

---

# CMPT 733

# Data Preparation

---

Instructor

Steven Bergner

Course website

<https://coursys.sfu.ca/2024sp-cmpt-733-g1/pages/>

Source

based on slides by Jiannan Wang

# Outline

---

1. Data Preparation Overview

2. Data Preparation Tasks

# Data Preparation Is **Still** the Bottleneck!!!

## 2014

The New York Times

### *For Big-Data Scientists, 'Janitor Work' Is Key Hurdle to Insights*

Yet far too much handcrafted work — what data scientists call “data wrangling,” “data munging” and “data janitor work” — is still required. Data scientists, according to interviews and expert estimates, spend from 50 percent to 80 percent of their time mired in this more mundane labor of collecting and preparing unruly digital data, before it can be explored for useful nuggets.

<https://www.nytimes.com/2014/08/18/technology/for-big-data-scientists-hurdle-to-insights-is-janitor-work.html>

## 2020

 ANACONDA

### The State of Data Science 2020 Moving from hype toward maturity

We were disappointed, if not surprised, to see that data wrangling still takes the lion's share of time in a typical data professional's day. Our respondents reported that almost half of their time is spent on the combined tasks of data loading and cleansing. Data

<https://www.anaconda.com/state-of-data-science-2020>

# Trend: Data Prep about 38% of effort

2022

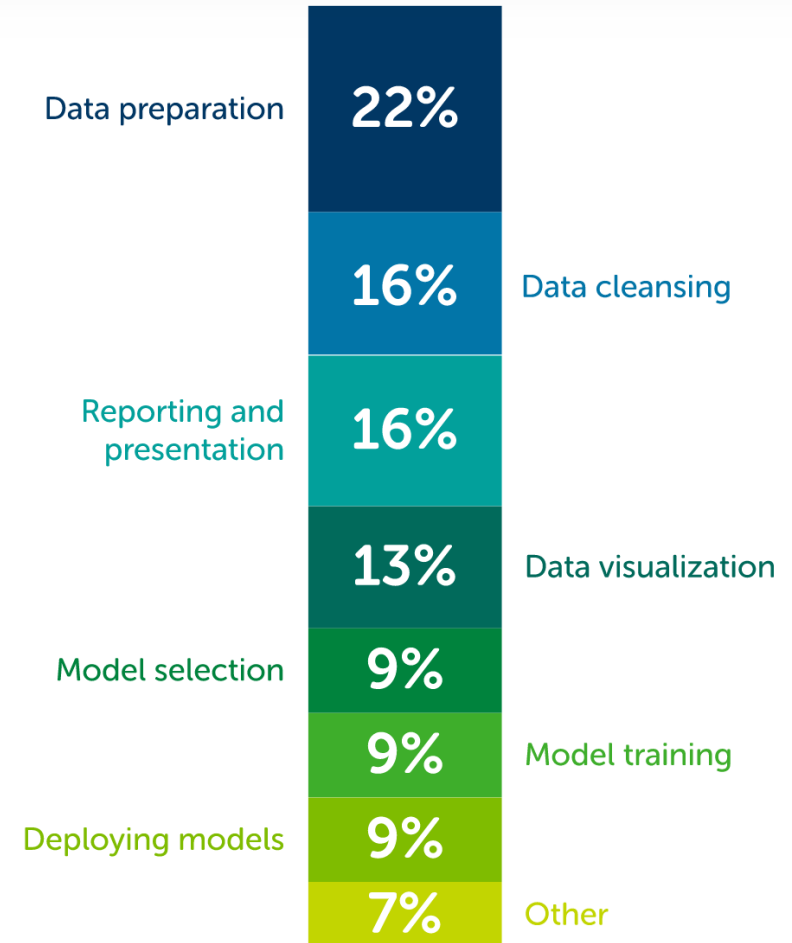


## DATA PROFESSIONALS AT WORK

### How do data scientists spend their time?

Data professionals spend their time on a variety of tasks that require diverse technical and non-technical skills. Respondents indicated they spend about 37.75% of their time on data preparation and cleansing. Beyond preparing and cleaning data, interpreting results remains critical. **Data visualization** (12.99%) and demonstrating data's value through reporting and presentation (16.20%) are essential steps toward making data actionable and providing answers to critical questions. Working with models through selection, training, and deployment takes about 26.44% of respondents' time (-8.56% YoY).

<https://www.anaconda.com/state-of-data-science-2022>



n = 1,966

# Why Is Data Preparation Hard?



Collection



Cleaning



Integration



Analysis

How much time is spent on preparation?

1. **Too many small problems** (e.g., standardize date, dedup address, etc)
2. Humans have **different levels of expertise** (in data science and programming)
3. **Domain specific** (finance, social science, healthcare, economics, etc.)

# Human-in-the-loop Data Preparation

---

## Three Directions

- Spreadsheet GUI
- Workflow GUI
- Notebook GUI

# Spreadsheet GUI

CUSTOMER ANALYSIS >

customer

Random

🔍

⌵

✎

Run Job

🔍

⌵

✎

Run Job

🔍

⌵

✎

Run Job

Preview

| #               | IMSI        | CONTRACT_END        | CONTRACT_START      | #      | SUBSCRIBER_AGE | ABC          | STATUS |
|-----------------|-------------|---------------------|---------------------|--------|----------------|--------------|--------|
| 310T - 310.26T  |             | Jan 2013 - Dec 2016 | Jan 2000 - Dec 2014 | 0 - 15 |                | 2 Categories |        |
| 310170226812721 | 6/4/16      | 7/29/09             |                     |        | ACTIVE         |              |        |
| 310160900766700 | 3/28/15     | 10/6/13             |                     | 1      | ACTIVE         |              |        |
| 310170546822541 | 9/23/16     | 1/9/07              |                     | 7      | ACTIVE         |              |        |
| 310005432849230 | 5/29/15     | 2/14/01             |                     | 13     | ACTIVE         |              |        |
| 310026939721905 | 9/11/15     | 9/18/10             |                     | 4      | ACTIVE         |              |        |
| 310026015466952 | 8/27/15     | 3/13/06             |                     | 8      | ACTIVE         |              |        |
| 310170484724861 | 1/16/16     | 5/11/04             |                     |        | ACTIVE         |              |        |
| 310170765640471 | 05-Jul-2011 | 9/11/06             |                     | 4      | INACTIVE       |              |        |
| 310260310245556 | 12/24/15    | 3/28/01             |                     | 13     | ACTIVE         |              |        |
| 310150834295817 | 3/6/15      | 7/26/00             |                     | 14     | ACTIVE         |              |        |
| 310160464252516 | 9/25/15     | 4/4/04              |                     | 10     | ACTIVE         |              |        |
| 310120438750772 | 4/30/16     | 9/8/04              |                     | 10     | ACTIVE         |              |        |
| 310260195729676 | 1/16/15     | 1/3/04              |                     | 11     | ACTIVE         |              |        |
| 310026261822800 | 8/13/13     | 11/23/08            |                     | 4      | INACTIVE       |              |        |
| 310005667082048 | 8/4/16      | 10/22/14            |                     |        | ACTIVE         |              |        |
| 310170836020164 | 1/22/15     | 10/19/14            |                     | 0      | ACTIVE         |              |        |
| 310160772267782 | 11/21/15    | 12/28/14            |                     |        | ACTIVE         |              |        |
| 310170116249240 | 27-Sep-2011 | 2/9/09              |                     |        | INACTIVE       |              |        |
| 310026110612337 | 5/29/15     | 3/29/05             |                     | 9      | ACTIVE         |              |        |
| 310260681676970 | 11/17/16    | 5/21/07             |                     | 7      | ACTIVE         |              |        |
| 310004436630316 | 9/15/16     | 7/24/11             |                     |        | ACTIVE         |              |        |
| 310120423699542 | 2/27/15     | 6/29/11             |                     | 3      | ACTIVE         |              |        |
| 310120773194729 | 4/28/16     | 6/15/04             |                     | 10     | ACTIVE         |              |        |
| 310030295859214 | 2/7/15      | 3/24/12             |                     | 2      | ACTIVE         |              |        |
| 310012150088547 | 13-Jan-2009 | 12/10/05            |                     | 3      | INACTIVE       |              |        |
| 310120387060694 | 10/1/16     | 10/25/11            |                     | 3      | ACTIVE         |              |        |

Hide Example Values

m / dd / yy

9/18/15

6/13/15

5/21/15

12/12/15

1/16/16

m / d / yy

6/5/15

4/4/15

12/8/16

7/2/14

11/6/15

dd - month-abbrev - yyyy

14-Nov-2012

11-Jul-2007

00-Jul-2010

12.65k

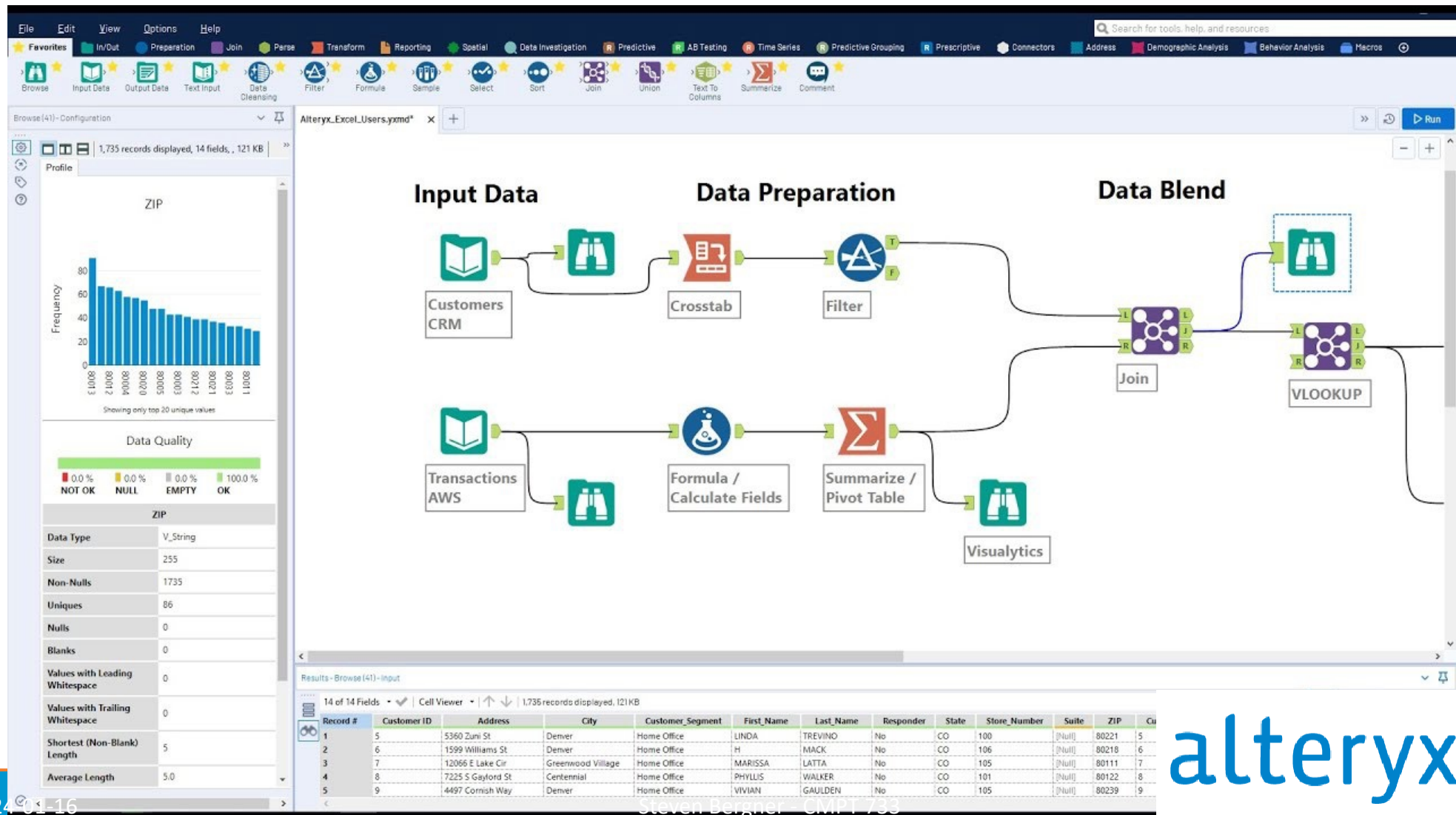
5.37k

Trifacta

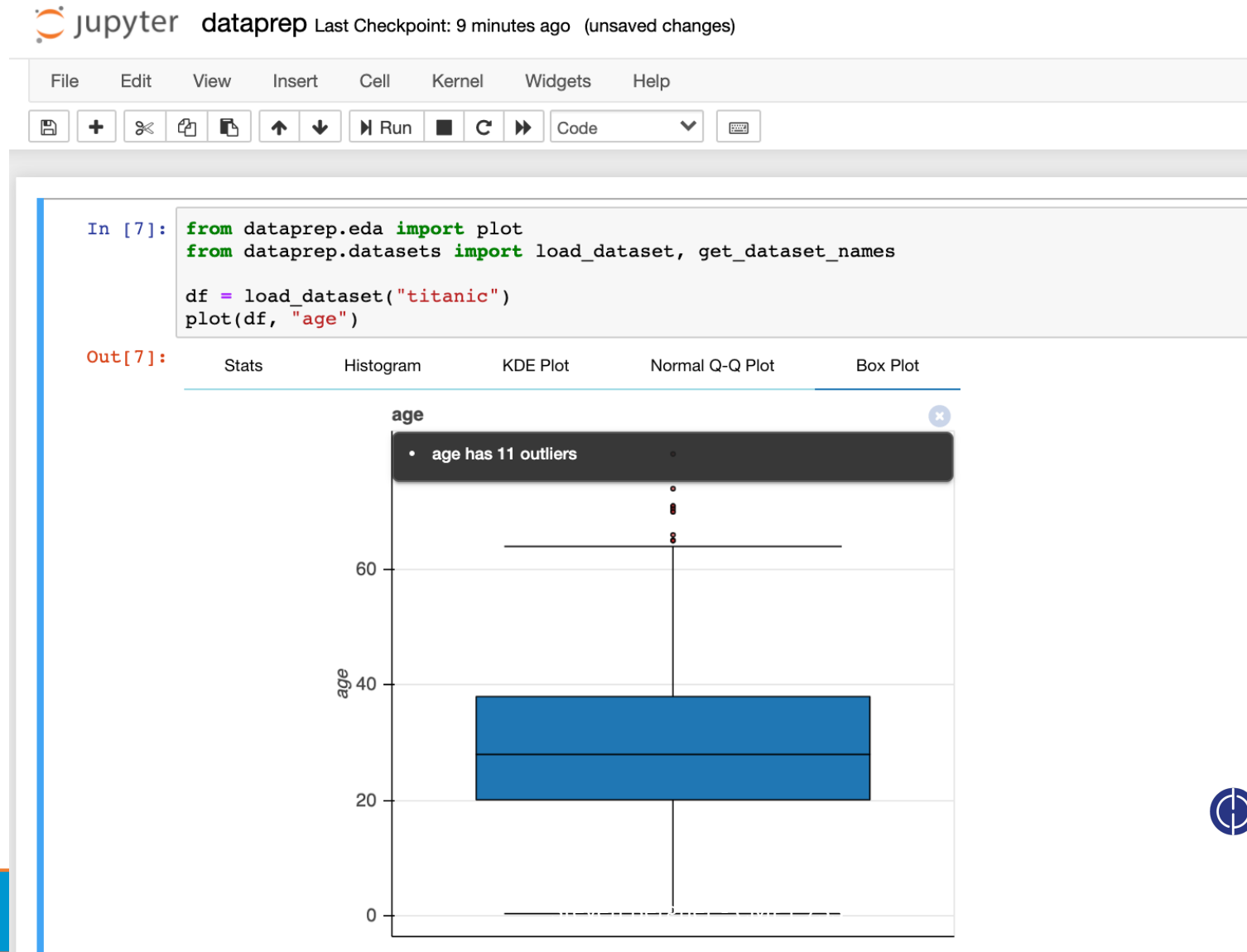
19 Columns 20,000 Rows 8 Data Types

Show only affected Rows

# Workflow GUI



# Notebook GUI



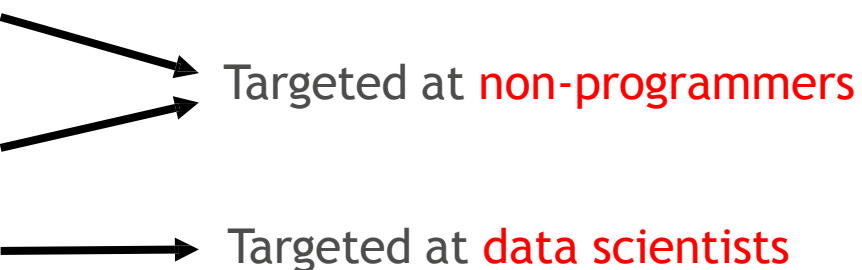
# Which Direction To Go?

---

“Data Prep Market was valued at USD 3.29 Billion in 2019 and is projected to reach **USD 18.11 Billion by 2027**, growing at a **CAGR of 25.64% from 2020 to 2027**”

Source: <https://www.verifiedmarketresearch.com/product/data-prep-market/>

## Three Directions

- Spreadsheet GUI
  - Workflow GUI
  - Notebook GUI
- 
- Targeted at **non-programmers**
- Targeted at **data scientists**

# Data Preparation Tasks

---

Data Collection

Data Cleaning

Data Integration

# Data Preparation Tasks

---

## Data Collection

- Where to collect
- How to collect

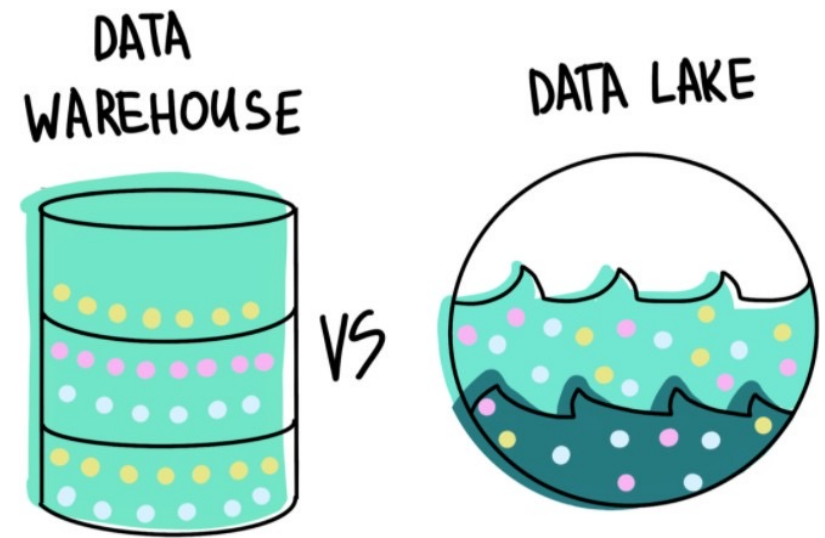
## Data Cleaning

## Data Integration

# Where to Collect?

## Internal Data

- Data Warehouse (Tabular Data)
- System Logs (Text Files)
- Documents (Word, Excel, PDF)
- Multimedia Data (Video, Audio, Image)

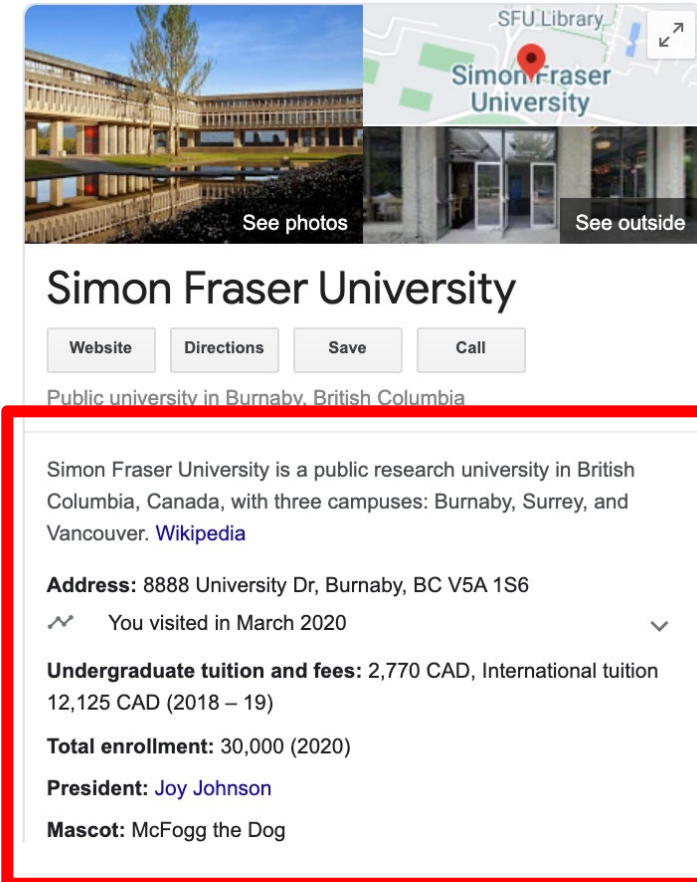


[Aside: [AWS Data Lake on S3](#)]

# Where to Collect?

## External Data

- Web Pages
- Web APIs (<https://github.com/public-apis/public-apis>)
- Open Data ([data.vancouver.ca](http://data.vancouver.ca), [www.data.gov](http://www.data.gov))
- Knowledge Graph (Wikidata, Freebase) 



**Simon Fraser University**

Public university in Burnaby, British Columbia

Simon Fraser University is a public research university in British Columbia, Canada, with three campuses: Burnaby, Surrey, and Vancouver. [Wikipedia](#)

**Address:** 8888 University Dr, Burnaby, BC V5A 1S6

 You visited in March 2020 

**Undergraduate tuition and fees:** 2,770 CAD, International tuition 12,125 CAD (2018 – 19)

**Total enrollment:** 30,000 (2020)

**President:** [Joy Johnson](#)

**Mascot:** McFogg the Dog

# Data Classification

## Data

### Structured

| Team                                                                                               | W  | L  | Pct  | GB  | Conf | Div | Home | Away | L10 | Strk |
|----------------------------------------------------------------------------------------------------|----|----|------|-----|------|-----|------|------|-----|------|
| 1  Rockets        | 20 | 4  | .833 | -   | 11-2 | 3-2 | 8-3  | 12-1 | 9-1 | W9   |
| 2  Warriors      | 21 | 6  | .778 | 0.5 | 9-4  | 2-1 | 8-3  | 13-3 | 8-2 | W6   |
| 3  Spurs        | 19 | 8  | .704 | 2.5 | 9-4  | 4-1 | 13-2 | 6-6  | 8-2 | W4   |
| 4  Timberwolves | 16 | 11 | .593 | 5.5 | 13-5 | 4-1 | 9-4  | 7-7  | 6-4 | W2   |

### Semi-structured

|                                                                                                                                    |                   |
|------------------------------------------------------------------------------------------------------------------------------------|-------------------|
| CLE - James Layup Shot: Missed                  | 06:48<br>CLE      |
| CLE - James Rebound (Off:1 Def:0)              | 06:46<br>CLE      |
| CLE - James Reverse Layup Shot: Made (2 PTS)  | 06:45<br>CLE 9-15 |
| Stoppage: Out-of-Bounds                                                                                                            | 06:29             |

### Unstructured

#### Is LeBron breaking the aging curve?



**Kevin Pelton**  
ESPN Staff Writer

5:10 AM PT

During his 15th NBA season, [Cleveland Cavaliers](#) star [LeBron James](#) is performing at a level that echoes the prime that saw him win four MVPs.

As James nears his 33rd birthday later this month, his performance at that age stands up to any of his predecessors, including [Michael Jordan](#)'s 1995-96 season that produced an MVP, a then-record 72 wins and a championship. (Because James entered the NBA directly out of high school, NBA experience isn't the best way to compare how he's aging to his peers. After all, Jordan's 15th year was actually his final one in the NBA at age 40.)

# Challenges

- Data Discovery

- How to find related data?

- Domain knowledge
- Information retrieval skills

- Data Privacy

- How to protect user privacy?

- Data masking
- Differential privacy

- Security

- How to avoid a data breach?

- Follow data access rules
- Encrypt highly confidential data

# Getting Data

---

From CSV Files

From JSON Files

From the Web

From HDFS

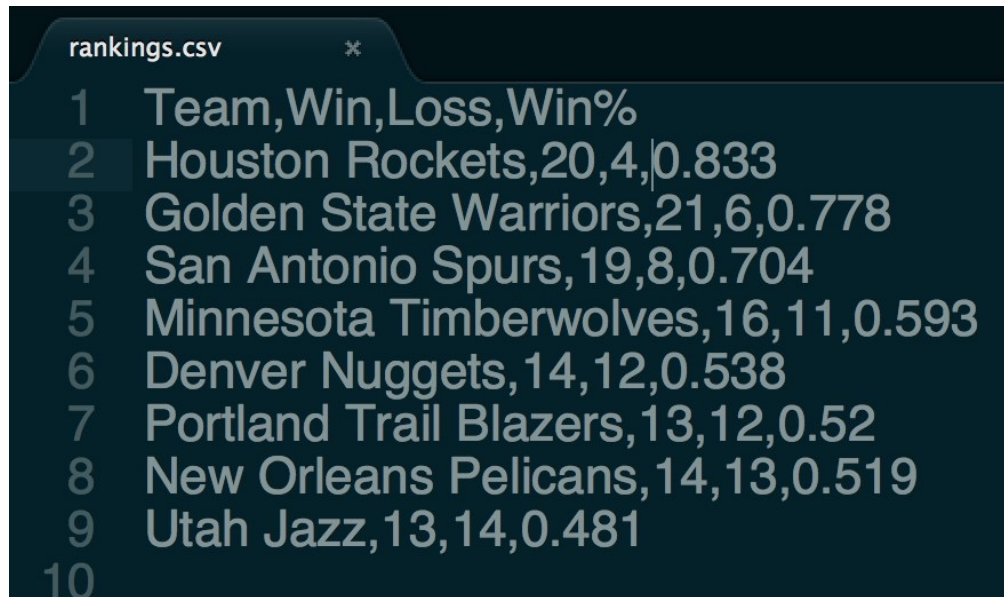
From Databases

From S3

From Web APIs

# Load Data From CSV Files

CSV is a file format for storing tabular data



A screenshot of a code editor window titled 'rankings.csv'. The editor displays 10 lines of CSV data. The first line is a header: 'Team,Win,Loss,Win%'. The subsequent lines list basketball teams with their win-loss records and win percentages. The data is as follows:

| Rank | Team                   | Win | Loss | Win%  |
|------|------------------------|-----|------|-------|
| 1    | Team                   | Win | Loss | Win%  |
| 2    | Houston Rockets        | 20  | 4    | 0.833 |
| 3    | Golden State Warriors  | 21  | 6    | 0.778 |
| 4    | San Antonio Spurs      | 19  | 8    | 0.704 |
| 5    | Minnesota Timberwolves | 16  | 11   | 0.593 |
| 6    | Denver Nuggets         | 14  | 12   | 0.538 |
| 7    | Portland Trail Blazers | 13  | 12   | 0.52  |
| 8    | New Orleans Pelicans   | 14  | 13   | 0.519 |
| 9    | Utah Jazz              | 13  | 14   | 0.481 |
| 10   |                        |     |      |       |

## Reading CSV File (pandas library)

```
import pandas as pd

df = pd.read_csv('rankings.csv')
```

# Load Data From JSON Files

JSON is a file format for storing nested data (array, dict)

```
players.json
1 {
2   "Kobe Bryant" :{
3     "Born": "08/23/1978",
4     "Number": ["8", "24"],
5     "Team": ["Los Angeles Lakers"]
6   },
7   "Michael Jordan":{
8     "Born": "02/17/1963",
9     "Number": ["23"],
10    "Team": ["Chicago Bulls", "Washington Wizards"]
11  }
12 }
```

## Reading JSON File (pandas Libarary)

```
import pandas as pd
df=pd.read_json("players.json")
```

# Web Scraping

---

## Open web pages

- urllib2 (<https://docs.python.org/2/library/urllib2.html>)
- request (<http://docs.python-requests.org/en/master/>)

## Parse web pages

- BeautifulSoup (<https://www.crummy.com/software/BeautifulSoup/>)
- lxml (<http://lxml.de/>)

## Putting everything together

- Scrapy (<https://scrapy.org/>)

# Before you scrape

---

Check to see if CSV, JSON, or XML version of an HTML page are available  
– better to use those

Check to see if there is a Python library that provides structured access  
(e.g., dataprep)

Check that you have permission to scrape

From [“Deb Nolan. Web Scraping & XML/Xpath”](#)

# If you do scrape

---

- Be careful to not to overburden the site with your requests
- Test code on small requests
- Save the results of each request so you don't have to repeat the request unnecessarily
- CAPTCHA



From "[Deb Nolan. Web Scraping & XML/Xpath](#)"

# Outline

---

## Data Collection

## Data Cleaning

- Dirty Data Problems
- Data Cleaning Tools
- Example: Outlier Detection

## Data Integration

# Dirty Data Problems

---

## From Stanford Data Integration Course:

- 1) Parsing text into fields (separator issues)
- 2) Missing required field (e.g. key field)
- 3) Different representations (iphone 2 vs iphone 2<sup>nd</sup> generation)
- 4) Fields too long (get truncated)
- 5) Formatting issues – especially dates
- 6) Outliers (age = 120)

# Data Cleaning Tools

## Python

- Missing Data (Pandas)
- Deduplication (Dedup)

## OpenRefine

- Open-source Software (<http://openrefine.org>)
- OpenRefine as a Service ([RefinePro](#))

## Data Wrangler

- The Stanford/Berkeley Wrangler research project
- Commercialized ([Trifacta](#))

## Not Many Tools.

That's why we are building DataPrep (<http://dataprep.ai>)

```
1 import pandas as pd
2 from dataprep.clean import clean_country
3 df = pd.DataFrame({"country": ["USA", "country: Canada", " France ",
4                               "233", " tr "]})
5 clean_country(df, "country")
```

|   | country         | country_clean |
|---|-----------------|---------------|
| 0 | USA             | United States |
| 1 | country: Canada | Canada        |
| 2 | France          | France        |
| 3 | 233             | Estonia       |
| 4 | tr              | Turkey        |

# Outlier Detection

---

The ages of employees in a US company

1 20 21 21 22 26 33 35 36 37 39 42 45 47 54 57 61 62

$$\text{Mean} = \frac{1}{n} \sum_{i=1}^n x_i = 37$$

$$\text{Stddev} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \text{mean})^2} = 16$$

$$[37 - 2 * 16, 37 + 2 * 16] = [4, 70]$$

# Outlier Detection

---

The ages of employees in a US company

|   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |     |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| 1 | 20 | 21 | 21 | 22 | 26 | 33 | 35 | 36 | 37 | 39 | 42 | 45 | 47 | 54 | 57 | 61 | 62 | 400 |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|

$$\text{Mean} = \frac{1}{n} \sum_{i=1}^n x_i = 56$$

$$\text{Stddev} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \text{mean})^2} = 83$$

$$[56 - 2 * 83, 56 + 2 * 83] = [-109, 221]$$

# Outlier Detection

---

The ages of employees in a US company

1 20 21 21 22 26 33 35 36 37 39 42 45 47 54 57 61 62 400

$$\text{Median} = \text{median}(X) = 37 \qquad [37 - 2 * 15, 37 + 2 * 15] = [7, 67]$$

$$\text{MAD} = \text{median}(X - \text{median}(X)) = 15$$

# Data Preparation Tasks

---

## Data Collection

## Data Cleaning

## Data Integration

- Data Integration Problem
- Three Steps (Schema Matching, Entity Resolution, Data Fusion)
- Example: Entity Resolution

# Data Integration Problem

Data Source 1 (from CourSys)

| First Name | Last Name | Mark |
|------------|-----------|------|
| Michael    | Jordan    | 50   |
| Kobe       | Bryant    | 48   |

Data Source 2 (from survey)

| Name        | Background        |
|-------------|-------------------|
| Mike Jordan | C++, CS, 4 years  |
| Kobe Bryant | Business, 2 years |



Integrated Data

| Name           | Mark | Background        |
|----------------|------|-------------------|
| Michael Jordan | 50   | C++, CS, 4 years  |
| Kobe Bryant    | 48   | Business, 2 years |

# Data Integration: Three Steps

---

## Schema Mapping

- Creating a global schema
- Mapping local schemas to the global schema

## Entity Resolution

- You will learn this in detail later

## Data Fusion

- Resolving conflicts based on some confidence scores

## Want to know more?

- Anhai Doan, Alon Y. Halevy, Zachary Ives. [Principles of Data Integration](#). Morgan Kaufmann Publishers, 2012.

# Entity Resolution

|                                                                                     |                                                                                                                                                                                                                                                                                                                                                                               |                                                                                                            |
|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|
|    | <p>Apple iPad 2 <b>MC775LL/A</b> Tablet (64GB Wifi + AT&amp;T 3G Black) <b>NEWE</b></p> <p>Apple iPad XX6LL/A Tablet (64GB, Wifi + AT&amp;T 3G, Black)<br/>NEWEST MODEL</p>                                                                                                                                                                                                   | <p><b>\$660</b> and up<br/>(3 stores)</p> <p><input type="checkbox"/> Compare<br/>(Share and Compare)</p>  |
|    | <p>Apple iPad 2 <b>MC775LL/A</b> 9.7" LED 64 GB Tablet<br/>Computer - Wi-Fi - 3G ...</p> <p><b>Brand Apple · Weight 1.40 lb · Screen size 9.70 in</b></p> <p>There's more to it. And even less of it. Two cameras for FaceTime and HD video recording. The dual-core A5 chip. The same 10-hour battery life. All in a thinner, lighter design.... <a href="#">more...</a></p> | <p><b>\$642</b> and up<br/>(10 stores)</p> <p><input type="checkbox"/> Compare<br/>(Share and Compare)</p> |
|  | <p><b>Black iPad 8gb</b></p> <p>The iPad 2 is the second and current generation of the iPad, a tablet computer designed, developed and marketed by Apple. It serves primarily as a platform for audio-visual media... <a href="#">more...</a></p>                                                                                                                             | <p><b>\$599</b><br/>eCRATER</p> <p><input type="checkbox"/> Compare<br/>(Share and Compare)</p>            |

# Output of Entity Resolution

---

| ID    | Product Name                           | Price |
|-------|----------------------------------------|-------|
| $r_1$ | iPad Two 16GB WiFi White               | \$490 |
| $r_2$ | iPad 2nd generation 16GB WiFi White    | \$469 |
| $r_3$ | iPhone 4th generation White 16GB       | \$545 |
| $r_4$ | Apple iPhone 3rd generation Black 16GB | \$375 |
| $r_5$ | Apple iPhone 4 16GB White              | \$520 |

$(r_1, r_2), (r_3, r_5)$

# Entity Resolution Techniques

## Similarity-based

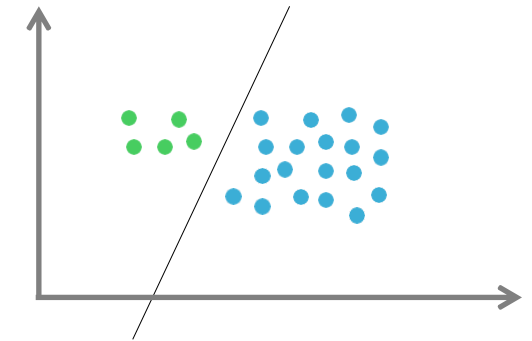
- Similarity Function  $\text{Jaccard}(r, s) = \left| \frac{r \cap s}{r \cup s} \right|$
- Threshold (e.g., 0.8)

$\text{Jaccard}(r1, r2) = 0.9 \geq 0.8$  Matching

$\text{Jaccard}(r4, r8) = 0.1 < 0.8$  Non-matching

## Learning-based

- Represent a pair of records as a feature vector



# Similarity-based

---

Suppose the similarity function is Jaccard.

## Problem Definition

Given a table  $T$  and a threshold  $\theta$ , the problem aims to find all record pairs  $(r, s) \in T \times T$  such that  $\text{Jaccard}(r, s) \geq \theta$

**The naïve solution needs  $n^2$  comparisons**

# Filtering-and-Verification

---

## Step 1. Filtering

- Removing obviously dissimilar pairs

## Step 2. Verification

- Computing Jaccard similarity only for the survived pairs

# How Does Filtering Work?

---

## What are “obviously dissimilar pairs”?

- Two records are obviously dissimilar if they do not share any word.
- In this case, their Jaccard similarity is zero, thus they will not be returned as a result and can be safely filtered.

## How can we efficiently return the record pairs that share at least one word?

- To help you understand the solution, let's first consider a simplified version of the problem, which assumes that each record only contains one word

# A simplified version

Suppose each record has only one word. Write an SQL query to do the filtering.

|                |        |
|----------------|--------|
| r <sub>1</sub> | Apple  |
| r <sub>2</sub> | Apple  |
| r <sub>3</sub> | Banana |
| r <sub>4</sub> | Orange |
| r <sub>5</sub> | Banana |

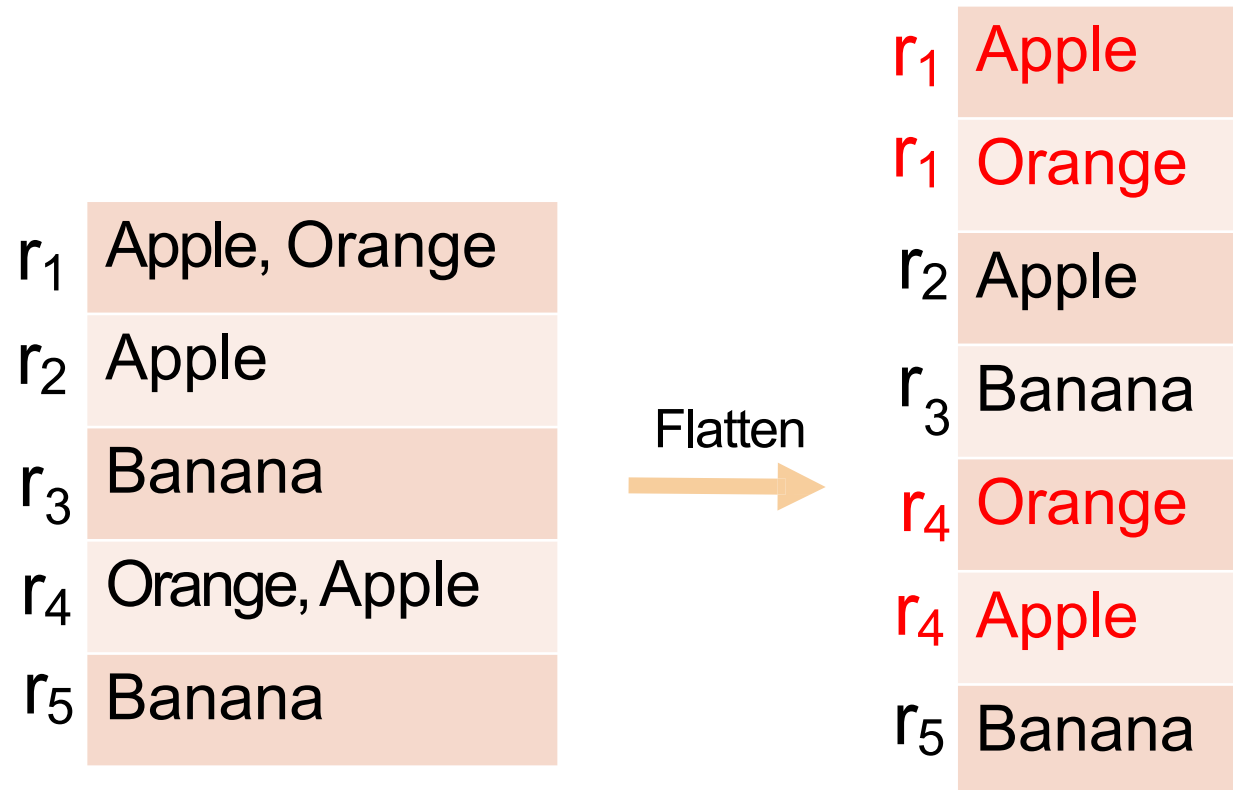
```
SELECT T1.id, T2.id
FROM Table T1, Table T2
WHERE T1.word = T2.word and T1.id < T2.id
```

Does it require  $n^2$  comparisons ?

**Output:** (r1, r2), (r3, r5)

# A general case

Suppose each record can have multiple words.



1. This new table can be thought of as the **inverted index** of the old table.
2. **Run the previous SQL on this new table** and remove redundant pairs.

# Not satisfied with efficiency?

---

## Exploring stronger filter conditions

- Filter the record pairs that share **zero** token
- Filter the record pairs that share **one** token
- ....
- Filter the record pairs that share **k** tokens

## Challenges

- How to develop efficient filter algorithms for these stronger conditions?

Jiannan Wang, Guoliang Li, Jianhua Feng.

[Can We Beat The Prefix Filtering? An Adaptive Framework for Similarity Join and Search.](#)

SIGMOD 2012:85-96.

# Not satisfied with result quality?

---

## TF-IDF

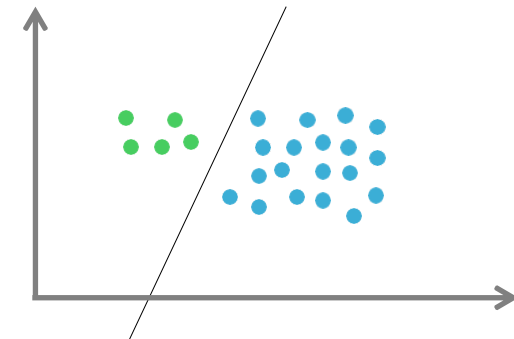
- Use weighted Jaccard:  $WJaccard(r, s) = \frac{wt(r \cap s)}{wt(r \cup s)}$

## Crowdsourcing

- Ask human to decide whether two records are matching or not

## Learning-based

- Model entity resolution as a classification problem



# Crowdsourcing

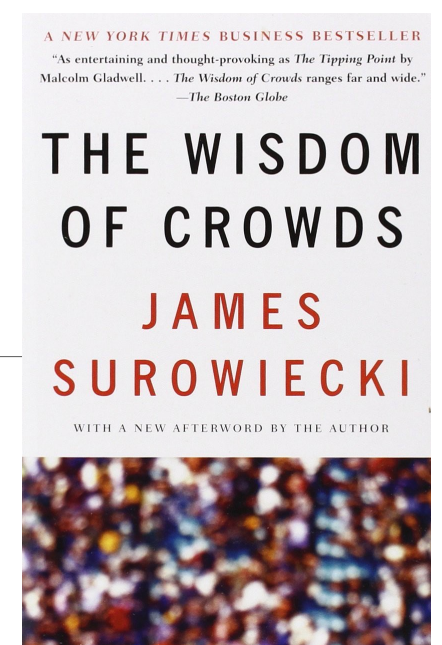
CMPT 884: Human-in-the-loop Data Management (SFU, Fall 2016)

<https://sfu-db.github.io/cmpt884-fall16/>

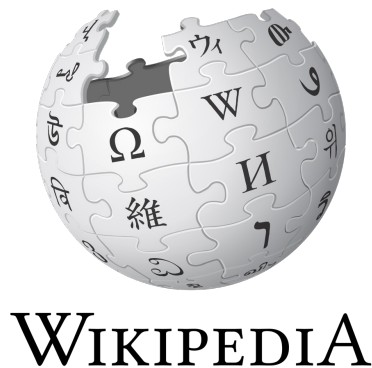
# The Wisdom of Crowds

## What does it mean?

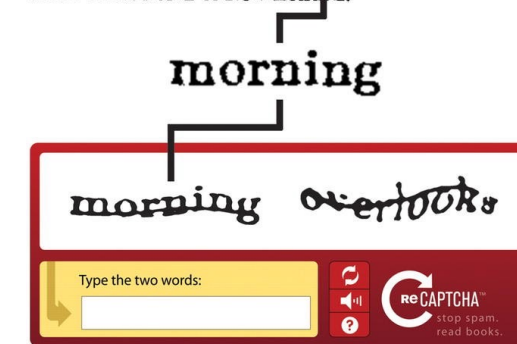
- Two heads are better than one



## Some famous examples



The Norwich line steamboat train, from New-London for Boston, this morning ran off the track seven miles north of New-London.



# Industrial Survey



| Company    | Team                                   | Persona                             |
|------------|----------------------------------------|-------------------------------------|
| Amazon     | Product classification                 | Largely single-case user            |
| Captricity | Focus of large part of company         | Largely single-case user            |
| Dropbox    | Single person consulting several teams | Multi-case user / Internal provider |
| Facebook   | Entities team                          | Multi-case user                     |
| Flipora    | Startup CTO                            | Multi-case user                     |
| GoDaddy    | Small business data extraction         | Multi-case user                     |
| Groupon    | Merchant data team                     | Multi-case user                     |
| Google     | Internal crowdsourcing team            | Internal provider                   |
| Google     | Web knowledge discovery team           | Multi-case user                     |
| LinkedIn   | Single person consulting several teams | Multi-case user / Internal provider |
| Microsoft  | Internal crowdsourcing team            | Internal provider                   |
| Microsoft  | Search relevance team                  | Multi-case user                     |
| Youtube    | Crowdsourcing team                     | Largely single-case user            |



# Amazon Mechanical Turk

500K+ workers\*

amazonmechanical turk

Get Started with Amazon Mechanical Turk

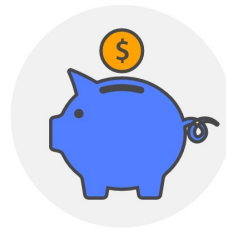


Create Tasks

Human intelligence through an API. Access a global, on-demand, 24/7 workforce.

Create a Requester account

or

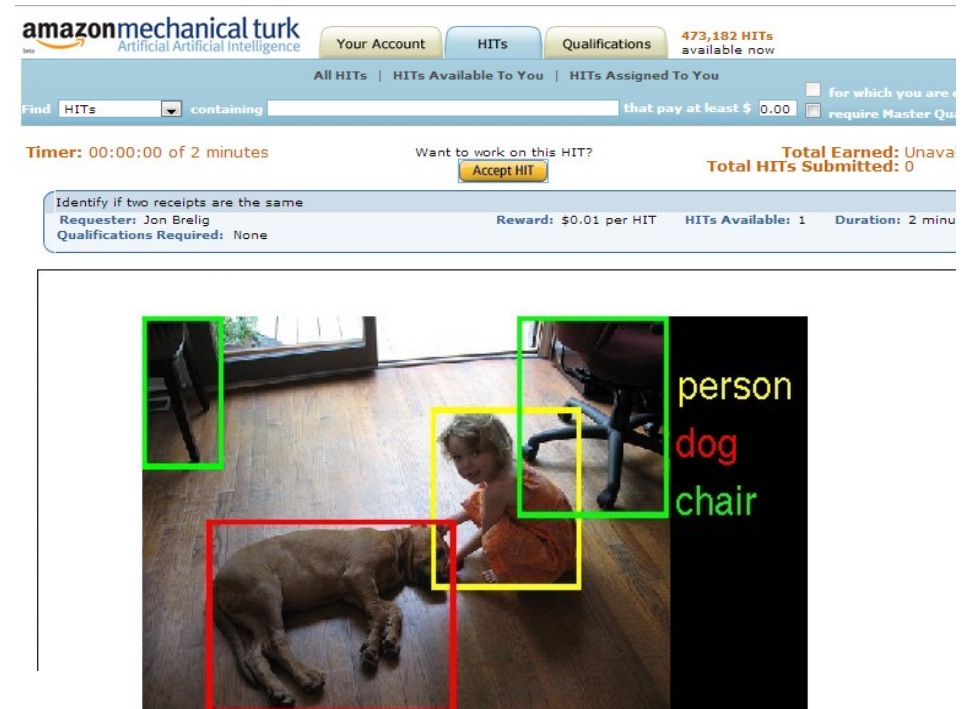


Make Money

Make money in your spare time. Get paid for completing simple tasks.

Create a Worker account

\* <https://requester.mturk.com/tour>

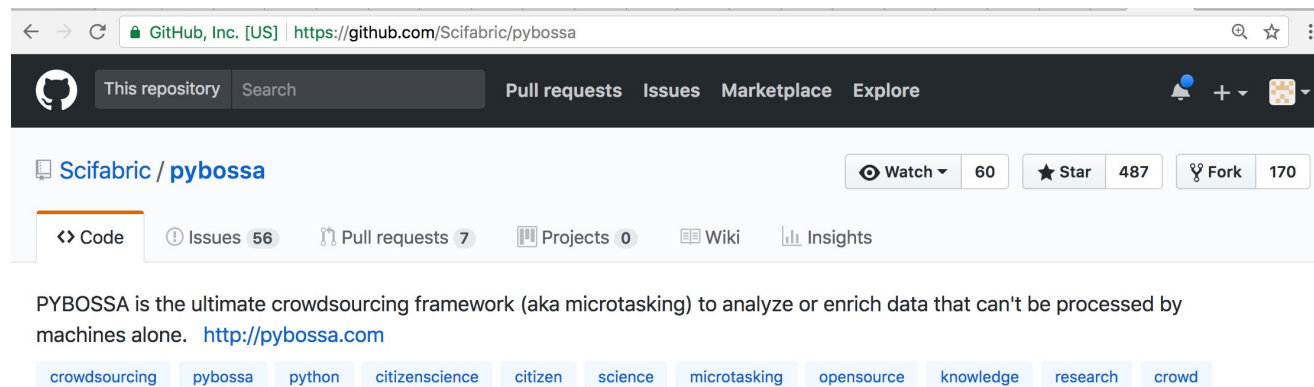


# Crowdsourcing may not work 😞

## What if your data is confidential?

- E.g., Medical Data, Customer Data

## Internal Crowdsourcing Platform



# Crowdsourcing may not work 😞

What if your data is so big?

- E.g., Label **10 million** images

# Crowdsourcing may not work ®

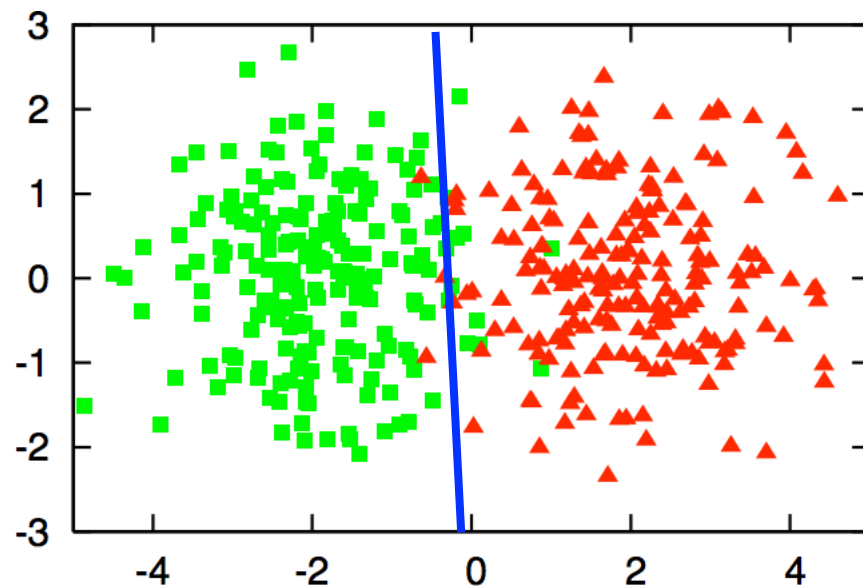
What if your data is so big?

- E.g., Label 10 million images

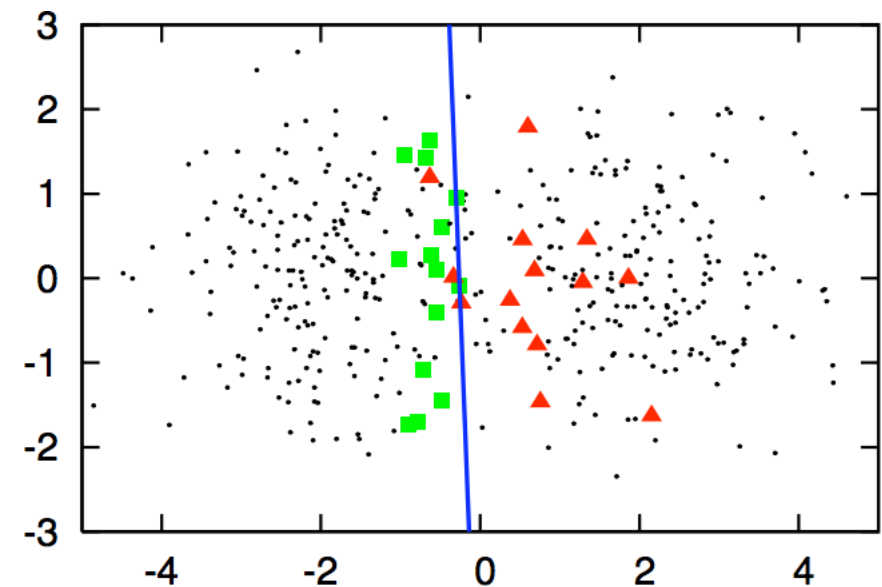
## Active Learning

# Active Learning

## Supervised Learning

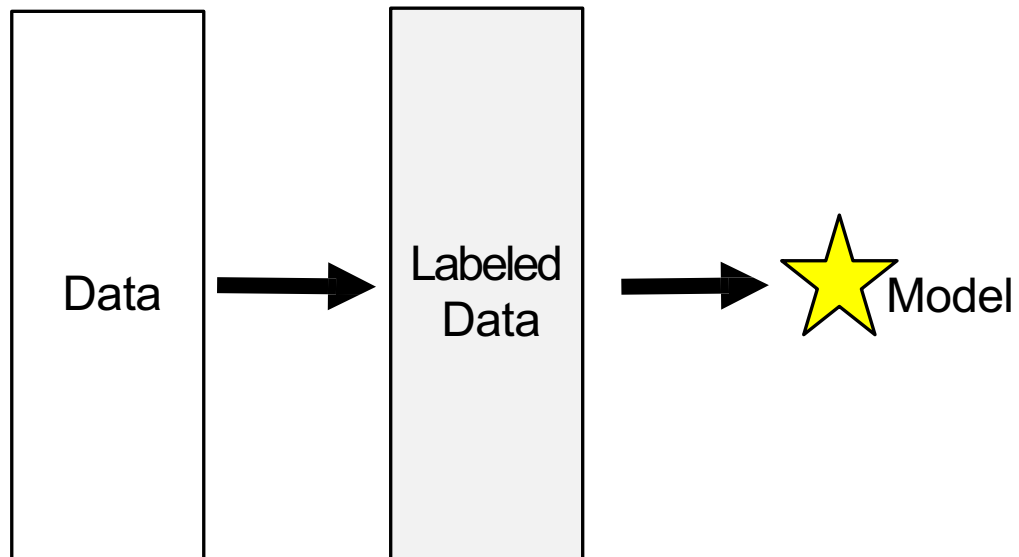


## Active Learning

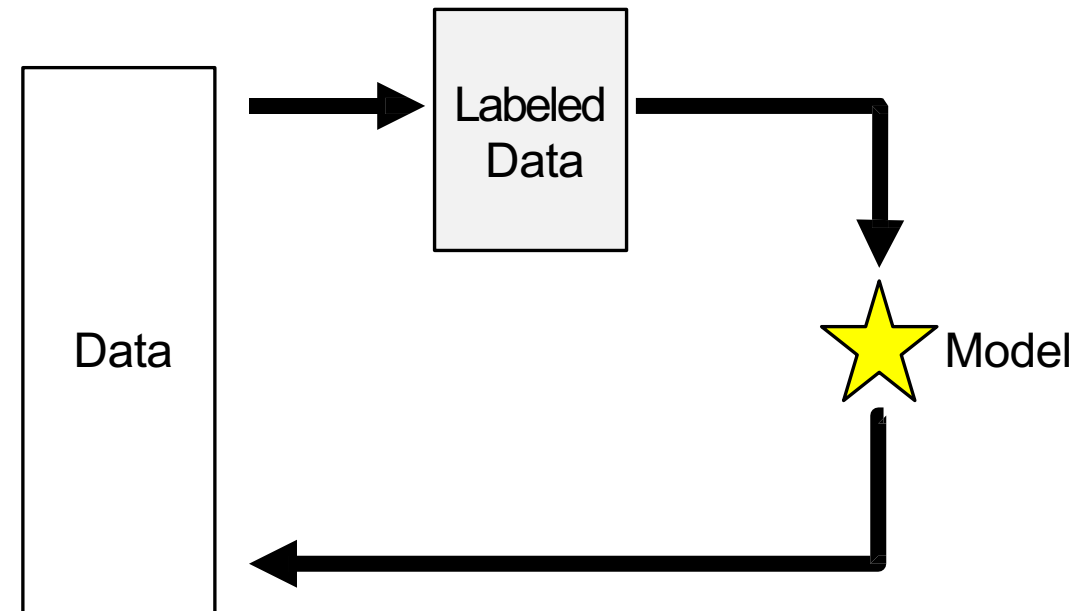


# Workflow

## Supervised Learning



## Active Learning



# Query Strategy

---

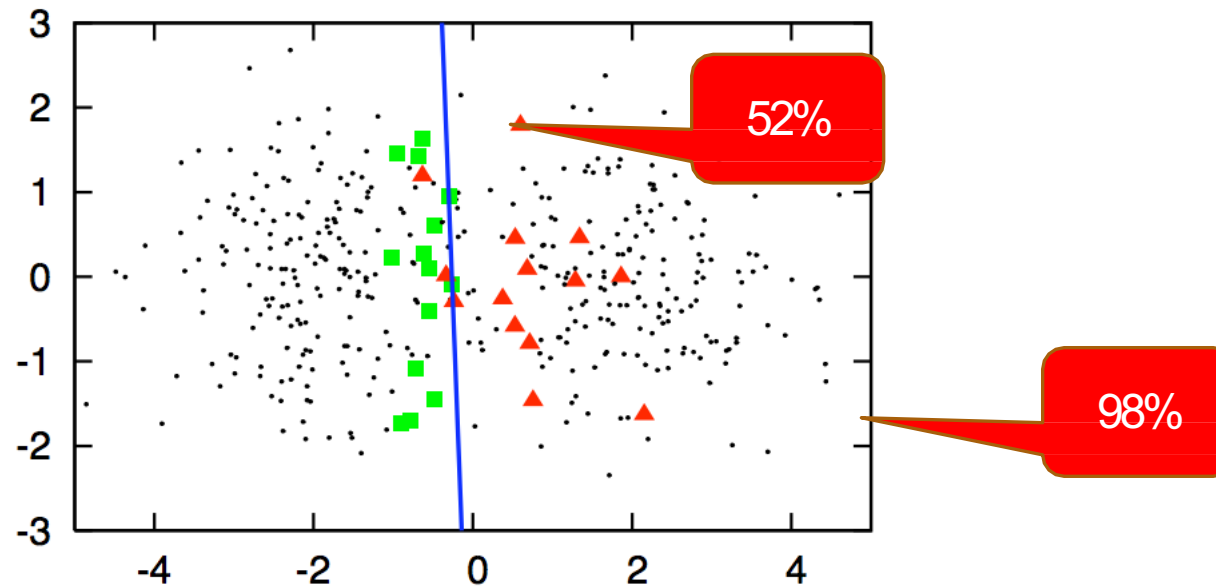
Which data points should be labeled?

- Uncertain Sampling
- Query-By-Committee
- Expected Error Reduction
- Expected Model Change
- Variance Reduction
- Density-Weighted Methods

Settles, Burr. "Active learning literature survey." University of Wisconsin, Madison 52.55-66 (2010): 11.

# Uncertain Sampling

Pick up most uncertain datapoints to label



Logistic Regression

- `predict_proba(X)`

# Summary

## Preppin' Data

A weekly challenge to help you learn to prepare data and use Tableau Prep

<https://preppindata.blogspot.com/>

---

### Data Collection

- Where to collect, How to Collect

### Data Cleaning

- Dirty Data Problems, Data-cleaning tools

### Data Integration

- Schema Mapping, Entity Resolution, Data Fusion

### Entity Resolution

- Similarity-based, Crowdsourcing, Active Learning