Multimedia Technology

Lecture 11: Recommendation System

Lecturer: *Dr*. Wan-Lei Zhao

Autumn Semester 2022

Email: wlzhao@xmu.edu.cn,

Outline

Overview about online Recommendation (1)

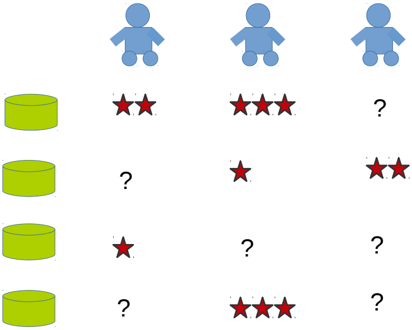
- We have so far covered about searches
- In this case, users are active
- We are going to talk about another issue
- Where the users are subjective: recommendation

Overview about online Recommendation (2)

- Online shopping
- Online services
- Direct the items to the one who is in need
  1. Products
  2. Services
  3. News
  4. Friends
  5. ...

Overview on the Recommendation System (1)

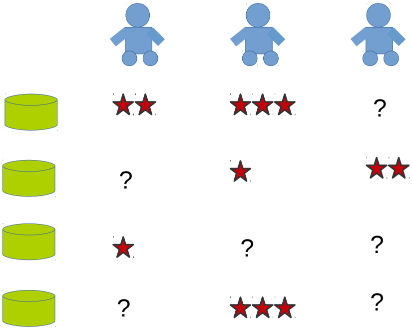
- Given a user, we already know he/she is interested in some items (movies, products, etc.)
- Recommend other items that he/she is also likely interested in



Overview on the Recommendation System (2)

- There are many ways to handle this issue
  1. Given we know the properties of each item
  2. We can recommend similar items that user already shows his interests

Overview on the Recommendation System (3)



- There are many ways to handle this issue
  1. Given we know the profile of each user
  2. We can recommend the items that fit to the profile of the user

Major Issues in Collaberative Recommendation

- Given the profiles of each user and the properties of each item are missing
- The recommendation is only based on ratings
  1. Ratings are incomplete
  2. Ratings from lazy user could be completely missing
- Collaberative Filtering comes to fit in this scenario

Example: Predicting Movie Ratings

- User rates movie using zero to five stars

Movie	Anna(1)	Jenny(2)	Tom(3)	Jack(4)
Gone with Wind	5	5	0	0
Sound of Music	5	?	?	0
Chinese Odyssey	?	4	0	0
Who Am I	0	0	3	4
Matrix	0	0	5	?

- The problem is
  1. How to automatically rate '?' entries
  2. With the estimated rates for undefined entries, it is possible to do the recommendation

Example: Predicting Movie Ratings (1)

- User rates movie using zero to five stars

Movie	Anna(1)	Jenny(2)	Tom(3)	Jack(4)
Gone with Wind	5	5	0	0
Sound of Music	5	?	?	0
Chinese Odyssey	?	4	0	0
Who Am I	0	0	3	4
Matrix	0	0	5	?

- n_u: number of users
- n_m: number of movies
- $r(i,j)$: if user j rated movie i
- $y^{(i,j)}$: defined iff $r(i,j) = 1$

Example: Predicting Movie Ratings (2)

- Given we predefined two features to describe the style of movies
- They are **action** and **romance**

Movie	Anna (1)	Jenny (2)	Tom (3)	Jack (4)	x_1 (romance)	x_2 (action)
Gone with Wind	5	5	0	0	0.90	0.00
Sound of Music	5	?	?	0	1.00	0.01
Chinese Odyssey	?	4	0	0	0.99	0.00
Who Am I	0	0	3	4	0.10	1.00
Matrix	0	0	5	?	0.00	0.90

- $y^{(i,j)}$ is approximated by $\theta_1^{(i)} \cdot x_1^{(i)} + \theta_2^{(i)} \cdot x_2^{(i)} + \theta_0^{(i)}$

Example: Predicting Movie Ratings (3)

- Given we predefined two features to describe the style of movies
- They are **action** and **romance**

Movie	Anna (1)	x_1 (romance)	x_2 (action)
Gone with Wind	5	0.90	0.00
Sound of Music	5	1.00	0.01
Chinese Odyssey	?	0.99	0.00
Who Am I	0	0.10	1.00
Matrix	0	0.00	0.90

- $y^{(i,1)}$ is approximated by $\theta_1^{(i)} \cdot x_1^{(i)} + \theta_2^{(i)} \cdot x_2^{(i)} + \theta_0^{(i)}$
- It is nothing more than a linear regression problem!!
- For each user, we can do linear regression for them one by one

Example: Predicting Movie Ratings (4)

- Given we are going to minimize following error:

$$\underset{\theta}{\operatorname{argmin}} \frac{1}{2} \sum_{r(i,1)=1} (\theta^T x^{(i)} - y^{(i,1)})^2 + \frac{\lambda}{2} \sum_{k=0}^{n} \theta_k^2 \tag{1}$$

- The second term $\frac{\lambda}{2} \sum_{k=0}^{n} \theta_k^2$ is for regularization
  - We favor θ with lower energy
- θ can be solved out by gradient descent

If $k \neq 0$:

$$\theta_k = \theta_k - \alpha \cdot (\sum_{i:r(i,1)=1} (\theta^T x^{(i)} - y^{(i,1)}) \cdot x_k^{(i)} + \lambda \theta_k) \tag{2}$$

If $k = 0$:

$$\theta_k = \theta_k - \alpha \cdot (\sum_{i:r(i,1)=1} (\theta^T x^{(i)} - y^{(i,1)}) \cdot x_k^{(i)}) \tag{3}$$

Example: Predicting Movie Ratings (5)

- The **objective** for a single user

$$\underset{\theta}{\operatorname{argmin}} \frac{1}{2} \sum_{r(i,1)=1} (\theta^T x^{(i)} - y^{(i,1)})^2 + \frac{\lambda}{2} \sum_{k=0}^{n} \theta_k^2 \tag{4}$$

- We can do this for all the users, the **objective** for each of the user

$$\underset{\theta^{(1)},\cdots,\theta^{(n_u)}}{\operatorname{argmin}} \frac{1}{2} \sum_{j=1}^{n_u} \sum_{r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,1)})^2 + \frac{\lambda}{2} \sum_{k=0}^{n} (\theta_k^{(j)})^2 \tag{5}$$

Example: Predicting Movie Ratings (6)

$$\underset{\theta^{(1)},\cdots,\theta^{(n_u)}}{\text{argmin}} \frac{1}{2}\sum_{j=1}^{n_u}\sum_{r(i,j)=1}((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2}\sum_{k=0}^{n}(\theta_k^{(j)})^2 \tag{6}$$

- We can have its gradient descent as follows

If $k \neq 0$:

$$\theta_k^{(j)} = \theta_k^{(j)} - \alpha\cdot(\sum_{j=1}^{n_u}\sum_{i:r(i,j)=1}((\theta^{(j)})^T x^{(i)} - y^{(i,j)})\cdot x_k^{(i)} + \lambda\theta_k^{(j)}) \tag{7}$$

If $k = 0$:

$$\theta_k^{(j)} = \theta_k^{(j)} - \alpha\cdot(\sum_{j=1}^{n_u}\sum_{i:r(i,j)=1}((\theta^{(j)})^T x^{(i)} - y^{(i,j)})\cdot x_k^{(i)}) \tag{8}$$

Example: Predicting Movie Ratings (7)

- Once we work out all the $\theta^{(j)}$, we can calculate the missing rating

$$\hat{y}^{(i,j)} = (\theta^{(j)})^T \cdot x^{(i)} + \theta_0^{(j)} \tag{9}$$

- In above approach, $x^{(i)}$s are assumed to be known
- In practice, it is hard to extract features from objects
- That is, $x^{(i)}$s are not defined explicitly
- **What we can do if $x^{(i)}$s are missing???**

Collaberative Filtering: the motivation (1)

- Given $x^{(i)}$s are missing

Movie	Anna (1)	x_1 (romance)	x_2 (action)
Gone with Wind	5	?	?
Sound of Music	5	?	?
Chinese Odyssey	?	?	?
Who Am I	0	?	?
Matrix	0	?	?

- How could we estimate $y^{(3,1)}$?
- It is simply very very HARD if it is not impossible

Collaberative Filtering: the motivation (2)

- Given $x^{(i)}$s are missing
- What happen if we know more ratings from different users?

Movie	Anna (1)	Jenny (2)	Tom (3)	Jack (4)	x_1 (romance)	x_2 (action)
Gone with Wind	5	5	0	0	?	?
Sound of Music	5	?	?	0	?	?
Chinese Odyssey	?	4	0	0	?	?
Who Am I	0	0	3	4	?	?
Matrix	0	0	5	?	?	?

1. Both Anna and Jenny give high rates to "Gone with Wind"
2. Jenny also gives high rate to "Chinese Odyssey"
3. Anna might give high rate to "Chinese Odyssey" too

Collaberative Filtering: the motivation (3)

- Given $x^{(i)}$s are missing
- We are going to learn $x^{(i)}$s and θ as well

Movie	Anna (1)	Jenny (2)	Tom (3)	Jack (4)	x_1 (romance)	x_2 (action)
Gone with Wind	5	5	0	0	?	?
Sound of Music	5	?	?	0	?	?
Chinese Odyssey	?	4	0	0	?	?
Who Am I	0	0	3	4	?	?
Matrix	0	0	5	?	?	?

Collaberative Filtering: the approach (1)

- Now, given θs are known, we try to learn $x^{(i)}$

$$\underset{x^{(i)}}{\operatorname{argmin}} \frac{1}{2} \sum_{r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2} \sum_{k=1}^{n} (x_k^{(i)})^2 \tag{10}$$

- Similarly, we try to learn $x^{(1)}, \cdots, x^{(n_m)}$

$$\underset{x^{(1)},\cdots,x^{(n_m)}}{\operatorname{argmin}} \frac{1}{2} \sum_{i=1}^{n_m} \sum_{r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2} \sum_{i=1}^{n_m} \sum_{k=1}^{n} (x_k^{(i)})^2 \tag{11}$$

Collaberative Filtering: the approach (2)

- Given θs are known, we try to learn $x^{(1)}, \cdots, x^{(n_m)}$

$$\underset{x^{(1)},\cdots,x^{(n_m)}}{\operatorname{argmin}} \frac{1}{2}\sum_{i=1}^{n_m}\sum_{r(i,j)=1}((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2}\sum_{i=1}^{n_m}\sum_{k=1}^{n}(x_k^{(i)})^2 \tag{12}$$

- Given $x^{(i)}$s are known, we try to learn $\theta^{(1)}, \cdots, \theta^{(n_u)}$

$$\underset{\theta^{(1)},\cdots,\theta^{(n_u)}}{\operatorname{argmin}} \frac{1}{2}\sum_{j=1}^{n_u}\sum_{r(i,j)=1}((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2}\sum_{k=0}^{n}(\theta_k^{(j)})^2 \tag{13}$$

- Eqn. 12 and Eqn. 13 form an egg-chicken loop, optimize them one after another

$$\theta^{(j)} \Rightarrow x^{(i)} \Rightarrow \theta^{(j)} \Rightarrow \theta^{(j)} \Rightarrow x^{(i)} \ldots$$

Collaberative Filtering: the approach (3)

- We can optimize them simultaneously:

$$J(x^{(i)}, \theta^{(j)}) = \frac{1}{2} \sum_{r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2} \sum_{k=1}^{n} (\theta_k^{(j)})^2 + \frac{\lambda}{2} \sum_{i=1}^{n_m} \sum_{k=1}^{n} (x_k^{(i)})^2$$

- We try to minimize

$$J(\theta^{(j)}, x^{(i)})$$

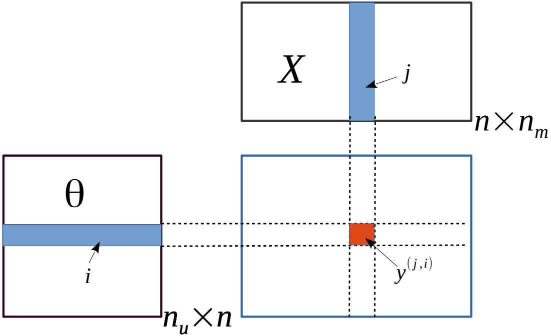
Collaberative Filtering: the approach (4)

1. Initialize $\theta^{(j)}, x^{(i)}$ to small random numbers
2. Minimize $J(\theta^{(j)}, x^{(i)})$ by gradient descent:
    1. $\theta_k^{(j)} = \theta_k^{(j)} - \alpha \cdot (\sum_{i:r(i,j)=1}((\theta^{(j)})^T x^{(i)} - y^{(i,j)}) \cdot x_k^{(i)}) + \lambda \theta_k^{(j)}$
    2. $x_k^{(i)} = x_k^{(i)} - \alpha \cdot (\sum_{i:r(i,j)=1}((\theta^{(j)})^T x^{(i)} - y^{(i,j)}) \cdot \theta_k^{(j)}) + \lambda x_k^{(i)}$

- Results:
    1. We are now able to estimate the rating for the missing entries
    2. We have a vector representation for each movie

Visualize the Problem

- The Optimization on θ and X could be sumarized as following
- Notice that R is a low rank matrix

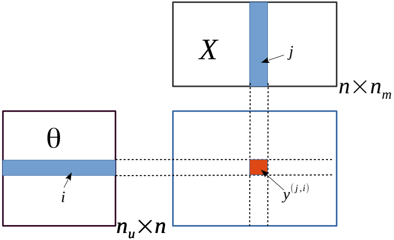


$$y^{(j,i)} = \theta_{i,*} \times X_{*,j} \tag{14}$$

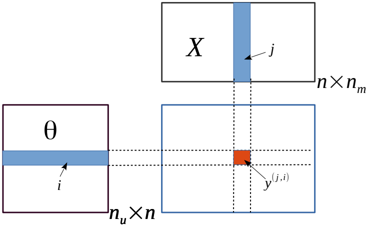
Solve it by Sochastic Gradient Descent

$$J(\theta^{(j)}, x^{(i)}) = \sum_{r(j,i)=1} L(R_{ji}, \theta_{i,*}, X_{*,j})$$

- Initialize $\theta^{(j)}$ and $x^{(i)}$
- **While** not converge
  1. **Select** one $\theta^{(j)}, x^{(i)}$ at random
  2. $\bar{\theta}_k^{(j)} = \theta_k^{(j)} - \alpha \cdot \frac{\partial J(\theta^{(j)}, x^{(i)})}{\partial \theta^{(j)}}$
  3. $x_k^{(i)} = x_k^{(i)} - \alpha \cdot \frac{\partial J(\theta^{(j)}, x^{(i)})}{\partial x^{(i)}}$
  4. $\theta_k^{(j)} = \bar{\theta}_k^{(j)}$
- End-**While**



Discussion about Collaberative Filtering



- How we can produce matrix **R**?
- How to solve such big matrix?

Q & A