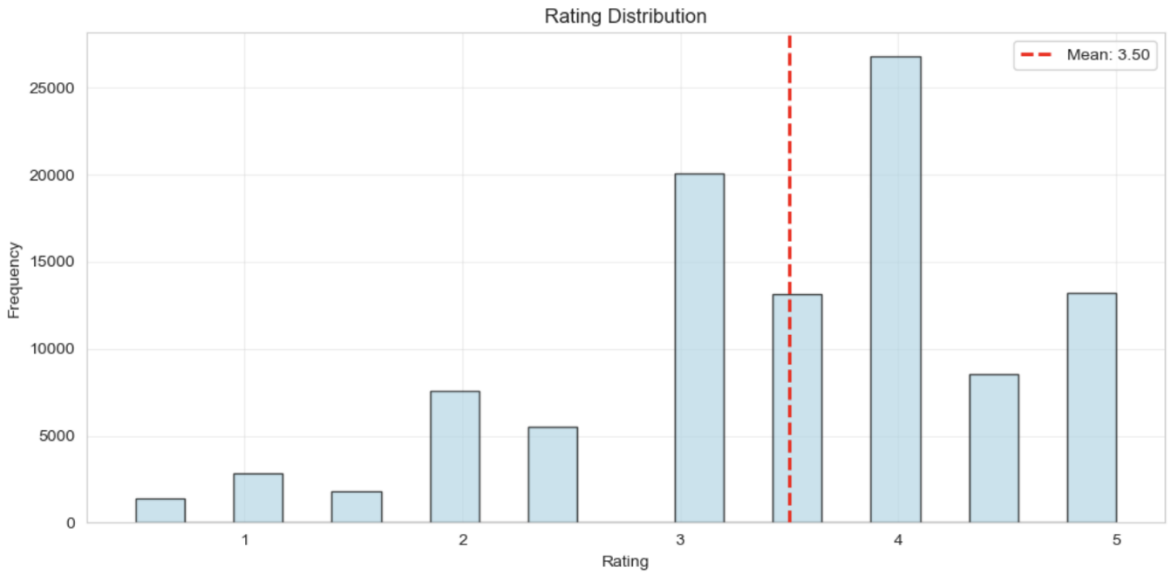
User-Based Collaborative Filtering for Group Recommendations

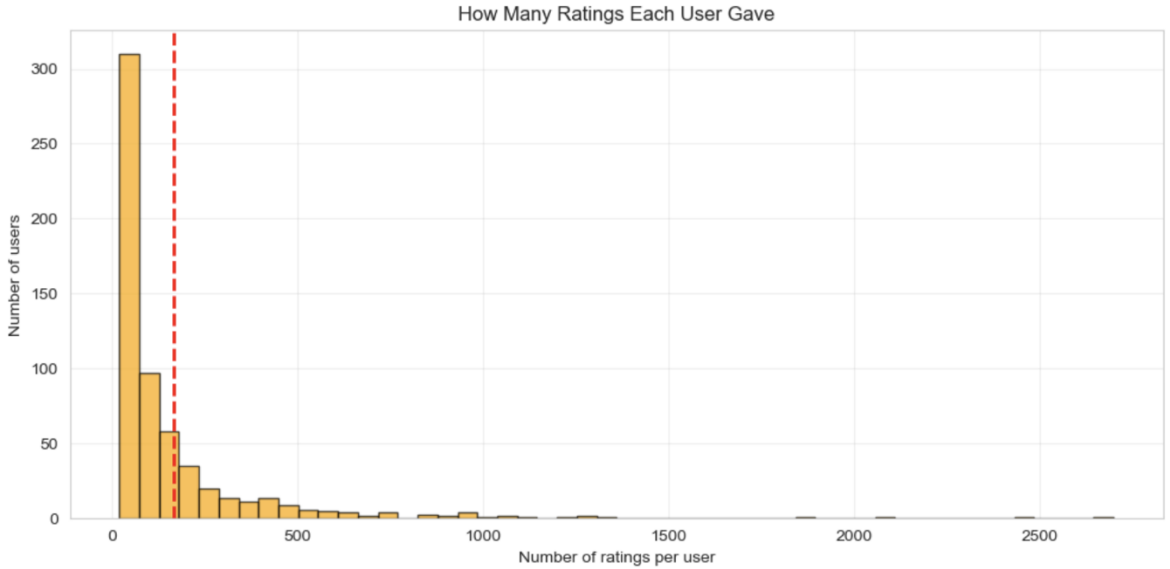
- Students: Oskari Perikangas, Xiaosi Huang

- Course: DATA.ML.360-2025-2026-1 Recommender Systems
- Date: November 4, 2025

Dataset & Challenges Identified

MovieLens 100K - 100836 ratings from 610 users – Containing 9724 different movies - Ratings on a 0.5 to 5 star





Challenge 1: Rating Bias

The average is 3.50, more high ratings than low ratings.
Most ratings cluster at 4 and 5 stars.
Users tend to rate movies they LIKE and skip movies they DISLIKE.
The data is skewed, it doesn't represent the true quality.
This bias affects the recommendation predictions

Challenge 2: Data Sparsity

Almost 10,000 movies in total, more users only rate 100-200 movies.
There are 4744 out of 9724 (48.8%) have 2 or fewer ratings.
Data Sparsity makes collaborative filtering very difficult.
Users don't overlap enough to find reliable similarities.

Finding Similar Users & Predicting Ratings

⭐ Taking **user 1** as an example

- **Similarity Formula** (Pearson Correlation):

$$\text{sim}(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

```
Top 10 similar users to User 1:
  1. User 476: 0.7869
  2. User 210: 0.7676
  3. User 297: 0.7063
  4. User 44: 0.6844
  5. User 394: 0.6506
  6. User 248: 0.6247
  7. User 369: 0.6121
  8. User 72: 0.5964
  9. User 344: 0.5876
  10. User 112: 0.5839
```

Prediction Formula (from class slides):

$$\text{pred}(a,p) = \bar{r}_a + \frac{\sum_{b \in N} \text{sim}(a,b) \cdot (r_{b,p} - \bar{r}_b)}{\sum_{b \in N} \text{sim}(a,b)}$$

Predicted ratings for User 1:

Movie_id	Predicted	Avg Rating	# Users Rated
318	5.00	4.43	317
1704	5.00	4.08	141
6874	4.97	3.96	131
8798	4.66	3.76	44
46970	3.75	3.25	28
48516	5.00	4.25	107
58559	4.62	4.24	149
60756	4.24	3.55	28
68157	4.80	4.14	88
71535	4.41	3.88	53
74458	4.62	4.02	67
77455	5.00	4.04	13
79132	5.00	4.07	143
80489	4.91	3.95	22
80906	5.00	4.29	12

Enhanced Pearson Similarity

Question: If User 476 has rated 11 movies in common with User 1, and User 297 has rated 17 movies in common, who should we trust MORE?

```
Comparison: Pearson (Part b) vs Enhanced Pearson (Part d)
Neighbor    Pearson      Enhanced     Common
------------------------------------------
User 476    0.7869       0.8069       11
User 210    0.7676       0.7876       11
User 297    0.7063       0.8463       17
User 44     0.6844       0.7444       13
```

$$\text{sim}(a,b) = \text{pearson}(a,b) + \frac{\text{common\_items} - 10}{50}$$

Where:

- **pearson(a,b)**: Pearson correlation between user a and b
- **common_items**: Number of movies both users rated
- **10**: Minimum common movies required
- **50**: Scaling factor (1 extra movie = +0.02 bonus)

Reward users with MORE data overlap

Example (User 297) :
For every extra movie beyond 10,
we add 0.02 to the similarity score.

If User 297 has 17 common movies:
17 minus 10 equals 7 extra movies
7 divided by 50 equals 0.14 bonus
We add 0.14 to their Pearson score

Group Recommendations

New challenge : How to Combine 3 Different Users' Preferences?

Example: Group User [1, 414, 599]

Steps: Find unrated popular movies and predict score

Find common movies , Aggregate using methods as below,

Method 1: Average aggregation

$$\text{group\_score}(m) = \frac{\sum_{u \in \text{group}} \text{pred}(u, m)}{|\text{group}|}$$

AVERAGE METHOD - Top 10

Rank	Movie	Average Score
1	1230	4.75
2	1276	4.38
3	1234	4.30
4	1304	4.29
5	1207	4.16
6	1193	4.16
7	1288	4.09
8	2324	4.09
9	8368	3.98
10	7147	3.96

Method 2: Least misery aggregation

$$\text{group\_score}(m) = \min_{u \in \text{group}} \text{pred}(u, m)$$

LEAST MISERY METHOD - Top 10

Rank	Movie	Min Score
1	1230	4.50
2	1276	3.76
3	1304	3.60
4	1234	3.60
5	1203	3.48
6	1207	3.36
7	1193	3.31
8	74458	3.29
9	2324	3.26
10	1288	3.19

COMPARISON

Movie	Average	Least Misery	Difference
1230	4.75	4.50	0.25
1276	4.38	3.76	0.62

What if we could have BOTH ?

High satisfaction for the group

Low disagreement between members

Group Recommendations

Method 3: Balancing Strategy - Minimize satisfaction differences between users

Formular: Balancing Strategy (Masthoff 2004)

$$\text{Argmin} \sum_{u,v \in G} |sat(u,i) - sat(v,i)|$$

Where:

- **sat(u, i)** = satisfaction of user u for item i
- **G** = group of users
- The item that minimizes disagreement is selected

```
Group: User [1, 414, 599]

User 1: 50 recommendations (avg: 4.99)
User 414: 21 recommendations (avg: 3.23)
User 599: 50 recommendations (avg: 3.00)

============================================================
COMPARISON: All Three Methods (Top 10 from Average)
============================================================
```

Movie	Average	Least Misery	Balancing	Disagreement
1230	4.75	4.50	4.70	0.50
1276	4.38	3.76	4.26	1.24
1234	4.30	3.60	4.16	1.40
1304	4.29	3.60	4.15	1.37
1207	4.16	3.36	4.00	1.60
1193	4.16	3.31	3.99	1.69
1288	4.09	3.19	3.91	1.81
2324	4.09	3.26	3.74	3.47
8368	3.98	2.99	3.58	4.03
7147	3.96	2.91	3.75	2.09