MACHINE LEARNING ALGORITHM
Unit-II Part-II

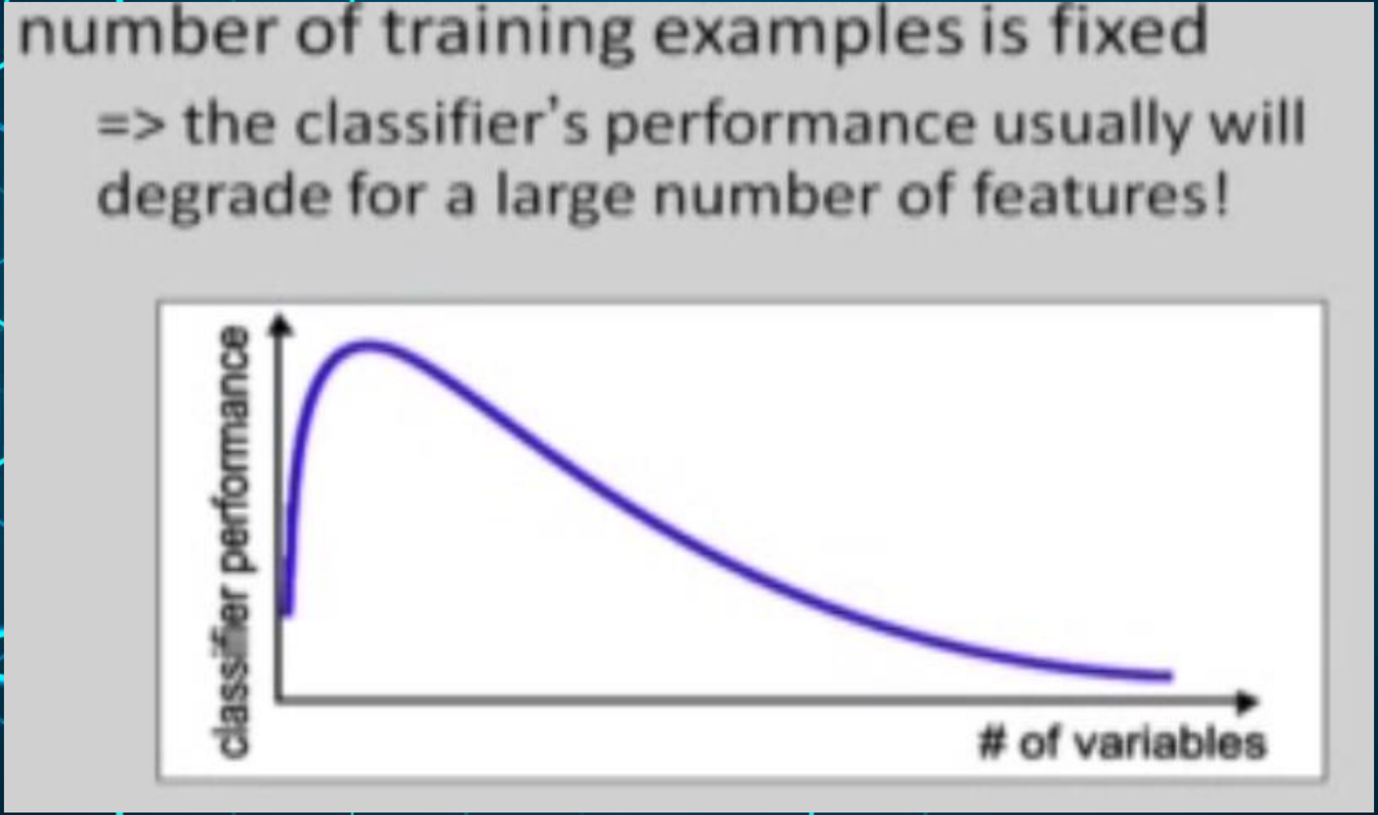
Dr. Gopal Sakarkar
Department of AI, GHRCE, Nagpur

YouTube Channel on Machine Learning Algorithms
https://tinyurl.com/GopalMachineLearningAlgorithms

Feature Reduction

- For given Test data instance , we have to find out near by instance, for this we need distance function.
- This distance function computed in the term of features.
- If number of features are large , there is a problem , because distance may not be represent actual distance.
- As feature contained information about target
- More feature means more information and better classification
- But , we have to properly handle
- - Irrelevant Feature : As it create noise
- - Redundant Feature : It lead to degradation of performance of algorithm .

Feature Reduction



number of training examples is fixed

=> the classifier's performance usually will degrade for a large number of features!

Feature Reduction: Type 1

- Feature Selection :

Let , F = { X1,X2,X3…….Xn}
F’ Ƈ F ={X1’,X2’,X3’……..Xm}
where F’ is a subset of F

- Here , to find subset of feature to optimize certain criteria.

Feature Reduction: Type 2

- Feature Extraction:
- It is a transform of the original set of feature into a new subset , which has a small number of dimension.

Feature Reduction

- **Forward Selection**:
- We started with empty set of feature set and add one feature at a time.
- Then try each of remaining features.
- Estimate classification/ regression error for adding each features.
- Selected a feature that gives maximum improvement
- Stop , when there is no significant improvement

Feature Reduction

- Backward Selection:
- We started with full feature set
- Then try remaining features
- Drop a feature with smallest improvement /impact on error

Feature Reduction

Filter Methods

1. IG- Information Gain Method
2. Chi-squre Test
3. Correlation Coef.

Wrapper Methods

1. Recursive feature elimination
2. Genetic Algorithms

Embedded Methods

1. Decision Tree

Selection of Optimal Feature / Correlation Coef. Method

A	B	C	D	E	T

Step1: Find the correlation of attribute like A with target attribute T

Wrapper Methods

A	B	C	D	E	T

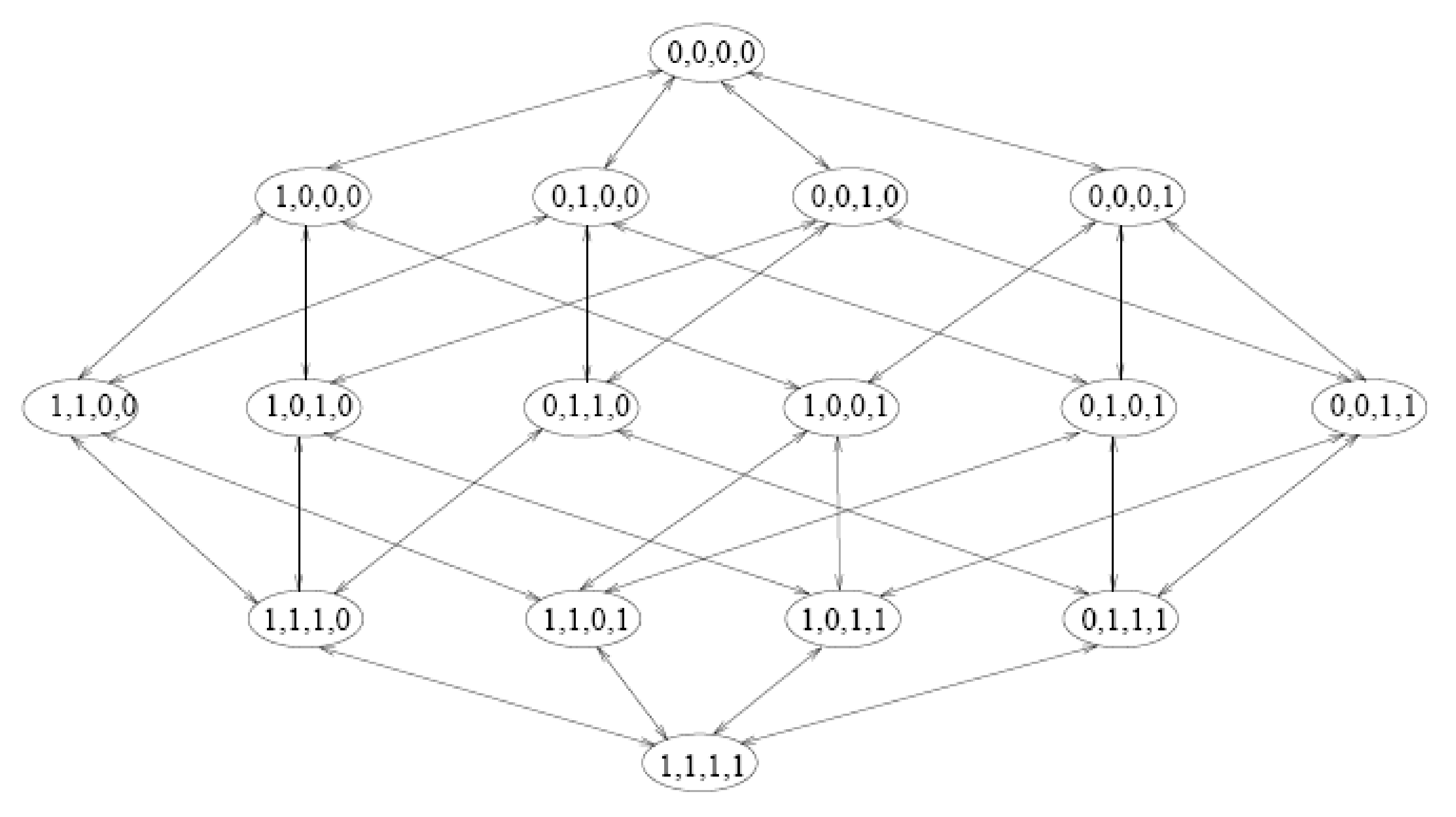
Here , we have to make a combination (wrapping) of attributes with target attribute.

Let us consider , attribute A and target attribute T will be provide to Machine Learning Algorithm which developed model M1.

Now , consider attribute A and B combine with T and provide to machine learning algorithm , which develop model M2.

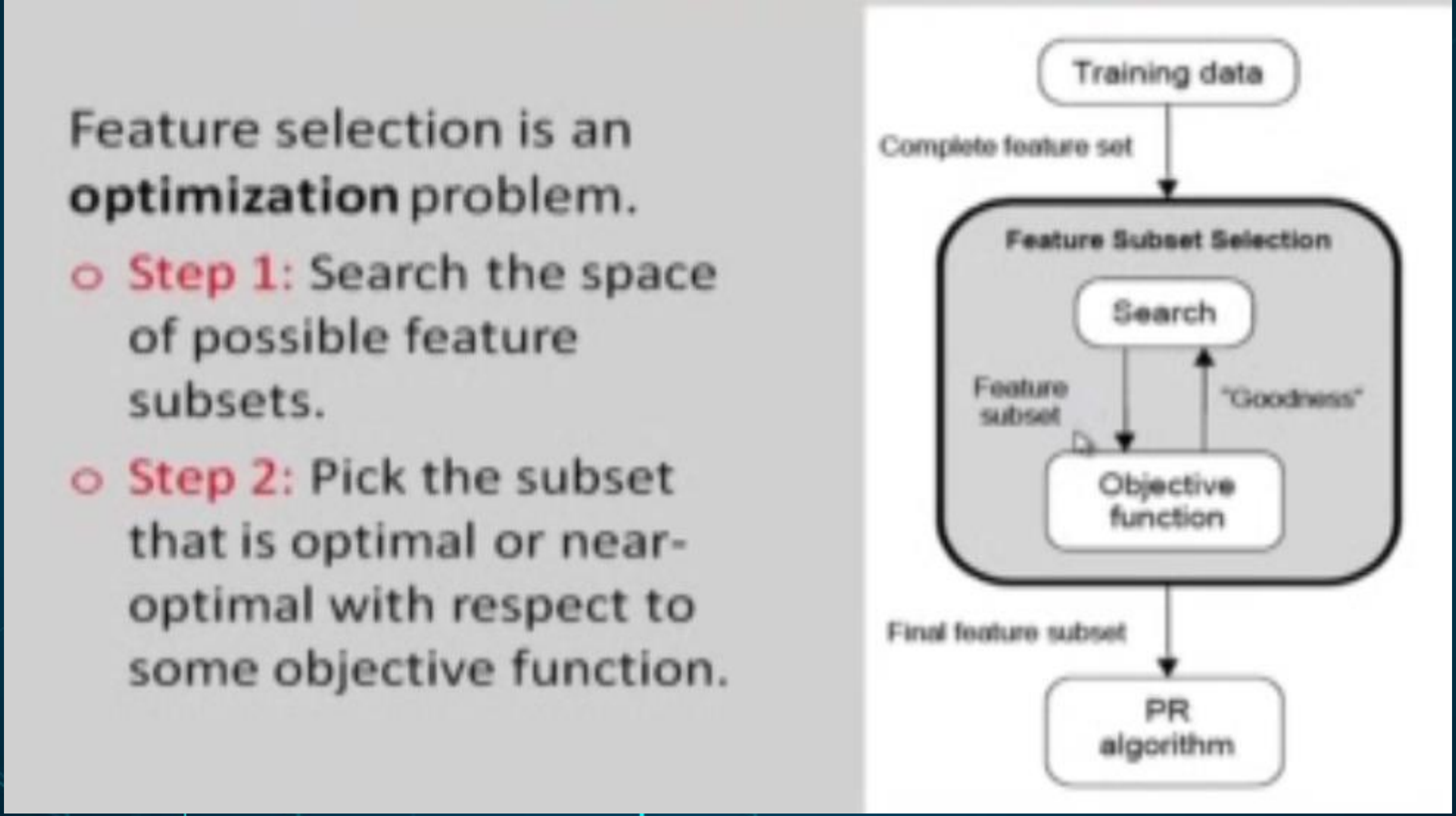
Now, A , B and C combine with T and provide to machine learning algorithm , which develop model M3 and so on .

Then find the best Model match with Objective.



N features, 2N possible feature subsets!

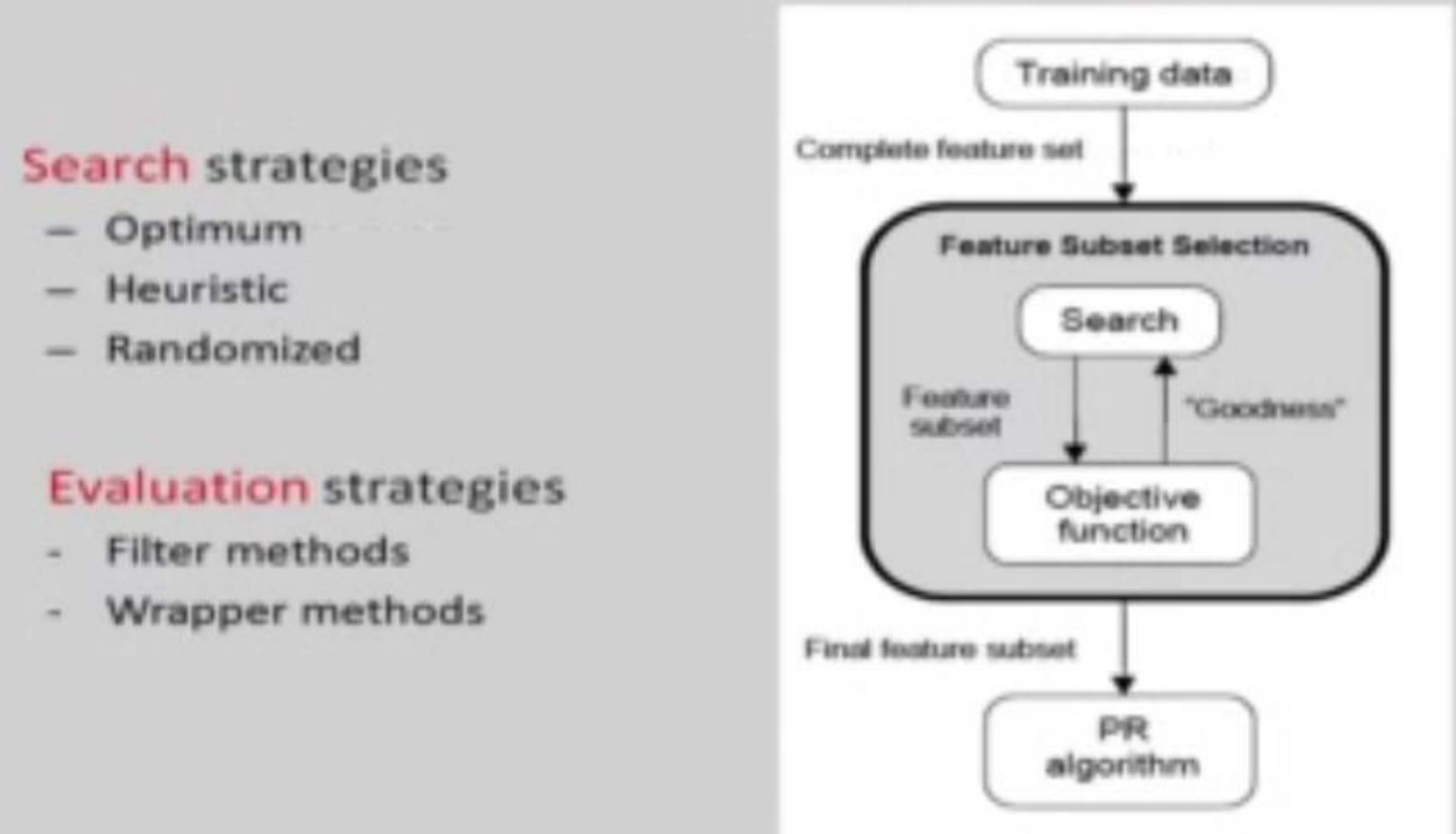
Feature Reduction Steps



Feature selection is an **optimization** problem.

- Step 1: Search the space of possible feature subsets.
- Step 2: Pick the subset that is optimal or near-optimal with respect to some objective function.

Feature Reduction Steps(cont'd)



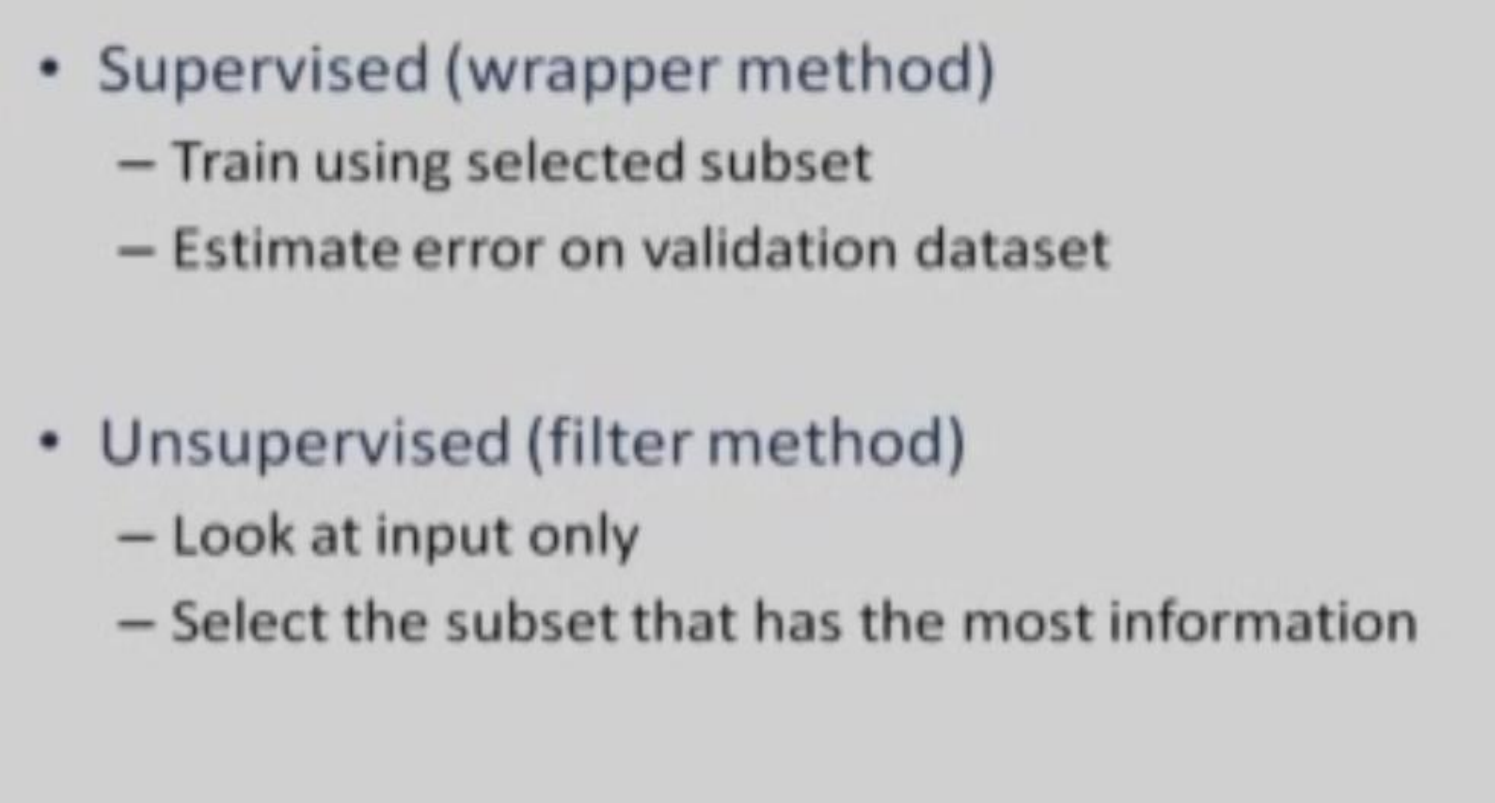
Search strategies

- Optimum
- Heuristic
- Randomized

Evaluation strategies

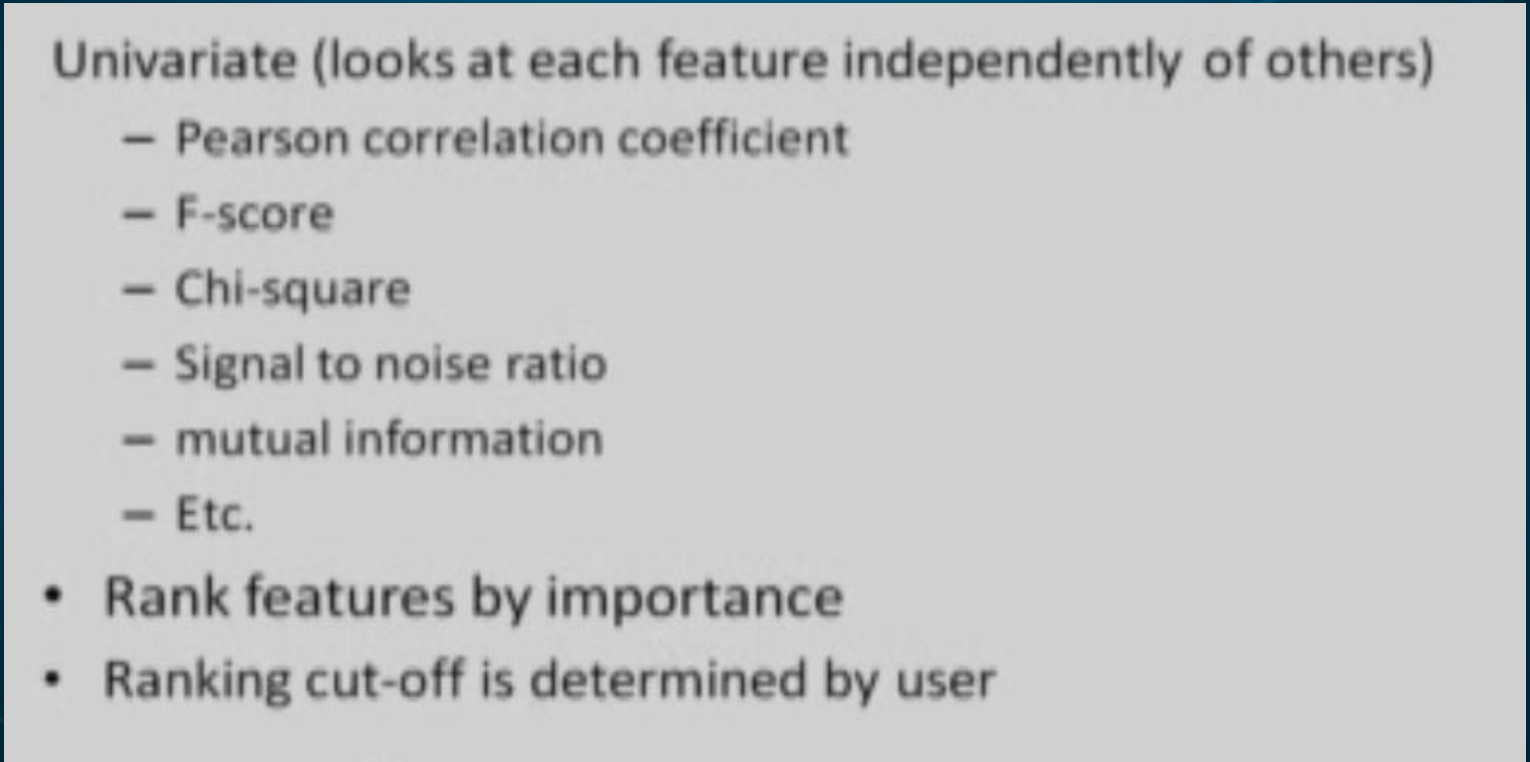
- Filter methods
- Wrapper methods

Evaluating feature subset



- Supervised (wrapper method)
  - Train using selected subset
  - Estimate error on validation dataset
- Unsupervised (filter method)
  - Look at input only
  - Select the subset that has the most information

Feature selection

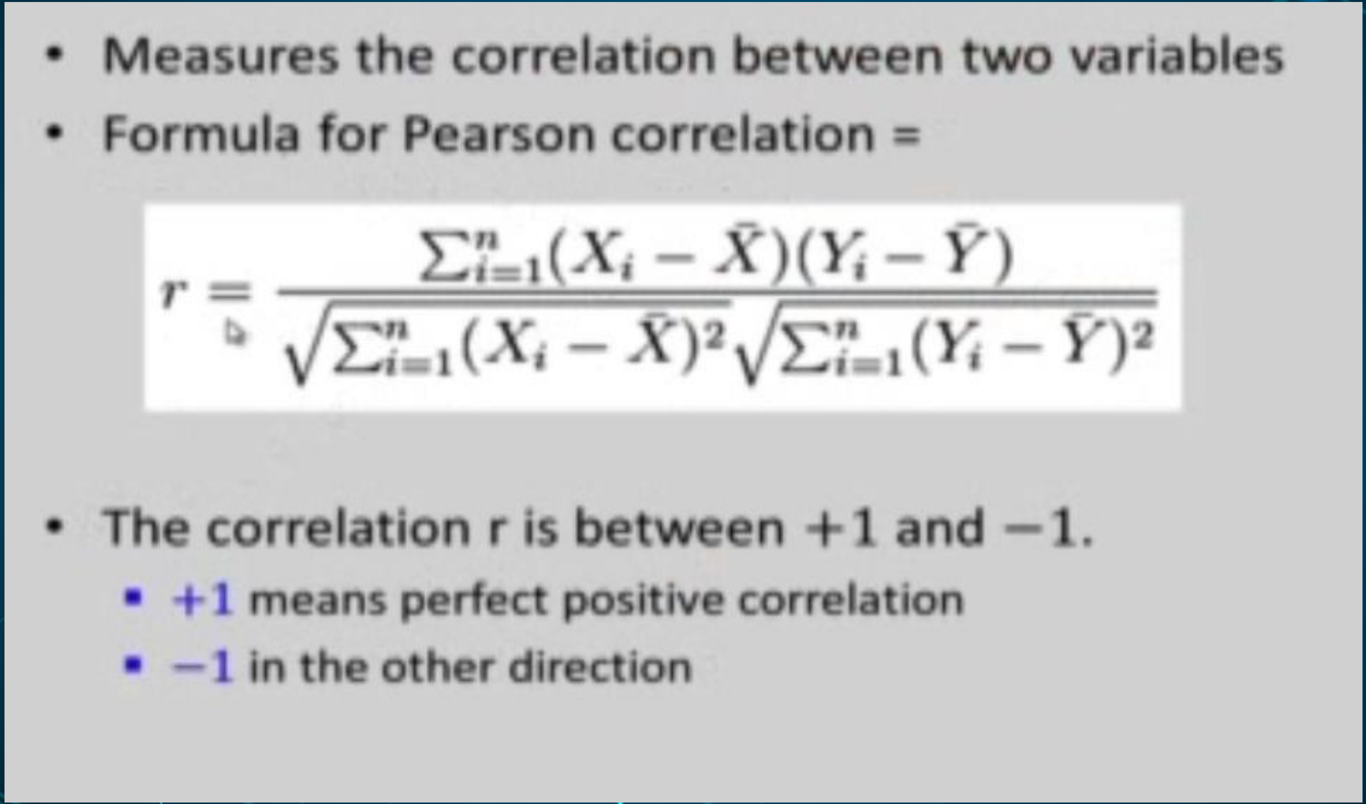


Univariate (looks at each feature independently of others)

- Pearson correlation coefficient
- F-score
- Chi-square
- Signal to noise ratio
- mutual information
- Etc.

- Rank features by importance
- Ranking cut-off is determined by user

Pearson correlation coefficient

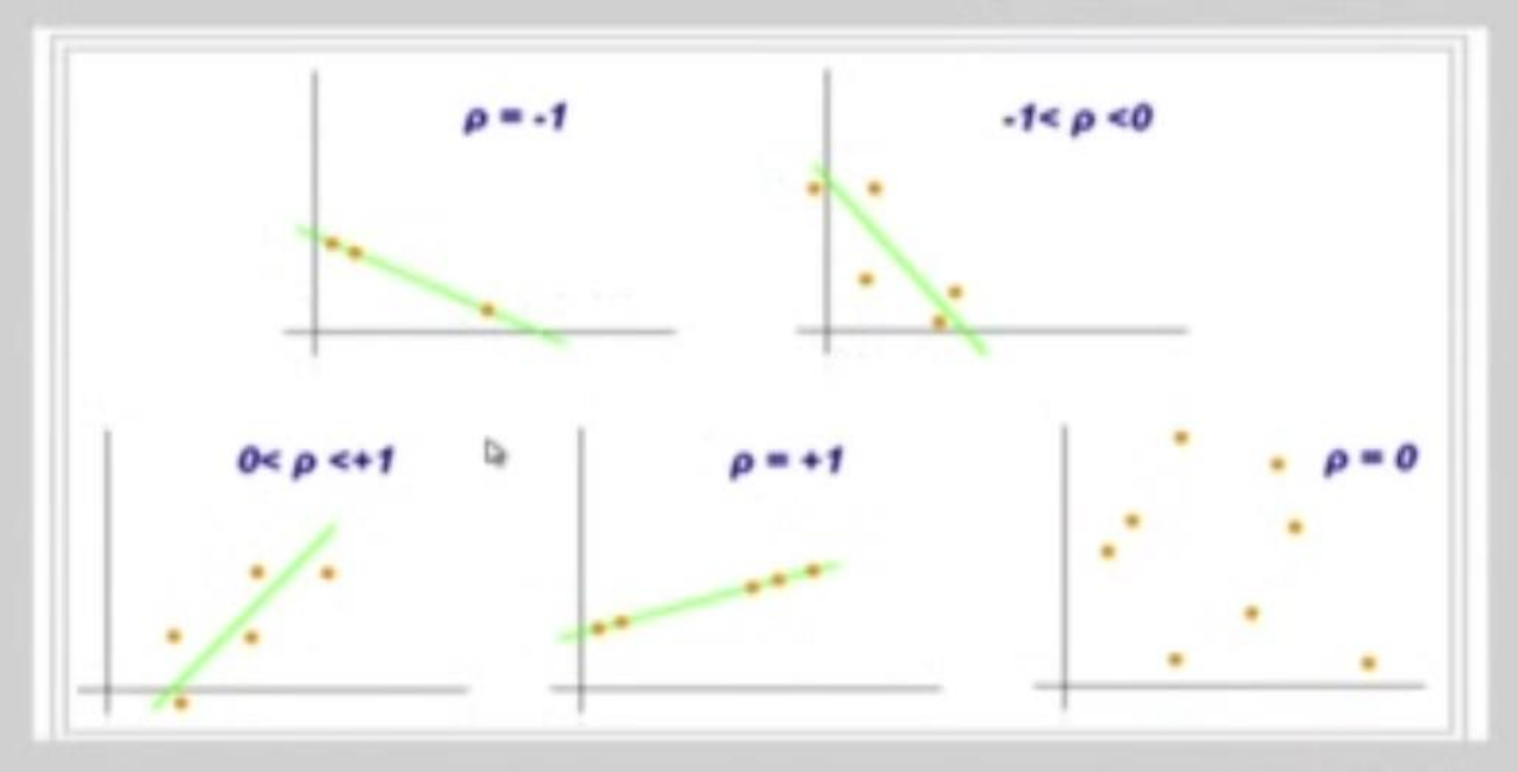


- Measures the correlation between two variables
- Formula for Pearson correlation =

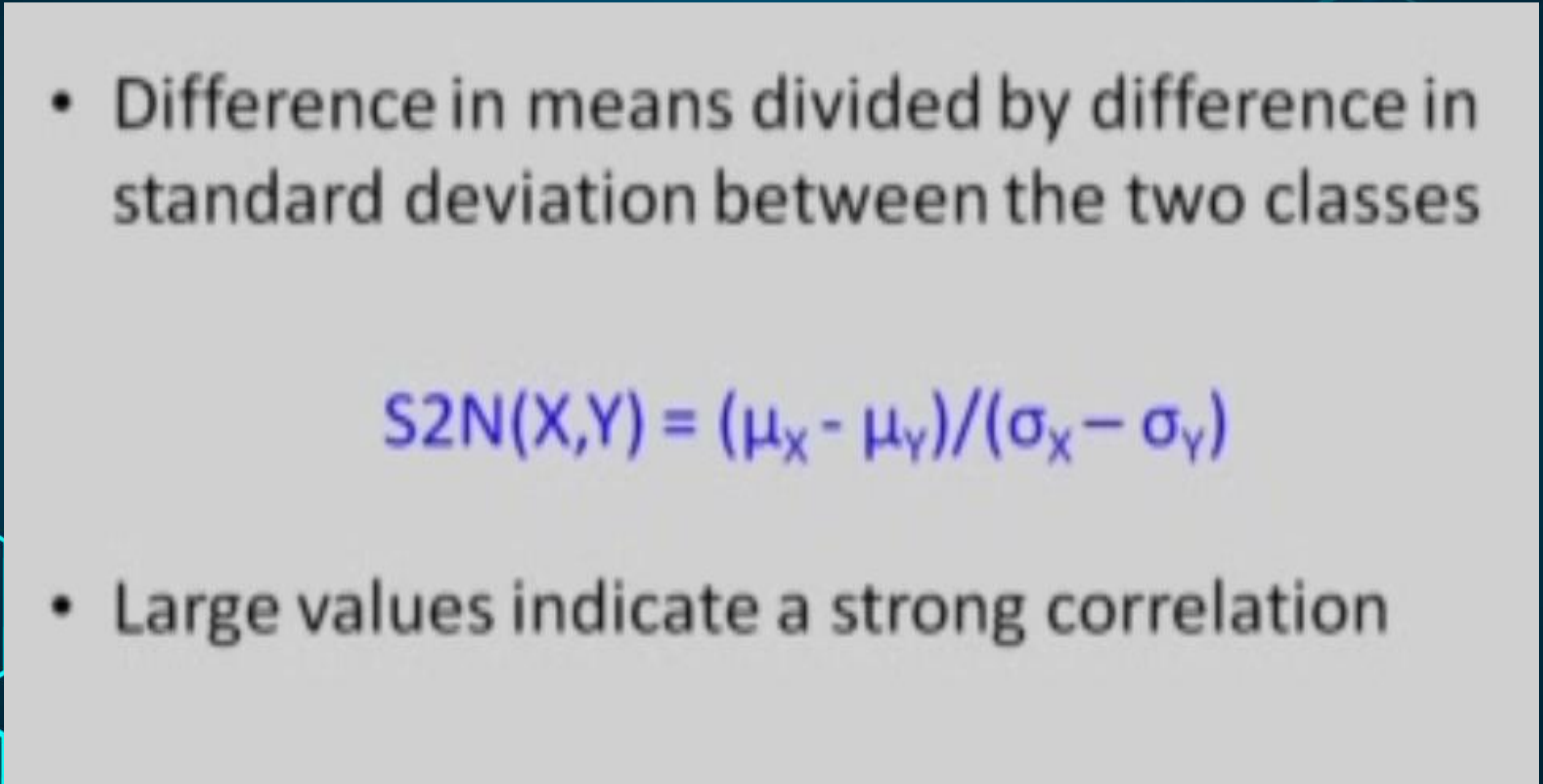
$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

- The correlation r is between +1 and −1.
  - +1 means perfect positive correlation
  - −1 in the other direction

Pearson correlation coefficient



Signal to noise ratio

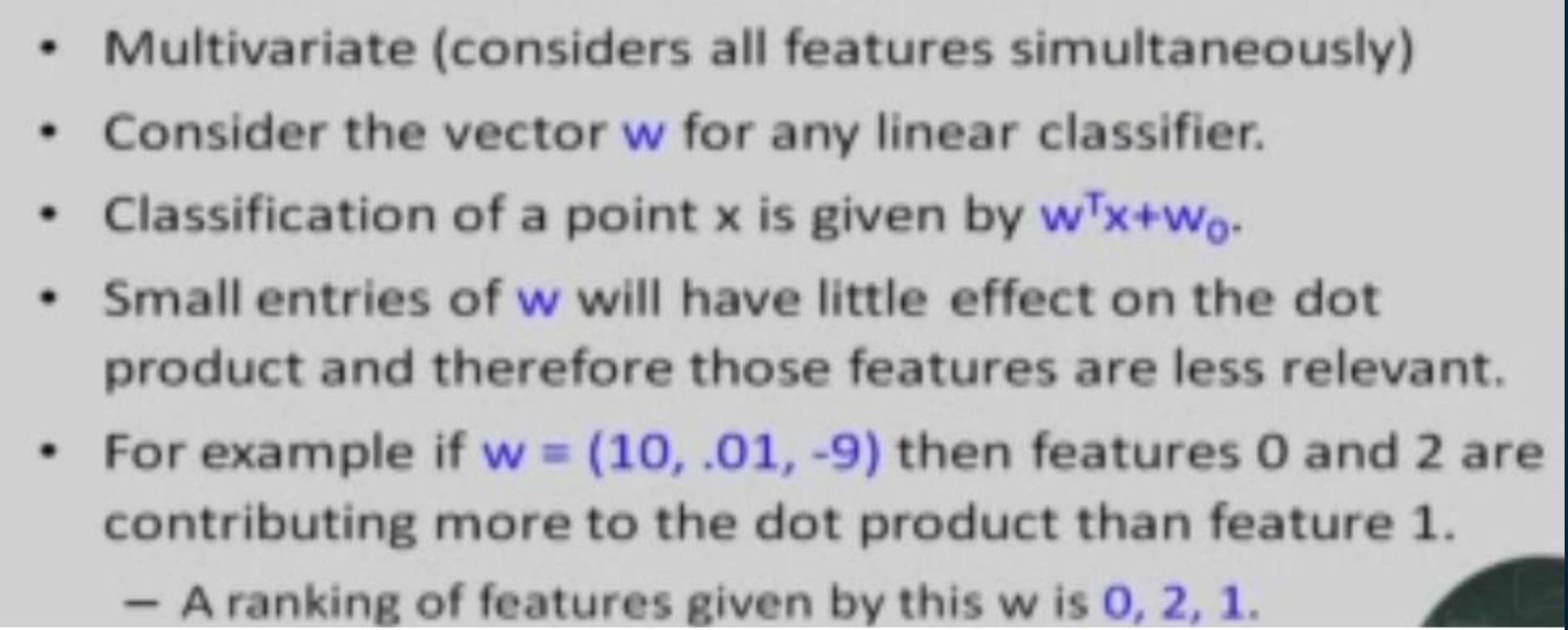


- Difference in means divided by difference in standard deviation between the two classes

$$S2N(X,Y) = (\mu_X - \mu_Y)/(\sigma_X - \sigma_Y)$$

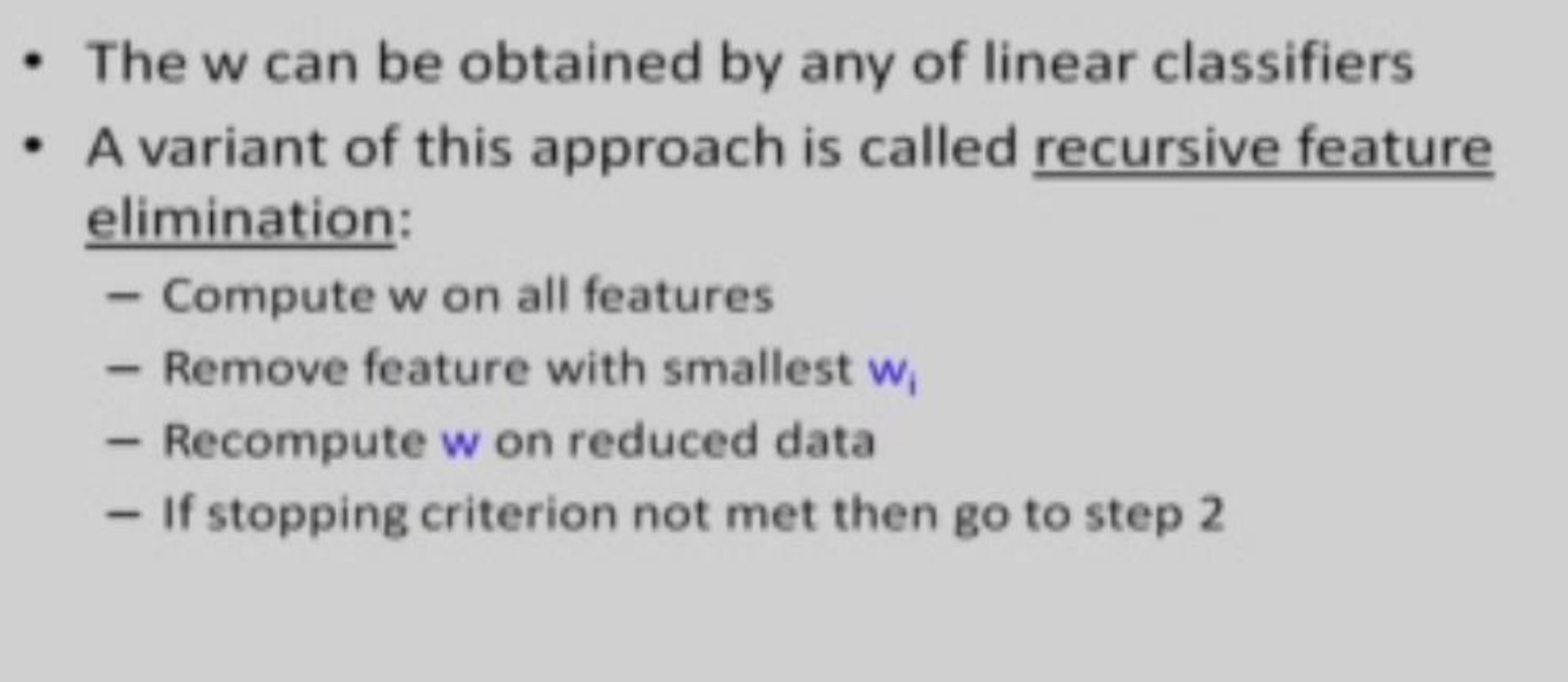
- Large values indicate a strong correlation

Multivariate feature selection



- Multivariate (considers all features simultaneously)
- Consider the vector w for any linear classifier.
- Classification of a point x is given by w^Tx+w_0.
- Small entries of w will have little effect on the dot product and therefore those features are less relevant.
- For example if w = (10, .01, -9) then features 0 and 2 are contributing more to the dot product than feature 1.
  - A ranking of features given by this w is 0, 2, 1.

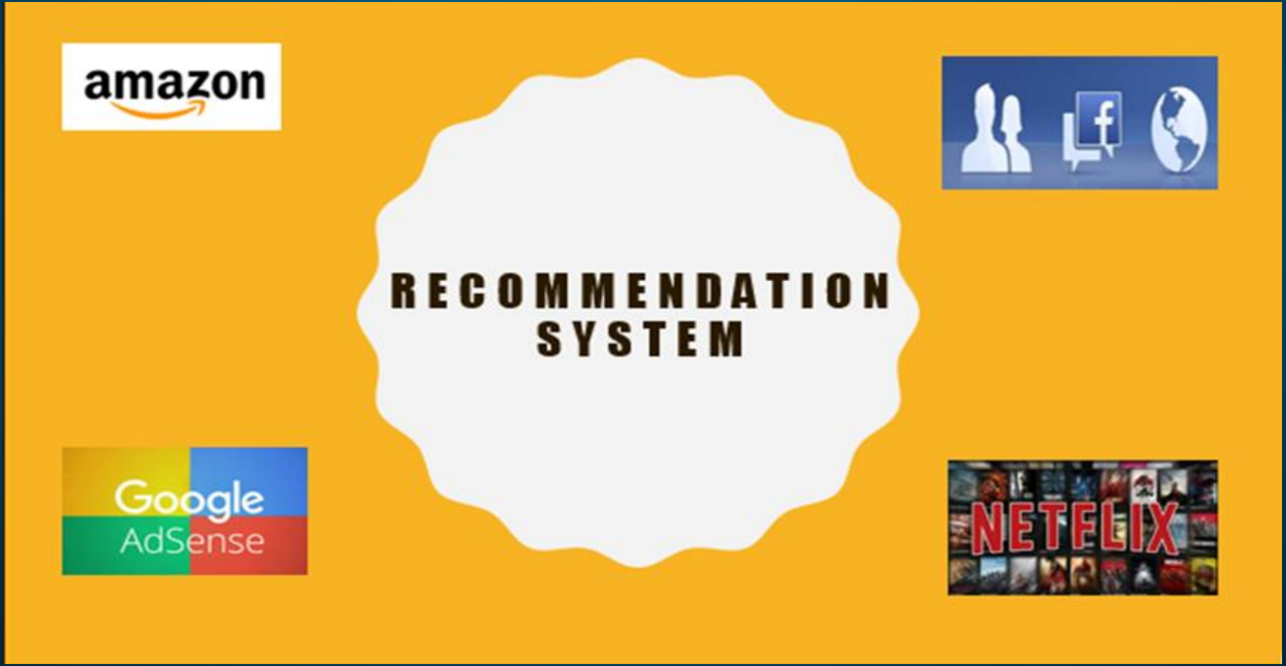
Multivariate feature selection



- The w can be obtained by any of linear classifiers
- A variant of this approach is called recursive feature elimination:
  - Compute w on all features
  - Remove feature with smallest w_i
  - Recompute w on reduced data
  - If stopping criterion not met then go to step 2

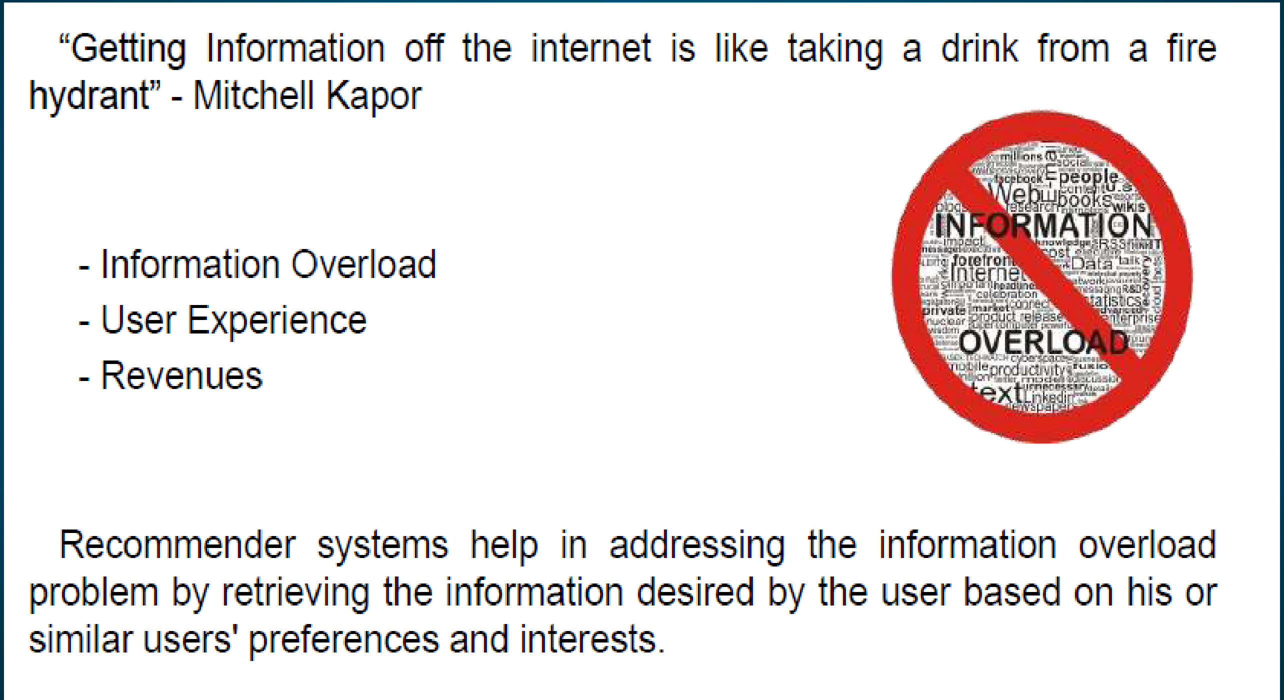
Collaborative Filtering Based Recommendation System

- It is instance based learning



Recommendation Systems

Why there is a need?



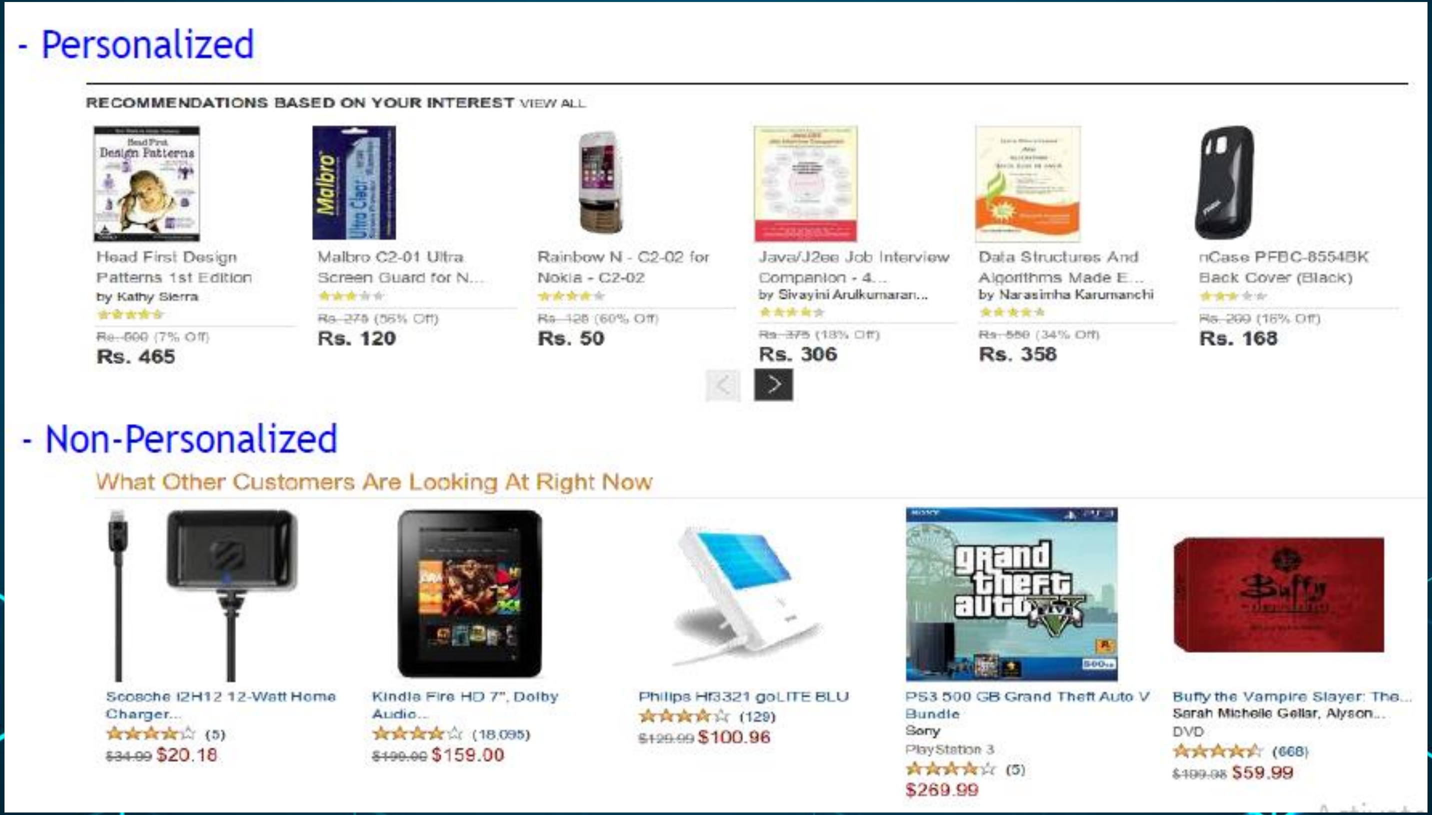
“Getting Information off the internet is like taking a drink from a fire hydrant” - Mitchell Kapor

- Information Overload
- User Experience
- Revenues

INFORMATION
OVERLOAD

Recommender systems help in addressing the information overload problem by retrieving the information desired by the user based on his or similar users' preferences and interests.

Types of Recommendation System



- Personalized

RECOMMENDATIONS BASED ON YOUR INTEREST VIEW ALL

Head First Design Patterns 1st Edition
by Kathy Sierra
★★★★★
~~Rs. 500~~ (7% Off)
Rs. 465

Malbro C2-01 Ultra Screen Guard for N...
★★★☆☆
~~Rs. 275~~ (56% Off)
Rs. 120

Rainbow N - C2-02 for Nokia - C2-02
★★★★☆
~~Rs. 125~~ (60% Off)
Rs. 50

Java/J2ee Job Interview Companion - 4...
by Sivayini Arulkumaran...
★★★★☆
~~Rs. 375~~ (18% Off)
Rs. 306

Data Structures And Algorithms Made E...
by Narasimha Karumanchi
★★★★★
~~Rs. 540~~ (34% Off)
Rs. 358

nCase PFBC-8554BK Back Cover (Black)
★★★☆☆
~~Rs. 200~~ (16% Off)
Rs. 168

‹ ›

- Non-Personalized

What Other Customers Are Looking At Right Now

Scosche I2H12 12-Watt Home Charger...
★★★★☆ (5)
~~$34.99~~ $20.18

Kindle Fire HD 7", Dolby Audio...
★★★★☆ (18,095)
~~$199.00~~ $159.00

Philips Hf3321 goLITE BLU
★★★★☆ (129)
~~$129.99~~ $100.96

PS3 500 GB Grand Theft Auto V Bundle
Sony
PlayStation 3
★★★★☆ (5)
$269.99

Buffy the Vampire Slayer: The...
Sarah Michelle Gellar, Alyson...
DVD
★★★★☆ (668)
~~$199.98~~ $59.99

Techniques : Data Acquisition

01

1. Explicit Data
- Customer Ratings
- Feedback
- Demographics
Physiographics

02

2. Implicit Data
- Purchase History
Click or Browse History

03

3. Product Information
- Product Taxonomy
- Product Attributes
- Product Descriptions

Techniques : Recommendation Generation

1. Collaborative Filtering method finds a subset of users who have similar tastes and preferences to the target user and use this subset for offering recommendations.

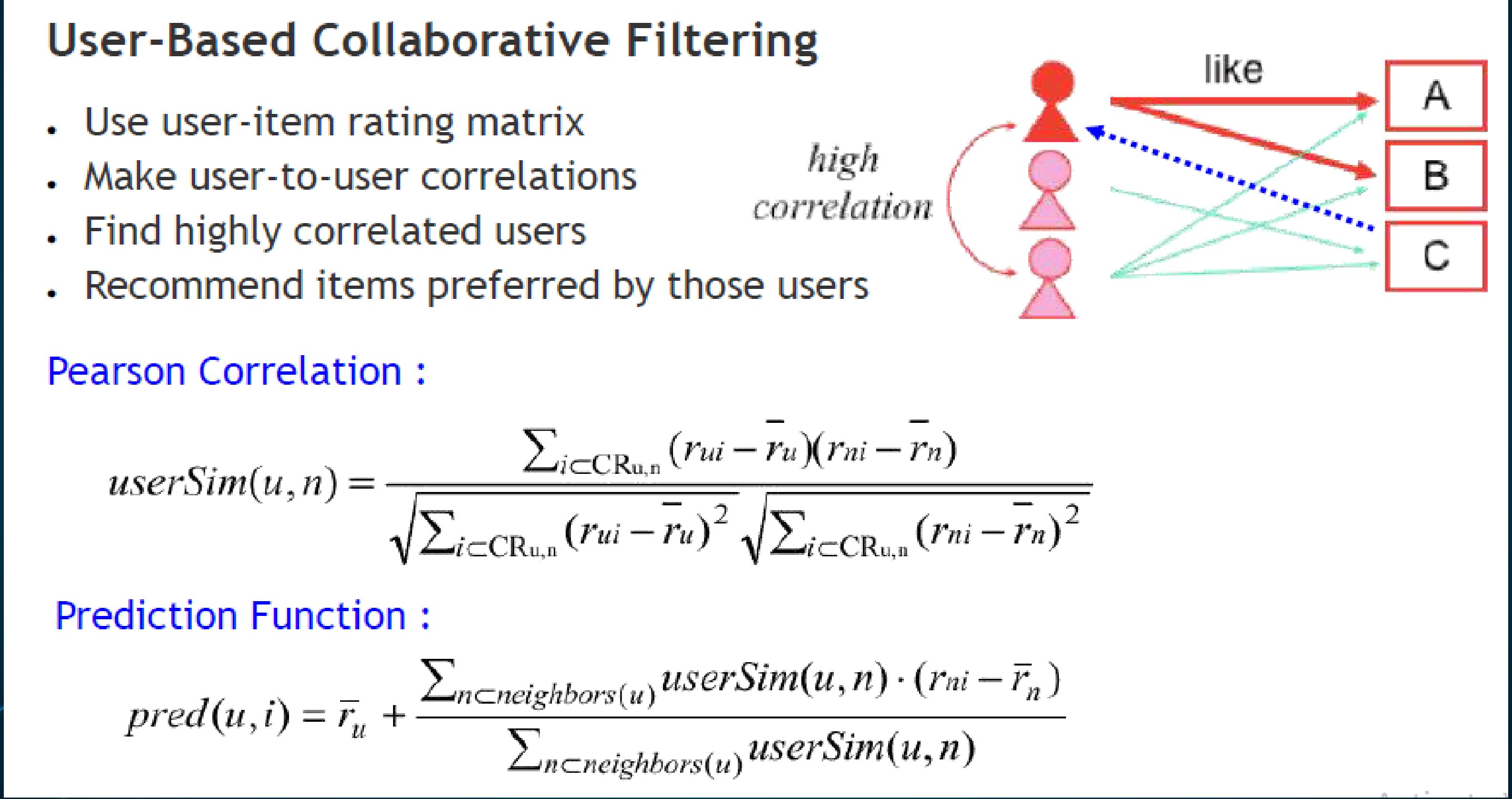
Basic Assumptions :

- Users with similar interests have common preferences.
- Sufficiently large number of user preferences are available.

Main Approaches :

- User Based
- Item Based

Techniques : Recommendation Generation



User-Based Collaborative Filtering

- Use user-item rating matrix
- Make user-to-user correlations
- Find highly correlated users
- Recommend items preferred by those users

high correlation
like
A
B
C

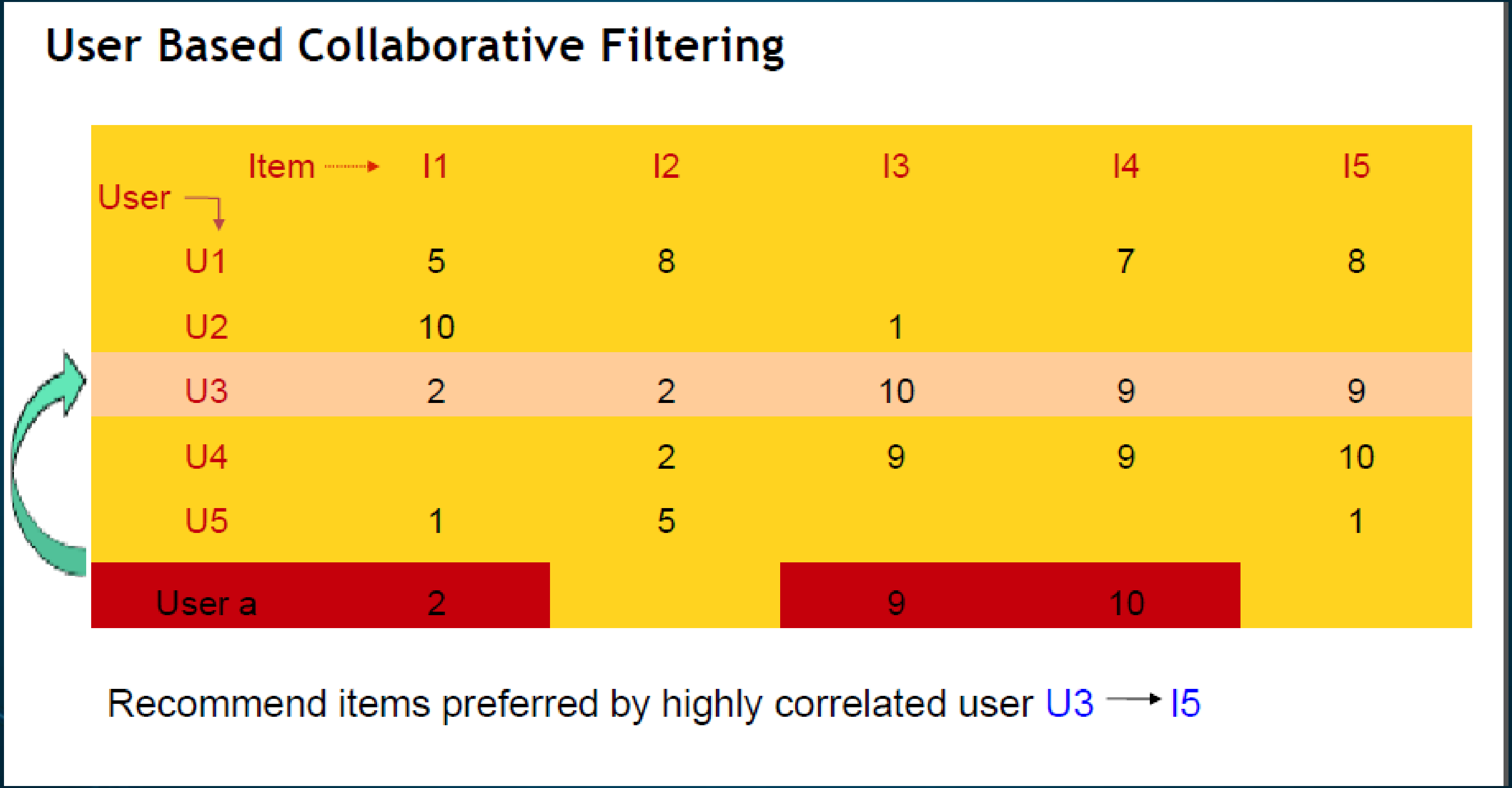
Pearson Correlation :

$$userSim(u,n)=\frac{\sum_{i\subset \mathrm{CR}_{u,n}}(r_{ui}-\bar{r}_u)(r_{ni}-\bar{r}_n)}{\sqrt{\sum_{i\subset \mathrm{CR}_{u,n}}(r_{ui}-\bar{r}_u)^2}\sqrt{\sum_{i\subset \mathrm{CR}_{u,n}}(r_{ni}-\bar{r}_n)^2}}$$

Prediction Function :

$$pred(u,i)=\bar{r}_u+\frac{\sum_{n\subset neighbors(u)} userSim(u,n)\cdot(r_{ni}-\bar{r}_n)}{\sum_{n\subset neighbors(u)} userSim(u,n)}$$

Types of Recommendation System



User Based Collaborative Filtering

User ↓ / Item →	I1	I2	I3	I4	I5
U1	5	8		7	8
U2	10		1		
U3	2	2	10	9	9
U4		2	9	9	10
U5	1	5			1
User a	2		9	10	

Recommend items preferred by highly correlated user U3 → I5

Types of Recommendation System

User Based Collaborative Filtering

● Advantage :

- No knowledge about item features needed

● Problems :

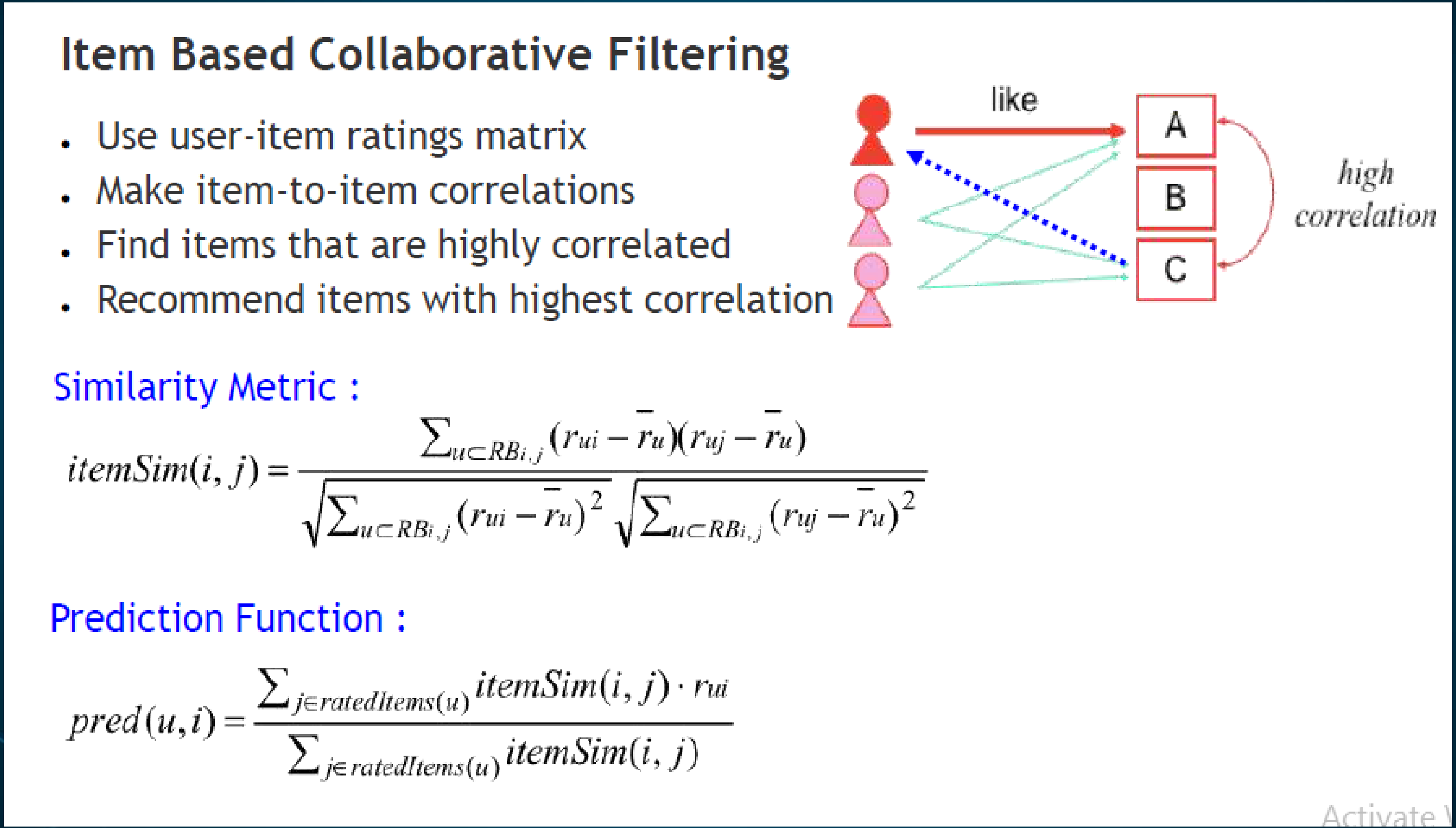
- New user cold start problem
- New item cold start problem: items with few rating cannot easily be recommended
- Sparsity problem: If there are many items to be recommended, user/rating matrix is sparse and it hard to find the users who have rated the same item.

The **sparsity problem** occurs when transactional or feedback data is sparse and insufficient for identifying neighbors and it is a major **issue** limiting the quality of recommendations and the applicability of collaborative filtering in general.

- Popularity Bias: Tend to recommend only popular items.

e.g. RINGO, GroupLens

Types of Recommendation System



Item Based Collaborative Filtering

- Use user-item ratings matrix
- Make item-to-item correlations
- Find items that are highly correlated
- Recommend items with highest correlation

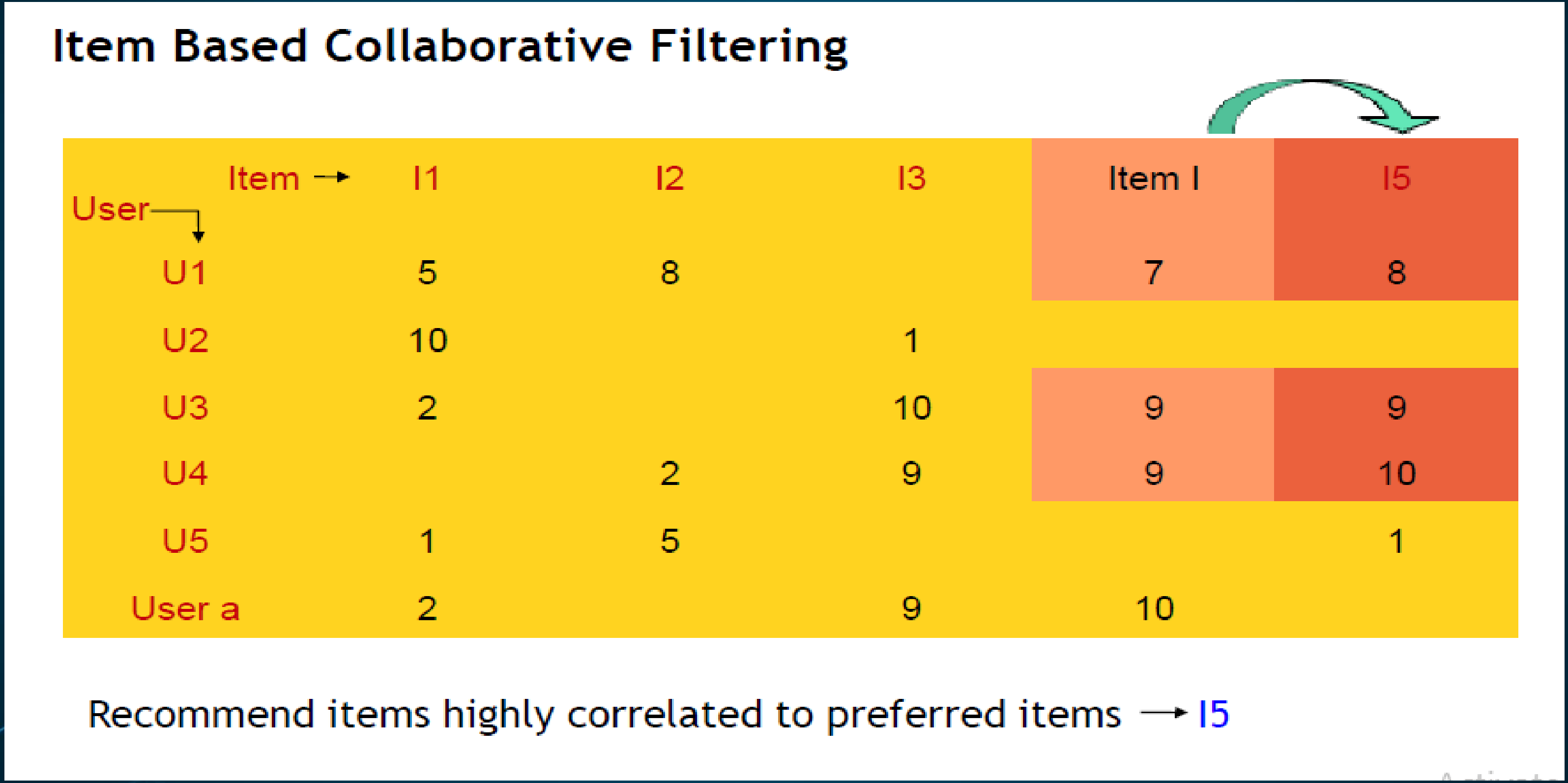
Similarity Metric :

$$itemSim(i,j) = \frac{\sum_{u \subset RB_{i,j}} (r_{ui} - \bar{r}_u)(r_{uj} - \bar{r}_u)}{\sqrt{\sum_{u \subset RB_{i,j}} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{u \subset RB_{i,j}} (r_{uj} - \bar{r}_u)^2}}$$

Prediction Function :

$$pred(u,i) = \frac{\sum_{j \in ratedItems(u)} itemSim(i,j) \cdot r_{ui}}{\sum_{j \in ratedItems(u)} itemSim(i,j)}$$

Types of Recommendation System



Item Based Collaborative Filtering

User ↓ / Item →	I1	I2	I3	Item I	I5
U1	5	8		7	8
U2	10		1		
U3	2		10	9	9
U4		2	9	9	10
U5	1	5			1
User a	2		9	10	

Recommend items highly correlated to preferred items → I5

Types of Recommendation System

Item Based Collaborative Filtering

● Advantage :

- No knowledge about item features needed
- Better scalability, because correlations between limited number of items instead of very large number of users
- Reduced sparsity problem

● Problems :

- New user cold start problem
- New item cold start problem

e.g. Amazon, eBay

Types of Recommendation System

2. Recommendations are based on the content of items rather on other user's   opinion.
User Profiles: Create user profiles to describe the types of items that user prefers.
e.g. User1 likes sci-fi, action and comedy.
Recommendation on the basis of keywords are also classified under content based.
e.g. IMDB, Last.fm

Types of Recommendation System

Content Based Systems Cont'd...

Advantage :

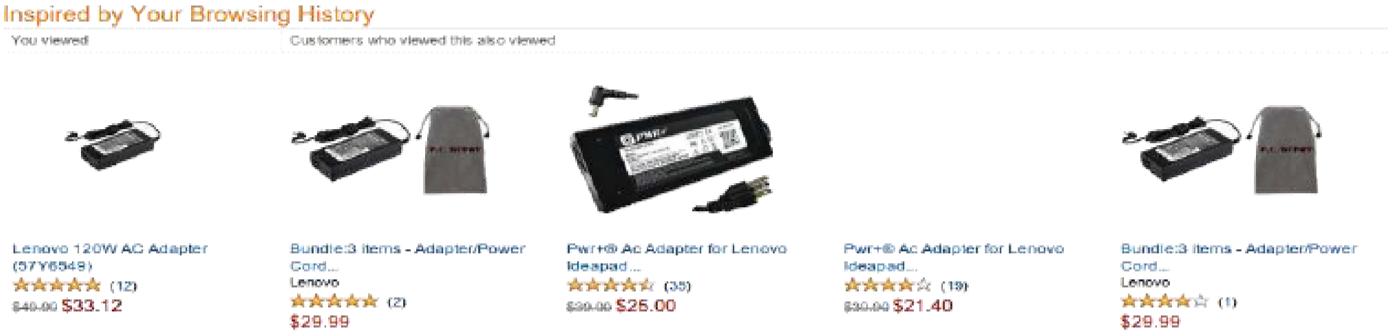
- No need for data on other users. No cold start and sparsity.
- Able to recommend users with unique taste.
- Able to recommend new and unpopular items.
- Can provide explanation for recommendation.

Limitations:

- Data should be in structured format.
- Unable to use quality judgments from other users

Types of Recommendation System

1. Amazon



Inspired by Your Browsing History

You viewed

Customers who viewed this also viewed

Lenovo 120W AC Adapter (57Y6549)
★★★★★ (12)
~~$40.00~~ $33.12

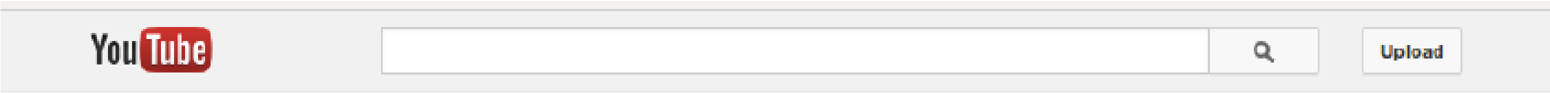
Bundle:3 items - Adapter/Power Cord...
Lenovo
★★★★★ (2)
$29.99

Pwr+® Ac Adapter for Lenovo Ideapad...
★★★★½ (35)
~~$30.00~~ $26.00

Pwr+® Ac Adapter for Lenovo Ideapad...
★★★★☆ (19)
~~$30.00~~ $21.40

Bundle:3 items - Adapter/Power Cord...
Lenovo
★★★★☆ (1)
$29.99

2. YouTube



Collaborative filtering based recommendation system

Let u is a set of users and s is a set of items.

P is utility function that finds out rating of item by users

P: u X s R

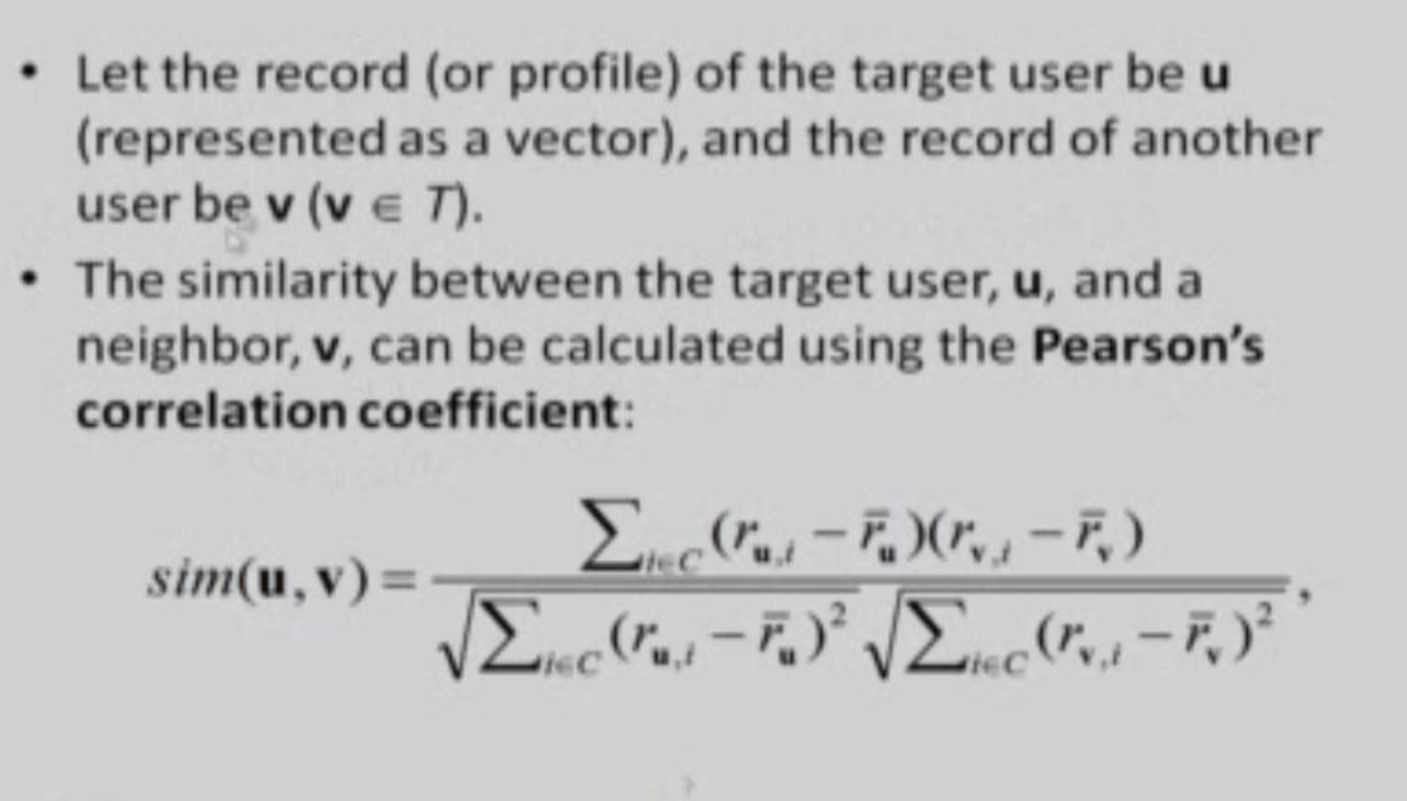
Its indicating that for a user u and for item s , the rating of the user for that item.

Now, Learn P from data.

Where training data set is past rating of users for rating prediction problem, or past purchase history .

Based on P to predict utility value of each item to each user.

Collaborative filtering based recommendation system

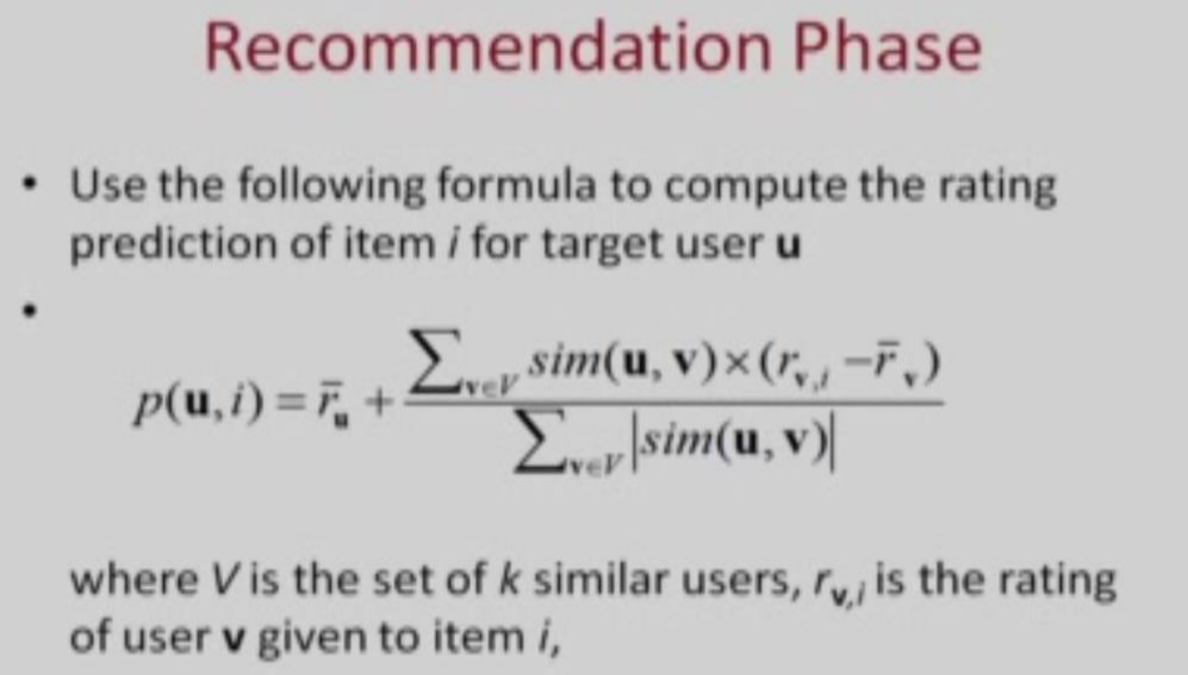


- Let the record (or profile) of the target user be **u** (represented as a vector), and the record of another user be **v** (**v** $\in T$).
- The similarity between the target user, **u**, and a neighbor, **v**, can be calculated using the **Pearson's correlation coefficient**:

$$sim(\mathbf{u},\mathbf{v}) = \frac{\sum_{i\in C}(r_{\mathbf{u},i}-\bar{r}_{\mathbf{u}})(r_{\mathbf{v},i}-\bar{r}_{\mathbf{v}})}{\sqrt{\sum_{i\in C}(r_{\mathbf{u},i}-\bar{r}_{\mathbf{u}})^2}\sqrt{\sum_{i\in C}(r_{\mathbf{v},i}-\bar{r}_{\mathbf{v}})^2}},$$

Collaborative filtering based recommendation system

In the previous phase , we have to find the similar user ,using KNN algorithms and now



Recommendation Phase

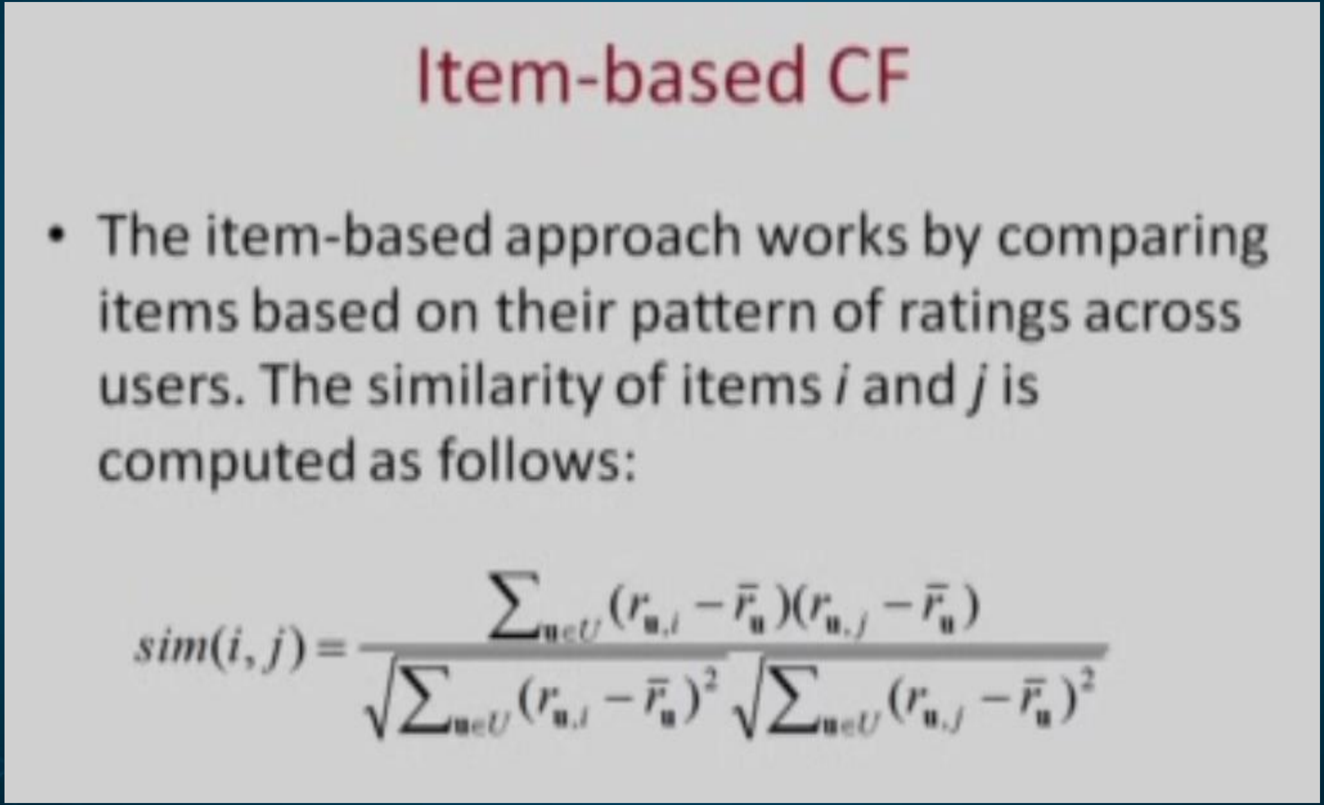
- Use the following formula to compute the rating prediction of item *i* for target user **u**
- $$p(\mathbf{u},i)=\bar{r}_{\mathbf{u}}+\frac{\sum_{\mathbf{v}\in V} sim(\mathbf{u},\mathbf{v})\times(r_{\mathbf{v},i}-\bar{r}_{\mathbf{v}})}{\sum_{\mathbf{v}\in V}\left|sim(\mathbf{u},\mathbf{v})\right|}$$

where *V* is the set of *k* similar users, $r_{\mathbf{v},i}$ is the rating of user **v** given to item *i*,

Collaborative filtering based recommendation system

- One of the drawback of user based recommendation system is , when number of users increased , then it is very difficult to handle it.
- Amazon has Million of users.

Collaborative filtering based recommendation system

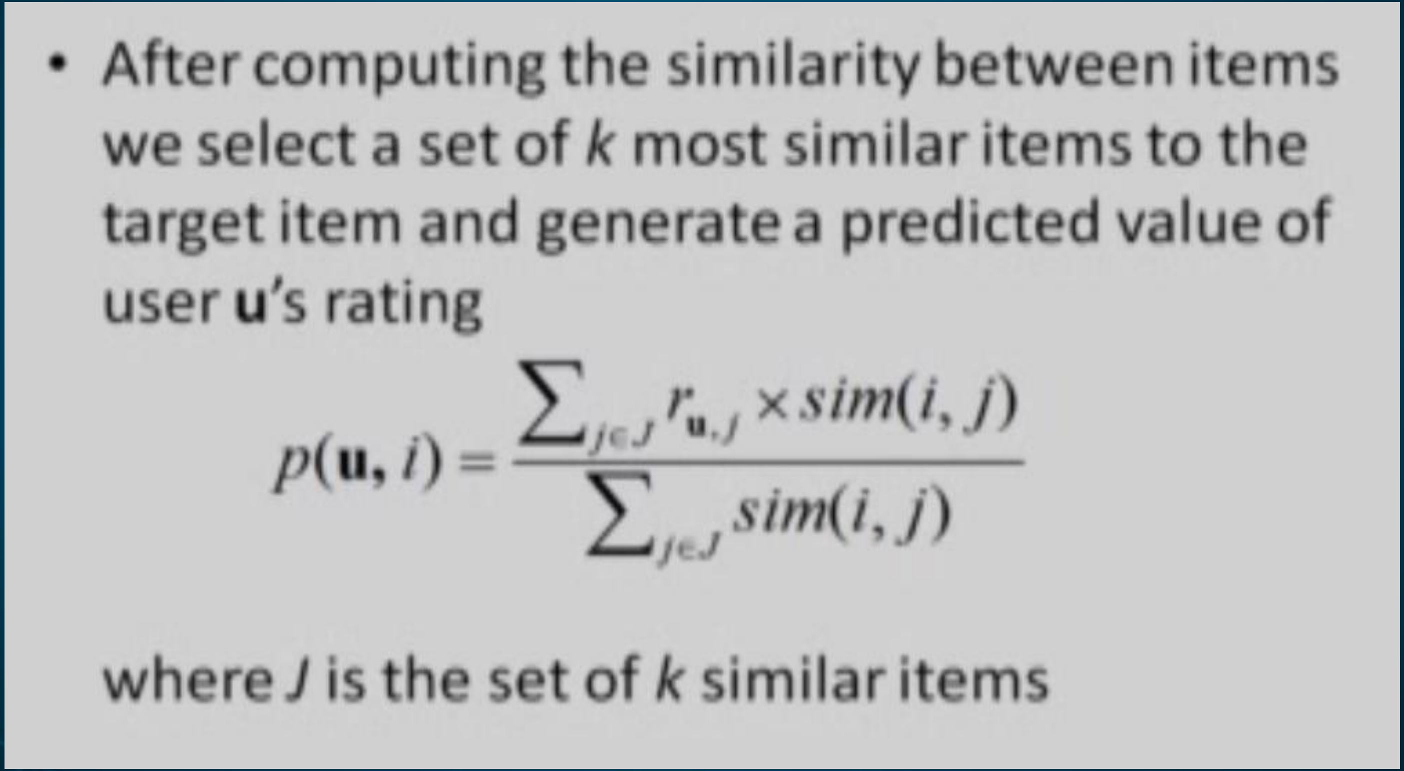


Item-based CF

- The item-based approach works by comparing items based on their pattern of ratings across users. The similarity of items *i* and *j* is computed as follows:

$$sim(i,j)=\frac{\sum_{u\in U}(r_{u,i}-\bar{r}_u)(r_{u,j}-\bar{r}_u)}{\sqrt{\sum_{u\in U}(r_{u,i}-\bar{r}_u)^2}\sqrt{\sum_{u\in U}(r_{u,j}-\bar{r}_u)^2}}$$

Collaborative filtering based recommendation system

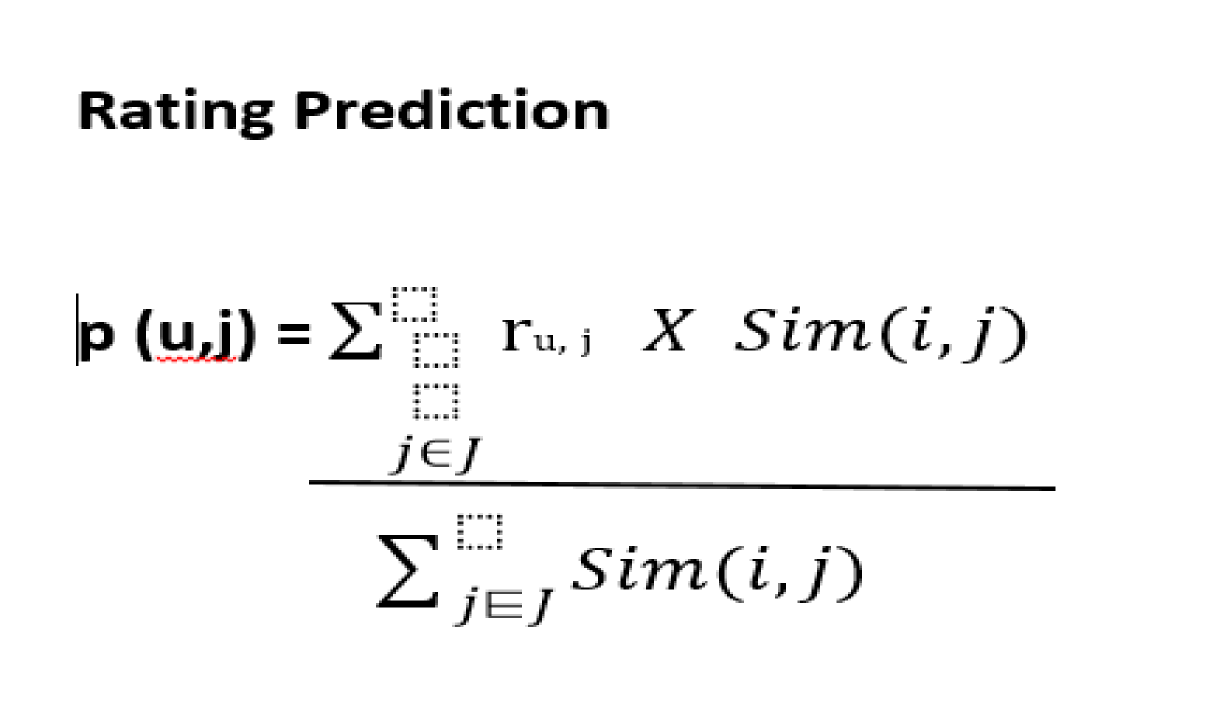


- After computing the similarity between items we select a set of *k* most similar items to the target item and generate a predicted value of user **u**'s rating

$$p(\mathbf{u}, i) = \frac{\sum_{j \in J} r_{\mathbf{u},j} \times sim(i, j)}{\sum_{j \in J} sim(i, j)}$$

where *J* is the set of *k* similar items

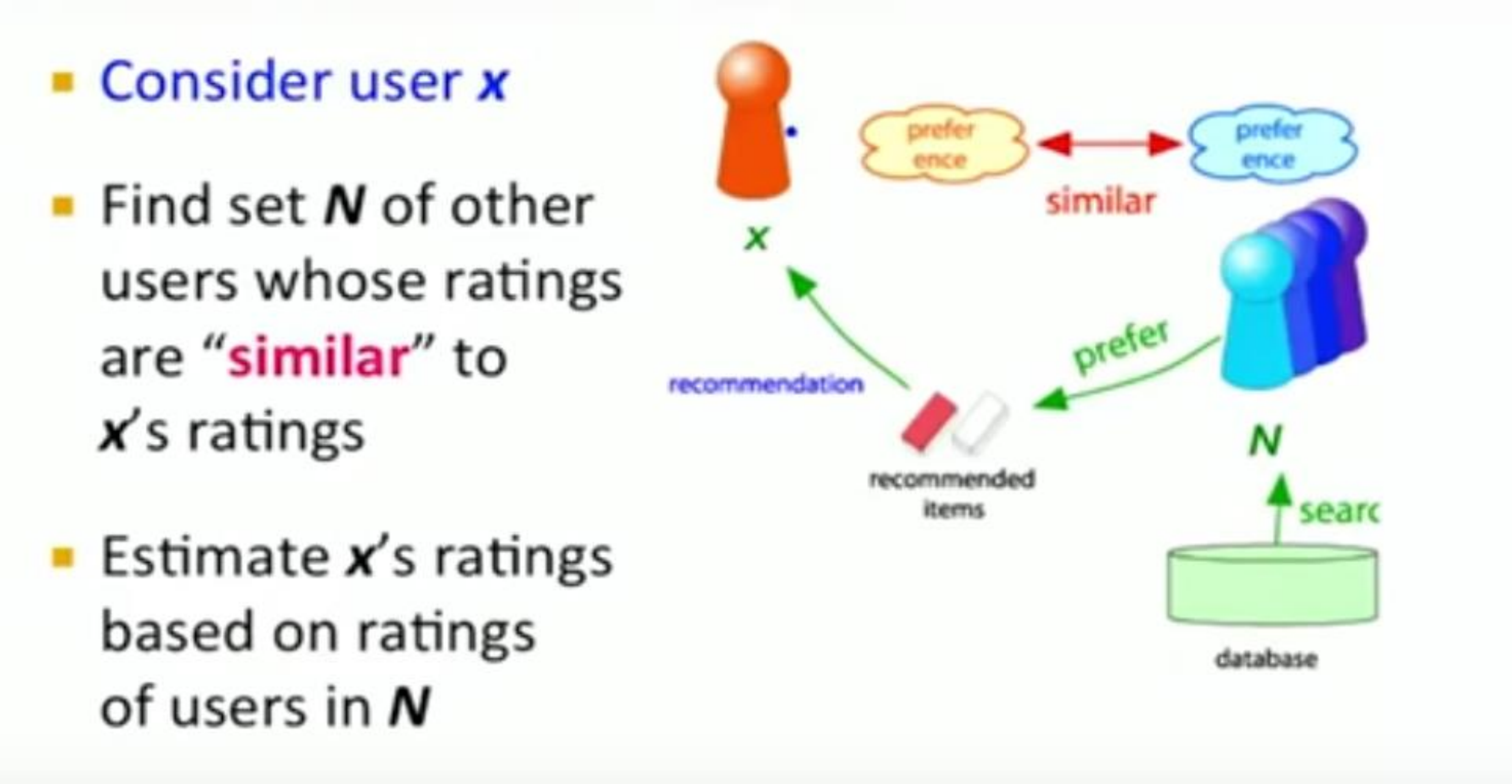
Collaborative filtering based recommendation system



Rating Prediction

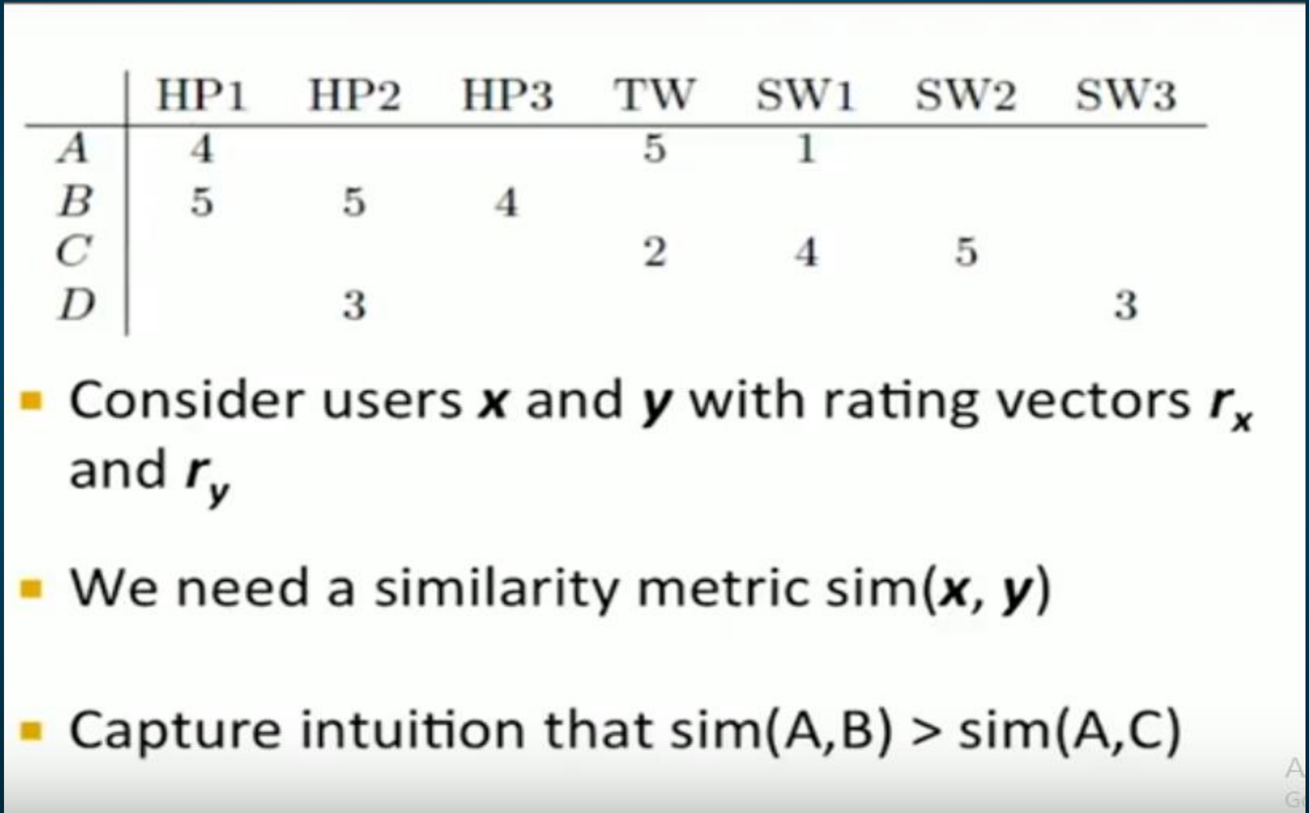
$$p\,(u,j) = \frac{\sum_{j \in J} r_{u,\,j} \;\; X \;\; Sim(i,j)}{\sum_{j \in J} Sim(i,j)}$$

Collaborative Filtering



- Consider user ***x***
- Find set ***N*** of other users whose ratings are "similar" to ***x***'s ratings
- Estimate ***x***'s ratings based on ratings of users in ***N***

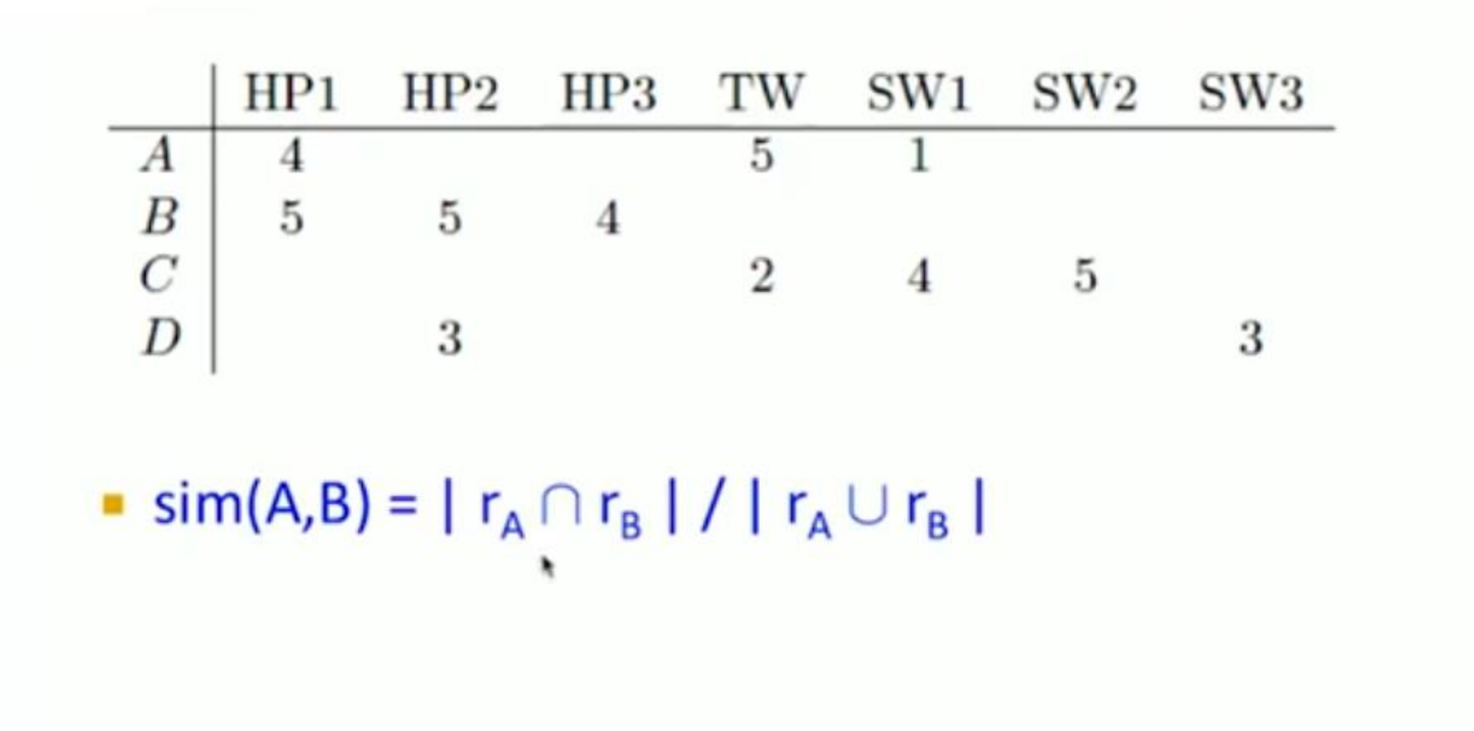
Finding Similar Users



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- Consider users ***x*** and ***y*** with rating vectors $\boldsymbol{r_x}$ and $\boldsymbol{r_y}$
- We need a similarity metric sim(***x***, ***y***)
- Capture intuition that sim(A,B) > sim(A,C)

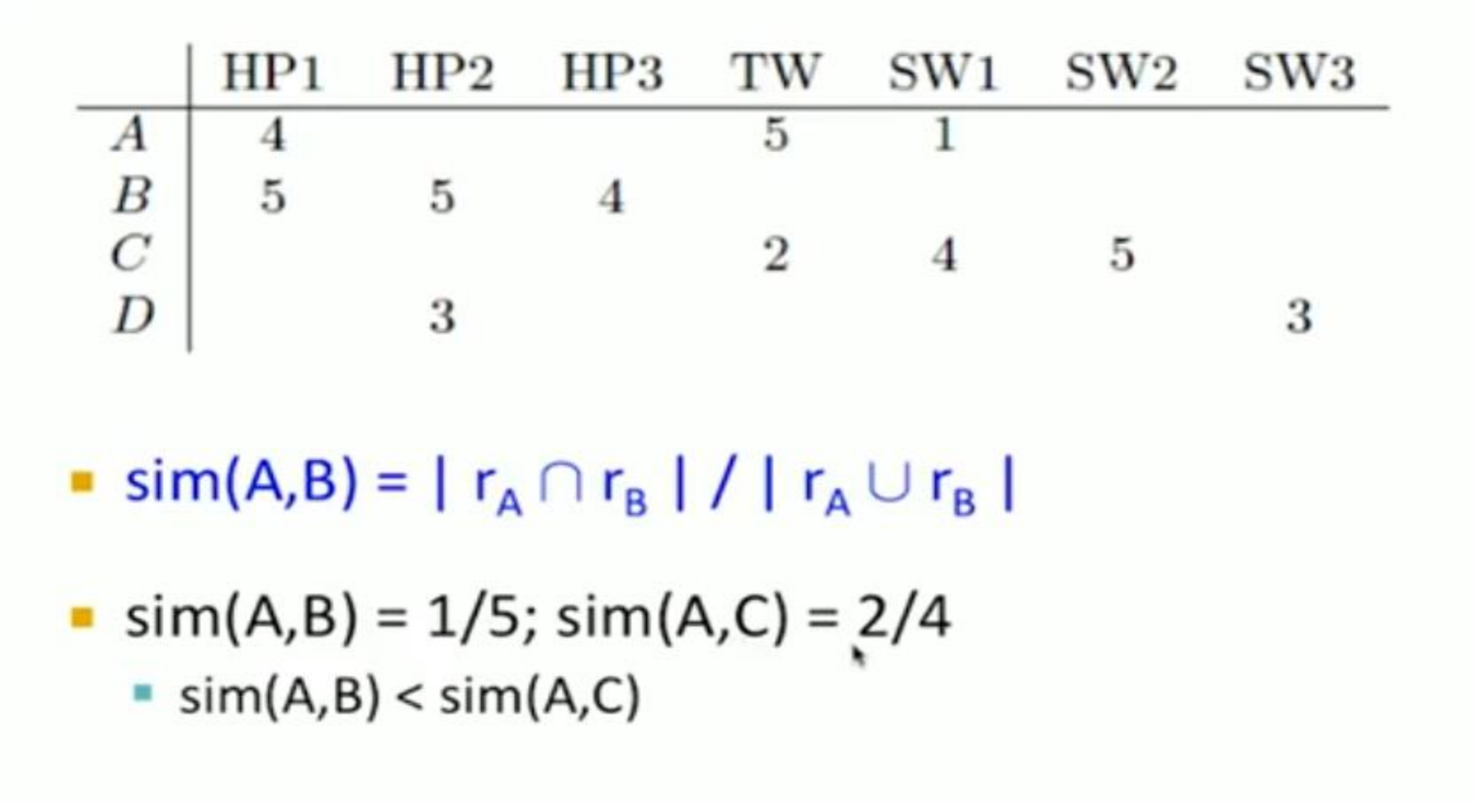
Jaccard Similarity



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- sim(A,B) = $| r_A \cap r_B | / | r_A \cup r_B |$

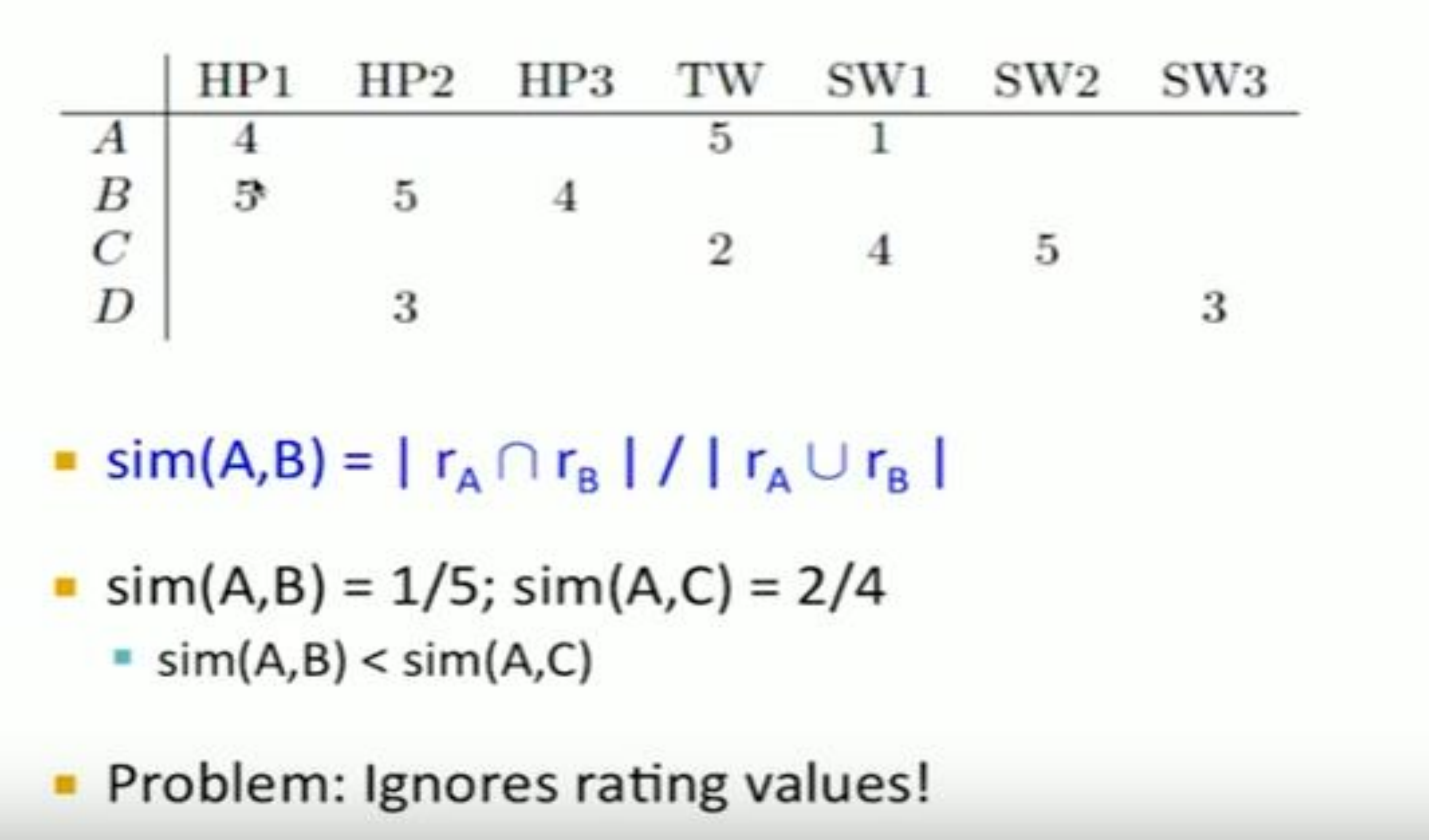
Jaccard Similarity



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- $\text{sim}(A,B) = | r_A \cap r_B | / | r_A \cup r_B |$
- sim(A,B) = 1/5; sim(A,C) = 2/4
  - sim(A,B) < sim(A,C)

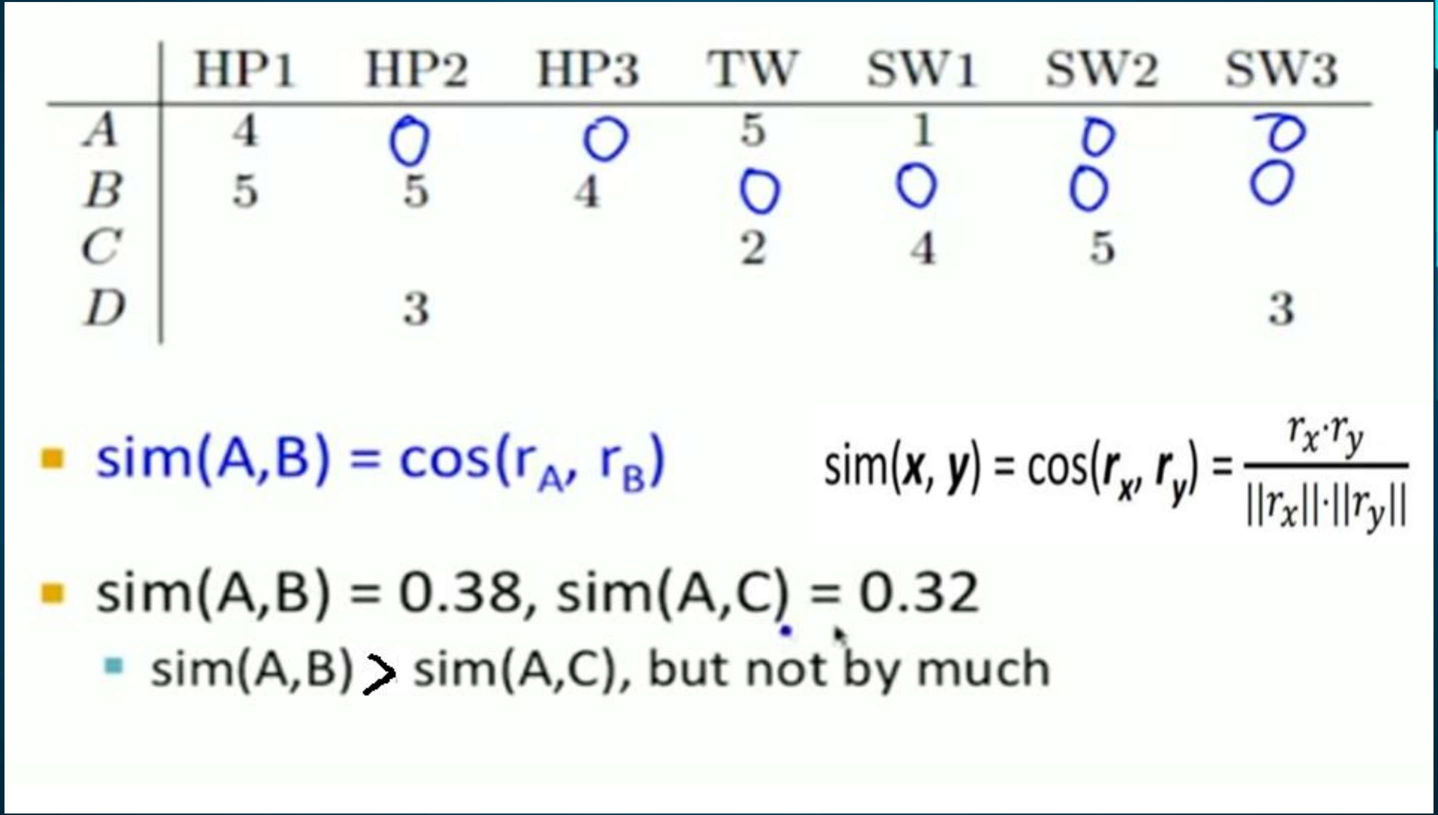
Jaccard Similarity



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- sim(A,B) = $| r_A \cap r_B | / | r_A \cup r_B |$
- sim(A,B) = 1/5; sim(A,C) = 2/4
  - sim(A,B) < sim(A,C)
- Problem: Ignores rating values!

Cosine Similarity



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4	0	0	5	1	0	0
B	5	5	4	0	0	0	0
C				2	4	5	
D		3					3

- sim(A,B) = cos(r_A, r_B)

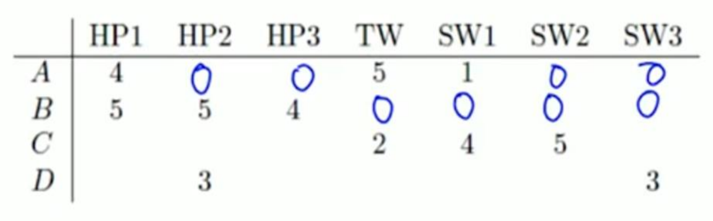
$$\text{sim}(\boldsymbol{x}, \boldsymbol{y}) = \cos(\boldsymbol{r_x}, \boldsymbol{r_y}) = \frac{r_x \cdot r_y}{||r_x|| \cdot ||r_y||}$$

- sim(A,B) = 0.38, sim(A,C) = 0.32
  - sim(A,B) > sim(A,C), but not by much

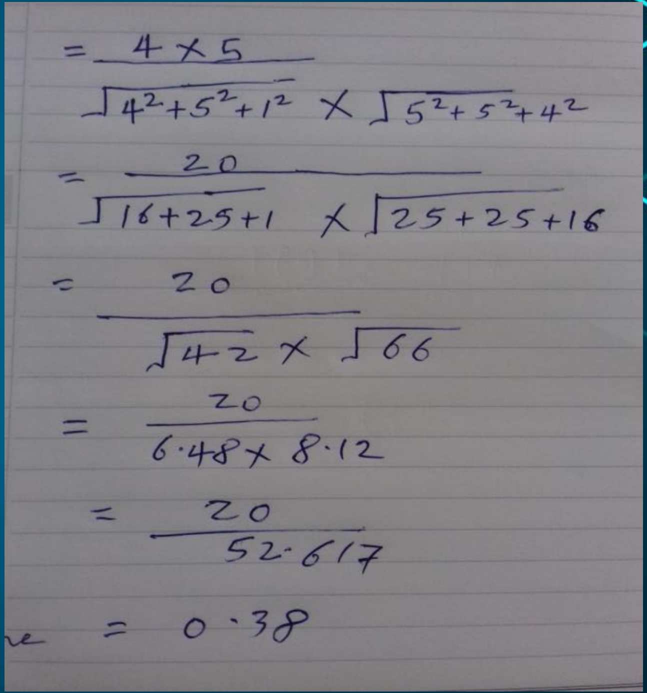
Cosine Similarity: To Solve

$$\text{sim}(\boldsymbol{x}, \boldsymbol{y}) = \cos(\boldsymbol{r_x}, \boldsymbol{r_y}) = \frac{r_x \cdot r_y}{||r_x|| \cdot ||r_y||}$$

$$\text{sim}(A,B) = \cos(r_A, r_B)$$



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4	0	0	5	1	0	0
B	5	5	4	0	0	0	0
C				2	4	5	
D		3					3



$$= \frac{4 \times 5}{\sqrt{4^2+5^2+1^2} \times \sqrt{5^2+5^2+4^2}}$$

$$= \frac{20}{\sqrt{16+25+1} \times \sqrt{25+25+16}}$$

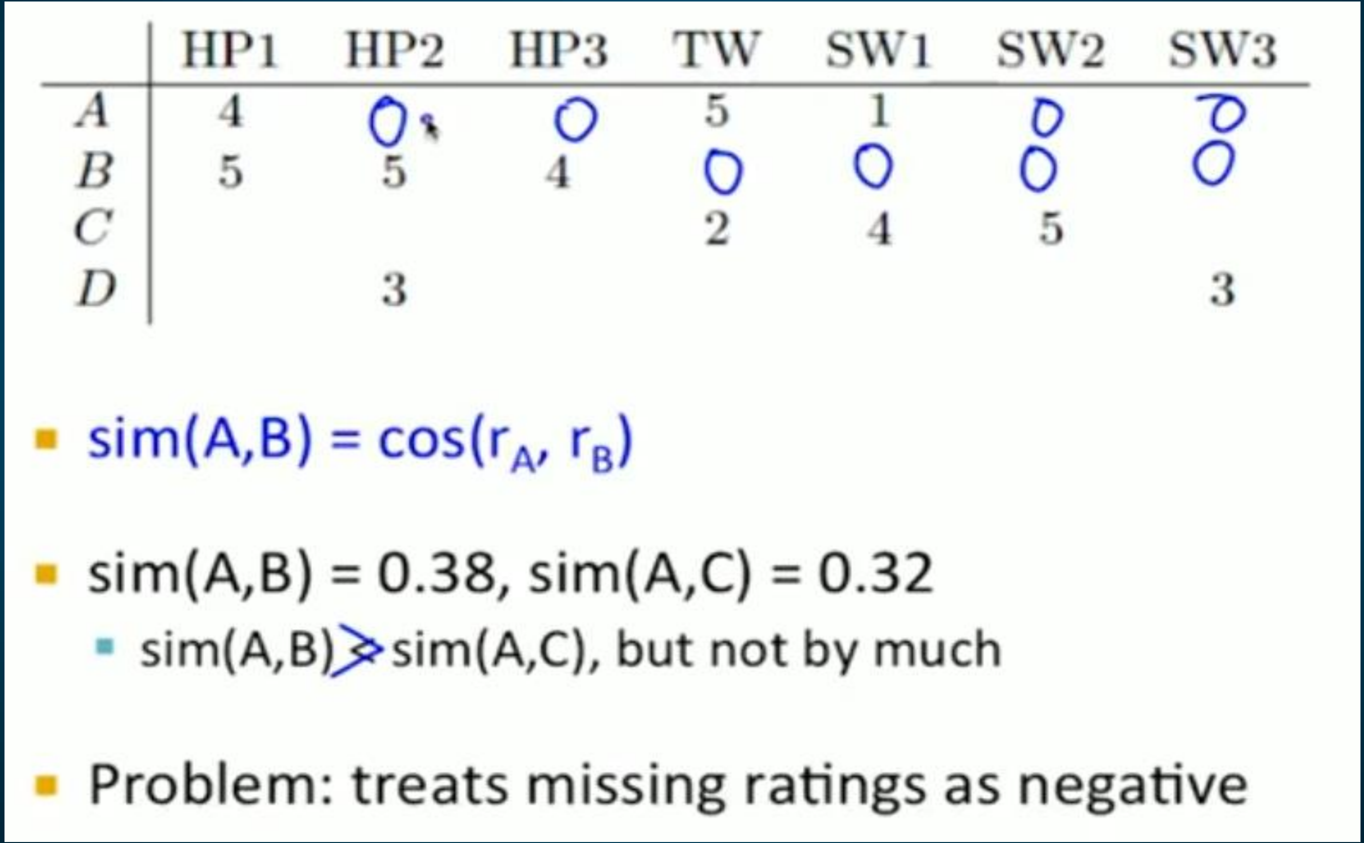
$$= \frac{20}{\sqrt{42} \times \sqrt{66}}$$

$$= \frac{20}{6.48 \times 8.12}$$

$$= \frac{20}{52.617}$$

$$= 0.38$$

Cosine Similarity



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4	0	0	5	1	0	0
B	5	5	4	0	0	0	0
C				2	4	5	
D		3					3

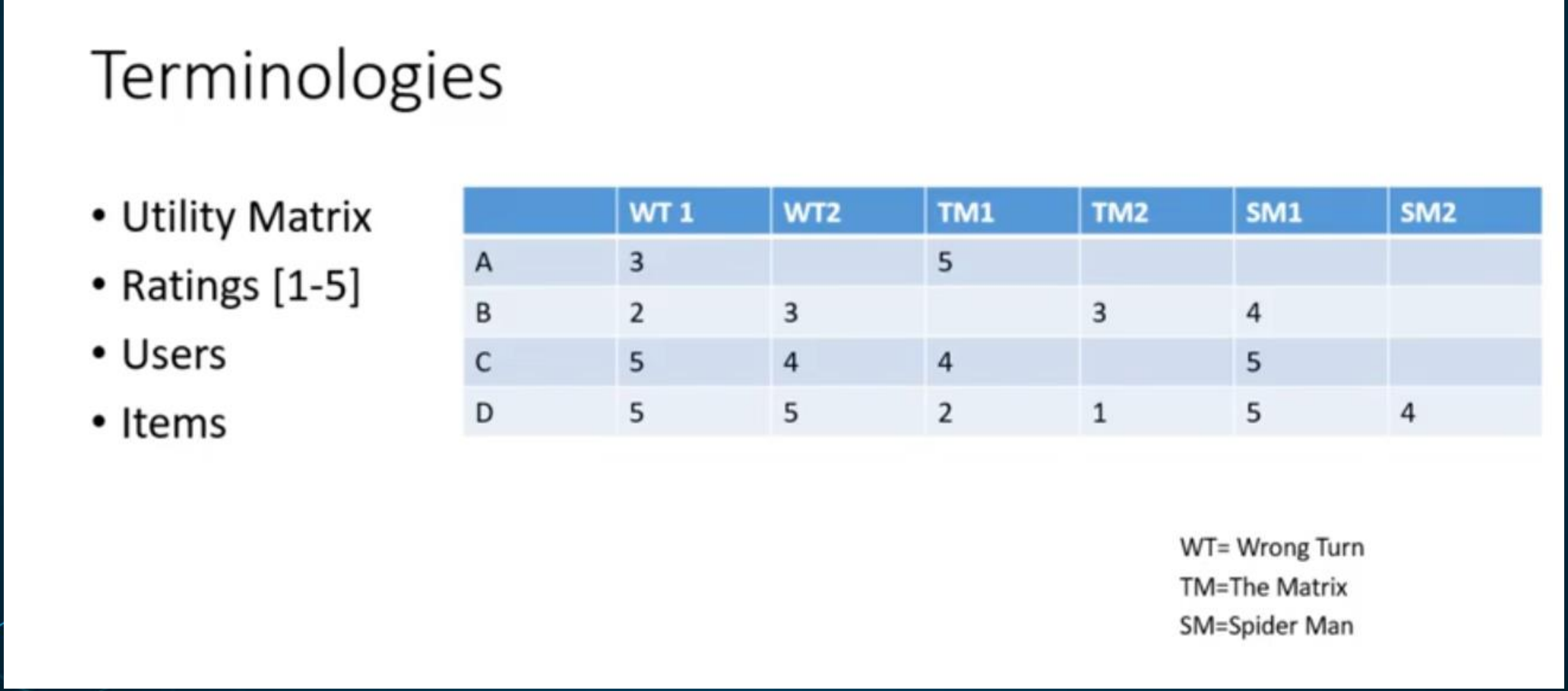
- sim(A,B) = cos(r_A, r_B)

- sim(A,B) = 0.38, sim(A,C) = 0.32
  - sim(A,B) > sim(A,C), but not by much

- Problem: treats missing ratings as negative

Cosine Similarity

- In the above example, there is only one common rating between A and B



Terminologies

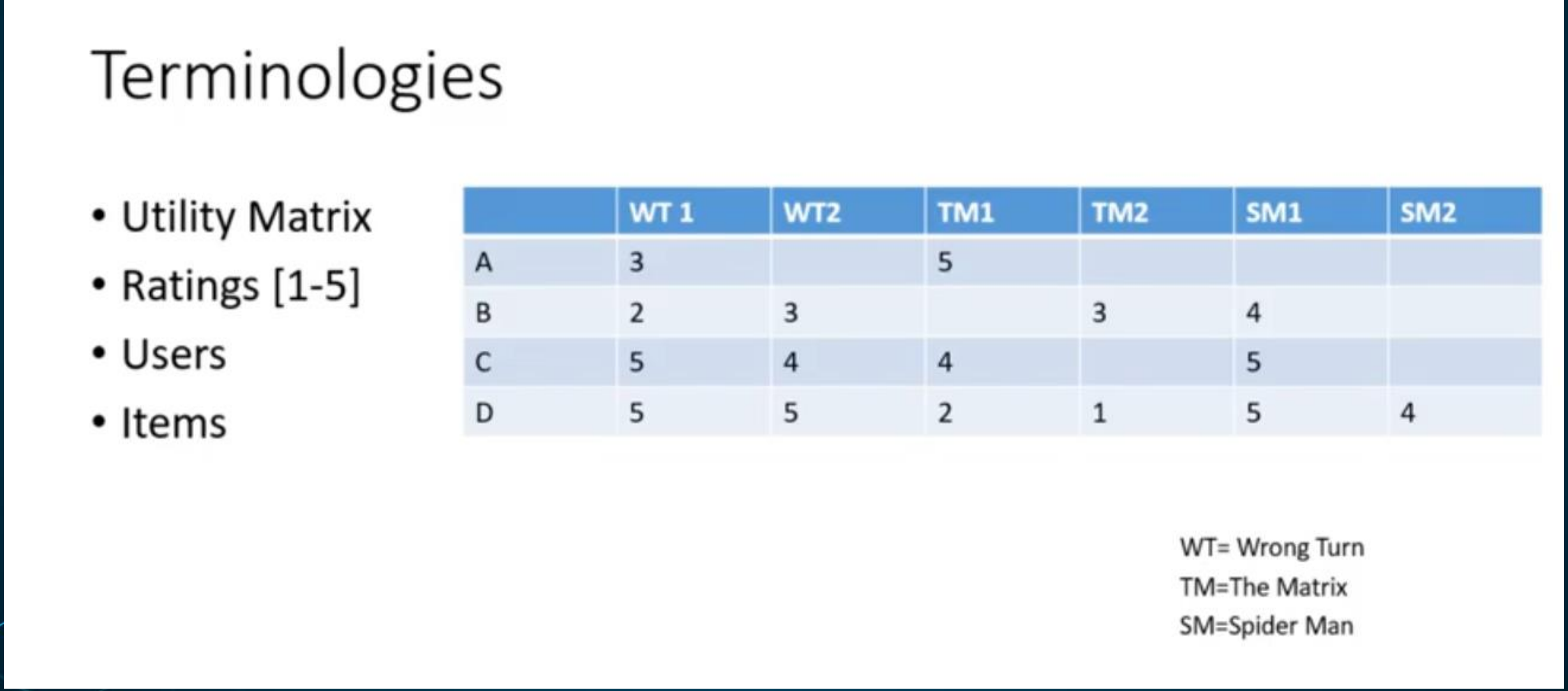
- Utility Matrix
- Ratings [1-5]
- Users
- Items

	WT 1	WT2	TM1	TM2	SM1	SM2
A	3		5			
B	2	3		3	4	
C	5	4	4		5	
D	5	5	2	1	5	4

WT= Wrong Turn
TM=The Matrix
SM=Spider Man

Cosine Similarity

- In this example let us consider A and C , they having two common rating ie. WT1 and TM1



Terminologies

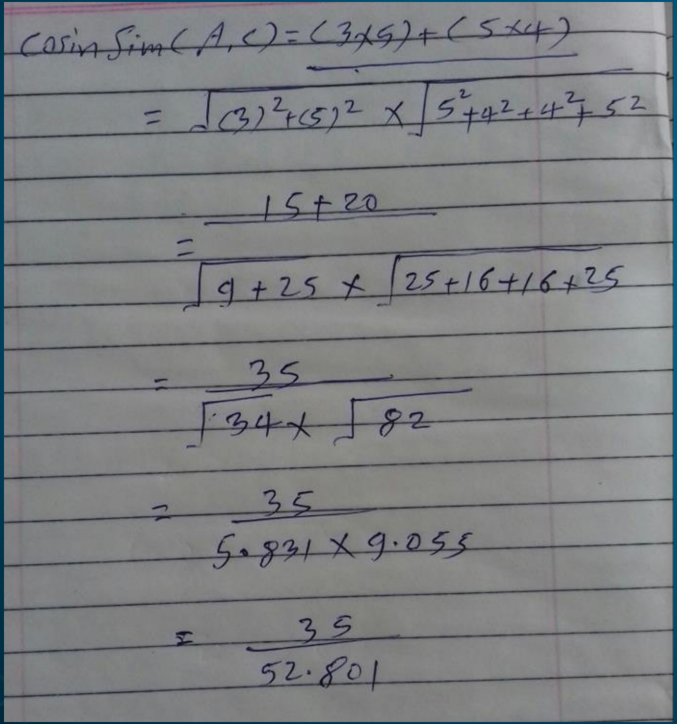
- Utility Matrix
- Ratings [1-5]
- Users
- Items

	WT 1	WT2	TM1	TM2	SM1	SM2
A	3		5			
B	2	3		3	4	
C	5	4	4		5	
D	5	5	2	1	5	4

WT= Wrong Turn
TM=The Matrix
SM=Spider Man

Cosine Similarity

- In this example let us consider A and C , they having two common rating ie. WT1 and TM1



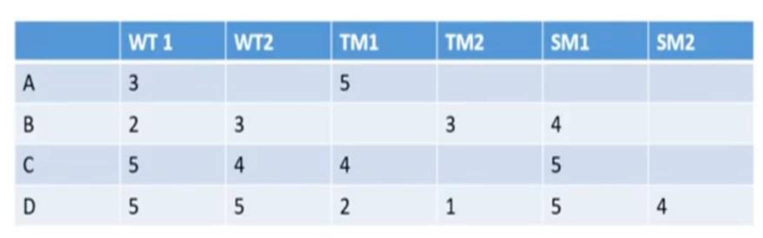
$$\text{cosin Sim}(A,C) = \frac{(3\times5)+(5\times4)}{\sqrt{(3)^2+(5)^2} \times \sqrt{5^2+4^2+4^2+5^2}}$$

$$= \frac{15+20}{\sqrt{9+25} \times \sqrt{25+16+16+25}}$$

$$= \frac{35}{\sqrt{34} \times \sqrt{82}}$$

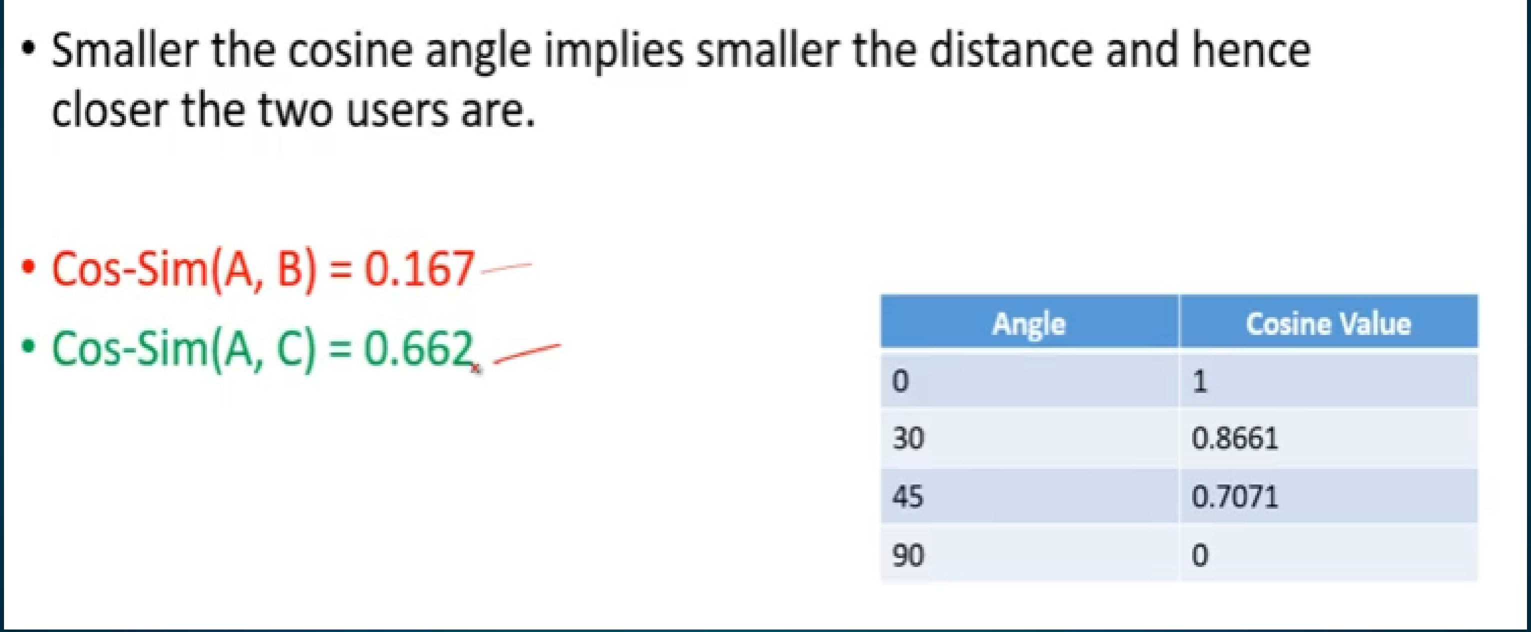
$$= \frac{35}{5.831 \times 9.055}$$

$$= \frac{35}{52.801}$$



	WT 1	WT2	TM1	TM2	SM1	SM2
A	3		5			
B	2	3		3	4	
C	5	4	4		5	
D	5	5	2	1	5	4

Conclusion From Results

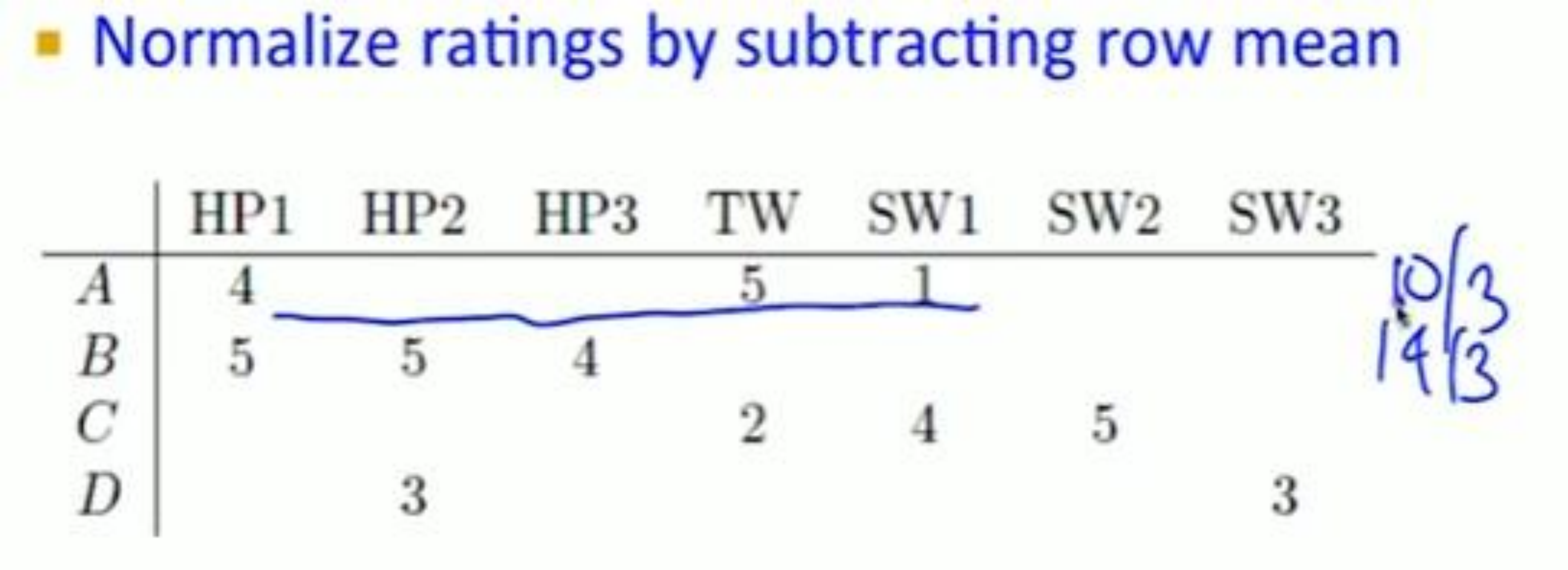


- Smaller the cosine angle implies smaller the distance and hence closer the two users are.

- Cos-Sim(A, B) = 0.167
- Cos-Sim(A, C) = 0.662

Angle	Cosine Value
0	1
30	0.8661
45	0.7071
90	0

Centered Cosine



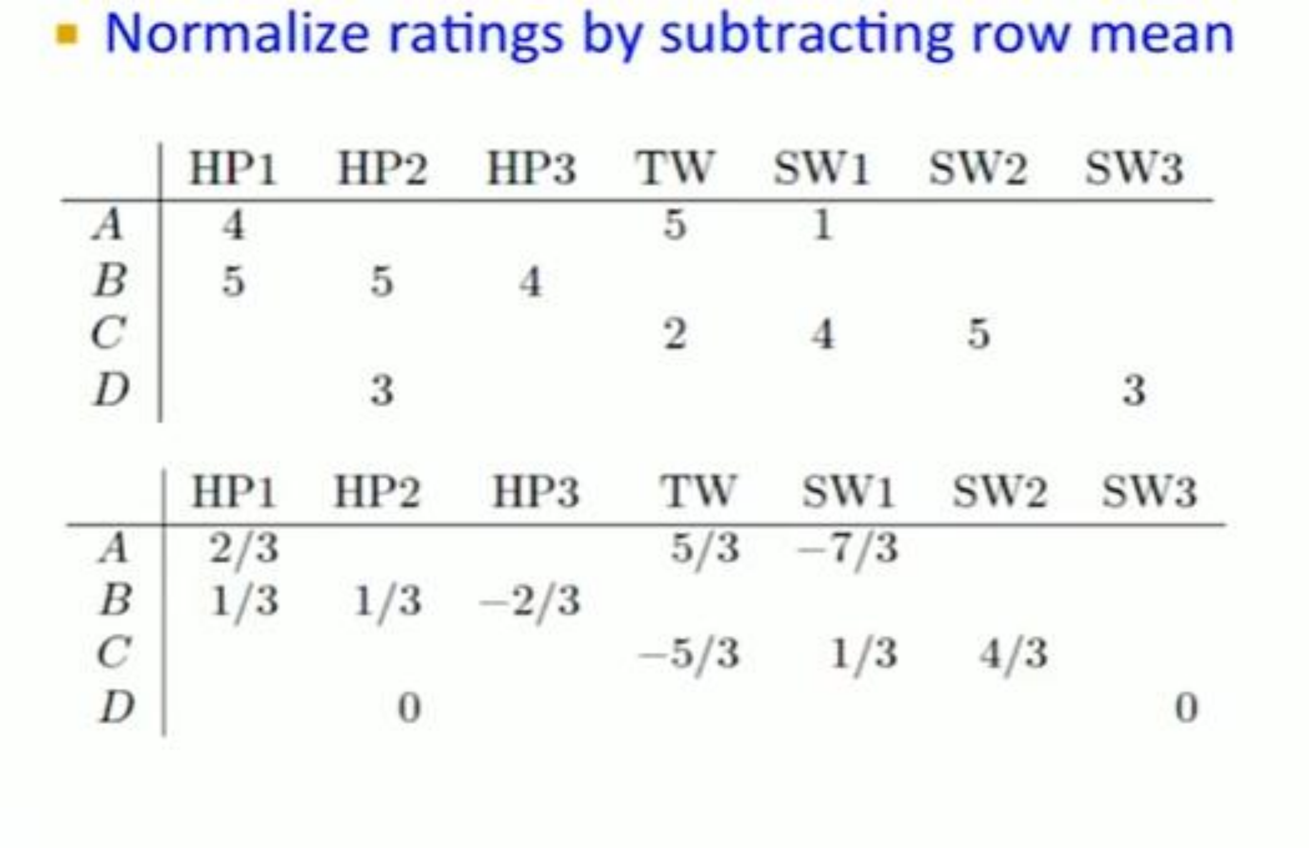
- Normalize ratings by subtracting row mean

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

10/3

14/3

Centered Cosine

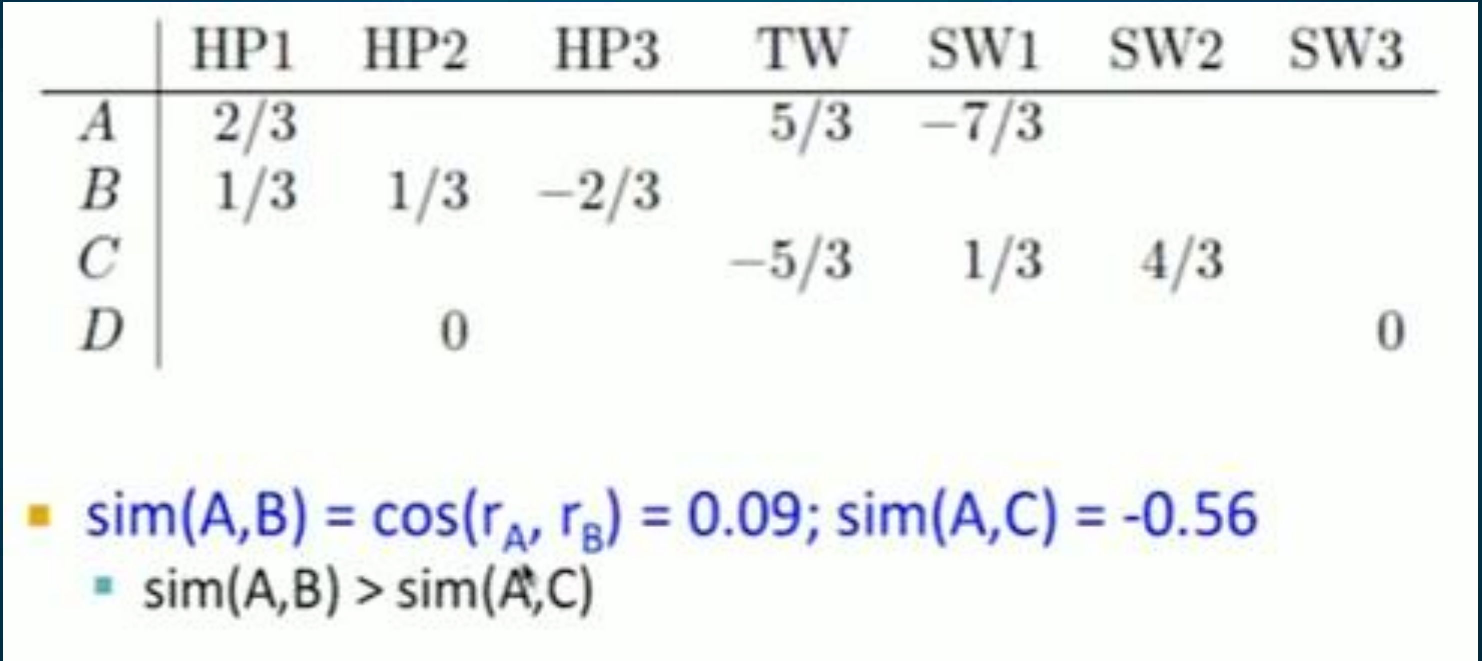


- Normalize ratings by subtracting row mean

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	$2/3$			$5/3$	$-7/3$		
B	$1/3$	$1/3$	$-2/3$				
C				$-5/3$	$1/3$	$4/3$	
D		0					0

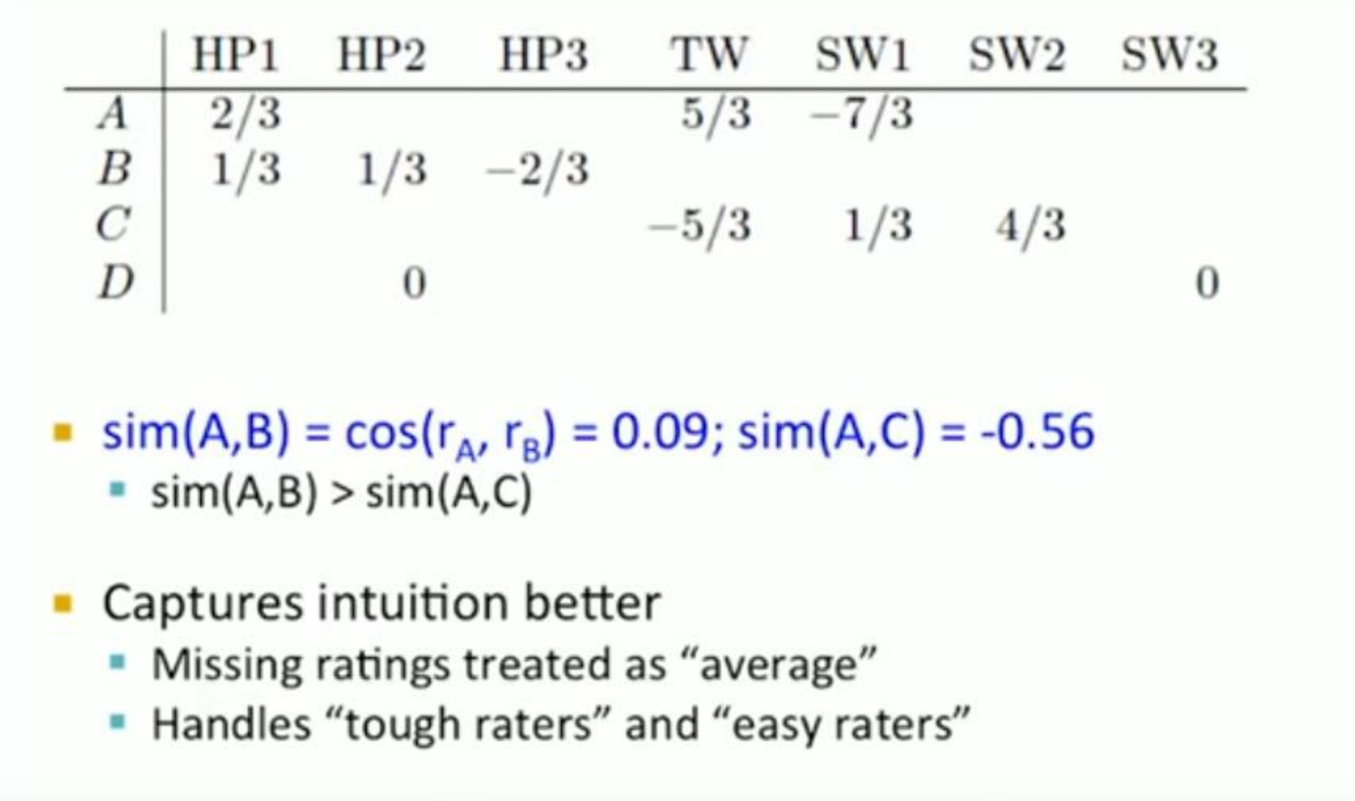
Centered Cosine



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	$2/3$			$5/3$	$-7/3$		
B	$1/3$	$1/3$	$-2/3$				
C				$-5/3$	$1/3$	$4/3$	
D		0					0

- sim(A,B) = cos(r_A, r_B) = 0.09; sim(A,C) = -0.56
  - sim(A,B) > sim(A,C)

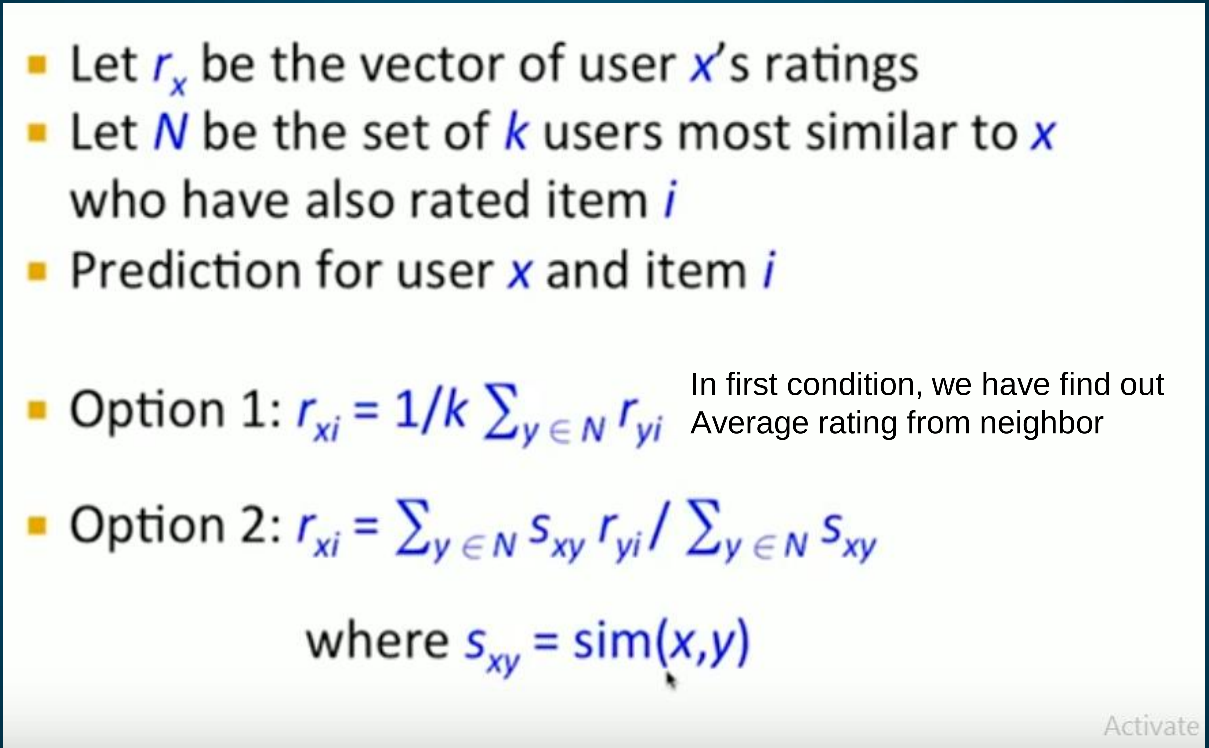
Centered Cosine



	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	$2/3$			$5/3$	$-7/3$		
B	$1/3$	$1/3$	$-2/3$				
C				$-5/3$	$1/3$	$4/3$	
D		0					0

- sim(A,B) = cos(r_A, r_B) = 0.09; sim(A,C) = -0.56
  - sim(A,B) > sim(A,C)
- Captures intuition better
  - Missing ratings treated as “average”
  - Handles “tough raters” and “easy raters”

Rating Prediction

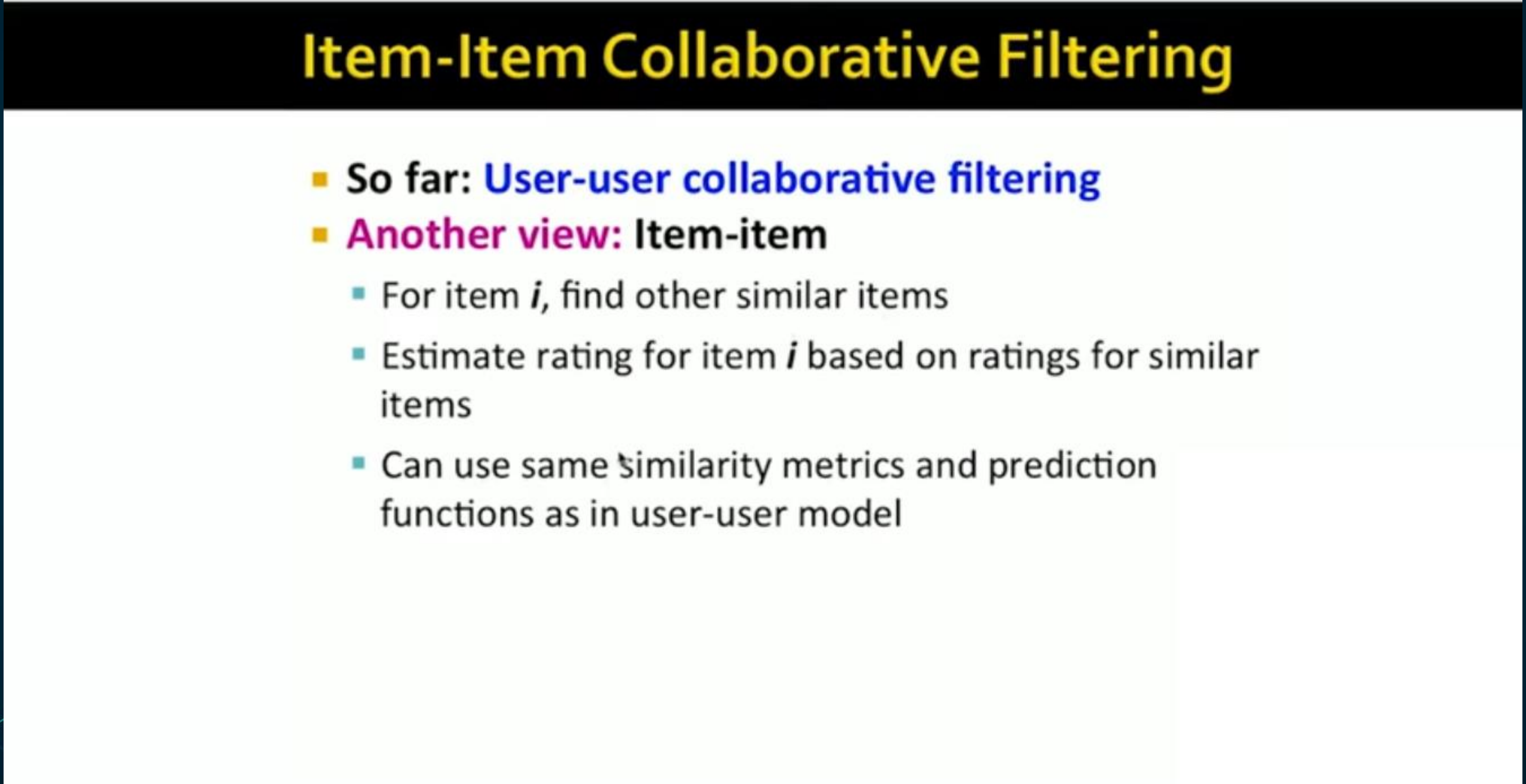


- Let r_x be the vector of user x's ratings
- Let N be the set of k users most similar to x who have also rated item i
- Prediction for user x and item i
- Option 1: $r_{xi} = 1/k \sum_{y \in N} r_{yi}$
- Option 2: $r_{xi} = \sum_{y \in N} s_{xy} r_{yi} / \sum_{y \in N} s_{xy}$

where $s_{xy} = \text{sim}(x,y)$

In first condition, we have find out
Average rating from neighbor

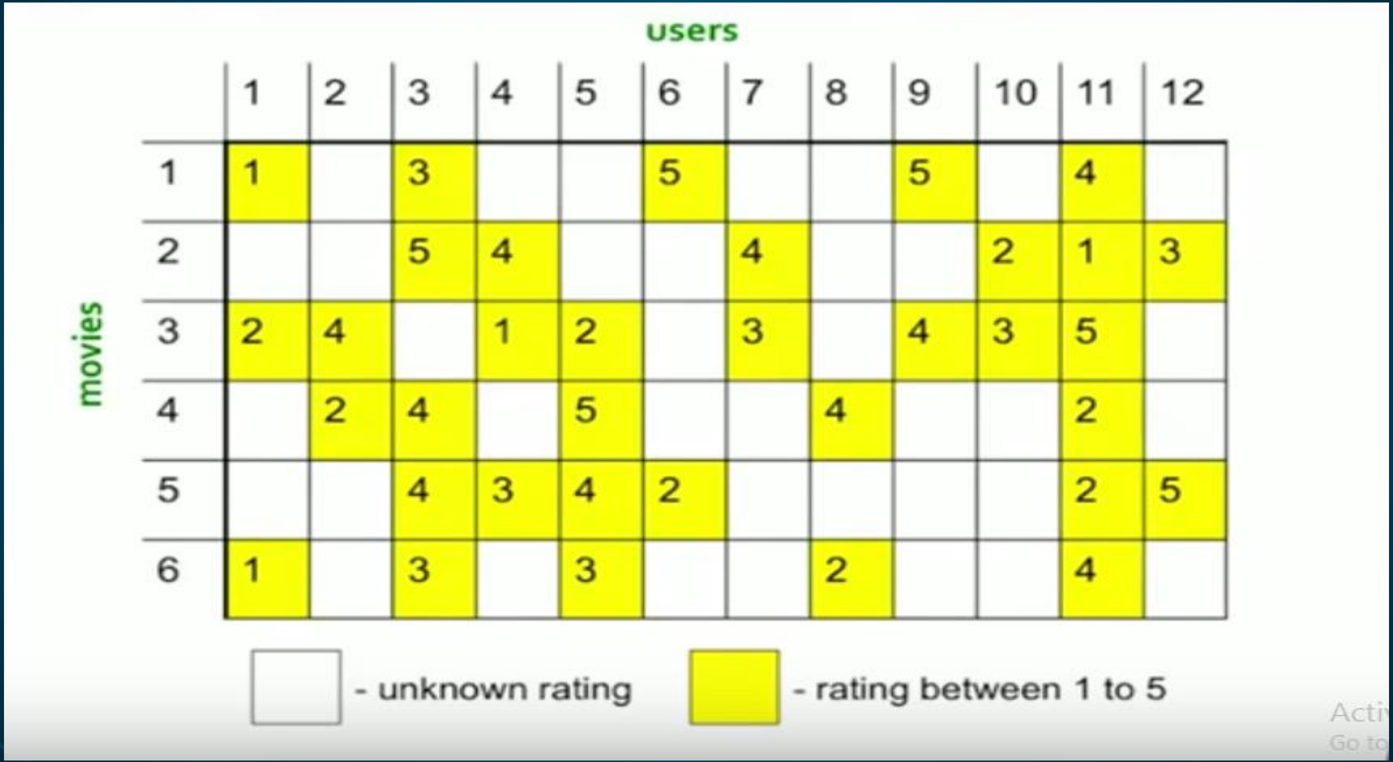
Rating Prediction



Item-Item Collaborative Filtering

- **So far: User-user collaborative filtering**
- **Another view: Item-item**
  - For item ***i***, find other similar items
  - Estimate rating for item ***i*** based on ratings for similar items
  - Can use same similarity metrics and prediction functions as in user-user model

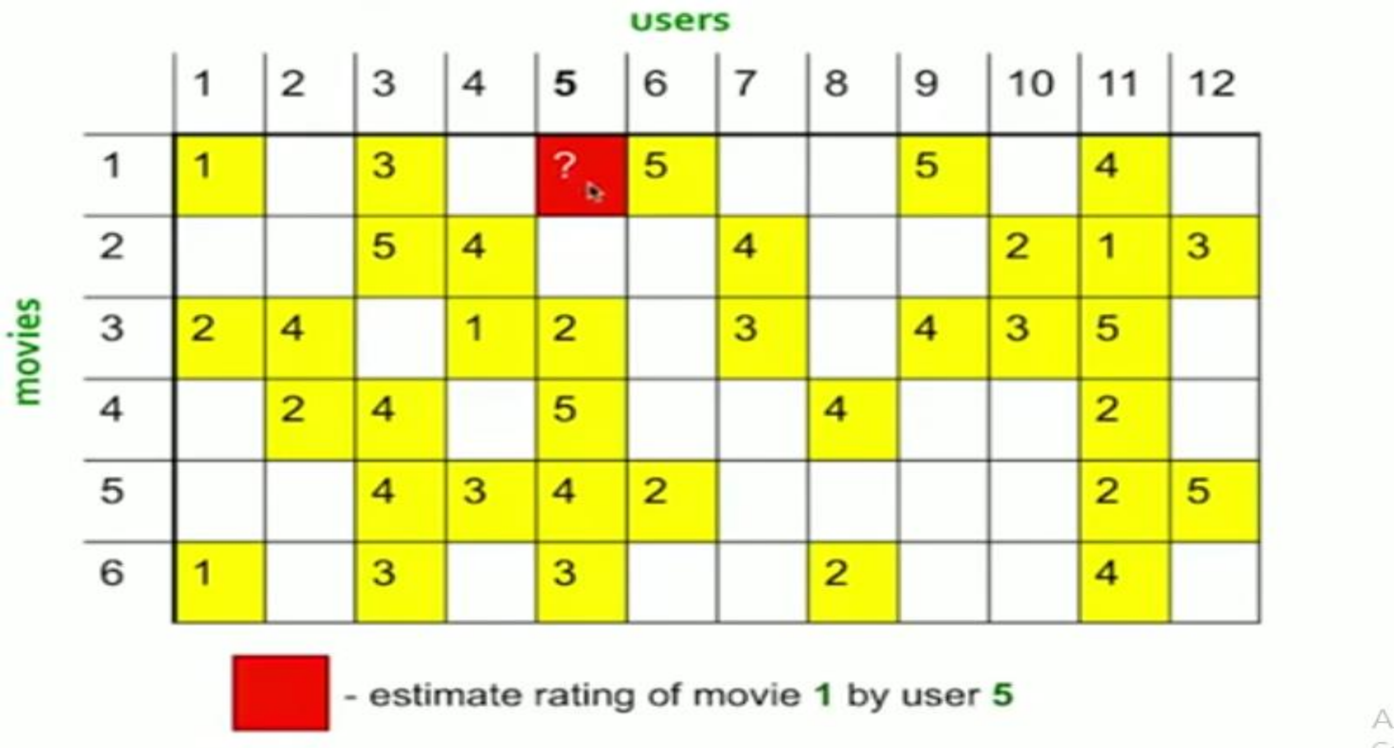
Item-Item CF



	users											
movies	1	2	3	4	5	6	7	8	9	10	11	12
1	1		3			5			5		4	
2			5	4			4			2	1	3
3	2	4		1	2		3		4	3	5	
4		2	4		5			4			2	
5			4	3	4	2					2	5
6	1		3		3			2			4	

□ - unknown rating 🟨 - rating between 1 to 5

Item-Item CF

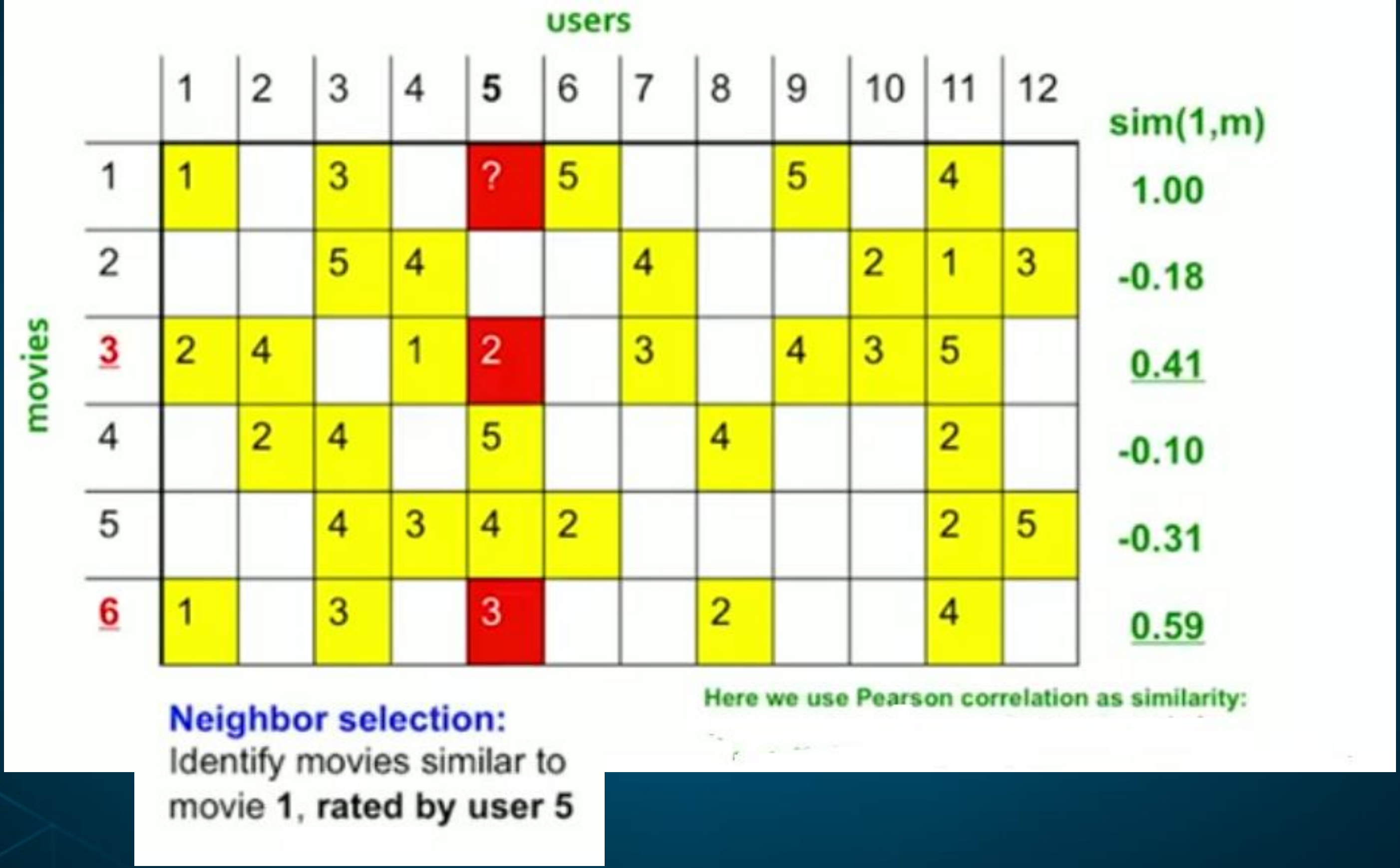


users

movies	1	2	3	4	5	6	7	8	9	10	11	12
1	1		3		?	5			5		4	
2			5	4			4			2	1	3
3	2	4		1	2		3		4	3	5	
4		2	4		5			4			2	
5			4	3	4	2					2	5
6	1		3		3			2			4	

🟥 - estimate rating of movie 1 by user 5

Item-Item CF



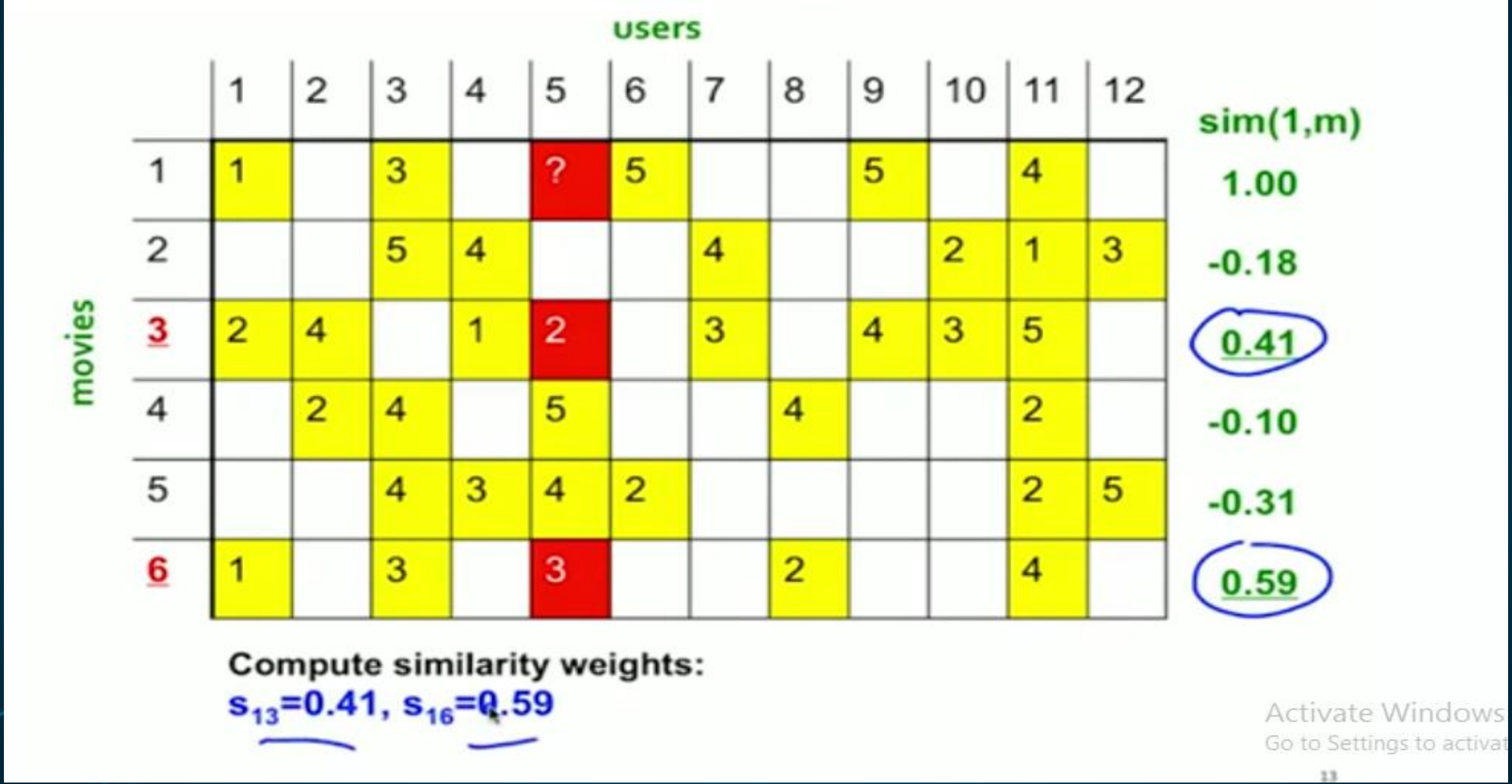
users

movies	1	2	3	4	**5**	6	7	8	9	10	11	12	sim(1,m)
1	1		3		?	5			5		4		1.00
2			5	4			4			2	1	3	-0.18
3	2	4		1	2		3		4	3	5		0.41
4		2	4		5			4			2		-0.10
5			4	3	4	2					2	5	-0.31
6	1		3		3			2			4		0.59

Here we use Pearson correlation as similarity:

Neighbor selection:
Identify movies similar to
movie **1**, **rated by user 5**

Item-Item CF

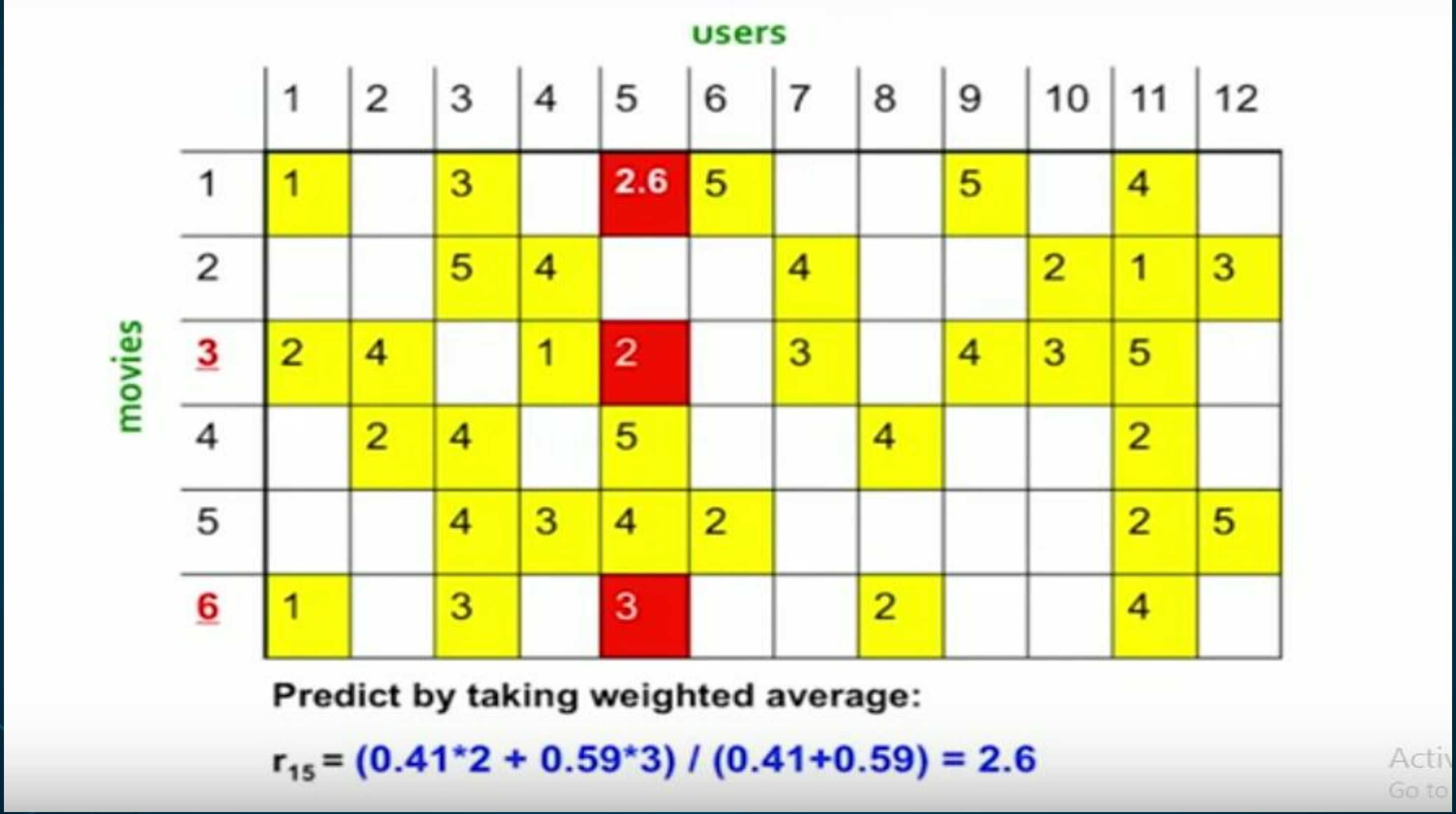


	users												
movies	1	2	3	4	5	6	7	8	9	10	11	12	sim(1,m)
1	1		3		?	5			5		4		1.00
2			5	4			4			2	1	3	-0.18
3	2	4		1	2		3		4	3	5		0.41
4		2	4		5			4			2		-0.10
5			4	3	4	2					2	5	-0.31
6	1		3		3			2			4		0.59

Compute similarity weights:

$s_{13}=0.41$, $s_{16}=0.59$

Item-Item CF



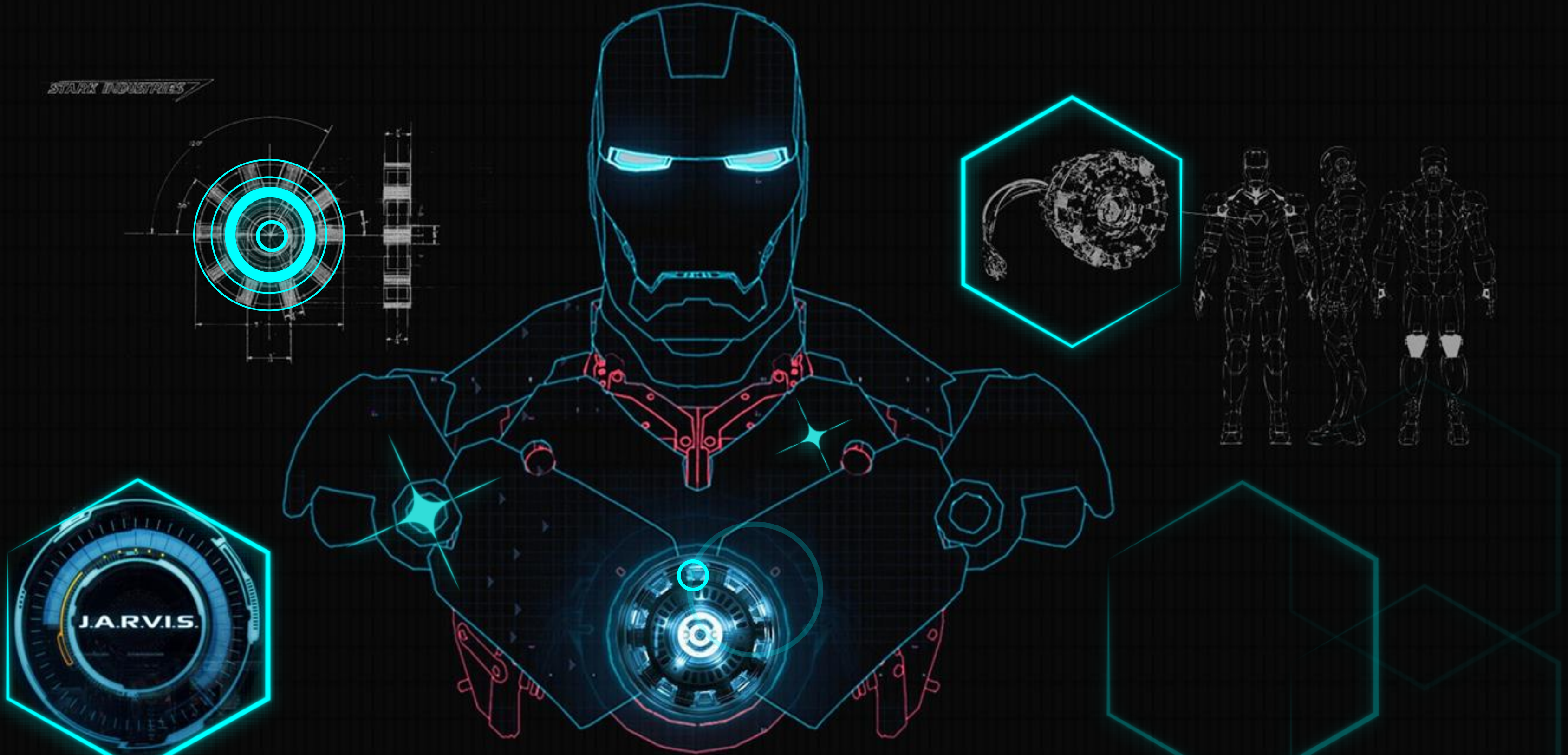
users

movies

	1	2	3	4	5	6	7	8	9	10	11	12
1	1		3		2.6	5			5		4	
2			5	4			4			2	1	3
3	2	4		1	2		3		4	3	5	
4		2	4		5			4			2	
5			4	3	4	2					2	5
6	1		3		3			2			4	

Predict by taking weighted average:

r_{15} = (0.41*2 + 0.59*3) / (0.41+0.59) = 2.6



Thank You