

# 데이터 과학

## L08: Similarity

---

Kookmin University

# 목차

## ❖ 비슷한 드라마 찾기

- ❖ Jaccard Similarity
- ❖ Cosine Similarity
- ❖ Centered Cosine Similarity (a.k.a. Pearson Correlation Coefficient)

## ❖ 별점 예측하기

- ❖ Collaborative Filtering
- ❖ Hybrid Methods

# 추천시스템

내가 재밌게 본 드라마와 비슷한 드라마는?

## NETFLIX

미생과 비슷한 콘텐츠



시그널과 비슷한 콘텐츠



# 유사도 Similarity

두 드라마의 비슷함의 정도 (유사도)를 어떻게 측정할 수 있을까?

1. 장르나 키워드가 비슷하면 비슷하다.
2. 사람들의 평가가 비슷하면 비슷하다..!?



# 장르·키워드 유사도

어떤 드라마가 서로 비슷할까?



한국 드라마  
TV프로그램-스릴러  
다크  
서스펜스



TV 드라마-범죄  
한국 드라마  
TV 프로그램-스릴러  
TV 드라마  
다크  
서스펜스



한국 드라마  
TV 드라마  
색다른 이야기

# 자카드 유사도 Jaccard Similarity

- 자카드 유사도 Jaccard Similarity: 두 집합이 얼마나 비슷한지를 측정

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

- $J(a, b) = 4/6 = 0.67$
- $J(a, c) = 1/6 = 0.17$
- $J(b, c) = 2/7 = 0.29$



한국 드라마  
TV 프로그램-스릴러  
다크  
서스펜스



TV 드라마-범죄  
한국 드라마  
TV 프로그램-스릴러  
TV 드라마  
다크  
서스펜스



한국 드라마  
TV 드라마  
색다른 이야기

Term-Frequency

# 키워드 뽑는 법: TF-IDF Inverse Document Frequency

줄거리나 설명글 등에서 자동으로 키워드를 뽑고 싶다.

⇒ 어떤 단어가 중요한 단어일까?

한 문서

$$TF_{ij} = \frac{f_{ij}}{\max_k f_{kj}}$$

문서 내 빈도수

해당 줄거리에 많이 등장하는 단어가 중요한 단어!

$$IDF_i = \log \frac{N}{n_i}$$

전체 문서 수 대비 빈도수

여러 문서

그런데 다른 드라마 줄거리에도 자주 나오는 단어는 중요한 단어가 아닐 것 같은데..?

$$\text{TF-IDF score: } w_{ij} = TF_{ij} \times IDF_i$$

각 줄거리마다 TF-IDF score가 가장 높은 단어 몇 개를 골라 키워드로 쓰자

단어에 대한 점수

$f_{ij}$  = 문서 j에서 단어 i가 등장한 빈도수  
 $n_i$  = 단어 i가 등장한 문서 수  
 $N$  = 전체 문서 수

# 평가 유사도

보이스, SKY캐슬 중에 터널과 더 비슷한 드라마는?



4.5

2.5

4.0

2.0

1.5



4.0

4.5

2.0



2.5

5.0

2.5

4.5

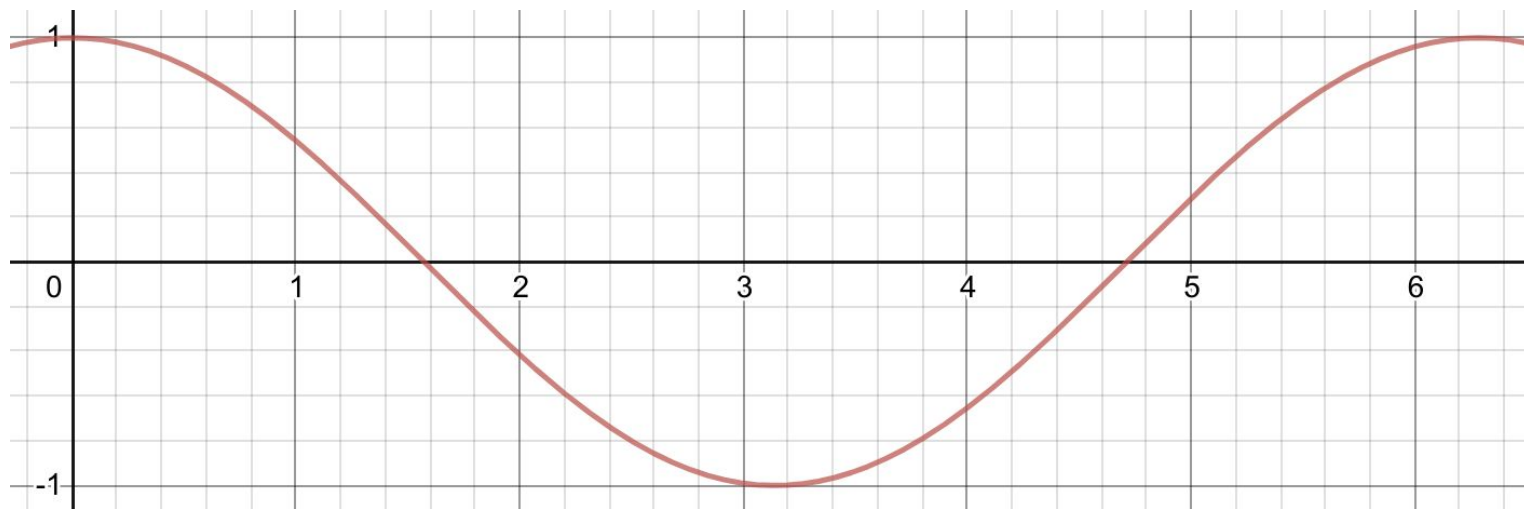
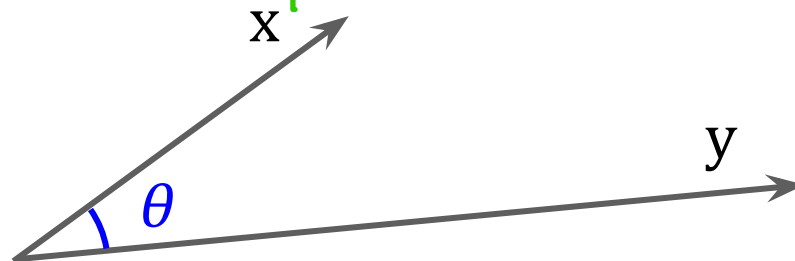


# 코사인 유사도 Cosine Similarity

~~벡터로 봤을 때, 같은 방향을 가리키면 유사도가 높다!~~

각도 0°일수록 ↓

$$\cos(\theta) = \frac{x \cdot y}{|x||y|}$$












# 코사인 유사도 Cosine Similarity

코사인 유사도로 드라마끼리의 유사도를 측정해보면?

$$U(x, y) = \cos(\theta) = \frac{x \cdot y}{|x||y|}$$

*Handwritten note:*  $|x| = \sqrt{a^2 + b^2 + c^2 + \dots}$

- $U(a, b) = 0.47$
- $U(a, c) = 0.67$
- $U(b, c) = 0.38$

	 A	 B	 C	 D	 E	 F
a 	4	2	<div style="border: 1px solid green; padding: 2px;">?</div>	4	2	1
b 	4		4			2
c 	2	5		2		4










# 코사인 유사도 Cosine Similarity

왜  $U(a, c)$ 의 유사도가  $U(a, b)$ 의 유사도보다 높게 계산되는가..?!

평가 안한 것을 0으로 생각하니까..!

$$U(x, y) = \cos(\theta) = \frac{x \cdot y}{|x||y|}$$

- $U(a, b) = 0.47$
- $U(a, c) = 0.67$
- $U(b, c) = 0.38$










	 A	 B	 C	 D	 E	 F
a 	4	2	0	4	2	1
b 	4	0	4	0	0	2
c 	2	5	0	2	0	4

# Centered Cosine Similarity

- 평가하지 않은 경우, **평균 점수**를 부여

- $U(a, b) = 0.96$
- $U(a, c) = 0.67$
- $U(b, c) = 0.38$

이제  $U(a, b)$ 가  $U(a, c)$ 보다 크긴 한데,  
왜  $U(a, c)$ 와  $U(b, c)$ 가 양수일까...?










	 A	 B	 C	 D	 E	 F	평균
 a	4	2	13/5	4	2	1	13/5
 b	4	10/3	4	10/3	10/3	2	10/3
 c	2	5	13/4	2	13/4	4	13/4

# Centered Cosine Similarity

- 평가하지 않은 경우, **평균 점수**를 부여
- 모든 평점을 평균 점수만큼 빼줌

- $U(a, b) = 0.70$
- $U(a, c) = -0.82$
- $U(b, c) = -0.43$

이제  $U(a, b)$ 가  $U(a, c)$ 보다 크긴 한데,  
왜  $U(a, c)$ 와  $U(b, c)$ 가 양수일까...?  
⇒ **모든 평점이 긍정적이라서...!**

	 A	 B	 C	 D	 E	 F	평균
 a	7/5	-5/3	0	7/5	-5/3	-8/5	13/5
 b	2/3	0	2/3	0	0	-4/3	10/3
 c	-5/4	7/4	0	-5/4	0	3/4	13/4

# 추천하기

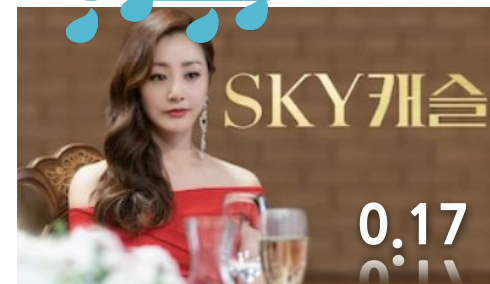
- 내가 재밌게 본 드라마 a와 유사도가 가장 높은 드라마 n개 추천하기
- 장르·키워드 유사도  $J(a, x)$ 와 평가 유사도  $U(a, x)$ 를 모두 활용

$$Score(a, x) = \alpha J(a, x) + (1 - \alpha)U(a, x)$$

- $Score(a, b) = 0.5 J(a, b) + 0.5 U(a, b) = 0.69$
- $Score(a, c) = 0.5 J(a, c) + 0.5 U(a, c) = 0.17$

( $\alpha$ 가 0.5일 때)

내가 재밌게 본 드라마



# 목차

## ❖ 비슷한 드라마 찾기

- ❖ Jaccard Similarity
- ❖ Cosine Similarity
- ❖ Centered Cosine Similarity (a.k.a. Pearson Correlation Coefficient)

## ❖ 별점 예측하기

- ❖ Collaborative Filtering
- ❖ Hybrid Methods

~~Latent~~ Factor  $\rightarrow$  Matrix

$t$

나눠주기

# 추천시스템

내가 이 드라마를 본다면, 별점을 몇점을 줄까?  
내 예상별점이 높은 드라마를 추천해줘!



한국 TV 인기 순위 1위

WATCHA

## 부부의 세계

2020 • JTBC • 스릴러/드라마/TV드라마

평점 ★4.5 (4,294명) • 예상 ★4.3



박하명 님의 취향저격 베스트 콘텐츠

NETFLIX





# 별점 예측 방법

- Collaborative Filtering
  - Item-Item Collaborative Filtering
  - User-User Collaborative Filtering
- Latent Factor Model

# Item-Item Collaborative Filtering

내가 이 드라마를 본다면, 별점을 몇점을 줄지 어떻게 예측할까?

Key Idea: 내가 이 드라마와 비슷한 드라마에 몇점을 주었나?



이미 본 비슷한 드라마



# Item-Item Collaborative Filtering

내(사용자 x)가 아이템 i에 매길 평점 예측하기

## 방법 1. 평균내기

$r_{xi}$ : 사용자 x가 아이템(드라마) i에 매길 평점

N: 내가 평가한 아이템 중에서 아이템 i와 가장 유사한 k개의 아이템 집합

$$\frac{4.5 \times 0.7 + 5.0 \times 0.4}{0.7 + 0.4}$$

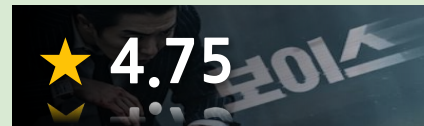
$$r_{xi} = \frac{1}{k} \sum_{j \in N} r_{xj}$$

비슷한 드라마

N: 이미 본 비슷한 드라마



$r_{xi}$ : 예상 별점



# Item-Item Collaborative Filtering

내(사용자 x)가 아이템 i에 매길 평점 예측하기

방법 2. 더 유사한 아이템에 가중치 줘서 평균내기

$r_{xi}$ : 사용자 x가 아이템(드라마) i에 매긴 평점

N: 내가 평가한 아이템 중에서 아이템 i와 가장 유사한 k개의 아이템 집합

$s_{ij}$ : 아이템 i와 아이템 j의 유사도

$$r_{xi} = \frac{\sum_{j \in N} s_{ij} \times r_{xj}}{\sum_{j \in N} s_{ij}}$$

N: 이미 본 비슷한 드라마

	유사도 0.7
	0.4

$r_{xi}$ : 예상 별점



# Item-Item Collaborative Filtering

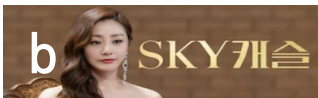
$|N| = 2$  일 때, 사용자 H가 드라마 a에 남길 평점 예측해보기

a와의  
유사도

1.00



-0.18



0.41



-0.10















-0.31



0.59



	 A	 B	 C	 D	 E	 F	 G	 H	 I	 J	 K	 L
a		4		5			5	?		3		1
b	3	1	2			4			4	5		
c		5	3	4		3		2	1		4	2
d		2			4			5		4	2	
e	5	2					2	4	3	4		
f		4			2			3		3		1

# Item-Item Collaborative Filtering

$|N| = 2$  일 때, 사용자 H가 드라마 a에 남길 평점 예측해보기

user

													
a와의 유사도													
1.00			4		5			5	2.6		3		1
-0.18		3	1	2			4			4	5		
0.41			5	3	4		3		2	1		4	2
-0.10			2			4			5		4	2	
-0.31		5	2					2	4	3	4		
0.59			4			2			3		3		1

# User-User Collaborative Filtering

내(사용자 x)가 아이템 i에 매길 평점 예측하기

- 나와 유사한 사람들이 아이템 i에 매긴 평점을 이용

유사한 사람들  
유사도

$r_{xi}$ : 사용자 x가 아이템(드라마) i에 매긴 평점

$N$ : 아이템 i를 평가한 사람 중에서 나와 가장 유사한 k 명의 사용자 집합

$s_{xy}$ : 사용자 x와 사용자 y의 유사도

$$r_{xi} = \frac{1}{k} \sum_{y \in N} r_{yi}$$

$$r_{xi} = \frac{\sum_{y \in N} s_{xy} r_{yi}}{\sum_{y \in N} s_{xy}}$$

# Item-Item vs User-User

- 이론상, user-user와 item-item은 동일한 정확도를 가짐
- 실제로는, item-item이 user-user보다 더 좋은 성능을 보임
  - Why?

→ 사람은 버킷수가 많음...



# Collaborative Filtering의 장·단점

- **장점 1:** 영화, 드라마, 도서 등등... 추천대상에 제한이 없다.
  - 평가 정보만 쓰니까!
- **단점 1: Cold Start**
  - 충분한 user와 평가 정보가 확보되어야 한다.
- **단점 2: Sparsity**
  - 평가 데이터 (user-item matrix)에 빈 곳이 많다.
  - 같은 드라마를 평가한 사용자가 몇 명 없다.
- **단점 3: First rater**
  - 한번도 평가되지 않은 드라마는 절대 추천되지 않는다.
  - 예) 신작, 매니악한 드라마 등
- **단점 4: Popularity Bias**
  - 독특한 취향을 가지는 user에게는 추천이 잘 되지 않는다
  - 주로 인기있는 드라마가 추천되기 십상이다

# Hybrid Methods

- **내용 기반 추천 방법** **Content-based method** 과 collaborative filtering을 섞는다.
  - 새로운 아이템 추천할 땐?  
⇒ 줄거리, 출연진, 키워드, 장르 등 Item Profile을 활용하여 추천한다.
  - 새로운 사용자에게 추천할 땐?  
⇒ 전반적으로 인기가 좋은 Item을 추천한다.
- **둘 이상의 추천시스템을 구현하고, 통합하여 추천하자!**
  - 예) 둘 이상의 추천결과를 **선형 결합**
    - **global baseline + collaborative filtering**

?

-



# Global Baseline Estimate → 아이템 기반의 평점 (base)

이미 높은 평점을 받은 드라마에는 나도 높은 평점을 주지 않을까?

진원이는 간간한 편인데, 평균보다 조금 낮게 평점을 주지 않을까?

- 진원이가 드라마 "이태원 클라쓰"를 보고 매길 평점 예측하기
  - 문제: 진원이는 "이태원 클라쓰"와 비슷한 드라마를 본 적이 없다...!
- 평점 가능해보기 (Global Baseline Estimate)
  - 평균 드라마 평점: 3.7점
  - "이태원 클라쓰"의 평점 평균: 4.2점 (평균보다 0.5점 높음) → item bias
  - 진원이의 평점 평균: 3.5점 (평균보다 0.2점 낮음)
  - 기본 점수 (Global baseline) 예측:  $3.7 + 0.5 - 0.2 = 4.0$ 점

# Global Baseline Estimate + CF

기본 평점 예측을 CF(Collaborative Filtering)에 적용하기

- 기본 평점 예측 (Global Baseline Estimate)
  - 진원이는 대략적으로 "이태원 클라쓰"에 4.0점을 매길 것이다.
- Collaborative Filtering
  - 진원이는 "이태원 클라쓰"와 유사한 "사랑의 불시착"을 봤는데...
  - 본인의 평점 평균보다 1.0점을 낮게 줬다.
- ~~최종 예측~~
  - 진원이는 "이태원 클라쓰"에  $4.0 - 1.0 = 3.0$ 점을 매길 것이다.

# Global Baseline Estimate + CF

기본 평점 예측을 CF(Collaborative Filtering)에 적용하기

$$r_{xi} = b_{xi} + \frac{\sum_{j \in N} s_{ij} \times (r_{xj} - b_{xj})}{\sum_{j \in N} s_{ij}}$$

baseline estimate for  $r_{xi}$

$$b_{xi} = \mu + b_x + b_i$$

기준 (bias를 고려하지 않을 때):

$$r_{xi} = \frac{\sum_{j \in N} s_{ij} \times r_{xj}}{\sum_{j \in N} s_{ij}}$$

이것도 나눠서

이것도 나눠서

이게 j의

||  
이것도 나눠서  
 $\mu$  = 전체 별점 평균  
 $b_x$  = 사용자 x의 bias  
 $b_i$  = 아이템 i의 bias

$r_{xj}$  = 이 유저가 j에 준 별점  
 $s_{ij}$  = i와 j가 함께 본 영화 수

# Questions?