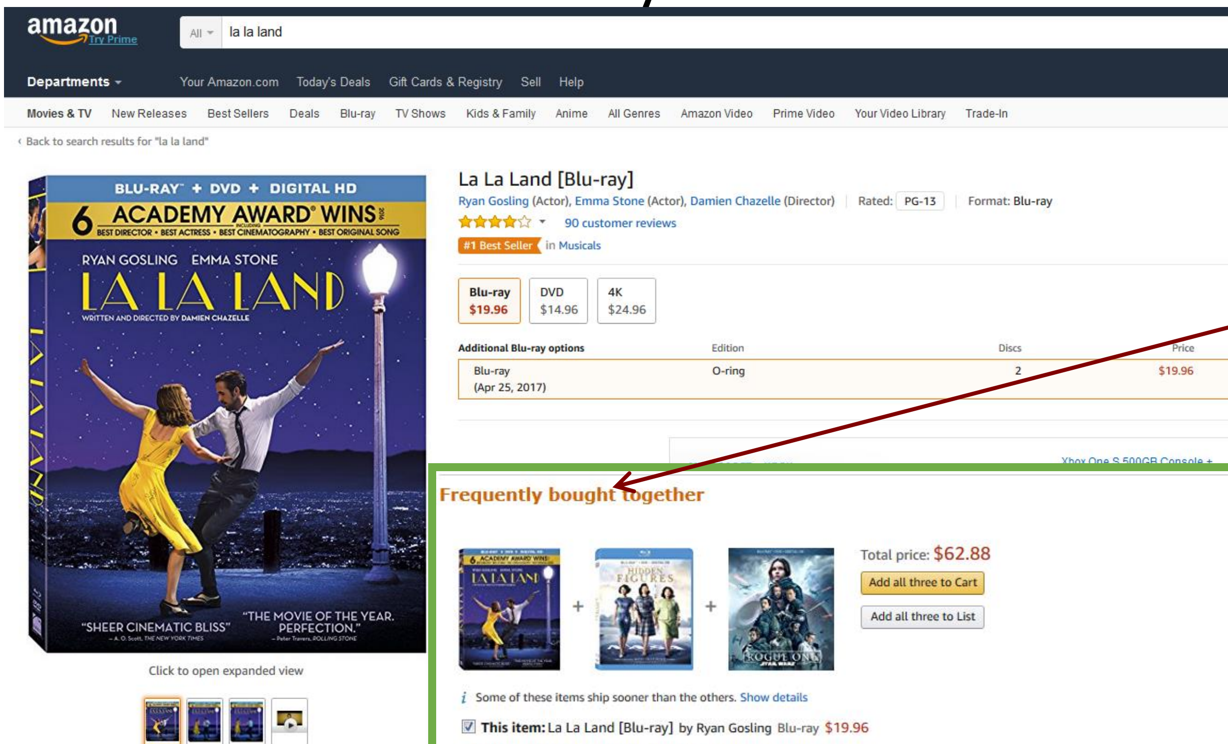
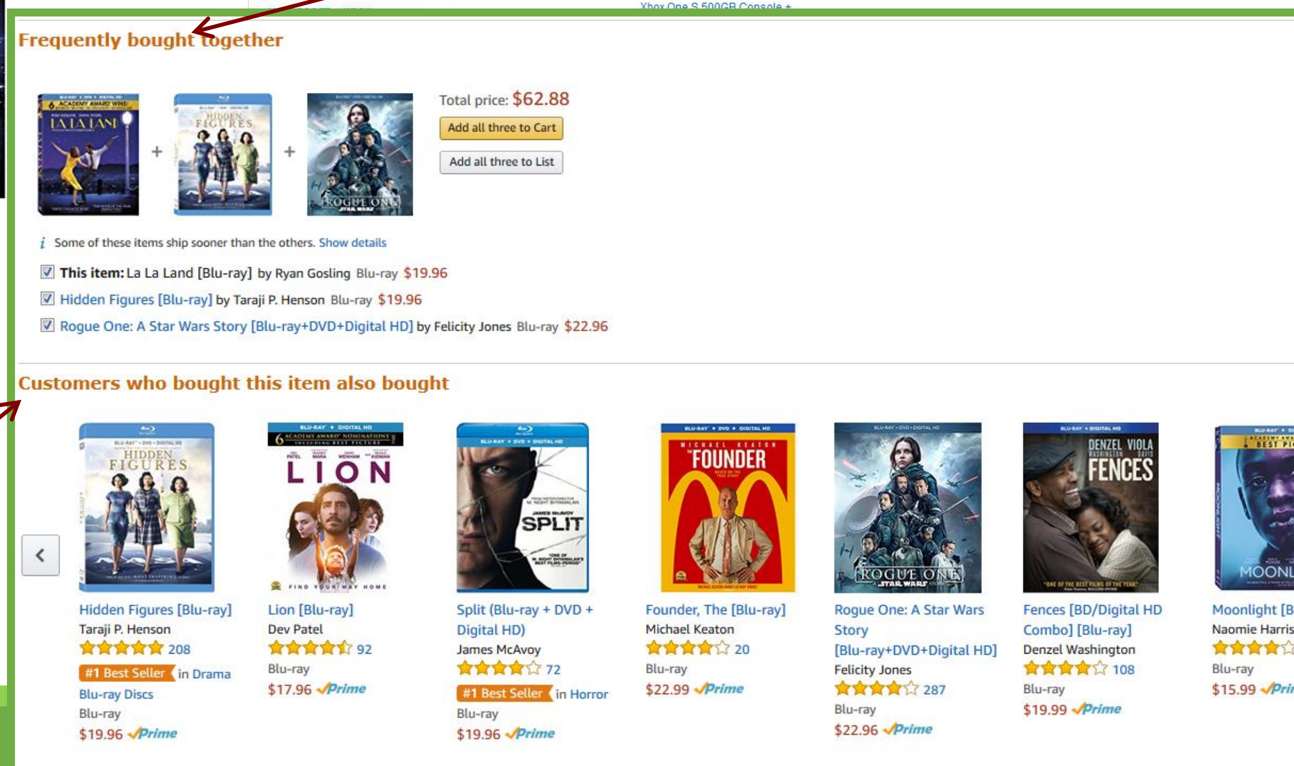
Recommendation Systems

Used in many recommender systems



Recommendations are based on association rules



Recommendations are based on collaborative filtering

What are Association Rules?

- Study of “what goes with what”
  - “Customers who bought X also bought Y”
  - What symptoms go with what diagnosis
- Transaction-based or event-based analysis
- Also called “market basket analysis” and “affinity analysis”

Barbie Doll and Candy

60%

If "Barbie Doll", Then "Candy Bar"

- Put them closer together in the store?
- Put them far apart in the store?
- Package candy bars with the dolls?
- Package Barbie + candy bars + poorly selling item?
- Raise the price on one, and lower it on the other?
- Do not advertise candy and Barbie together?
- Offer candies in the shape of a Barbie doll?

Banking Services Case Study

- Examine associations between various retail banking services used by customers
  - ATM debit card
  - Automobile installment loan
  - Credit card
  - CD
  - Checking account
  - Savings account
  - Home mortgage
  - ……

Healthcare Symptoms and Disease

- Symptoms and illnesses that manifest together
- “We know this usually comes with this, so we’re going to start you on this drug regimen as a precaution”

Tiny Example: iPhone Cases

Transaction	Case Colors Purchased			
1	red	white	green	
2	white	orange		
3	white	blue		
4	red	white	orange	
5	red	blue		
6	white	blue		
7	white	orange		
8	red	white	blue	green
9	red	white	blue	
10	yellow			

Terms

- Itemset
  - A set of items
  - {red, white}, {red, white, green}…..
- Rule
  - IF……THEN……
  - "If {red, white}, then {green}"

    Antecedent Consequent
- Antecedent and consequent are disjoint (i.e., have no items in common)

Many Rules are Possible

- Many Rules are Possible
  - E.g. Transaction 1 supports several rules, such as

Transaction	Case Colors Purchased		
1	red	white	green

    - "If red, then white"
    - "If red and white, then green"
    - ……
- Problem: computation time grows exponentially as # items increases
- Solution: consider only "frequent item sets"

Support

- **Support for an itemset**
  - % of transactions that include an itemset
  - Example: support for the item set {red, white} is 4 out of 10 transactions, or 40%

Transaction	Case Colors Purchased			
1	red	white	green	
2	White	orange		
3	white	blue		
4	red	white	orange	
5	red	Blue		
6	white	Blue		
7	white	Orange		
8	red	white	blue	Green
9	red	white	Blue	
10	yellow			

Measures of Rule Performance

- Confidence
  - % of antecedent transactions that also have the consequent item set
  - Likelihood that consequent will be found in transactions with antecedent
- Lift Ratio
  - Confidence/(Benchmark Confidence)
  - **Benchmark confidence**=% of transactions with consequent to all transactions
  - *S*hows how effective the rule is in finding consequents (i.e., more useful than just selecting transactions randomly)
  - Lift Ratio> 1

Process of Rule Selection

- Generate all rules that meet specified support & confidence
  - Find frequent item sets (those with sufficient support)
  - From these item sets, generate rules with sufficient confidence
  - Assess rule performance using lift ratio

Caution: The Role of Chance

- Random data can generate apparently interesting association rules
- The more rules you produce, the greater this danger
- Rules based on large numbers of records are less subject to this danger

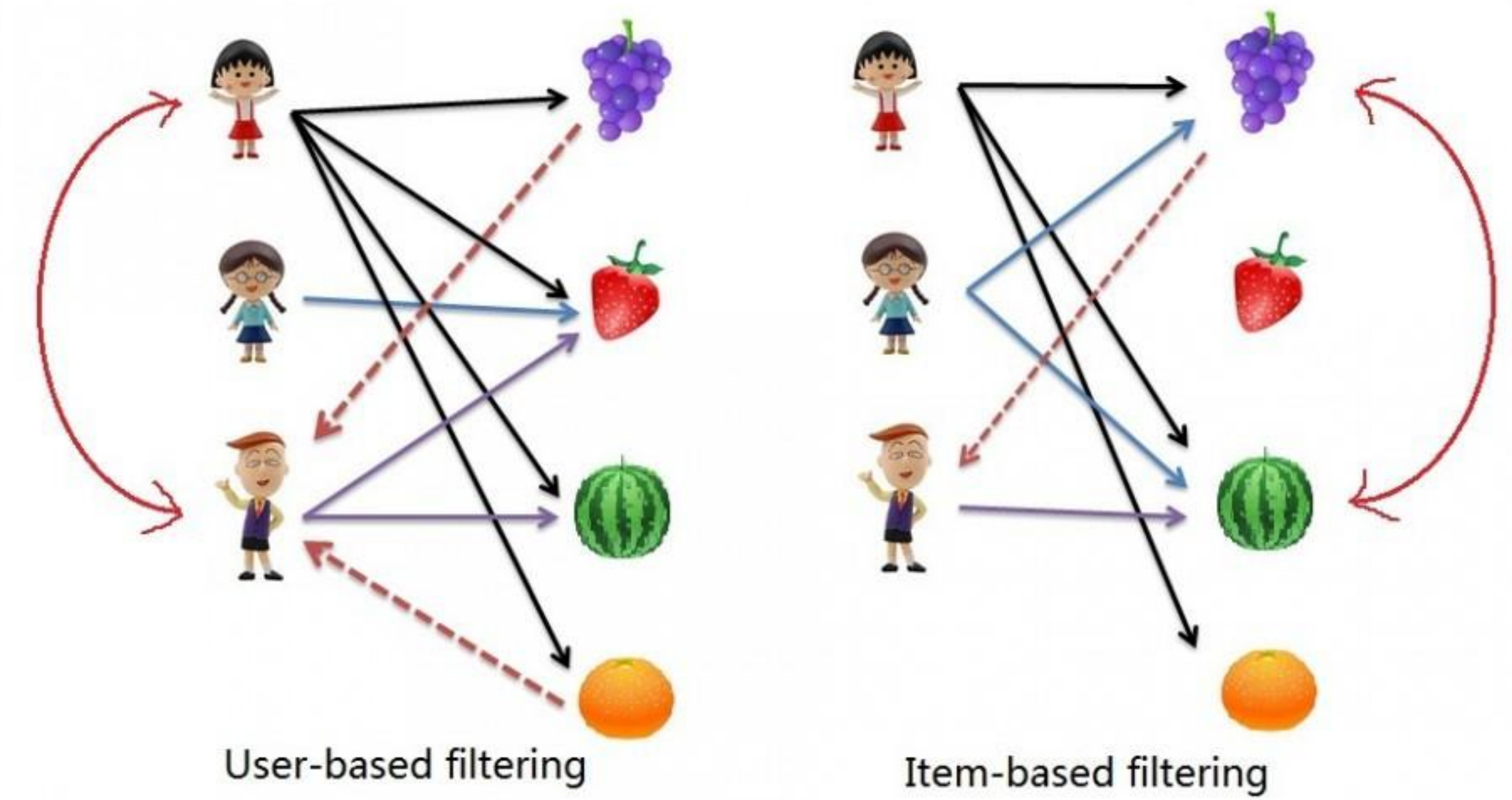
Summary – Association Rules

- Association rules (or *affinity analysis,* or *market basket analysis*) produce rules on associations between items from a database of transactions
- Widely used in **recommender systems**
- Consider only "frequent" item sets (=support) to reduce computation complexity
- Measure performance of rules using *confidence* and *lift*

Collaborative Filtering

- User-based (UBCF)
  - For a user, find other users who share his/her preferences
  - Recommend the highest-rated item that the user does <u>not</u> have.
  - User-user correlations cannot be calculated in advance
  - Slow
- Item-based (IBCF) – for a user considering an item, find other item that is most similar to the item.
  - Ability to calculate item-item correlations in advance greatly speeds up the algorithm

Collaborative Filtering



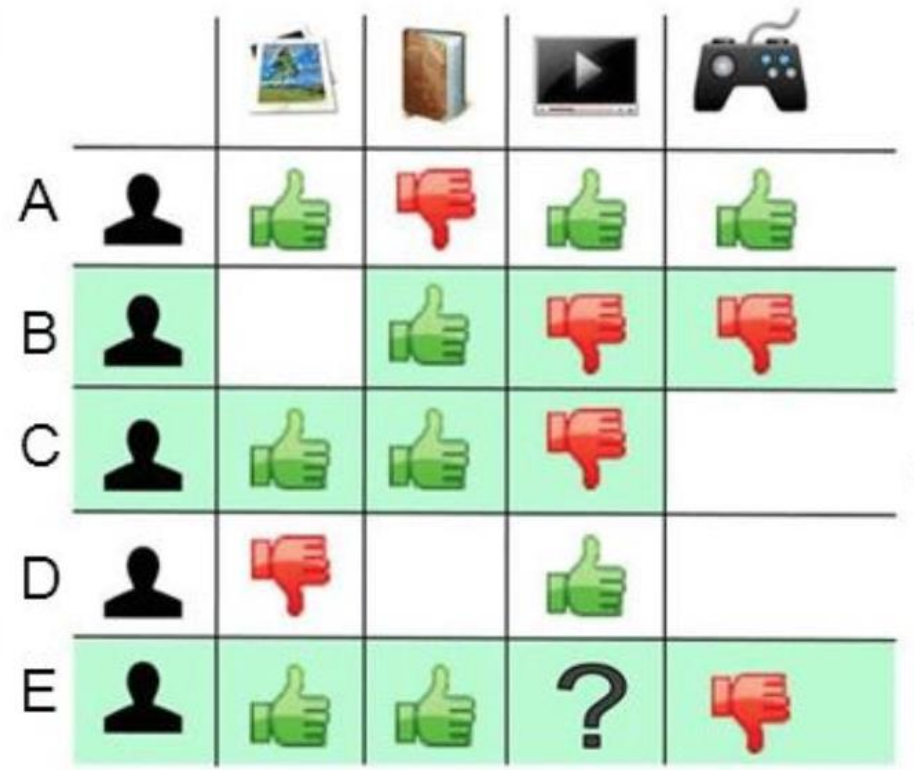
Sample Rating Data

User	Item1	Item2	Item3	Item4	Item5	Item6	…
1	5	5	3	3	2	3	
2	3	2	2	4	1	4	
3	4	3	4	3	3	1	
4	2	4	5	5	4	2	
5	5	1	3	4	1	3	
…							

UBCF

- 1. Measure how similar each user is to the new one using popular similarity measures such as Euclidean and Cosine.
- 2. Next, identify the most similar users with the following options:
  - Calculate the top k users using k-nearest-neighbors
  - Then take account of the users whose similarity is above a defined threshold
- 3. Items purchased by the most similar users are rated with two approaches:
  - Average rating
  - Weighted average rating, using the similarities as weights
- 4. Finally, pick the top-rated items.

UBCF

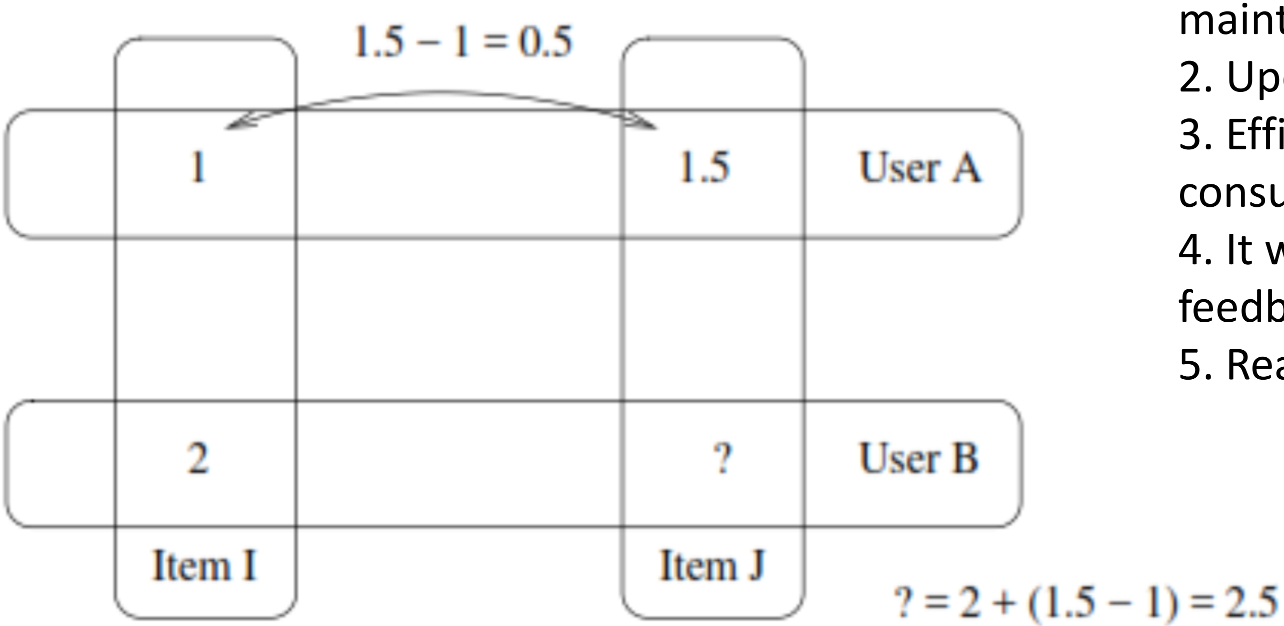


PROS:
•Easy to implement.
•Context independent.
•Accurate (compared to content-based).
CONS:
•Sparsity
•Scalability
•Cold-start
•New item

IBCF

- 1. For two given items, measure how similar they are in terms of having received similar ratings by similar users.
- 2. Next, find the k-most similar items for each given item.
- 3. Finally, for each user, identify the items that are most similar to the user's purchases.

IBCF



Pros

1. Easy to implement and maintain.
2. Updatable online
3. Efficient at the time of consultation
4. It works with little user feedback.
5. Reasonably accurate

Content-Based: Data

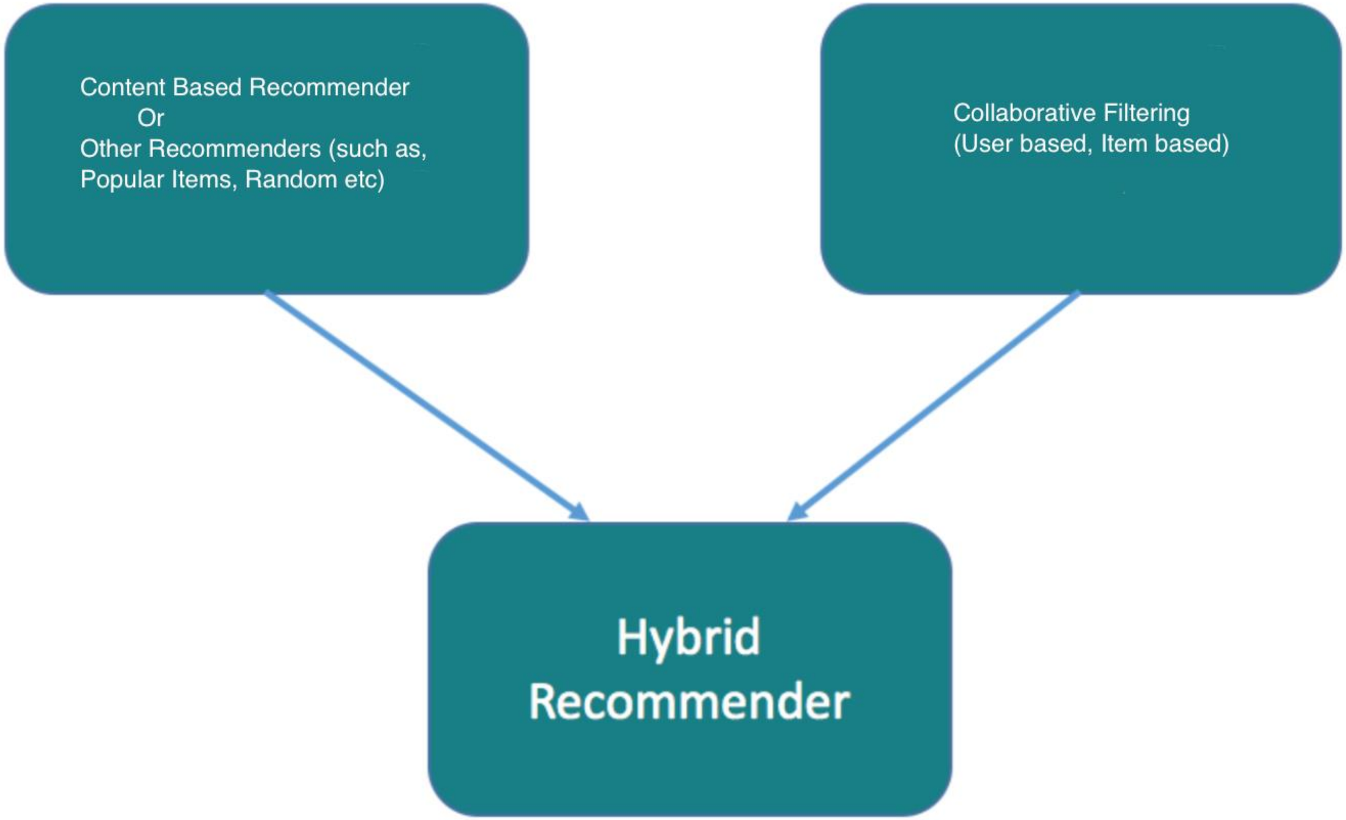
User	Movie	Rating
1	2	5
1	14	3
1	43	4
1	52	3
1	119	1
…		

Movie	Genre1	Genre2	Genre 3	…
1	Drama	Sci-Fi	War	
2	Horror	Sci-Fi	War	
3	Drama	Comedy	Action	
4	Drama	Sci-Fi	Action	
5	Drama	Comedy	Action	
…				

Content-based Recomm: Example

- 1. Cluster the movies based on their genre affiliation using k-means
- 2. Get user information: which movies did s/he watch?
- 3. Map user's movie to cluster
- 4. Get the user's favorite cluster
  - calculate the average movie rating for each cluster by the selected user
  - If the user does not like any cluster, i.e., average rating<3, no recommendation
  - Otherwise, get the cluster with the highest average rating by the user
- 5. Get movies from the user's favorite cluster
  - No "like", randomly pick 100 from the first cluster
  - With "like", pick all movies from that cluster
- 6. Only select those movies that the user has not watched yet
- 7. Only select a specific number of movies from the above list

Hybrid Recommender



Association Rules vs. Collaborative Filtering

- Association rules: focus entirely on frequent (popular) item combinations. Data rows are single transactions. Ignores user dimension. Often used in displays (what goes with what).
- Collaborative filtering: focus is on user preferences. Data rows are user purchases or ratings over time. Can capture "long tail" of user preferences – useful for recommendations involving unusual items