Recommeder System-3 | Start at 9:05pm

Agenda
1. Regression based system
2. Collabrotive Filtering

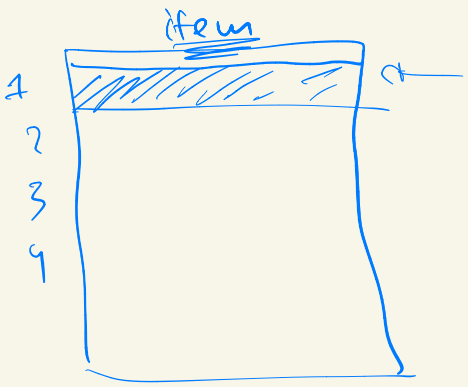
→ Content based → Distances → ↗ • Genre →
• user → attributes →

Market basket Analysis ↔ Looking at past data
↳ Customers also bought

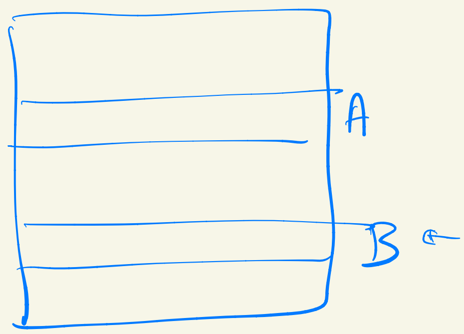
---

* Recommeder based system

⇒ Item table



user table



⇒ genre

Movie Id	Action	Comedy	Romance
→ 1	0	1	0
→ 1	0	1	0
2	0	1	0

Toy

user

id	avg-rating	time	rating
A	4.5	11 hrs	5*
B	4.2*	9 hr	4*
Archit	4*	12hr	(5, 4

⇒ Best recommed → highest rated movie for a user

3 0 0 1 Aashit 4* 12hr

Titanic

⇒ provided data → movie rating

↓

regression model.

⇒ find user meta data → all movies meta data → Predict Ratings

↓

Recommend all movies $\geq 4^*$

⇒ Users table →

User	Age	Time

Rating

user	movie	rating

⇒ User age Time avg_rating avg_time movie rating

⇒ Song Company ⇒ Spotify ⇒

↳ RS

active → passive

- avg rating per item ← active
- Singer ← Passive
- location ← Passive
- avg rating by the user ← active

⇒ at the start you will never have active feature.

& New users → Content based

↳ • ask for preference

↳ Initially recommend using preferences

↳ 3-4 months → active feature system

⇒ Song can have attribut → genre, tempo, beats, pitch, F

↳ Total Likes
↳ Total relisten

↳

Market basket analysis

⇒ user inputs → are not reliables →
↳ active

⇒ 2020 Genre, Tempo, Singer,
→
→
→

market

Product feature

Sales ↑
↓

⇒ Domain Expertise → High Sales ↑ More Business ↑

→ ① Customer also bought → They items are actually related

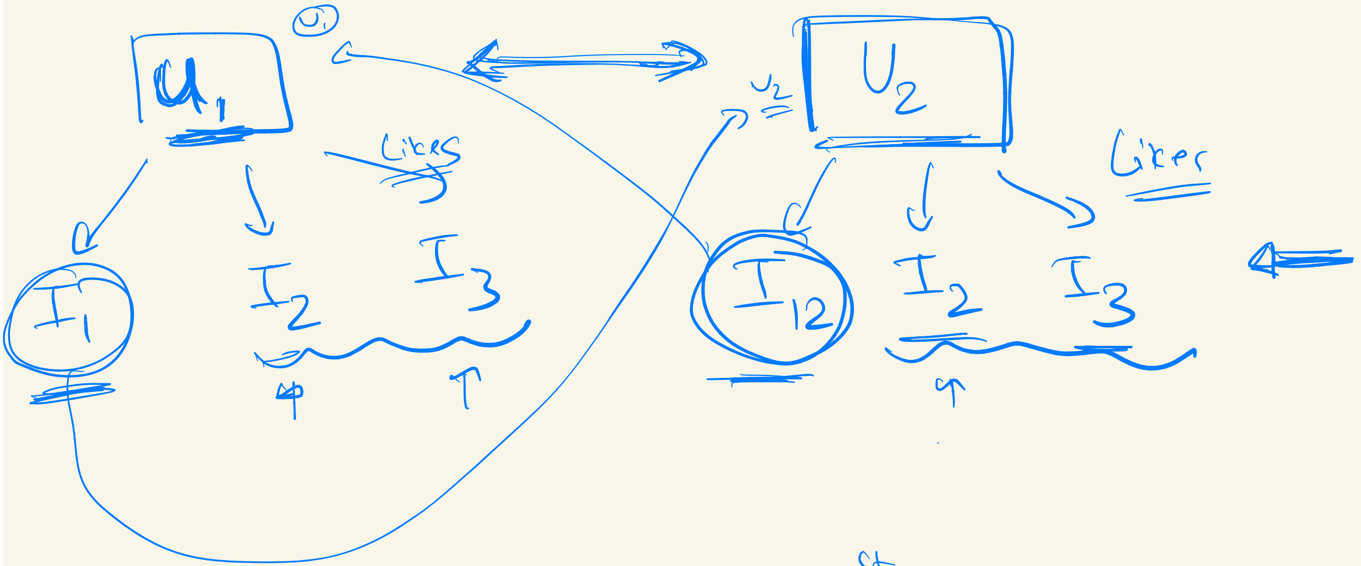
→ ② High rating → Some one like the movie/show

① Ads →
② Popular →

⇒ * RS w/o any Domain Experience

↳ Works everywhere

(Cross recommendation)



Different than

⇒ Content based filtering in 1^{st} class.

Distance b/w U_1 & U_2 (Based on user attributes)

= age, avg_rating, avg_ln_

⇒ Calculate Similarity based on user likes

⇒	= [I_1,	I_2	I_3	I_{12}]	← Likes
U_1 =	1	1	1	0	←
U_2 =	0	1	1	1	←

$\Rightarrow$

$$
\begin{array}{c}
\\ U_1 \\ U_2
\end{array}
\begin{bmatrix}
I_1 & I_2 & I_3 & I_{12} \\
4 & 5 & 5 & \\
 & 5 & 4 & 5
\end{bmatrix}
\quad \text{rating}
$$

$\Rightarrow$ 1 Million users $\times$ 100k Songs

$$
\begin{array}{c}
\\ U_1 \\ U_2 \\ \vdots \\ U_{1M}
\end{array}
\begin{bmatrix}
I_1 & I_2 & - & - & - & I_{100k} \\
4 & 3 & 3 & - & - & 4 \\
 & 5 & 4 & & & 5 \\
 & & & & & \\
 & 1 & & & & 4
\end{bmatrix}
$$

$\Rightarrow$ 100k $\Rightarrow$ most users will have 100-200 movies

100k Columns $\Rightarrow$ 100 entries

Sparsity

Sparse data →

$$
\begin{array}{c|ccccccccc}
 & I_1 & I_2 & & & & & & & \\
U_1 & 1 & 4 & 5 & 0 & 0 & 0 & 1 & & \\
U_2 & 0 & 1 & 0 & 0 & 5 & 0 & 3 & 0 & 0 \\
U_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\vdots & & & & & & & & & \\
 & 1 & & & 2 & & 4 & & & \\
\end{array}
$$

⇒ 5M users
100k movies

$$\frac{5\times10^6 \times 100}{5\times10^6 \times 10^5} = \frac{1}{1000} =$$

99% cells are empty

⇒ If I go and try to calculate similari

Iterate over 5M rows
100k columns

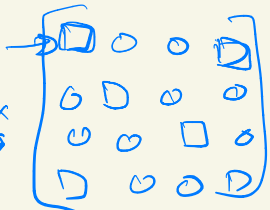
⇒ Matrix Factorisation:

$$\Rightarrow \begin{bmatrix} 2 & 3 & 4 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 1 & 1 \\ 2 & 3 \\ 4 & 5 \end{bmatrix} = \begin{bmatrix} 2\times1 + 3\times2 + 4\times4 & 2\times1 + 3\times3 + 4\times4 \\ 4\times1 + 5\times2 + 6\times4 & \vdots \end{bmatrix}$$

2×3 $(n\times d)$ $\quad$ 3×2 $(d\times m)$ $\quad$ $\boxed{2\times2}$ $(n\times m)$

$\Rightarrow$ 5M × 100k matrix

5M×100k

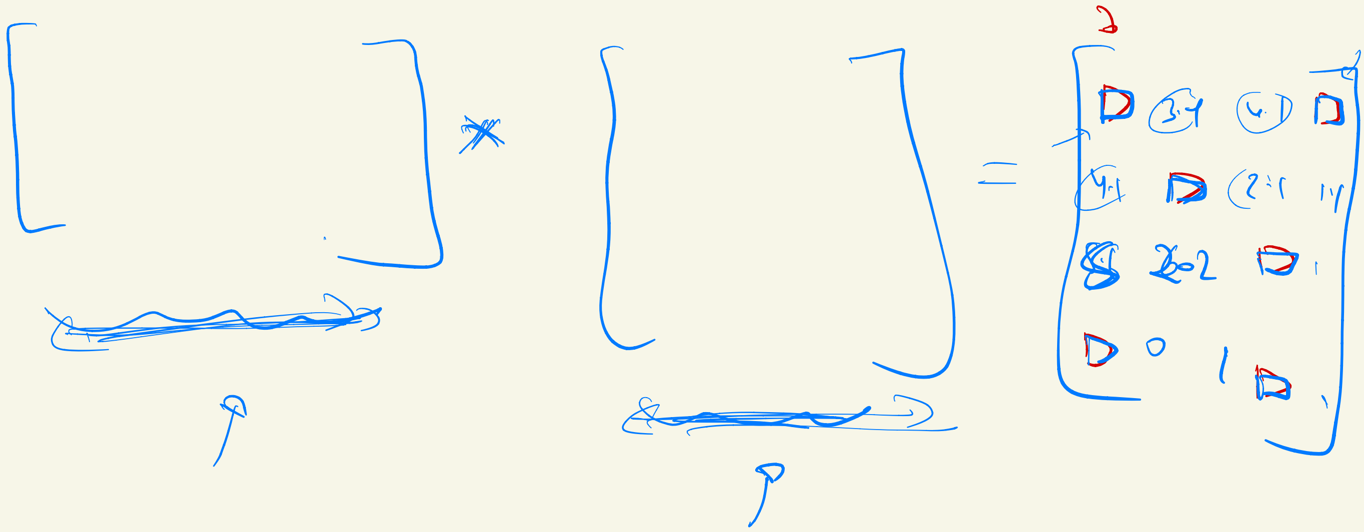


original matrix

sparse

A: $5M \times 50$ $\quad$ B: $50 \times 100k$

$A \times B = C$



⇒ ⇒ 5M × 20 matrix ←

20 × 100k matrix ←

⇒ Matrix factorisation → Maths → very popular

⇒ Latent Feature | Embedding | encoding

↳ But biproduct created while simplyfying a problem;

- PCA ←
- TSNE ←
- Auto encoder (NN)
- NLP | CV
  ↳ embedding

⇒ r ⇒ Sparse a lot of Zeros

matrix

↓

A, B → Latentfeature / embedding

⇒

↓

$A \times B = r^{*}_{matrix}$ → $\boxed{r_{i,j} \text{ when } r_{i,j} \neq 0 \approx r^{*}_{i,j}}$

⇒ LF / embedded do not have any buisness logic

↳ Independtly

$$u_1 \Rightarrow [\; I_1 \quad I_2 \quad \ldots \quad I_{100} \;]$$

rating for each item by user

$$A \Rightarrow [\,0.13,\ 0.22,\ \ldots\,]$$

no meaning of this data

mathematical constructs.

$$\underset{n \times m}{r} \xrightarrow{\sim} \underset{n \times m}{r'} = \underset{n \times d}{P} \quad \underset{d \times m}{Q}$$

$$\boxed{r' = P^T \cdot Q}$$

assuming d is provided

hyper parameter

Error $= \sum \left(r_{i,j} - r'_{i,j} \right)$ where $r_{i,j} \neq 0$

$$= \sum_i \sum_j \left(r_{i,j} - p_u^T q_i \right)^2 \leftarrow \text{gradient descent}$$

$\hookrightarrow$ minimize this

Q 56 items how many parameter you will have to learn
20 users

$d = 4$

$(56 \times 4) \leftarrow$
$+$
(4×20)

$\Rightarrow$ Total params
$\Downarrow$
Total cells in your embedding

$\Rightarrow$  $\Leftarrow$ error

you need to remember:

⇒ Advance version of above equation

ALS → Alternating least Square :

⇒ →

⇒ CMFREC library

ALS → Alternating least Square

$$\Rightarrow \sum_i \sum_j \underbrace{r_{ij} = P_i^T Q_j} +$$

LR

$$\Rightarrow w_1 x_1 + w_2 x_2 + \ldots + w_n x_n + \lambda (w_1^2 + w_2^2 + \ldots + w_n^2)$$

$$+ \lambda \left(\| \vec{w} \|^2 \right)$$

$q[\ \ ]$ $\quad$ $p[q]$

$$\Rightarrow \lambda \left(\|q\|^2 + \|p\|^2 \right)$$

idea $\gamma'_{iu} = p_u^T q_i$

avg of all the value prediction $\Rightarrow$ Predict changes from your mean

$\Rightarrow$ How biased are users in comparision to other users

$\Rightarrow \mu \longleftrightarrow b_u$

$\mu \longleftrightarrow b_i$

$\Rightarrow$

$$\boxed{\gamma'_{iu} = \mu + p_u^T q_i + b_u + b_i + d(\ )}$$

$\Rightarrow \gamma_{ui} - \gamma'_{ui}$ $\Rightarrow$ 1,2 values might heavily skew

$\Rightarrow$ every user - item pair will have a bias

a user ~~every~~ $\Rightarrow$ -1 rating   1 ←

a item $\Rightarrow$ +1 rating   subtract

⇒ Compare avg of the item against overall avg

⇒ Find all user × item matrix (Sparse matrix

↓

divide into two dense matrix ( )
A, B

↓

Cross Product of A, B → will Predict the probable ratings

↓

Recommend top rated items