Lecture 6: Recommendation Systems

Advanced Business Analytics (CIS442D)
Simon Business School
2/8/2017

Based on Chapter 13 in "The analytics edge" by Bertsimas, O`Hair, and Pulleyblank

1

Outline

- Applications
- Traditional methods
  - Collaborative filtering
  - Content-based filtering
- Matrix factorization

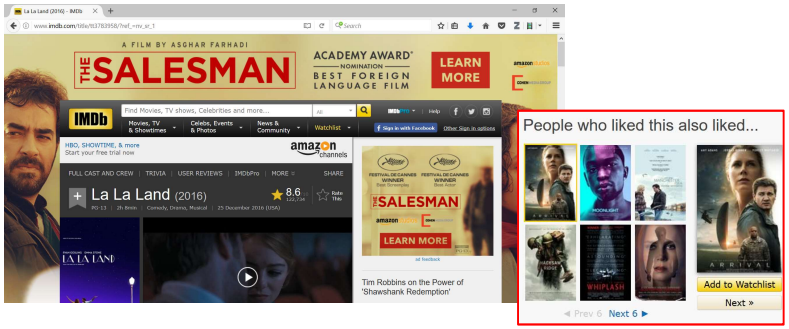
2

Recommendation/Recommender systems

- Tools and techniques to provide suggestions for **items** to **users**

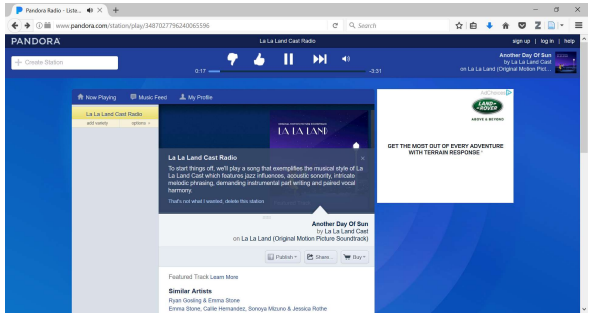
3

Movies



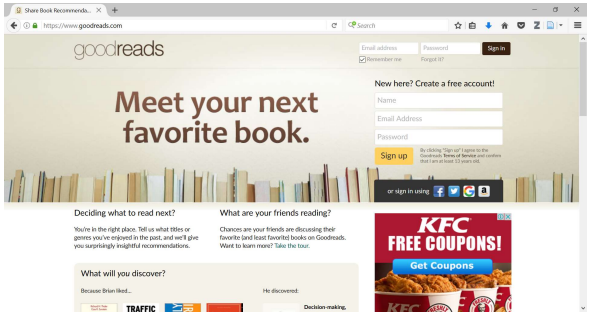
4

Music



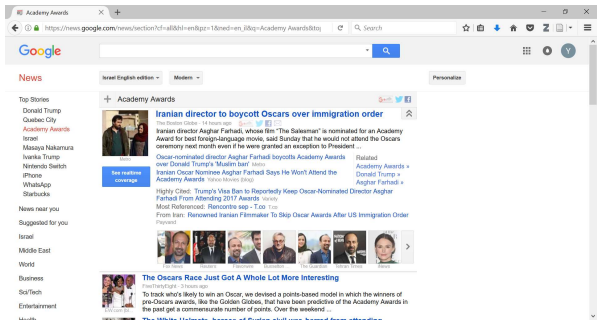
5

Books



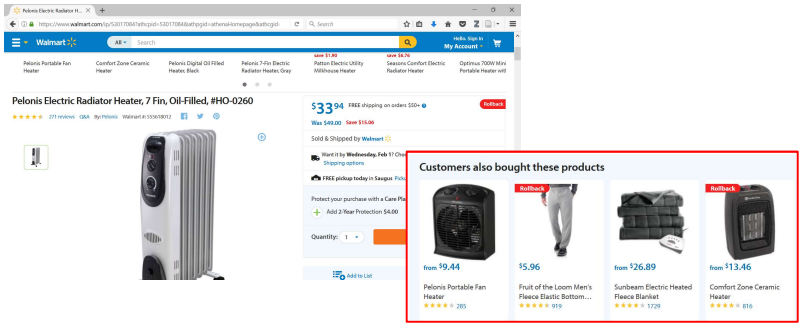
6

News



7

E-commerce



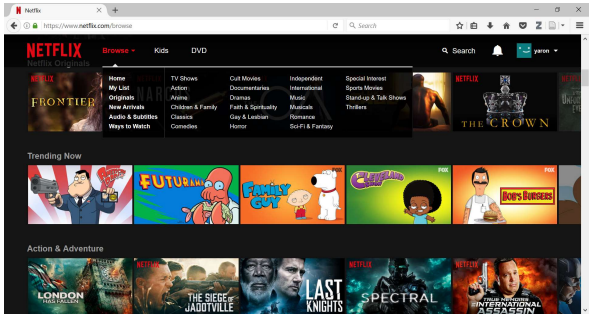
8

Recommendation systems

- Tools and techniques to provide suggestions for **items** to **users**
- Applications
  - Entertainment – Hulo, Netflix, Pandora, Spotify, YouTube, …
  - Content – news websites, e-learning (coursera), …
  - E-commerce – online retailing (Amazon, Wallmart), …
  - Services – travel (Expedia), expert consultations (Yelp), …
  - Social media – Facebook, LinkedIn, Google+, …
- Help customers find items they are interested in
- Help companies find potential customers
  - (Arguably) accounts for 35% of Amazon revenue
  - The Netflix challenge

9

Netflix



10

Netflix

- 1997: founded, offers movie rentals by mail
- 1998: launched website
- 2000: Introduced the Cinematch recommendation system
  - Drives sales (60% of subscribers add recommended movies)
  - Keeps larger part of the library in circulation
  - Based on linear models
  - 75% of prediction within 1/2 a star
  - Half of Netflix users give 5 star to recommended movies

11

Netflix – cont.

- 2006: launched the “Netflix prize”
  - Improve prediction accuracy by 10%
  - Prize: 1M dollar
  - 100M ratings, 480K users, 18K movies
  - Anonymized data (user IDs)
- 2007: video streaming
- 2009: winner announced
  - Combination of techniques including nearest neighbors and matrix factorization
- 2016: Revenue 8.83B

12

Recommendation systems – requirements

- Utilize large data sets
- Real-time
- Accurate (or could hurt customer satisfaction)
- Personalized
- Work well with new and existing users

13

Example: movie recommendation

- Historical data
  - Movie ratings by users (1-5)
  - Example: (Amy, Inception, 4), …, (Bob, Forest Gump, 5)
- Objective: predict Eva's rating for Inception
- Two main types of recommendation systems
- Collaborative filtering:
  - Recommend based on **user** attributes
  - Find similar people to Eva
- Content-based filtering:
  - Recommend based on **item** attributes
  - Find similar movies to those Eva likes

14

Collaborative filtering

- Objective: predict movie rating of a user
- Basic idea
  - Represent users as a vector of items
  - Compute similarity measure between users
  - Return the weighted average of ratings of other users who rated the movie

	Forest Gump (F)	Godfather (G)	Inception (I)	Jaws (J)
Amy (A)	5		4	3
Bob (B)	3	5	2	5
Carl (C)		3	5	4
Dan (D)	4	5	4	
Eva (E)	4	4		3

15

Measuring similarities

- How similar are Amy and Bob?

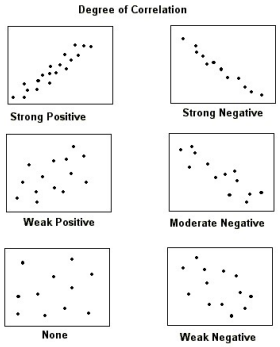
	Forest Gump (F)	Godfather (G)	Inception (I)	Jaws (J)
Amy (A)	5		4	3
Bob (B)	3	5	2	5

- Correlation
  - Compute $\mu_A = \frac{5+4+3}{3} = 4, \;\; \mu_B = \frac{3+5+2+5}{4} = 3.75$
  - Compute $\sigma_A = \sqrt{\frac{1}{3}\left((5-4)^2 + (4-4)^2 + (3-4)^2\right)} = 0.82, \;\; \sigma_B = 1.3$
  - $S_{A,B} = \frac{\frac{1}{n}\sum_{i=1}^{n}(r_{A,i}-\mu_A)(r_{B,i}-\mu_B)}{\sigma_A,\sigma_B} = \frac{\frac{1}{3}[(5-4)(3-3.75)+(4-4)(2-3.75) \;\; (3-4)(5-3.75)]}{0.82 \cdot 1.3} = -0.63$

* Many other types of similarity measures are used in practice. There are also variants of this method which results in an unbiased estimators, and which computes the standard deviation based on the items (movies) rated by both customers.

Using correlation for prediction

- Correlation: a measure of the linear relation between two variables
- Negative/positive
- Not slope



17

Making prediction

- Similarity matrix
- Prediction:

	Amy (A)	Bob (B)	Carl (C)	Dan (D)	Eva (E)
Amy (A)	1	-0.63	0	-0.43	1.3
Bob (B)	-0.63	1	-0.94	0.91	-0.37
Carl (C)	0	-0.94	1	-1.3	-0.43
Dan (D)	-0.43	0.91	-1.3	1	0.25
Eva (E)	1.3	-0.37	-0.43	0.25	1

$$P_{E,I} = \mu_E + \frac{\sum_{u \in U_I} S_{u,I} \left(r_{u,I} - \mu_u\right)}{\sum_{u \in U_I} S_{u,I}}$$

$$= 3.67 + \frac{1.3 \cdot (4-4) - 0.37 \cdot (2-3.75) - 0.43 \cdot (5-4) + 0.25 \cdot (4-4.33)}{1.3 - 0.37 - 0.43 + 0.25}$$

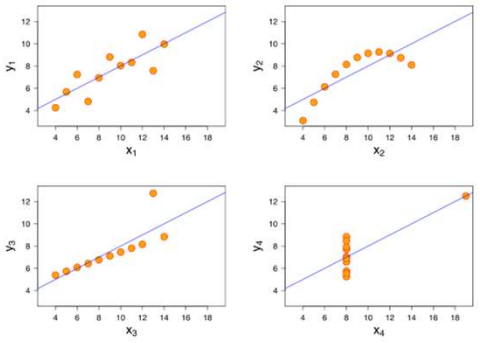
$$= 3.85$$

	Inception (I)	μ	σ	$S_{E,I}$
Amy (A)	4	4	0.82	1.3
Bob (B)	2	3.75	1.30	-0.37
Carl (C)	5	4	0.82	-0.43
Dan (D)	4	4.333	0.47	0.25
Eva (E)		3.667	0.47	1

18

Anscombe's quartet

- Use mean, standard deviation, and correlation to fit linear model
- The data in the figure shares the same
  - Mean
  - Standard deviation
  - Correlation
- Reference [link]



19

Collaborative filtering

- Pros
  - Works well in practice
  - Independent of particular domain
- Cons
  - Computationally intensive
  - "Cold start" problem
    - Requires some amount of information to find similar users
    - First rate problem

20

Content-based filtering

- Objective: predict movie rating of a user
- Basic idea
  - Rank based on similar **items**

		Forest Gump (F)	Godfather (G)	Inception (I)	Jaws (J)
Common to all movies	Movie length	…	…	…	…
	Genre	…	…	…	…
	Main actors	…	…	…	…
	Year	…	…	…	…
	Director	…	…	…	…
Specific to user	Rating (**Ema**)	4	4		3

  - Compute similarity between **movies**
  - Predict: $P_{E,I} = \frac{\sum_{i \in Ite}{}_{E} s_{I,i} r_{E,i}}{\sum_{i \in Items_E} s_{I,i}}$

Content-based filtering – cont.

- Alternatives
  - Only use neighbors
  - Construct a predictive model based on users ratings

	Forest Gump (F)	Godfather (G)	Jaws (J)	Inception (I)
Movie length	…	…	…	…
Genre	…	…	…	…
Main actors	…	…	…	…
Year	…	…	…	…
Director	…	…	…	…
Rating	4	4	3	

Training data

  - Clustering

Content-based filtering

- Pros
  - Can generate recommendation with a single data point
  - Transparent – can infer criteria for recommendation
- Cons
  - Requires domain knowledge
    - What are the important features
    - Extract automatically features for every item
  - Over-specialization: hard to make recommendations for items not purchased before

Matrix factorization

- Represent items (movies) using characteristics

	The King's Speech	Pulp Fiction
Amount of violence (1-5)	1	5
Drama/comedy (0-1)	0.1	0.5
Popularity of the cast (1-10)	8	9

- Represent users using the same characteristics

	User 1
Drama/comedy	0.9
Amount of violence	0.1
Popularity of the cast	0.75

- Rating of user 1 for "The King's Speech": $1 \cdot 0.9 + 0.1 \cdot 0.1 + 8 \cdot 0.75 = 6.91$

Matrix factorization

- How to find coefficients at the right scale that also fit ratings?

	The King's Speech	Pulp Fiction
Amount of violence (1-5)	a_1	b_1
Drama/comedy (0-1)	a_2	b_2
Popularity of the cast (1-10)	a_3	b_3

	User 1
Amount of violence	u_1
Drama/comedy	u_2
Popularity of the cast	u_3

- All we know is

	User 1
The King's Speech	3
Pulp Fiction	5

- Solve:
  - $a_1 \cdot u_1 + a_2 \cdot u_2 + a_3 \cdot u_3 = 3$
  - $b_1 \cdot u_1 + b_2 \cdot u_2 + b_3 \cdot u_3 = 5$

25

Matrix factorization

	Forest Gump (F)	Godfather (G)	Inception (I)	Jaws (J)
Amy (A)	5		4	3
Bob (B)	3	5	2	5
Carl (C)		3	5	4
Dan (D)	4	5	4	
Eva (E)	4	4		3

Roughly

	X_1	X_2
Amy (A)		
Bob (B)		
Carl (C)		
Dan (D)		
Eva (E)		

×

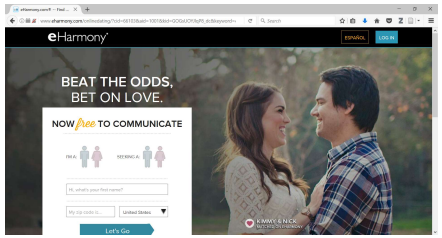
	Forest Gump (F)	Godfather (G)	Inception (I)	Jaws (J)
X_1				
X_2				

- Example: SVD (See Section 5.3 in [2])

26

eHarmony

- Online dating site for long term relationships
- Over 20M users
- 4% of the marriages in the US in 2012 were a result of eHarmony
- "opposites attract, then they attack"
- Users fill questionnaire with 400+ questions about characteristics, beliefs, values, emotional health and skills
- Users only get access to photos the algorithm proposed



27

Other considerations and challenges

- Computational – algorithms should work with real-world datasets
- Transparency
- Privacy
  - Personal recommendations are based on user data
  - 2nd Netflix competition
  - Target [link]
- Diversity of items – discovery of items in the early recommendation stage
- Exploration – exploitation
- Context
- Social networks
- Group recommendations
- Robustness
- Unique domain characteristics

28

References

1. The analytics edge / Bertsimas, O`Hair, and Pulleyblank
2. Introduction to Recommender Systems Handbook / Ricci, Rokach and Shapira
3. The BellKor Solution to the Netflix Grand Prize / Koren

29