

제2장 이웃 기반 협업 필터링

2.1 개요

이웃 기반 협업 필터링 알고리즘(Collaborative Filtering Using kNN)은 협업 필터링의 초기 알고리즘!

아래 사실을 전제로 합니다 :

- 유사한 User는 Rating 방식에서 유사한 Pattern
- 유사한 Item에는 유사한 Rating을 부여

사용자 기반 협업 필터링

타겟 유저 A의 추천 제공을 위해 유사한 유저들의 평점을 이용하고,
예측된 A의 평점은 “Peer Group”의 각 아이템의 평점에 대한 가중 평균으로 산정합니다.
이웃 사용자 평점을 활용해 예측하기 때문에
Neighbors는 사용자들 간의 유사도로 결정합니다.

아이템 기반 협업 필터링

타겟 아이템 B에 대한 추천 만들기 위해 B와 유사한 아이템 집합 S 결정하고,
특정 사용자 A의 아이템 B에 대한 평점 예측을 위해 A에 대한 집합 S의 평점 결정 필요합니다.
사용자가 평가한 평점 중 가장 비슷한 아이템에 대한 평점을 활용한 예측이기 때문에
Neighbors는 아이템 간의 유사도로 결정합니다.

User-Item Rating Matrix(사용자-아이템 평점 행렬)

$m \times n$ 행렬 $R = [r_{uj}]$

1. 유저-아이템 조합 평점 값 예측 :

r_{uj} 에서의 사용자 u 에 대한 아이템 j 의 누락된 평점을 예측합니다.

2. 상위- k 아이템 or 상위- k 사용자 결정 :

현식적으로 위 방법보다 특정 사용자의 가장 관련 있는 상위- k 개 아이템이나 상위- k 개 사용자를 탐색합니다. 보통 상위- k 의 아이템 결정하는 문제가 상위- k 사용자 찾는 문제보다 흔합니다.

상위- k 아이템을 결정하려면 그 사용자의 각각 아이템에 대한 평점 예측이 필요하기 때문에 두 방법론은 매우 밀접하게 연관되어 있습니다.

2.2 평점 행렬의 주요 특징

$m \times n$ 행렬 $R = [r_{uj}]$

m : 사용자 수

n : 아이템 수

R : 평점 행렬

r_{uj} : 사용자 u 의 아이템 j 에 대한 평가

평점은 여러 방법으로 나타낼 수 있습니다.

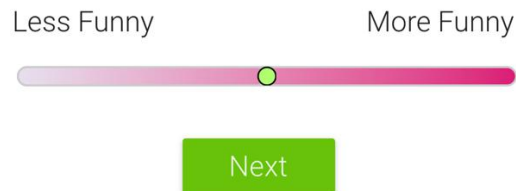
1. **연속 평점** - 평점은 연속형이며 아이템에 대한 좋음과 싫음을 나타냅니다.

▼ Jester Joke 추천 엔진, 사용자에게 무한한 가능성에 대한 부담을 줄 수 있다는 단점이 있습니다.

First rate two jokes.

Q: If a person who speaks three languages is called "trilingual," and a person who speaks two languages is called "bilingual," what do you call a person who only speaks one language?

A: American!



2. **인터벌 기반 평점** - 숫자 값은 명시적으로 평점 사이 거리를 정의하고 거리는 같다고 가정합니다.

▼ 주로 5점이나 7점 스케일을 주로 사용하며 가장 흔한 평점 시스템 중 하나입니다.



Figure 1.1: Example of 5-point interval ratings

3. **서수 평점** - 순서형의 범주형 값이 쓰이지만 거리가 동일하지 않다고 가정합니다. (단순 이론)

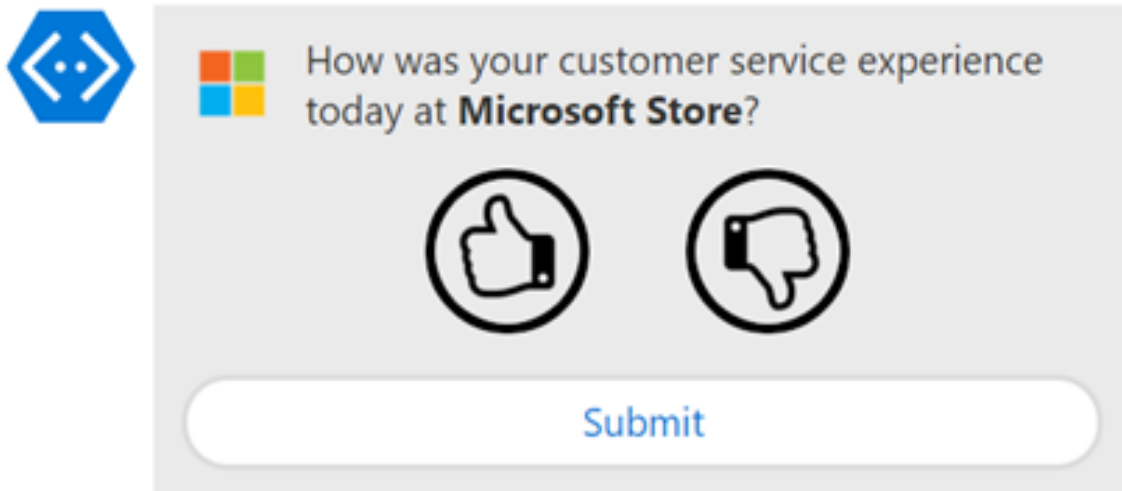
▼ 인터벌 기반 평점과 유사, 선택지가 짝수인 경우 중립이 없으며 '강제 선택'이 요구됩니다.

Ordinal ranking scale

| | Strongly agree 5 | Agree 4 | Neutral 3 | Disagree 2 | Strongly disagree 1 |
|-----------------------|---------------------|------------|--------------|---------------|------------------------|
| Site managers | 4 | 3 | 5 | 5 | 0 |
| Head office personnel | 0 | 1 | 1 | 9 | 6 |
| Operatives | 7 | 5 | 4 | 1 | 0 |

4. 이진 평점 - 긍정 혹은 부정 두 개의 옵션만이 존재합니다.

▼ 마찬가지로 강제 선택이 사용자에게 부과됩니다. 중립을 선택하는 경우는 평가를 하지 않습니다.



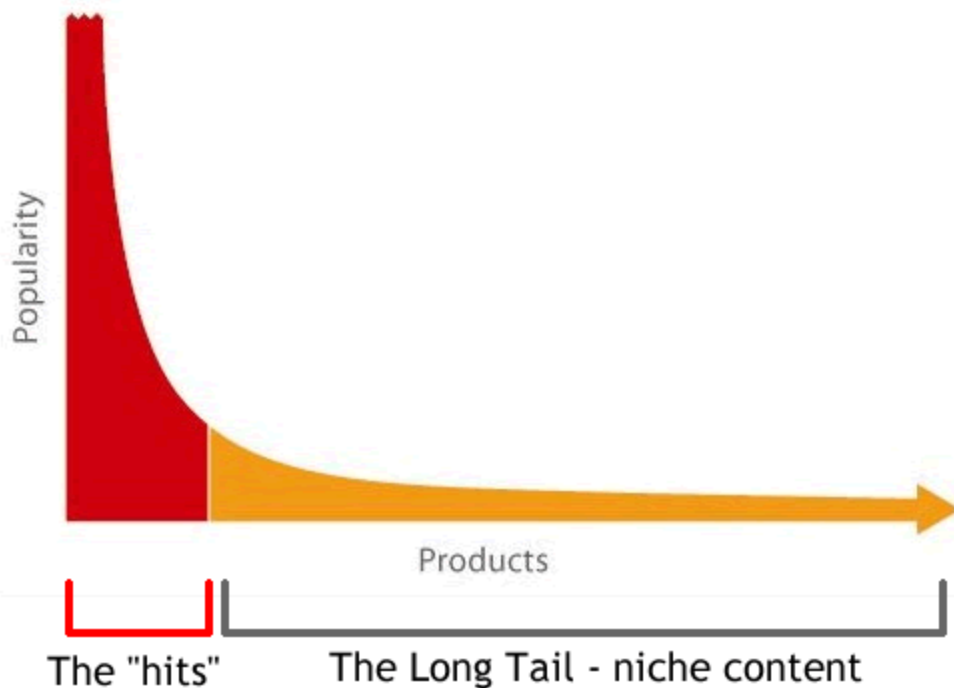
The image shows a feedback form from the Microsoft Store. On the left is the Microsoft logo. To its right is the text "How was your customer service experience today at **Microsoft Store**?". Below this text are two circular icons: a thumbs up and a thumbs down. At the bottom of the form is a white button with the word "Submit" in blue text.

5. 단항 평점 - 긍정 선호도만 지정합니다.

▼ 실제 환경에서 많이 발생합니다. 구매 행동도 긍정적인 한 표로 계산합니다.



아이템에 대한 평점 분포는 롱테일 형태로 실제 환경의 속성을 반영합니다.
즉, 아이템 중 극히 일부만이 자주 평가됩니다. (인기 아이템)



“hits”에 해당하는 인기 아이템은 주로 판매자에게 이익이 거의 없는 치열한 분야의 아이템이며 반대로 낮은 빈도를 보이는 아이템은 이윤이 더 크게 남아서 판매자 입장에서 적은 빈도의 아이템을 추천하는 것이 수익성 면에서 이득입니다. 실제로, 아마존과 같은 회사는 롱테일에 위치한 아이템을 팔며 이윤을 많이 남겼다고 합니다.

인기 있는 아이템을 제안하는 경우가 많지만, 이는 사용자 입장에서 추천이 다양하지 않다고 느끼는 부정적인 영향을 줄 수도 있고 반복해서 유명 추천 아이템을 제안하면 지루함을 느낄 수 있습니다.

롱테일 분포는 사용자가 자주 평가한 아이템 수가 적음을 의미하기 때문에 많은 평점을 받은 상품이 적은 평점의 상품을 대변할 수 없습니다.

2.3 이웃 기반 방법론의 평점 예측

‘이웃’이라는 개념은 예측을 위해 유사한 사용자 혹은 유사한 상품을 찾아야 한다는 뜻을 갖고 있습니다. 평점 행렬을 통해 추천을 진행하기 위해 사용자-사용자 유사도나 아이템-아이템 유사도를 이용합니다. 다음은 이웃 기반 방법론을 활용한 평점 예측의 예시입니다.

이웃 기반 모델

- Alice와 Bob은 과거 비슷한 패턴으로 영화 평점을 부여
- Bob은 아직 ‘범죄도시2’를 보지 않았음
- Alice는 ‘범죄도시2’를 매우 재밌게 보고 5점을 부여
- 이때, Bob의 평점을 예측하기 위해 Alice가 부여한 평점을 이용

아이템 기반 모델

- Bob은 과거 ‘범죄도시’와 ‘나쁜 녀석들’을 재밌게 보고 5점을 부여
- 이때, Bob의 ‘범죄도시2’에 대한 평점을 예측하기 위해 Bob의 과거 평점을 이용

2.3.1 사용자 기반 이웃 모델

이 접근법에서는 **사용자 기반 이웃**은 평점 예측이 계산되는 타깃 유저와 유사한 사용자를 찾기 위해 정의됩니다.

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/5de38ce4-ee2b-46e3-b53d-16c391f810df/사용자기반이웃모델.ipynb>

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|-----|-----|-----|-----|-----|-----|
| 1 | 7.0 | 6.0 | 7.0 | 4.0 | 5.0 | 4.0 |
| 2 | 6.0 | 7.0 | NaN | 4.0 | 3.0 | 4.0 |
| 3 | NaN | 3.0 | 3.0 | 1.0 | 1.0 | NaN |
| 4 | 1.0 | 2.0 | 2.0 | 3.0 | 3.0 | 4.0 |
| 5 | 1.0 | NaN | 1.0 | 2.0 | 3.0 | 3.0 |

교재에서 예시로 사용한 데이터프레임

1. 타깃 사용자와 다른 사용자들 간의 유사도를 계산합니다. (코사인 유사도, 피어슨 계수)

```
def compute_cos_similarity(v1, v2):
    norm1 = np.linalg.norm(v1)
    norm2 = np.linalg.norm(v2)
    dot = np.dot(v1, v2)
    return dot / (norm1 * norm2)

def compute_pearson_similarity(v1, v2):
    return np.dot((v1 - np.mean(v1)), (v2 - np.mean(v2))) / (
        (np.linalg.norm(v1 - np.mean(v1))) * (np.linalg.norm(v2 - np.mean(v2))))
```

2. 구해진 유사도를 기반으로 피어 그룹을 분류합니다.

사용자 3은 사용자 1과 사용자 2와 유사도가 크기 때문에 같은 피어 그룹으로 분류합니다.

1 & 3: 0.894

2 & 3: 0.939

3. 피어 그룹의 평점을 이용해 타깃 사용자의 평점을 예측합니다.

사용자 3의 아이템 1에 대한 평점을 예측하기 위해,
사용자 1과 사용자 2의 아이템 1에 대한 평점을 참고합니다.
계산 시,

- 사용자의 **평균 평점**
- 타깃 사용자와의 **상관계수**

를 참고합니다. ‘평균 평점’은 사용자마다 평점에 각기 다른 스케일을 부여하기 때문에 고려해야 하며, ‘상관계수’는 타깃 사용자와 얼마나 유사한지에 따라 가중치를 달리해야 하기 때문입니다.
아래는 사용자 3의 아이템 1에 대한 평점을 예측하는 식입니다.

$$\hat{r}_{31} = 2 + \frac{1.5 * 0.894 + 1.2 * 0.939}{0.894 + 0.939} \approx 3.35$$

유사도 함수, 피어 그룹 필터링과 예측함수는 다양한 변형이 존재합니다.

실제 컨텍스트에서 평점 분포는 롱테일 분포를 따르기 때문에 인기 아이템의 평점은 사용자 간의 차별성이 적기 때문에 추천 품질을 악화시킬 수 있습니다.

2.3.2 아이템 기반 이웃 모델

아이템 기반 모델에서의 ‘Peer Group’은 사용자가 아닌 **아이템**으로 구성되어 있습니다.

아이템 기반 방법론의 경우, 피어슨 상관 관계를 열에서 활용할 수 있지만 **조정된 코사인 (Adjusted Cosine)**이 일반적으로 더 우수한 결과를 보입니다.

사용자 u 의 타깃 아이템 t 의 평점을 예측하는 경우,

1. 조정된 코사인 유사도를 기반으로 아이템 t 와 가장 유사한 상위- k 아이템들을 결정합니다.
2. 아이템 t 에 상위- k 로 매칭되는 아이템들은 $Q_t(u)$ 로 설정합니다.
3. 이 평점들의 가중 평균 값을 예측 값이라고 합니다.

$$\hat{r}_{ut} = \frac{\sum_{j \in Q_t(u)} \text{AdjustedCosine}(j,t) * r_{uj}}{\sum_{j \in Q_t(u)} |\text{AdjustedCosine}(j,t)|}$$

4. 유사한 항목에 대한 사용자 u 자신의 평점을 활용합니다.

교재에서 같은 데이터에 아이템 기반 이웃 모델을 적용했습니다.

조정된 코사인 유사도에 따르면 아이템 1은 아이템 2와 아이템 3과 같은 피어 그룹이 되며, 아이템 6은 아이템 4와 아이템 5와 같은 피어 그룹으로 분류됩니다.

사용자 3의 아이템 2와 아이템 3에 대한 가중 평균은 아이템 1의 평점을 예측하는 데에 사용되고,

사용자 3의 아이템 4와 아이템 5에 대한 가중 평균은 아이템 6의 평점을 예측하는 데에 사용됩니다.

$$\hat{r}_{31} = \frac{3 * 0.735 + 3 * 0.912}{0.735 + 0.912} = 3$$

2.3.3 효율적 구현 및 계산 복잡성

모든 사용자-아이템 쌍에 대해 가능한 평점을 **모두** 예측하고 이에 대한 순위를 매기는 것이 현재 추천 시스템에 쓰이고 있는 가장 기본적인 접근 방식입니다.

예측 과정이 중간 계산 과정에서 재사용되기 때문에 중간 계산을 저장하고 순위 과정에 사용하기 위해 ‘오프라인 단계’를 사용하는 것이 권장됩니다.

오프라인 단계에서는 유사도 값과 피어 그룹을 계산합니다.

$n' \ll n$ 이 사용자의 지정된 평점의 최대 수

$m' \ll m$ 이 아이템의 지정된 평점의 최대 수

n 은 한 쌍의 사용자 간의 유사도를 계산하기 위한 최대 실행 시간이고,

m 은 한 쌍의 아이템 간의 유사도를 계산하기 위한 최대 실행 시간입니다.

사용자 기반 방법론의 경우, 타깃 사용자의 피어그룹 결정 과정에서 $O(m * n')$ 시간이 요구될 수 있습니다.

모든 사용자의 피어그룹을 계산하기 위한 오프라인 실행 시간은 $O(m^2 * n')$ 로 정의됩니다.

아이템 기반 방법론의 경우, 오프라인 실행 시간은 $O(n^2 * m')$ 로 정의됩니다.

k 값을 다양하게 접근하기 위해서는 사용자 쌍 사이에 0이 아닌 유사도 쌍을 모두 저장해야 합니다.

사용자 기반 방법론의 공간 요구는 $O(m^2)$ 인 반면, 아이템 기반 방법론은 $O(n^2)$ 를 요구합니다.

일반적으로 사용자 수가 아이템 수보다 많기 때문에 사용자 기반 방법론의 공간 요구 사항은 **아이템 기반 방법론**보다 대체로 큼니다.

2.3.4 사용자 기반 방법론과 아이템 기반 방법론의 비교

아이템 기반 방법론은 사용자 자체 평점을 추천에 활용하기 때문에 더 관련성이 높은 추천을 제공합니다. 유사 아이템을 타깃 항목으로 인지하고 그 아이템에 대한 사용자가 지정한 평점을 활용해 타깃의 평점을 추측합니다. **매우 직관적**이며, 추천에 구체적인 **근거를 제시**하고, 나은 정확도를 보입니다.

반면, **사용자 기반 방법론**은 더 다양한 추천을 제공합니다. 더 큰 다양성은 뜻밖의 발견을 장려하는 고유의 장점이 있습니다. 하지만 다른 사용자의 선호도를 근거로 하기에 추천에 구체적인 근거를 제시하는 것(privacy problem)은 불가능합니다.

또한, 아래와 같은 이유로 아이템 기반 방법론은 사용자 기반 방법론보다 변화에 안정적입니다.

- 사용자 수는 대체로 아이템 수보다 크다는 점
- 신규 사용자는 상용 시스템에 신규 아이템보다 더 자주 추가된다는 점

둘 이상의 사용자가 공통적으로 평점을 부여한 아이템의 숫자는 매우 적을 수 있지만, 둘 이상의 아이템에 대해 공통으로 평점을 부여한 사용자의 수는 매우 클 수 있습니다. 그리고 아이템 이웃은 신규 사용자 추가에 대해 변화될 이유가 없기 때문에 이웃 아이템 계산의 빈도는 조절이 가능합니다.

2.3.5 이웃 기반 방법론의 장점과 단점

이웃 기반 방법론은 단순하고 직관적이기 때문에 적용하기도 쉽고 문제를 파악하는 데에도 용이합니다. 또한, 아이템 기반 방법론의 경우 ‘해석 가능성’이 주목할 만합니다.

하지만, 가장 큰 단점은 대규모 환경에서 오프라인 단계가 시간적으로나 공간적으로나 큰 비용을 요구하기 때문에 효율성이 떨어진다는 것입니다. 또한, 사용자의 이웃 중 A에 대한 평점을 매기지 않았다면 사용자에게 A에 대한 평점 예측은 불가능합니다.

2.3.6 사용자 기반과 아이템 기반 방법론의 통합된 관점

사용자 기반과 아이템 기반 방법론을 통합해 평점 행렬 R 의 엔트리 r_{uj} 를 예측할 수 있습니다.

1. 평점 행렬에서 행과 열간의 코사인 계수를 이용해 평점 행렬의 타깃 엔트리 (u, j) 와 가장 유사한 행, 열을 결정합니다. 사용자 기반 방법론에서는 열을 이용하고, 아이템 기반 방법론에서는 행을 이용합니다.

2. 위 단계에서 찾은 행과 열의 평점의 가중 조합을 이용해 타깃 엔트리 (u, j) 를 예측합니다.

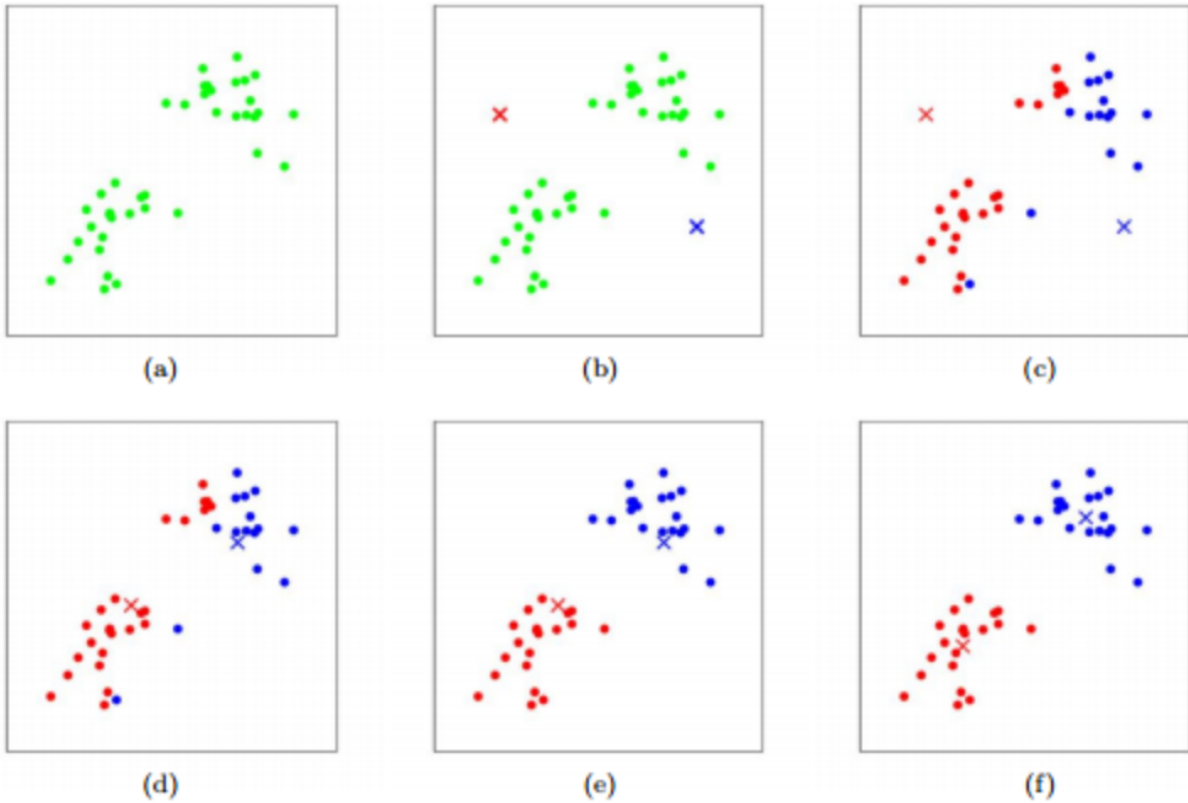
2.4 클러스터링과 이웃 기반 방법론

앞서 말했듯, 이웃 기반 방법론의 주요 문제점은 ‘오프라인 단계에서의 복잡성’입니다.

이를 해결하기 위해 ‘클러스터링’ 기반 방법론을 도입하는데, 가장 근접한 이웃 단계를 클러스터링 단계로 대체합니다. 클러스터링은 각각 가능한 타깃이 중심이 될 필요 없이 더 작은 피어 그룹을 생성합니다. 클러스터 내에서의 상위- k 의 가장 근접한 동료가 예측에 활용되며 이는 효율성 측면에서 큰 이점이 있습니다.

클러스터링 세분화 시, 효율성과 정확도는 트레이드오프 관계에 있습니다. 클러스터가 세분화될수록 효율성은 개선되지만 정확도는 떨어집니다. 하지만 많은 경우 효율성의 큰 증가는 정확도의 작은 감소로 이루어지기 때문에 효율성을 우선시하는 경우가 많습니다.

효율성을 중요시하는 까닭에 평점 행렬이 불완전해지지만, 따라서 클러스터링 방법론은 아주 큰 불완전 데이터셋에 적용됩니다. 이 관점에서 k 평균 방법론(k-means clustering)은 쉽게 적용될 수 있습니다.



k 평균 접근법의 기본 구조는 k 개의 서로 다른 클러스터를 표현하는 k 개의 중심 포인트(평균)을 이용하는 것입니다. k 평균 방법론에서 클러스터링에 대한 해결책은 k 개의 대표로 표현될 수 있습니다. $\bar{Y}_1 \dots \bar{Y}_k$ 의 k 개의 대표 집합이 있을 때, 각 데이터 포인트는 유사도나 거리 함수를 이용해 최근접 대표에 할당됩니다. 따라서, 클러스터는 대표에 의존적이며 그 반대로 가능합니다. 이 상호 의존성은 반복 작업을 통해 달성됩니다.

이웃 기반 방법론의 경우, 다음 두 단계의 방법 작업을 통해 수렴합니다.

1. $m \times n$ 행렬의 각 열에 대해 근접한 $\bar{Y}_1 \dots \bar{Y}_k$ 를 할당함으로써 클러스터 $C_1 \dots C_k$ 를 결정합니다. 대체로 유사도 계산에 있어서 유클리드 거리 혹은 맨해튼 거리를 이용합니다.
2. 각 $i \in \{1 \dots k\}$ 에 대해 C_i 의 현재 포인트의 집합의 중심에 대해 \bar{Y}_i 를 재설정합니다.

2.5 차원 축소와 이웃 기반 방법론

차원 축소를 통해 이웃 기반 방법론은 품질과 효율성을 동시에 개선시킬 수 있습니다. $m \times n$ 행렬을 $m \times d$ 행렬로 축소시키면 모든 차원이 완전히 지정되어 있기 때문에 계산이 정확하며, 유사도 산출을 더욱 효율적입니다.

차원 축소는 특이값 분해(SVD: Singular Value decomposition)로 진행되며 다음 식을 따릅니다.

$$A = U D V^T$$

Left singular vectors

Singular values

Right singular vectors

U 는 $m \times m$ 행렬, V 는 $n \times n$ 행렬로, D 를 구할 수 있습니다. 이를 통해 이웃 기반 모델의 희소성 문제를 완화할 수 있고 피어 그룹 결정에 효율성을 불러옵니다.

2.6 이웃 방법론의 회귀 모델링 관점

사용자 기반 및 아이템 기반 방법론 모두 ‘이웃 사용자의 동일한 항목에 대한 평점’ 또는 ‘이웃 항목의 동일한 사용자의 평점’의 선형함수로 평점을 예측합니다.

즉, 사용자 기반 이웃 방법론의 경우 예측 평점은 동일 아이템의 다른 평점의 가중 선형 조합입니다.

조합 가중치로 유사도 값을 사용하는 것은 오히려 휴리스틱적이고 임의적입니다. 모델링 관점에서 최적성으로 인해 평점을 결합한 가중치가 더 합리적입니다.

2.7 이웃 기반 방법에 대한 그래프 모델

평점의 희소성은 이웃 기반 방법의 유사도 계산에 문제를 일으킵니다. 구조적 전이성이나 순위 기술을 사용해 이웃 기반 방법론에서 유사도를 정의하기 위해 여러 그래프 모델이 사용됩니다. 그래프는 네트워크 도메인의 많은 알고리즘 도구를 가능하게 하는 추상화 작업입니다.

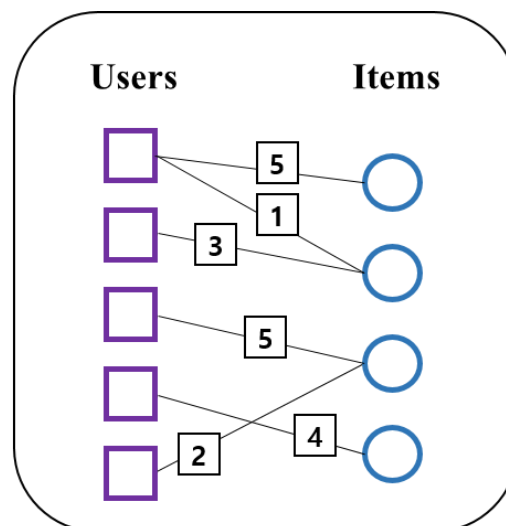
그래프는 다양한 사용자와 아이템 간의 관계에 관한 **구조적 표현**을 제공합니다.

 <https://arxiv.org/pdf/1706.02263.pdf>

2.7.1 사용자-아이템 그래프

| | Items | | | |
|-------|-------|---|---|---|
| Users | 5 | 1 | 0 | 0 |
| | 0 | 3 | 0 | 0 |
| | 0 | 0 | 5 | 0 |
| | 0 | 0 | 0 | 4 |
| | 0 | 0 | 2 | 0 |
| | | | | |

Rating matrix M



Bipartite graph

지역을 정의하기 위해 피어슨 상관계수가 아닌 사용자-아이템 그래프에서 구조 측정 값을 사용할 수 있습니다. 이는 추천 과정에서 가장자리의 구조적 전이성을 사용할 수 있기 때문에 희소 평점 행렬에 더 효과적입니다.

그래프를 통해 노드 간의 간접 연결이라는 개념으로 이웃을 구성할 수 있습니다.

2.7.2 사용자-사용자 그래프

사용자-사용자 연결은 사용자-아이템 그래프의 짝수 번째 hop으로 정의됩니다.

사용자-아이템 그래프보다 그래프의 가장자리가 더 유익하다는 장점이 있으며 전이성을 사용해 평점을 예측하기 때문에 매우 희소한 행렬에서 작동할 수 있습니다.

2.7.3 아이템-아이템 그래프

아이템-아이템 그래프는 상관관계 그래프로 불리기도 합니다. 가장자리의 가중치가 랜덤 워크 확률에 해당하는 그래프가 생성됩니다. 또한 상관관계 그래프의 구성에 평점 값이 사용되지 않는다는 점도 주목할 만합니다.

2.8 요약

- 협업 필터링은 분류 및 회귀 문제의 일반화로 볼 수 있습니다.
- 이웃 기반 방법론은 최근접 이웃 분류 및 회귀 방법에서 영감을 얻었습니다.
- 사용자 기반 방법에서는 유사도 함수를 사용해 이웃을 계산하고 이웃의 평점을 사용자의 평점을 추정하기 위해 사용합니다.
- 아이템 기반 방법에서는 유사도 함수가 아이템 집단에 적용되며 유사 아이템에 대한 사용자 자신의 평점을 사용해 평점을 예측합니다.

- 아이템 기반 방법론은 관련성이 높은 추천을 제공하지만 다양성이 다소 줄어듭니다.
- 이웃 기반 방법론의 속도(효율성)을 높이기 위해 클러스터링을 주로 사용합니다.
- 이웃 기반 방법론은 유사도 값을 사용한 추론 방식으로 가중치를 선택한 선형 모델로 볼 수 있으며, linear regression model을 사용해 이러한 가중치를 학습할 수 있습니다.
- 대부분의 평점 행렬은 희소 행렬이기 때문에 차원 감소나 그래프 기반 모델을 사용해 해결 하곤 합니다.