Recommender Systems

Statistical Data Mining II

Spring 2016

Rachael Hageman Blair

Outline

- Introduction
- Nearest Neighbor Methods
- Terminology
- Collaborative Filtering
- Conclusions

- Thinking points

What is Recommender System (Rec Sys)

- Take a guess?

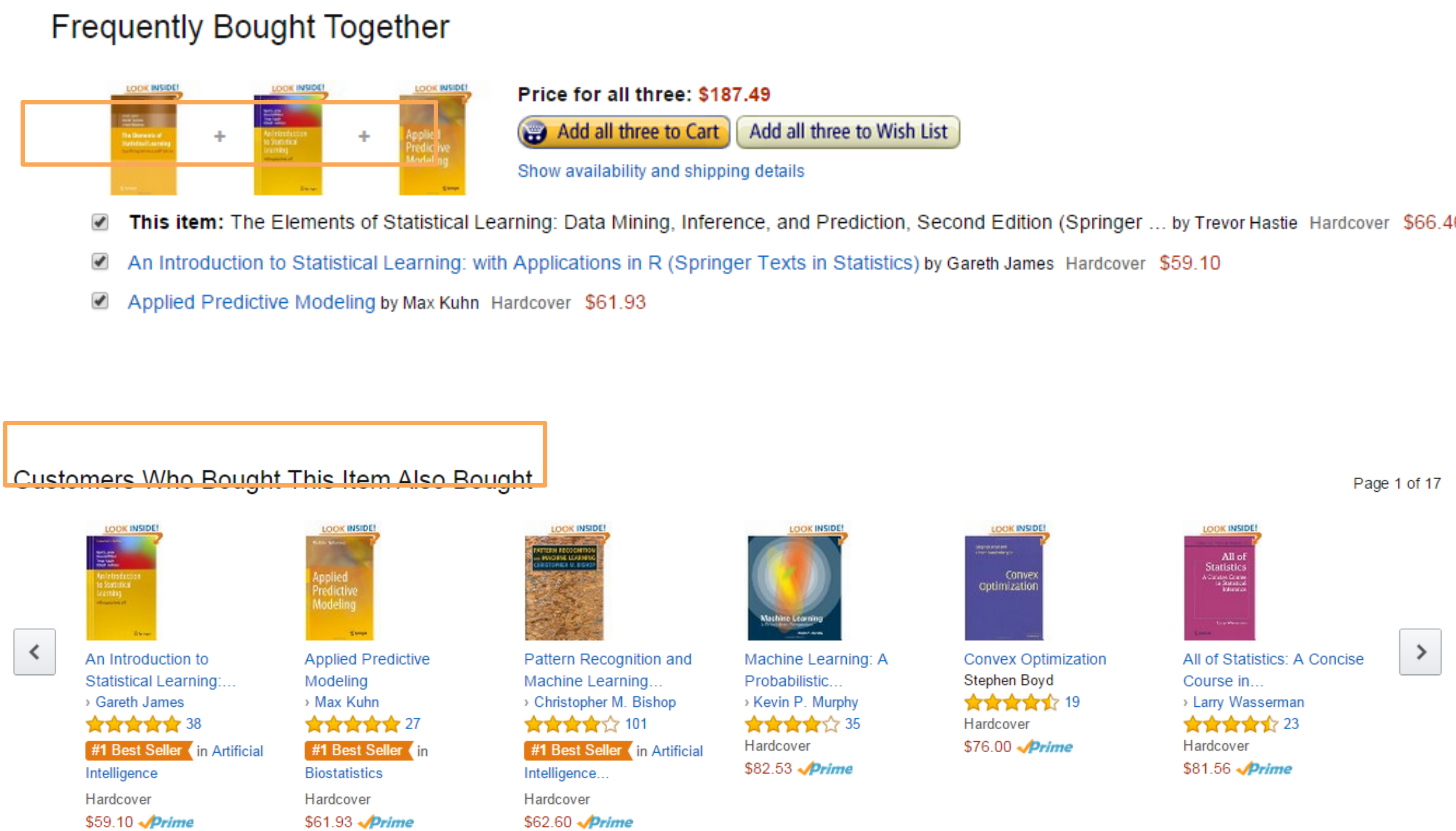
> Yes! Rec Sys is a method designed to make recommendations!

- There are many forms of recommendation system.

- Example: Amazon, News, Blogs, Friends.

Non-personalized recommendations

Personalized recommendations



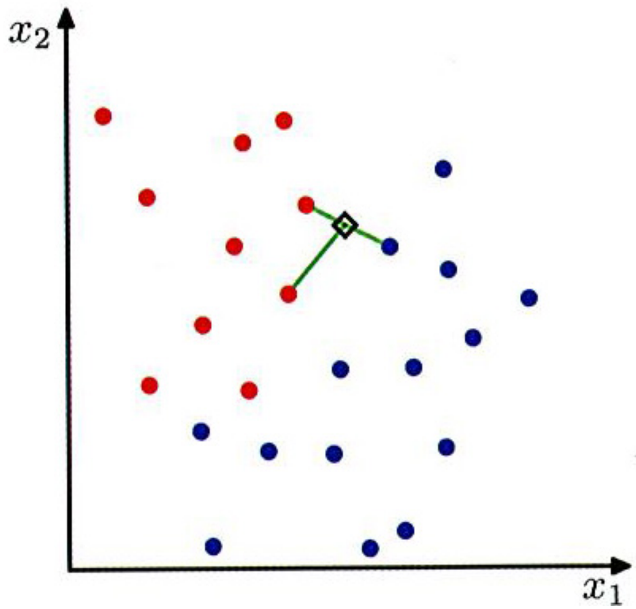
http://www.amazon.com/Elements-Statistical-Learning-Prediction-Statistics/dp/0387848576/ref=sr_1_1?s=books&ie=UTF8&qid=1416592593&sr=1-1&keywords=elements+of+statistical+learning

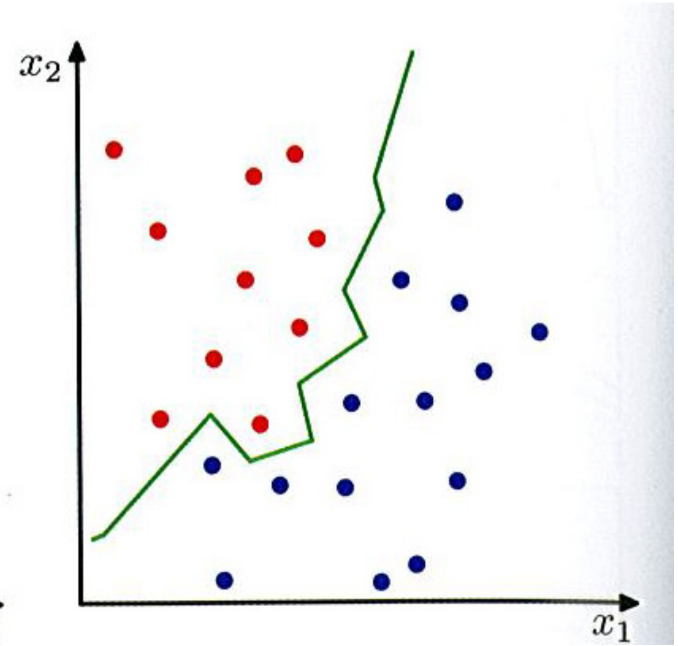
Power of Recommendation Systems

- Netflix: 2/3 of the movies watched are recommended
- Google News: recommendations generate 38% more clickthrough
- Amazon: 35% sales from recommendations

*new wave of competitions…..
Netflix – 1 million dollar prize to beat algorithm by 10%.

Nearest-neighbor methods

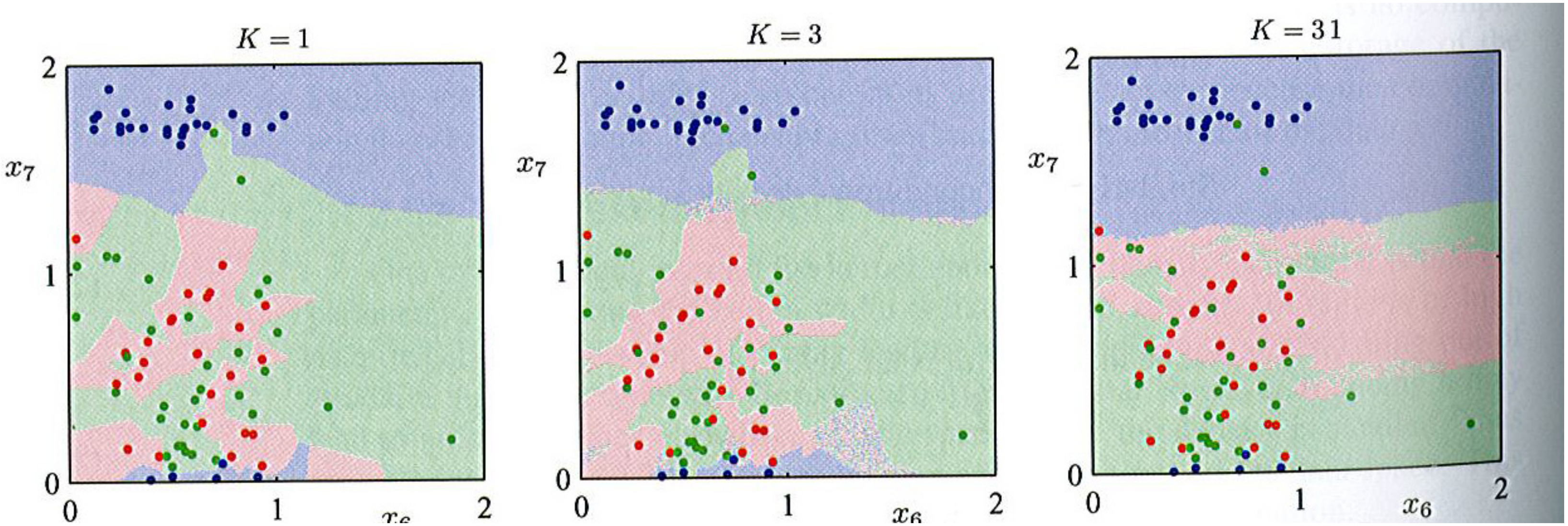




A new point arrives, it is classified according to the majority class membership of its K closest neighbors.

When K=1, the decision boundary is a hyper-plane that form perpendicular bisectors for pairs of points from different classes.

Nearest-neighbor methods



K - pertains to the fit. The k- nearest neighbor fit for $\hat{Y}(x)$ is defined as follows:

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i.$$

Neighborhood of x defined by the k closes points x_i.

Small K - many small regions.
Larger K - fewer large regions.

Terminology

- Content-based systems: examine properties of the items recommended. For instance, if a netflix user watches a lot of westerns, then a movie recommended related to "westerns" may be suggested.

- Collaborative filtering schemes: recommend items based on similarity between users and/or items. The items recommended to a user are those that are preferred by similar users.

Terminology

- Utility Matrix – the data is represented as user-item pairs. Entries of the matrix reflect a given "rating" of the user on the item. Values come from an ordinal set.

	Harry Potter			Twilight	Star Wars		
	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- Data is large and sparse.
- Goal: Predict the empty spaces that would be highly rated for a person.

Terminology

	Harry Potter			Twilight	Star Wars		
	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

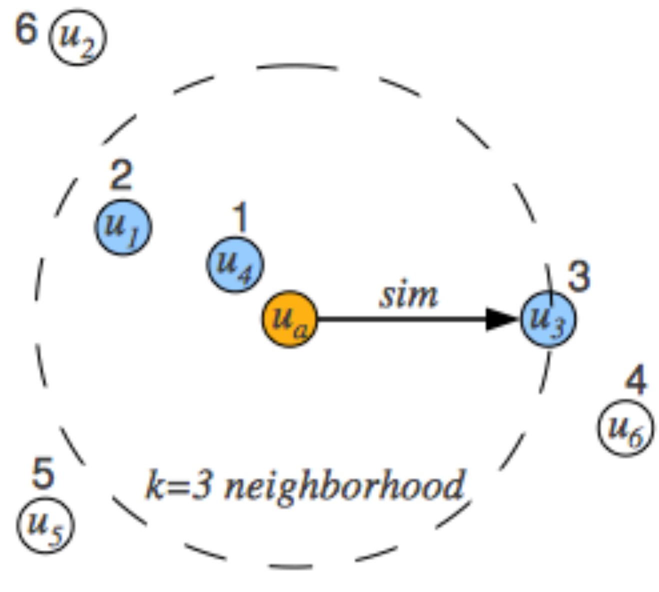
<u>How can we populate the utility matrix?</u>

1. Ask people to rate items they buy/watch/read.
2. Make inferences based on users' behavior – includes purchasing, watching, adding to cart, or even clicking. They are watching your every click!

Terminology

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	?	4.0	4.0	2.0	1.0	2.0	?	?
u_2	3.0	?	?	?	5.0	1.0	?	?
u_3	3.0	?	?	3.0	2.0	2.0	?	3.0
u_4	4.0	?	?	2.0	1.0	1.0	2.0	4.0
u_5	1.0	1.0	?	?	?	?	?	1.0
u_6	?	1.0	?	?	1.0	1.0	?	1.0
u_a	?	?	4.0	3.0	?	1.0	?	5.0
r_a	3.5	4.0			1.3		2.0	

(a)



(b)

ure 1: User-based collaborative filtering example with (a) rating matrix and estimated ngs for the active user, and (b) user neighborhood formation.

Terminology

Long tail – can explain in part the advantages of online retailers compared to traditional "brick and mortar stores".

Large bookstore at the corner -> thousands of books

Bookstore online -> millions of books

Physical newspaper -> limited articles

Online news -> unlimited articles

Terminology

Long tail – can explain in part the advantages of online retailers compared to traditional “brick and mortar stores”.

Large bookstore at the corner -> thousands of books

Bookstore online -> millions of books

Physical newspaper -> limited articles

Online news -> unlimited articles

Shaped by aggregate numbers
A finite number of suggestions

Terminology

Long tail – can explain in part the advantages of online retailers compared to traditional "brick and mortar stores".

Large bookstore at the corner -> thousands of books

Bookstore online -> millions of books

Shaped by aggregate numbers
A finite number of suggestions

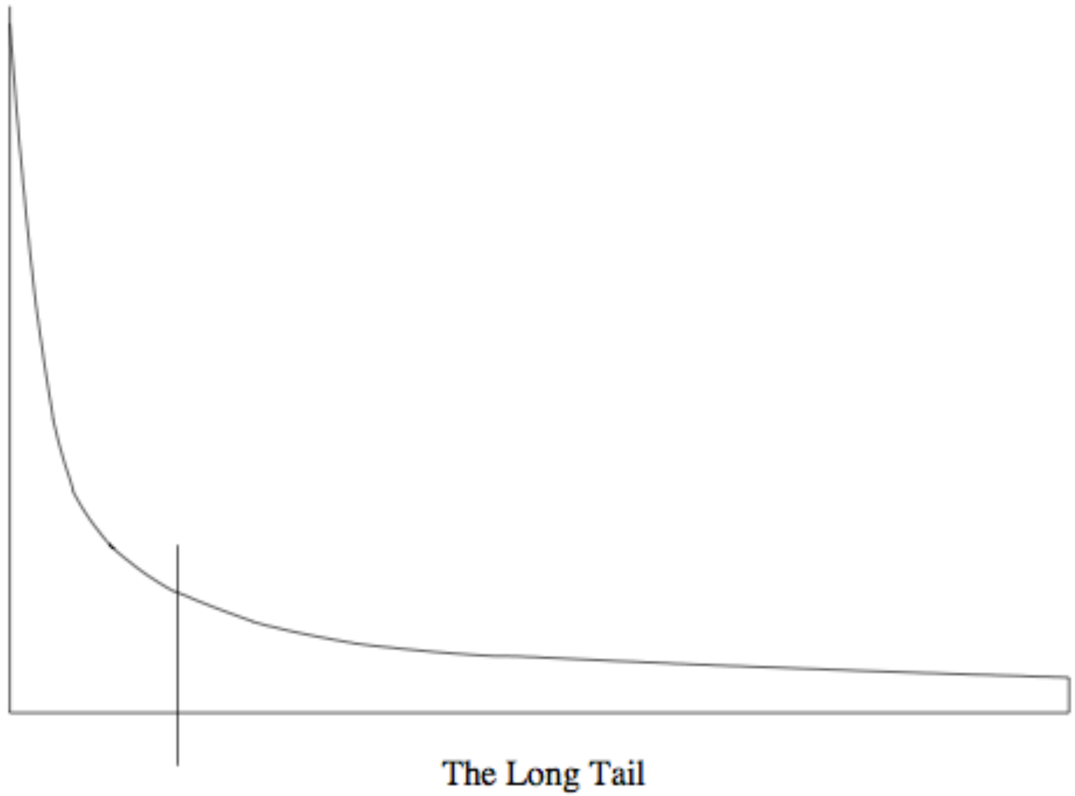
Physical newspaper -> limited articles

Online news -> unlimited articles

Forced to make recommendations
Can't show everything.

Terminology

Long tail – can explain in part the advantages of online retailers compared to traditional “brick and mortar stores”. They can use the entire distribution.



Into Thin Air and *Touching the Void*

An extreme example of how the long tail, together with a well designed recommendation system can influence events is the story told by Chris Anderson about a book called *Touching the Void*. This mountain-climbing book was not a big seller in its day, but many years after it was published, another book on the same topic, called *Into Thin Air* was published. Amazon's recommendation system noticed a few people who bought both books, and started recommending *Touching the Void* to people who bought, or were considering, *Into Thin Air*. Had there been no on-line bookseller, *Touching the Void* might never have been seen by potential buyers, but in the on-line world, *Touching the Void* eventually became very popular in its own right, in fact, more so than *Into Thin Air*.

Data Acquisition

Two Basic Approaches:

1. Ask users' to rate items.

   Generally, people don't want to waste time for a stranger or companies gain. Responses may be bias.

2. Can make inferences from users' behavior.

   If you click "it", you like it. Binary, but 0 is an NA.

Popular approaches

Two Basic Approaches to recommender systems:

1. Content-based

   Examine similarity of items.

2. Collaborative Filtering

   Focus on the relationship between users and items.

Popular approaches

Two Basic Approaches to recommender systems:

1. Content-based

   Examine similarity of items.

2. Collaborative Filtering

   Focus on the relationship between users and items.

   - memory-based ~ work in real memory
   - model-based ~ work on clusters of preference (scalability)

Collaborative Filtering

Formally, we have a set of users: $U = \{u_1, u_2, \ldots, u_m\}$

and a set of items: $I = \{i_1, i_2, \ldots, i_n\}$.

Ratings are stored in a $(m \times n)$ user-item rating matrix $R = (r_{jk})$, where each row represents a user u_j and columns represent items i_k.

*The matrix is sparse!

*Item ratings are on a fixed scale --- > regression.

Collaborative Filtering

Objective: Create recommendations for an active user, $u_a \in U$.

Unknown items: We define the set of items unknown to the user u_a as: $I_a = I \setminus \{i_l \in I \mid r_{al} = 1\}$.

Classical Tasks:

- Predict ratings for all items I_a.
- Create top-rated lists for user u_a.

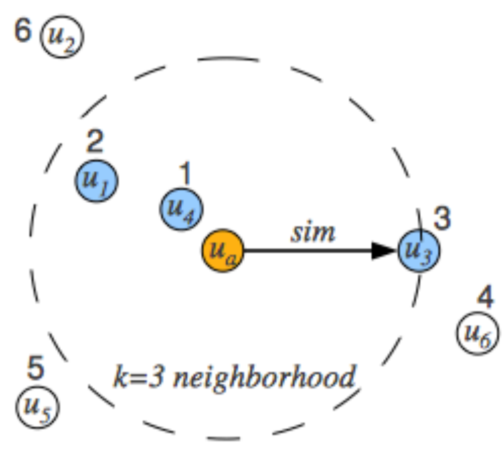
Collaborative Filtering

User-based collaborative filtering:

- Memory based
- Mimics “word of mouth”.
- Users with similar preferences rate items similarly.

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	?	4.0	4.0	2.0	1.0	2.0	?	?
u_2	3.0	?	?	?	5.0	1.0	?	?
u_3	3.0	?	?	3.0	2.0	2.0	?	3.0
u_4	4.0	?	?	2.0	1.0	1.0	2.0	4.0
u_5	1.0	1.0	?	?	?	?	?	1.0
u_6	?	1.0	?	?	1.0	1.0	?	1.0
u_a	?	?	4.0	3.0	?	1.0	?	5.0
r_a	3.5	4.0			1.3		2.0	

(a)



(b)

ure 1: User-based collaborative filtering example with (a) rating matrix and estimated ngs for the active user, and (b) user neighborhood formation.

Collaborative Filtering

User-based collaborative filtering:

$$\mathrm{sim}_{\mathrm{Pearson}}(\vec{x},\vec{y}) = \frac{\sum_{i\in I}(\vec{x}_i\,\bar{\vec{x}})(\vec{y}_i\,\bar{\vec{y}})}{(|I|-1)\,\mathrm{sd}(\vec{x})\,\mathrm{sd}(\vec{y})} \tag{1}$$

and

$$\mathrm{sim}_{\mathrm{Cosine}}(\vec{x},\vec{y}) = \frac{\vec{x}\cdot\vec{y}}{\|\vec{x}\|\|\vec{y}\|}, \tag{2}$$

where $\vec{x} = r_x$ and $\vec{y} = r_y$ represent the row vectors in $\mathbf{R}$ with the two users' profile vectors. $\mathrm{sd}(\cdot)$ is the standard deviation and $\|\cdot\|$ is the l^2-norm of a vector. For calculating similarity using rating data only the dimensions (items) are used which were rated by both users.

Collaborative Filtering

User-based collaborative filtering:

Once the neighborhood is identified, the user ratings are aggregated to form a prediction:

$$\hat{r}_{aj} = \frac{1}{|\mathcal{N}(a)|} \sum_{i \in \mathcal{N}(a)} r_{ij}$$

Can also think about a weighted average:

$$\hat{r}_{aj} = \frac{1}{\sum_{i \in \mathcal{N}(a)} s_{ai}} \sum_{i \in \mathcal{N}(a)} s_{ai} r_{ij}$$

Where s_{ai} is the similarity between the active and the i^{th} user.

Collaborative Filtering

User-based collaborative filtering:

Elimination of rating bias, can be done via normalization, or z-score, etc. For example:

$$h(r_{ui}) = r_{ui} - \bar{r}_u,$$

*Note: heavy need for memory: users have to all be compared via similarity, and the user database has to be in memory.

Collaborative Filtering

Item-based collaborative filtering:

- model-based – looks at similarities of items.
- n x n similarity matrix, items with low relationships are removed from the matrix (dimension reduction/filtering).

n x k --- trimmed

Weighted sum of user rating's on similar items.

$$\hat{r}_{ui} = \frac{1}{\sum_{j \in S(i)} s_{ij}} \sum_{j \in S(i)} s_{ij} r_{uj}$$

Collaborative Filtering

Item-based collaborative filtering:

S	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	u_a
i_1	-	0.1	0	**0.3**	**0.2**	**0.4**	0	0.1	2
i_2	0.1	-	**0.8**	**0.9**	0	**0.2**	0.1	0	?
i_3	0	**0.8**	-	0	**0.4**	0.1	0.3	**0.5**	?
i_4	**0.3**	**0.9**	0	-	0	**0.3**	0	0.1	?
i_5	**0.2**	0	**0.7**	0	-	**0.2**	0.1	0	4
i_6	**0.4**	**0.2**	0.1	**0.3**	0.1	-	0	0.1	?
i_7	0	**0.1**	**0.3**	0	0	0	-	0	?
i_8	**0.1**	0	**0.9**	**0.1**	0	0.1	0	-	5
r_a	-	0.0	4.6	2.8	-	2.7	0.0	-	

$k=3$

Everything else
Is disgarded

We know these.

Collaborative Filtering

Item-based collaborative filtering:

S	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	u_a
i_1	-	0.1	0	**0.3**	**0.2**	**0.4**	0	0.1	2
i_2	0.1	-	**0.8**	**0.9**	0	**0.2**	0.1	0	?
i_3	0	**0.8**	-	0	**0.4**	0.1	0.3	**0.5**	?
i_4	**0.3**	**0.9**	0	-	0	**0.3**	0	0.1	?
i_5	**0.2**	0	**0.7**	0	-	**0.2**	0.1	0	4
i_6	**0.4**	**0.2**	0.1	**0.3**	0.1	-	0	0.1	?
i_7	0	**0.1**	**0.3**	0	0	0	-	0	?
i_8	**0.1**	0	**0.9**	**0.1**	0	0.1	0	-	5
r_a	-	0.0	4.6	2.8	-	2.7	0.0	-	

$k=3$

Everything else
Is disgarded

```
> 1/(.7+.9)*(2*0 + 4*.7 + 5*0.9)
[1] 4.5625
```

Collaborative Filtering 0-1

Less information available on 0-1:

- People don’t want to do ratings.
- Preference has to be inferred by analyzing usage behavior.

In this setting, the 1’s are easy to understand!

Collaborative Filtering 0-1

Less information available on 0-1:

- People don't want to do ratings.
- Preference has to be inferred by analyzing usage behavior.

In this setting, the 1's are easy to understand!

The zeros are more mysterious:

S1) The customer did not need the product right now.

S2) The customer did not know about the product.

S3) The customer did not like the product.

Same issues hold in "click-stream" modeling.

Collaborative Filtering 0-1

In the 0-1 case with $r_{jk} \in 0, 1$ where we define:

$$r_{jk} = \begin{cases} 1 & \text{user } u_j \text{ is known to have a preference for item } i_k \\ 0 & \text{otherwise.} \end{cases}$$

Two possibilities:

- Assume "missing".
- Assume "negative".
- Some trade-off's have been proposed.

$$\mathrm{sim}_{\mathrm{Jaccard}}(\mathcal{X}, \mathcal{Y}) = \frac{|\mathcal{X} \cap \mathcal{Y}|}{|\mathcal{X} \cup \mathcal{Y}|}, \tag{7}$$

where $\mathcal{X}$ and $\mathcal{Y}$ are the sets of the items with a 1 in user profiles u_a and u_b, respectively.

Conclusions

- Recommender systems are not going away.
- Collaborative Filtering via user-based effective but expensive.
- Collaborative Filtering via item-based more realistic in terms of computation, subtly less accurate.
- Really a “many” supervised learning problems.

- The 0-1 case less understood, but more relevant.

Thinking points.....

- The 0-1 case less understood, but more relevant.

- Conceptually compare and draw parallels the ideas of association rules, to the 0-1 case for recommender systems (using Jaccard Similarity).
- Is there a useful analogy?
- How do the "outputs" and "overall objectives" compare?

Discussion Board open until next Friday.

Up to +5 points extra credit on Homework Grade.

Medical Applications: Why Recommender System

- Handles data of huge size well
- Handles sparse data well

From Netflix to Heart Attacks: Collaborative Filtering in Medical Datasets

Shahzaib Hassan
University of Michigan
2260 Hayward St.
Ann Arbor, MI 48109
shahzaib@eecs.umich.edu

Zeeshan Syed
University of Michigan
2260 Hayward St.
Ann Arbor, MI 48109
zhs@eecs.umich.edu

Users = Patients
Items = patient characteristics & outcome of interest
Ratings = "Ratings"

	Sex	Age	Blood pressure	…	Disease outcome
A					
B					
C					
D					
…					

- http://cran.r-project.org/web/packages/recommenderlab/vignettes/recommenderlab.pdf