강의 계획 및 평가 방법

주차	날짜	강의 내용	과제 주제	대면/비대면	평가
1	03/06	강의 소개		Online	
2	03/13	데이터 마이닝 절차		A704	
3	03/20	데이터 탐색 및 시각화		B224	
4	03/27	차원 축소	과제 1	Online	
5	04/03	예측성능 평가		Online	과제 1 (10%)
6	04/10	다중 선형 회귀분석		A704	
7	04/17	중간 프로젝트 발표		A704	30%
8	04/24	k-최근접이웃 알고리즘 나이브 베이즈 분류		Online	
9	05/01 보강: 06/15(목)	휴업일(근로자의 날) 동영상 강의		Online	
10	05/08	분류와 회귀 나무	과제 2	Online	
11	05/15	로지스틱 회귀분석		A704	과제 2 (10%)
12	05/22	신경망 판별 분석		Online	
13	05/29 보강: 06/02(금) 19시	대체 공휴일(부처님 오신 날) 연관 규칙		Online	
14	06/05	군집 분석		A704	
15	06/12	기말 프로젝트 발표		B224	40%

Data Mining for Business Analytics

Ch. 14 Associating Rules and Collaborative Filtering

2023.06.02.

Contents

14.1 Association Rules

- 연관 규칙(Association Rules) 또는 연관성 분석
- “무엇이 무엇과 잘 어울리는지(what goes with what)”를 밝혀내는 것
- ‘장바구니 분석(Market Basket Analysis)’: 서로 다른 아이템의 구매 사이에 의존성을 결정하기 위해 고객의 거래 데이터베이스를 분석

Discovering Association Rules in Transaction Database

- ‘장바구니 데이터베이스’(market basket database) : 많은 양의 거래 레코드로 구성
- 어떤 아이템의 집단이 일관성 있게 구매되고 있는지에 관심이 있음
- ex) 상품 진열의 의사결정, 교차판매, 판매촉진, 카탈로그 디자인, 구매패턴에 근거한 고객 세그먼트(segment) 식별
- “if-then” 문장의 형식으로 연관규칙 제공 (확률적)

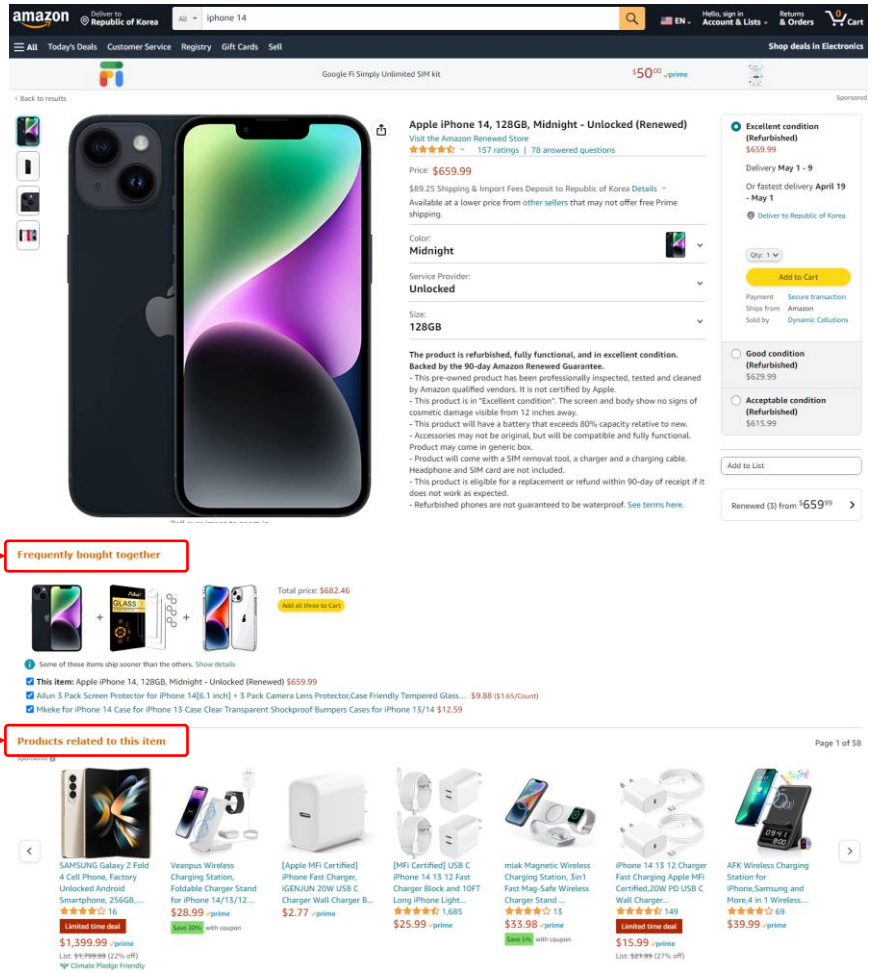
14.1 Association Rules

Discovering Association Rules in Transaction Database

- 온라인 추천 시스템(recommendation systems or recommender systems)에서 주로 사용
- 구매를 위해 아이템을 검토하는 고객에게 1번째 아이템과 함께 종종 구매되는 다른 아이템들을 제시
- ex) [Amazone.com]
  iphone 14를 검색하는 사용자에게 이 휴대폰과 함께 종종 구매되는 케이스와 화면보호기 등을 보여 줌

Frequently bought together

Products related to this item



14.1 Association Rules

Example 1: Synthetic Data on Purchase of Phone Faceplates

- 휴대폰 케이스 구매에 대한 가상 데이터
- 고객들이 6가지 다른 색상의 케이스 중 어떤 색상의 케이스를 같이 구매할 가능성이 큰 지 파악

Transaction	구매된 케이스 색상			
1	red	white	green	
2	white	orange		
3	white	blue		
4	red	white	orange	
5	red	blue		
6	white	blue		
7	red	blue		
8	red	white	blue	green
9	red	white	blue	
10	white			

Generating Candidate Rules

- Idea: 아이템들 사이의 모든 가능한 규칙들을 “if-then” 형식으로 열거한 후 가장 실제 의존성을 잘 표현하는 것들만 선택
- 조건과 결론 부분: 공통 아이템을 갖지 않는 아이템들의 집합(item set)
- 아이템 셋: 사람들이 구매한 것의 기록이 아니라, 하나의 아이템을 포함하는 아이템들의 가능한 조합
- ex) 규칙 예 (조건: red and white, 결론: green)
- “if red and white, then green”

14.1 Association Rules

Generating Candidate Rules

- 첫 단계: 아이템들 사이의 연관성을 표시하는 후보가 될 수 있는 모든 규칙 생성
- 이상적인 경우: p개의 서로 다른 아이템들의 모든 가능한 조합 검토 → 규칙들의 개수: $3^p - 2^{p+1} + 1$
- 현실적인 방법: 빈발 아이템 셋(frequent item set) 고려 (데이터베이스 내에서 빈도 수가 높은 조합)
- Support(지지도)
  - ✓ 전체 거래 개수 대비 조건과 결론에 포함된 모든 아이템 셋의 거래 개수 비율
  - ✓ 얼마나 많은 데이터가 해당 규칙의 타당성을 "support"하는지 계량

Transaction	구매된 케이스 색상			
1	red	white	green	
2	white	orange		
3	white	blue		
4	red	white	orange	
5	red	blue		
6	white	blue		
7	red	blue		
8	red	white	blue	green
9	red	white	blue	
10	white			

- ex) $Support(red \Rightarrow white) = P(\{red, white\}) = \frac{4}{10} = 40\%$
- 빈발 아이템 셋: 사용자에 의해 결정된 선택된 최소 지지도(support)를 초과하는 지지도를 갖는 아이템 셋

$$Support(X \Rightarrow Y) = \frac{n(X \cup Y)}{N} = \frac{\text{조건과 결론의 모든 } itemset\text{이 포함된 거래의 수}}{\text{전체 거래의 수}}$$

14.1 Association Rules

The Apriori Algorithm

- Agrawal(1993)
- 초기에 하나의 아이템 만으로 이루어진 빈발 아이템 셋(one-item sets)을 생성한 후, 2개, 3개의 아이템 등으로 이루어진 아이템 셋들을 모든 크기의 빈발 아이템 셋을 생성할 때까지 재귀적으로 생성
- one-item sets
  - ✓ 각 아이템 별로 support 계산
  - ✓ 요구되는 최소 support 보다 작은 아이템 셋 제외
- **핵심 idea: 어떤 one-item sets가 최소 support를 넘지 않는다면 그것을 포함하는 더 큰 아이템 셋도 최소 support를 넘지 않을 것이다.**
- k-item sets의 생성을 위해서 이전 단계에서 생성된 (k-1)-item sets 사용
- 각 단계에서 전체 데이터베이스를 한 번만 검토 → 데이터베이스 내에서 희귀한 아이템이 많은 경우에도 빠른 수행 속도 보장

14.1 Association Rules

The Apriori Algorithm

Assume: Minimum Support = 3

1-item sets

Itemset	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

2-item sets

Itemset	Count
{Bread, Milk}	3
{Bread, Beer}	2
{Bread, Diaper}	3
{Milk, Beer}	2
{Milk, Diaper}	3
{Beer, Diaper}	3

3-item sets

Itemset	Count
{Bread, Milk, Diaper}	3

✓ 'Coke' 또는 'Eggs'를 포함하는 아이템 셋 후보를 생성하지 않음

✓ 'Bread, Beer' 또는 'Milk, Beer'를 포함하는 아이템 셋 후보를 생성하지 않음

- 아이템들의 모든 조합을 생성하는 경우

$${}^6C_1 + {}^6C_2 + {}^6C_3 = 41$$

- 최소 Support를 적용한 경우

$$6 + 6 + 1 = 13$$

14.1 Association Rules

Selecting Strong Rules

- 목표: 생성된 많은 규칙들 중에서 조건과 결론 아이템 셋 사이에 의존성이 강한 규칙을 찾는 것

Support (지지도) and Confidence (신뢰도)

- Confidence: 조건과 결론에 모두 존재하는 아이템 셋과 조건에 존재하는 아이템 셋 비교

$$Confidence(X \Rightarrow Y) = \frac{n(X \cup Y)}{n(X)} = \frac{\text{조건과 결론의 모든 } itemset\text{이 포함된 거래의 수}}{\text{조건의 } itemset\text{이 포함된 거래의 수}}$$

- ex) 10,000개의 거래 기록
  - ✓ 오렌지 주스와 감기약이 함께 있는 거래: 2,000개, 수프가 포함된 거래: 800개
  - ✓ If (오렌지 주스 and 감기약) then (수프)
  - ✓ $Confidence = \frac{800}{2000} = 40\%$

cf. $Support(X \Rightarrow Y) = \frac{n(X \cup Y)}{N} = \frac{\text{조건과 결론의 모든 } itemset\text{이 포함된 거래의 수}}{\text{전체 거래의 수}}$

14.1 Association Rules

Selecting Strong Rules

Support (지지도) and Confidence (신뢰도)

- Support와 Confidence의 관계
- Support: 임의로 선택된 거래가 조건과 결론의 모든 아이템을 포함할 확률 → 규칙 생성

$$Support = \hat{P}\left(\text{조건 } AND \text{ 결론}\right)$$

- Confidence: 임의로 선택된 거래가 조건의 모든 아이템을 포함한다고 할 때, 결론의 모든 아이템들도 포함할 확률 → 규칙 선택

$$Confidence = \frac{\hat{P}\left(\text{조건 } AND \text{ 결론}\right)}{\hat{P}\left(\text{조건}\right)} = \hat{P}\left(\text{결론} \mid \text{조건}\right)$$

- Confidence가 높음 → 강한 연관규칙
- But, 조건과 결론의 support가 크다면, 서로 상호 독립적이라고 할지라도 confidence가 커질 수 있음
- 거의 모든 고객들이 바나나를 사고, 또 거의 모든 고객들이 아이스크림을 산다면, 두 아이템 사이에 연관이 있든 없든 연관규칙(If 바나나 then 아이스크림)의 confidence는 높아짐

14.1 Association Rules

Selecting Strong Rules

cf. $Confidence = \frac{\hat{P}(\text{조건 } AND \text{ 결론})}{\hat{P}(\text{조건})} = \hat{P}(\text{결론} \mid \text{조건})$

Lift Ratio (향상도)

- 연관규칙의 강도를 판단하는 더 좋은 방법: 규칙의 confidence를 기준값과 비교
- 기준값은 거래 내의 **결론 아이템 셋이 각 규칙의 조건 아이템 셋과 독립**이라고 가정

$$P(\text{조건 } AND \text{ 결론}) = P(\text{조건}) \times P(\text{결론})$$

$$Benchmark\ Confidence = \frac{P(\text{조건}) \times P(\text{결론})}{P(\text{조건})} = P(\text{결론}) = \frac{\text{결론의 } itemset\text{이 포함된 거래의 수}}{\text{데이터베이스 내의 전체 거래 수}}$$

- Lift

$$Lift\ ratio = \frac{confidence}{benchmark\ confidence} = \frac{P(\text{결론} \mid \text{조건})}{P(\text{결론})} = \frac{P(\text{조건 } AND \text{ 결론})}{P(\text{조건}) \times P(\text{결론})}$$

- Lift 〉 1.0 → 규칙에 대한 유용함을 의미
- 조건과 결론의 아이템 셋이 그 둘 사이가 독립적일 때 기대할 수 있는 것보다 더 큰 의미를 가짐

14.1 Association Rules

Selecting Strong Rules

Leverage and Conviction

cf. $Lift\ ratio = \frac{P(\text{결론} \mid \text{조건})}{P(\text{결론})} = \frac{P(\text{조건 } AND \text{ 결론})}{P(\text{조건}) \times P(\text{결론})}$

- Leverage

$$Leverage = P(\text{조건 } AND \text{ 결론}) - P(\text{조건}) \times P(\text{결론})$$

  - ✓ [−1, 1]의 값을 가짐
  - ✓ if 0이면, 조건과 결론의 아이템 셋이 독립을 의미
  - ✓ if leverage 〉 0 → lift 〉 1 if leverage 〈 0 → lift 〈 1
  - ✓ 매출의 경우, 아이템들의 독립적인 판매에 비해 그 아이템들이 얼마나 더 자주 함께 판매되고 있는지 알려 줌

- Conviction: 조건에 대한 확신

$$Conviction = \frac{P(not\ \text{결론})}{P(not\ \text{결론} \mid \text{조건})} = \frac{P(\text{조건}) \times P(not\ \text{결론})}{P(\text{조건 } AND\ not\ \text{결론})}$$

  - ✓ [0, ∞)의 값을 가짐
  - ✓ if 1이면, 조건과 결론의 아이템 셋이 독립을 의미
  - ✓ if ∞ 이면, 조건과 결론의 아이템들이 항상 함께 판매됨
  - ✓ if conviction 〉 1 → 조건의 아이템 셋이 결론의 아이템 셋의 발생 여부를 예측하는데 유용한 품목

14.1 Association Rules

[실습] Table 14.4

Data Format

Transaction	Red	White	Blue	Orange	Green	Yellow
1	1	1	0	0	1	0
2	0	1	0	1	0	0
3	0	1	1	0	0	0
4	1	1	0	1	0	0
5	1	0	1	0	0	0
6	0	1	1	0	0	0
7	1	0	1	0	0	0
8	1	1	1	0	1	0
9	1	1	1	0	0	0
10	0	0	0	0	0	1

✓ Apriori 알고리즘을 사용한다고 가정

{ yellow } count: 1

{ red, orange } count: 1

→ one-item sets 생성 후, drop

Itemset	Count
{ red }	6
{ white }	7
{ blue }	6
{ orange }	2
{ green }	2
{ red, white }	4
{ red, blue }	4
{ red, green }	2
{ white, blue }	4
{ white, orange }	2
{ white, green }	2
{ red, white, blue }	2
{ red, white, green }	2

- Support(count로 가정하는 경우)

{red} = 6 , {red, white, green} = 2

14.1 Association Rules

The Process of Rule Selection

- Step 1: 요구되는 support를 만족하는 아이템 셋인 빈발 아이템 셋을 모두 찾음
  - ✓ 데이터베이스 내에 드물게 발생하는 아이템의 조합을 제거
  - ✓ 많은 계산량이 소요됨
- Step 2: 이 빈발 아이템 셋에서 confidence 요구조건에 맞는 연관규칙들을 생성
  - ✓ 높은 confidence를 갖는 것들을 선별

Itemset	Count
{ red }	6
{ white }	7
{ blue }	6
{ orange }	2
{ green }	2
{ red, white }	4
{ red, blue }	4
{ red, green }	2
{ white, blue }	4
{ white, orange }	2
{ white, green }	2
{ red, white, blue }	2
{ red, white, green }	2

규칙	Confidence	Lift
$\{red, white\} \Rightarrow \{green\}$	$\frac{support\ of \{red, white, green\}}{support\ of \{red, white\}} = \frac{2}{4} = 50\%$	$\frac{confidence\ of\ rule}{benchmark\ confidence} = \frac{50\%}{20\%} = 2.5$
$\{green\} \Rightarrow \{red\}$	$\frac{support\ of \{green, red\}}{support\ of \{green\}} = \frac{2}{2} = 100\%$	$\frac{confidence\ of\ rule}{benchmark\ confidence} = \frac{100\%}{60\%} = 1.67$
$\{white, green\} \Rightarrow \{red\}$	$\frac{support\ of \{white, green, red\}}{support\ of \{white, green\}} = \frac{2}{2} = 100\%$	$\frac{confidence\ of\ rule}{benchmark\ confidence} = \frac{100\%}{60\%} = 1.67$

✓ If 최소 Confidence = 70% → 2, 3번째 규칙 선택

cf. $Confidence = \frac{\hat{P}(\text{조건 } AND \text{ 결론})}{\hat{P}(\text{조건})} = \hat{P}(\text{결론} \mid \text{조건})$

14.1 Association Rules

[실습] Table 14.4

The Process of Rule Selection

- Apriori 알고리즘 적용 결과
  - ✓ minimum support: 20%
  - ✓ minimum confidence: 50%

	antecedents	consequents	support	confidence	lift	leverage
12	(Red, White)	(Green)	0.2	0.5	2.500000	0.12
15	(Green)	(Red, White)	0.2	1.0	2.500000	0.12
4	(Green)	(Red)	0.2	1.0	1.666667	0.08
14	(Green, White)	(Red)	0.2	1.0	1.666667	0.08
7	(Orange)	(White)	0.2	1.0	1.428571	0.06
8	(Green)	(White)	0.2	1.0	1.428571	0.06

Interpreting the Results

- 규칙 #7의 해석: 만약 'orange' 색이 구매되면 100%의 신뢰도로 'white'도 구매된다. 이 규칙의 lift는 1.43
- Support: 전체적인 크기에 대한 영향력을 시사함
  - ✓ 적은 양의 거래들만 영향을 받는다면 그 규칙은 유용성이 떨어질 수 있음
- Lift: 무작위 선택과 비교해서 해당 규칙이 결론을 찾는데 얼마나 효율적인지 보여줌
  - ✓ 매우 효율적인 규칙이 비효율적인 규칙에 비해 선호되지만, 여전히 support를 고려해야 함
  - ✓ 매우 낮은 support를 갖는 매우 효율적인 규칙은 훨씬 더 큰 support를 갖는 덜 효율적인 규칙만큼 바람직하지 않을 수 있음
- Confidence: 어느 정도로 결론이 찾아질 지 알려주기 때문에 규칙의 실질적인 유용성을 결정하는 데 유용
  - ✓ confidence가 작은 규칙으로 얻어진 결론은 빈도가 너무 작아서 조건과 관련있는 모든 거래에서 그 결론에 투자할 만한 가치가 거의 없음

cf. $Lift\ ratio = \frac{confidence}{benchmark\ confidence} = \frac{P(\text{결론} \mid \text{조건})}{P(\text{결론})}$ $Confidence = \frac{\text{조건과 결론의 모든 } itemset\text{이 포함된 거래의 수}}{\text{조건의 } itemset\text{이 포함된 거래의 수}}$

14.1 Association Rules

[실습] Table 14.6

Rules and Chance

- 통계적인 관점에서 의미있는 규칙인가에 관한 질문: “단지 우연히 발생하는 연관성을 발견하고 있는가?”
- ex) transactions: 50개, items: 9개

Transaction	Items	Transaction	Items	Transaction	Items
1	8	18	8	35	3 4 6 8
2	3 4 8	19		36	1 4 8
3	8	20	9	37	4 7 8
4	3 9	21	2 5 6 8	38	8 9
5	9	22	4 6 9	39	4 5 7 9
6	1 8	23	4 9	40	2 8 9
7	6 9	24	8 9	41	2 5 9
8	3 5 7 9	25	6 8	42	1 2 7 9
9	8	26	1 6 8	43	5 8
10		27	5 8	44	1 7 8
11	1 7 9	28	4 8 9	45	8
12	1 4 5 8 9	29	9	46	2 7 9
13	5 7 9	30	8	47	4 6 9
14	6 7 8	31	1 5 8	48	9
15	3 7 9	32	3 6 9	49	9
16	1 4 9	33	7 9	50	6 7 8
17	6 7 8	34	7 8 9		

	antecedents	consequents	support	confidence	lift	leverage
3	(8, 3)	(4)	0.04	1.0	4.545455	0.0312
1	(1, 5)	(8)	0.04	1.0	1.851852	0.0184
2	(2, 7)	(9)	0.04	1.0	1.851852	0.0184
4	(3, 4)	(8)	0.04	1.0	1.851852	0.0184
5	(3, 7)	(9)	0.04	1.0	1.851852	0.0184
6	(4, 5)	(9)	0.04	1.0	1.851852	0.0184

- 규칙 #3의 lift
  - ✓ 아이템 4가 아이템 셋 {3, 8}과 연관되지 않았다고 할 때 보다, 아이템 3과 8을 구매하는 경우에 아이템 4도 함께 구매할 가능성이 4.5배가 됨을 의미함
- But, 이 데이터 사이에는 근본적으로 연관관계가 없음(임의로 생성된 데이터)

14.1 Association Rules

Rules and Chance

- 통계적인 관점에서 의미있는 규칙인가에 관한 질문: “단지 우연히 발생하는 연관성을 발견하고 있는가?”
- 우연성에 의해서 유발될 수 있는 가짜 연관성을 평가하는 원칙

1. 보다 많은 레코드에 기반한 규칙일수록 결론이 좀 더 견고함
2. 더 많은 분명한 규칙들을 세밀하게 고려할수록, 적어도 일부가 우연한 표본추출의 결과에 근거할 가능성이 더 큼
   → 고려하는 규칙이 많아질수록 우연의 가능성이 높아짐

   ex) 한 사람이 동전을 10번 던져서 10번의 앞면이 나왔다면 매우 놀라운 일

   but, 만일 1000명이 동전을 10번씩 던져서 그 중 10번의 앞면이 나온 사람이 있다면 그다지 놀랍지 않을 수 있음

- 적절한 접근 방법
  - ✓ 실제 적용성에 대해서 하향식(top-down)으로 규칙을 고려하면서, 인간의 의사결정 과정에 적절히 관여할 수 있는 것 이상은 고려하지 않음

14.1 Association Rules

[실습] Table 14.8

Example 2: Rules for Similar Book Purchases

- 다양한 유형의 서적들과 관련된 거래 사이의 연관성 분석
- transaction: 2,000개, item(서적): 11가지 유형
- minimum support: 5% (200/4000)
- minimum confidence: 50%
- 81개의 규칙 생성

	ChildBks	YouthBks	CookBks	DoItYBks	RefBks	ArtBks	GeogBks	ItalCook	ItalAtlas	ItalArt	Florence
0	0	1	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0
2	1	1	1	0	1	0	1	1	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0

14.1 Association Rules

[실습] Table 14.8

Example 2: Rules for Similar Book Purchases

- 동일한 유형의 서적이 조건과 결론에 나타나는 규칙
  - ✓ { Child, Cook, Dolt, Ref }
    → rule # 70, 71, 72, 73
  - ✓ { Child, Youth, Cook, Dolt }
    → rule # 56, 58, 60
  - ✓ { Child, Cook, Dolt }
    → rule # 22, 25
- 해당 규칙이 유용하지 않음을 의미하지 않음 → 비즈니스 측면에서 가능한 조치를 고려하여 아이템 셋의 개수를 줄일 수 있음

Number of rules 81

	antecedents	consequents	support	confidence	lift	leverage
64	(YouthBks, RefBks)	(ChildBks, CookBks)	0.05525	0.68000	2.80992	0.03559
73	(DoltYBks, RefBks)	(ChildBks, CookBks)	0.06125	0.66216	2.73621	0.03886
60	(DoltYBks, YouthBks)	(ChildBks, CookBks)	0.06700	0.64891	2.68145	0.04201
80	(GeogBks, RefBks)	(ChildBks, CookBks)	0.05025	0.61468	2.54000	0.03047
69	(GeogBks, YouthBks)	(ChildBks, CookBks)	0.06325	0.60526	2.50109	0.03796
77	(GeogBks, DoltYBks)	(ChildBks, CookBks)	0.06050	0.59901	2.47525	0.03606
66	(GeogBks, ChildBks, CookBks)	(YouthBks)	0.06325	0.57763	2.42445	0.03716
70	(ChildBks, CookBks, RefBks)	(DoltYBks)	0.06125	0.59179	2.32301	0.03488
47	(GeogBks, DoltYBks)	(YouthBks)	0.05450	0.53960	2.26486	0.03044
62	(ChildBks, CookBks, RefBks)	(YouthBks)	0.05525	0.53382	2.24057	0.03059
58	(ChildBks, CookBks, DoltYBks)	(YouthBks)	0.06700	0.52446	2.20131	0.03656
56	(ChildBks, YouthBks, CookBks)	(DoltYBks)	0.06700	0.55833	2.19169	0.03643
33	(ChildBks, RefBks)	(DoltYBks)	0.07100	0.55361	2.17314	0.03833
74	(GeogBks, ChildBks, CookBks)	(DoltYBks)	0.06050	0.55251	2.16884	0.03260
19	(GeogBks, ChildBks)	(YouthBks)	0.07550	0.51624	2.16680	0.04066
46	(GeogBks, CookBks)	(YouthBks)	0.08025	0.51360	2.15572	0.04302
61	(ChildBks, YouthBks, RefBks)	(CookBks)	0.05525	0.89113	2.14471	0.02949
16	(ChildBks, YouthBks)	(DoltYBks)	0.08025	0.54407	2.13569	0.04267
51	(CookBks, RefBks)	(DoltYBks)	0.07450	0.53309	2.09262	0.03890
28	(RefBks)	(ChildBks, CookBks)	0.10350	0.50549	2.08882	0.05395
72	(DoltYBks, CookBks, RefBks)	(ChildBks)	0.06125	0.82215	2.08667	0.03190
15	(YouthBks)	(ChildBks, CookBks)	0.12000	0.50367	2.08129	0.06234
71	(ChildBks, DoltYBks, RefBks)	(CookBks)	0.06125	0.86268	2.07624	0.03175
22	(ChildBks, CookBks)	(DoltYBks)	0.12775	0.52789	2.07220	0.06610
25	(DoltYBks)	(ChildBks, CookBks)	0.12775	0.50147	2.07220	0.06610

14.1 Association Rules

[실습] Table 14.8

Example 2: Rules for Similar Book Purchases

- 조건과 결론의 아이템 셋을 구성하는 아이템의 개수를 줄인 결과
- 조건: 2개 이하
- 결론: 1개

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
47	(GeogBks, DoItYBks)	(YouthBks)	0.10100	0.23825	0.05450	0.539604	2.264864	0.030437	1.654554
33	(ChildBks, RefBks)	(DoItYBks)	0.12825	0.25475	0.07100	0.553606	2.173135	0.038328	1.669490
19	(GeogBks, ChildBks)	(YouthBks)	0.14625	0.23825	0.07550	0.516239	2.166797	0.040656	1.574642
46	(GeogBks, CookBks)	(YouthBks)	0.15625	0.23825	0.08025	0.513600	2.155719	0.043023	1.566098
16	(ChildBks, YouthBks)	(DoItYBks)	0.14750	0.25475	0.08025	0.544068	2.135693	0.042674	1.634563
51	(CookBks, RefBks)	(DoItYBks)	0.13975	0.25475	0.07450	0.533095	2.092619	0.038899	1.596148
22	(ChildBks, CookBks)	(DoItYBks)	0.24200	0.25475	0.12775	0.527893	2.072198	0.066101	1.578560
48	(GeogBks, YouthBks)	(DoItYBks)	0.10450	0.25475	0.05450	0.521531	2.047227	0.027879	1.557573
42	(YouthBks, CookBks)	(DoItYBks)	0.16100	0.25475	0.08375	0.520186	2.041948	0.042735	1.553207
43	(YouthBks, RefBks)	(CookBks)	0.08125	0.41550	0.06825	0.840000	2.021661	0.034491	3.653125

14.2 Collaborative Filtering

- 추천 시스템(Recommendation System)
  - ✓ 사용자의 정보 뿐만 아니라 비슷한 다른 사용자들의 정보에 기반하여 개인 맞춤화된 추천 제공
  - ✓ ex) Amazon, Netflix, Google, Spotify, Pandora, …
- 협업 필터링(Collaborative Filtering)
  - ✓ 사용자들의 다양한 선호도를 고려하여(“collaboration”), 방대한 양의 아이템 셋으로부터 특정 사용자와 관련있는 아이템을 식별한다(“filtering”)는 개념
- 아마존의 협업 필터링: item-to-item collaborative filtering
  - ✓ 한 사용자가 과거에 무엇을 구매했는지, 어떤 항목들이 실제로 그 사용자의 가상의 쇼핑 카트에 담겼었고, 어떤 아이템들을 그들이 선호했는지, 그리고 다른 고객들은 무엇을 관심있게 보고 구매하였는지와 같은 정보 활용

14.2 Collaborative Filtering

Data Type and Format

- 모든 아이템-사용자 정보 필요 → 각각의 아이템-사용자 조합에서 그 아이템에 대한 사용자의 선호도를 측정하는 측도(measure) 필요
- 선호도: 점수화된 평가(rating) or 이진 행동(하나의 구매, "like" 또는 한번의 클릭과 같은)
- 사용자 (n 명): $(u_1, u_2, \cdots, u_n)$
- 아이템 (p 개): $(i_1, i_2, \cdots, i_p)$ → $n \times p$ 행렬
- 행렬의 각 셀: 각 아이템에 대한 특정 사용자의 선호도에 해당하는 점수 또는 이진화된 사건
- 구매 내역 행렬은 0을 많이 가지고 있고(sparse), 평가점수 행렬은 결측값을 많이 포함하고 있음(때때로 이 결측값은 '흥미없음'을 의미하기도 함)

	Item ID			
User ID	I_1	I_2	...	I_p
U_1	$r_{1,1}$	$r_{1,2}$	...	$r_{1,p}$
U_2	$r_{2,1}$	$r_{2,2}$	...	$r_{2,p}$
⋮	⋮	⋮	⋮	⋮
U_n	$r_{n,1}$	$r_{n,2}$	...	$r_{n,p}$

- 사용자와 아이템의 수가 많다면, 선호도 데이터($r_{u,i}$)를 행렬 대신에 하나의 행으로 표현하는 것이 효과적
- 행의 각 셀: 사용자 ID, 아이템, 선호도 정보를 지닌 트리플($U_u, I_i, r_{u,i}$)을 포함

14.2 Collaborative Filtering

Example 3: Netflix Prize Contest

- 2006년 시네매치(Cinematch)라고 불리우는 추천 시스템의 성능을 향상시키기 위한 목적으로 100만 달러가 걸린 경연대회(www.netflixprize.com) 개최
- 데이터 셋: 영화에 대한 고객 평가(rating: 1 to 5)를 포함하는 [사용자 ID, 영화 ID, 평가점수, 날짜]
- 영화에 대한 평가점수 뿐만 아니라, 영화가 특정 고객에 의해서 평가가 되었는지 여부를 고려함
- 고객이 평가를 해야겠다고 결정한 영화들에 대한 정보가 고객의 선호도에 매우 중요하게 작용했음 → 평가점수에 대한 정보를 '평가됨/평가되지 않음'으로 구성된 이진 행렬로 변환시키는 것이 유용할 수 있음

	Movie ID								
User ID	**1**	**5**	**8**	**17**	**18**	**28**	**30**	**44**	**48**
30878	4	1			3	3	4	5	
124105	4								
822109	5								
823519	3		1	4		4	5		
885013	4	5							
893988	3						4	4	
1248029	3					2	4		3
1503895	4								
1842128	4						3		
2238063	3								

14.2 Collaborative Filtering

User-Based Collaborative Filtering: “People Like You”

- Idea: 비슷한 선호도를 지닌 사람들을 찾음. 그리고 그들이 좋아하지만 아직 구매하지 않은 아이템을 추천
  - ✓ 관심 대상의 사용자와 가장 비슷한 사용자(이웃)를 찾음. 사용자의 선호도와 다른 사용자들의 선호 비교
  - ✓ 사용자가 아직 구매하지 않은 아이템들 만을 고려. 그 사용자의 이웃들이 가장 선호하는 것을 추천
- 아마존의 “이 물품을 구매한 고객들이 또 다시 구매한 물품… (Customers who bought this item also bought…)”에 내재된 방식
- 구글 검색에서 각각의 검색 결과 근처에 보이는 “유사한 페이지(Similar pages)” 링크를 만드는 데 사용

- 사용자 기반 최우선-N 추천(User-based top-N recommendation)
  - ✓ 1단계: 사용자와 다른 사용자들 간의 거리를 측정하는 거리(또는 근접성) metric의 선택 필요
  - ✓ 거리나 필요한 이웃들의 개수에 threshold 적용하여 최근접 이웃 결정

14.2 Collaborative Filtering

User-Based Collaborative Filtering: "People Like You"

	Movie ID								
User ID	1	5	8	17	18	28	30	44	48
30878	4	1			3	3	4	5	
823519	3		1	4		4	5		

- 근접성 측정 방법: **피어슨 상관 계수(Pearson Correlation)**
- 사용자 U_1 의 아이템 $I_1, I_2, \cdots, I_p$ 들에 대한 평가점수: $r_{1,1}, r_{1,2}, \cdots, r_{1,p}$, 평균: $\bar{r}_1$
- 사용자 U_2 의 아이템 $I_1, I_2, \cdots, I_p$ 들에 대한 평가점수: $r_{2,1}, r_{2,2}, \cdots, r_{2,p}$, 평균: $\bar{r}_2$
- 피어슨 상관 근접성(Pearson Correlation Proximity)

※ 두 사용자 모두에게 평가된 항목들만 반영

$$Corr(U_1, U_2) = \frac{\sum(r_{1,i} - \bar{r}_1)(r_{2,i} - \bar{r}_2)}{\sqrt{\sum(r_{1,i} - \bar{r}_1)^2}\sqrt{\sum(r_{2,i} - \bar{r}_2)^2}}$$

- ex) 사용자 30878과 823519의 상관계수 계산
- 사용자들의 평가점수 평균 계산

$$\bar{r}_{30878} = \frac{4+1+3+3+4+5}{6} = 3.333 \qquad \bar{r}_{823519} = \frac{3+1+4+4+5}{5} = 3.4$$

$$Corr(U_{30878}, U_{823519}) = \frac{(4-3.333)(3-3.4) + (3-3.333)(4-3.4) + (4-3.333)(5-3.4)}{\sqrt{(4-3.333)^2 + (3-3.333)^2 + (4-3.333)^2}\sqrt{(3-3.4)^2 + (4-3.4)^2 + (5-3.4)^2}}$$

$$= \frac{0.6}{1.75} = 0.34$$

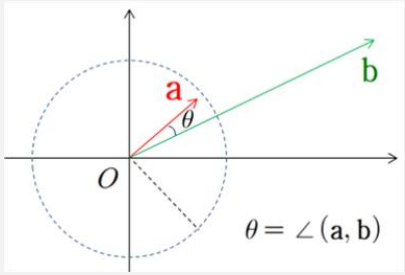
14.2 Collaborative Filtering

User-Based Collaborative Filtering: "People Like You"

	Movie ID								
User ID	1	5	8	17	18	28	30	44	48
30878	4	1			3	3	4	5	
823519	3		1	4		4	5		

- 근접성 측정 방법: **코사인 유사도(Cosine Similarity)**
- 사용자 U_1 의 아이템 $I_1, I_2, \cdots, I_p$ 들에 대한 평가점수: $r_{1,1}, r_{1,2}, \cdots, r_{1,p}$, 평균: $\bar{r}_1$
- 코사인 유사도(Cosine Similarity)

$$CosSim(U_1, U_2) = \frac{\sum(r_{1,i} \cdot r_{2,i})}{\sqrt{\sum {r_{1,i}}^2}\sqrt{\sum {r_{2,i}}^2}}$$



- ex) 사용자 30878과 823519의 코사인 유사도 계산

$$CosSim(U_{30878}, U_{823519}) = \frac{(4\times3)+(3\times4)+(4\times5)}{\sqrt{4^2+3^2+4^2}\ \sqrt{3^2+4^2+5^2}} = \frac{44}{45.277} = 0.972$$

- 데이터가 이진 행렬(구매 또는 비구매)일 때, 코사인 유사도는 사용자가 구매한 모든 아이템들로 계산해야 함

- 2단계: k-최근접 사용자들을 관찰하고, 그들이 평가하고 구매한 다른 모든 아이템들 중에서 최고 항목을 선정하여 사용자에게 추천
- 최고 항목
  - ✓ 이진화된 구매 데이터의 경우, 가장 많이 구매된 항목
  - ✓ 평가점수 데이터의 경우, 가장 좋게 평가된, 혹은 이 둘의 가중치의 아이템

14.2 Collaborative Filtering

Item-Based Collaborative Filtering

- 사용자들의 수가 아이템의 수보다 훨씬 큰 경우, 비슷한 사용자들보다 비슷한 아이템들을 찾는 것이 효율적이고 빠름
- Idea: 이 아이템을 좋아한 사용자는, 'A' 아이템도 좋아할 것이다.
- 사용자가 특정 아이템에 관심을 표현할 때, 아이템 기반 협업 필터링 알고리즘
- (임의의 사용자가) 관심을 가지는 항목과 공동으로 평가 혹은 구매한 아이템들을 찾음
- 비슷한 아이템들 중에서 가장 인기있거나 상관관계가 높은 항목 추천
- ex) 영화 1: 평균 $\bar{r}_1 = 3.7$ 영화 5: 평균 $\bar{r}_5 = 3$

$$Corr(I_1, I_5) = \frac{(4-3.7)(1-3)+(4-3.7)(5-3)}{\sqrt{(4-3.7)^2+(4-3.7)^2}\sqrt{(1-3)^2+(5-3)^2}} = 0$$

	Movie ID	
User ID	1	5
30878	4	1
124105	4	
822109	5	
823519	3	
885013	4	5
893988	3	
1248029	3	
1503895	4	
1842128	4	
2238063	3	

- 모든 영화들 간의 유사도 계산 후, 만약 어떤 사용자가 특정한 영화를 높이 평가하였다면 그 영화와 가장 높은 양의 상관관계를 가지는 영화를 추천
- 단점: 사용자들의 취향에 비해 아이템들 간의 다양성이 작아서 추천이 너무 명백해 보임

14.2 Collaborative Filtering

[실습] Table 14.11

Collaborative Filtering in Python

```
## User-based filtering
# compute cosine similarity between users
sim_options = {'name': 'cosine', 'user_based': True}
algo = KNNBasic(sim_options=sim_options)
algo.fit(trainset)

# Than predict ratings for all pairs (u, i) that are NOT in the training set.
predictions = algo.test(testset)

top_n = get_top_n(predictions, n=4)

# Print the recommended items for each user
print()
print('Top-4 recommended items for each user')
for uid, user_ratings in list(top_n.items())[:5]:
    print('User {}'.format(uid))
    for prediction in user_ratings:
        print('  Item {0.iid} ({0.est:.2f})'.format(prediction), end='')
    print()
print()
```

```
Computing the cosine similarity matrix...
Done computing similarity matrix.

Top-4 recommended items for each user
User 6
  Item 6 (5.00)  Item 77 (2.50)  Item 60 (1.00)
User 222
  Item 77 (3.50)  Item 75 (2.78)
User 424
  Item 14 (3.50)  Item 45 (3.10)  Item 54 (2.34)
User 87
  Item 27 (3.00)  Item 54 (3.00)  Item 82 (3.00)  Item 32 (1.00)
User 121
  Item 98 (3.48)  Item 32 (2.83)
```

```
## Item-based filtering
# compute cosine similarity between users
sim_options = {'name': 'cosine', 'user_based': False}
algo = KNNBasic(sim_options=sim_options)
algo.fit(trainset)

# Than predict ratings for all pairs (u, i) that are NOT in the training set.
predictions = algo.test(testset)
top_n = get_top_n(predictions, n=4)

# Print the recommended items for each user
print()
print('Top-4 recommended items for each user')
for uid, user_ratings in list(top_n.items())[:5]:
    print('User {}'.format(uid))
    for prediction in user_ratings:
        print('  Item {0.iid} ({0.est:.2f})'.format(prediction), end='')
    print()
```

```
Computing the cosine similarity matrix...
Done computing similarity matrix.

Top-4 recommended items for each user
User 6
  Item 77 (3.00)  Item 60 (3.00)  Item 6 (3.00)
User 222
  Item 77 (2.24)  Item 75 (2.00)
User 424
  Item 54 (3.47)  Item 14 (3.44)  Item 45 (3.00)
User 87
  Item 27 (3.00)  Item 32 (3.00)  Item 82 (3.00)  Item 54 (2.50)
User 121
  Item 32 (3.06)  Item 98 (2.31)
```

14.2 Collaborative Filtering

Advantages and Weaknesses of Collaborative Filtering

- ‘Long Tail(비주류)’ 아이템들에 대해서도 유용한 추천 제공 → 비슷한 사용자들을 충분히 보유하고 있다면, 그래서 각 사용자가 유사한 성향의 다른 사용자들을 발견할 수 있기 때문
- 한계점: 새로운 사용자들이나 새로운 아이템에 대한 추천 불가능 → 다양한 접근 방법 존재
- 사용자-기반 협업 필터링(User-based CF)
  - ✓ 낮게 평가되거나 원하지 않는 아이템들의 데이터는 고려하지 못 함 → 원하지 않는 아이템을 찾을 수 없음
  - ✓ 사용자의 수가 매우 많아지면 계산의 어려움이 있음 → 아이템 기반 알고리즘, 사용자들의 군집화, 차원 축소, 특이값 분해(SVD: Singular Value Decomposition) 등의 방법 사용
- 본질적으로 비지도 학습 → 사용자들의 feedback을 받아서 추천의 적합성 검토에 활용

14.2 Collaborative Filtering

Collaborative Filtering vs. Association Rules

	Association Rules	Collaborative Filtering
Frequent itemsets vs. Personalized recommendations (빈발 아이템 셋 vs. 개인 맞춤형 추천)	• 빈발 아이템의 조합을 찾고, 오직 찾은 아이템들에 대한 추천 제공 • 사용자 선호도의 head(주류)를 찾음 • 수많은 거래 데이터 필요 • 각 사용자에 대한 다수의 거래를 포함할 수 있음 • 포괄적이고 객관성을 지닌 규칙을 생성하므로 일반적인 전략 수립에 사용(상점의 제품 배치, 진단검진의 순서 설정 등)	• 모든 아이템들에 대해 개인 맞춤형(personalized) 추천 제공하여 특이한 선호도를 가진 사용자에게도 추천 제공 • 사용자 선호도의 long-tail 포착 가능 • 여러 사용자로부터 최대한 많은 아이템 데이터 필요 • 사용자 수준에서 적용 가능 • 특정 사용자를 위한 추천 생성 및 개인 맞춤형 도구
Transaction data vs. User data (거래 데이터 vs. 사용자 데이터)	• '다수의 거래/장바구니' 안에 있는 다른 아이템과의 공동 구매 내역을 기반으로 추천 • 같은 아이템이 반복적으로 구매될 경우(식료품 쇼핑 등) 각각의 장바구니를 고려하는 것이 유용	• 적을 수도 있는 다른 '사용자'와의 공동 구매 내역 또는 동일한 평가점수를 바탕으로 추천 • 각 아이템이 주로 한 번 구매되거나 평가점수가 매겨질 경우(책, 음반, 영화 등) 각각의 사용자를 고려하는 것이 유용
Binary data and Ratings data (이진 데이터와 평가점수 데이터)	• 각 아이템을 이진 데이터('1': 구매, '0': 비구매)로 처리	• 이진 데이터 및 수치화된 평가점수 데이터 모두 활용 가능
Two or more items (2개 이상의 아이템)	• 조건과 결론 모두 한 개 이상의 아이템 포함 가능 → 하나의 추천은 여러 아이템들로 이루어진 하나의 묶음일 수 있음(우유, 쿠키, 시리얼 모두 구입 시 10% 할인 등)	• 단일 아이템 추천(당신과 비슷한 유형의 사람들이 구매한 가장 인기있는 아이템) 또는 전혀 연관성이 없을 수 있는 여러 단일 아이템(당신과 비슷한 유형의 사람들이 구매한 가장 인기있는 2개의 아이템) 추천

14.3 Summary

Association Rule

- 같이 구매된 아이템들에 대한 일반적인 규칙을 찾음 → “if X가 구매되면, then Y도 구매될 가능성이 높다” → 간단하고 명확한 규칙 생성
- 규칙 생성 과정
  1. 빈발 아이템셋에 근거한 후보 규칙들의 집합 생성(Apriori 알고리즘 등)
  2. 후보 규칙들로부터 아이템들 사이에 가장 강한 연관성을 보여주는 규칙들 생성
- 규칙의 불확실성을 평가하기 위해 Support와 Confidence 지표 사용
- 향상비(Lift ratio)는 임의의 조합과 비교함으로써 실제 연관성을 탐색하기 위한 규칙의 효율성 비교

- 단점
  - ✓ 생성되는 규칙이 너무 많음 → 유용하고 강한 규칙들로 이루어진 작은 집합으로 줄이기 위한 방법 필요
  - ✓ 희소한 조합은 최소 지지도 조건을 맞추지 못하기 때문에 무시되는 경향이 있음 → 데이터 상 거의 동일하게 빈발하는 아이템들을 활용하는 게 더 좋음(예를 들어 서점 거래 데이터베이스에서 연관규칙을 얻을 때, 개별 서적의 제목보다는 서적의 유형을 사용)

14.3 Summary

Collaborative Filtering

- 아이템을 구매하거나 높은 평가점수를 매기는 등 한 아이템에 대해 비슷한 행동을 취한 사용자들로부터 형성된 아이템 사이의 관계를 기반으로 함
- 사용자 기반 협업 필터링(User-based CF)
  - ✓ 아이템-사용자 조합의 데이터를 활용하여 사용자 간 유사도 계산 후 각 사용자 별 맞춤형(personalized) 추천 제공
- 중요 요소
  - ✓ 추천에 대해 사용자들로부터 피드백을 받는다는 점
  - ✓ 사용자들이 각 아이템에 대해 충분한 정보를 가져야 한다는 점
- 단점
  - ✓ 새로운 사용자 또는 아이템들에 대해 추천을 생성할 수 없다는 점
  - ✓ 수많은 사용자들이 존재하는 경우 계산과정이 복잡해짐 → Item-based CF, 차원 축소 기법 등의 방법 활용