Project presentation

Machine Learning and Big Data Processing(ELEC–Y591)

Cédric Hannotier Mathieu Petitjean Hasan Can Yildirim

June 13, 2018

Outline

Context

Matching

Similarity matching:

$$Sim(r_1, r_2) = \frac{\sum Sim_{\cos}(r_{1i}, r_{2i})}{|supp(r_1) \cup supp(r_2)|}$$

Adapted to the dataset: rating ρ and timestamp d, giving more value to statistically rare elements.

$$Score(r, aux) = \sum_{i \in supp(aux)} \frac{1}{\log|supp(i)|} \left(e^{-\frac{|\rho_i - \rho'_i|}{\rho_0}} + e^{-\frac{|d_i - d'_i|}{d_0}} \right)$$

Matching

1: **for** each record r_i in R **do**
2: **for** each entry aux_i in aux **do**
3: Compute $Score(r_i, aux_i)$
4: **end for**
5: Compute $\sigma_S = \texttt{stdev}(Score)$
6: Find $S_1 = \max(Score(r_i, aux))$
7: Find $S_2 = \max(Score(r_i, aux) \backslash \{S_1\})$
8: Compute $\phi = (S_1 - S_2)/\sigma_S$
9: **if** $\phi > 1.5$ **then**
10: Match found !
11: **end if**
12: **end for**

$$\phi = \frac{S_1 - S_2}{\sigma_S}$$

Outline

Data overview

The Netflix dataset

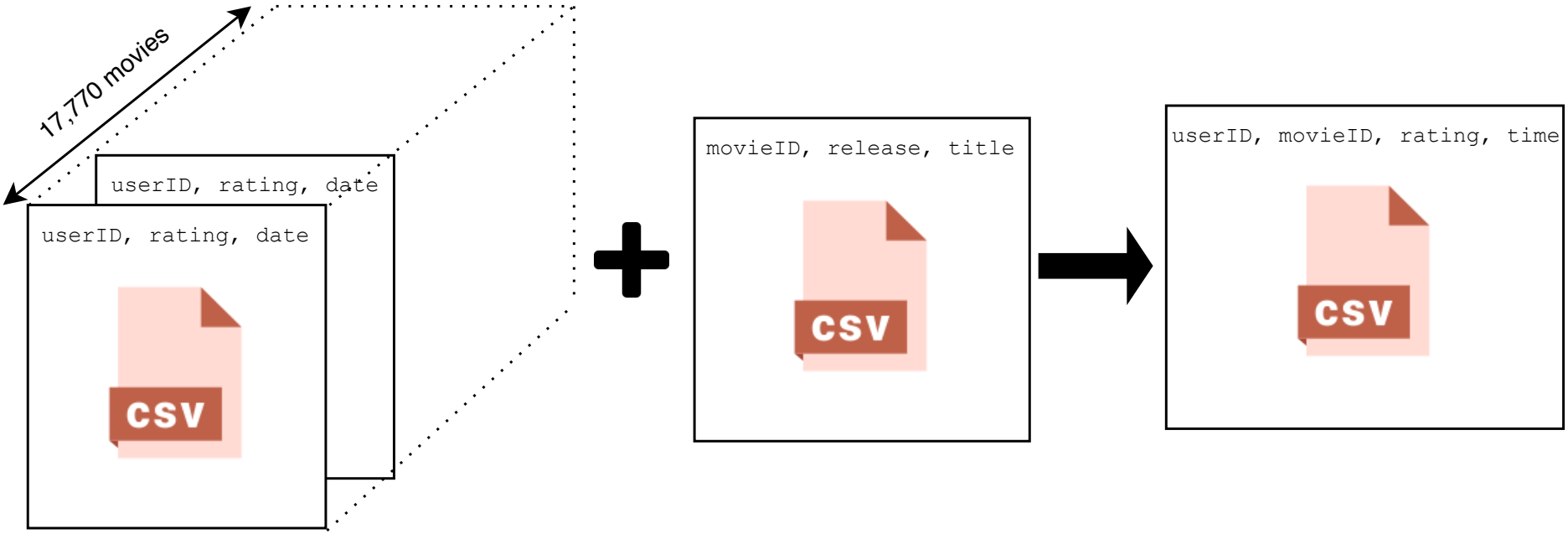
- 5.5 GB of data
- 17,770 movies
- 480,000 users
- $\simeq$ 100 million ratings

against

The MovieLens dataset

- 875.6 MB of data
- 27,278 movies
- 138,493 users
- $\simeq$ 20 million ratings

Netflix data reshaping



Data filtering

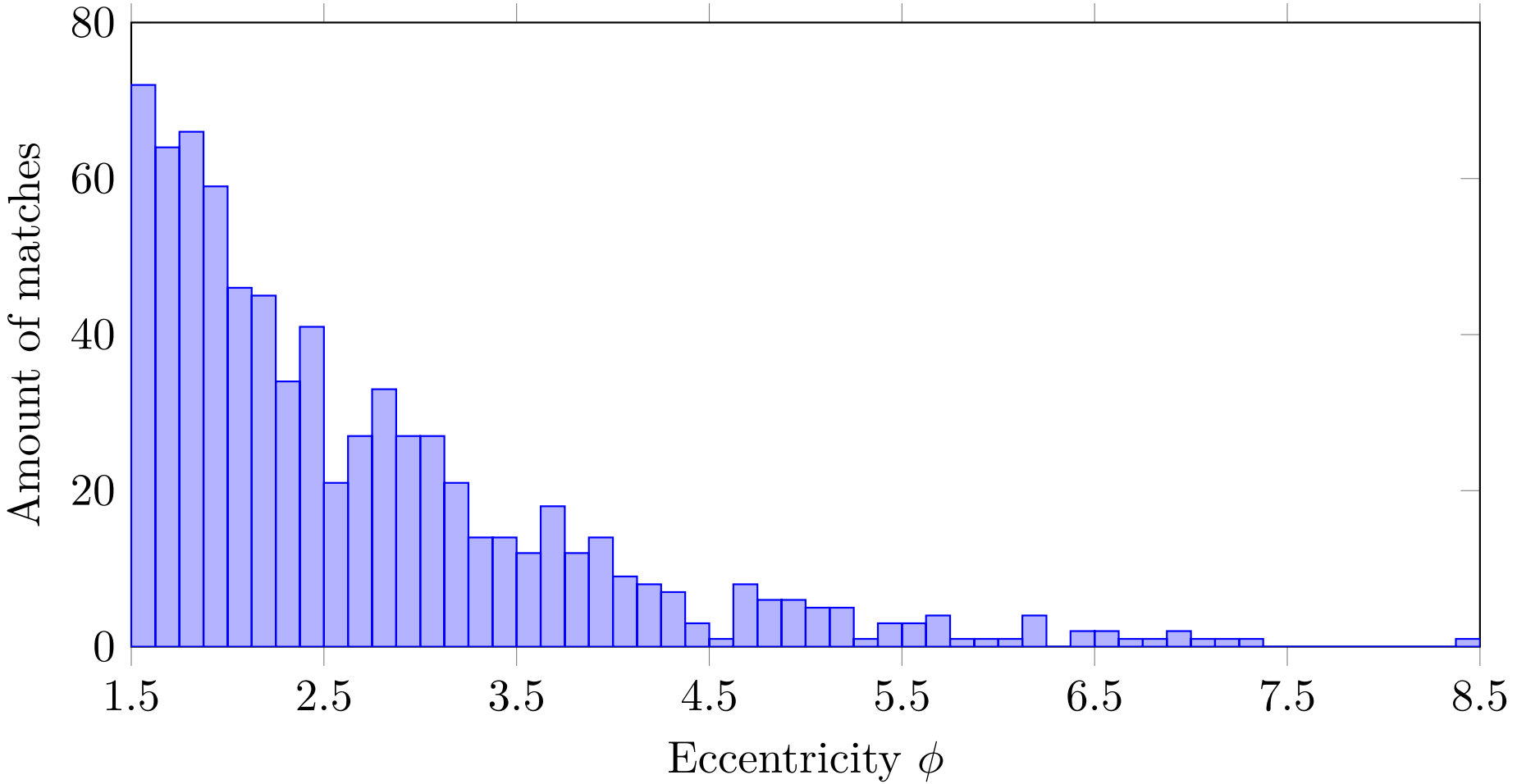
1. Discard MovieLens entries based on timestamps.
2. Discard all movies not present on both datasets. A movie was uniquely identified by its title and release date.
   ⚠"Lord of the Rings, The" and "The Lord of the Rings (2001)"
3. Recast timestamps: from YYYY-MM-DD and elapsed seconds to common reference.
4. Rounded MovieLens ratings.

$\Longrightarrow$ 5800 common movies, 52,875 users remaining in ML and 478,756 in Netflix.

Outline

Matches

- 3600 Netflix and 3000 MovieLens users
- a few more than 800 matches, 5 with $\phi > 7$



Validation

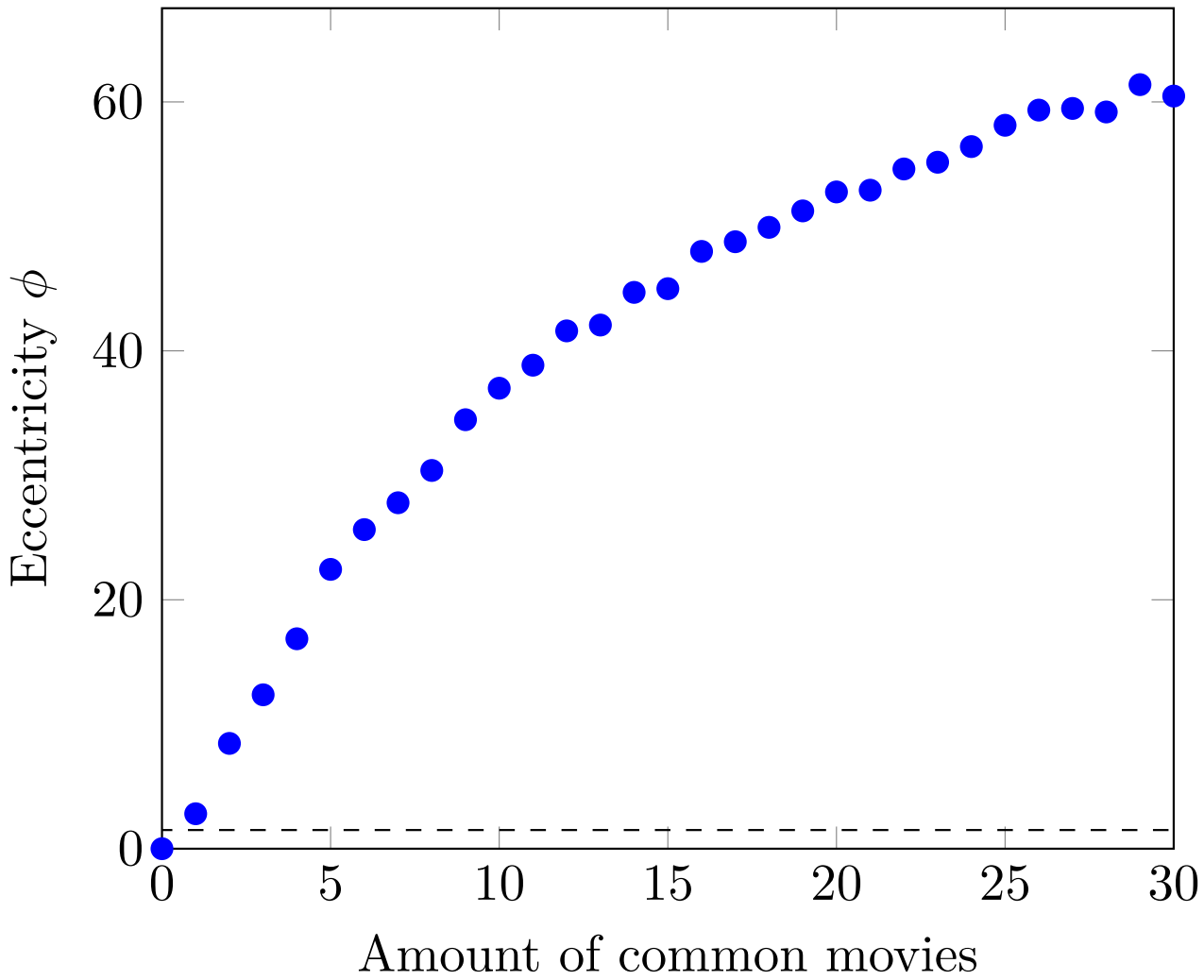
How to validate the results?

- No knowledge of the ground truth of matching users

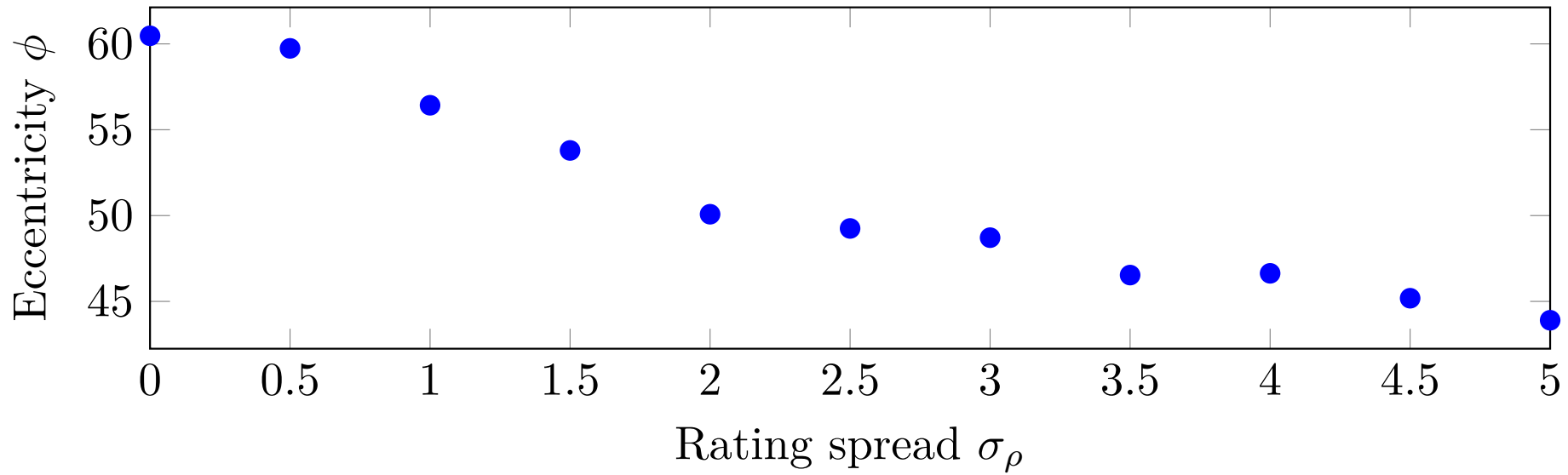
Validation procedure

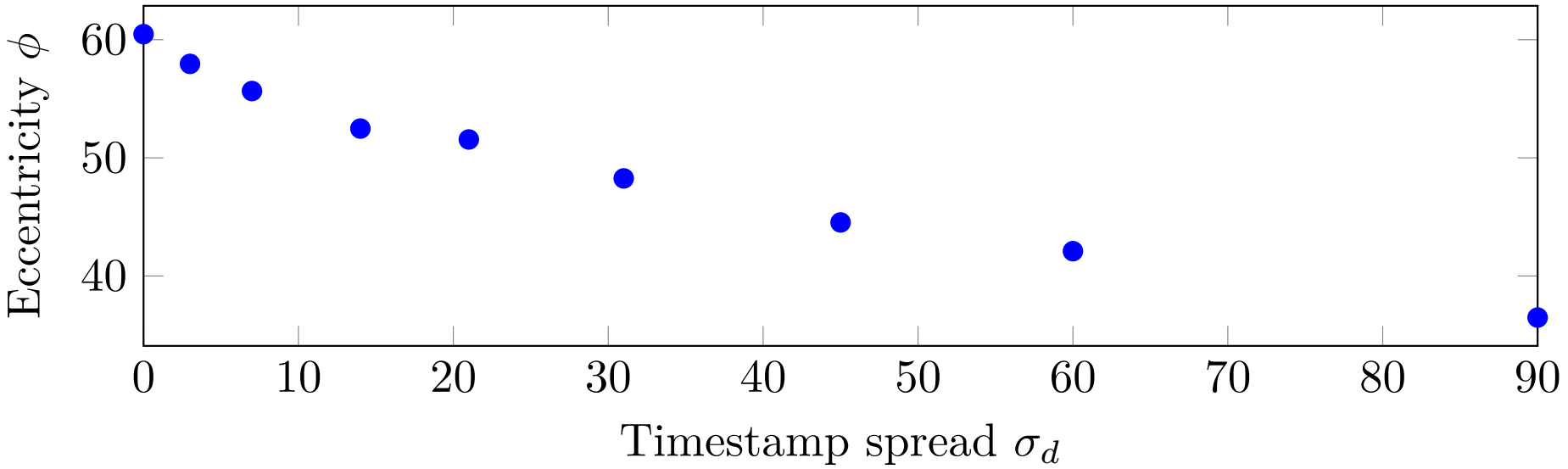
- Put a dummy user in both datasets
- Vary the number of common movies
- Perturb the rating and the timestamp with uniform distributed noise

Validation


Eccentricity ϕ
0
20
40
60
0
5
10
15
20
25
30
Amount of common movies

Robustness


Eccentricity ϕ
60
55
50
45
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Rating spread σ_ρ


Eccentricity ϕ
60
50
40
0
10
20
30
40
50
60
70
80
90
Timestamp spread σ_d

Conclusion

- $\phi_{\max} \approx 8.5$ while the validation process showed $\phi > 30$
- The original Netflix attack: $\phi_{\text{match}} = \{18, 25\}$

$\Rightarrow$ cannot conclude statistical quasi-certainty of de-anonymization

Possible improvements

- Use more than 0.04 % of the possible user combinations
- Tuning of the parameters ϕ, d_0, ρ_0
- Add more features (e.g. movie genres)
- Increase the errors impact on the scoring