Michael Grossberg

Intro to Data Science CS59969

Recommender Systems

Many Applications

Further Resource

Survey on recommender systems by Michael D. Ekstrand et al.

•http://files.grouplens.org/papers/FnT%20CF%20Recsys%20Survey.pdf

Good slides from Stanford lecture by Lester Mackey

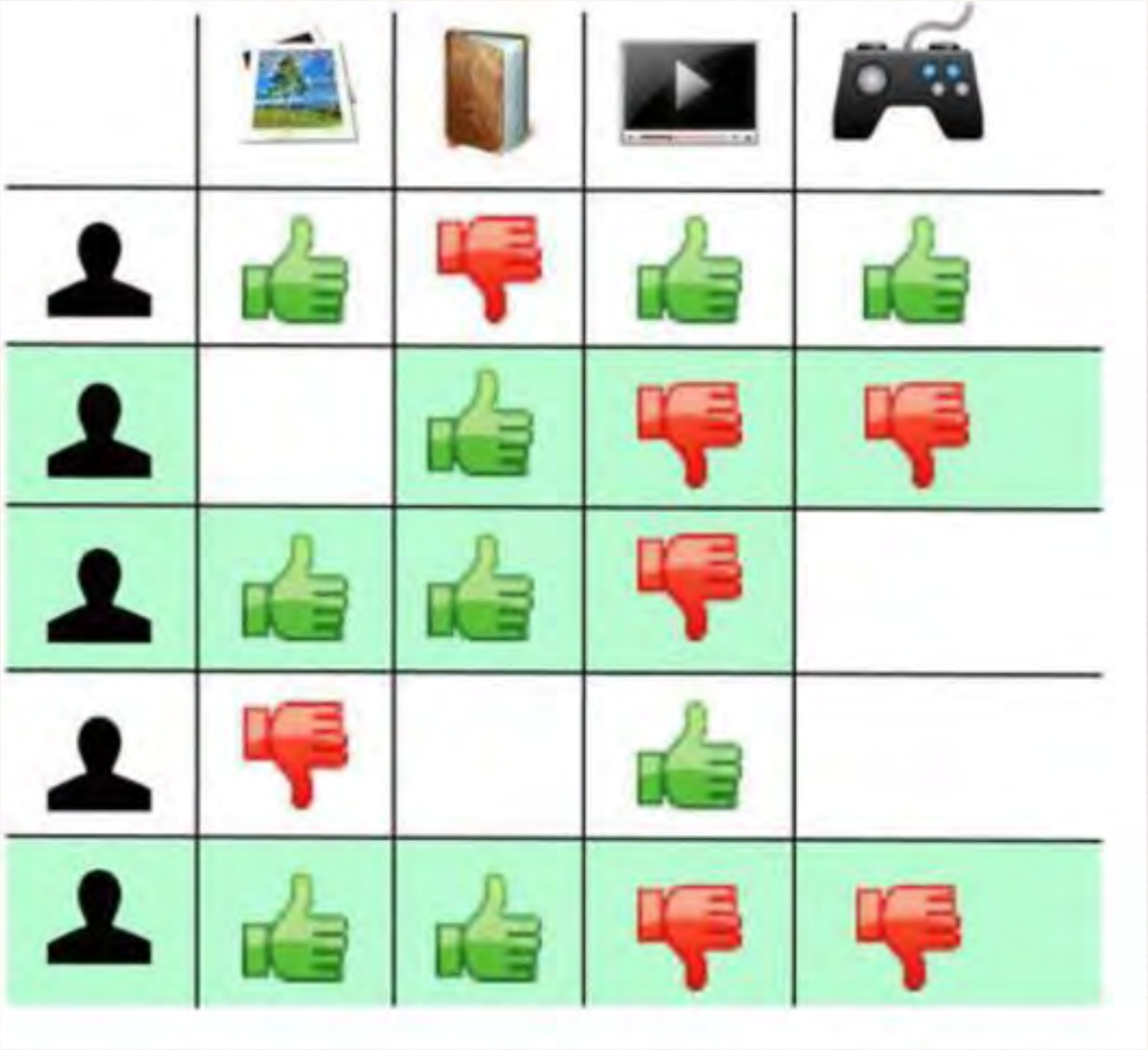
•http://web.stanford.edu/~lmackey/papers/cf_slides-pml09.pdf

Recommender Data

Items

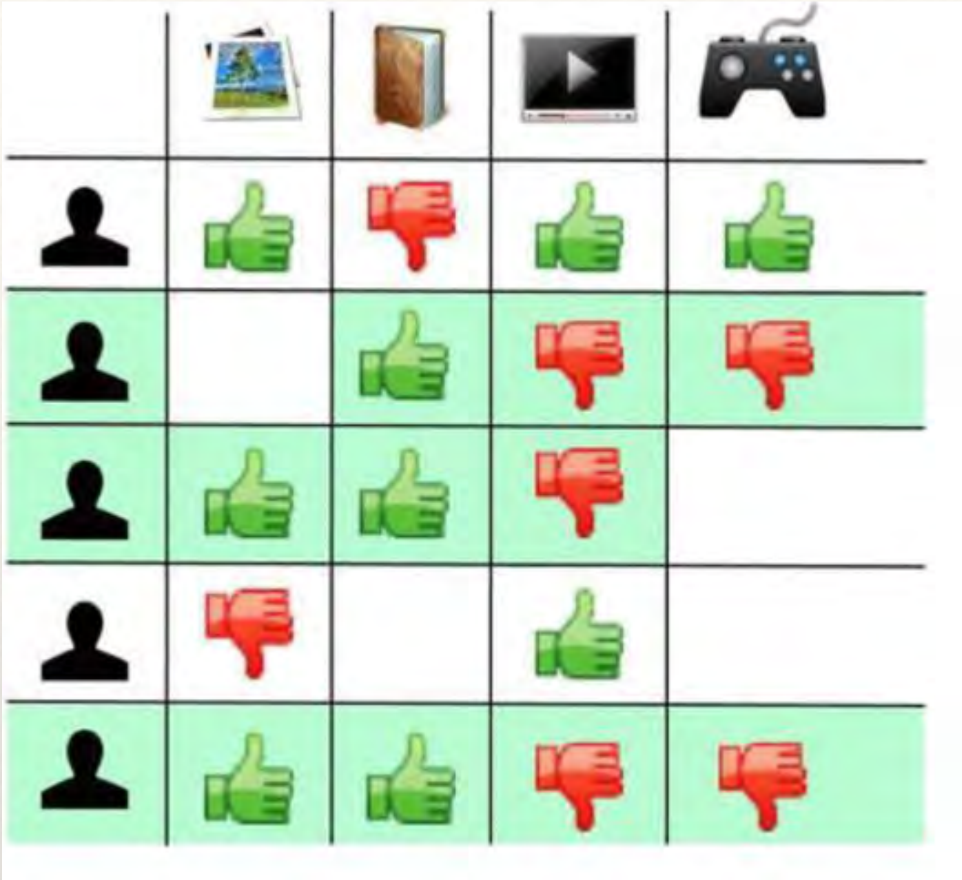
Ratings

Users



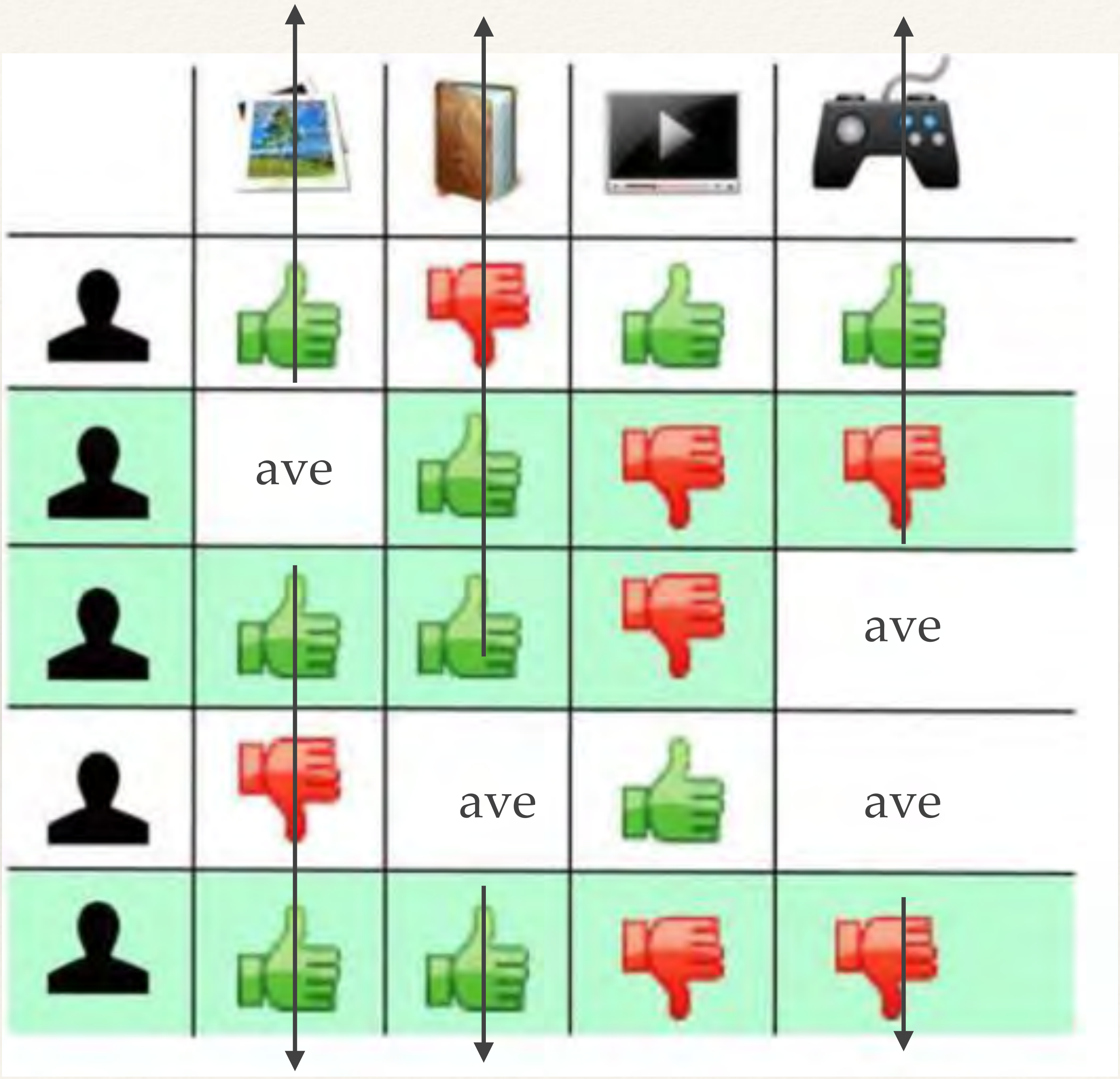
Missing Data

Problem



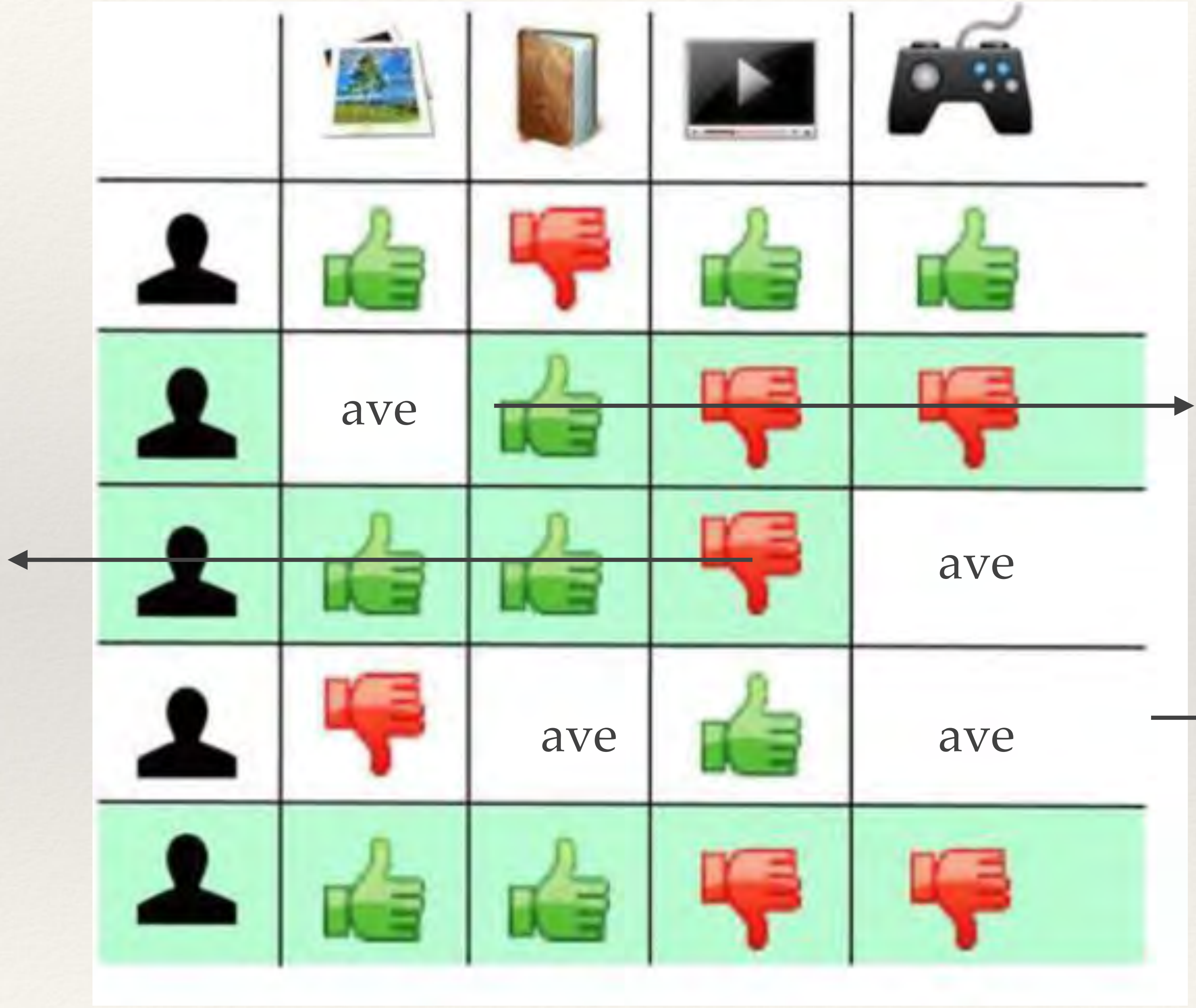
- Fill in missing data
- Make recommendations: You might like this movie

Global Fill in Missing Data



Average cols

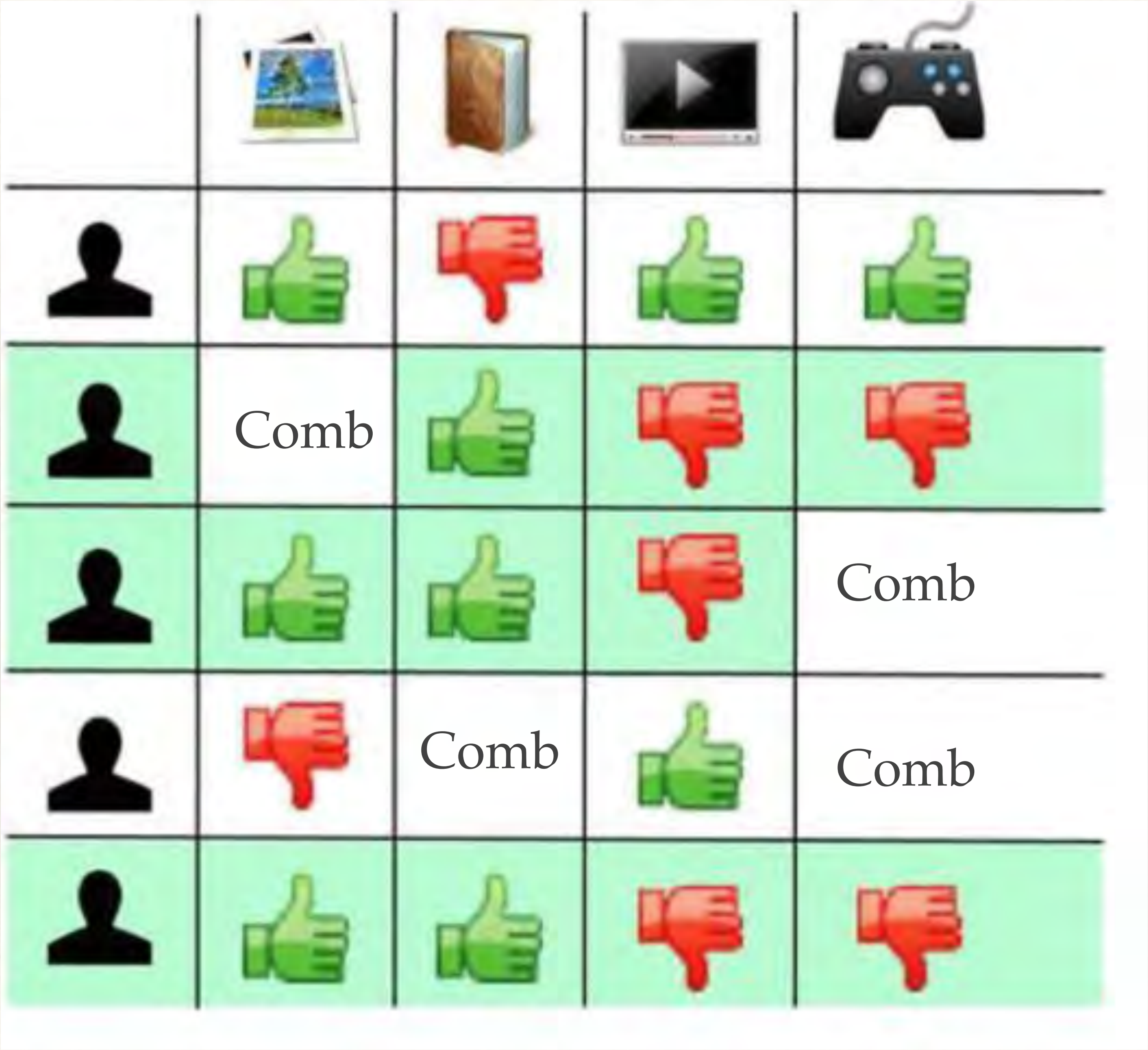
Global Fill in Missing Data



Average rows

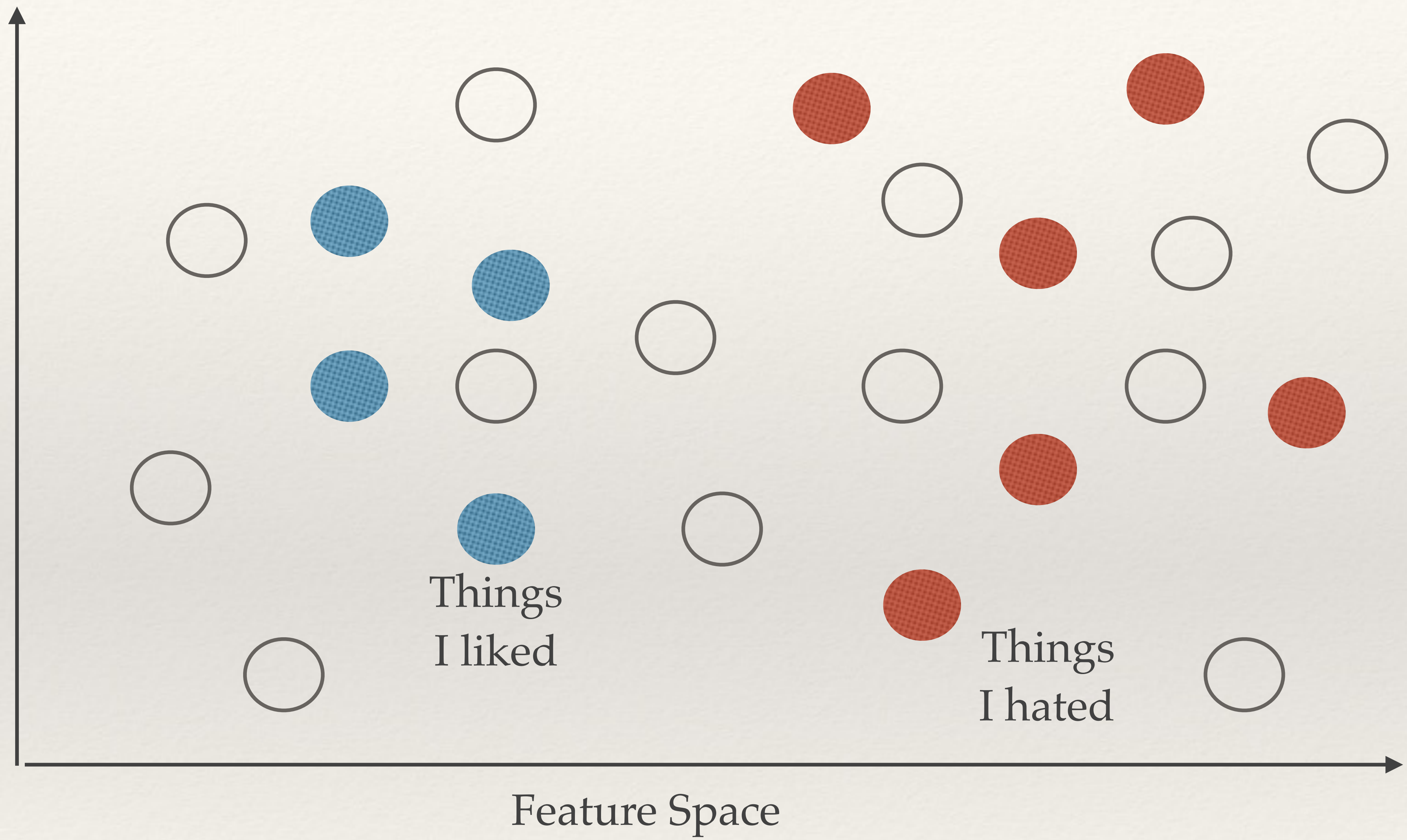
Global Fill in Missing Data

Combine: ave rating + user bias + item bias?



Not very personalized

Approach1:
Content Based Filtering



Classic Classification Problem

Approach: 2
Collaborative Filtering

These users like these toys

These users like these toys

Which kind of user are you?

Differences

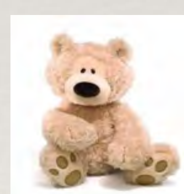
Content Based Recommendation




Likes

Recommends

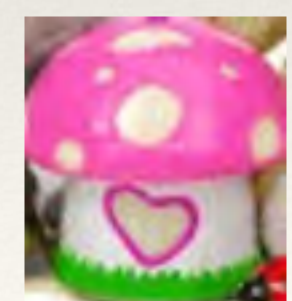
Collaborative Recommendation


Likes

Similar Users

Similar preferences



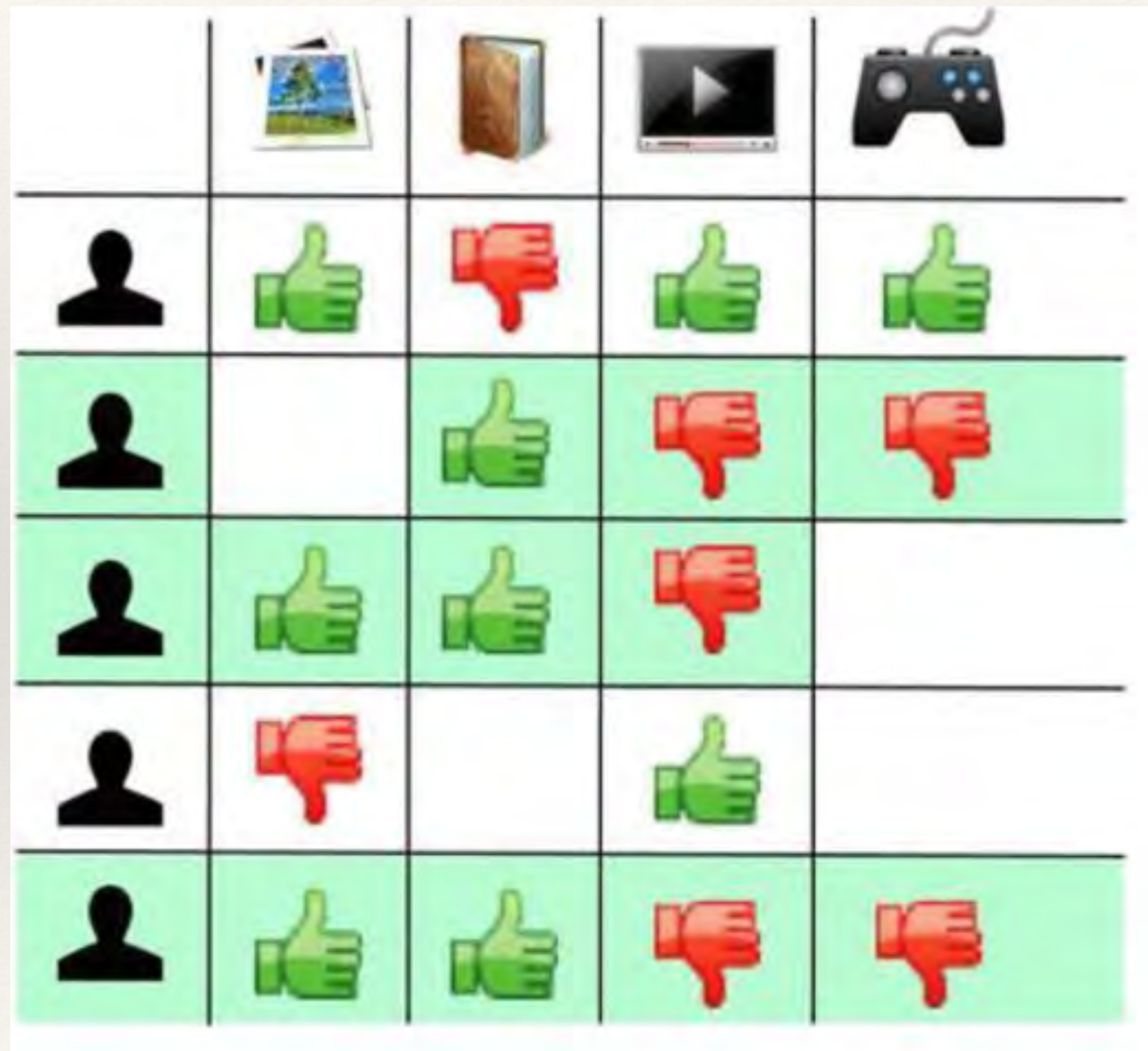
Recommends

Some advantages

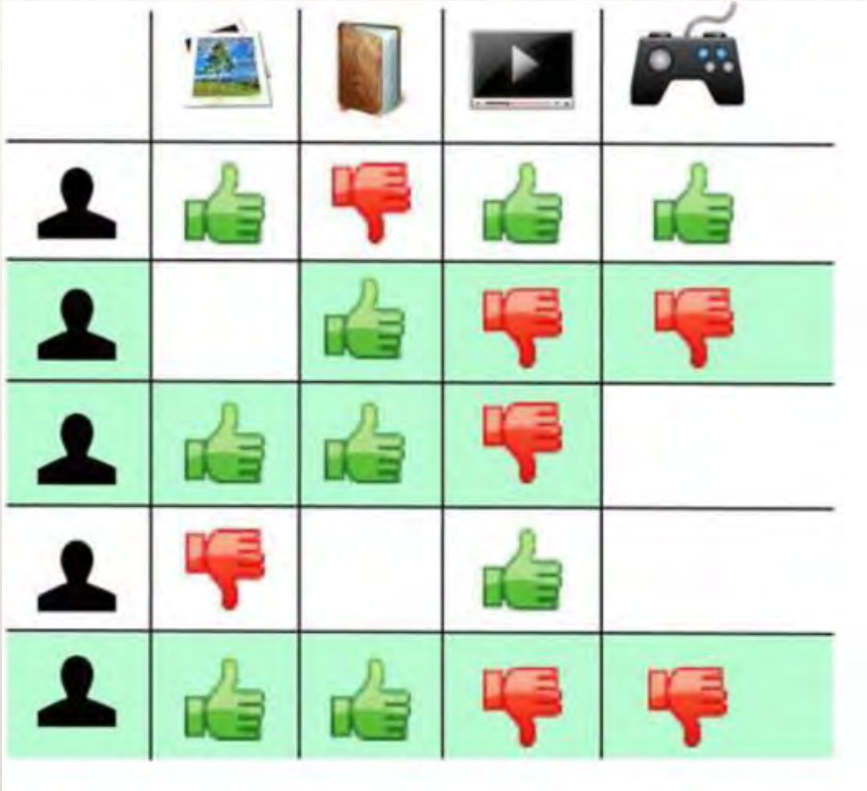
Content-Based: Start from single item

Collaborative: Discover New Things

User (row) Based Collaborative Filtering



Approach: Regression



Pros:

–Reduces recommendations to a well-studied problem

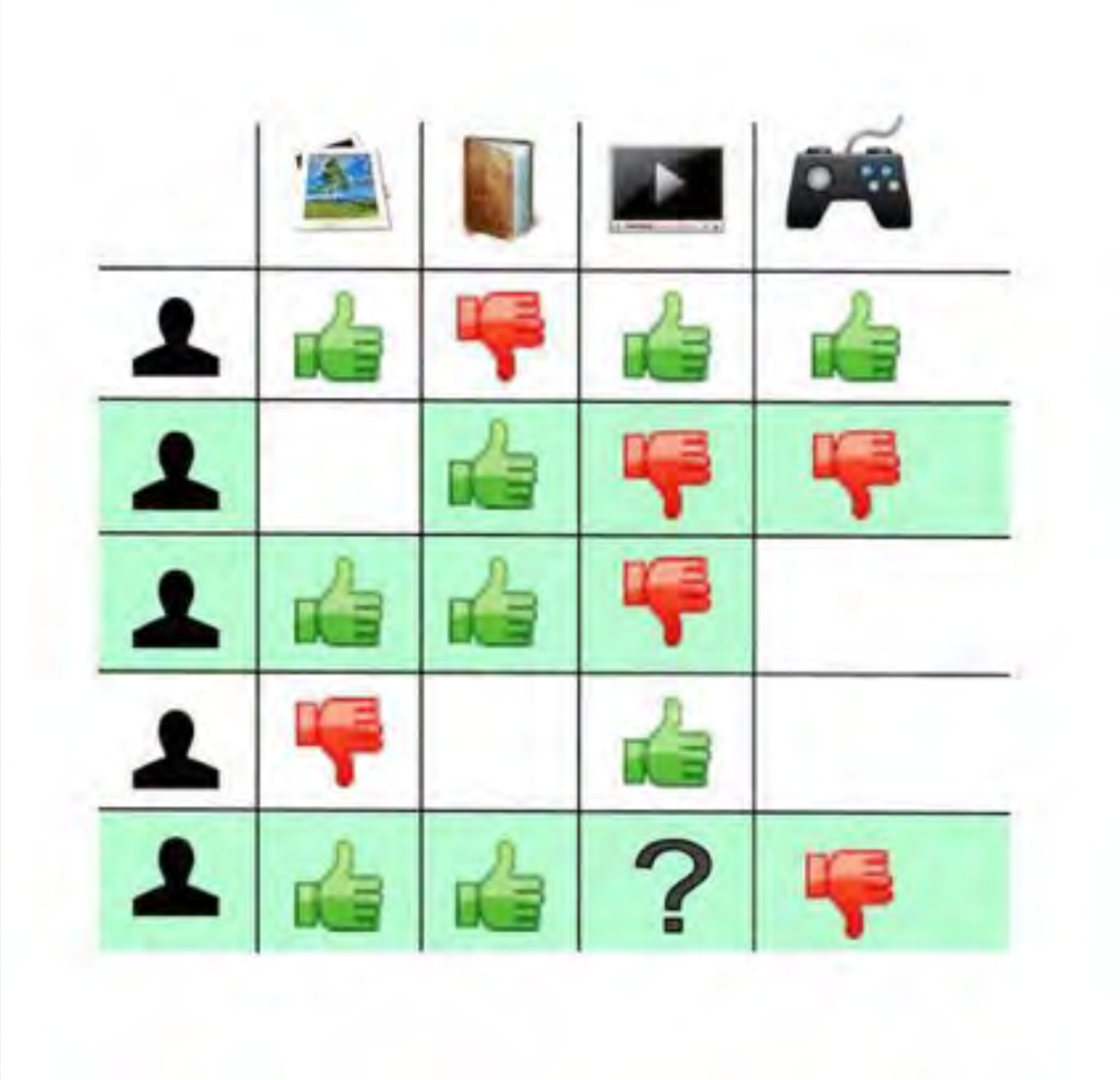
–Many good prediction algorithms available

•Cons:

–Have to handle tons of missing data

–Training M predictors is expensive

KNN



Find neighbor users -> predict user's item rating (average)

Find neighbor items -> predict user's item rating (average)

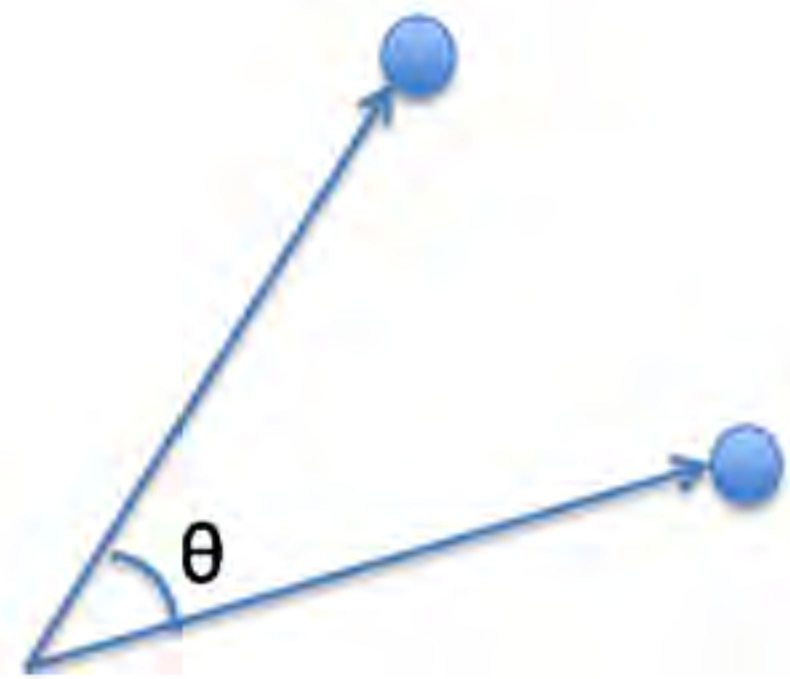
Similarity Score Option: Pearson

$$s(u,v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_u \cap I_v} (r_{v,i} - \bar{r}_v)^2}}$$

Similarity Score Option: Cosine

missing values 0

$$\cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}$$

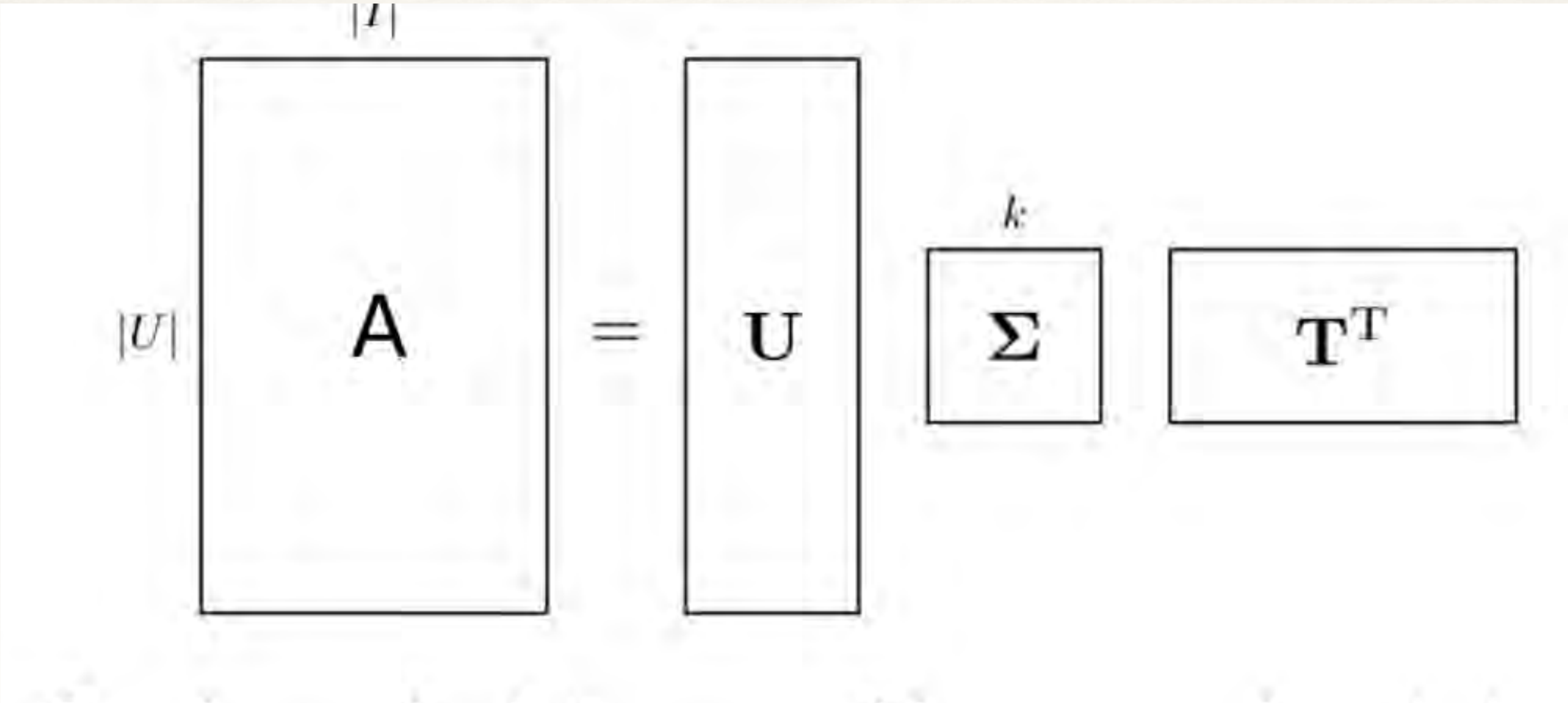


Netflix Prize

SVD Idea

- SimonFunk did this publicly on his blog with the title “Try this at home”
- http://sifter.org/~simon/journal/20061027.2.html

SDV



SVD Method

Leskovec, Rajaraman, Ullman

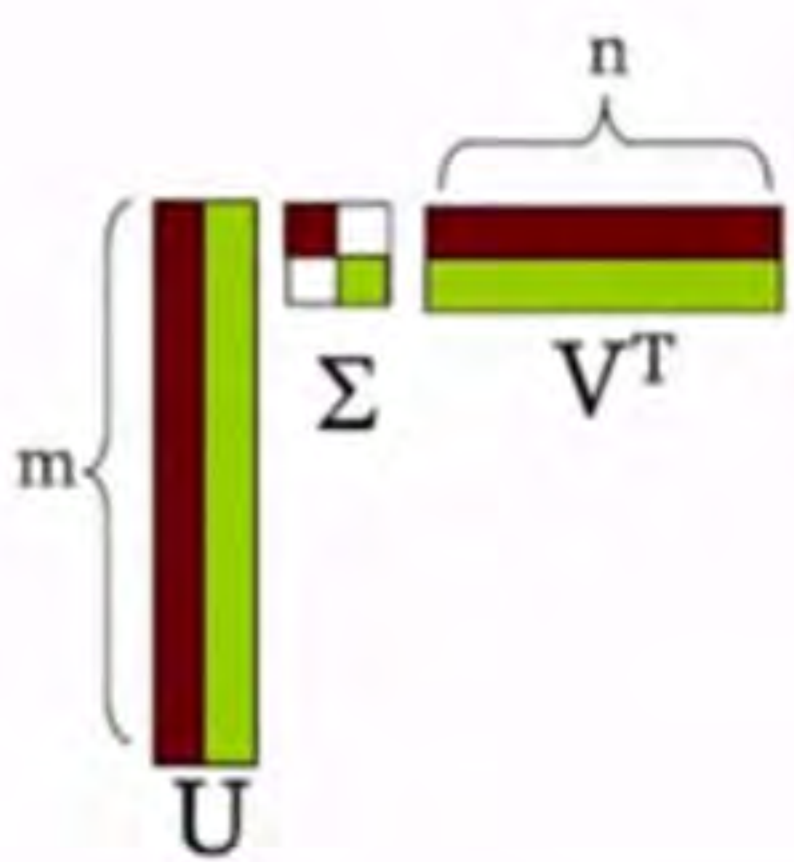
- https://www.youtube.com/watch?v=YKmkAoIUxkU

SVD approach

$A = U \Sigma V^T$ - example: Users to Movies

$$
\begin{array}{ccccc}
\text{Matrix} & \text{Alien} & \text{Serenity} & \text{Casablanca} & \text{Amelie} \\
\end{array}
$$

$$
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 \\
3 & 3 & 3 & 0 & 0 \\
4 & 4 & 4 & 0 & 0 \\
5 & 5 & 5 & 0 & 0 \\
0 & 2 & 0 & 4 & 4 \\
0 & 0 & 0 & 5 & 5 \\
0 & 1 & 0 & 2 & 2
\end{bmatrix}
=
$$



SVD approach

- $A = U \Sigma V^T$ - example: **Users to Movies**

U: Users x Topics

Σ: Topics x Topics

$$
\begin{array}{ccccc}
\text{Matrix} & \text{Alien} & \text{Serenity} & \text{Casablanca} & \text{Amelie}
\end{array}
$$

$$
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 \\
3 & 3 & 3 & 0 & 0 \\
4 & 4 & 4 & 0 & 0 \\
5 & 5 & 5 & 0 & 0 \\
0 & 2 & 0 & 4 & 4 \\
0 & 0 & 0 & 5 & 5 \\
0 & 1 & 0 & 2 & 2
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{0.13} & 0.02 & -0.01 \\
\mathbf{0.41} & 0.07 & -0.03 \\
\mathbf{0.55} & 0.09 & -0.04 \\
\mathbf{0.68} & 0.11 & -0.05 \\
0.15 & \mathbf{-0.59} & \mathbf{0.65} \\
0.07 & \mathbf{-0.73} & \mathbf{-0.67} \\
0.07 & \mathbf{-0.29} & \mathbf{0.32}
\end{bmatrix}
\times
\begin{bmatrix}
\mathbf{12.4} & 0 & 0 \\
0 & \mathbf{9.5} & 0 \\
0 & 0 & \mathbf{1.3}
\end{bmatrix}
\times
\begin{bmatrix}
\mathbf{0.56} & \mathbf{0.59} & \mathbf{0.56} & 0.09 & 0.09 \\
0.12 & -0.02 & 0.12 & \mathbf{-0.69} & \mathbf{-0.69} \\
0.40 & \mathbf{-0.80} & 0.40 & 0.09 & 0.09
\end{bmatrix}
$$

V^T: Topics x Movies

SVD advantages

- Not only good for estimating missing data
- We might actually care about the topics more