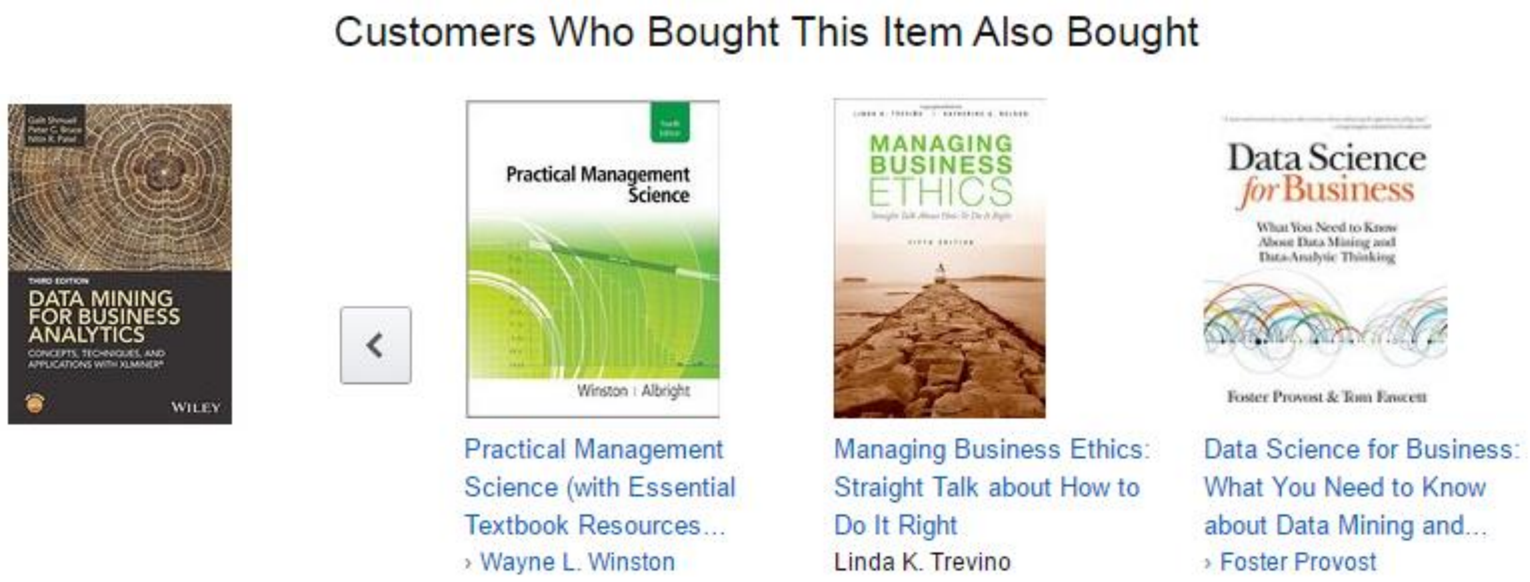
Collaborative Filtering - An Introduction

Collaborative Filtering

- User based methods
- Item based methods



Item-user matrix

- Cells are user preferences, r_{ij}, for items
- Preferences can be ratings, or binary (buy, click, like)

User ID	Item ID I_1	I_2	$\cdots$	I_p
U_1	$r_{1,1}$	$r_{1,2}$	$\cdots$	$r_{1,p}$
U_2	$r_{2,1}$	$r_{2,2}$	$\cdots$	$r_{2,p}$
$\vdots$				
U_n	$r_{n,1}$	$r_{n,2}$	$\cdots$	$r_{n,p}$

More efficient to store as rows of triplets

Each row has the user ID, the item ID, and the user’s rating of that item

(U_u, I_i, r_{ui})

User-based Collaborative Filtering

- Start with a single user who will be the target of the recommendations

- Find other users who are most similar, based on comparing preference vectors

Measuring Proximity

- Like nearest-neighbor algorithm
- But Euclidean distance does not do well
- Correlation proximity does better (Pearson)
- For each user pair, find the co-rated items, calculate correlation between the vectors of their ratings for those items
  - Note that the average ratings for each user are across all products, not just the co-rated ones

$$\mathrm{Corr}(U_1, U_2) = \frac{\sum (r_{1,i} - \overline{r}_1)(r_{2,i} - \overline{r}_2)}{\sqrt{\sum (r_{1,i} - \overline{r}_1)^2} \sqrt{\sum (r_{2,i} - \overline{r}_2)^2}}$$

Example – Tiny Netflix subset

	Movie ID								
Customer ID	1	5	8	17	18	28	30	44	48
30878	4	1			3	3	4	5	
124105	4								
822109	5								
823519	3		1	4		4	5		
885013	4	5							
893988	3						4	4	
1248029	3					2	4		3
1503895	4								
1842128	4						3		
2238063	3								

TABLE 14.8 SAMPLE OF RECORDS FROM THE NETFLIX PRIZE CONTEST, FOR A SUBSET OF 10 CUSTOMERS AND 9 MOVIES

Consider users 30878 and 823519

Correlation between users 30878 and 823519

First find average ratings for each user:

$$\overline{r}_{30878} = (4 + 1 + 3 + 3 + 4 + 5)/6 = 3.333$$

$$\overline{r}_{823519} = (3 + 1 + 4 + 4 + 5)/5 = 3.4$$

Find correlation using departure from avg. ratings for the co-rated movies (movies 1, 28 and 30):

$$\text{Corr}(U_1, U_2) = \frac{\sum(r_{1,i} - \overline{r}_1)(r_{2,i} - \overline{r}_2)}{\sqrt{\sum(r_{1,i} - \overline{r}_1)^2}\sqrt{\sum(r_{2,i} - \overline{r}_2)^2}}$$

$$\text{Corr}(U_{30878}, U_{823519}) = \frac{(4 - 3.333)(3 - 3.4) + (3 - 3.333)(4 - 3.4) + (4 - 3.333)(5 - 3.4)}{\sqrt{(4 - 3.333)^2 + (3 - 3.333)^2 + (4 - 3.333)^2}\sqrt{(3 - 3.4)^2 + (4 - 3.4)^2 + (5 - 3.4)^2}}$$

$$= 0.6/1.75 = 0.34$$

	Movie ID								
Customer ID	1	5	8	17	18	28	30	44	48
30878	4	1			3	3	4	5	
124105	4								
822109	5								
823519	3		1	4		4	5		
885013	4	5							
893988	3						4	4	
1248029	3					2	4		3
1503895	4								
1842128	4						3		
2238063	3								

TABLE 14.8 SAMPLE OF RECORDS FROM THE NETFLIX PRIZE CONTEST, FOR A SUBSET OF 10 CUSTOMERS AND 9 MOVIES

Cosine Similarity

Like correlation coefficient, except do not subtract the means

Use raw ratings instead of departures from averages

$$\text{Cos Sim}(U_{30878}, U_{823519}) = \frac{4 \times 3 + 3 \times 4 + 4 \times 5}{\sqrt{4^2 + 3^2 + 4^2}\sqrt{3^2 + 4^2 + 5^2}}$$

$$= 44/45.277 = 0.972$$

Customer ID	Movie ID 1	5	8	17	18	28	30	44	48
30878	4	1			3	3	4	5	
124105	4								
822109	5								
823519	3		1	4		4	5		
885013	4	5							
893988	3						4	4	
1248029	3					2	4		3
1503895	4								
1842128	4						3		
2238063	3								

TABLE 14.8 SAMPLE OF RECORDS FROM THE NETFLIX PRIZE CONTEST, FOR A SUBSET OF 10 CUSTOMERS AND 9 MOVIES

Ranges from 0 (no similarity) to 1 (perfect match)

Using the similarity info to make recommendations

- Given a new user, identify k-nearest users
- Consider all the items they rated/purchased, except for the co-rated ones
- Among these other items, what is the best one? "Best" could be
  - Most purchased
  - Highest rated
  - Most rated
- That "best" item is the recommendation for the new user

Cold Start

- Collaborative filtering suffers from what is called a *cold start*: it cannot be used as is to create recommendations for new users or new items.
- For a user who rated a single item, the correlation coefficient between this and other users (in user-generated collaborative filtering) will have a denominator of zero and the cosine proximity will be 1 regardless of the rating.
- In a similar vein, users with just one item, and items with just one user, do not qualify as candidates for nearby neighbors

Item-based collaborative filtering

- When the number of users is huge, user-based calculations pose an obstacle (similarity measures cannot be calculated until user shows up)
- Alternative – when a user purchases an item, focus on similar items

1. Find co-rated (co-purchased) items (by any user)
2. Recommend the most popular or most correlated item

Item-based collaborative filtering

- Similarity is now computed between items, instead of users. For example, in the Netflix sample, the correlation between movie 1 (with average $r1 = 3.7$) and movie 5 (with average $r5 = 3$) is:

$$\mathrm{Corr}(I_1, I_5) = \frac{(4-3.7)(1-3)+(4-3.7)(5-3)}{\sqrt{(4-3.7)^2+(4-3.7)^2}\sqrt{(1-3)^2+(5-3)^2}} = 0$$

	Movie ID								
Customer ID	1	5	8	17	18	28	30	44	48
30878	4	1			3	3	4	5	
124105	4								
822109	5								
823519	3		1	4		4	5		
885013	4	5							
893988	3						4	4	
1248029	3					2	4		3
1503895	4								
1842128	4						3		
2238063	3								

TABLE 14.8 SAMPLE OF RECORDS FROM THE NETFLIX PRIZE CONTEST, FOR A SUBSET OF 10 CUSTOMERS AND 9 MOVIES

- Thus we can compute similarity between all the movies.
- This can be done offline.
- In real time, for a user who rates a certain movie highly, we can look up the movie correlation table and recommend the movie with the highest positive correlation to the user's newly rated movie.

Summary – Collaborative Filtering

- User-based – for a new user, find other users who share his/her preferences, recommend the highest-rated item that new user does not have.
  - User-user correlations cannot be calculated until new user appears on the scene… so it is slow if lots of users
- Item-based – for a new user considering an item, find other item that is most similar in terms of user preferences.
  - Ability to calculate item-item correlations in advance greatly speeds up the algorithm
  - The disadvantage of item-based recommendations is that there is less diversity between items (compared to users' taste), and therefore, the recommendations are often obvious.

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).
- Binary Data
- Two or more items

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
- Binary as well as Ratings data
- Between pairs of items or users

Slide Contents - References

- The contents of this presentation were sourced and assembled from
  - Data Mining for Business Analytics: Concepts, Techniques and Applications in R, by Galit Shmueli et al., Wiley India, 2018.