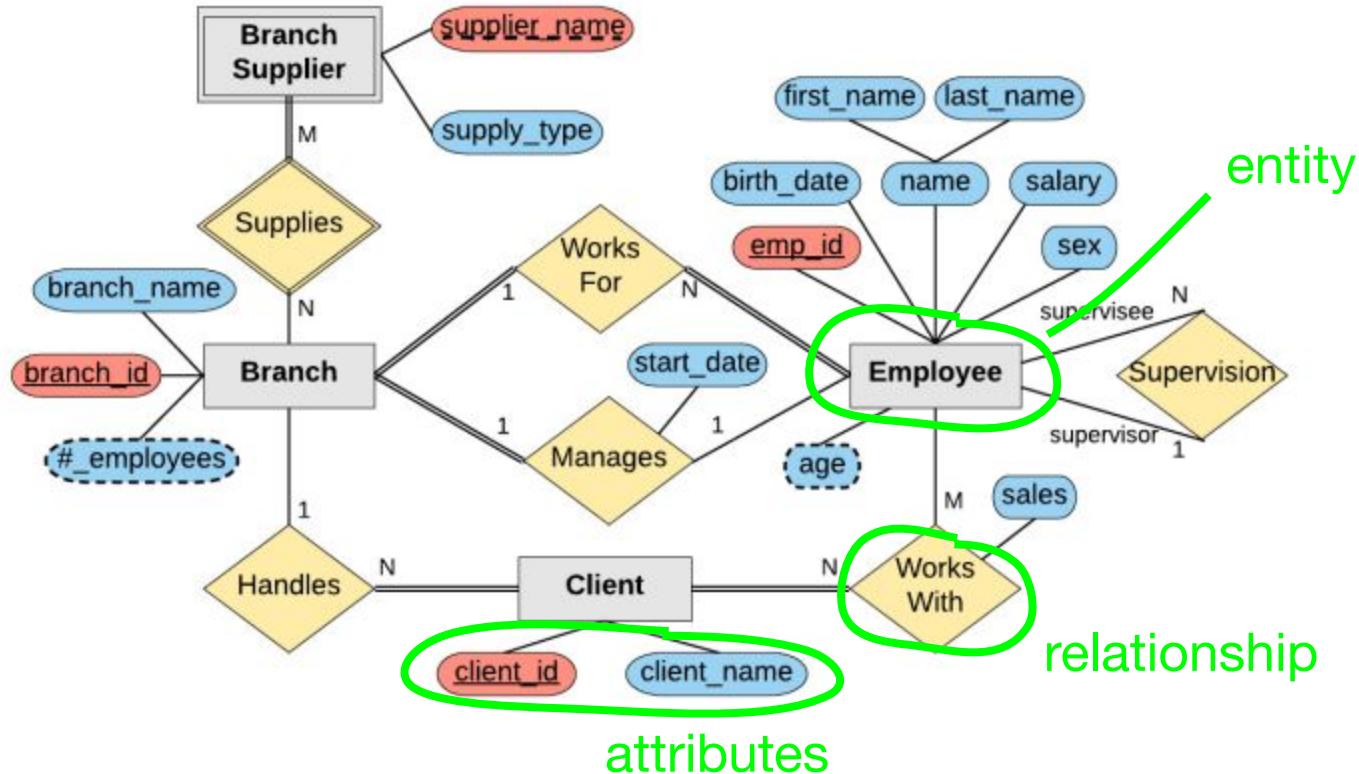
In other projects

Wikimedia Commons

Languages



Q: “How do you think about the world?”  
A: “See my entity–relationship diagram!”



Q: “How do you think about the world?”

A: “See my entity–relationship diagram!”

How to store my  
data on a computer?



# Q1: “How should I store my data on a computer?”

## Q2: “How do I think about the world?”

- “The world is simple: one type of entity, all with the same attributes”

→ **Flat model**

```
66.249.65.107 - - 08/Oct/2007:04:54:20 -0400] "GET /support.html  
HTTP/1.1" 200 1117  
+http://www.google.com/bot.html"
```

- “The world contains many types of entities, connected by relationships”

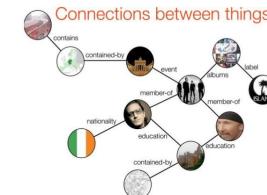
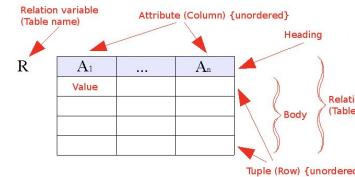
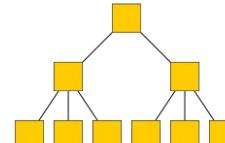
→ **Relational model**

- “The world is a hierarchy of entities”

→ **Document model**

- “The world is a complex network of entities”

→ **Network model**



# Flat model

- Example: log files; e.g., Apache web server (httpd)
  - Entities = requests from clients to server

```
66.249.65.107 - - [08/Oct/2007:04:54:20 -0400] "GET /support.html  
HTTP/1.1" 200 11179 "-" "Mozilla/5.0 (compatible; Googlebot/2.1;  
+http://www.google.com/bot.html)"
```

```
111.111.111.111 - - [08/Oct/2007:11:17:55 -0400] "GET / HTTP/1.1" 200  
10801  
"http://www.google.com/search?q=in+love+with+ada+lovelace+what+to+do&ie=ut  
f-8&oe=utf-8&aq=t&rls=org.mozilla:en-US:official&client=firefox-a"  
"Mozilla/5.0 (Windows; U; Windows NT 5.2; en-US; rv:1.8.1.7)  
Gecko/20070914 Firefox/2.0.0.7"
```

- Another common format: CSV (“comma-separated vector”)

# Q1: “How should I store my data on a computer?”

## Q2: “How do I think about the world?”

- “The world is simple: one type of entity, all with the same attributes”

→ **Flat model**

```
66.249.65.107 - - [08/Oct/2007:04:54:20 -0400] "GET /support.html  
HTTP/1.1" 200 11179 "-" "Mozilla/5.0 (compatible; Googlebot/2.1;  
+http://www.google.com/bot.html)"
```

- “The world contains many types of entities, connected by relationships”

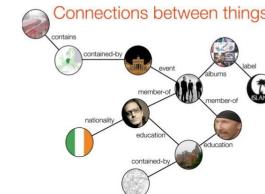
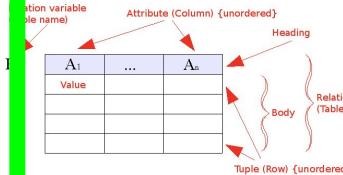
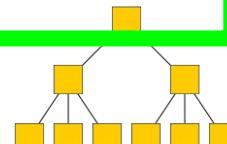
→ **Relational model**

- “The world is a hierarchy of entities”

→ **Document model**

- “The world is a complex network of entities”

→ **Network model**



# Relational model

- “The world contains many types of entities, connected by relationships”
- The relational model is ubiquitous:
  - MySQL, PostgreSQL, Oracle, DB2, SQLite, ...
  - You use it many times every day
- Data represented as tables (“relations”) describing
  - entities,
  - relationships between entities

id	name
1	Bush
2	Trump
3	Obama

president	succes sor
1	3
3	2

# Processing data in the relational model: SQL

- *Declarative* language for core data manipulations
- You think about what you want, not how to compute it

Imperative

```
//dogs = [{name: 'Fido', owner_id: 1}, {...}, ...]  
//owners = [{id: 1, name: 'Bob'}, {...}, ...]  
  
var dogsWithOwners = []  
var dog, owner  
  
for(var di=0; di < dogs.length; di++) {  
    dog = dogs[di]  
  
    for(var oi=0; oi < owners.length; oi++) {  
        owner = owners[oi]  
        if (owner && dog.owner_id == owner.id) {  
            dogsWithOwners.push({  
                dog: dog,  
                owner: owner  
            })  
        }  
    }  
}
```

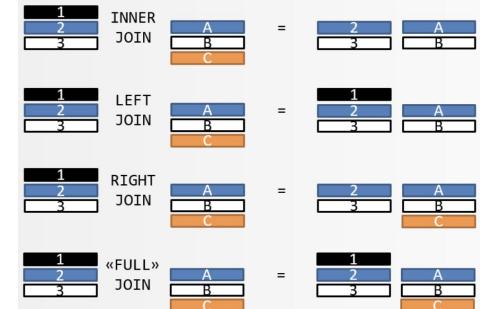
Declarative

```
SELECT * from dogs  
INNER JOIN owners  
WHERE dogs.owner_id = owners.id
```

# SQL

```
SELECT * from dogs  
INNER JOIN owners  
WHERE dogs.owner_id = owners.id
```

- You should know basics of SQL
- Need a refresher? → Watch/do online tutorials!
- Key concepts:
  - Select (!), update, delete
  - Unique keys
  - Joins (inner, left outer, right outer, full)
  - Sorting
  - Aggregation (group by, count, min, max, avg, etc.)





## POLLING TIME

- “Have you worked with SQL joins?”
- Scan QR code or go to <https://web.speakup.info/room/join/66626>



# SQL implementations



etc.

```
#!/usr/bin/python

import MySQLdb

# Open database connection
db = MySQLdb.connect("localhost","testuser","test123","TESTDB" )

# prepare a cursor object using cursor() method
cursor = db.cursor()

sql = "SELECT * FROM EMPLOYEE \
      WHERE INCOME > '%d'" % (1000)
try:
    # Execute the SQL command
    cursor.execute(sql)
    # Fetch all the rows in a list of lists.
    results = cursor.fetchall()
    for row in results:
        fname = row[0]
        lname = row[1]
        age = row[2]
        sex = row[3]
        income = row[4]
        # Now print fetched result
        print "fname=%s, lname=%s, age=%d, sex=%s, income=%d" % \
              (fname, lname, age, sex, income )
except:
    print "Error: unable to fetch data"

# disconnect from server
db.close()
```

# SQL and “SQL”

- The declarative-programming principles of SQL are widespread, even where it's less obvious

# “SQL”: Pandas (Python library)

- Similar to SQL (declarative), with additional elements of functional programming (map(), filter(), etc.)
- SQL “table”  $\longleftrightarrow$  Pandas “DataFrame”
- Need intro? Come to Friday’s lab session!

# Pandas vs. SQL

- + Pandas is lightweight and fast.
- + Natively Python, i.e., full SQL expressiveness plus the expressiveness of Python, especially for function evaluation.
- + Integration with plotting functions like Matplotlib.
  
- In Pandas, tables must fit into memory.
- No post-load indexing functionality: indices are built when a table is created.
- No transactions, journaling, etc. (matters for parallel applications)
- Large, complex joins are slower.

# “SQL”: Unix command line

```
cat users.txt \
| awk '$2 >= 18 && $2 <= 25' \
| join -1 1 -2 1 url_visits.txt - \
| cut -f 4 \
| sort \
| uniq -c \
| sort -k 1,1 -n -r \
| head -n 5
```

# Q1: “How should I store my data on a computer?”

## Q2: “How do I think about the world?”

- “The world is simple: one type of entity, all with the same attributes”

→ **Flat model**

- “The world contains many types of entities, connected by relationships”

→ **Relational model**

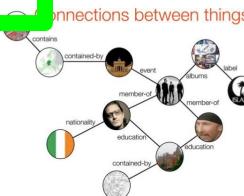
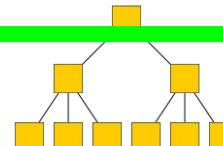
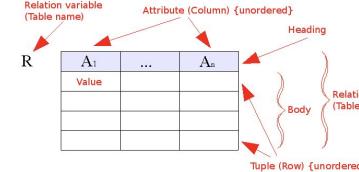
- “The world is a hierarchy of entities”

→ **Document model**

- “The world is a complex network of entities”

→ **Network model**

```
66.249.65.107 - - [08/Oct/2007:04:54:20 -0400] "GET /support.html  
HTTP/1.1" 200 11179 "-" "Mozilla/5.0 (compatible; Googlebot/2.1;  
+http://www.google.com/bot.html)"
```



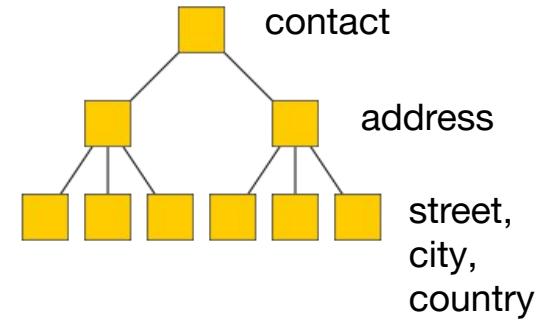
# Document model

- “The world is a hierarchy of entities”
- XML format:

```
<contact>
  <id>656</id>
  <firstname>Chuck</firstname>
  <lastname>Smith</lastname>
  <phone>(123) 555-0178</phone>
  <phone>(890) 555-0133</phone>
  <address>
    <street>Rue de l'Ale 8</street>
    <city>Lausanne</city>
    <zip>1007</zip>
    <country>CH</country>
  </address>
</contact>
```

- JSON format:

```
contact: {
  id: 656,
  firstname: "Chuck",
  lastname: "Smith",
  phones: ["(123) 555-0178",
            "(890) 555-0133"],
  address: {
    street: "Rue de l'Ale 8",
    city: "Lausanne",
    zip: 1007,
    country: "CH"
  }
}
```



- Document model

```
<contact>
  <id>656</id>
  <firstname>Chuck</firstname>
  <lastname>Smith</lastname>
  <phone>(123) 555-0178</phone>
  <phone>(890) 555-0133</phone>
  <address>
    <street>Rue de l'Ale 8</street>
    <city>Lausanne</city>
    <zip>1007</zip>
    <country>CH</country>
  </address>
</contact>
```

## Think for a minute:

If we want to use a relational DB (e.g., MySQL)  
instead of XML, how can we store  
2 phone numbers for the same person?

(Feel free to discuss with your neighbor.)

# Solution to “Think for a minute”

- Document model

```
<contact>
  <id>656</id>
  <firstname>Chuck</firstname>
  <lastname>Smith</lastname>
  <phone>(123) 555-0178</phone>
  <phone>(890) 555-0133</phone>
  <address>
    <street>Rue de l'Ale 8</street>
    <city>Lausanne</city>
    <zip>1007</zip>
    <country>CH</country>
  </address>
</contact>
```

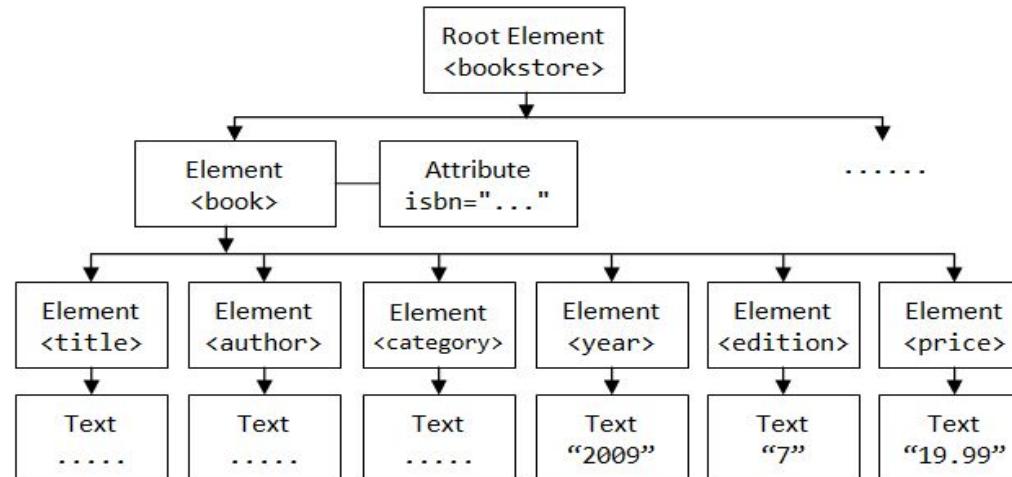
- Same in relational model

<b>id</b>	<b>first name</b>	...
656	Chuck	...
...	...	...

<b>id</b>	<b>phone</b>
656	(123) 555-0178
656	(890) 555-0133
...	...

# Processing XML and JSON

- Document structure = tree
- Processing via tree traversal (depth- or breadth-first search)
- Or use proper query language, such as [XQuery](#) or [jq](#)



# Q1: “How should I store my data on a computer?”

## Q2: “How do I think about the world?”

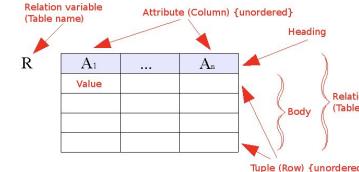
- “The world is simple: one type of entity, all with the same attributes”

→ **Flat model**

```
66.249.65.107 - - [08/Oct/2007:04:54:20 -0400] "GET /support.html  
HTTP/1.1" 200 11179 "-" "Mozilla/5.0 (compatible; Googlebot/2.1;  
+http://www.google.com/bot.html)"
```

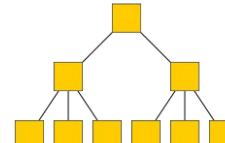
- “The world contains many types of entities, connected by relationships”

→ **Relational model**



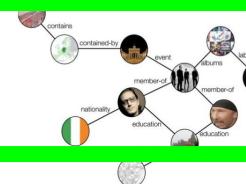
- “The world is a hierarchy of entities”

→ **Document model**



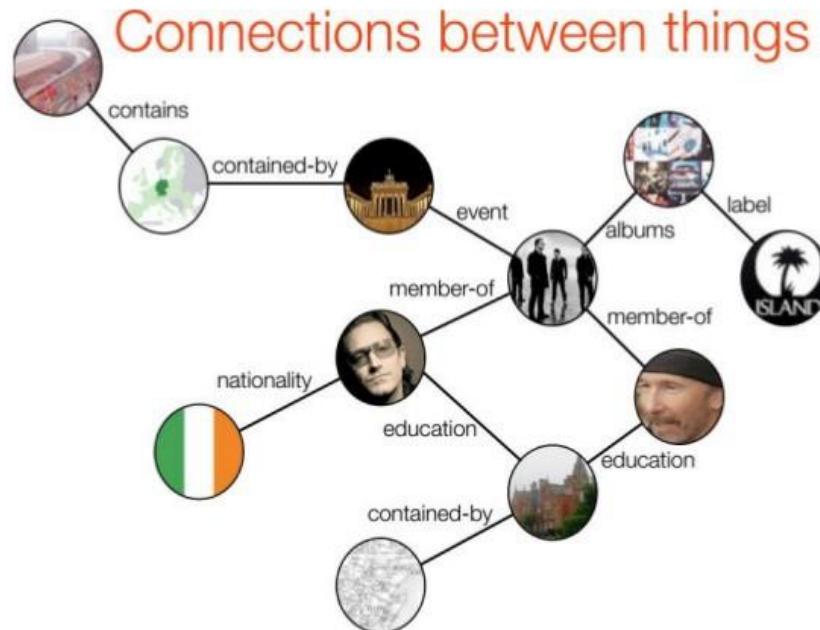
- “The world is a complex network of entities”

→ **Network model**



# Network model

- “The world is a complex network of entities”



# “How should I store my data on a computer?”

—A word (or two) on binary formats

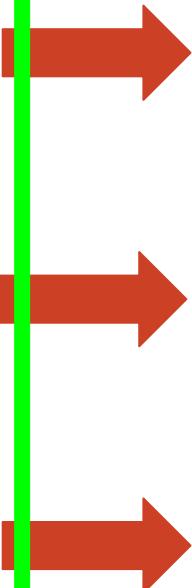
- Binary format often the key to performance, **avoiding expensive parsing**
- Modern binary formats support nested structures, various levels of schema enforcement, compression, etc.
- Python [pickle](#), Java [Serializable](#), [Protocol Buffers](#) (Google), [Avro](#) (supports schema evolution), [Parquet](#) (column-oriented), etc.

→ Consider converting to a binary format at the beginning of your processing pipeline (especially when using “big data”)

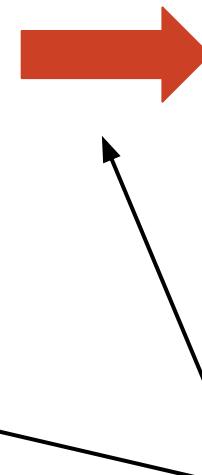
# Cooking with data



**Part 2:**  
**Data sources**



**Part 1:**  
**Data models**



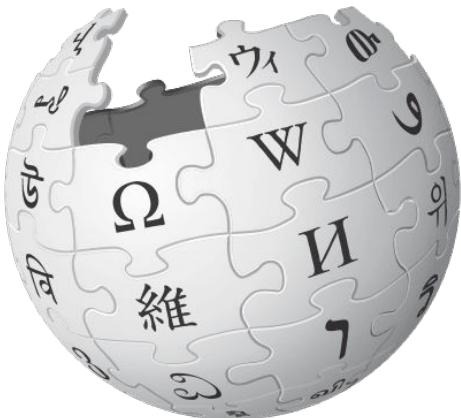
**Part 3:**  
**Data wrangling**

# Data sources at Web companies

## Examples from Facebook

- Application databases
  - Web server logs
  - Client-side event logs
  - API server logs
  - Ad server logs
  - Search server logs
  - Advertisement landing page content
  - Wikipedia
  - Images and video
- 
- The diagram illustrates the classification of data sources. A vertical red bracket on the right side of the list groups the items into three categories: 'Structured data (with clear schema)', 'Semi-structured data ("self-describing" structure; CSV etc.)', and 'Unstructured data'. The first four items (Application databases, Web server logs, Client-side event logs, API server logs) are grouped under 'Structured data'. The next three items (Ad server logs, Search server logs, Advertisement landing page content) are grouped under 'Semi-structured data'. The last two items (Wikipedia, Images and video) are grouped under 'Unstructured data'.
- Structured data (with clear schema)
- Semi-structured data (“self-describing” structure; CSV etc.)
- Unstructured data

# Another example: Wikipedia



- 300+ languages
- 42 million entities
- Mind-boggling richness of data



**WIKIPEDIA**  
The Free Encyclopedia

Article Talk

Read Edit View history

Search Wikipedia



Not logged in Talk Contributions Create account Log in

# San Francisco

From Wikipedia, the free encyclopedia  
(Redirected from San Francisco, California)

Coordinates: 37°47'N 122°25'W



This article is about the city and county in California. For other uses, see [San Francisco \(disambiguation\)](#).

**San Francisco** (initials SF<sup>[17]</sup>) (/sæn frən'skəʊ/, Spanish for Saint Francis; Spanish: [san fran'sisko]), officially the **City and County of San Francisco**, is the cultural, commercial, and financial center of Northern California. The consolidated city-county covers an area of about 47.9 square miles (124 km<sup>2</sup>)<sup>[18]</sup> at the north end of the San Francisco Peninsula in the San Francisco Bay Area. It is the fourth-most populous city in California, and the 13th-most populous in the United States, with a 2016 census-estimated population of 870,887.<sup>[19]</sup> The population is projected to reach 1 million by 2033.<sup>[19]</sup>

San Francisco was founded on June 29, 1776, when colonists from Spain established Presidio of San Francisco at the Golden Gate and Mission San Francisco de Asís a few miles away, all named for St. Francis of Assisi.<sup>[1]</sup> The California Gold Rush of 1849 brought rapid growth, making it the largest city on the West Coast at the time. San Francisco became a consolidated city-county in 1856.<sup>[20]</sup> After three-quarters of the city was destroyed by the 1906 earthquake and fire,<sup>[21]</sup> San Francisco was quickly rebuilt, hosting the Panama-Pacific International Exposition nine years later. In World War II, San Francisco was a major port of embarkation for service members shipping out to the Pacific Theater.<sup>[22]</sup> It then became the birthplace of the United Nations in 1945.<sup>[23][24][25]</sup> After the war, the confluence of returning servicemen, massive immigration, liberalizing attitudes, along with the rise of the "hippie" counterculture, the Sexual Revolution, the Peace Movement growing from opposition to United States involvement in the Vietnam War, and other factors led to the Summer of Love and the gay rights movement, cementing San Francisco as a center of liberal activism in the United States. Politically, the city votes strongly along liberal Democratic Party lines.

A popular tourist destination,<sup>[26]</sup> San Francisco is known for its cool summers, fog, steep rolling hills, eclectic mix of architecture, and landmarks, including the **Golden Gate Bridge**, **cable cars**, the former **Alcatraz Federal Penitentiary**, **Fisherman's Wharf**, and its **Chinatown** district. San Francisco is also the headquarters of five major banking institutions and various other companies such as Levi Strauss & Co., Gap Inc., Fitbit, Salesforce.com, Dropbox, Reddit, Square Inc., Delta, Airbnb, Weekly, Pacific Gas and

Electric Company, Yelp, Pinterest, Twitter, Uber, Lyft, Mozilla, Wikimedia Foundation, and many more. The city is home to number of educational and cultural institutions, such as the University of California, Berkeley, the California Academy of Sciences, the California Museum, the San Francisco Museum of Modern Art, and the California Academy of Sciences.

San Francisco has several nicknames, including "The City by the Bay", "Goat City", and as well as older ones like "The City that Knows How", "Baghdad City".<sup>[17]</sup> As of 2017, San Francisco is ranked high on world liveability rankings.

## Contents [hide]

- 1 History
- 2 Geography
  - 2.1 Cityscape
    - 2.1.1 Neighborhoods
  - 2.2 Climate
- 3 Demographics
  - 3.1 Race, ethnicity, religion and languages
  - 3.2 Education, households, and income
    - 3.2.1 Homelessness
- 4 Economy

## San Francisco, California

### Consolidated city-county

#### City and County of San Francisco



San Francisco and the Golden Gate Bridge from Marin Headlands



Flag



Seal

link

Learning resources from  
Wikiversity

### Places adjacent to San Francisco

[show]

### City and County of San Francisco

[show]

### Articles relating to the City and County of San Francisco

[show]

**Authority control** WorldCat Identities · VIAF: 143700861 · LCCN: n79018452 · ISNI: 0000 0004 0461 8991 · GND: 4051520-5 · SUDOC: 040776433 · NDL: 00628542

**Categories:** San Francisco | 1850 establishments in California | California counties | Cities in the San Francisco Bay Area | Consolidated city-counties in the United States | Counties in the San Francisco Bay Area | County seats in California | Hudson's Bay Company trading posts | Incorporated cities and towns in California | Populated coastal places in California | Populated places established in 1776 | Port cities and towns of the West Coast of the United States | Spanish mission settlements in North America

Location of San Francisco in California  
Coordinates: 37°47'N 122°25'W

Country	United States
State	California



32

# Wikipedia

## How to work with Wikipedia?



- XML dumps with wiki markup, SQL database dumps
- Issues: Unicode, size, recency, etc.
- To make your life easier:
  - (1) Find projects on GitHub to help you
  - (2) Use more structured versions (p.t.o.)

# Wikidata

- “Database version” of Wikipedia
- {fr:Suisse, de:Schweiz, it:Svizzera, en:Switzerland, ...} → Q39

[Main page](#)  
[Contents](#)  
[Featured content](#)  
[Current events](#)  
[Random article](#)  
[Donate to Wikipedia](#)  
[Wikipedia store](#)

---

[Interaction](#)  
[Help](#)  
[About Wikipedia](#)  
[Community portal](#)  
[Recent changes](#)  
[Contact page](#)

---

[Tools](#)  
[What links here](#)  
[Related changes](#)  
[Upload file](#)  
[Special pages](#)  
[Permanent link](#)  
[Page information](#)  
Wikidata item  
[Cite this page](#)

This  
"Swi

Switzer

city of E  
bordered  
a [landl](#)  
(15,940

is conce  
econom

The est  
against  
[Westph](#)

internat  
frequen  
Switzer  
a found  
[Econon](#)

Spanni  
German  
rooted i



# Switzerland (Q39)

federal republic in Western Europe

Swiss Confederation | CH | SUI | Suisse | Schweiz | Svizzera |

edit

▼ In more languages Configure

Language	Label	Description	Also known as
English	Switzerland	federal republic in Western Europe	Swiss Confederation CH SUI Suisse Schweiz Svizzera 
German	Schweiz	Staat in Mitteleuropa	Schweizerische Eidgenossenschaft Eidgenossenschaft CH SUI
Swiss German	Schwyz	No description defined	
French	Suisse	pays d'Europe	Confédération helvétique Confédération suisse CH SUI

All entered languages

## Statements

instance of	<span style="color: #ccc;">☰</span> sovereign state <span style="color: #ccc;">▶</span> 1 reference	edit
	<span style="color: #ccc;">☰</span> country start time	edit

12 September 1848 Gregorian

# Wikidata

- “Database version” of Wikipedia
- {fr:Suisse, de:Schweiz, it:Svizzera, en:Switzerland, ...} → Q39
- Both API access and full database dumps
- Available as
  - JSON (document model)
  - RDF (network model)

[Main page](#)  
[Contents](#)  
[Featured content](#)  
[Current events](#)  
[Random article](#)  
[Donate to Wikipedia](#)  
[Wikipedia store](#)

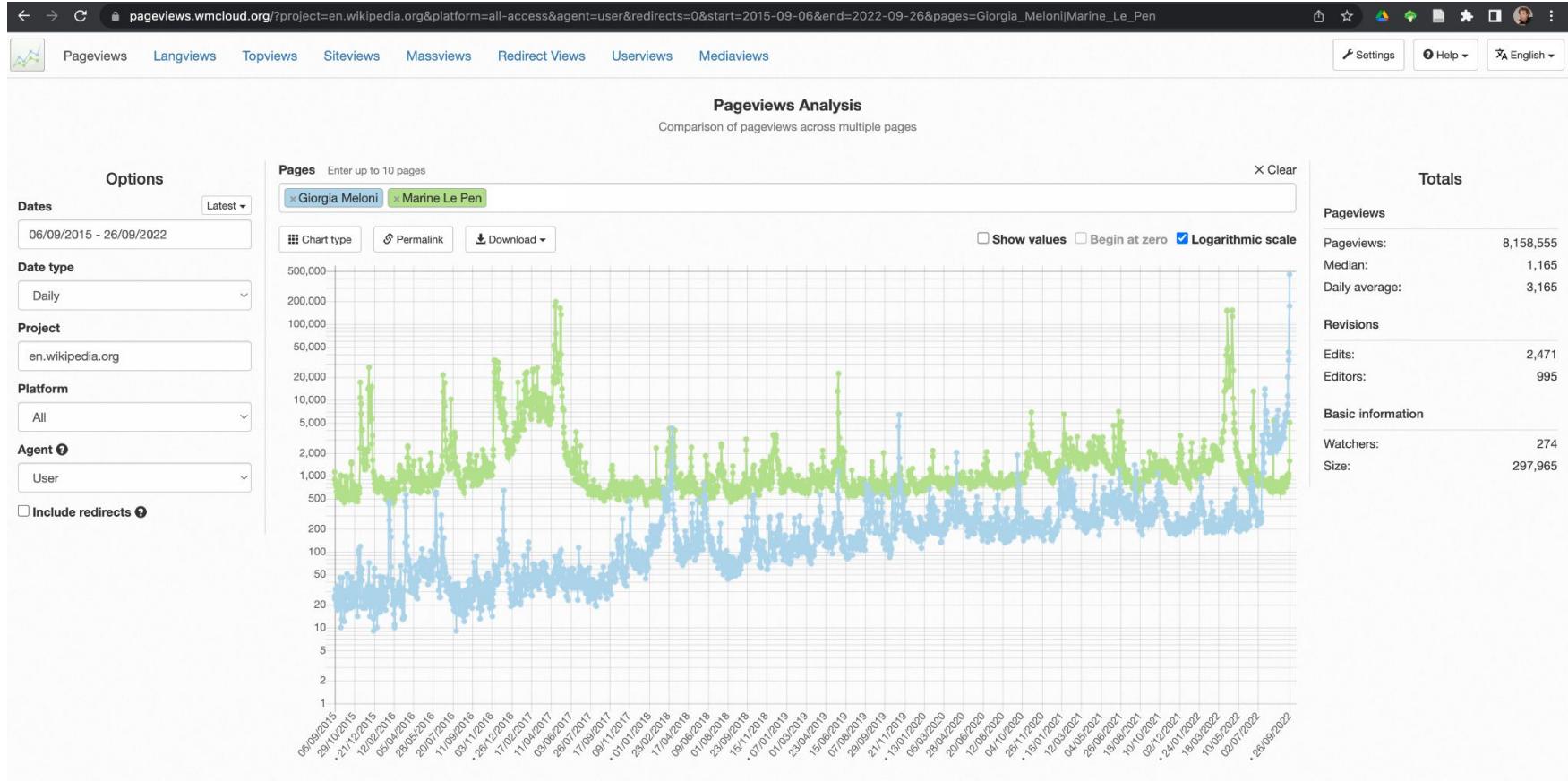
[Interaction](#)

[Help](#)  
[About Wikipedia](#)  
[Community portal](#)  
[Recent changes](#)  
[Contact page](#)

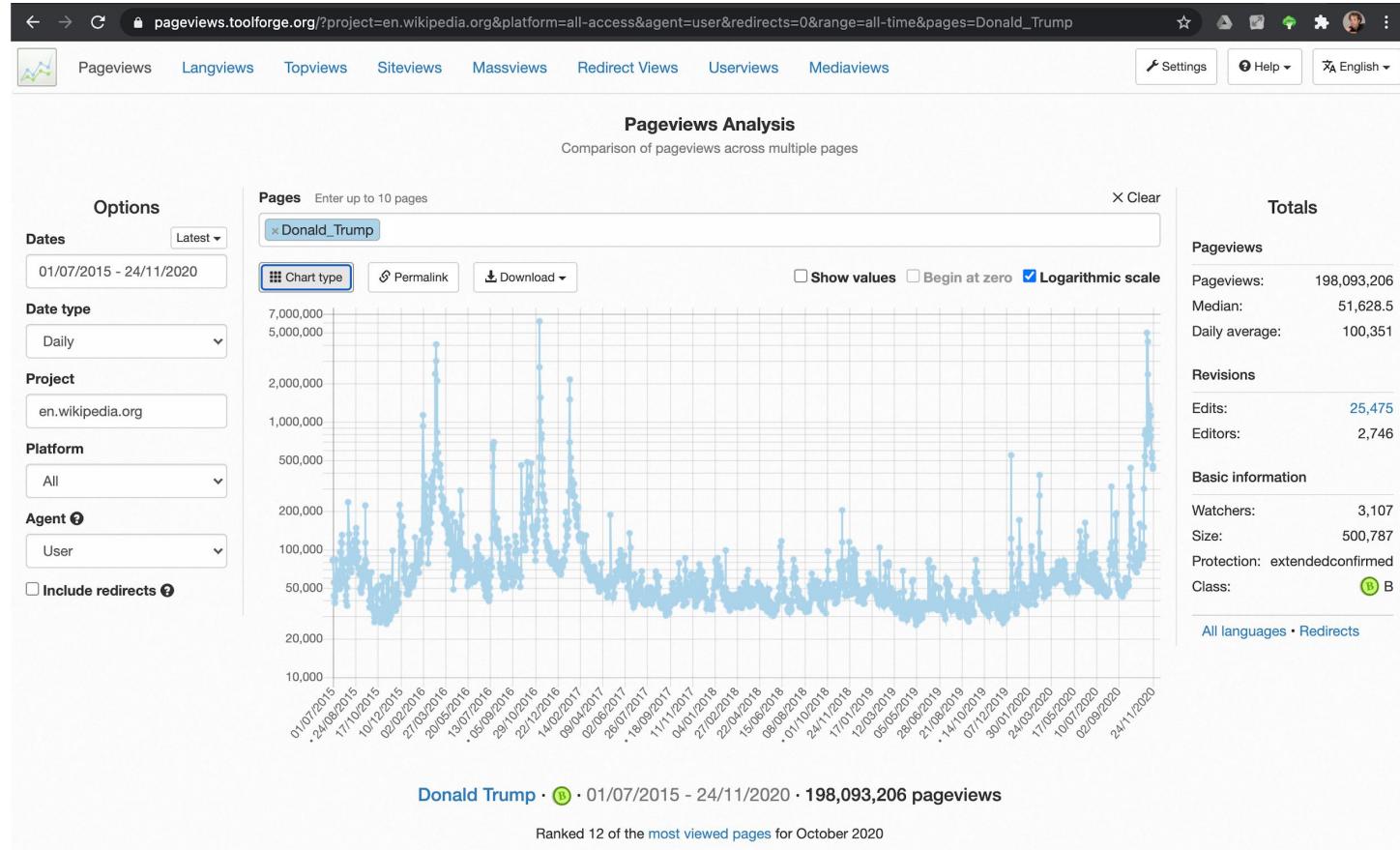
[Tools](#)

[What links here](#)  
[Related changes](#)  
[Upload file](#)  
[Special pages](#)  
[Permanent link](#)  
[Page information](#)  
[Wikidata item](#)  
[Cite this page](#)

# Wikipedia pageview logs



# Wikipedia pageview logs



# Crawling and processing webpages: HTML

Plenty of bulk-downloadable HTML data:

- Common Crawl dataset, about 1.82 billion web pages -- huge!
- (... but less than 0.1% of Google's Web crawl, as of 2015)
- 145 TB, hosted on Amazon S3, also available for download

... but if you need a specific website: use a  
**crawler/“spider”**: Apache Nutch, Storm, Heritrix 3,  
Scrapy, etc. (or simply wget...)

# Useful HTML tools

**Requests** <http://docs.python-requests.org/en/master/>

An elegant and simple HTTP library for Python

**Scrapy** <https://scrapy.org/>

An open-source framework to build Web crawlers

**Beautiful Soup** <http://www.crummy.com/software/BeautifulSoup/>

A Python API for handling real HTML

**Plain ol' /regular/express\*ion/s...**

# Schema.org: microformats for Web pages

- Nuggets of structured information embedded in (semantically) unstructured HTML

```
<div itemscope itemtype="http://schema.org/Movie">
  <h1 itemprop="name">Avatar</h1>
  <div itemprop="director" itemscope itemtype="http://schema.org/Person">
    Director: <span itemprop="name">James Cameron</span>
    (born <time itemprop="birthDate" datetime="1954-08-16">August 16, 1954</time>)
  </div>
  <span itemprop="genre">Science fiction</span>
  <a href="../movies/avatar-theatrical-trailer.html" itemprop="trailer">Trailer</a>
</div>
```



# Web services

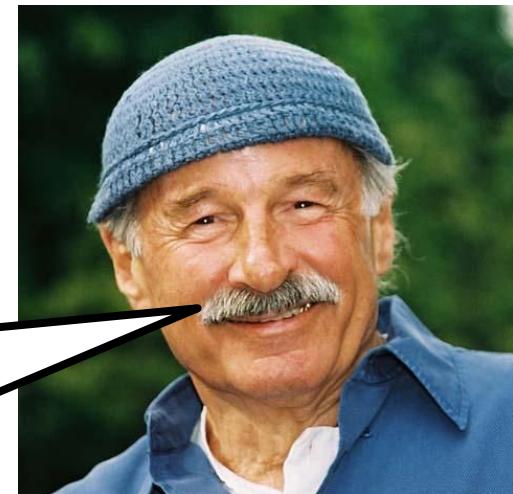
- Most large web sites today actively discourage “screen-scraping” to get their content
- Instead: Web service APIs, for interoperable machine-to-machine interaction over a network
- The preferred way to get data from online sources
- Most common framework: REST
  - You request a URL from the server via HTTP
  - The server responds with a text file (e.g., JSON, XML, plain text)

# REST example

- ```
{  
  "user": {  
    "name": "Jane",  
    "gender": "female",  
    "location": {  
      "href":  
        "http://www.example.org/us/  
        /ny/new_york",  
      "text": "New York"  
    }  
  }  
}
```
- ← This resource is a description of a user named Jane
- Requested by sending GET request for the resource's URL, e.g., via [curl](#):  
`curl http://www.example.org/users/jane/`
  - If users need to modify the resource, they GET it, modify it, and PUT it back
  - The href to the location resource allows savvy clients to get more information with another simple GET request
  - Implication: Clients cannot be too “thin”; need to understand resource formats!

# Joe Zawinul

I said, “What’s that?” and he said, “That’s **jazz**.” “How do you write that?” And he spelt it out. Somehow I saw my name in there, and I liked this word.

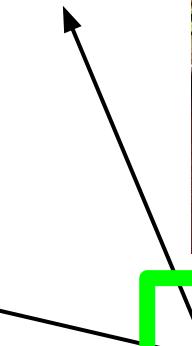


# Robert West

I said, “What’s that?” and he said, “That’s **REST**.” “How do you write that?” And he spelt it out. Somehow I saw my name in there, and I liked this word.



# Cooking with data



Part 2:  
Data sources

Part 1:  
Data models

Part 3:  
Data wrangling

# Working with raw data sucks

Data comes in all shapes and sizes

- CSV files, PDFs, SQL dumps, .jpg, ...

Different files have different formatting

- Empty string or space instead of NULL, extra header rows, character encoding, ...

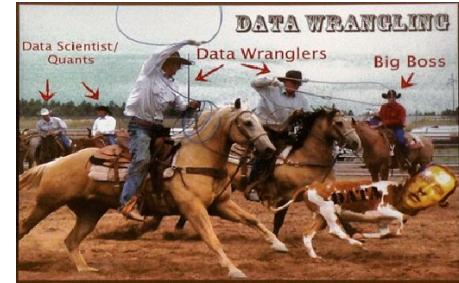
“Dirty” data

- Unwanted anomalies, duplicates

---

# **Raw data without thinking: A recipe for disaster!**

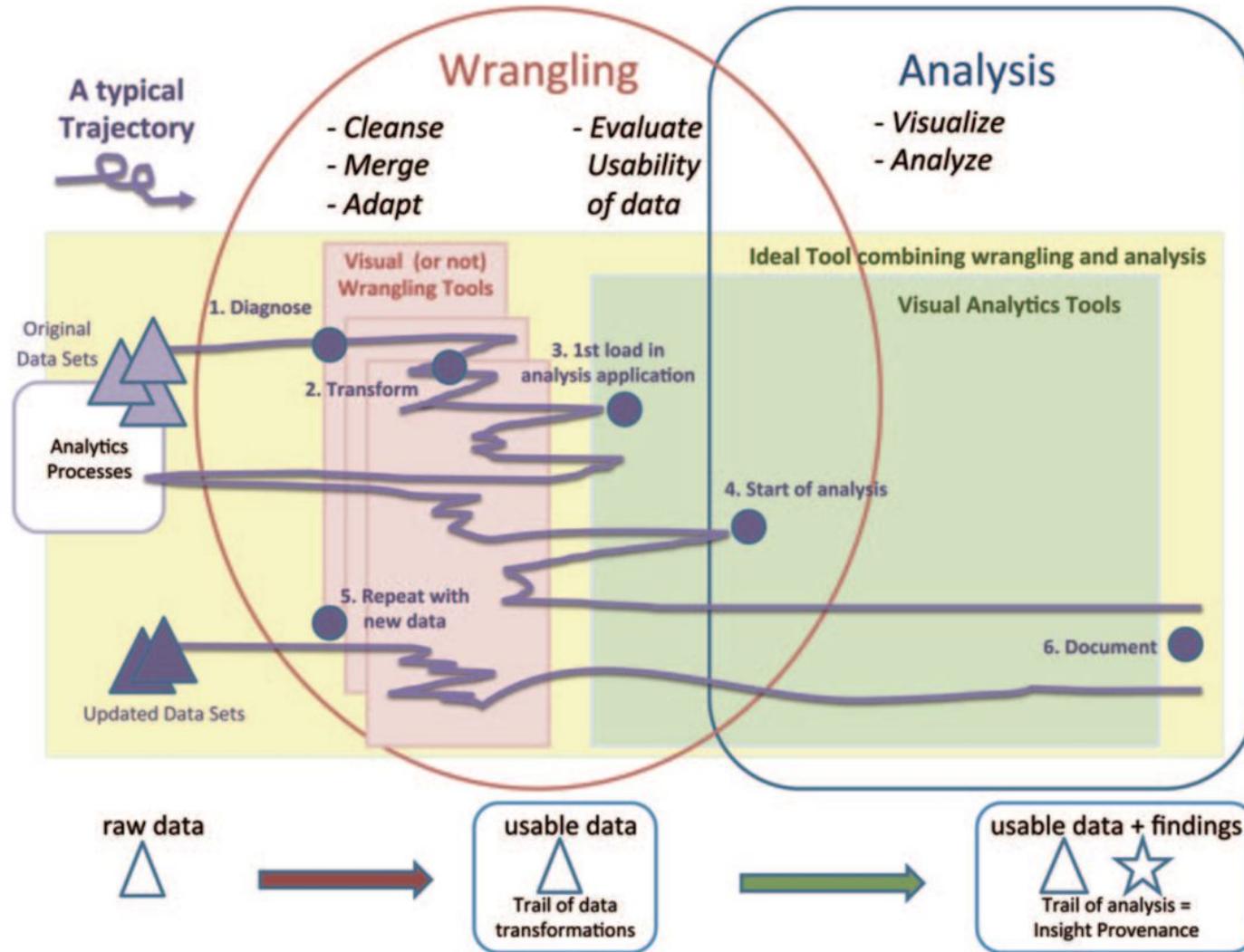
# What is data wrangling?



- **Goal:** extract and standardize the raw data
  - Combine multiple data sources
  - Clean data anomalies
- **Strategy:** Combine automation with interactive visualizations to aid in cleaning
- **Outcome:** Better efficiency and scale of data importing

Wrangling  
takes  
**between 50%**  
**and 80% of**  
**your time...**

[Source]



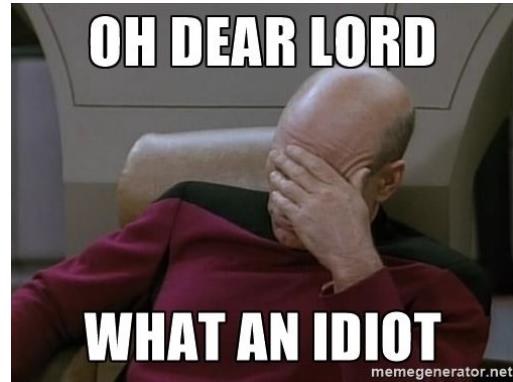
# Types of data problems

- Missing data
- Incorrect data
- Inconsistent representations of the same data
- About 75% of data problems require human intervention (e.g., experts, crowdsourcing, etc.)
- Tradeoff between cleaning data vs. over-sanitizing data



[link](#)

# “Dirty data” horror stories



“Dear Idiot” letter

17,000 men are pregnant

As the crow flies

CHF 10,000 compute-cluster bill

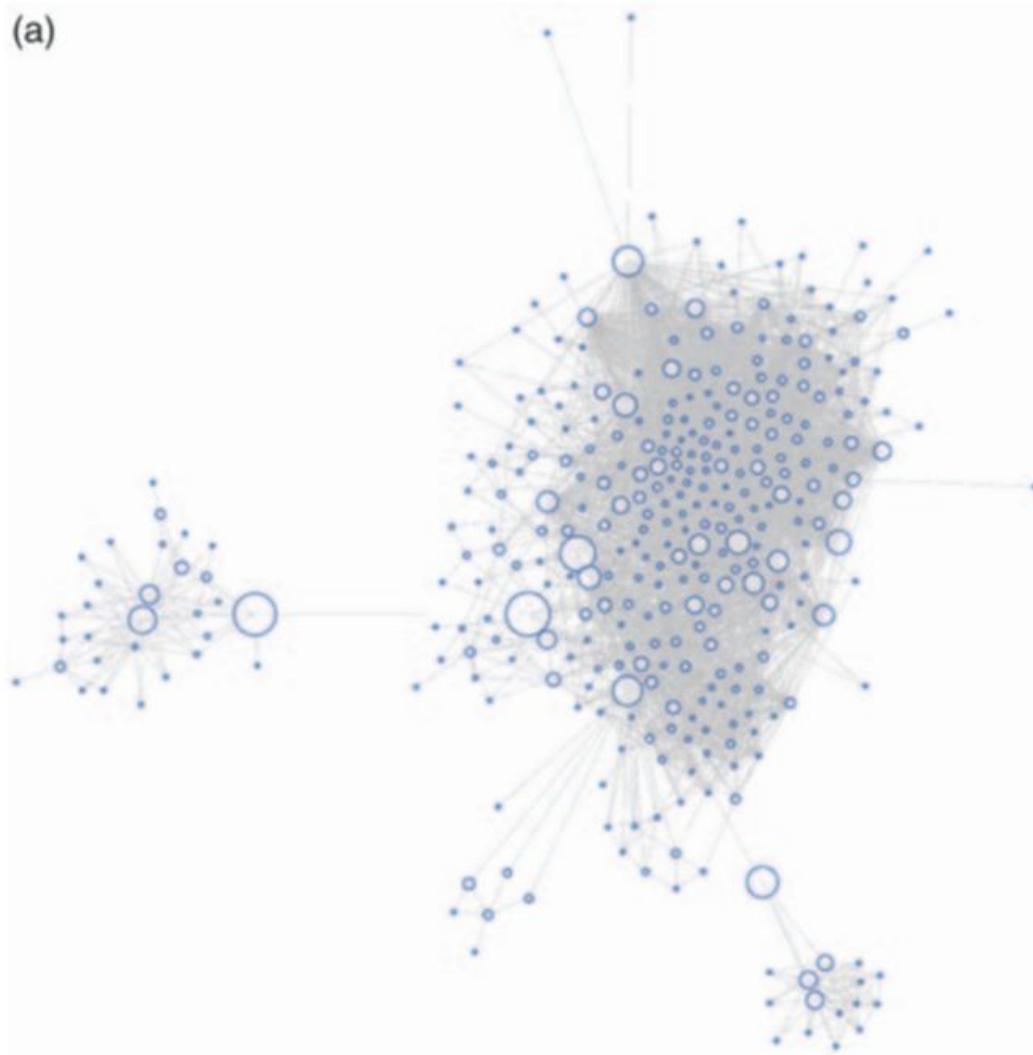
[\[Source\]](#)

# Diagnosing data problems

- Visualizations and basic stats can convey issues in “raw” data
- Different representations highlight different types of issues:
  - Outliers often stand out in the right kind of plot
  - Missing data will cause gaps or zero values in the right kind of plot
- Becomes increasingly difficult as data gets larger
  - Sampling to the rescue!

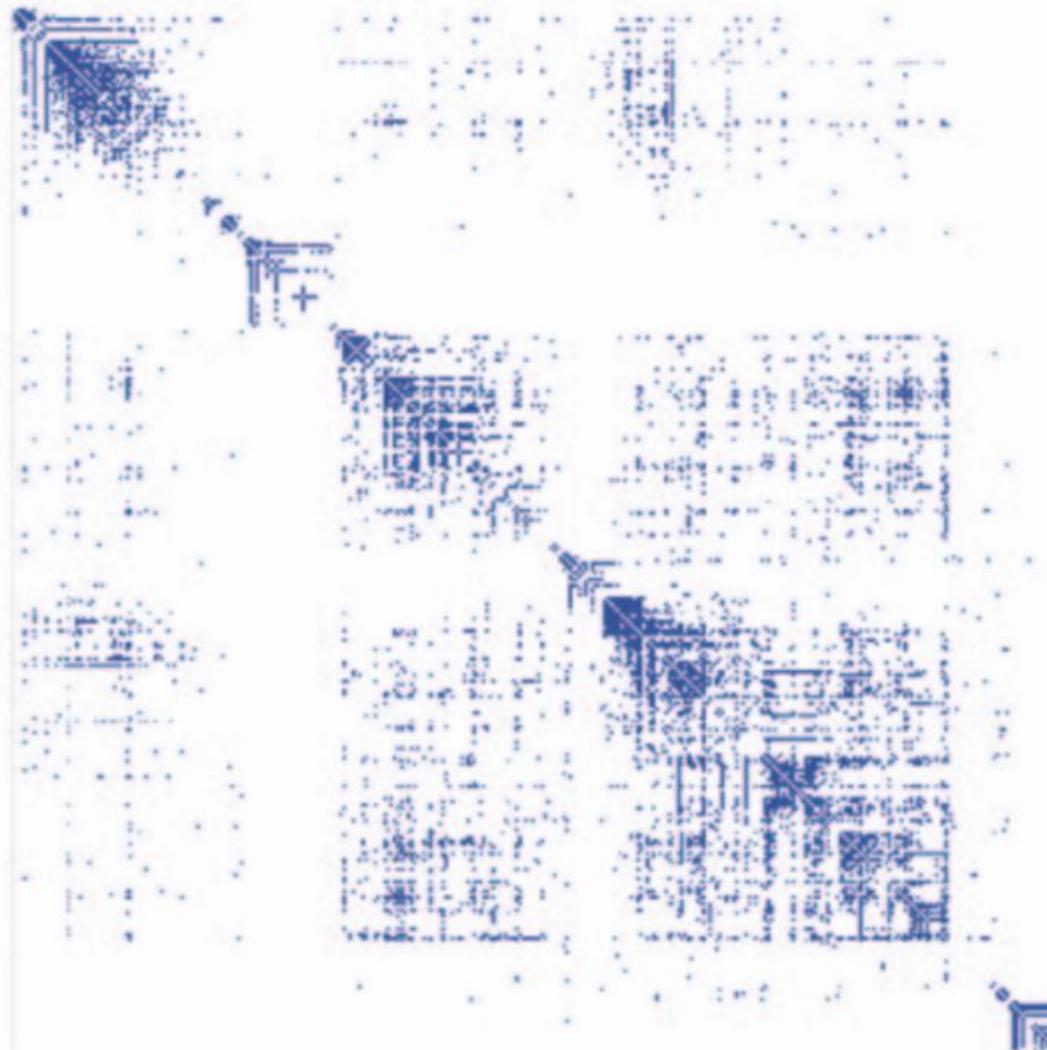
(a)

# Facebook graph



## Matrix view (1)

Automatic permutation of rows and columns to highlight patterns of connectivity



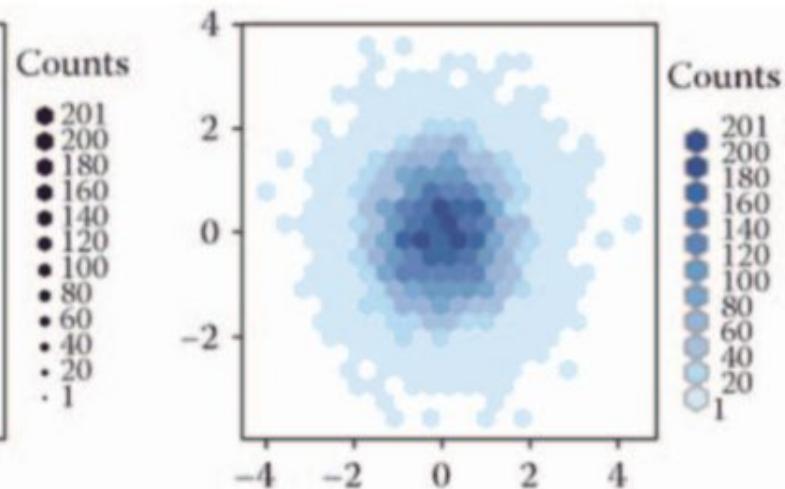
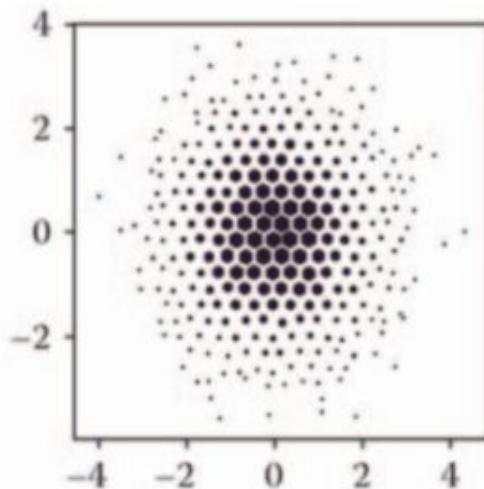
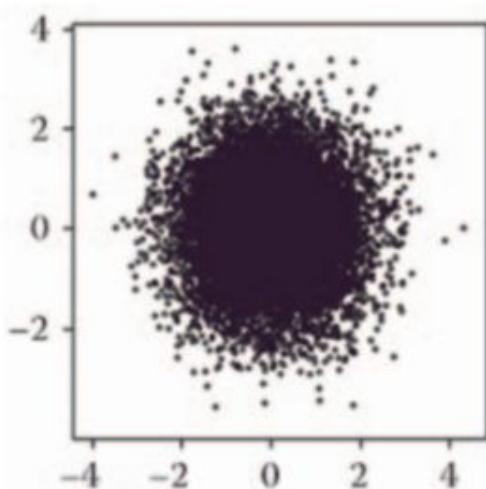
## Matrix view (2)

Rows and columns sorted in the order in which data was retrieved via the Facebook API

**Can you guess what's going on here?**

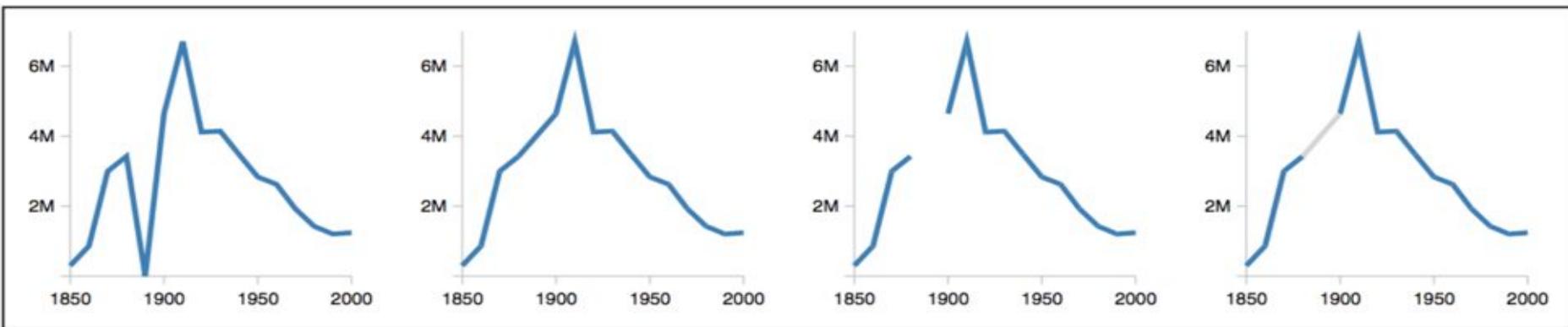
|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |     |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|

# Viz at scale? Careful!



# Dealing with missing data

U.S. census counts of people working as “farm laborers”; values from 1890 are **missing due to records being burned in a fire**



- Set values to zero?
- Interpolate based on existing data?
- Omit missing data?

Knowledge about domain  
and data collection should  
drive your choice!

# Inconsistent data: “My name is Willy”

The screenshot shows a LinkedIn profile page. At the top, there's a navigation bar with icons for Home, People (1), Inbox (1), Notifications (10), and a profile picture. Below the bar, a banner for a workshop in Geneva is visible. The main profile area features a circular photo of a smiling man with glasses. Below the photo, the name "Willy W." is followed by a "3rd" connection indicator, the title "Data Scientist", and the company "Sportsbet.com.au • University of Melbourne". The location is listed as "Melbourne, Australia • 273 connections". A blue "Connect" button is at the bottom.

| First name | Last name |
|------------|-----------|
| Willy      | NULL      |
| ...        | ...       |

## Experiments on Pattern-based Reli

Willy Yap and Timothy Baldwin  
NICTA Victoria Research Laboratory  
Department of Computer Science and Software Engineering  
University of Melbourne  
willy@csse.unimelb.edu.au, tim@csse.unimelb.edu.au

# Before you start analyzing your data

- “Do I have **missing data**?” “If data were missing, how could I know?”
- “Do I have **corrupted data**?” (May arise from measurement errors, processing bugs, etc.)
- **Parse/transform data** into appropriate format for your specific analysis (see “Part 1: Data models”)
- Don’t be surprised if you need to come back to this stage!

# Desiderata

It's always ideal if you can put your hands on the **code/documentation about the dataset** you are analyzing (provenance)

It's always ideal if the provided **data format is nicely parseable** (otherwise you need regexes, or maybe even pay humans)

# Highly non-parseable data

"All the News That's Fit to Print."

VOL. LXXXVIII...No. 29,768. Entered as Second-Class Matter. Postoffice, New York, N.Y.

NEW YORK, WEDNESDAY, JULY 26, 1939. P THREE

BARKLEY DEMANDS LENDING BILL VOTE BEFORE QUITTING

Senate Is Told It Cannot Go Home Until Action Is Taken 'One Way or the Other'

FOR ONE MORE JOB EFFORT

He and Rayburn of House Talk With Roosevelt and Then He Delivers Ultimatum

By CHARLES W. HURD Special to The New York Times.

WASHINGTON, July 25.—The Administration's \$2,490,000,000 Works Financing Bill went before the Senate late today accompanied by an ultimatum from Senator Barkley, the usually mild-mannered man who would not be permitted to adjourn until this measure had been disposed of "one way or the other."

Senator Barkley asked that a chance be given to the program on the ground that previous to the bill's introduction he had been unable to get the Senate to agree to a bill to give the nation employment problem.

He recited the previous efforts, the emergency works created by the WPA, the PWA and the CCC; he listed the long-term programs involved in the Social Security Act, the Wages and Hours Law and the *lending* measures thus far created

by Charles W. Hurd

a boat and carried the blue-eyed boy back into his arms. Mrs. McMornan added quickly:

"Dens Fendler. I was lost on the mountain," he replied weakly. Given some coffee, he appeared somewhat refreshed and insisted on telephoning his parents, Mr. and Mrs. Donald Fendler, to assure them of his safety. They were reached at a Bangor hospital. "I'm all right, mama," he told

Continued on Page Three

Fendler Boy Found Alive in Woods Eight Days After Becoming Lost

HEAT OF 90° HERE ADDS TO HUGE LOSS

BUDGET IS REVISED SO KINDERGARTENS

JAPAN BLOCKS RIVER CLOSING RIV

Copyright, 1939, by The New York Times Company.

centuries suffis

Type the two words:

reCAPTCHA™ stop spam. read books.

badly by mosquitoes and flies. He had subsisted on berries and drank stagnant water from pools in the rocks until he reached fresh water, the boy told McMornan. At one time he heard an airplane but he could not remember which day it was.

Nor could he say definitely when his aimless wanderings through the woods ended.

A Freakish Storm in Boston A freakish thunderstorm accom-

City in an effort to escape the oppressive heat and humidity. Week-day attendance records for the season were shattered at several resorts. One drowning and many rescues were reported.

Hundreds of fires burned in the dry forests and brushlands of Pennsylvania, New Jersey and New York.

bers have conferred with resources, parents and school administrators in the hope of finding means to meet an apparent \$8,300,000 deficit. Warnings had been issued from time to time that unless more funds were provided the school system would be "wrecked" by the end of the year.

Economy suggestions came from Mayor La Guardia and other city officials. In an attempt to save

In Washington, Secretary Hull said that the United States would hold Japan responsible for any injury to Americans or damage to their property resulting from the closing of the river.

Chairman Pittman of the Senate Foreign Relations Committee pledged his support to the Vandenberg resolution for abrogation.

4<sup>th</sup> [Page 1.]

The German extremely docile tests in the Bal

The Soviet said Russia has warships in th that the total sian submarine Germany's together." Japan

Entire NY  
Times archive  
(since 1851)  
digitized as of  
2015

# Example from Bob's research (AD 2013)

Q: What do

Our method:  
Consenting IE users, 18 mo

URL  
yahoo.com?q=the+onion  
theonion.com  
theonion.com/Area-Man-Sad  
bing.com  
bing.com?q=feijoada+recipe  
allrecipes.com/tasty-feijoada  
food.com/best-feijoada-recipe

Referrer  
referrer  
yahoo.com  
yahoo.com?q=the+onion

- Find Amazon add-to-cart events heuristically in logs:  
Referrer: <http://www.amazon.com/Forks-Over-Knives-Plant-Based-Health/dp/1615190457>  
URL: <http://www.amazon.com/gp/cart/view-upsell.html?...>
- Get product info for product id using Amazon API
- Consider all add-to-cart events for category "Diets & Weight Loss"



- Collaborated with clinician at Washington Hospital Center, Washington, D.C.
- Data: All CHF admissions to emergency department for time period of our browsing logs

Original recipe makes 8 servings [Change](#)

Ingredients [Edit and Save](#)

Original recipe makes 8 servings [Change](#)

1 (12 ounce) package dry black beans, soaked overnight

1 1/2 cups chopped onion, divided

1/2 cup green onions, chopped

1 clove garlic, chopped

2 smoked ham hocks

8 ounces diced ham

1/2 pound thickly sliced bacon, diced

1/2 cup chopped (optional)

1/4 cup chopped (optional)

Sodium 299 mg 12%  
\* Percent Daily Values are based on a 2,000 calorie diet.

See More

powered by Esha Research, Inc.

1233377501 11336381 Diamonds, SC, USA

[Paper with results and plots](#)

# Feedback

Give us feedback on this lecture here:

<https://go.epfl.ch/ada2022-lec2-feedback>

Feedback form available for each lecture and lab session

- What did you (not) like about this lecture?
- What was (not) well explained?
- On what would you like more (fewer) details?
- Is it nicer to follow the lecture online or offline?
- ...