

# **Programming with Data Bootcamp: Lecture 3**

Slides courtesy of Sam Madden /  
Tim Kraska (6.S079)

## **Key ideas:**

- Entity Resolution (Data Integration)
- Value Imputation (Data Cleaning)
- Outlier Detection (Data Cleaning)
- Species Estimation (Data Cleaning)

<http://dsg.csail.mit.edu/6.S079/>



## EXAMPLE TASK



*How many people work in the US IT industry?*

*What is the avg revenue per employee in the tech industry?*

# EXAMPLE TASK

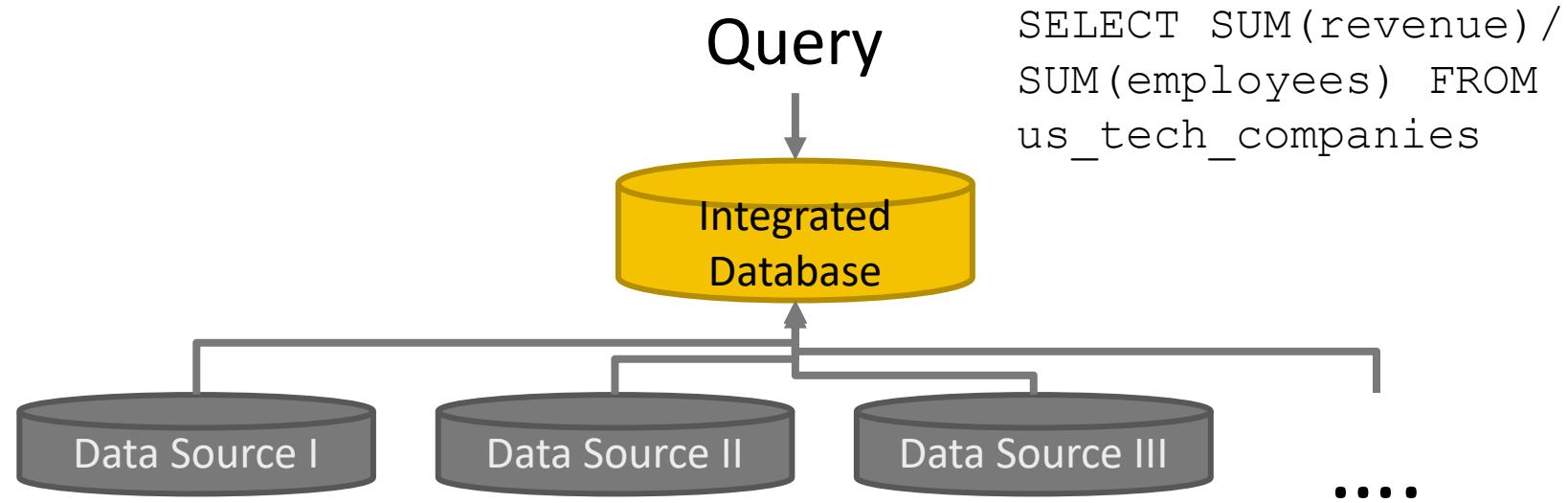
Rank [1]	Company	Fiscal Year Ending	Revenue (\$B) USD	Employees	Headquarters
1	Apple Inc.	30 September 2017 <sup>[2]</sup>	\$229.2 <sup>[1][3]</sup>	123,000 <sup>[3]</sup>	Cupertino, CA, US
2	Samsung Electronics	31 December 2017 <sup>[4]</sup>	\$211.9 <sup>[1][5][6]</sup>	320,670 <sup>[7][8]</sup>	Suwon, South Korea
3	Amazon	31 December 2017 <sup>[9][10]</sup>	\$177.9 <sup>[1][10]</sup>	613,300 <sup>[11]</sup>	Seattle, WA, US
4	Foxconn	31 December 2017 <sup>[12][13]</sup>	\$154.7–158 <sup>[1][13][14]</sup>	803,126 <sup>[15]</sup>	New Taipei City, Taiwan
5	Alphabet Inc.	31 December 2017 <sup>[16][17]</sup>	\$110.8 <sup>[1][17]</sup>	80,110 <sup>[18]</sup>	Mountain View, CA, US
6	Microsoft	30 June 2017 <sup>[19]</sup>	\$90.0 <sup>[1]</sup>	124,000 <sup>[19]</sup>	Redmond, WA, US
7	Huawei	31 December 2017 <sup>[20][21]</sup>	\$89.3–92.5 <sup>[1][21]</sup>	180,000	Shenzhen, China
8	Hitachi	31 March 2018 <sup>[22]</sup>	\$84.6 <sup>[1]</sup>	307,275	Tokyo, Japan
9	IBM	31 December 2017 <sup>[23][24]</sup>	\$79.1 <sup>[1]</sup>	397,800	Armonk, NY, US
10	Dell Technologies	31 January 2018 <sup>[25][26]</sup>	\$78.7 <sup>[1][26]</sup>	145,000 <sup>[25]</sup>	Round Rock, TX, US
11	Sony	31 March 2018 <sup>[27]</sup>	\$77. <sup>[1][28]</sup>	117,300 <sup>[27]</sup>	Tokyo, Japan
12	Panasonic	31 March 2018 <sup>[29]</sup>	\$72.0 <sup>[1]</sup>	274,143	Osaka, Japan
13	Intel	31 December 2017 <sup>[30]</sup>	\$62.8 <sup>[1]</sup>	102,700	Santa Clara, CA, US
14	LG Electronics	31 December 2017 <sup>[31]</sup>	\$54.3 <sup>[1]</sup>	74,000	Seoul, South Korea
15	JD.com	31 December 2017 <sup>[32]</sup>	\$54.0 <sup>[1]</sup>	157,831	Beijing, China
16	HP Inc.	31 October 2017 <sup>[33]</sup>	\$52.0 <sup>[1]</sup>	49,000	Palo Alto, CA, US

Private and semipublic companies with the most employees in the world			
Rank ♦	Employer	Country	Employees ♦
1	Walmart	United States	2,200,000
2	China National Petroleum	China	1,382,401
3	China Post Group	China	935,191
4	State Grid	China	917,717
5	Hon Hai Precision Industry (Foxconn)	Taiwan	667,680
6	Volkswagen	Germany	664,496
7	Amazon	United States	647,500
8	Sinopec Group	China	619,151
9	Compass Group	United Kingdom	595,841
10	United States Postal Service	United States	565,802

United States Largest Private Employers (as of 2017) <sup>[1][2][3][4]</sup> [hide]			
Rank ♦	Employer	Global number of Employees ♦	Median annual pay
1	Walmart	2,300,000	\$19,177
2	Amazon	469,690	\$36,969
	Deutsche Post DHL	499,018	
3	United Parcel Service	456,415	\$53,443
4	Yum! Brands	450,000	\$9,111
5	Kroger	449,000	\$21,075
6	Home Depot	413,000	\$20,095
7	Berkshire Hathaway	377,000	\$53,510 (BH directly employs c. 30 people. All the others are employed by the companies BH purchases.)
8	International Business Machines	366,000	\$55,088
9	FedEx	357,000	\$50,017
10	Target Corporation	345,000	\$20,581
11	General Electric	313,000	\$57,211
12	Walgreens Boots Alliance	290,000	\$31,132
13	Starbucks	277,000	\$12,754
14	Albertsons	273,000	
15	PepsiCo	263,000	\$47,801
16	Wells Fargo	262,700	\$60,466
17	Cognizant Technology Solutions	260,000	\$31,998
18	UnitedHealth Group	260,000	\$58,378
19	Lowe's	240,000	\$23,905
20	AT&T	268,540	\$95,814

,name, domain, year founded, industry, size range, locality, country, linkedin url, current employee estimate, total employee estimate 5872184, ibm, ibm.com, 1911, information technology and services, 10001+, "new york, new york, united states", united states, linkedin.com/company/ibm, 274047, 716906  
4425416, tata consultancy services, tcs.com, 1968, information technology and services, 10001+, "bombay, maharashtra, india", india, linkedin.com/company/tata-consultancy-services, 190771, 341369  
21074, accenture, accenture.com, 1989, information technology and services, 10001+, "dublin, dublin, ireland", ireland, linkedin.com/company/accenture, 190689, 455768  
2309813, us army, goarmy.com, 1800, military, 10001+, "alexandria, virginia, united states", united states, linkedin.com/company/us-army, 162163, 445958  
1558607, ey, ey.com, 1989, accounting, 10001+, "london, greater london, united kingdom", united kingdom, linkedin.com/company/ernstandyoung, 158363, 428960  
3844889, hewlett-packard, hpe.com, 1939, information technology and services, 10001+, "palo alto, california, united states", united states, linkedin.com/company/hewlett-packard-enterprise, 127952, 412952  
2959148, cognizant technology solutions, cognizant.com, 1994, information technology and services, 10001+, "teaneck, new jersey, united states", united states, linkedin.com/company/cognizant, 122031, 210020  
5944912, walmart, walmartcareers.com, 1962, retail, 10001+, "withee, wisconsin, united states", united states, linkedin.com/company/walmart, 120753, 272827  
3727010, microsoft, microsoft.com, 1975, computer software, 10001+, "redmond, washington, united states", united states, linkedin.com/company/microsoft, 116196, 276983  
3300741, at&t, att.com, 1876, telecommunications, 10001+, "dallas, texas, united states", united states, linkedin.com/company/at&t, 115188, 269659  
5412257, united states air force, airforce.com, 1947, defense & space, 10001+, "randolph, texas, united states", united states, linkedin.com/company/united-states-air-force, 113997, 316549  
2780814, pwc, pwc.com, 1998, accounting, 10001+, "new york, new york, united states", united states, linkedin.com/company/pwc, 111372, 379447  
3972223, wells fargo, wellsfargo.com, financial services, 10001+, "san francisco, california, united states", united states, linkedin.com/company/wellsfargo, 109532, 264101  
1454663, infosys, infosys.com, 1981, information technology and services, 10001+, "bangalore, karnataka, india", india, linkedin.com/company/infosys, 104753, 215718  
3221953, deloitte, deloitte.com, 1900, management consulting, 10001+, "new york, new york, united states", united states, linkedin.com/company/deloitte, 104112, 329145  
2994580, citi, citigroup.com, 1812, financial services, 10001+, "new york, new york, united states", united states, linkedin.com/company/citi, 101482, 298171  
5372097, bank of america, bankofamerica.com, 1968, banking, 10001+, "charlotte, north carolina, united states", united

# EXAMPLE TASK



On average what is the revenue per employee in the tech sector in the US?

# CLICKER: [HTTPS://CLICKER/MIT.EDU/6.S079/](https://clicker.mit.edu/6.S079/)

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	null
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
International Business Machine	10504; 1 New Orchard Rd	380k	-\$999B	\$12B
Microsoft	Albuquerque, Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	-	\$0.9B	\$1B
Tamr	64 Church St, Cambridge, MA 02138, United States	20	null	\$-Y

How many different **types** of errors can you find, which could influence our result (avg revenue per employee in the US )?

- a) 1-2 error types
- b) 3-4 error types
- c) 5-6 error types
- d) 7-8
- e) over 8

# CLICKER: [HTTPS://CLICKER/MIT.EDU/6.S079/](https://clicker.mit.edu/6.S079/)

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	null
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
International Business Machine	10504; 1 New Orchard Rd	380k	\$-999B	\$12B
Microsoft	Albuquerque, Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	-	\$0.9B	\$1B
Tamr	United States	0	~\$11B	~\$11B

**Duplicate Entities (Entity Resolution)**  
**Pattern Violation**

**Outdated data / wrong data**  
**Spelling mistakes / abbreviations**

**Encoding Error (nb in thousands)**

**Rule Violations**

**Missing values (known unknowns)**

# CLICKER: CLICKER.CSAIL.MIT.EDU/6.S080/

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	null
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
International Business Machine	10504; 1 New Orchard Rd	380k	-\$999B	\$12B
Microsoft	Albuquerque, Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	-	\$0.9B	\$1B
Tamr	64 Church St, Cambridge, MA 02138, United States	20	null	\$-Y
Amazon	??	??	??	??
Facebook	??	??	??	??
??	??	??	??	??
??	??	??	??	??

Unknown Unknowns

# OUTLINE

## Data Integration

- **Different schemas** → Schema matching
- **Duplicates** → Entity resolution
- **Contradicting data** → Data fusion

## Data Cleaning

- **Missing values** → Value imputation
- **Wrong data** → Outlier detection
- **Missing records** → Species estimation

# OUTLINE

## Data Integration

- **Different schemas** → Schema matching
- **Duplicates** → Entity resolution
- **Contradicting data** → data fusion

## Data Cleaning

- **Missing values** → Value imputation
- **Wrong data** → Outlier detection
- **Missing records** → Species estimation

# SCHEMA MATCHING

**Goal is to match columns from two tables, to produce a single table with the same schema**

**Complicated because people use different names for columns**

**Typical approach: find columns with a similar name, the same data type, and high overlap in values**

**Measuring similarity and overlap are problems we will need to address in entity resolution as well ...**

# OUTLINE

## Data Integration

- **Different schemas** → Schema matching
- **Duplicates** → Entity resolution
- **Contradicting data** → data fusion

## Data Cleaning

- **Missing values** → Value imputation
- **Wrong data** → Outlier detection
- **Missing records** → Species estimation

# ENTITY RESOLUTION

“[The] problem of identifying and linking/grouping different manifestations of the same real world object.”

## Challenges

- Fundamental ambiguity
- Diversity in representations (format, truncation, ambiguity)
- Errors
- Missing data
- Records from different times
- Relationships in addition to equality

# TEXT SIMILARITY

## Customer

Id	Name	Street	City	State	P-Code	Age
1	J Smith	123 University Ave	Seattle	Washington	98106	42
2	Mary Jones	245 3rd St	Redmond	WA	98052-1234	30
3	Bob Wilson	345 Broadway	Seattle	Washington	98101	19
4	M Jones	245 Third Street	Redmond	NULL	98052	299
5	Robert Wilson	345 Broadway St	Seattle	WA	98101	19
6	James Smith	123 Univ Ave	Seatle	WA	NULL	41
7	J Widom	123 University Ave	Palo Alto	CA	94305	NULL
...	...	...	...	...	...	...

# TEXTUAL SIMILARITY

## String Similarity function:

- $\text{Sim}(\text{string}, \text{string}) \rightarrow \text{numeric value}$

## A “good” similarity function:

- Strings representing the same concept  $\Rightarrow$  high similarity
- Strings representing different concepts  $\Rightarrow$  low similarity

# EDIT DISTANCE

**EditDistance(s1, s2):**

- Minimum number of edits to transform s1 to s2

**Edit:**

- Insert a character
- Delete a character
- Substitute a character

**Note:**  $\text{EditDistance}(s1, s2) = \text{EditDistance}(s2, s1)$

# EDIT DISTANCE

**EditDistance ("Provdince", "Providence") = 2**

Provdince  $\longrightarrow$  Providence  $\longrightarrow$  Providence

**EditDistance("Seattle", "Redmond") = 6**

Seattle ~~Reattle~~  $\longrightarrow$  ~~Redttle~~  $\longrightarrow$   
Redmtle Redm~~o~~ $\rightarrow$  Redmone  $\longrightarrow$   
 $\longrightarrow$  Redmond

# EDIT DISTANCE PROBLEMS

115<sup>th</sup> Waterman St., Providence, RI



EditDistance = 1

110<sup>th</sup> Waterman St., Providence, RI

Waterman Street, Providence, RI

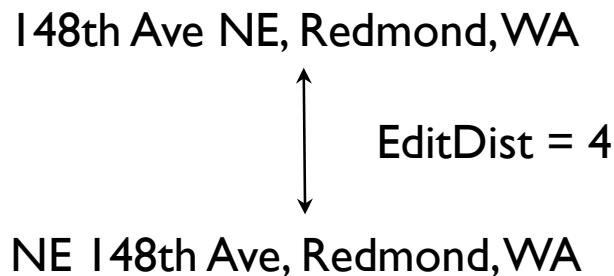
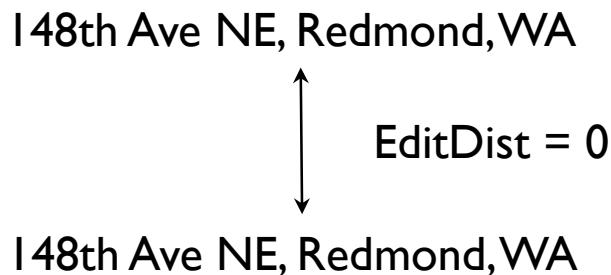


EditDistance = 4

Waterman St, Providence, RI

## Character Level vs. Word Level Similarity?

# EDIT DISTANCE PROBLEMS



Order sensitive Similarity?

# JACCARD SIMILARITY

- **Statistical measure**
- **Originally defined over sets**
- **String = set of words**

$$Jaccard(s1, s2) = \frac{|s1 \cap s2|}{|s1 \cup s2|}$$

- **Range of values = [0,1]**

# JACCARD SIMILARITY

| 48th Ave NE, Redmond, WA



| 40th Ave NE, Redmond, WA

$$Jaccard = \frac{4}{4 + 2} \approx 0.66$$

# JACCARD SIMILARITY

148th Ave NE, Redmond, WA



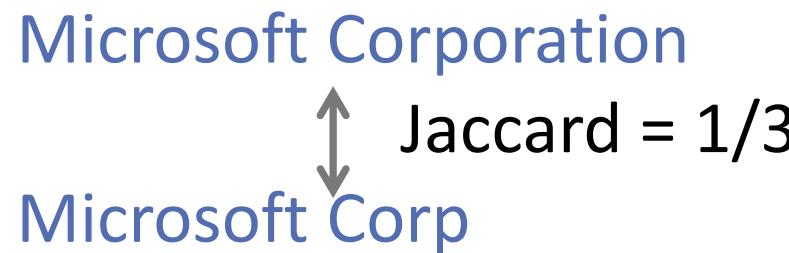
$$Jaccard = \frac{5}{5} = 1.0$$

NE 148th Ave, Redmond, WA

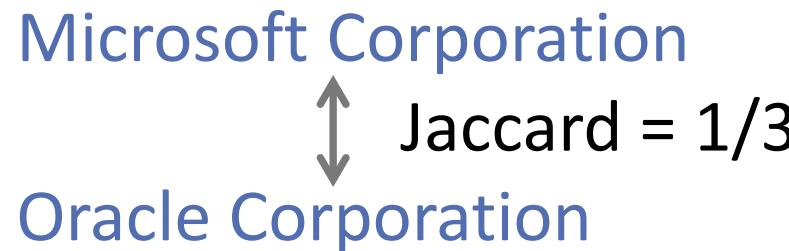
# CLICKER:

## WHICH JACCARD SIMILARITY IS WRONG

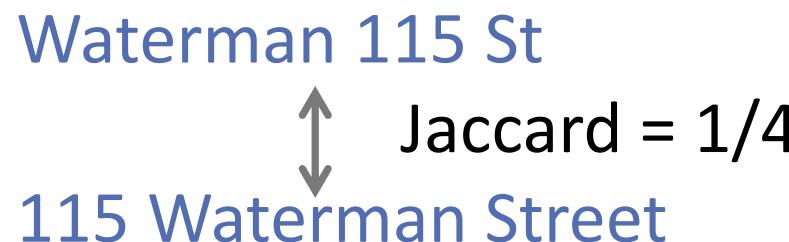
A)



B)



C)



# WHAT CAN WE DO ABOUT?

Microsoft Corporation



Microsoft Corp

Microsoft Corporation



Oracle Corporation

# JACCARD SIMILARITY

*Weight Function =  $wt: Elements \rightarrow \mathbb{R}^+$*

$$WtJaccard(s1, s2) = \frac{wt(s1 \cap s2)}{wt(s1 \cup s2)}$$

$$wt(s) = \sum_{e \in s} wt(e)$$

$wt(\text{"Microsoft"}) > wt(\text{"Corporation"})$

$Wt(\text{"Oracle"}) > wt(\text{"Corporation"})$

# IDF WEIGHTED

- IDF: Inverse Document Frequency

$$wt(word) = \log_e \left( \frac{\text{size of corpus}}{\text{frequency}(word)} \right)$$

- $\text{frequency}(word)$  = defined using some “corpus”:
  - large table of records
  - Wikipedia?

# IDF WEIGHTED JACCARD

**Microsoft Corporation**



$$\begin{aligned} \text{WtJaccard} &= 12.21 / (12.21 + 4.21 + 4.38) \\ &= 12.21 / 20.8 = 0.59 \end{aligned}$$

**Microsoft Corp**

**Microsoft Corporation**



$$\text{WtJaccard} = 4.21 / 26.57 = 0.16$$

**Oracle Corporation**

$$\log_e \left( \frac{1,000,000}{5} \right)$$

Word	Freq	IDF
Microsoft	5	12.21
Oracle	39	10.15
Corporation	14782	4.21
Corp	12496	4.38

Corpus size = 1M records

# OTHER SIMILARITY FUNCTIONS

- Affine edit distance
- Cosine similarity
- Hamming distance
- Generalized edit distance
- Jaro distance
- Monge-Elkan distance
- Q-gram
- Smith-Waterman distance
- Soundex distance
- TF/IDF
- ...many more

- No universally good similarity function
- Choice of similarity function depends on domains of interest, data instances, etc.

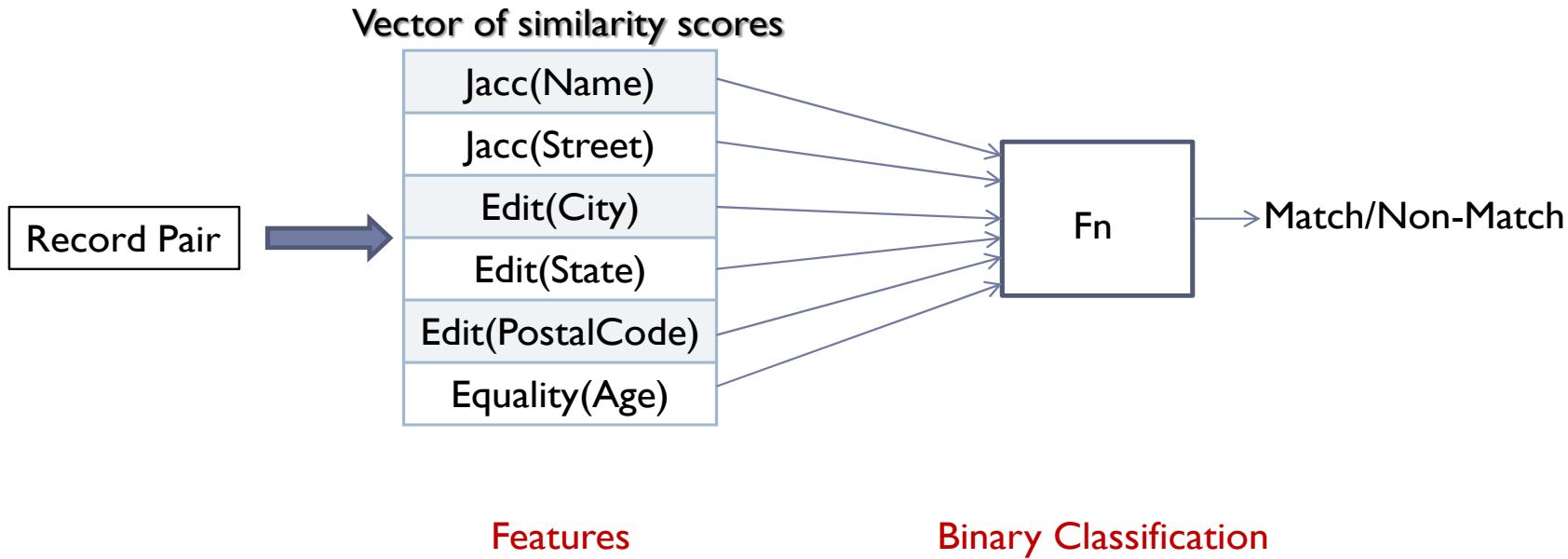
# RECORD MATCHING PROBLEMS

## Customer

Id	Name	Street	City	State	P-Code	Age
1	J Smith	123 University Ave	Seattle	Washington	98106	42
2	Mary Jones	245 3rd St	Redmond	WA	98052-1234	30
3	Bob Wilson	345 Broadway	Seattle	Washington	98101	19
4	M Jones	245 Third Street	Redmond	NULL	98052	299
5	Robert Wilson	345 Broadway St	Seattle	WA	98101	19
6	James Smith	123 Univ Ave	Seatle	WA	NULL	41
7	J Widom	123 University Ave	Palo Alto	CA	94305	NULL
...	...	...	...	...	...	...



# COMBINING SIMILARITY FUNCTIONS



# LEARNING-BASED APPROACH

Bob Wilson	345 Broadway	Seattle	Washington	98101	19
Robert Wilson	345 Broadway St	Seattle	WA	98101	19

Match

B Wilson	123 Broadway	Boise	Idaho	83712	19
Robert Wilson	345 Broadway St	Seattle	WA	98101	19

Non-Match

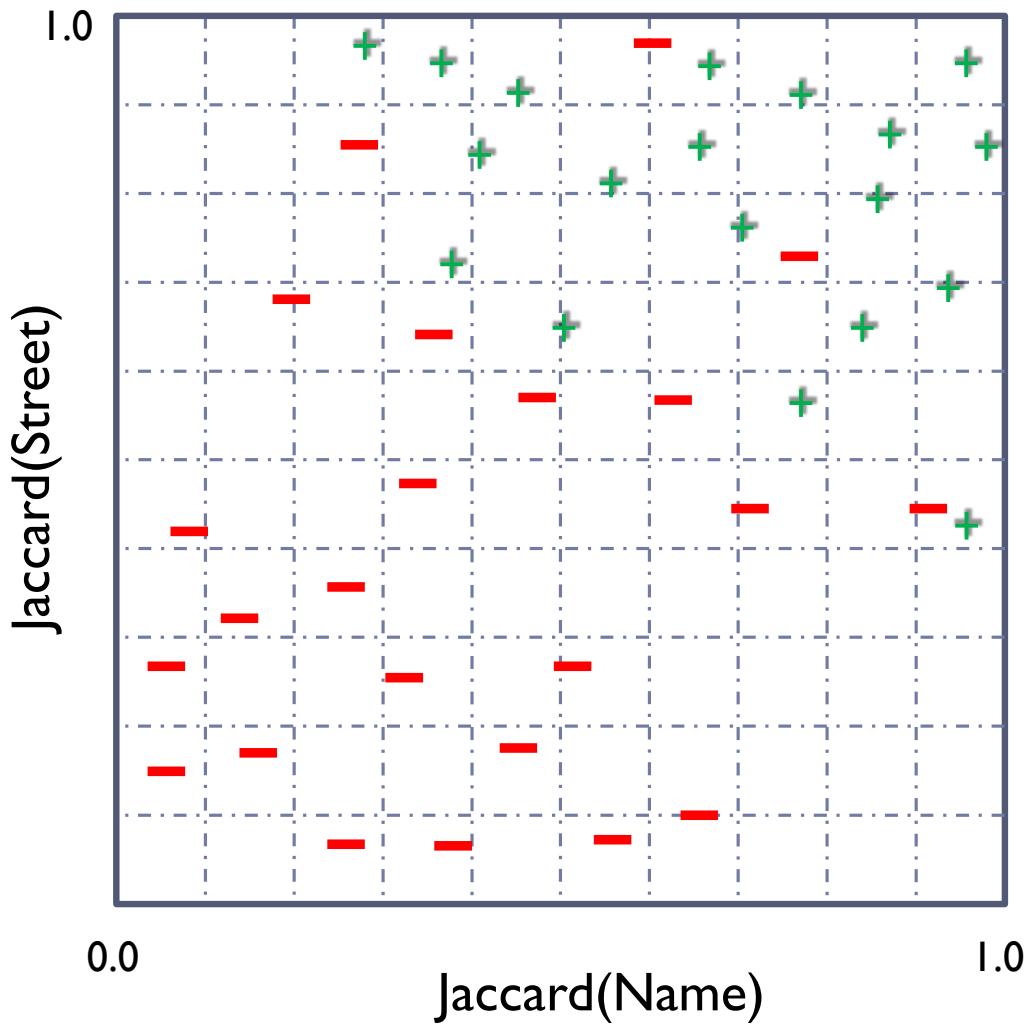
Mary Jones	245 3rd St	Redmond	WA	98052-1234	30
M Jones	245 Third Street	Redmond	NULL	98052	299

Match

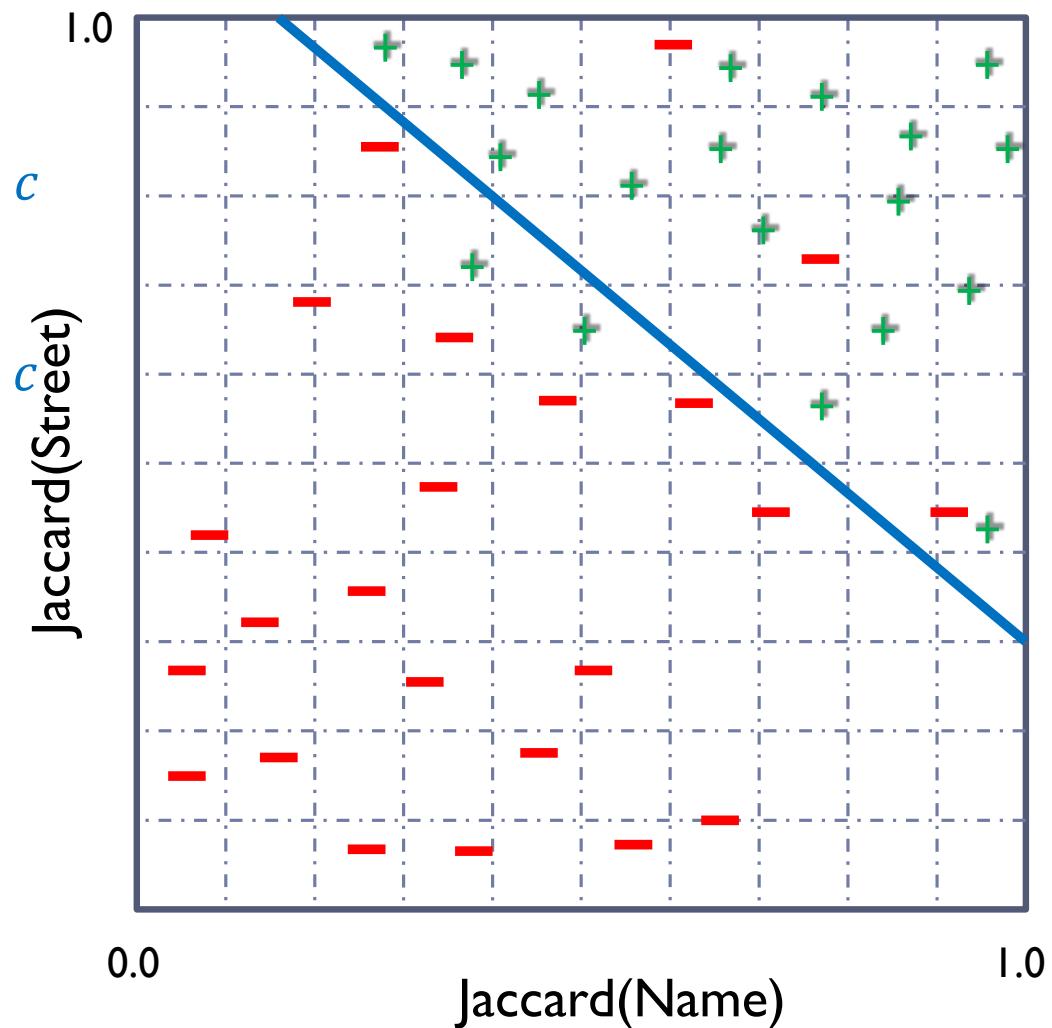
Mary Jones	245 3rd St	Redmond	WA	98052-1234	30
Robert Wilson	345 Broadway St	Seattle	WA	98101	19

Non-Match

# LEARNING BASED APPROACH



# LEARNING BASED APPROACH



0.73  $Ja_{-}$  (Name)  
+  
0.89  $Ja_{+}$  (Street)  $\geq 1$

# OUTLINE

## Data Integration

- **Different schemas** → Schema matching
- **Duplicates** → Entity resolution
- **Contradicting data** → data fusion

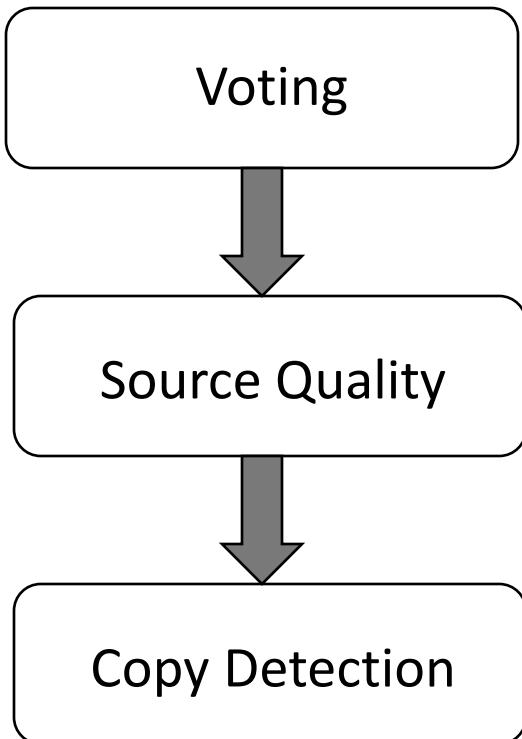
## Data Cleaning

- **Missing values** → Value imputation
- **Wrong data** → Outlier detection
- **Missing records** → Species estimation

# DATA FUSION'S THREE COMPONENTS

**Data fusion: voting + source quality + copy detection**

- Resolves inconsistency across diversity of sources



	S1	S2	S3	S4	S5
Jagadish	UM	<u>ATT</u>	UM	UM	<u>UI</u>
Dewitt	MSR	MSR	<u>UW</u>	<u>UW</u>	<u>UW</u>
Bernstein	MSR	MSR	MSR	MSR	MSR
Carey	UCI	<u>ATT</u>	<u>BEA</u>	<u>BEA</u>	<u>BEA</u>
Franklin	UCB	UCB	<u>UMD</u>	<u>UMD</u>	<u>UMD</u>

# OUTLINE

## Data Integration

- **Different schemas** → Schema matching
- **Duplicates** → Entity resolution
- **Contradicting data** → data fusion

## Data Cleaning

- **Missing values** → Value imputation
- **Wrong data** → Outlier detection
- **Missing records** → Species estimation

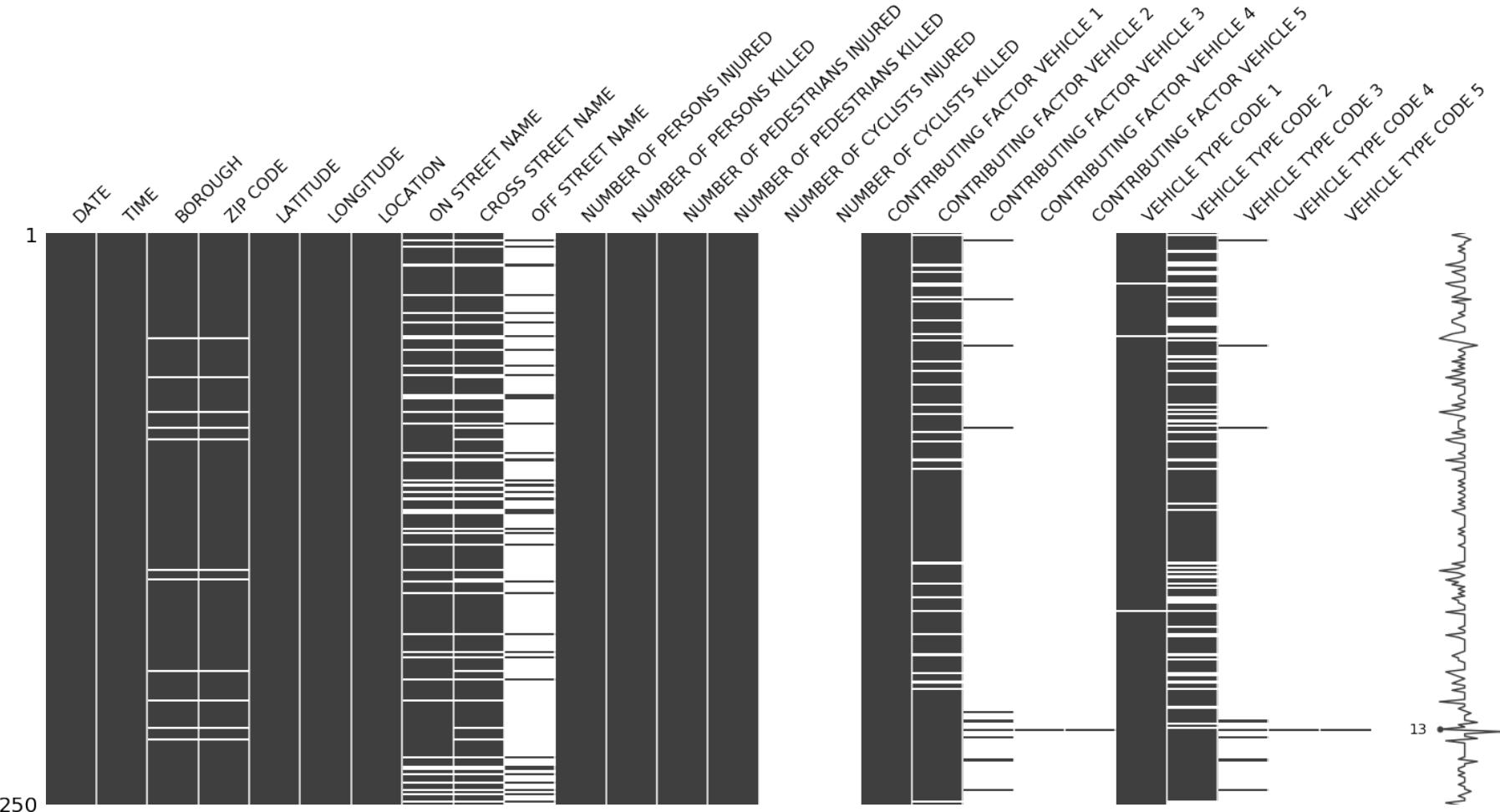
# WHY ARE THE VALUES MISSING?

- **Missing Completely at Random (MCAR)**
  - Includes missing by design. For example: Survey randomly selects questions to reduce load
- **Missing at Random (MAR)**
  - Better name: Missing Conditionally at Random
  - Systematic relationship between the propensity of missing values and the *observed* data, but *not* the missing data.  
--> if we can control for this conditional variable, we can get a random subset.
- **Missing Not at Random, MNAR**
  - Relationship between the propensity of a value to be missing and its values.
  - Lowest education are missing on education or the sickest people are most likely to drop out of the study.
  - MNAR is called “non-ignorable” because the missing data mechanism itself has to be modeled as you deal with the missing data.

**Note:** null values are often encoded in various ways. Be aware of it!  
Null, “null”, n/a, “”, 0, “empty”, 99999, 200.

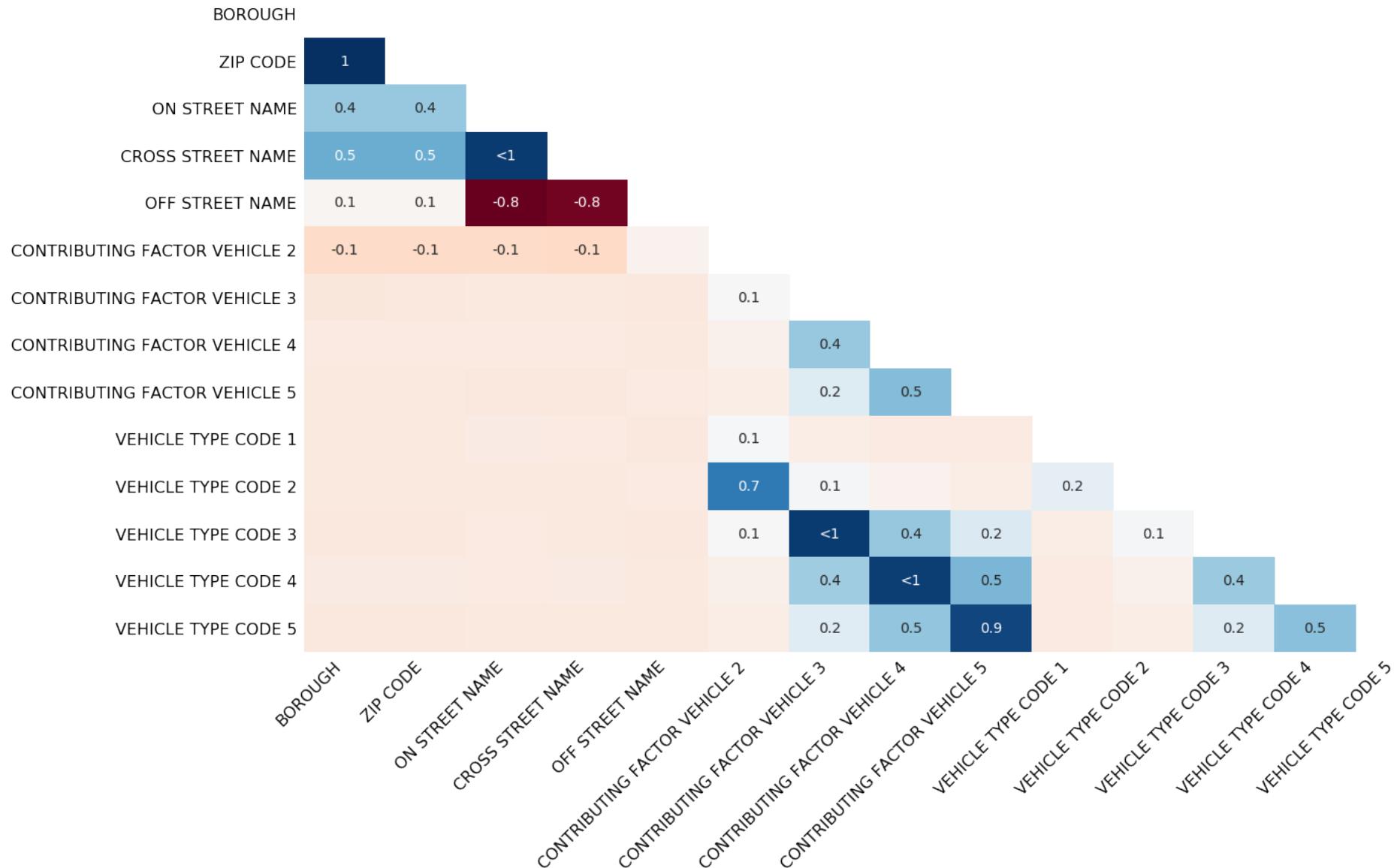
**HOW DO YOU START ADDRESSING  
MISSING VALUES?**

# VISUALIZATIONS TO DETECT BIAS

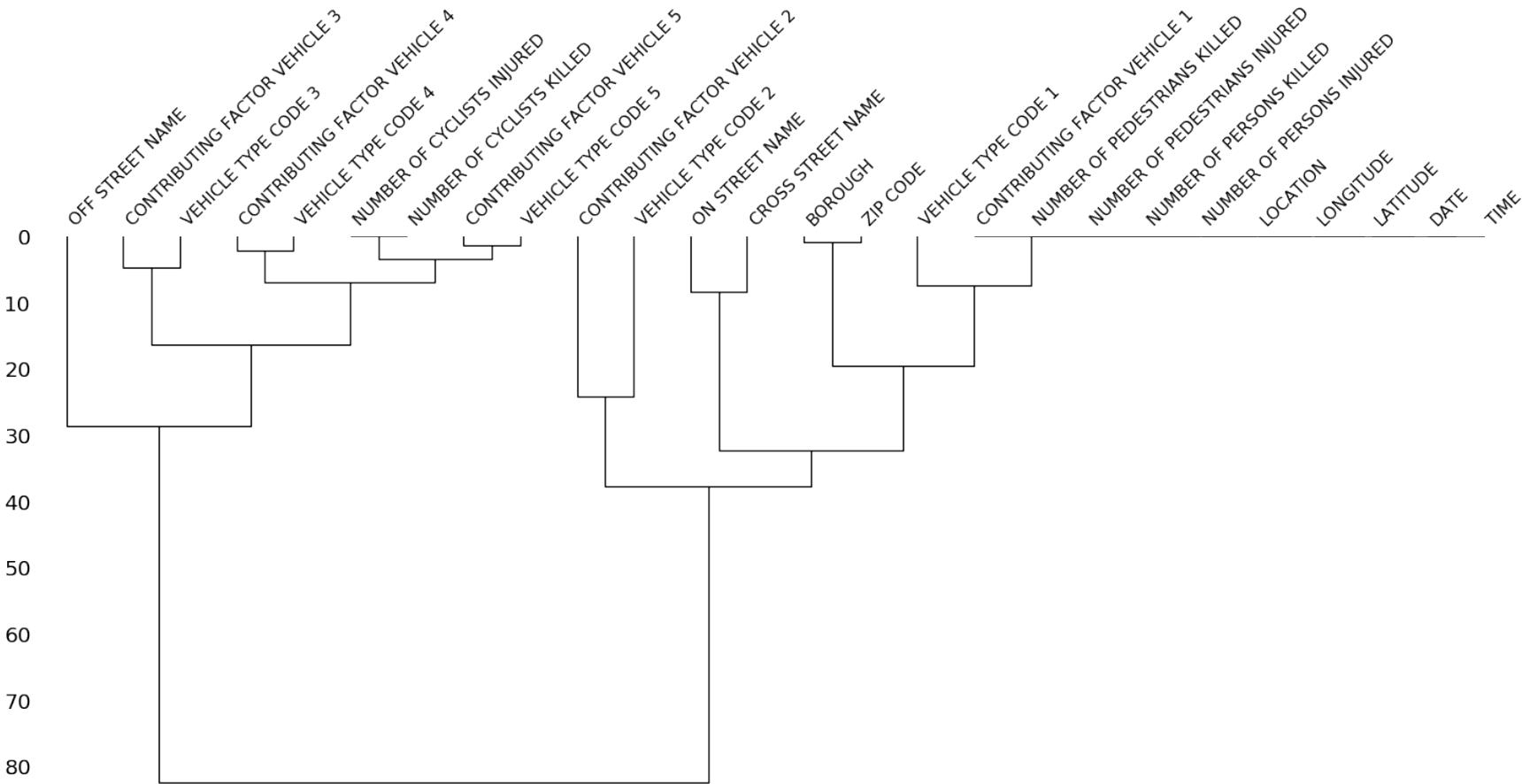


A lot of tips here: <https://github.com/ResidentMario/missingno>

# VISUALIZATIONS TO DETECT BIAS

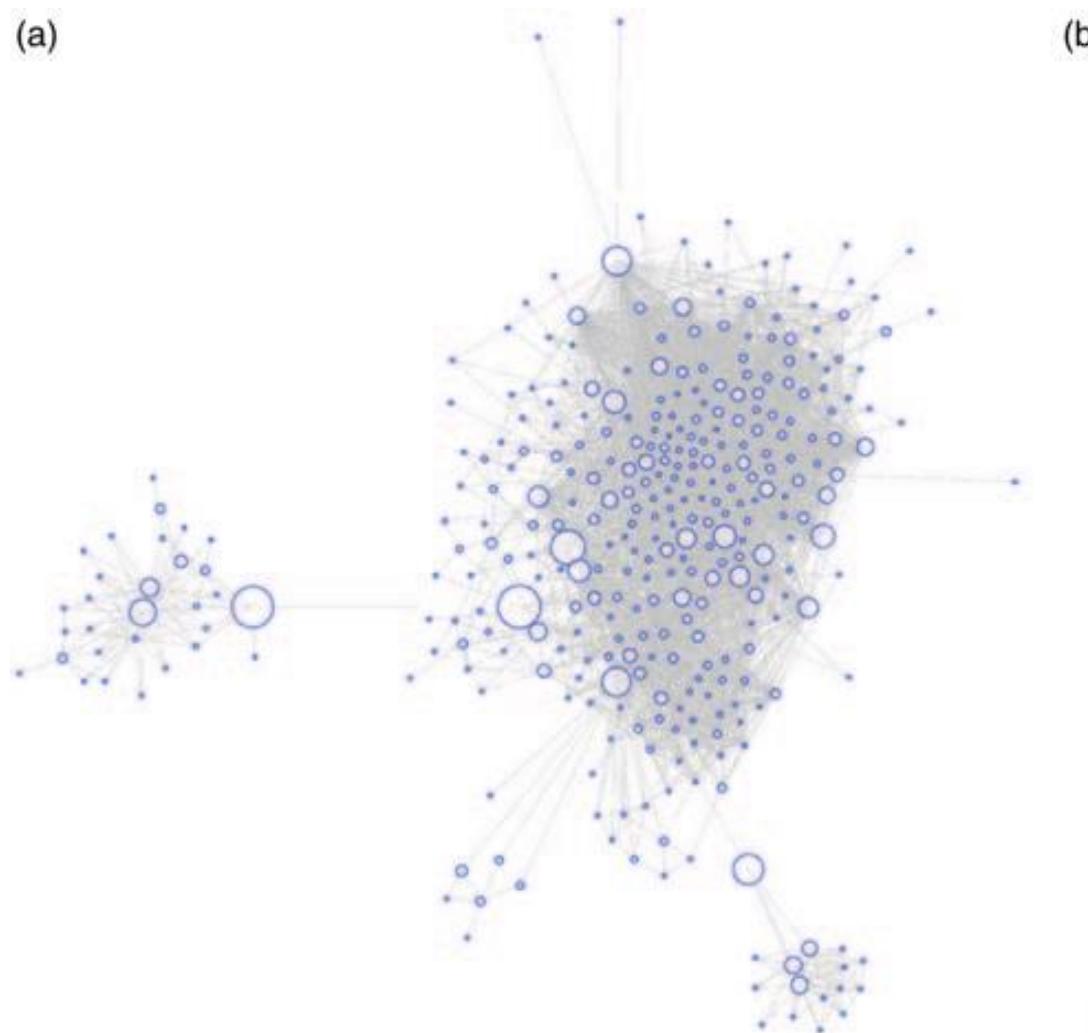


# VISUALIZATIONS TO DETECT BIAS



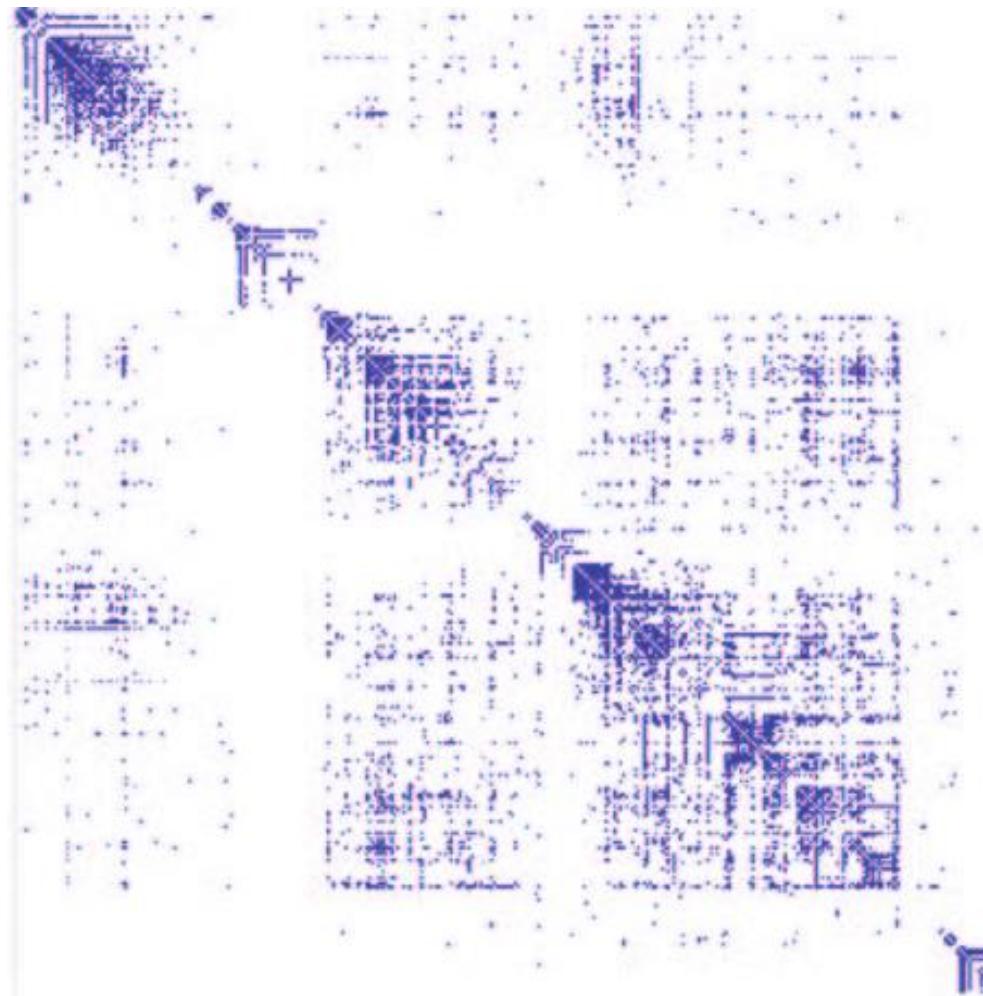
Alternative: Frequent pattern mining

# FACEBOOK SOCIAL GRAPH: VISUALIZATION THE NODE-LINK DIAGRAM



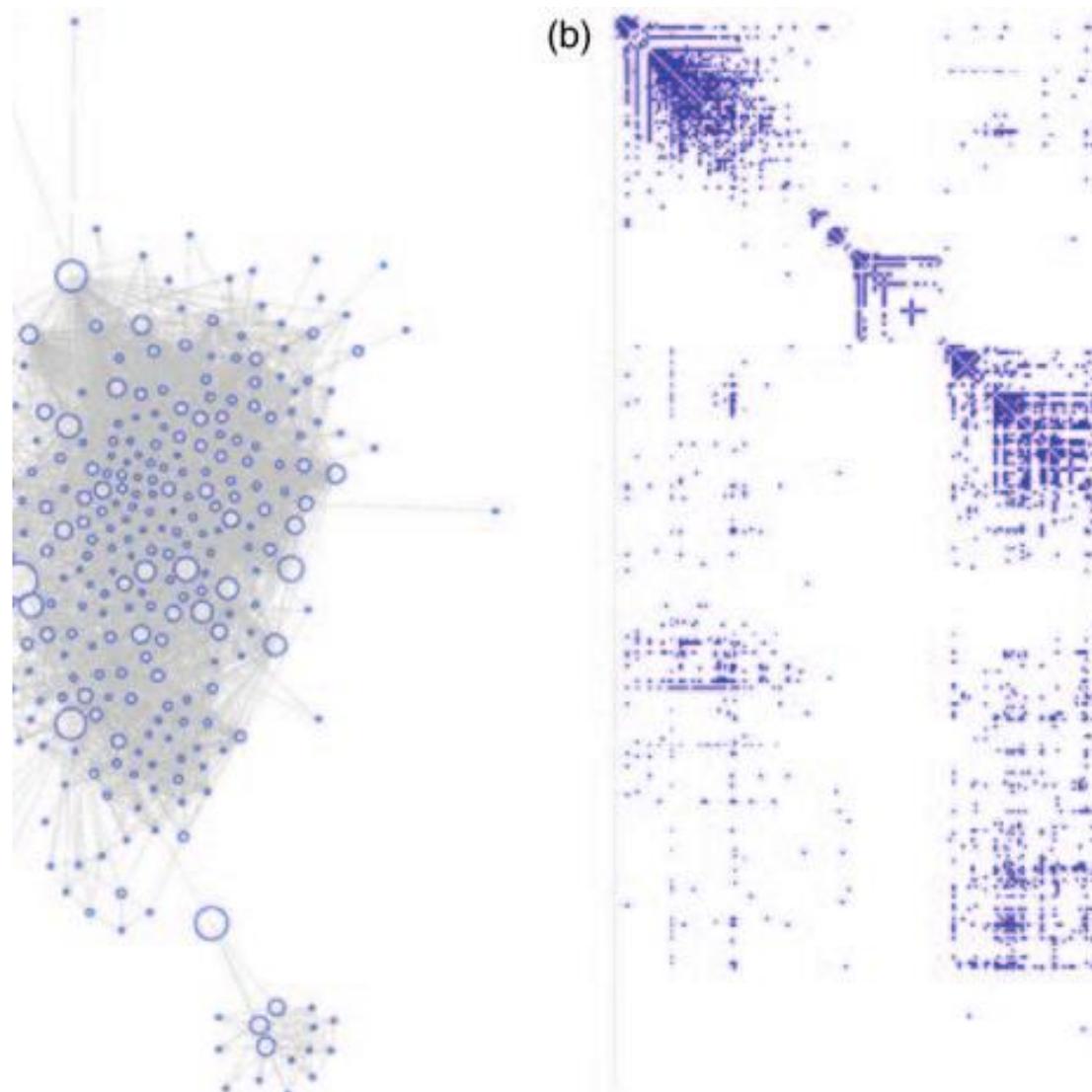
[Sean Kandel et al: Research directions in data wrangling: Visualizations and transformations for usable and credible data, Information Visualization, 2011]

# FACEBOOK SOCIAL GRAPH: VISUALIZATION THE NODE-LINK DIAGRAM



[Sean Kandel et al: Research directions in data wrangling: Visualizations and transformations for usable and credible data, Information Visualization, 2011]

# FACEBOOK SOCIAL GRAPH: SORTING BY RAW DATA



[Sean Kandel et al: Research directions in data wrangling: Visualizations and transformations for usable and credible data, Information Visualization, 2011]

**CLASS TASK:**  
COME UP WITH AT LEAST 5 TECHNIQUES  
TO DEAL WITH MISSING VALUES

# TECHNIQUES TO DEAL WITH MISSING VALUES (ONLY FOR MCAR / MAR)

- Pairwise (rarely used)
- Listwise deletion (better)
- Mean Substitution
- Dummy variable adjustment
- Maximum Likelihood Estimation
- Random sample from existing values/ reasonable distribution
- Multiple Imputation

Special cases:

- Last Observation
- Techniques for categorical values

# PAIRWISE AND LISTWISE DELETION

```
SELECT SUM(revenue) /  
SUM(employees) FROM  
us_tech_companies
```

## Pairwise Deletion

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66	\$215B	\$45B
IBM	1 New Orchard Rd; 10504, USA	380k	\$80B	\$12B
Microsoft	Albuquerque, New Mexico, USA	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States		\$5M	\$8M
Fair	64 Church St, Cambridge, MA 02138, USA	20	\$-X	\$-Y

## Listwise Deletion

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66	\$215B	\$45B
IBM	1 New Orchard Rd; 10504, USA	380k	\$80B	\$12B
Microsoft	Albuquerque, New Mexico, USA	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States		\$5M	\$8M
Fair	64 Church St, Cambridge, MA 02138, USA	20	\$-X	\$-Y

# PAIRWISE AND LISTWISE DELETION

## Pairwise Deletion

- Only cases relating to each pair of variables with missing data involved in an analysis are deleted.
- Advantage: keeps as many cases as possible for each analysis, uses all information possible with each analysis
- Disadvantage: cannot compare analyses because sample is different each time, sample size vary for each parameter estimation, can obtain nonsense results

## Listwise Deletion

- Only analyze cases with available data on each variable
- Advantage: simplicity and comparability across analyses
- Disadvantage: reduces statistical power (reduced sample size), not use all information, estimates may be biased if data not MCAR

# FIRST INITIAL CLEANING

## **Look for fields with very high percentage of missing fields**

- It may be necessary to exclude field and use an alternative

## **Look for records with a high percentage of missing fields**

- Consider excluding the case
- For example, someone who has started inputting a survey and given up after two questions!

**Document that you did delete them. Very risky to forget it**

# UNIVARIATE SINGLE IMPUTATION

## MEAN SUBSTITUTION

### Mean Substitution (do not use)

- Replace missing value with the sample mean or mode. Then, run analyses as if all complete cases
- Advantage: We can use complete case analyses
- Disadvantage: Reduces variability, weakens the correlation estimates because it ignores the relationship between variables, it creates artificial band
- Unless the proportion of missing data is low, do not use this method.
- Inappropriate for categorical variables.

### Dummy variable adjustment

- Create an indicator variable for missing value (1 for missing, 0 for observed), impute missing value to a constant (such as mean)

# MULTIVARIATE SINGLE IMPUTATION

## Regression imputation

- Replace missing values with predicted score from regression equation. Use complete cases to regress the variable with incomplete data on the other complete variables.

# MULTIVARIATE IMPUTATION

## Regression imputation

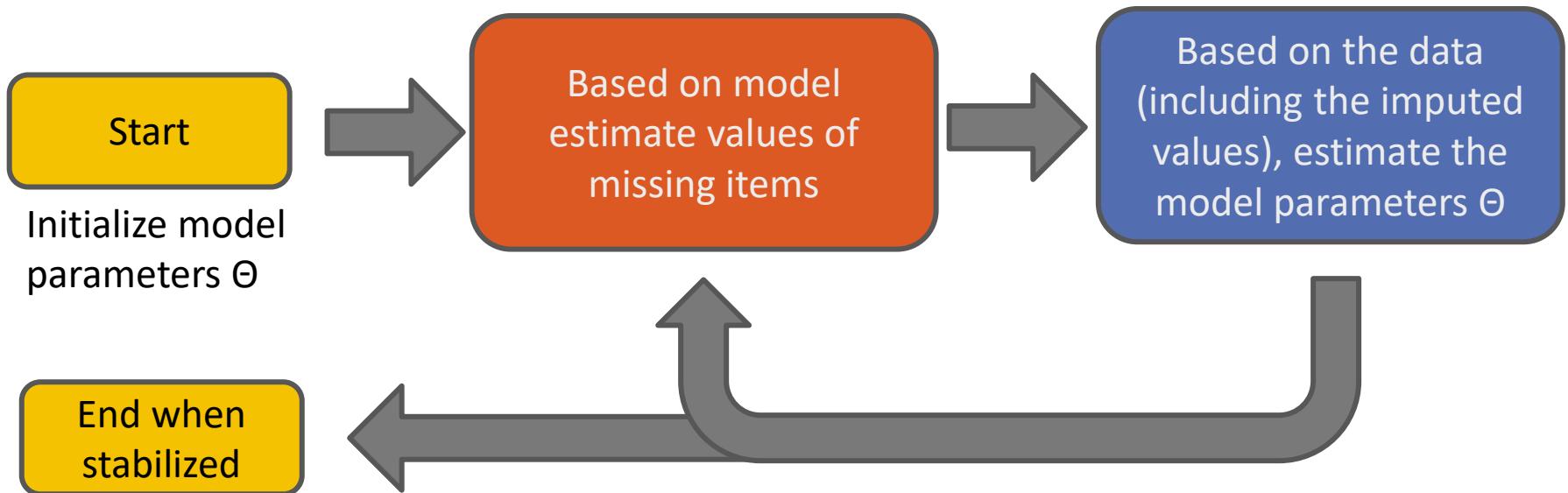
- Replace missing values with predicted score from regression equation. Use complete cases to regress the variable with incomplete data on the other complete variables.
- Advantage: Uses information from the observed data, gives better results than previous ones
- Disadvantage: over-estimates model fit and correlation estimates, weakens variance

## Maximum Likelihood Estimation (MICE)

- Identifies the set of parameter values that produces the highest log-likelihood.
  - ML estimate: value that is most likely to have resulted in the observed data.
- Advantage: uses full information (both complete and incomplete) to calculate the log-likelihood, unbiased parameter estimates with MCAR/MAR data
- Disadvantage: Standard errors biased downward but this can be adjusted by using observed information matrix

<https://scikit-learn.org/stable/modules/impute.html>

# EM ALGORITHM



# EM IMPUTATION METHODS

According to the key result of Dempster, Laird and Rubin (1977),  $\theta^{(t+1)}$  is better estimate than  $\theta^{(t)}$ , because the change from  $\theta^{(t)}$  to  $\theta^{(t+1)}$  in each iteration increases the log likelihood,

$$l(\theta^{(t+1)}|Y_{obs}) \geq l(\theta^{(t)}|Y_{obs}).$$

Therefore, iteration of EM algorithm can be considered in two steps: **Expectation Step** and **Maximization Step**.

**E-Step:** In this step, the function  $Q(\theta|\theta^{(t)})$  is calculated as the conditional expectation of complete data log likelihood over the conditional predictive distribution,  $f(Y_{mis}|Y_{obs}, \theta^{(t)})$ , of  $Y_{mis}$  given  $Y_{obs}$  and a current estimate of  $\theta$ , say  $\theta^{(t)}$ .

**M-Step:** In this step, estimation of  $\theta^{(t+1)}$  is carried out as if there were no missing data which is achieved by maximizing  $Q(\theta|\theta^{(t)})$  from E-step.

In order to define convergency of iterations, differences of parameter estimations derived in the each iteration are considered. If the difference of consecutive estimates less than selected threshold value, then iterations are stopped. Estimations from the last iteration are used as parameter estimations.

We will cover this algorithm in more depth later

# MULTIVARIATE SINGLE IMPUTATION

## Nearest neighbors imputation

- KNN defines for each sample or individual a set of K-nearest neighbors and then replaces the missing data for a given variable by averaging (non-missing) values of its neighbors
- Advantage: Simple, uses information from the observed data, experimentally shows good performance
- Disadvantage: not statistically grounded, might over-estimates model fit and correlation

## Fuzzy K-means Clustering

## Bayesian Principal Component Analysis

## Deep Learning-based approaches

....

<https://scikit-learn.org/stable/modules/impute.html>

# SIMPLE STOCHASTIC IMPUTATION

## Random sample from existing values:

- Randomly generate an integer from 1 to  $n - n_{\text{missing}}$ , then replace the missing value with the corresponding observation that you chose randomly

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	\$10B
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66k	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
Microsoft	Albuquerque, New Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States		\$5M	\$8M

# SIMPLE STOCHASTIC IMPUTATION

## Random sample from existing values:

- Randomly generate an integer from 1 to  $\max(n_{\text{missing}})$  then replace the missing value with the corresponding observation that you chose randomly

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	\$10B
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66k	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
Microsoft	Albuquerque, New Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	66k	\$5M	\$8M

- Randomly generate number between 1 and 4: Say 2 → Replace  $Y_{3,5}$  by  $Y_{2,3} = 66k$

# SIMPLE STOCHASTIC IMPUTATION

## Random sample from existing values:

- Randomly generate an integer from 1 to  $n-n_{\text{missing}}$ , then replace the missing value with the corresponding observation that you chose randomly

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	\$10B
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66k	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
Microsoft	Albuquerque, New Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	66k	\$5M	\$8M

- Randomly generate number between 1 and 4: Say 2 → Replace  $Y_{3,5}$  by  $Y_{2,3}=66k$
- Disadvantage: It may change the distribution of data
- **Hot-deck approach:** draws are made from units with complete data that are 'similar' to the one with missing values (donors).

# SIMPLE STOCHASTIC IMPUTATION

## Random sample from existing values:

- Randomly generate an integer from 1 to  $n-n_{\text{missing}}$ , then replace the missing value with the corresponding observation that you chose randomly

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	\$10B
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66k	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
Microsoft	Albuquerque, New Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	66k	\$5M	\$8M

- Randomly generate number between 1 and 4: Say 2 → Replace  $Y_{3,5}$  by  $Y_{2,3}=66k$
- Disadvantage: It may change the distribution of data
- **Hot-deck approach:** draws are made from units with complete data that are 'similar' to the one with missing values (donors).

## Randomly sample from a reasonable distribution

- Very similar, just based on samples from a distribution.
- For example, if gender is missing and you have the information that there are about the sample number of females and males in the population.  $\text{Gender} \sim \text{Ber}(p=0.5)$  or estimate  $p$  from the observed sample
- Disadvantage: distributional assumption may not be reliable (or correct), even the assumption is correct, its representativeness is doubtful

# MULTIPLE IMPUTATION (MI)

Multiple imputation (MI) one of the most attractive methods for general-purpose handling of missing data in multivariate analysis.

1. Impute missing values using an appropriate model that incorporates random variation.
2. Do this  $M$  times producing  $M$  “complete” data sets.
3. Perform the desired analysis on each data set using standard complete-data methods.
4. Average the values of the parameter estimates across the  $M$  samples to produce a single point estimate.
5. Calculate the standard errors by (a) averaging the squared standard errors of the  $M$  estimates (b) calculating the variance of the  $M$  parameter estimates across samples, and (c) combining the two quantities using a simple formula

# LAST OBSERVATION CARRIED FORWARD

- This method is specific to time or longitudinal data problems.
- For each individual, NAs are replaced by the last observed value of that variable. Then, analyze data as if data were fully observed.
- Disadvantage: The covariance structure and distribution change seriously

Cases	1	2	3	4	5	6
1	3.8	3.1	2.0	2.0	2.0	2.0
2	4.1	3.5	2.8	2.4	2.8	3.0
3	2.7	2.4	2.9	3.5	3.5	3.5

# CATEGORICAL VALUES

## Extra category

- This is bad practice
- In many statistical analysis the impact of this strategy depends on how missing values are divided among the real categories, and how the probability of a value being missing depends on other variables;
- very dissimilar classes can be lumped into one group;
- severe bias can arise, in any direction, and when used to stratify for adjustment (or correct for confounding) the completed categorical variable will not do its job properly.

## Better techniques:

- Maximum Likelihood Estimation
- KNN
- Stochastic variants

# CLICKER

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA	50000	\$100000M	\$40000M
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	70000	\$200000M	\$50000M
IBM	1 New Orchard Rd; 10504	400000	\$100000M	null
Microsoft	Albuquerque, New Mexico	130000	\$125000M	\$40000M
Tableau	Seattle, Washington, United States	4000	\$1000M	null
Tamr	64 Church St, Cambridge, MA 02138, USA	30	\$10M	\$1M
Einblick Analytics	null	8	\$0.01M	\$0M
Determined AI	California	15	null	\$0.01M

Calculate the result for    `SELECT SUM(revenue) /SUM(employees) FROM s_tech_companies`  
With listwise deletion, mean and linear regression substitution

For this example, which technique to deal with null values leads to the lowest revenue per employee value:

- a) Listwise deletion
- b) Mean substitution
- c) Regression imputation

# CLICKER

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA	50000	\$100000M	\$40000M
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	70000	\$200000M	\$50000M
IBM	<del>1 New Orchard Rd; 10504</del>	400000	\$100000M	null
Microsoft	Albuquerque, New Mexico	130000	\$125000M	\$40000M
Tableau	Seattle, Washington, United States	4000	\$1000M	null
Tamr	64 Church St, Cambridge, MA 02138, USA	30	\$10M	\$1M
Einblick Analytics	null	8	\$0.01M	\$0M
Determined AI	California	15	null	\$0.01M

Calculate the result for    SELECT SUM(revenue) /SUM(employees)   FROM s\_tech\_companies  
with

- a) Listwise deletion: \$425B / \$250k = \$1.7M per employee
- b) Mean substitution:
- c) Regression imputation

# CLICKER

Name	Address	#Employees	Revenue (M)	Profit (M)
Google	1600 Amphitheatre Parkway, Mountain View, CA	50000	\$100000M	\$40000M
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	70000	\$200000M	\$50000M
IBM	1 New Orchard Rd; 10504	400000	\$100000M	null
Microsoft	Albuquerque, New Mexico	130000	\$125000M	\$40000M
Tableau	Seattle, Washington, United States	4000	\$1000M	null
Tamr	64 Church St, Cambridge, MA 02138, USA	30	\$10M	\$1M
Einblick Analytics	null	8	\$0.01M	\$0M
Determined AI	California	15	\$75000M	\$0.01M

Calculate the result for    `SELECT SUM(revenue) /SUM(employees) FROM s_tech_companies`  
with

- a) Listwise deletion:  $\$425B / \$250k = \$1.7M$  per employee
- b) Mean substitution:  $\$600B / 654k = \$0.92M$  per employee
- c) Regression imputation

# CLICKER

Name	Address	#Employees	Revenue (M)	Profit (M)
Google	1600 Amphitheatre Parkway, Mountain View, CA	50000	\$100000M	\$40000M
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	70000	\$200000M	\$50000M
IBM	1 New Orchard Rd; 10504	400000	\$100000M	null
Microsoft	Albuquerque, New Mexico	130000	\$125000M	\$40000M
Tableau	Seattle, Washington, United States	4000	\$1000M	null
Tamr	64 Church St, Cambridge, MA 02138, USA	30	\$10M	\$1M
Einblick Analytics	null	8	\$0.01M	\$0M
Determined AI	California	15	\$55000M	\$0.01M

Calculate the result for    `SELECT SUM(revenue) /SUM(employees) FROM s_tech_companies`  
with

- a) Listwise deletion:  $\$425B / \$250k = \$1.7M$  per employee
- b) Mean substitution:  $\$600B / 654k = \$0.92M$  per employee
- c) Regression imputation:  $\$580B / 654k = \$0.89M$  per employee

$$\text{Rev} = 55346 + 0.212 * \text{emp}$$

|

# OUTLINE

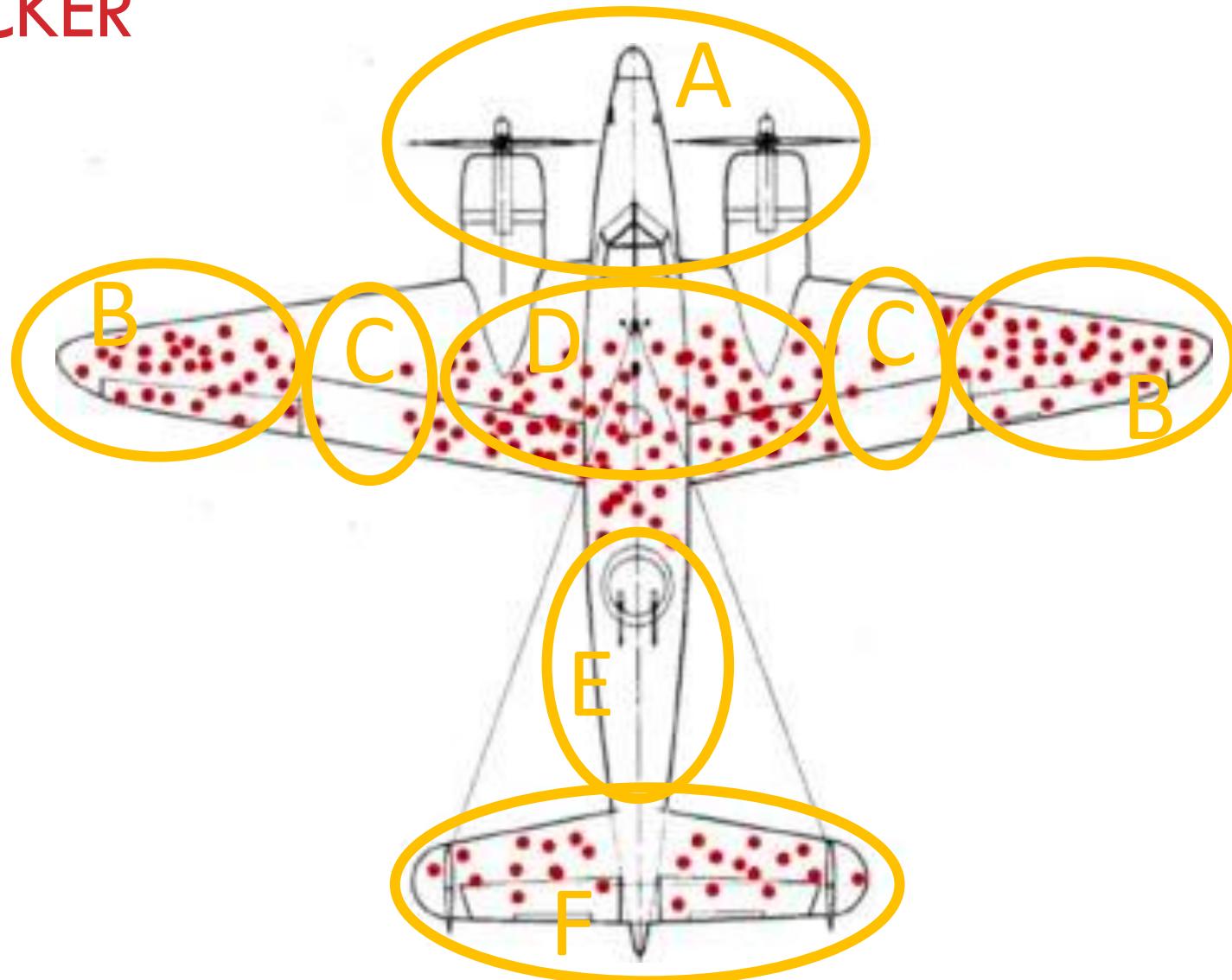
## Data Integration

- **Different schemas** → Schema matching
- **Duplicates** → Entity resolution
- **Contradicting data** → data fusion

## Data Cleaning

- **Missing values** → Value imputation
- **Wrong data** → Outlier detection
- **Missing records** → Species estimation ←

# CLICKER

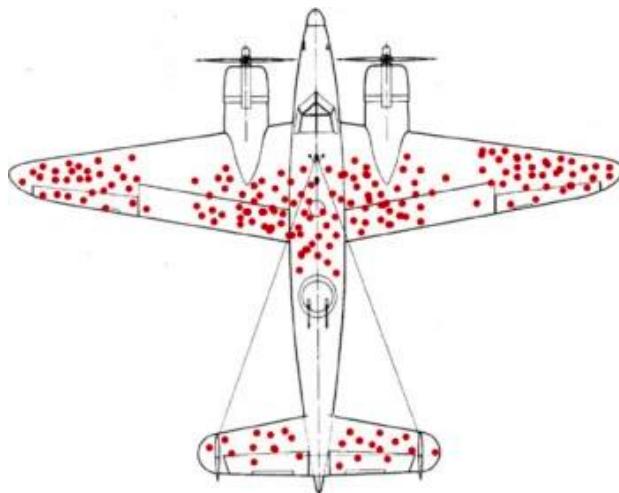


Where would you enforce the plane?

# UNKNOWN UNKNOWNS

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	null
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
International Business Machine	1 New Orchard Rd; 10504	380k	\$-999B	\$12B
Microsoft	Albuquerque, Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	-	\$0.9B	\$1B
Tamr	64 Church St, Cambridge, MA 02138, United States	20	null	\$-Y
Amazon	??	??	??	??
Facebook	??	??	??	??
??	??	??	??	??
??	??	??	??	??

# IF YOU CAN ESTIMATE THEM DEPENDS ON THE SAMPLING SCENARIO



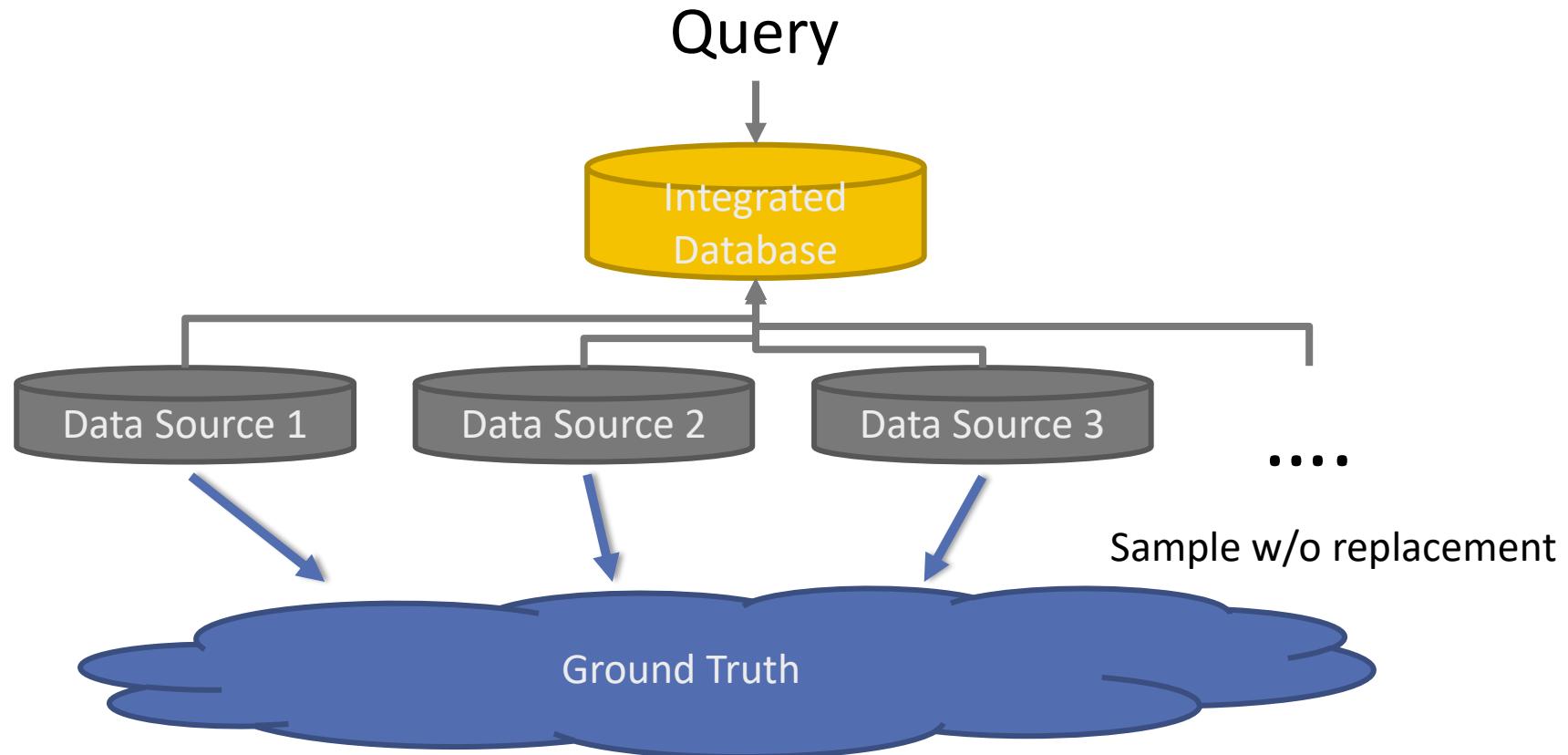
VS

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	null
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
International Business Machine	1 New Orchard Rd; 10504	380k	\$-999B	\$12B
Microsoft	Albuquerque, Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	-	\$0.9B	\$1B
Tamr	64 Church St, Cambridge, MA 02138, United States	20	null	\$-Y
Amazon	??	??	??	??
Facebook	??	??	??	??
??	??	??	??	??
??	??	??	??	??

# THE IMPACT OF THE UNKNOWN UNKNOWNS ON QUERY RESULTS

*How many people work  
in the US IT industry*

```
SELECT SUM(employees)  
FROM us_tech_companies
```



Assumption: Enough data sources , Data sources are (semi-) independent

# Sampling - Statistic

 $\sum$ 

	Name	Address	#Employees	Revenue	Profit	Frequency
Google	Google	Address I	60k	\$89B	\$10B	5
Apple	Apple	Address II	66k	\$215B	\$45B	4
IBM	IBM	Address II	380k	\$80B	\$12B	4
Microsoft	Microsoft	Address	120k	\$85B	\$85B	5
Tableau	Tableau	Address	3.2k	\$500	\$8M	2
tamr	Tamr	Address	20	\$-X	\$-Y	1

## Fingerprint (i.e., f-statistic):

$f_1: 1$  



**Singletons** (items which were  
exactly observed once)

$f_2: 1$  

$f_4: 2$   

$f_5: 2$   

# MANY WAYS TO ESTIMATE THE NUMBER OF MISSING ITEMS

- Good-Turing Estimate / Chao84
- Chao92
- Pattern Maximum Likelihood
- Linear programming-based solutions (see Valiant brothers)
- ...

# ESTIMATING THE NUMBER OF DISTINCT BUTTERFLY SPECIES



17500 species known in the world

# GOOD-TURING / CHAO84 ESTIMATE

$$\hat{N} = \frac{c}{\left(1 - \frac{f_1/n}{n}\right)}$$

Unique Items

Missing mass

Number of Unknown Unknowns:

Number of Unknown Unknowns:

$$M = \hat{N} - C$$

Note, we usually prefer **Chao92**: A. Chao and S. Lee, “**Estimating the Number of Classes via Sample Coverage**,” *Journal of the American Statistical Association*, vol. 87, no. 417, pp. 210–217, 1992  
over **Chao84**: A. Chao, “**Nonparametric Estimation of the Number of Classes in a Population**,” *SJS*, vol. 11, no. 4, 1984

# A NAÏVE ESTIMATOR FOR THE IMPACT OF THE UNKNOWN UNKNOWNS



```
SELECT SUM(employees)  
FROM us_tech_companies
```



$$\sum employees, \Delta(employees, fingerprint)$$

$$\Delta_{Naive} = M \cdot \emptyset$$

Estimate of  
Unknown  
Unknowns  
Count

Average  
Value of  
Knowns  
(aka mean substitution)

# A NAÏVE ESTIMATOR FOR THE IMPACT OF THE UNKNOWN UNKNOWNS

Number of unique records  
i.e.,  $\text{count}(\ast)$

$\Delta_{Naive}$  =

$$\frac{c}{\left(1 - f_1/n\right)}$$

Estimated number of missing records

Value sum over all unique items

$$\sum_{\{c\}} v$$

$c$

Mean value

# EXAMPLE

MIT Fan DB

FanID	Name	Address	Email	FanOf	Genre
2	Tim	46 Pumpkin St	timk	Nickelback, Creed, Limp Bizkit	Terrible
3	Matt	Vassar Str	Mattp	Nickelback	Terrible

MIT CSAIL DB

ID	Name
10	Tim
14	Matt

MIT Department DB

ID	Name
10	Tim
14	Joana

FanID	Name	Address	Email	FanOf	Genre	Frequency
2	Tim	46 Pumpkin St	timk	Nickelback, Creed, Limp Bizkit	Terrible	3
3	Matt	Vassar Str	Mattp	Nickelback	Terrible	2
4	Joana					1

$$\#Missing = \frac{c}{(1 - f_1/n)} = \frac{3}{(1 - 1/6)} = 3.6$$

Note estimator shouldn't be used if sample coverage is below 80% ( $1 - f_1/n$ ) and such a small number of data sources (independent samples)

# EXAMPLE

$$\#Missing = \frac{c}{(1 - f_1/n)} = \frac{3}{(1 - 1/6)} = 3.6$$

FanID	Name	Address	Email	FanOf	Genre	Frequency
2	Tim	46 Pumpkin St	timk	Nickelback, Creed, Limp Bizkit	Terrible	3
3	Matt	Vassar Str	Mattp	Nickelback	Terrible	2
4	Joana			Cold Play	OK	1

# EXAMPLE

$$\#Missing = \frac{c}{\left(1 - f_1/n\right)} = \frac{3}{\left(1 - 1/6\right)} = 3.6$$

FanID	Name	Address	Email	FanOf	Genre	Frequency
2	Tim	46 Pumpkin St	timk	Nickelback, Creed, Limp Bizkit	Terrible	3
3	Matt	Vassar Str	Mattp	Nickelback	Terrible	2
4	Joana			Cold Play	OK	1
...	...	..	...	...	...	...
5	Sam	Christmas St	Samm	Celine Dion	As cheesy as deep-fried camembert <sup>1</sup>	



vevo

<sup>1</sup> <https://www.telegraph.co.uk/music/concerts/cheesy-deep-fried-camembert-celeine-dion-o2-arena-review/>

# OUTLINE

## Data Integration

- **Different schemas** → Schema matching
- **Duplicates** → Entity resolution
- **Contradicting data** → data fusion

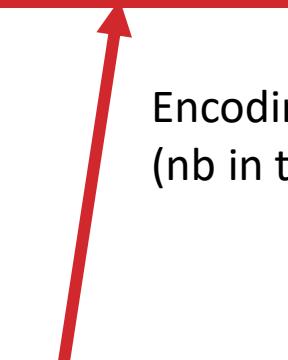
## Data Cleaning

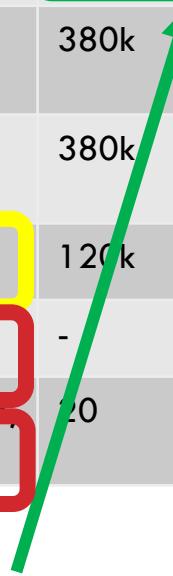
- **Missing values** → Value imputation
- **Wrong data** → Outlier detection ←
- **Missing records** → Species estimation

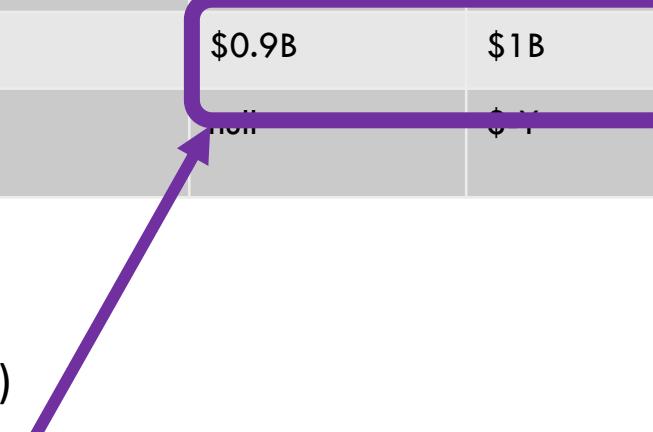
# WRONG DATA: RULE-BASED APPROACHES

Name	Address	#Employees	Revenue	Profit
Google	1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA	60k	\$89B	null
Apple	1 Infinite Loop; Cupertino, CA 95014, USA	66	\$215B	\$45B
IBM	1 New Orchard Rd; New York 10504, USA	380k	\$80B	\$12B
International Business Machine	10504; 1 New Orchard Rd	380k	-\$999B	\$12B
Microsoft	Albuquerque, Mexico	120k	\$85B	\$85B
Tableau	Seattle, Washington, United States	-	\$0.9B	\$1B
Tamr	United States	20	null	\$11B

 Outdated data / wrong data

 Spelling mistakes / abbreviations

 Encoding Error  
(nb in thousands)

 Rule Violations

# TWO COMPONENTS

## 1. Detection

## 2. Repair

- Detection techniques can be used for repair
- Missing value techniques

# COMMON TECHNIQUES

- **Consistency Violations**
- **Outliers**
- **Manual Validation / (Crowd-)Sourcing**

# ERROR DETECTION

FD: [country]  $\rightarrow$  [capital]

CFD: [country = China]  $\rightarrow$  [capital = Beijing]

**emp**

	name	country	capital	city	salary	tax
r1	Nan	China	Beijing	Beijing	50000	1000
r2	Yin	China	Shanghai	Hongkong	40000	1200
r3	Si	Netherlands	Den Hagg	Utrecht	60000	1400
r4	Lei	Netherlands	Amsterdam	Amsterdam	35000	800

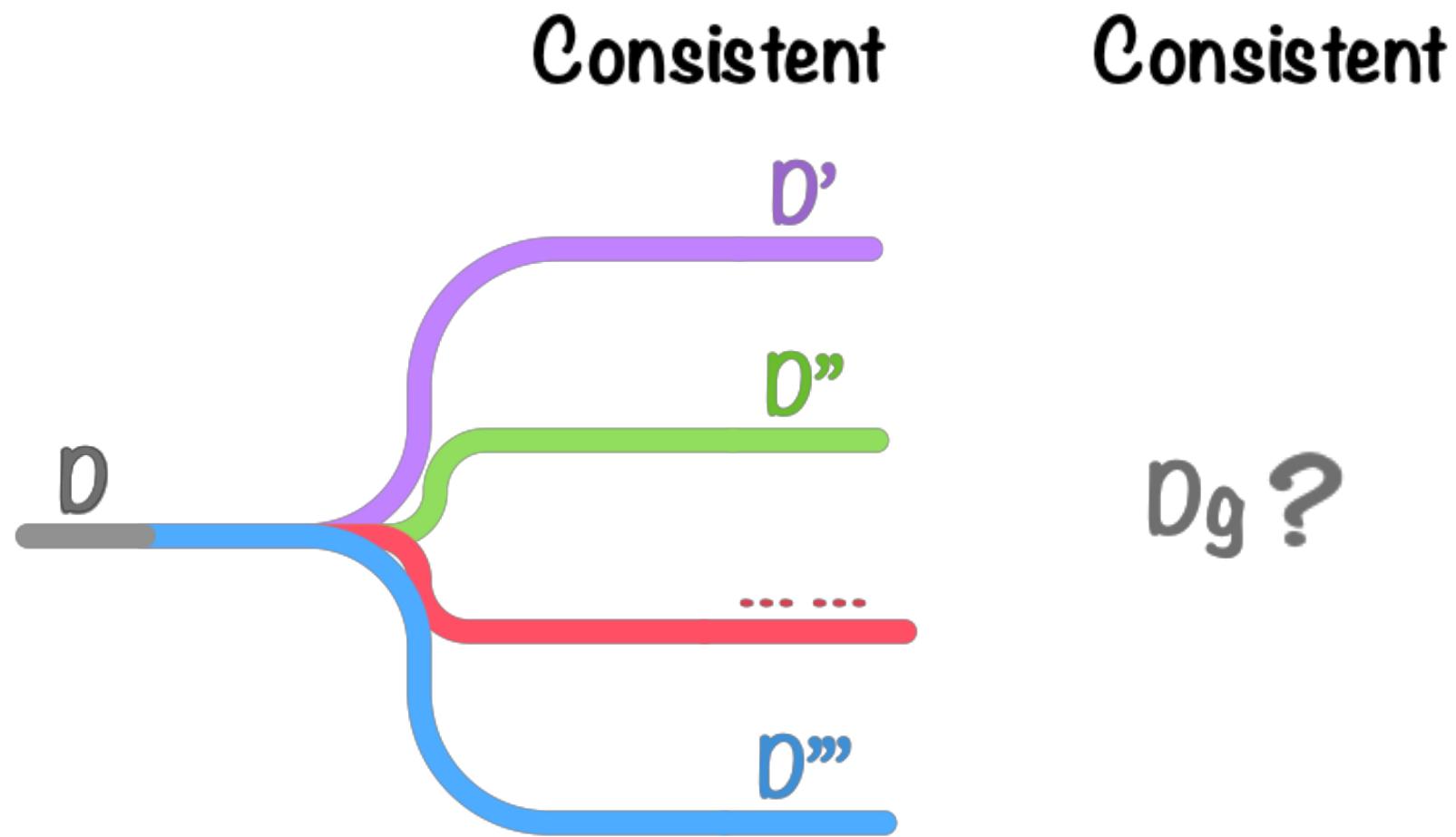
**cap**

	country	capital
s1	China	Beijing
s2	Canada	Ottawa
s3	...	...

CD:  $\exists t1, t2 (t1.salary > t2.salary \text{ and } t1.tax < t2.tax)$

MD:  $(\text{emp}[\text{country}] = \text{cap}[\text{country}]) \rightarrow (\text{emp}[\text{capital}] \leftrightarrow \text{cap}[\text{capital}])$

# COMPUTING A CONSISTENT DATABASE



find a  $D'$  such that  $\text{dist}(D, D')$  is minimum

# COMPUTING A CONSISTENT DATABASE

FD1: [nationality] -> [capital]

FD2: [areacode] -> [capital]

	name	nationality	capital	areacode	bornAt	salary	tax
r1	Nan	China	Beijing	10	Shenyang	50000	1000
r2	Yan	China	Shanghai	10	Hangzhou	40000	900
r3	Si	China	Beijing	10	Changsha	60000	1400
r4	Miura	China	Tokyo	3	Kyoto	35000	800

Equivalence  
class

Vertex  
cover

SAT  
solver

....

# CONFIDENCE VALUES INTERACTION

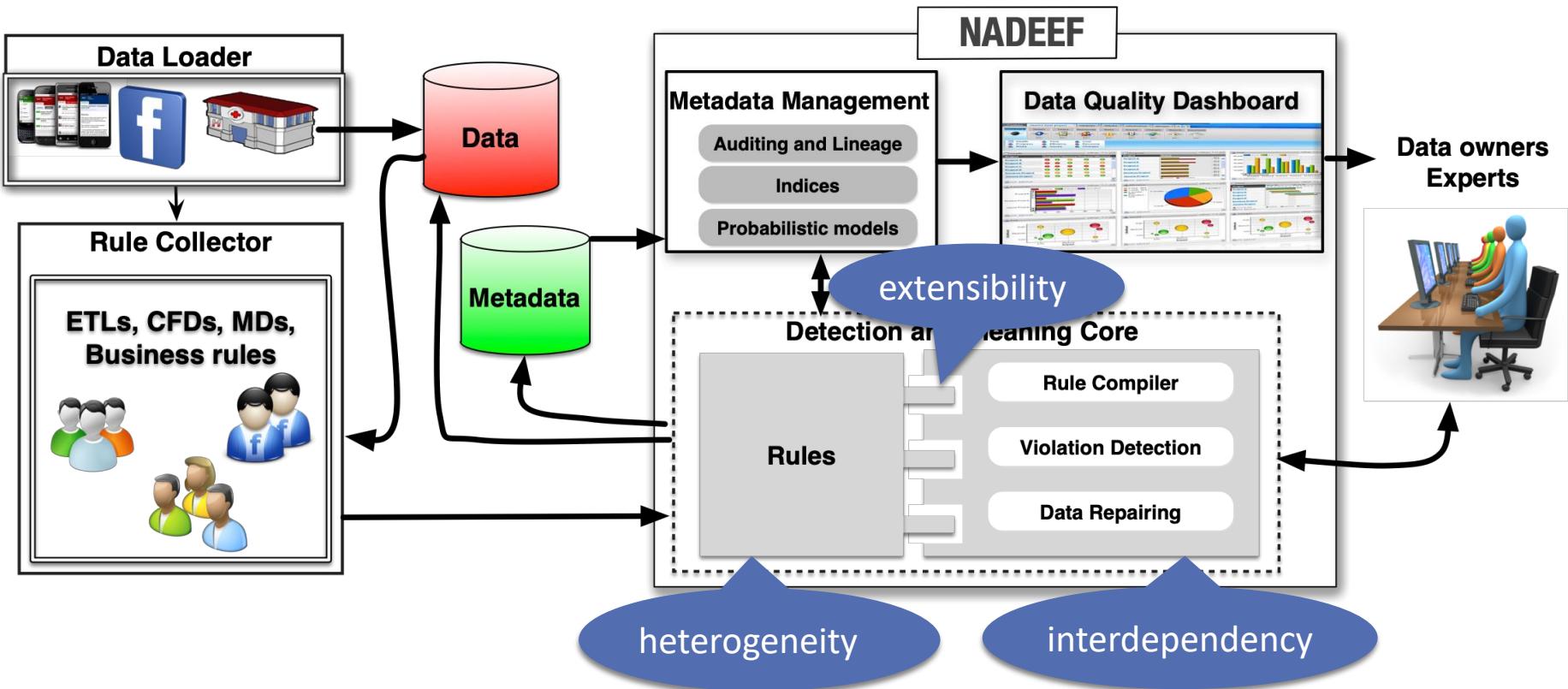


FD: [nationality] -> [capital]

MD: ((nationality, country) -> (capital, capital))

	name	nationality	capital	bornAt
r1	Nan (0.9)	China (1.0)	Beijing (1.0)	Shenyang (0.9)
r2	Yan (0.8)	China (1.0)	Beijing (0.5)	Hangzhou (0.9)
r3	Si (0.9)	Canada (1.0)	Ottawa (1.0)	Changsha (0.8)
r4	Miura (0.9)	Canada (0.9)	Vancouver (0.5)	Kyoto (1.0)

	country	capital
s1	China (1.0)	Beijing (1.0)
s2	Canada (1.0)	Ottawa (1.0)
s3	Japan (1.0)	Tokyo (1.0)



File Refiner About

## Rule Editor

**Detect**

Repair

Block

Iterator

```
8  @Override
9  public Collection<Violation> detect(TuplePair tuplePair) {
10     List<Violation> result = new ArrayList<>();
11     Tuple left = tuplePair.getLeft();
12     Tuple right = tuplePair.getRight();
13
14     if (
15         Metrics.getEqual(
16             left.get("name"), right.get("name")) == 1.0 &&
17         Metrics.getLevenshtein(
18             left.get("address"), right.get("address")) > 0.8 &&
19         Metrics.getEqual(
20             left.get("gender"), right.get("gender")) == 1.0
21     ) {
22         Violation v = new Violation(getRuleName());
23         v.addTuple(left);
24         v.addTuple(right);
25         result.add(v);
26     }
27     return result;
28 }
29
30
```

Close Save changes

# ANOMALY/OUTLIER DETECTION

## What are anomalies/outliers?

- The set of data points that are considerably different than the remainder of the data

## Variants of Anomaly/Outlier Detection Problems

- Given a database D, find all the data points  $\mathbf{x} \in D$  with anomaly scores greater than some threshold t
- Given a database D, find all the data points  $\mathbf{x} \in D$  having the top-n largest anomaly scores  $f(\mathbf{x})$
- Given a database D, containing mostly normal (but unlabeled) data points, and a test point  $\mathbf{x}$ , compute the anomaly score of  $\mathbf{x}$  with respect to D

## Applications:

- Credit card fraud detection, telecommunication fraud detection, network intrusion detection, fault detection

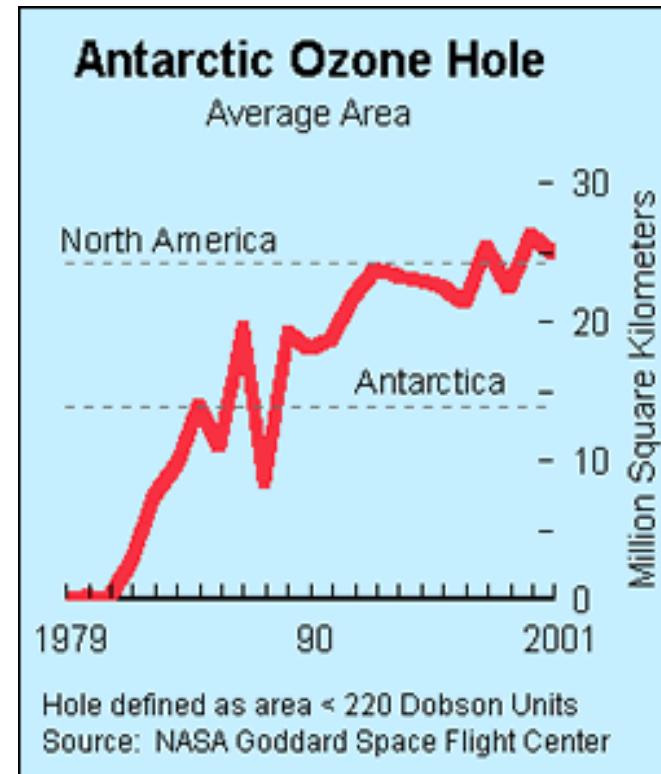
# IMPORTANCE OF ANOMALY DETECTION

## Ozone Depletion History

In 1985 three researchers (Farman, Gardinar and Shanklin) were puzzled by data gathered by the British Antarctic Survey showing that ozone levels for Antarctica had dropped 10% below normal levels

Why did the Nimbus 7 satellite, which had instruments aboard for recording ozone levels, not record similarly low ozone concentrations?

The ozone concentrations recorded by the satellite were so low they were being treated as outliers by a computer program and discarded!



Sources:

<http://exploringdata.cqu.edu.au/ozone.html>  
<http://www.epa.gov/ozone/science/hole/size.html>

# ANOMALY DETECTION

## Challenges

- How many outliers are there in the data?
- Method is unsupervised
  - Validation can be quite challenging (just like for clustering)
- Finding needle in a haystack

## Working assumption:

- There are considerably more “normal” observations than “abnormal” observations (outliers/anomalies) in the data

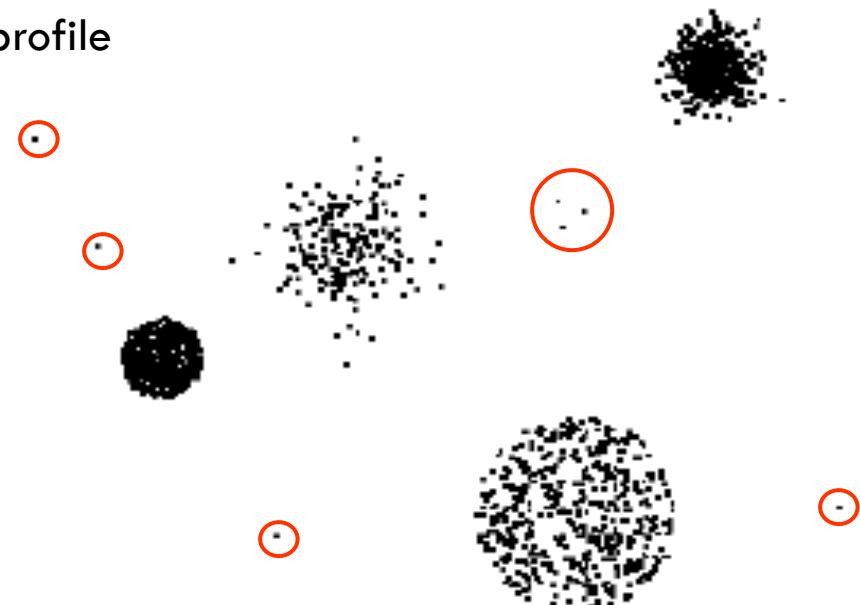
# ANOMALY DETECTION SCHEMES

## General Steps

- Build a profile of the “normal” behavior
  - Profile can be patterns or summary statistics for the overall population
- Use the “normal” profile to detect anomalies
  - Anomalies are observations whose characteristics differ significantly from the normal profile

## Types of anomaly detection schemes

- Graphical
- Model-based
- Distance-based
- Clustering-based

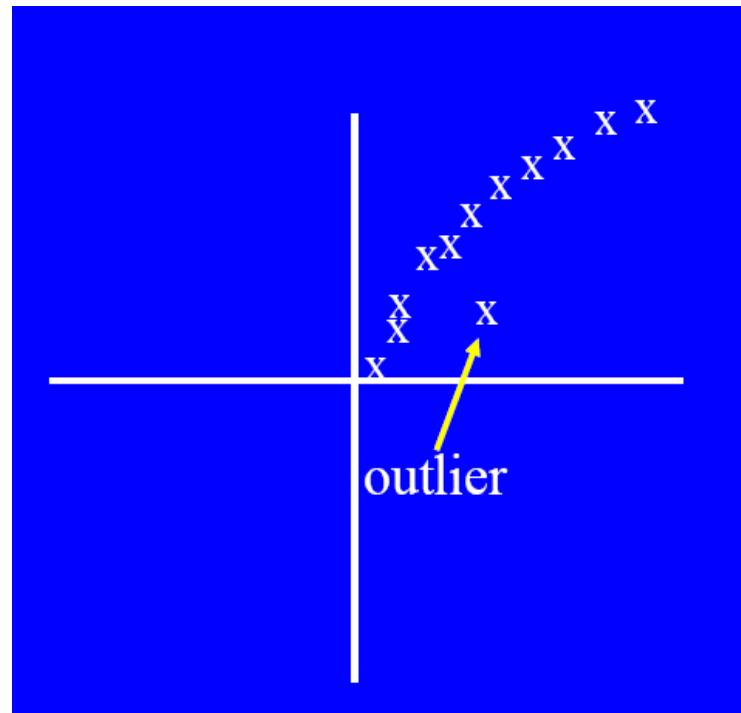
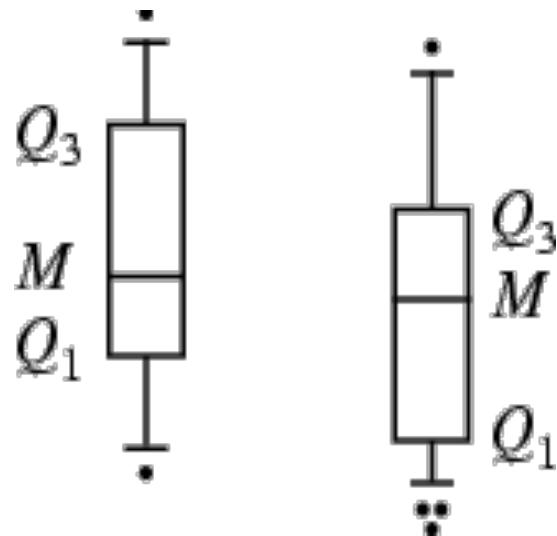


# GRAPHICAL APPROACHES

Boxplot (1-D), Scatter plot (2-D), Spin plot (3-D)

## Limitations

- Time consuming
- Subjective

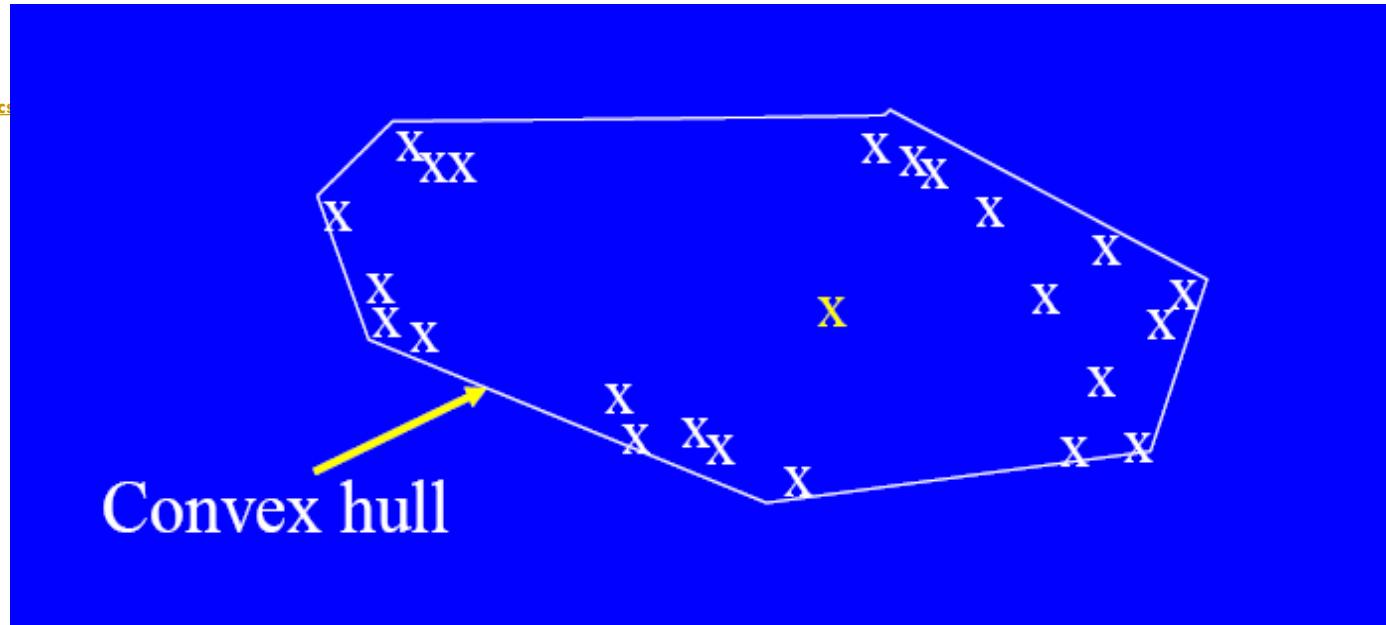


# CONVEX HULL METHOD

Extreme points are assumed to be outliers

Use convex hull method to detect extreme values

[http://cgm.csail.mit.edu/6.S091/convex\\_hull.html](http://cgm.csail.mit.edu/6.S091/convex_hull.html)

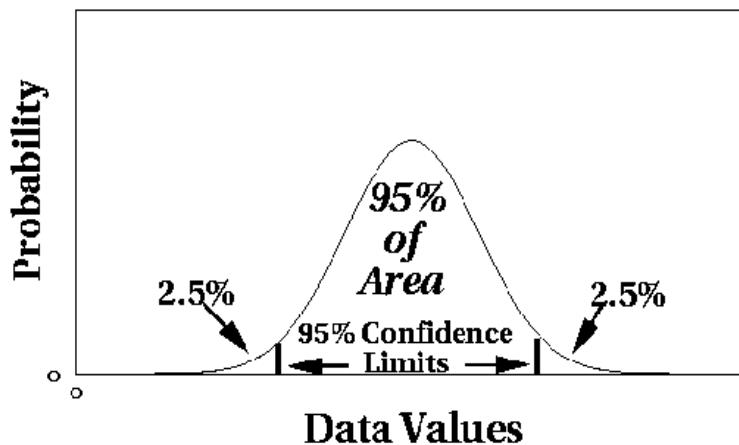


# STATISTICAL APPROACHES---MODEL-BASED

**Assume a parametric model describing the distribution of the data (e.g., normal distribution)**

**Apply a statistical test that depends on**

- Data distribution
- Parameter of distribution (e.g., mean, variance)
- Number of expected outliers (confidence limit)



# GRUBBS' TEST

**Detect outliers in univariate data**

**Assume data comes from normal distribution**

**Detects one outlier at a time, remove the outlier, and repeat**

- $H_0$ : There is no outlier in data
- $H_A$ : There is at least one outlier

**Grubbs' test statistic:**

**Reject  $H_0$  if:**

$$G = \frac{\max|X - \bar{X}|}{S}$$

$$G > \frac{(N-1)}{\sqrt{N}} \sqrt{\frac{t^2_{(\alpha/2, N-2)}}{N-2 + t^2_{(\alpha/2, N-2)}}}$$

# STATISTICAL-BASED – LIKELIHOOD APPROACH

**Assume the data set D contains samples from a mixture of two probability distributions:**

- M (majority distribution)
- A (anomalous distribution)

**General Approach:**

- Initially, assume all the data points belong to M
- Let  $L_t(D)$  be the log likelihood of D at time t
- For each point  $x_t$  that belongs to M, move it to A
  - Let  $L_{t+1}(D)$  be the new log likelihood.
  - Compute the difference,  $\Delta = L_t(D) - L_{t+1}(D)$
  - If  $\Delta > c$  (some threshold), then  $x_t$  is declared as an anomaly and moved permanently from M to A

# LIMITATIONS OF STATISTICAL APPROACHES

**Most of the tests are for a single attribute**

**In many cases, data distribution/model may not be known**

**For high dimensional data, it may be difficult to estimate the true distribution**

# DISTANCE-BASED APPROACHES

**Data is represented as a vector of features**

**Three major approaches**

- Nearest-neighbor based
- Density based
- Clustering based

# NEAREST-NEIGHBOR BASED APPROACH

## Approach:

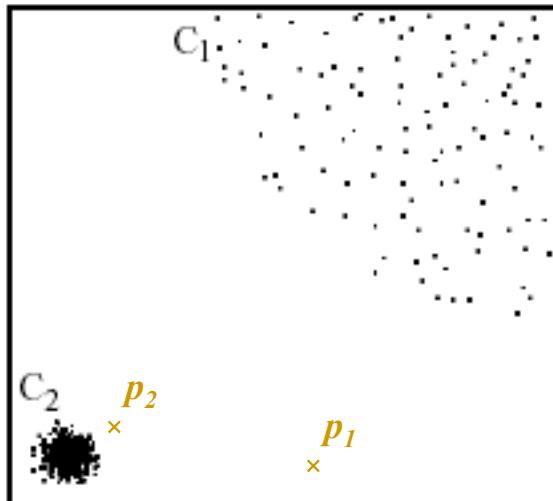
- Compute the distance between every pair of data points
- There are various ways to define outliers:
  - Data points for which there are fewer than  $p$  neighboring points within a distance  $D$
  - The top  $n$  data points whose distance to the  $k$ th nearest neighbor is greatest
  - The top  $n$  data points whose average distance to the  $k$  nearest neighbors is greatest

# DENSITY-BASED: LOF APPROACH

For each point, compute the density of its local neighborhood; e.g. use DBSCAN's approach

Compute local outlier factor (LOF) of a sample  $p$  as the average of the ratios of the density of sample  $p$  and the density of its nearest neighbors

Outliers are points with largest LOF value



In the NN approach,  $p_2$  is not considered as outlier, while LOF approach find both  $p_1$  and  $p_2$  as outliers

Alternative approach: directly use density function; e.g. DENCLUE's density function

# CLUSTERING-BASED

**Idea:** Use a clustering algorithm that has some notion of outliers!

**Problem** what parameters should I choose for the algorithm; e.g. DBSCAN?

**Rule of Thumb:** Less than  $x\%$  of the data should be outliers (with  $x$  typically chosen between 0.1 and 10);  $x$  might be determined with other methods; e.g. statistical tests.

	FN	LN	St	city	CC	country	tel	gd
$t_1:$	David	Jordan	12 Holywell Street	Oxford	44	UK	66700543	Male
$t_2:$	Paul	Simon	5 Ratcliffe Terrace	Oxford	44	UK	44944631	Male

(a)  $D_1$ : An instance of schema bank

	FN	LN	str	city	CC	country	phn	when	where
$r_1:$	David	Jordan	12 Holywell Street	Oxford	44	UK	66700543	1pm 6/05/2012	Netherlands
$r_2:$	Paul	Simon	5 Ratcliffe Terrace	Oxford	44	UK	44944631	11am 2/12/2011	Netherlands
$r_3:$	David	Jordan	12 Holywell Street	Oxford	44	Netherlands	66700541	6am 6/05/2012	US
$r_4:$	Peter	Austin	7 Market Street	Amsterdam	31	UK	55384922	9am 6/02/2012	Netherlands

(b) Database  $D_2$ : An instance of schema tran

r1: (on table tran) if a customer's CC is 31, but his/her country is neither Netherlands nor Holland, update the country to Netherlands;

```
Class Rule1 { /* for  $\varphi_1$  */
set<cell> vio(Tuple s1) {
    if ( $s_1[CC]=31 \wedge (s_1[country] \neq \text{Netherlands} \vee s_1[country] \neq \text{Holland})$ )
        return {  $s_1[CC, country]$ ; }
    return {};
}
set<Expression> fix (set<cell> V) {
    return {  $V.s_1[country] \leftarrow \text{Netherlands}$ ; }
} } /* end of class definition */
```

r2: (on tables bank and tran) if the same person from different tables has different phones, the phone number from table bank is more reliable;

```
Class Rule2 { /* for  $\varphi_2$  */
set<cell> vio (Tuple s1, Tuple s2) { /*  $s_1$  in bank,  $s_2$  in tran */
    if ( $s_1[LN, St, city]=s_2[LN, str, city] \wedge s_1[FN] \approx s_2[FN] \wedge s_1[tel] \neq s_2[phn]$ )
        return {  $s_1[FN, LN, St, city, tel], s_2[FN, LN, str, city, phn]$ ; }
    return {};
}
set<Expression> fix (set<cell> V) {
    return {  $V.s_2[phn] \leftarrow V.s_1[tel]$ ; }
} } /* end of class definition */
```

r3: (on table tran) a country code (CC) uniquely determines a country;

```
Class Rule3 { /* for  $\varphi_3$  */
set<cell> vio (Tuple s1, Tuple t2) { /*  $s_1, s_2$  in table tran */
    if ( $s_1[CC] = s_2[CC] \wedge s_1[country] \neq s_2[country]$ )
        return {  $s_1[CC, country], s_2[CC, country]$ ; }
    return {};
}
set<Expression> fix (set<cell> V) {
    set<Expression> fixes;
    fixes.insert( $V.s_1[country] \leftarrow V.s_2[country]$ );
    fixes.insert( $V.s_2[country] \leftarrow V.s_1[country]$ );
    return fixes;
} } /* end of class definition */
```

r4: (on table tran) if two purchases of the same person happened in the Netherlands and the US (East Coast) within 1 hour (assuming 6 hours' time difference between these two countries), these two purchases are either a fraud or were erroneously recorded.

```
Class Rule4 { /* for  $\varphi_4$  */
set<cell> vio (Tuple s1, Tuple s2) { /*  $s_1, s_2$  in table tran */
    if ( $s_1[LN, city, CC, tel] = s_2[LN, city, CC, tel]$ 
         $\wedge s_1[where] = \text{Netherlands} \wedge s_2[where] = \text{US} \wedge s_1[FN] \approx s_2[FN]$ 
         $\wedge (s_1[when] - s_2[when] \geq 5) \wedge (s_1[when] - s_2[when] \leq 7)$ )
        return {  $s_1[FN, LN, city, CC, tel, when, where],$ 
                 $s_2[FN, LN, city, CC, tel, when, where]$ ; }
    return {};
} } /* end of class definition */
```

Figure 3: Sample rules