

1. 데이터 Import

1. item_purchase_data.csv -> item_purchase_data 테이블

2. 데이터 이해

1. 2022-08-01부터 2022-09-01까지 6,620명 고객의 일/시별 구매상품 데이터
2. 상품 종류(product_id) = 13,506

SQL문을 활용한 장바구니 분석

A 상품ID, B(함께 구매한 상품) 상품 ID, A 상품 구매횟수 , B 상품 구매횟수, A와 B를 함께 구매한 횟수

A 상품 ID	B 상품 ID	A 상품 구매횟수	B 상품 구매횟수	A와 B를 함께 구매한 횟수
0	1	1,167	1,515	863
1	4	1,515	864	252
0	4	1,167	864	213
1	18	1,515	724	183
1	34	1,515	371	152
1	71	1,515	282	132
0	18	1,167	724	132
0	71	1,167	282	129
1	5	1,515	454	122
0	34	1,167	371	117
0	5	1,167	454	88
1	48	1,515	356	85
1	16	1,515	194	76
1	2	1,515	287	74

A 상품ID, B(함께 구매한 상품) 상품 ID, 지지도(support) , 신뢰도(confidence), 향상도(lift)

A 상품 ID	B 상품 ID	지지도(support)	신뢰도(confidence)	향상도(lift)
1	2	0.01	0.05	2.30
8	9	0.00	0.11	30.08
1	14	0.00	0.00	1.89
1	7	0.01	0.05	2.54
0	1	0.06	0.74	6.59
7	18	0.00	0.04	0.65
1	19	0.00	0.03	2.27
1	20	0.00	0.03	1.93
1	21	0.00	0.00	1.52
20	21	0.00	0.00	1.76
0	23	0.00	0.02	4.65
1	23	0.00	0.01	3.77
0	24	0.00	0.00	3.57
1	24	0.00	0.00	3.85

Mysql 데이터베이스에 Import한 데이터셋을 pymysql을 통해 조회

```
import pymysql
import pandas as pd

connect = pymysql.connect(host='localhost', port=3306, user=user, passwd=passwd, db=db, charset='utf8')
cursor = connect.cursor(pymysql.cursors.DictCursor)
sql = "select * from item_purchase_data"
cursor.execute(sql)
result = cursor.fetchall()
df = pd.DataFrame(result)
df
```

	browser_id	session_id	event_type	product_id	datetime
0	65488256b12588728948ddc7c02af98038af3966942a17...	359c87d01bbc948d9a328e5245b81ce3d5cbab58d93d08...	purchase	0	2022-08-01 21:56:55
1	9989477ea6b57bcc1b6b8aa0bd564c81e3c12a52daaeb5...	7cc8ce3f93fa8bd8b11cffd532f00f1ac86e3940854017...	purchase	1	2022-08-01 21:56:44
2	9989477ea6b57bcc1b6b8aa0bd564c81e3c12a52daaeb5...	7cc8ce3f93fa8bd8b11cffd532f00f1ac86e3940854017...	purchase	2	2022-08-01 21:47:44
3	32d56c5de512f40a5b232a6deec6549108dd26afa744db...	a0ca9aa2f91dc60d17185cce6f5b9d087e1eb93a227374...	purchase	0	2022-08-02 05:36:21
4	b66448459b0346f097bd292a96b374f840633fd9f7de78...	e8f82c9918d1c401c8228699125545ccd955868b16e2ad...	purchase	3	2022-08-02 01:41:53
...
13501	316ff4f8d2581386a5589fec2c415a0969a3ad164c2dd3...	870d215c0aebbd7bc88769b531a8277a477e016ba76ced...	purchase	71	2022-08-21 12:50:32
13502	316ff4f8d2581386a5589fec2c415a0969a3ad164c2dd3...	870d215c0aebbd7bc88769b531a8277a477e016ba76ced...	purchase	438	2022-08-21 12:56:32
13503	83d41a9b8440c02c486a85885bdf64476668498e9f964e...	7b84d3d2efd5a69d7d3c4021deb8ac0e2c0142caf1eaed...	purchase	106	2022-08-21 14:41:44
13504	7a55e40674b9f746752fa500e68077038d0482aa26d40f...	45f4c81cb69106ff5f61c94c45a772c73604e6edcdd623...	purchase	5	2022-08-22 01:10:41
13505	1c3d5f3c64cfc1cce223057ebdcbffd0ca8a6119006a64...	2389bb63de609ffd86d564631f292fee243c75577bcc39...	purchase	20	2022-08-21 19:59:37

13506 rows × 5 columns

apriori를 활용한 장바구니 분석

A 상품ID, B(함께 구매한 상품) 상품 ID,
지지도(support), 신뢰도(confidence),
향상도(lift)

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(0)	(1)	0.331349	0.433760	0.221197	0.667565	1.539019	0.077471	1.703311
1	(1)	(0)	0.433760	0.331349	0.221197	0.509952	1.539019	0.077471	1.364461
2	(4)	(1)	0.115213	0.433760	0.064305	0.558140	1.286747	0.014330	1.281490
3	(52)	(1)	0.021137	0.433760	0.010717	0.507042	1.168946	0.001549	1.148658
4	(71)	(1)	0.070259	0.433760	0.036916	0.525424	1.211323	0.006440	1.193148
5	(132)	(1)	0.021435	0.433760	0.011015	0.513889	1.184731	0.001718	1.164836
6	(0, 4)	(1)	0.050313	0.433760	0.033343	0.662722	1.527854	0.011520	1.678852
7	(1, 4)	(0)	0.064305	0.331349	0.033343	0.518519	1.564873	0.012036	1.388737
8	(0, 5)	(1)	0.025901	0.433760	0.017565	0.678161	1.563447	0.006330	1.759388
9	(1, 5)	(0)	0.034534	0.331349	0.017565	0.508621	1.535002	0.006122	1.360764
10	(0, 7)	(1)	0.015779	0.433760	0.010122	0.641509	1.478950	0.003278	1.579511
11	(1, 7)	(0)	0.018160	0.331349	0.010122	0.557377	1.682147	0.004105	1.510657
12	(0, 18)	(1)	0.035427	0.433760	0.025901	0.731092	1.685477	0.010534	2.105705
13	(1, 18)	(0)	0.050908	0.331349	0.025901	0.508772	1.535458	0.009032	1.361183
14	(0, 34)	(1)	0.029175	0.433760	0.021733	0.744898	1.717304	0.009078	2.219661
15	(1, 34)	(0)	0.039297	0.331349	0.021733	0.553030	1.669029	0.008712	1.495966
16	(0, 48)	(1)	0.016374	0.433760	0.011313	0.690909	1.592837	0.004211	1.831953
17	(48, 1)	(0)	0.021733	0.331349	0.011313	0.520548	1.570998	0.004112	1.394616
18	(0, 71)	(1)	0.033046	0.433760	0.022030	0.666667	1.536948	0.007697	1.698720
19	(1, 71)	(0)	0.036916	0.331349	0.022030	0.596774	1.801046	0.009798	1.658255

장바구니 분석 결과에 대한 해석과 마케팅 액션 플랜 제안

[표 1]

A 상품 ID	B 상품 ID	지지도(support)	신뢰도(confidence)	향상도(lift)
433	436	0.00	1.00	10,000.00
199	358	0.00	1.00	10,000.00
392	393	0.00	1.00	10,000.00
325	387	0.00	0.67	6,666.67
242	243	0.00	0.57	5,714.29
184	185	0.00	0.50	5,000.00
94	413	0.00	0.50	5,000.00
277	354	0.00	1.00	5,000.00
286	347	0.00	0.50	5,000.00

[표 2]

A 상품 ID	B 상품 ID	지지도(support)	신뢰도(confidence)	향상도(lift)
0	1	0.06	0.74	6.59
1	4	0.02	0.17	2.60
0	4	0.02	0.18	2.85
1	18	0.01	0.12	2.25
1	34	0.01	0.10	3.65
0	18	0.01	0.11	2.11
1	71	0.01	0.09	4.17
0	71	0.01	0.11	5.29
1	5	0.01	0.08	2.40

값이 클수록 연관성이 높은 것으로 해석되는 향상도를 기준으로 내림차순 정렬할 경우 [표 1] 과 같은 결과가 나왔다. 특히 상품 433번과 436번, 상품 199번과 358번, 상품 392번과 393번은 향상도가 무려 10,000으로 매우 높은 양의 상관관계를 가졌다. 따라서 해당 상품이 오프라인 상품이라면, 샴푸와 린스 묶음 상품처럼 433번 상품과 436번 상품을 묶어서 판매할 수 있고 온라인 강의, 영상과 같은 온라인 상품이라면 시리즈물 등으로 판단되어 또한 함께 묶어서 판매하거나 A 상품 구매 후 일정 기간 내 B 상품 구매 시 할인을 제공하는 방식 등을 제안해볼 수 있다.

하지만 향상도를 기준으로 보았을 경우 지지도가 0.00에 가까운 상품들이 대부분이었는데, 이는 우연에 의한 구입이거나 매우 소수의 구입으로 판단되어 지지도, 신뢰도, 향상도 순으로 내림차순 정렬하여 [표 2]와 같은 결과가 나왔다.

전체 13,506 건의 거래 중 함께 구매한 건수가 863건으로 가장 많은 0번 상품과 1번 상품의 지지도가 0.06으로 가장 높게 나왔다. 신뢰도 또한 0.74로 높을 뿐 아니라 향상도가 6.59로 높은 연관관계를 보여준다. 따라서 해당 분석을 통해 상품 0번과 1번의 묶음 상품 판매 전략을 가장 강하게 제안해볼 수 있을 것으로 보인다.

결과 테이블로 추출한 결과를 구글 스프레드 시트에 업로드하고 Redash 대시보드에 연동한 결과

