Entity Resolution: Behind the Scenes

Arnaud Esteve - Devoxx Paris - 2025

QUICK WORD OF INTRO

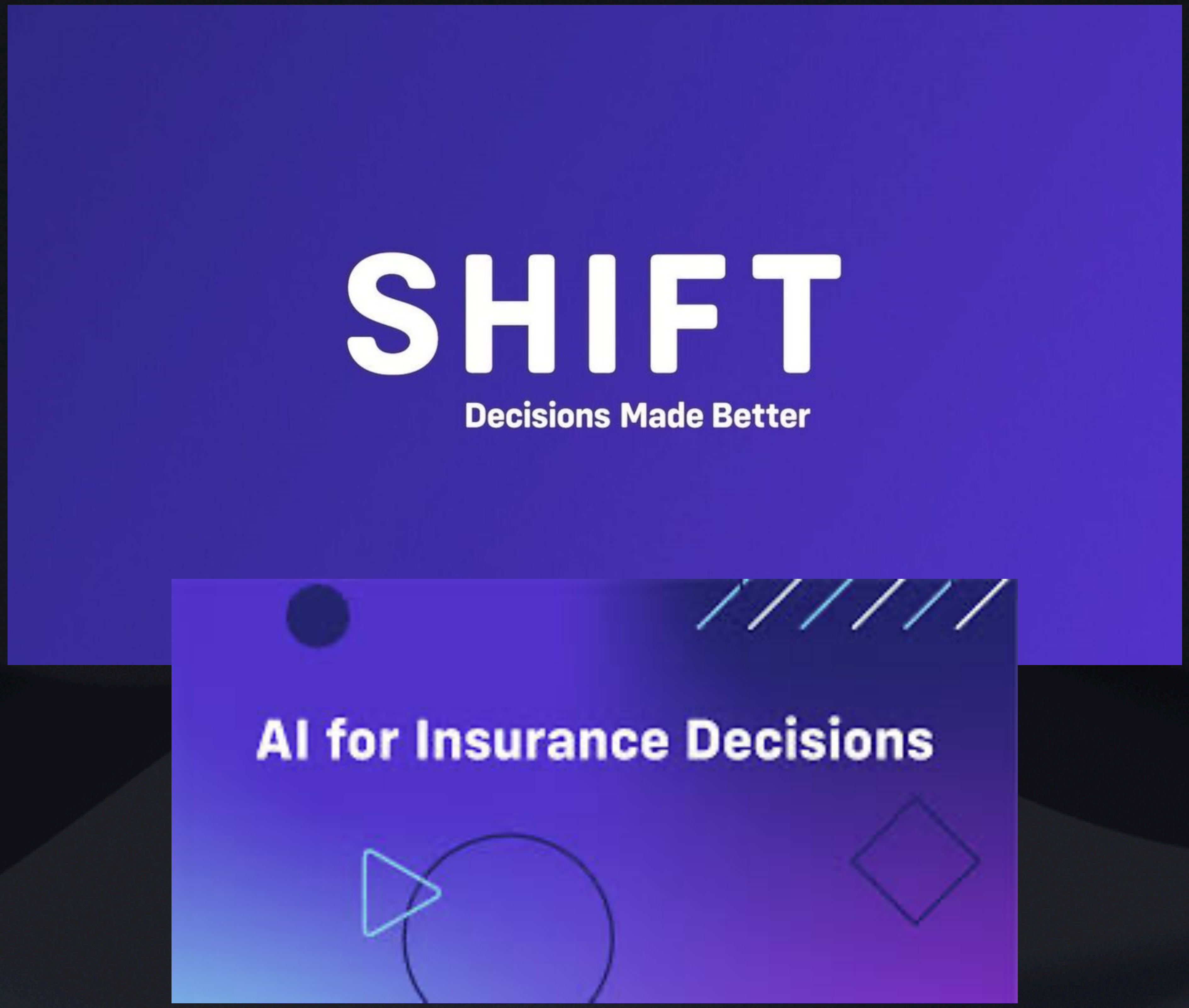
Arnaul~~t~~ Estève

Arnau~~t~~ Esteve~~s~~

Arnaul~~d~~ Esteve~~z~~

Arnaud Esteve

Data Scientist



A COMMON TASK

Matching entities that look different but are the same

Consolidation

Record Linkage

Reference reconciliation

Householding

Identity Resolution

Data Harmonisation

Deduplication

Object reconciliation

Data Unification

Fuzzy Matching

Identity Reconciliation

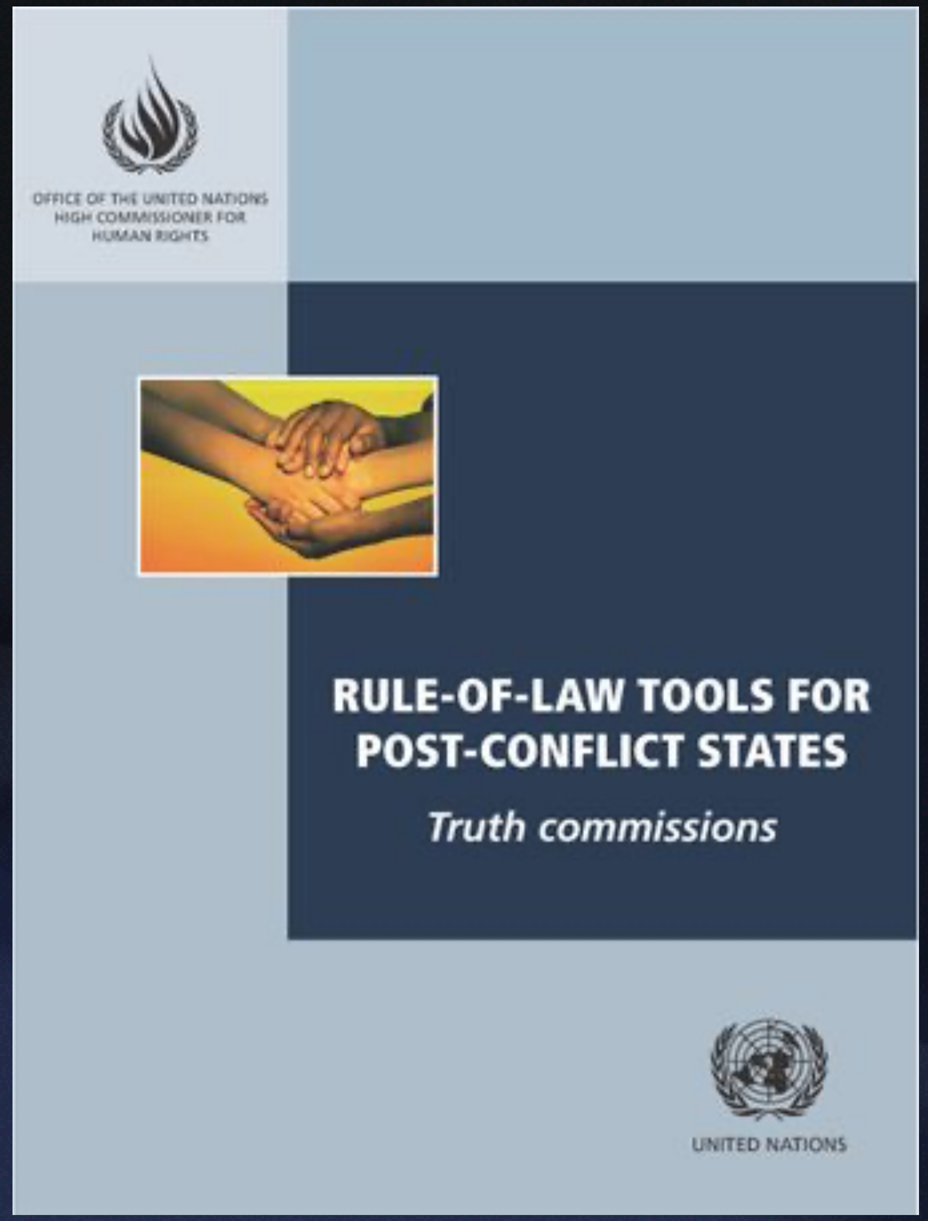
Data Matching

Cross Linking

Entity Matching

Master Data Management

In real life



Credit: Geoff Thale and Adriana Beltran

Did we solve it well, though?

"We relied on email"

"We did it best effort"

`If … and if … and if … or if … and if … or if …`

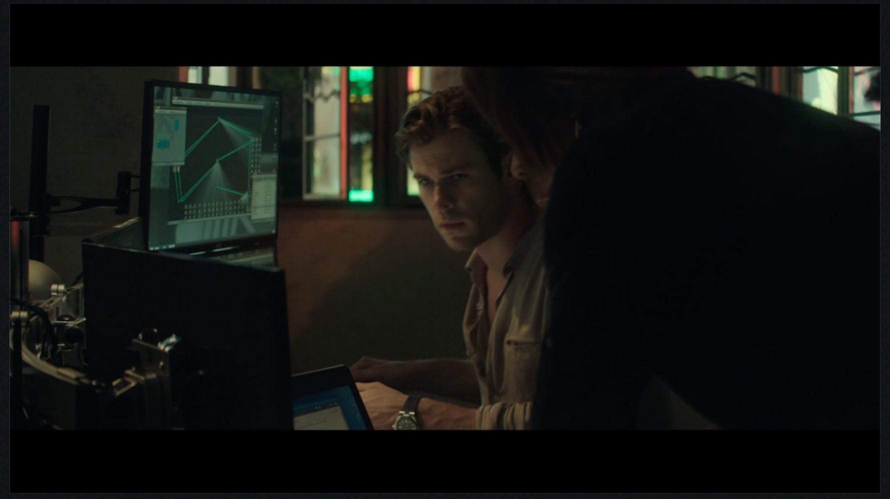
Can we do it differently?

A Private Investigator Story

The "human way"

vs.

The "programmatic way"



Highlighting Info	*Preparing records* →	Cleansing
Matching dossiers	*Matching records* →	Linking
Pulling threads	*Grouping records* →	Clustering
Solving identity	*Merging records* →	Canonicalisation
Case Review	*Checking records* →	Validation

Cleansing: Dusting for clues

Why is data messy?

- Missing data
- Format inconsistencies and variations: `1923-02-01` vs. `02/01/23`
- Alternatives: `Katherine` ⟷ `Kate` . `Junior` ⟷ `Jr.`
- Typos & non-uniform characters
- OCR / Digitalisation mistakes
- Phonetic Mistakes

Cleaning Data

A few techniques

It's not just about cleaning, it's about attribute alignment!

- **Removing noise:** `<HTML />`, …
- **Normalisation:** latin, UPPER, ~~diàcrïtîcs~~.
- **Parsing:** extract more information
- **Encoding:** phonetic, geo
- **Detecting:** outliers / special (eg. `1970-01-01`)
- **Variations:** William <=> Bill

But there's more:
Make data computable

Encoding

Phonetic example

!!Add NAMEPRISM!!

https://github.com/jamesturk/jellyfish

```
from jellyfish import match_rating_codex, metaphone, nysiis, soundex
name = "Stephen"
soundex_code = soundex(name)
metaphone_code = metaphone(name)
nysiis_code = nysiis(name)
match_rating_codex_code = match_rating_codex(name)
```

Alternate spellings

Original	Stephen	Steven
Soundex	S315	S315
Metaphone	STFN	STFN
NYSIIS	STAFAN	STAFAN
Match Rating Codex	STPHN	STVN

Original	Mohamed	Muhammad
Soundex	M530	M530
Metaphone	MHMT	MHMT
NYSIIS	MAHANAD	MAHANAD
Match Rating Codex	MHMD	MHMD

Alternate spellings + typo

Original	Rashami	Rashmyi
Soundex	R250	R250
Metaphone	RXM	RXMY
NYSIIS	RASAN	RASNY
Match Rating Codex	RSHM	RSHMY

Original	Lucía	Lizía
Soundex	L200	L200
Metaphone	LS	LS
NYSIIS	LACÍ	LASÍ
Match Rating Codex	LCÍ	LZÍ

Parsing & Encoding

Geo-coding + Geo-Hashing Example

```
GET https://nominatim.openstreetmap.org/search?
q=palais%20congrès%20maillot&format=json&addressdetails=1
```

```
{
    "place_id": 89127893,
    "licence": "Data © OpenStreetMap contributors, ODbL 1.0. http://osm.org/copyright",
    "osm_type": "node",
    "osm_id": 3969536957,
    "lat": "48.8784493",
    "lon": "2.2837635",
    …
    "address": {
        "railway": "Palais des Congrès",
        "road": "Place de la Porte Maillot",
        "city_block": "Quartier des Ternes",
        "suburb": "17th Arrondissement",
        "city_district": "Paris",
        "city": "Paris",
        "ISO3166-2-lvl6": "FR-75C",
        "region": "Metropolitan France",
        "postcode": "75017",
        "country": "France",
        "country_code": "fr"
    },
    …
}
```

encoded

parsed & validated

hashed

```
geohash.encode(latitude=48.8784493, longitude=2.2837635)
```

u09w5fncxe37

```
GET https://nominatim.openstreetmap.org/search?
q=82%20Boulevard%20Pereire&format=json&addressdetails=1
```

```
{
    "place_id":89276492,
    "licence":"Data © OpenStreetMap contributors, ODbL 1.0. http://osm.org/copyright",
    "osm_type":"node",
    "osm_id":8668961639,
    "lat":"48.8871762",
    "lon":"2.3042596",
    …
    "address": {
        "amenity":"Etoile - Wagram",
        "road":"Boulevard Pereire",
        "city_block":"Quartier de la Plaine-de-Monceau",
        "suburb":"17th Arrondissement",
        "city_district":"Paris",
        "city":"Paris",
        "ISO3166-2-lvl6":"FR-75C",
        "region":"Metropolitan France",
        "postcode":"75017",
        "country":"France",
        "country_code":"fr"
    },
    …
}
```

encoded

parsed & validated

```
geohash.encode(latitude=48.8871762, longitude=2.3042596)
```

u09wh7tumnhm

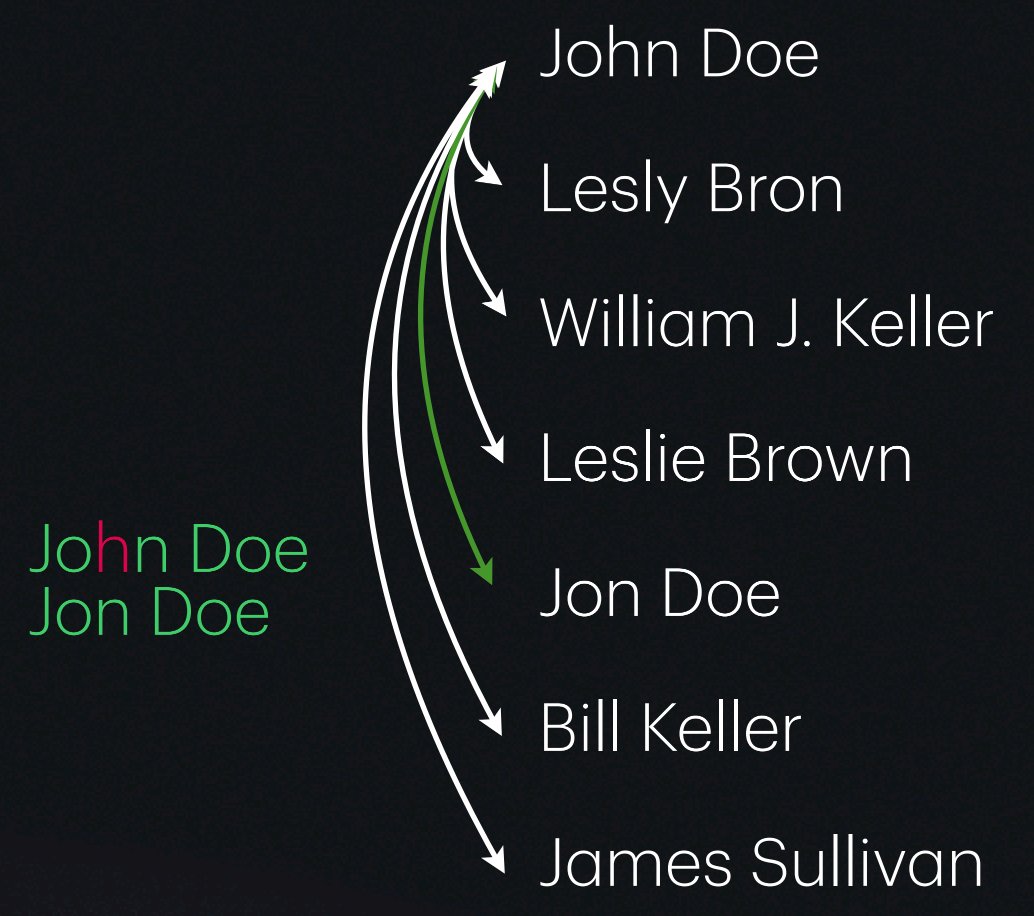
“palais des Congrès maillot”	u09w5fncxe37
“82 Bd Pereire”	u09wh7tumnhm
“Tour Eiffel, Paris”	u09tuny9c3wb
“Gare des Bénédictins, Limoges”	u00uub4ztwv4


Linking:
The detective's Hunch

Human approach

Comparing pairs

- One record to another

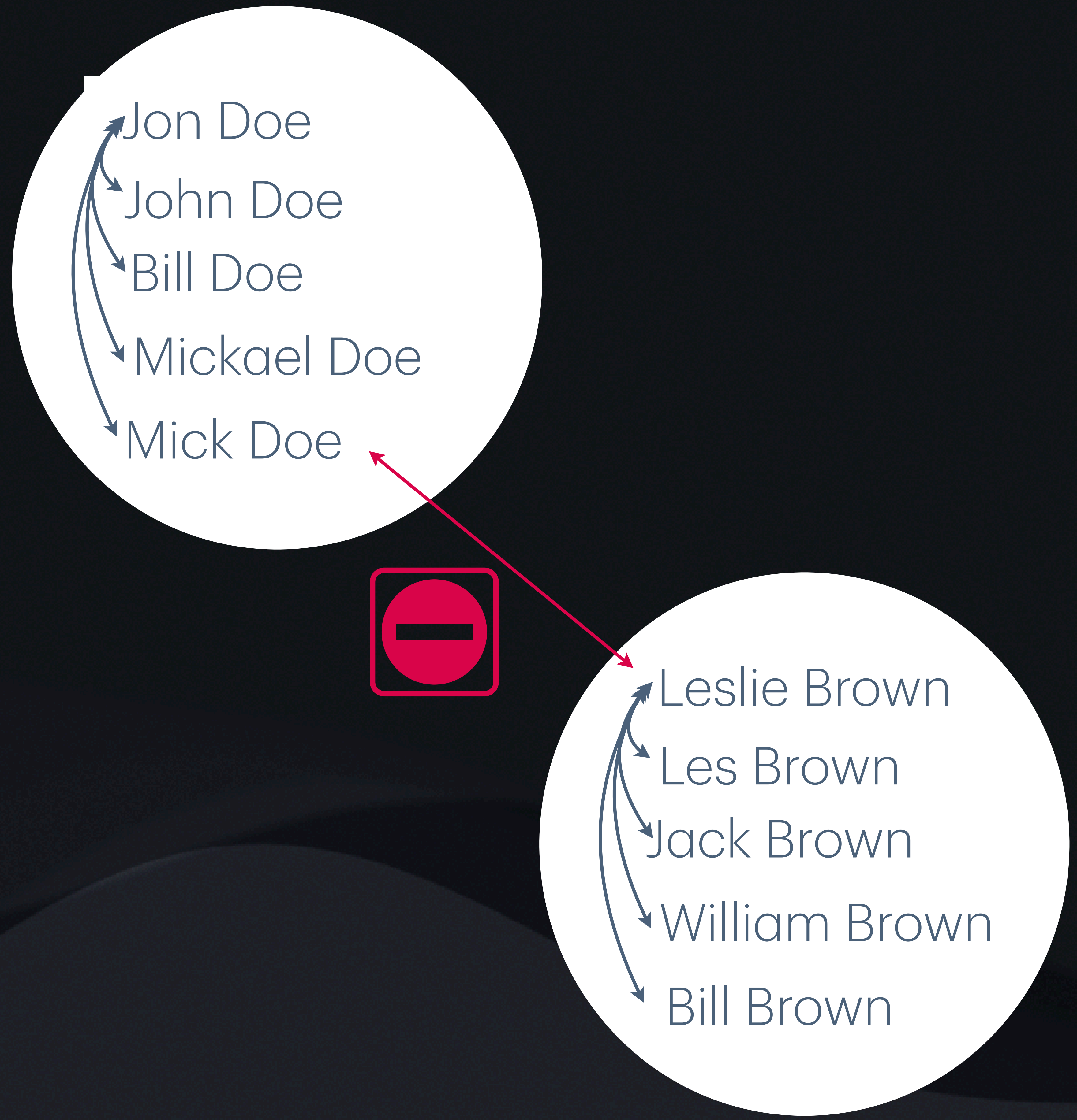


- Problem:
  Quadratic Explosion

Blocking

Tackling Quadratic Explosion

- Compare records w/in blocks ONLY
- Overlap blocks!!
  - "Same phonetic last name + same year"
  - "Same substr(geocode(address), 4)"



How do we compare?

We need to be fuzzy: "How far are these?"

- Numeric:
  - usually easy (subtract)
  - think normalisation
- Strings:
  - edit distance
  - there are many!

CHRSISTOPHAR CHRISTOPHER

Levenshtein = 0.63

Damerau-Levenshtein = 0.727

Jaro = 0.776

Jaro-Winkler = 0.843

++

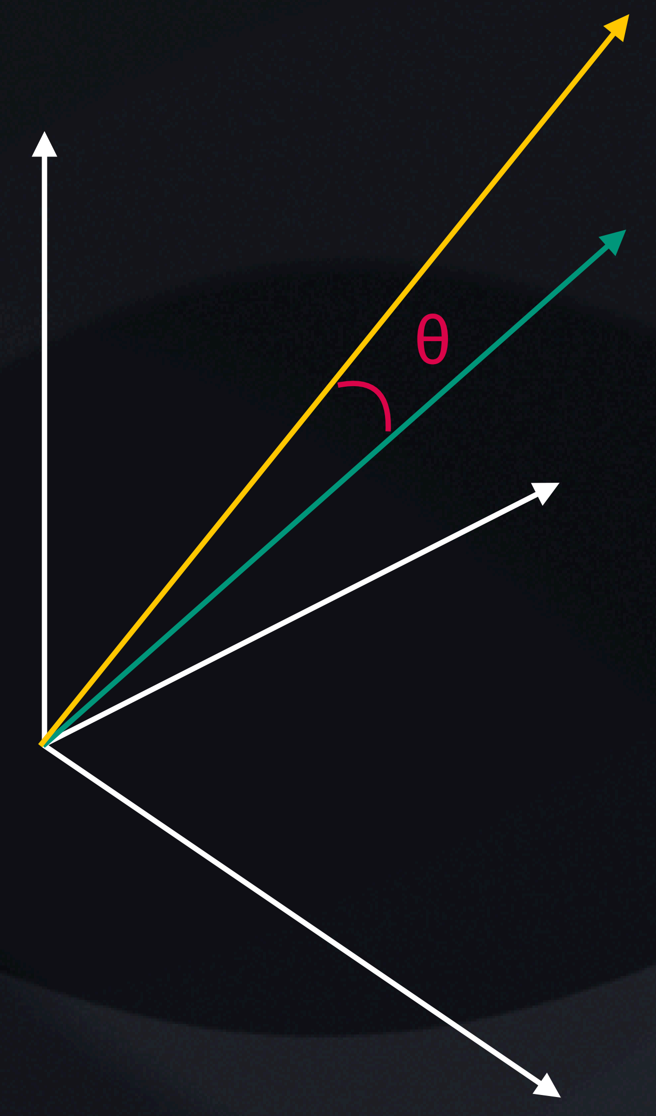
Domain Specific

https://github.com/easonanalytica/company_name_matcher

```
from company_name_matcher import CompanyNameMatcher

matcher = CompanyNameMatcher("sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2")
similarity = matcher.compare_companies("MERCK & CO", "MERCK AND COMPANY")
print(f"Similarity: {similarity}") # 0.903 ...
```

text-embedding



$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Cosine Similarity

How do we use distance metrics?

```
if jw(first_name) < 0.9 and lev(dob) < 0.9
or if jw(street_name) < 3 and jw(last_name) < 0.9
or if cos(company_name) < 0.7 and lev(phone) < 0.8
```

Problem:
Combinatorial Explosion

Do you trust more?

Same first name, same date of birth

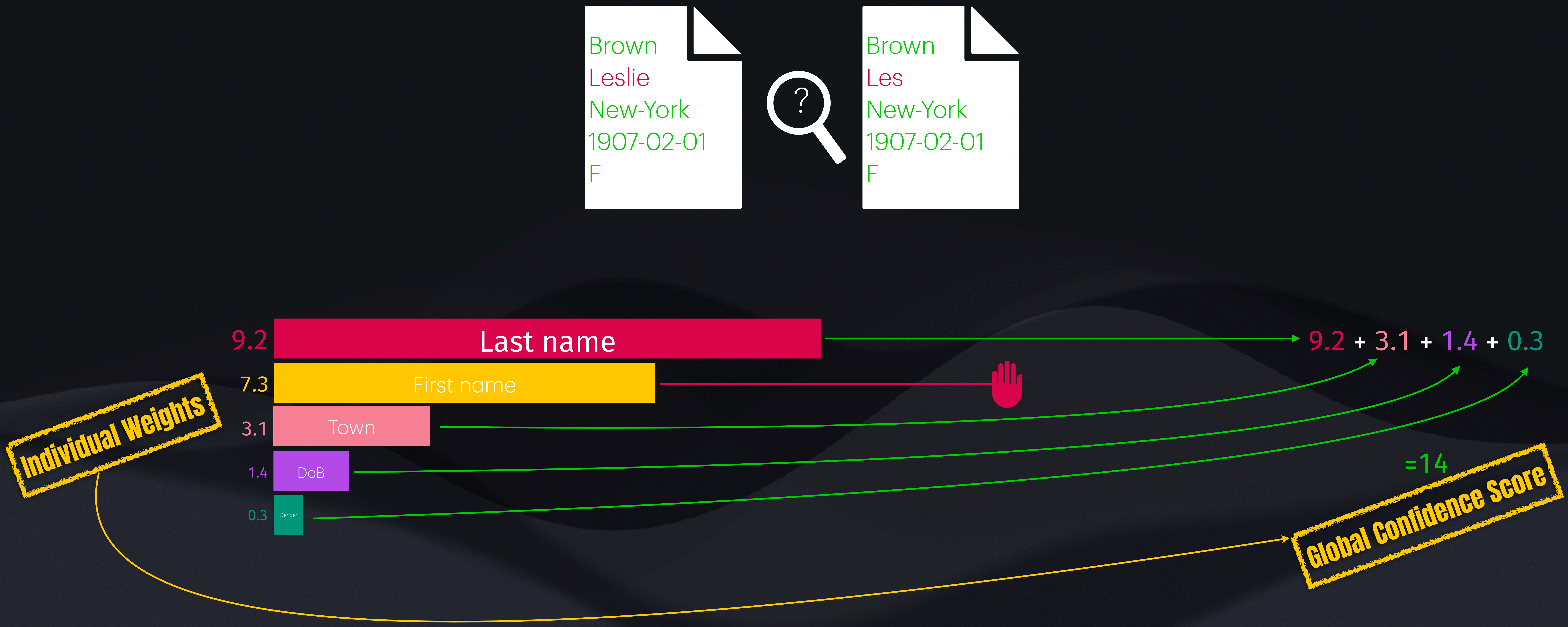
Or

Close full name, same date of birth

We need

<u>CONFIDENCE + WEIGHTS</u>

QUANTIFYING CONFIDENCE



Beyond intuition: Embracing Probabilities

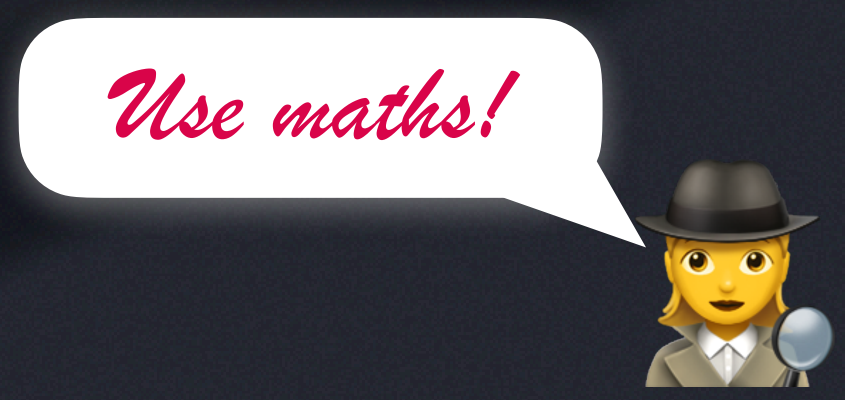
WHAT IS A WEIGHT?

How likely is it for two records to be a match if their last name match?

Bayesian probability […] is an interpretation of the concept of probability, in which, instead of frequency or propensity of some phenomenon, probability is interpreted as reasonable expectation[2] representing a state of knowledge[3] or as quantification of a personal belief.[4]

Intuitively $P(\text{Match} \mid \text{first name matches})$

For n features $(f_1, \ldots, f_n)$ $P(\text{Match} \mid f_1, \ldots, f_n)$



M, U AND WEIGHTS

Bayes Theorem

$$P(\text{Match} \mid \text{Observation}) = \frac{P(\text{Observation} \mid \text{Match}) \times P(\text{Match})}{P(\text{Observation})} \quad \lambda$$

$$P(\overline{\text{Match}} \mid \text{Observation}) = \frac{P(\text{Observation} \mid \overline{\text{Match}}) \times P(\overline{\text{Match}})}{P(\text{Observation})} \quad 1 - \lambda$$

💡 λ = P(Match) = Probability that 2 random records match

Substitution

$$Odd(\text{Match} \mid \text{Observation}) = \frac{P(\text{Observation} \mid \text{Match}) \cdot \lambda}{\cancel{P(\text{Observation})}} \times \frac{\cancel{P(\text{Observation})}}{P(\text{Observation} \mid \overline{\text{Match}}) \cdot (1 - \lambda)}$$

$$Odd(\text{Match} \mid \text{Observation}) = \frac{\lambda}{1 - \lambda} \times \frac{P(\text{Observation} \mid \text{Match})}{P(\text{Observation} \mid \overline{\text{Match}})}$$

We have many features

$$Odd(\text{Match} \mid \text{Observation}) = \frac{\lambda}{1-\lambda} \times \prod_{i=1}^{n} \frac{P(f_i \mid \text{Match})}{P(f_i \mid \overline{\text{Match}})}$$

$$\frac{m_f}{u_f} = K_f$$

Bayesian Factor

m_f = P(f | Match) = When 2 records match, how likely is it that they have the same last name?

m measures feature's **ACCURACY**

u_f = P(f | ~Match) = When 2 records do not match, how likely is it that they have the same gender?

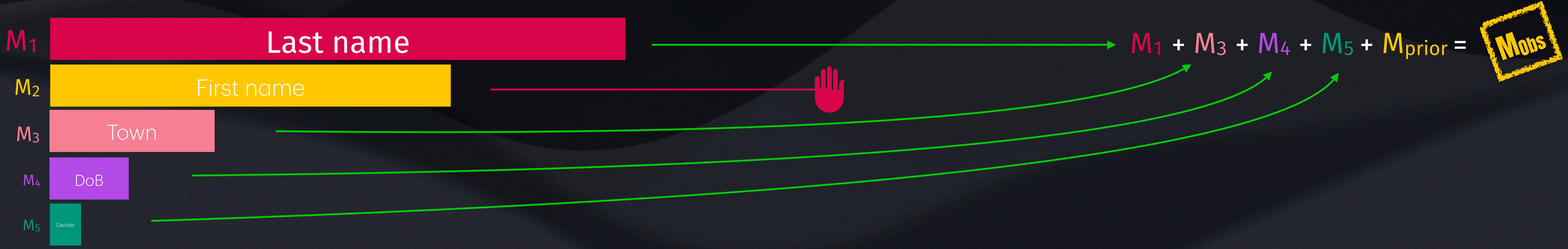
u measures feature's **COINCIDENCE**

WE WANT TO ADD

$$Odd(\text{Match} \mid \text{Observation}) = \frac{\lambda}{1-\lambda} \times \prod_{i=1}^{n} K_i$$

$$\underbrace{log_2(Odd(\text{Match} \mid \text{Observation}))}_{M_{obs}} = \underbrace{log_2(\frac{\lambda}{1-\lambda})}_{M_{prior}} + \sum_{i=1}^{n} \underbrace{log_2(K_i)}_{M_f}$$

$$M_{Obs} = M_{Prior} + \sum_{i=1}^{n} M_{f_i}$$



Fellegi-Sunter in Action: Meet SPLINK

Meet Splink

- MIT Licensed, Python
- 🇬🇧 Ministry of Justice
- Implements the Fellegi-Sunter model…

Doc: https://moj-analytical-services.github.io/splink/index.html

… in an interesting way

Estimating Parameters

λ: “How many matches do we expect?”

🕵️‍♀️ ➡️ Educated guess

u: “How often do people have the same name?”

🕵️‍♀️ ➡️ Random Sampling

Picking Parameters

m: "How clean is the data?"

"How often is the last name mistyped?"

Do we have a labelling column?

✅ Yes!

eg: Social Security Number

Supervised Learning

```
estimate_m_from_label_column("ssn")
```

ℹ️ [Maximum Likelihood Function]

⛔ No

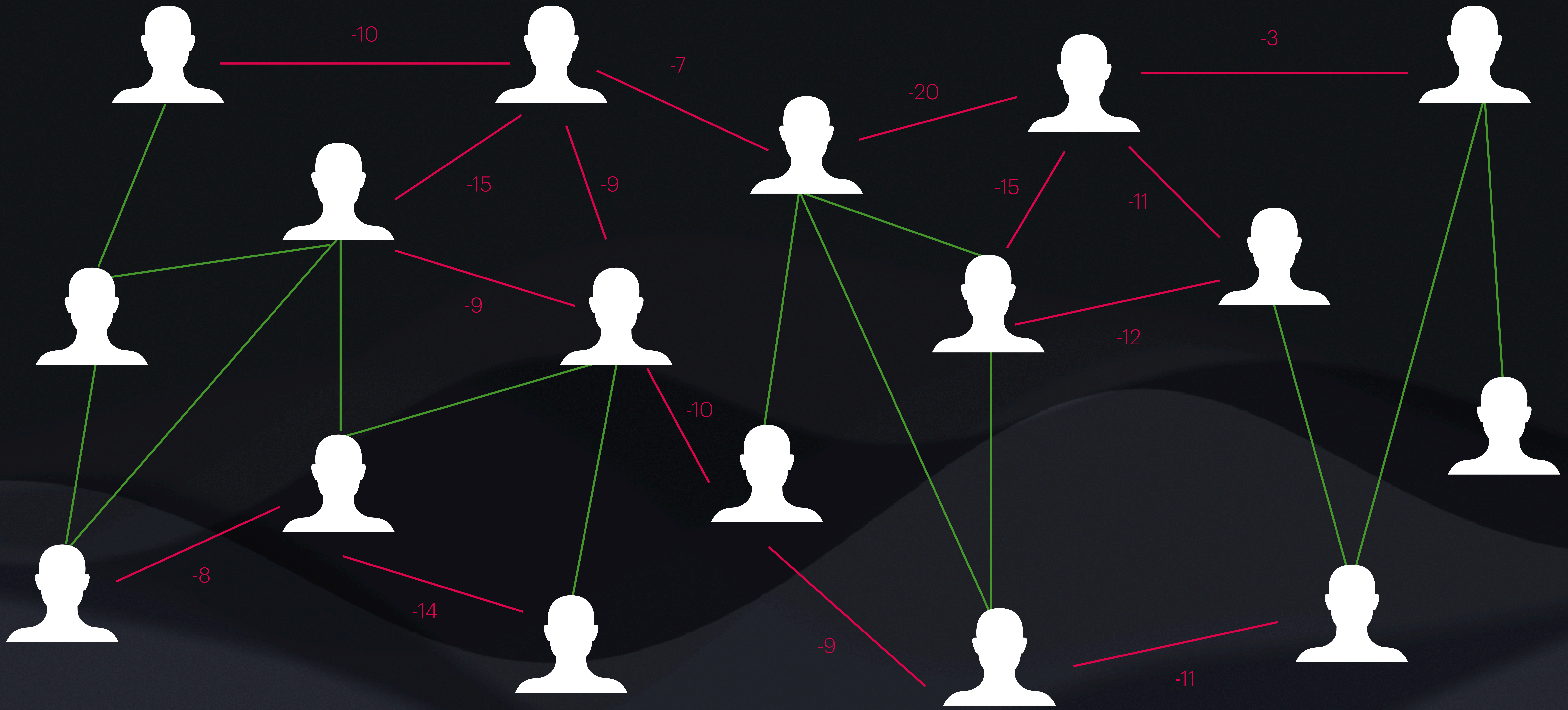
Unsupervised Learning

```
estimate_parameters_using_expectation_maximisation(block)
```

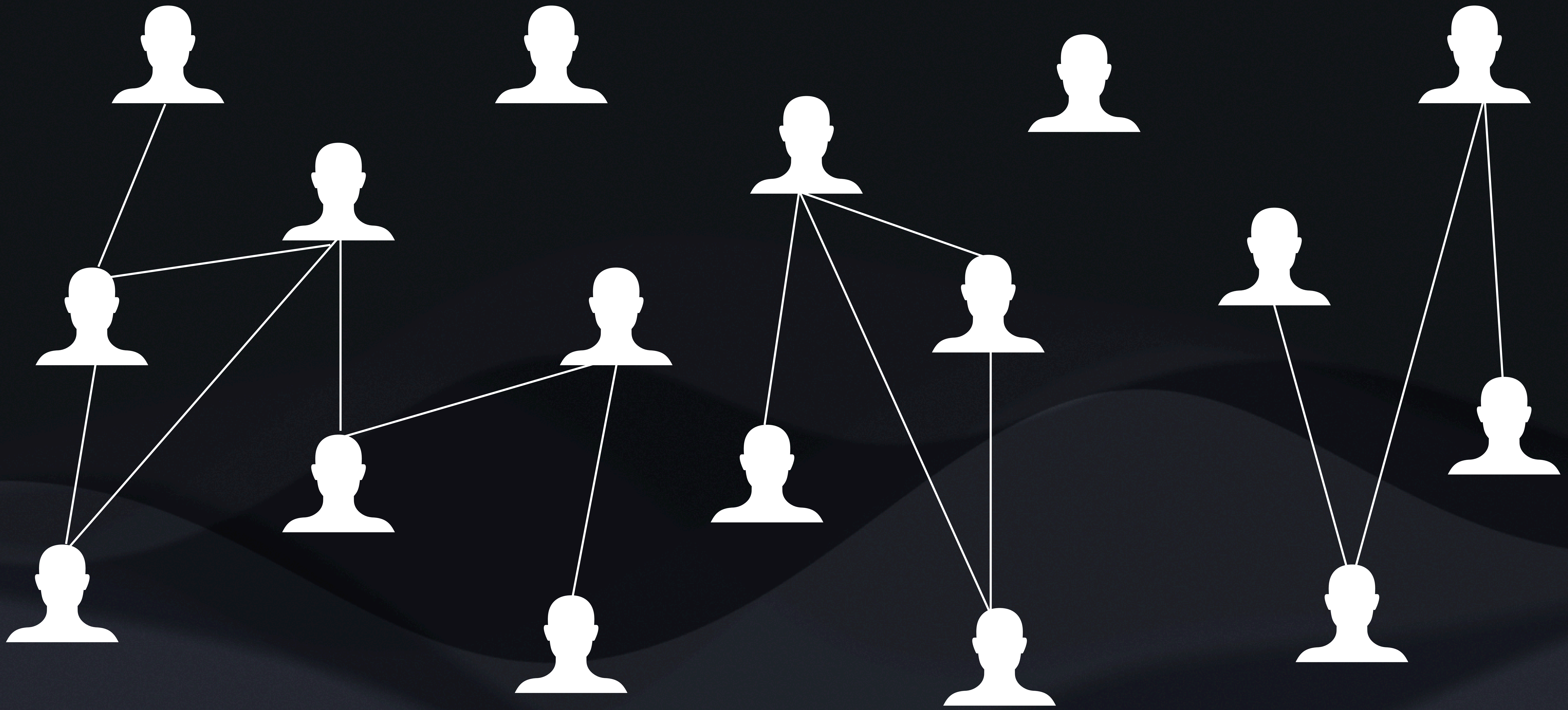
ℹ️ [Expectation Maximization]

Clustering: Grouping the Evidence

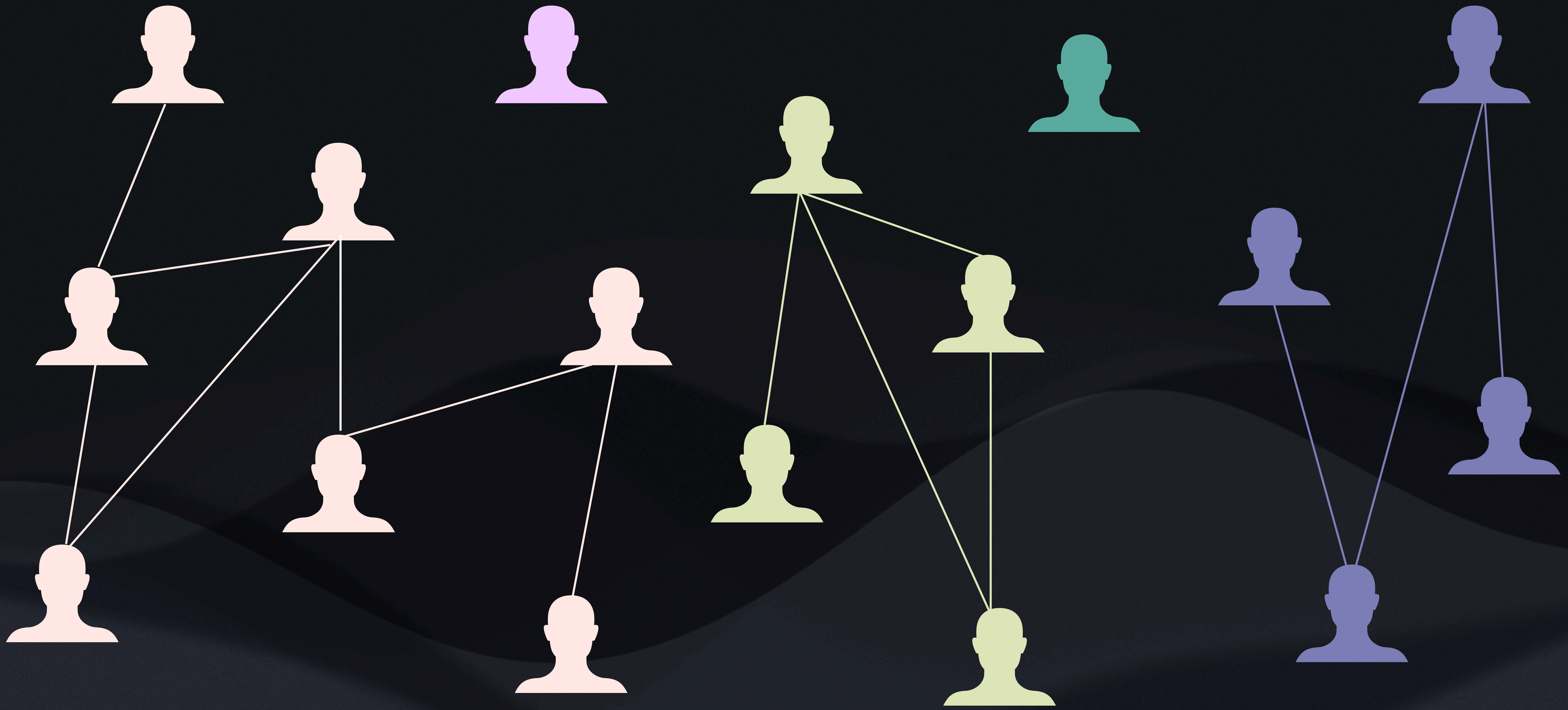
Links, now what?



LINKS, NOW WHAT?

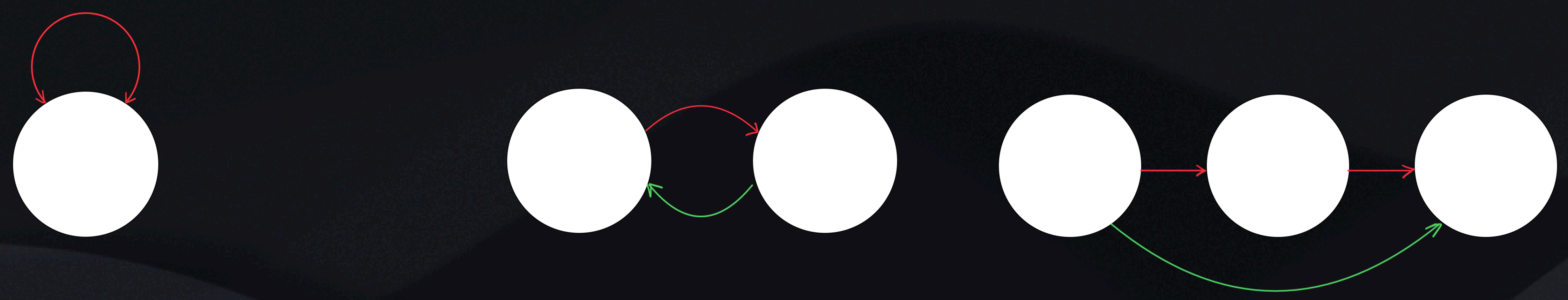


Links, now what?



Connected Components

In graph theory, a **component** of an undirected graph is a connected subgraph that is not part of any larger connected subgraph. […] Components are sometimes called **connected components**.

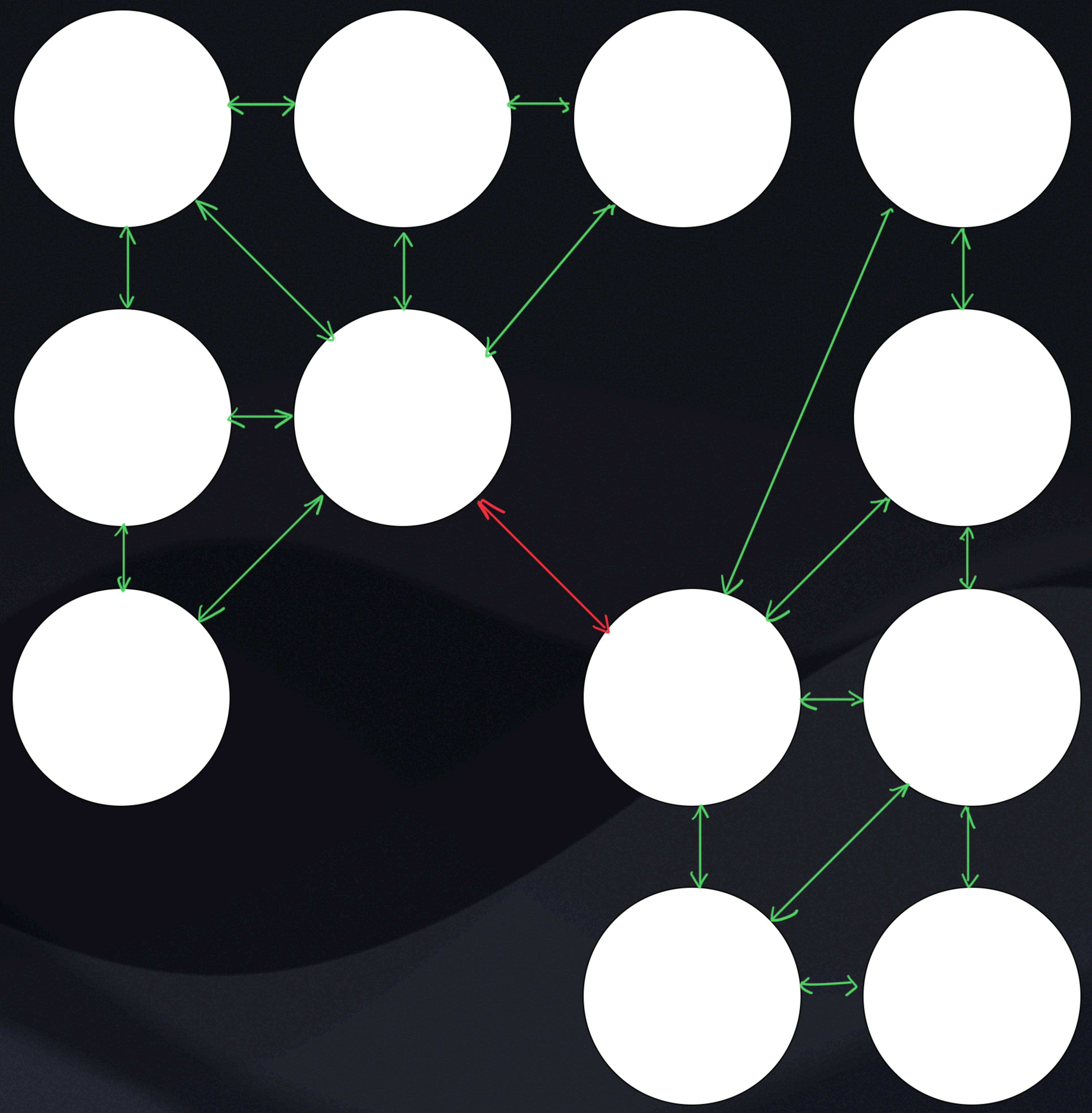


✅ Reflexive, Symmetric, Transitive relations (edges)

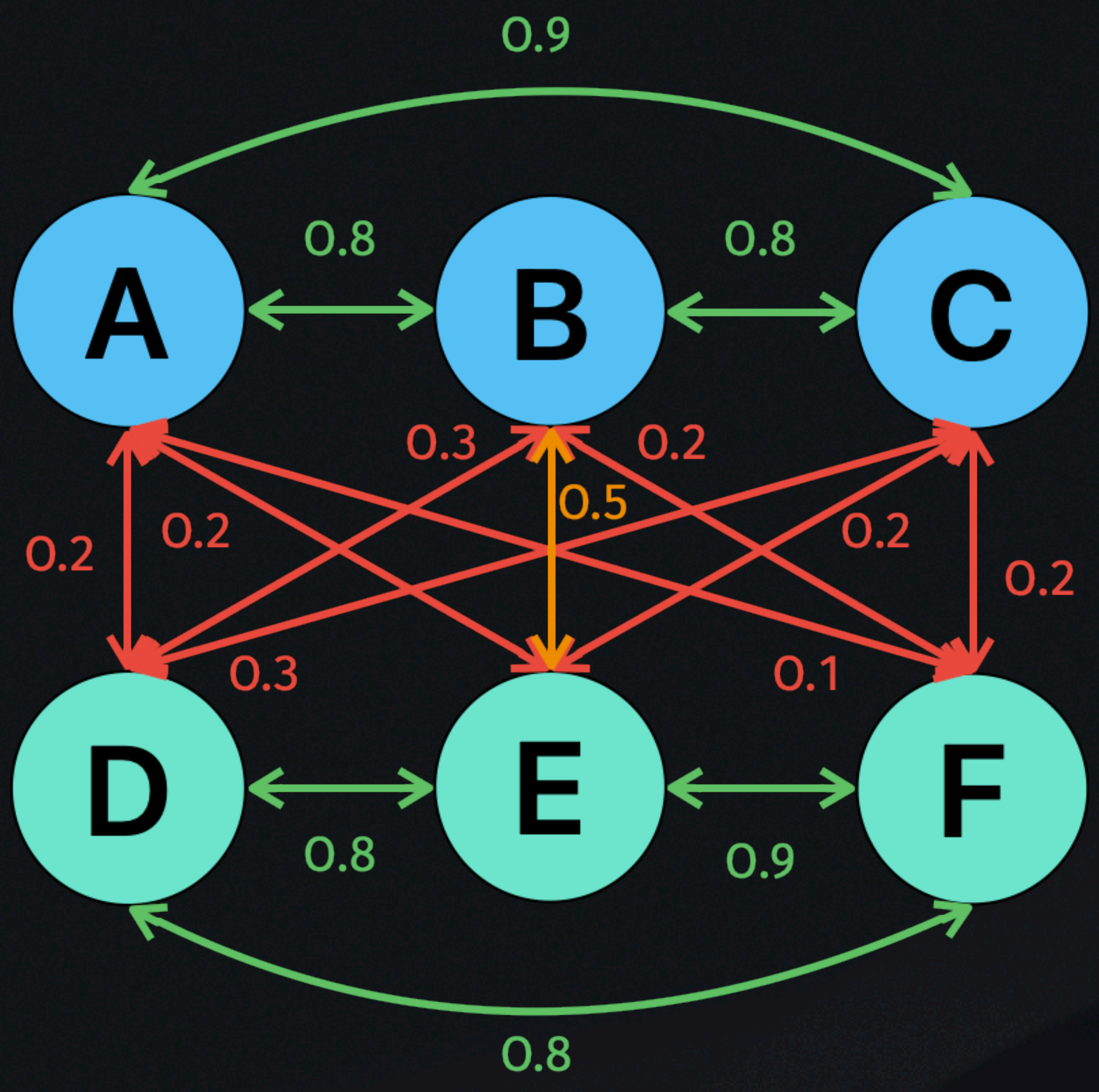
✅ Easy to implement (DFS, Union Find, …)

5.4. Connected Components

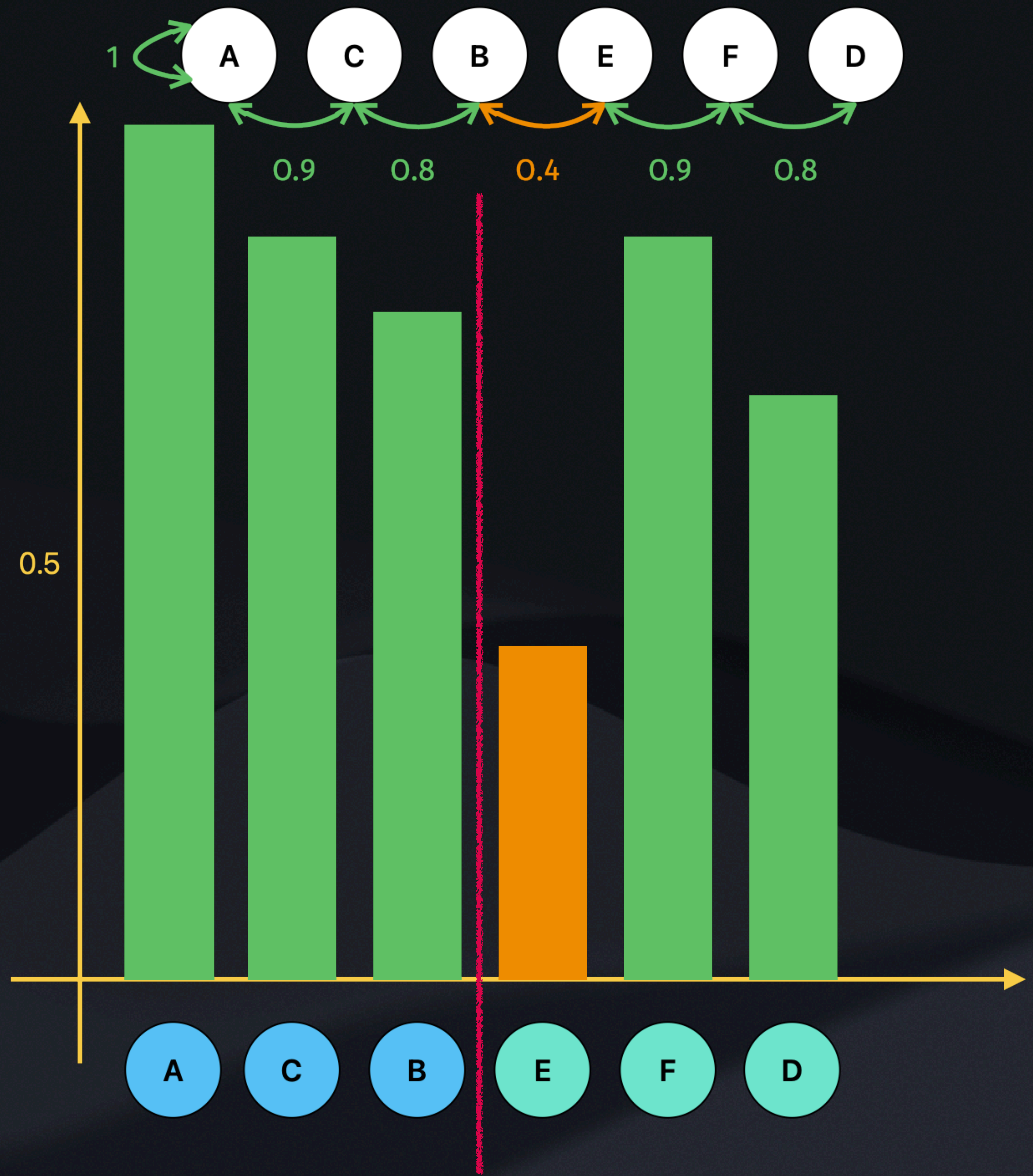
Weakness



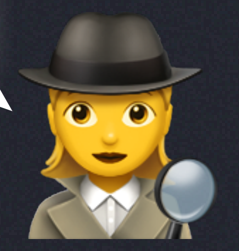
Clustering Algo: OPTICS



Reachability distance



M is a similarity metric!



Canonicalisation: Assembling the Truth

COMMON TECHNIQUES

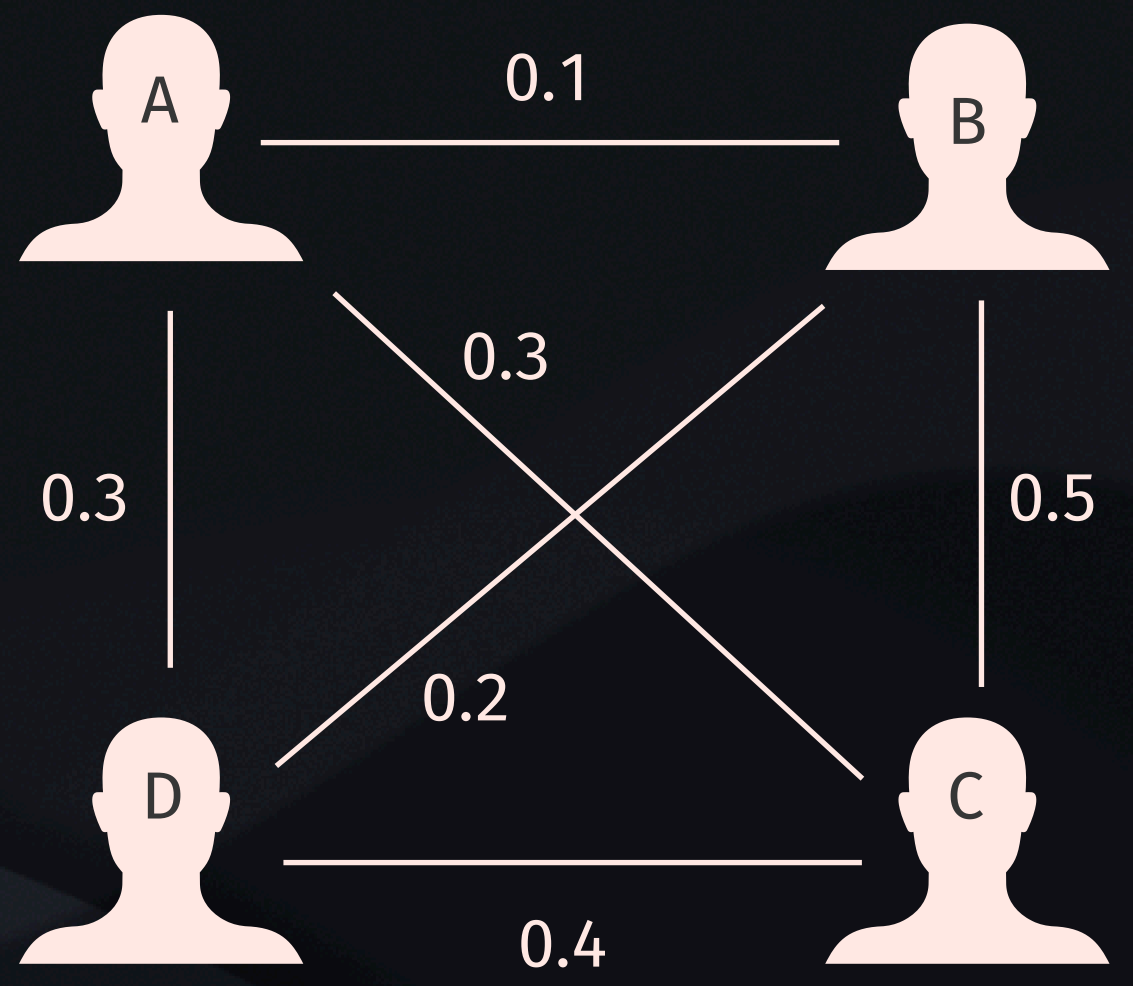
HEURISTICS

- Pick a random record in the cluster
- Majority Voting: most common value
- Mean, median, … for numerical value
- Most informative: longest string, most decimals
- Prioritised Source: more trustable

Using distances

Mini-Max

Remember we have metrics!



A	max(0.1, 0.3, 0.3) = 0.3
B	max(0.1, 0.5, 0.2) = 0.5
C	max(0.3, 0.5, 0.4) = 0.5
D	max(0.3, 0.2, 0.4) = 0.4

min

Evaluation
Reviewing the case

Monitor vs. Evaluate

Monitor

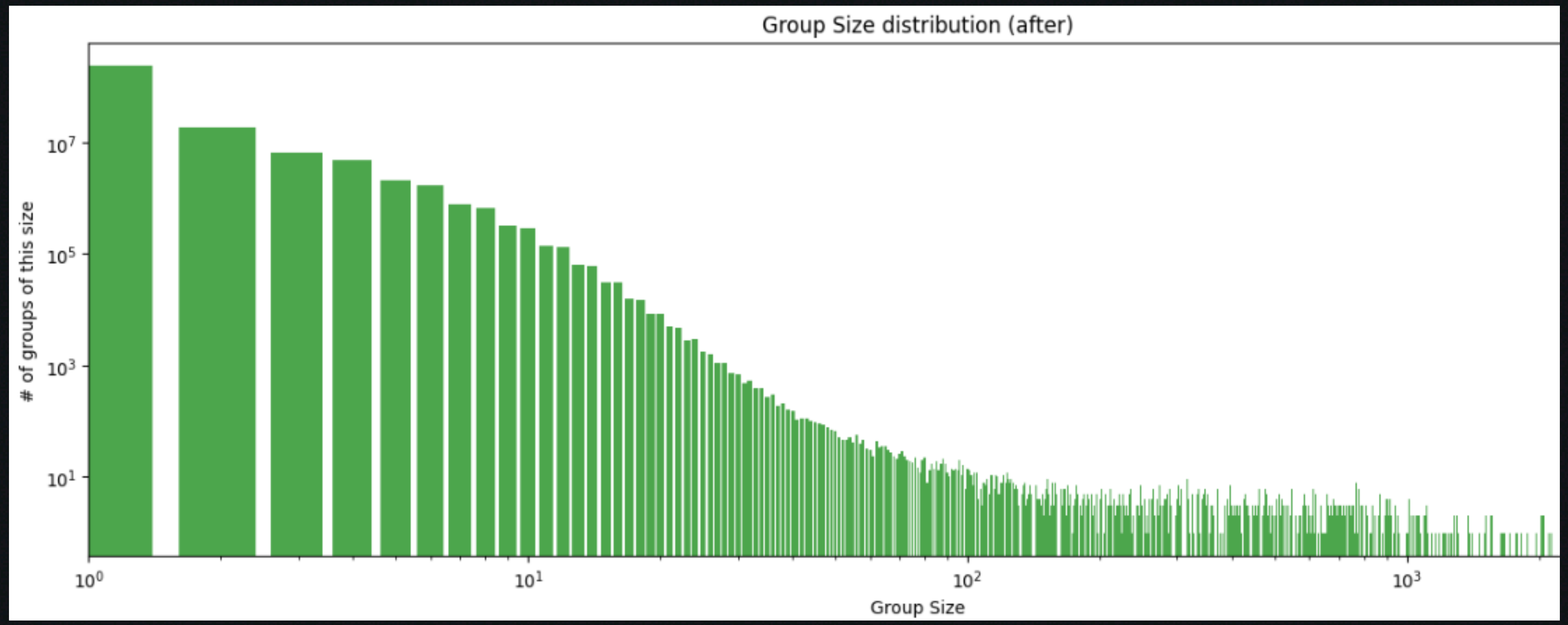
- Continuous supervision
- Unsupervised, no labelling
- Birds-Eye view
- Identify suspicious clusters

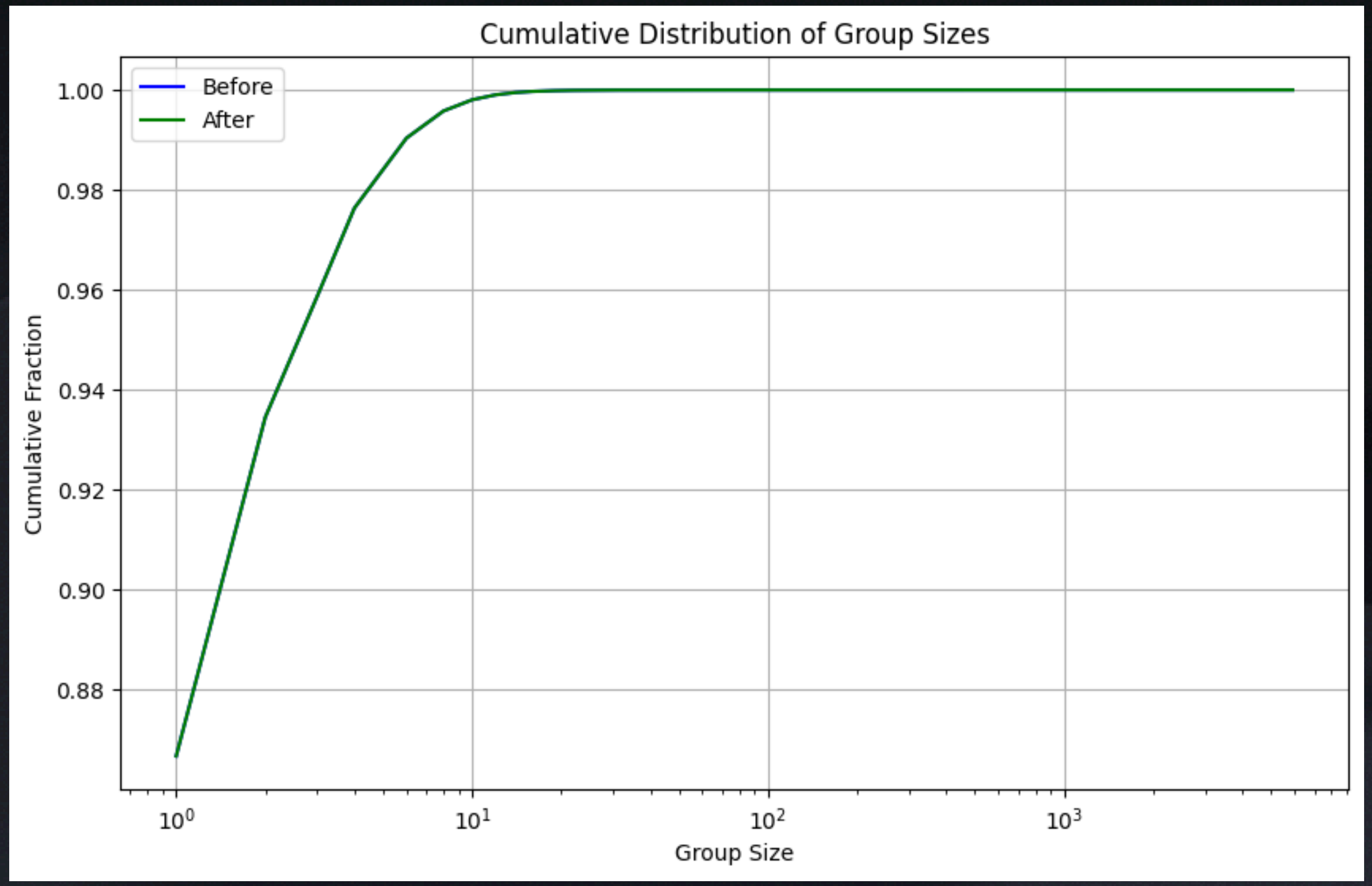
Evaluate

- Performance metrics
- Requires labelling
- Detailed metrics (recall, precision)
- Start from a sample and extrapolate

(Monitor) Cluster Stats

- Mean / Median cluster size
- Histogram
- Cumulative chart
- Matching rate
- Diversity Index: Hill Numbers





(Monitor) Entropy

In information theory, the **entropy** of a random variable quantifies the average level of uncertainty or information associated with the variable's potential states or possible outcomes. […]

https://en.wikipedia.org/wiki/Entropy_(information_theory)

$$H(X) = -\sum_{x \in X} p(x) \times log(p(x))$$

{Leslie, Leslie, Lesly, Leslie, Leslie, Leslie}

x	count(x)	p(x)	log(p(x))	p log(p)
Leslie	5 →	0,83	-0,08	-0,07
Lesly	1 →	0,17	-0,78	-0,13
H(X)				0,20

👀 *Frequentist approach of probabilities!*

{John, Jon, Joon, Johnathan, Jhon, John}

x	count(x)	p(x)	log(p(x))	p log(p)
John	2	0,33	-0,48	-0,16
Jon	1	0,17	-0,78	-0,13
Joon	1	0,17	-0,78	-0,13
Johnathan	1	0,17	-0,78	-0,13
Jhon	1	0,17	-0,78	-0,13
John	1	0,17	-0,78	-0,13
H(X)				0,81

Data Scientist takeaways

Cleansing	*Part of my work is to represent the data differently*
Linking	*Using maths: distance metric, (bayesian) probabilities*
Clustering	*By being probabilistic, we can leverage more clustering methods*
Canonicalisation	*Sometimes, heuristics are fine! We can go further though*
Validation	*Look into the data distribution! Then use stats*

Blocking
Linking
Clustering
LSH
Token-Based
Deterministic (Key-Based)
Graph
Meta-Blocking
Deterministic
Rule-Based Grouping
GROUP BY
Phonetic Encoding
Similarity Functions
Edit Distance
Levenshtein
Jaro-Winkler
Monge-Elkan
Cosine Similarity
Jaccard Index
Probabilistic / Stochastic
Cond...nal Random Fields
Fellegi & Sunter
Splink
FastLink
Bayesian Networks
DBlink
Exchanger
CPT
Parameter Learning
Structure Learning
MCMC
esian Networks_ A C...
Document · 337 KB
Graph
GNN
ML
LLM
2312
PDF Document · 882 KB
Random Forests
Support Vector Machines
Active Learning
Deep Learning
ITER+ CliqueRank
icde18-entity-resolution
PDF Document · 441 KB
Label Propagation
2108
PDF Document · 628 KB
Markov Random Fields
ER_with_Markov_Logic
PDF Document · 146 KB
Knowledge Graph Embeddings
zhao1701/extend deep-ER
zhao1701/extending-deep
github.com
deepmatcher-sigmod...
PDF Document · 1,7 MB
1710
PDF Document · 2,2 MB
Dedupe
dedupeio/dedupe: :id: A pytho...
github.com
2211
PDF Document · 1,5 MB
Zingg
zinggAI/zingg: Scalable ide...
github.com
P16-1059
PDF Document · 251 KB
Merge-Center Clustering
Connected Components
Center Clustering
Density-Based
OPTICS
HDBSCAN
DBSCAN
Hierarchical
Single / Complete / Average Linkage
ingle-linkage clustering
en.wikipedia.org
Correlation Clustering
Correlation clustering
en.wikipedia.org
Similarity-Based
Canopy Clustering
K-Means