



장바구니 발사대(C2)

<input checked="" type="radio"/> 팀장	유정하
팀원	박정우 유정하 이윤화 황선희
데이터셋	retail rocket
발표자료	Retail Rocket 이커머스 구매 퍼널 이탈 분석.pptx

▼ 데이터셋 구성 및 파일 설명

Retail Rocket (Recommender System Dataset) 개요

Retail Rocket (Recommender System Dataset)은 실제 이커머스 서비스에서 수집된 사용자 행동 로그 기반 데이터셋으로, 추천 시스템 연구 및 고객 행동 분석을 목적으로 공개되었다. 본 데이터셋은 사용자가 상품을 조회(view)하고, 장바구니에 추가(addtocart)하며, 구매(transaction)에 이르는 전 과정을 이벤트 단위로 기록하고 있어 이커머스 서비스의 구매 퍼널 분석과 행동 기반 예측 분석에 적합하다.

1. events.csv

설명

이 파일은 이커머스 서비스에서 발생한 사용자 행동 로그 데이터로, 사용자가 상품을 조회(view), 장바구니에 추가(addtocart), 실제 구매(transaction)한 모든 이벤트가 시간 순서대로 기록되어 있다.

주요 컬럼

- `visitorid` : 사용자 식별자
- `itemid` : 상품 식별자
- `event` : 사용자 행동 유형 (view / addtocart / transaction)
- `timestamp` : 이벤트 발생 시점 (Unix timestamp)
- `transactionid` : 사용자의 구매 이벤트를 구분하기 위한 거래 식별자

활용 목적

- 구매 퍼널 분석
- 전환율 및 이탈률 계산
- 사용자 행동 패턴 분석
- 이탈(Churn) 예측을 위한 핵심 데이터

2. item_properties_part1.csv

설명

이 파일은 상품의 속성 정보와 변경 이력을 기록한 로그 데이터로, 각 상품이 특정 시점에 어떤 속성(property)과 값(value)을 가졌는지를 나타낸다. 상품 하나당 여러 속성과 시점이 기록되어 있어 데이터 용량이 큰 것이 특징이다.

주요 컬럼

- `itemid` : 상품 식별자
- `property` : 상품 속성 종류 (예: categoryid, brand 등)
- `value` : 속성 값
- `timestamp` : 해당 속성이 기록된 시점

활용 목적

- 상품 메타데이터 확보
- 상품-카테고리 매핑
- 카테고리별 전환율 분석

3. item_properties_part2.csv

설명

`item_properties_part1.csv` 와 동일한 구조의 데이터로, 원본 데이터의 용량이 커 두 개의 파일로 분할되어 제공된다. 의미적인 차이는 없으며, 분석 시 두 파일을 병합하여 하나의 상품 속성 테이블로 사용한다.

활용 목적

- part1과 병합하여 전체 상품 속성 정보 구성
- 상품-카테고리 관계 분석 보완

4. category_tree.csv

설명

이 파일은 상품 카테고리의 계층 구조(hierarchy)를 나타내는 데이터로, 각 카테고리가 어떤 상위 카테고리에 속하는지를 정의한다. 이를 통해 상품을 대·중·소 분류 체계로 분석할 수 있다.

주요 컬럼

- `categoryid` : 카테고리 ID
- `parentid` : 상위 카테고리 ID

활용 목적

- 카테고리 구조 이해
- 상위/하위 카테고리 기준 분석
- 카테고리 레벨별 퍼널 및 전환율 비교

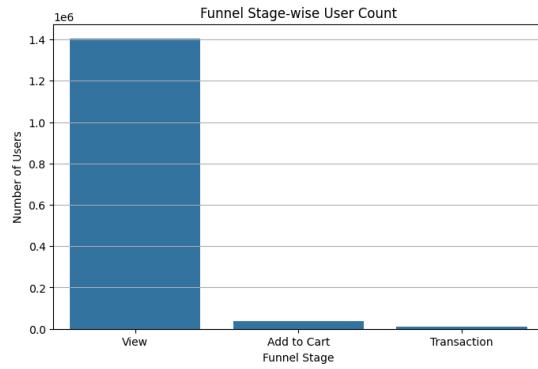
▼ 황선희

https://colab.research.google.com/drive/1h_Sa9ET-64ny8D7EDZoo_aFrS91-T-C?usp=sharing

가설 1

| 대부분의 사용자는 구매 퍼널의 초기 단계에서 이탈할 것이다.

[이벤트 기준]



```
view      2,664,312
addtocart  69,332
transaction 22,457
```

- 상품을 조회(view) 한 행동: 약 266만 건
- 장바구니에 추가(addtocart): 약 6.9만 건
- 실제 구매(transaction): 약 2.2만 건

view → addtocart

```
69,332 / 2,664,312 ≈ 2.6%
```

👉 약 97%가 조회 단계에서 이탈
addtocart → transaction

```
22,457 / 69,332 ≈ 32.4%
```

👉 장바구니까지 온 사람은 상대적으로 잘 산다
퍼널 구조상 View → Cart 전환에서 사용자 급감이 관찰됨

[사용자 수 기준]
(np.int64(1404179), np.int64(37722), np.int64(11719))

- view 경험 사용자 1,404,179명
- addtocart 경험 사용자 37,722명
- transaction 경험 사용자 11,719명

view → cart = 0.0269 = 2.7%
cart → buy = 0.3107 = 31.1%
view → buy = 0.00835 = 0.83%

가설 2

| 사용자 행동 패턴에 따라 구매 전환 가능성은 유의미한 차이를 보일 것이다.

▼ 코드

```
import pandas as pd

# 1. 데이터 로드
events = pd.read_csv("events.csv")

# 2. timestamp 변환
events["timestamp"] = pd.to_datetime(events["timestamp"], unit="ms")

# 3. 사용자 × 이벤트 집계
user_event_cnt = (
    events
    .groupby(["visitorid", "event"])
    .size()
    .unstack(fill_value=0)
    .reset_index()
)

# 4. recency 계산
last_event_time = (
    events
    .groupby("visitorid")["timestamp"]
    .max()
    .reset_index(name="last_event_time")
)

data_end = events["timestamp"].max()

last_event_time["recency_days"] = (
    (data_end - last_event_time["last_event_time"]).dt.days
)

# 5. 활동 기간 계산
activity_span = (
    events
    .groupby("visitorid")["timestamp"]
    .agg(first_time="min", last_time="max")
```

```

        .reset_index()
    )

activity_span["active_days"] = (
    (activity_span["last_time"] - activity_span["first_time"]).dt.days + 1
)

# 6. 병합 → user_df 완성
user_df = (
    user_event_cnt
    .merge(last_event_time[["visitorid", "recency_days"]], on="visitorid")
    .merge(activity_span[["visitorid", "active_days"]], on="visitorid")
)
user_df.head()

# 행동 패턴 세그먼트 정의
user_df["segment"] = "view_only"

user_df.loc[
    (user_df["addtocart"] > 0) & (user_df["transaction"] == 0),
    "segment"
] = "cart_only"

user_df.loc[
    user_df["transaction"] > 0,
    "segment"
] = "buyer"

user_df["segment"].value_counts(normalize=True) * 100

```

[세그먼트 정의]

▼ ✓ 비이탈(0) / 조기 이탈(1) 정의

● 조기 이탈 (early_churn = 1)

| View 행동만 하고, Cart 또는 Transaction 행동 없이 이탈한 사용자

```

view ≥ 1
AND addtocart = 0
AND transaction = 0

```

사이트에 방문은 했지만 상품 탐색을 최소 수준만 하고 구매 퍼널에 진입하지 않고 종료 퍼널 '초기 단계'에서 이탈한 사용자

● ● 비이탈 (early_churn = 0)

| Cart 또는 Transaction 행동을 1회 이상 수행한 사용자

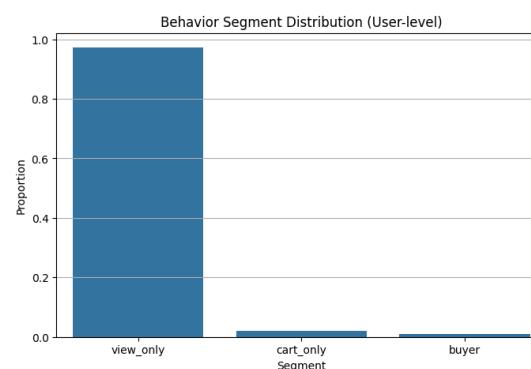
```

addtocart ≥ 1
OR transaction ≥ 1

```

구매 의도가 행동으로 드러난 사용자 실제 구매 여부와 관계없이 퍼널 중·후반까지 도달, 전환 가능성 있는 사용자

세그먼트	조건	의미
View-only	view ≥ 1, cart = 0, buy = 0	조기 이탈
Cart-only	cart ≥ 1, buy = 0	전환 가능
Buyer	buy ≥ 1	전환 성공



segment	proportion
view_only	97.38878
cart_only	1.928558
buyer	0.832564

- 약 97명: 보기만 하고 떠남 (view_only)
- 약 2명: 담아봤지만 안 삼 (cart_only)
- 약 1명도 안 됨: 실제 구매 (buyer)

transaction	addtocart	view
count	11719.000000	11719.000000
mean	1.916290	2.343801
std	8.850529	11.846539
min	1.000000	0.000000
25%	1.000000	1.000000
50%	1.000000	1.000000

transaction	addtocart	view
75%	1.000000	2.000000
max	559.000000	719.000000

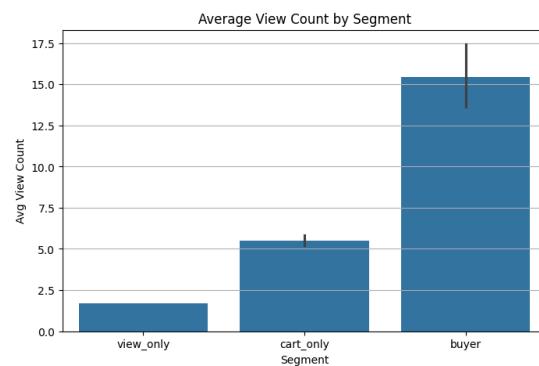
[이탈 vs 비이탈 고객 행동 차이]

- 전체 사용자 중 약 97%가 초기 이탈 고객
- 약 2.8%만이 비이탈 고객(cart 이상 행동)

early_churn = 0 (비이탈)

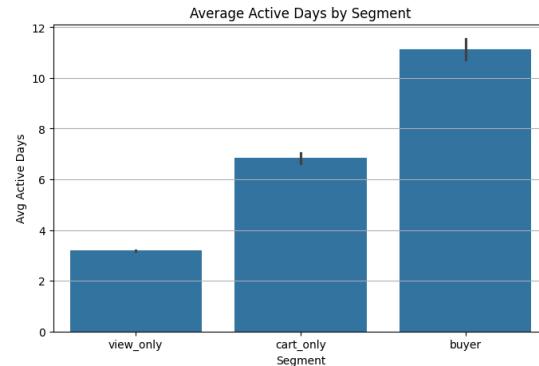
- avg_view 8.49
 - median_view 3
 - avg_recency 65.3
 - avg_active_days 8.13
- early_churn = 1(초기 이탈)
- avg_view 1.71
 - median_view 1
 - avg_recency 68.0
 - avg_active_days 3.18

조회 수(view)



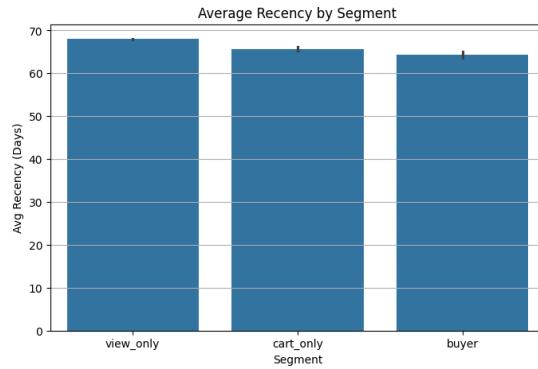
- 비이탈 고객: 평균 8.5회
 - 초기 이탈: 평균 1.7회
- 👉 행동 강도 차이가 5배 이상

활동 기간(active_days)



- 비이탈: 8.1일
 - 초기 이탈: 3.2일
- 👉 비이탈 고객은 여러 날에 걸쳐 탐색
- 👉 이탈 고객은 짧게 보고 바로 떠남

최근성(recency)



- 둘 다 비슷하지만
 - 이탈 고객이 조금 더 오래 전에 마지막 행동
- 👉 “이미 오래전에 떠난 사용자” 특성도 일부 반영

가설 3

| 구매 전환 이전 단계에서 이탈하는 고객은 행동 로그를 통해 조기에 식별할 수 있을 것이다.

▼ 코드

```
# 조기 이탈(Early Churn) 정의

user_df["early_churn"] = (
    (user_df["view"] > 0) &
    (user_df["addtocart"] == 0) &
    (user_df["transaction"] == 0)
).astype(int)

user_df["early_churn"].value_counts(normalize=True) * 100

# 이탈 vs 비이탈 고객 행동 차이
churn_summary = (
    user_df
    .groupby("early_churn")
    .agg(
        users=("visitorid", "count"),
        avg_view=("view", "mean"),
        median_view=("view", "median"),
        avg_recency=("recency_days", "mean"),
        avg_active_days=("active_days", "mean")
    )
)
churn_summary

# 로지스틱 회귀

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report

X = user_df[["view", "recency_days", "active_days"]]
y = user_df["early_churn"]

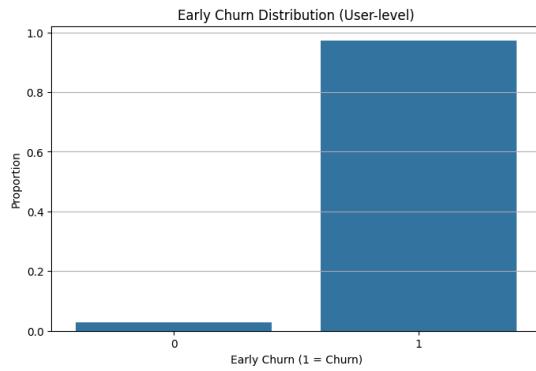
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=42, stratify=y
)

model = LogisticRegression(max_iter=1000)
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

print(classification_report(y_test, y_pred))
```

[조기 이탈 고객 비율]



[예측 모델 결과 해석]

	precision	recall	f1-score	support
0	0.81	0.52	0.63	11660
1	0.65	0.88	0.75	11659
accuracy			0.70	23319
macro avg	0.73	0.70	0.69	23319
weighted avg	0.73	0.70	0.69	23319

early_churn	count
1	38865
0	38865

- 본 모델은 이탈 고객과 비이탈 고객을 1:1로 구성한 데이터셋을 기반으로 학습되었으며, 클래스 불균형의 영향을 제거한 상태에서 행동 기반 변수의 구분 가능성을 평가하였다.
- 전체 정확도(Accuracy)는 0.70으로, 무작위 예측(50%) 대비 의미 있게 높은 수준의 분류 성능을 보였다.

- 이탈 고객(1) 성능 해석
- 재현율(Recall) 0.88 → 실제 이탈 고객의 88%를 올바르게 이탈로 예측하였다.
- 정밀도(Precision) 0.65 → 이탈로 예측된 사용자 중 65%가 실제 이탈 고객이었다.

👉 행동 로그(view, recency, active_days)만으로도 이탈 고객을 비교적 높은 비율로 식별할 수 있음

- 비이탈 고객(0) 성능 해석
- 정밀도(Precision) 0.81 → 비이탈로 예측된 사용자 중 81%가 실제 비이탈 고객이었다.
- 재현율(Recall) 0.52 → 실제 비이탈 고객의 약 절반만 정확히 식별하였다.

👉 비이탈 고객은 이탈 고객 대비 행동 패턴만으로 구분하기가 상대적으로 어려운 집단임을 시사한다.

▼ 이윤화

▼ 26.02.06

판명정하기(장바구니 발사대)/팀장정하기(유정하)/데이터 탐색/가설설정-검증

▼ 전처리

[Untitled33.ipynb](#)

▼ 🎯 문제 정의

어떤 사용자가 실제로 구매할 가능성이 높은지 사전에 파악하기 어렵다.>>많은 사람들이 장바구니에 상품을 담지만, 실제 구매는 매우 적게 이루어진다.

▼ 💡 목표

사용자의 상품 행동 데이터(조회, 장바구니 등)을 기반으로 해당 상품을 구내할 가능성을 예측하는 모델을 개발 하는 것을 목표로 정해서 하면 될거 같습니다.

▼ 26.02.07~26.02.08

▼ events 파일 분석

▼ 이 데이터??
온라인 쇼핑몰 사용자 행동 로그 데이터
(사람들이 언제/무엇을/어떻게 행동했는지? 짹혀 있음)

▼ 데이터 크기

- 총 행(row): 2,756,101개
- 총 컬럼(column): 5개

▼ 컬럼

컬럼명	이게 뭐야?	예시
timestamp	행동한 시간	2015-07-15 18:23
visitorid	사람 ID (익명)	123456
event	무슨 행동을 했는지	view / addtocart / transaction
itemid	상품 ID	461686
transactionid	주문 번호	구매했을 때만 있음

핵심컬럼

행동	개수	비율
view	2,664,312	96.6%
addtocart	69,332	2.5%
transaction	22,457	0.8%

1 조회가 가장 많은 상품 TOP 10 (View 기준)

👉 의미: "사람들이 관심은 많이 갖는 상품"

순위	itemid	조회 수
1	187946	3,410
2	461686	2,539
3	5411	2,325
4	370653	1,854
5	219512	1,740
6	298009	1,642
7	96924	1,633
8	309778	1,565
9	257040	1,531
10	384302	1,528

⭐ 해석 포인트

- 조회 TOP 상품 ≠ 구매 TOP 상품
- 관심 상품과 실제 구매 상품은 다르다 → 퍼닐 이탈의 핵심 힌트

2 구매까지 가장 많이 이어진 상품 TOP 10 (Transaction 기준)

👉 의미: "실제로 돈을 쓰게 만든 상품"

순위	itemid	구매 수
1	461686	133
2	119736	97
3	213834	92
4	312728	46
5	7943	46
6	445351	45
7	48030	41
8	420960	38
9	248455	38
10	17478	37

⭐ 해석 포인트

- 조회 TOP에 있던 상품(187946)은 구매 TOP에 없음
- 반면 461686은 조회도 높고 구매도 높음
→ ⚡ 전환형 상품 vs 탐색형 상품이 존재

3 구매율이 가장 높은 시간대 (시간대별 전환율)

👉 계산 방식

시간대별 구매 수 / 시간대별 조회 수

순위	시간대	구매율
1	17시	1.16%
2	18시	1.09%
3	19시	1.07%
4	16시	1.04%
5	21시	1.03%

📌 해석 포인트

- 오후 4~7시 전환 최고
- 단순 트래픽 많은 시간 ≠ 구매 잘 되는 시간
- → 마케팅 / 알림 / 할인 노출 시간 최적화 가능

4 TOP 10 구매 상품 (or 카테고리) 요약

👉 이 데이터셋 특성상

카테고리는 item_properties와 조인 필요

현재 이벤트 데이터 기준으로는 상품 단위 분석이 가장 정확

>> 구매 상위 상품은 전체 조회 상위 상품과 일부만 겹치며, 이는 사용자 다수가 탐색 목적의 조회 이후 구매로 이어지지 않을 시사한다.

5 왜 97%가 빠질까? (WHY 분석)

📊 실제 수치

- 전체 조회: 2,664,312
- 전체 구매: 22,457
- 👉 이탈률 99.15%

▼ 🧠 원인

원인 1: 구조적으로 "view"가 너무 가볍다

- 클릭 한 번 = view
- 스크롤 / 실수 클릭 / 비교 탐색 포함
- 구매 의도가 없는 행동이 대다수

원인 2: 장바구니(addtocart) 단계 부재

- 퍼널>>view → transaction
- 중간 행동 없음 → 전환율이 낮아 보일 수밖에 없음

원인 3: 탐색형 사용자 비중이 압도적

- 조회 TOP ≠ 구매 TOP
- 많은 사용자가:
 - 가격 비교
 - 상품 구경
 - 다음에 사려고 저장(실제 데이터엔 없음)

원인 4: 시간대 영향

- 구매는 특정 시간대에 집중
- 나머지 시간은 저의도 트래픽

6 위 내용 정리(결론)

전체 조회 대비 구매 전환율은 약 0.85%로 매우 낮게 나타났으며, 이는 이벤트 데이터 특성상 탐색 목적의 조회가 대부분을 차지하기 때문이라 판단이 됩니다. 또한 조회 상위 상품과 구매 상위 상품의 불일치는 사용자 행동이 단순 관심에서 실제 구매로 이어지지 않는 구조적 퍼널 이탈률을 시사합니다.

▼ 박정우

rocketipynb

데이터 특징

- 총 데이터 수: 2,756,101 rows
- 컬럼 수: 5개

컬럼명	설명
timestamp	이벤트 발생 시간 (밀리초 단위)
visitorid	방문자 ID
event	행동 종류
itemid	상품 ID
transactionid	구매 발생 시 거래 ID

이벤트 종류 분포

이벤트 종류	건수	비율
view	2,664,312	약 96.6%
addtocart	69,332	약 2.5%
transaction	22,457	약 0.8%

조회는 엄청 많은데 구매 전환은 매우 낮음

전환율

단계	전환율
View → Cart	2.60%
Cart → Purchase	32.39%
View → Purchase	0.84%

```
#데이터 타입 변환
df['datetime'] = pd.to_datetime(df['timestamp'], unit='ms')
df = df.drop('timestamp', axis=1)
```

```
#transactionid는 필요 없는 컬럼이기 때문에 드랍
df = df.drop('transactionid', axis=1)
```

데이터 기간

```
# 데이터 기간과 분석 기준일을 출력하세요
min_date = df['datetime'].min()
max_date = df['datetime'].max()

print(f"데이터 시작일:{min_date}")
print(f"데이터 종료일:{max_date}")
print(f"데이터 기간:{(max_date - min_date).days}일")
[22] ✓  데이터 시작일:2015-05-03 03:00:04.384800
데이터 종료일:2015-09-18 02:59:47.788800
데이터 기간:137일
```

```
# 많이 조회한 상품
top_view_items = (
    df[df['event'] == "view"]
    .groupby("itemid")
    .size()
    .reset_index(name="view_count")
    .sort_values("view_count", ascending=False)
)

top_view_items.head()
```

itemid	view_count
94618	187946
232277	461686
2717	5411
186535	370653
110421	219512

```
#구매를 많이 한 상품
top_buy_items = (
    df[df['event'] == "transaction"]
    .groupby("itemid")
    .size()
    .reset_index(name="buy_count")
    .sort_values("buy_count", ascending=False)
)

top_buy_items.head()
```

itemid	buy_count
11898	461686
3055	119736
5436	213834
185	7943
8064	312728

```
df['hour'] = df['datetime'].dt.hour
df.head(1)
```

visitorid	event	itemid	transactionid	datetime	hour	
0	257597	view	355908	NaN	2015-06-02 05:02:12.117	5

시간대별 view 수
hour_view = (
df[df["event"] == "view"]
.groupby("hour")
.size()
.reset_index(name="view_cnt")
)
hour_view.head()
...
hour view_cnt
0 0 139454
1 1 136080
2 2 141527
3 3 146751
4 4 143289

#시간대별 transaction 수
hour_buy = (
df[df["event"] == "transaction"]
.groupby("hour")
.size()
.reset_index(name="buy_cnt")
)
hour_buy.head()
...
hour buy_cnt
0 0 1119
1 1 1037
2 2 1012
3 3 836
4 4 747

2026-02-09

문제 상품 리포트

event	addtocart	transaction	view	view_to_cart	cart_to_buy	view_to_buy	issue_type
itemid							
5411	9	0	2325	0.003871	0.000000	0.000000	조회 대비 구매 전환 낮음
15948	0	0	711	0.000000	0.000000	0.000000	조회만 높고 장바구니 없음
17114	4	0	542	0.007380	0.000000	0.000000	조회만 높고 장바구니 없음
29100	0	0	982	0.000000	0.000000	0.000000	조회만 높고 장바구니 없음
82389	0	0	541	0.000000	0.000000	0.000000	조회만 높고 장바구니 없음
84160	6	5	856	0.007009	0.833333	0.005841	조회만 높고 장바구니 없음
91755	0	0	1024	0.000000	0.000000	0.000000	조회 대비 구매 전환 낮음
96924	0	0	1633	0.000000	0.000000	0.000000	조회 대비 구매 전환 낮음
102306	0	0	1013	0.000000	0.000000	0.000000	조회 대비 구매 전환 낮음
109618	0	0	764	0.000000	0.000000	0.000000	조회만 높고 장바구니 없음

1) “조회만 높고 장바구니 없음” 상품이 많다

예:

- 15948 (view 711, cart 0)
- 29100 (view 982, cart 0)
- 82389 (view 541, cart 0)

👉 인사이트

👉 고객이 상품을 관심 있게 보긴 하는데 구매 의사가 안 생김

액션 아이템

(1) 썸네일/가격/옵션 점검

- 장바구니 0인 상품은 UI 문제 가능성 최우선

(2) PDP 개선 실험

- 리뷰 강조
- 혜택 배너 추가
- 배송 정보 upfront 노출

(3) 광고 유입 재점검

- “조회만 높음” 상품은 광고 타겟이 임동할 가능성 큼

2) “조회 대비 구매 전환 낮음” 상품은 ‘기회 손실’이다

예:

- 5411 (view 2325, cart 9, buy 0)
- 96924 (view 1633, cart 0, buy 0)
- 102306 (view 1013, cart 0)

👉 인사이트

👉 노출량이 엄청 큰데 매출이 없음

액션 아이템

(1) 우선순위 Top 개선 타겟

- view가 1000 이상인데 buy 0이면
매출 잠재 손실 매우 큼

(2) 경쟁상품 비교 체크

- 가격/배송/리뷰 수 비교

(3) 프로모션 후보로 선정

- 쿠폰/무료배송/번들 추천 적용

3) 84160 같은 상품은 "장바구니는 잘 담기는데 분류가 애매"

- view 856
- cart 6
- transaction 5
- cart_to_buy 0.83

📌 인사이트

👉 구매 전환은 매우 좋은 상품인데
view_to_cart가 낮아서 문제로 잡힘

액션

- ☒ 더 많은 노출을 줘야 하는 상품
 - 추천 영역 노출
 - 검색 상단 배치
 - 광고 확장

▼ 유저화

전처리

event를 category로 바꿈

object는 문자열을 행마다 들고 있지만 category는 실제 문자열은 한 번만 저장하고 각 행은 정수 코드만 가진다

```
events = pd.read_csv(  
    'events.csv',  
    dtype={  
        'event': 'category'  
    }  
)
```

timestamp를 datetime64로 변환

숫자를 날짜/시간 타입인 datetime64로 변환

```
events['timestamp'] = pd.to_datetime(events['timestamp'], unit='ms')
```

데이터 검증/품질 체크

전체 이벤트 중 거래 ID가 존재하는 비율과 거래 이벤트(transaction)에서 ID가 없는 행이 있는지 확인

```
events['transactionid'].notna().mean(),  
events.loc[events['event']=='transaction','transactionid'].isna().sum()
```

(np.float64(0.008148104877143472), np.int64(0))

전체 이벤트 중 transactionid가 있는 비율은 0.81% 정도이며 transaction 이벤트 중 transactionid가 없는 개수는 0(문제가 되는 transaction 이벤트 없음)이다

유저 기준 퍼널 conversion funnel

본 사람 중에 몇 %가 장바구니 담고 그 중 몇 %가 구매했는지

- u_view: 한 번이라도 view한 유저 수
- u_cart: 한 번이라도 addtocart한 유저 수
- u_buy: 한 번이라도 transaction한 유저 수
- 이벤트 횟수가 아니라 사람 수
- conversion (user-based): 유저 기준 전환율 계산
 - view → addtocart: view한 유저 중 몇 %가 장바구니까지 갔는지
 - addtocart → transaction: 장바구니 담은 사람 중 구매한 비율
 - view → transaction: 처음 본 사람 중 최종 구매까지 간 비율(흔히 말하는 전체 구매 전환율)

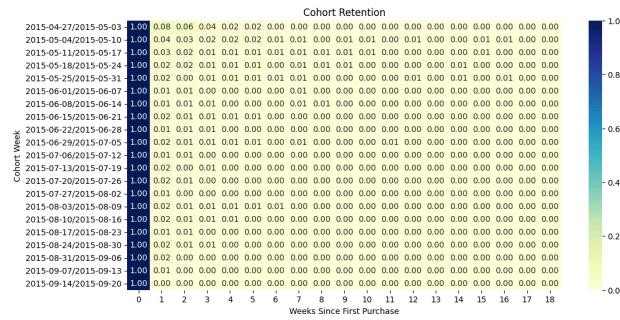
unique visitors (view): 1,404,179
unique visitors (addtocart): 37,722
unique visitors (transaction): 11,719

- view → addtocart: 2.69%
- addtocart → transaction: 31.07%

- view → transaction: 0.83%

구매 기준 코호트 리텐션 cohort retention 분석

첫 구매 주를 기준으로, 유저들이 이후 몇 주 동안 다시 구매하는지 보여주는 주자별 리텐션



첫 구매 이후 재구매율이 매우 낮고, 특히 1주차에 급격히 떨어짐

3~4주차 이후는 거의 0에 수렴 → 장기 리텐션은 거의 없음

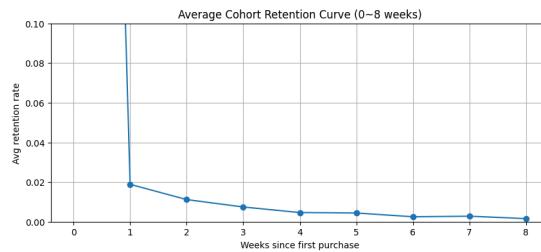
대부분이 1회성 구매에 가깝고 첫 구매 후 1~2주 안에 승부가 갈리는 구조

리텐션 테이블에서 1, 2, 4주차만 뽑아서 평균을 알아봄

week_index	0
W1	0.018844
W2	0.011153
W4	0.004613

dtype: float64

1.88%, 1.12%, 0.46%로 첫 구매 후 재구매율이 주차가 지날수록 급격히 감소



신규 유저의 첫 날 vs 기존 유저

각 유저별 첫 방문일(first_day)을 계산하고, 이벤트가 첫 방문일에 발생했는지 여부(is_new_day)로 나눴다.

이렇게 해서 신규 유저 첫날 이벤트와 기존 유저 이벤트로 분리한 뒤, 각 그룹에서 유저 기반 퍼널(view → cart → buy)과 전환율을 계산

결과적으로 new_day는 각 유저의 첫 방문일에 발생한 이벤트만 포함하며 existing_day는 첫날 이후에 발생한 이벤트를 포함한다.

	users	view_users	cart_users	buy_users	view_to_cart	cart_to_buy	view_to_buy
new_day	1,407,580	1,403,860	32,106	8,959	0.020285	0.247399	0.006038
existing_day	143,769	143,411	7,070	3,114	0.047319	0.380481	0.020821

아래는 퍼센트로 바꾼 것

	view_to_cart	cart_to_buy	view_to_buy
new_day	2.03	24.74	0.60
existing_day	4.73	38.05	2.08

알 수 있는 사실

: 신규 유저 첫날보다 기존 유저가 훨씬 구매 전환이 높다

거래 규모와 구매량 분포

buyer_cnt는 거래에 참여한 고유 사용자 수이고 item_cnt는 거래에 포함된 아이템 수이다.

	buyer_cnt	item_cnt
count	17672.0	17672.000000
mean	1.0	1.270767
std	0.0	0.974793
min	1.0	1.000000
25%	1.0	1.000000
50%	1.0	1.000000
75%	1.0	1.000000
max	1.0	31.000000

대부분의 거래는 한 명이 참여(buyer_cnt = 1)
 평균 거래당 아이템 수는 1.27개
 일부 거래에서 다수 아이템 구매 발생(최대 31개)

RFM 요약

	R	F	M
count	11719.00	11719.00	11719.00
mean	70.85	1.51	1.92
std	39.09	7.34	8.85
min	1.00	1.00	1.00
25%	38.00	1.00	1.00
50%	71.00	1.00	1.00
75%	105.00	1.00	1.00
max	138.00	502.00	559.00

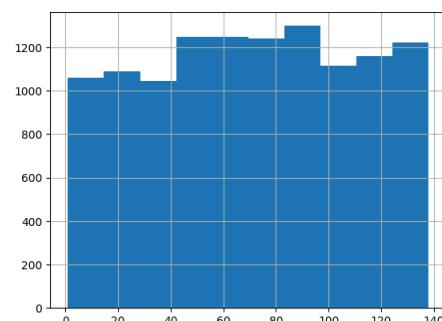
- R (Recency)
 - 평균 70.85일, 최소 1일, 최대 138일
 - 최근 구매까지의 기간이 사람마다 다름
- F (Frequency)
 - 평균 구매 횟수 1.51, 대부분 1회 구매 (50%, 75% = 1)
 - 일부 고객은 매우 많은 구매(F 최대 502) → 소수의 충성 고객
- M (Monetary)
 - 평균 구매 수량/금액 1.92, 대부분 1 (중앙값 1)
 - 일부 거래 규모가 큼 (최대 559)

대부분의 고객은 1회 구매만 하고 소수 고객이 반복 구매와 대량 구매를 통해 매출을 끌어올림
 RFM 요약을 통해 충성 고객과 비활성 고객을 구분하거나 마케팅 타겟팅 가능

RFM

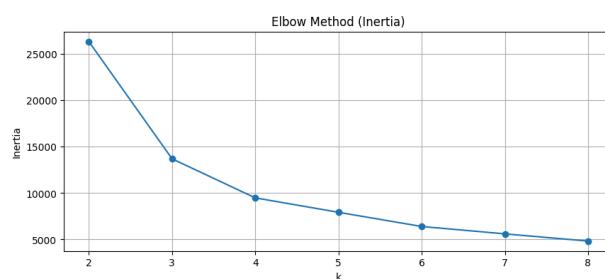
R은 log를 취하지 않은 이유

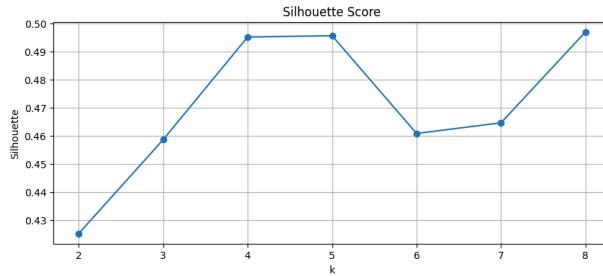
rfm['R'].hist()로 알아본 결과 고르게 분포되어 있어서



k를 4로 잡은 이유

elbow에서 k=4 이후부터는 감소폭이 완만해짐
 silhouette에서 k가 4, 5일 때 거의 최고 수준이고 k=8이 약간 더 높아보이지만 8개 군집은 설명, 운영이 어렵다





RFM 데이터를 기반으로 KMeans 클러스터링 수행

네 개의 클러스터로 나누었다.

F와 M은 값이 크게 치우쳐 있어서 로그 변환을 했다.

각 지표를 표준화(StandardScaler)하여 KMeans 클러스터링을 수행

이렇게 생성된 클러스터별로 사용자 수, R/F/M 평균 및 중간값, 전체 대비 비중(share)을 계산

- Cluster 1 (48.7%, 5,706명)
 - R_mean은 104이다. 마지막 구매가 오래되었다.
 - F=1, M=1 → 대부분 한 번만 구매
 - 휴면/이탈 위험이 큰 1회 구매자인데 전체의 거의 절반을 차지한다
 - 강한 reactivation이 필요하다
 - 재방문 쿠폰/무료배송 같은 강한 인센티브
 - 신상품/재입고 알림 (관심 카테고리 기반이면 더 좋음)
- Cluster 0 (45.1%, 5,286명)
 - R_mean은 36일로 최근 구매가 비교적 최근
 - F=1, M=1 (평균, 중앙값 모두 1) → 대부분 한 번만 구매
 - 신규 1회 구매자이며 활성 가능성 있음
 - 첫 구매 후 7~14일 내 2차 구매 유도(추천, 쿠폰, 재구매 리마인드)가 제일 먹힐 군집
- Cluster 3 (5.5%, 643명)
 - R_mean은 63일이다. 중간 정도로 오래됨.
 - F_mean = 2, M_mean = 6 (중앙값은 F=2, M=4)
 - 반복 구매자
 - 혜택을 세게 주기보단 개인화 추천, 멤버십/포인트, 세트 추천으로 구매 빈도와 객단을 키우기
- Cluster 2 (0.7%, 84명)
 - R_mean은 62일인데 F_mean = 54, M_mean = 67로 엄청 크다
 - 중앙값도 F_med = 33, M_med = 43으로 소수가 속한 집단이지만 진짜 해비 유저 집단으로 볼 수 있다
 - VIP/핵심 매출 기여 고객군
 - 너무 작은 집단이라 평균이 과장될 수도 있지만 중앙값도 높아서 실제 VIP가 맞을 가능성이 크다
 - VIP 전용 혜택, 우선 배송/CS, 신상품 선공개, 구매 기반 리워드로 이탈 방지 최우선 집단으로 여겨야

```
cluster
0    5286
1    5706
2     84
3    643
Name: count, dtype: int64
```

cluster	users	share	R_mean	F_mean	M_mean	R_med	F_med	M_med
1	5706	0.486902	104	1	1	104	1	1
0	5286	0.451062	36	1	1	37	1	1
3	643	0.054868	63	2	6	60	2	4
2	84	0.007168	62	54	67	64	33	43

Churn

구매 이력 있는 유저들을 대상으로 이탈(churn) 라벨을 만들었음

과거에 한 번이라도 구매한 유저 중에서 기준일(ref_date) 이후 60일 동안,

다시 안 사면 churn = 1, 다시 사면 churn = 0

기준일(ref_date)는 데이터의 맨 끝에서 60일을 빼서 만들었다.

```
label_days = 60
max_date = events["timestamp"].max()
ref_date = max_date - pd.Timedelta(days=label_days)
```

결과로

```
churn
1    0.983105
0    0.016892
Name: proportion, dtype: float64
```

이렇게 나왔는데 이는 과거에 구매한 사람 중 거의 전원이 이후 60일 동안 재구매를 안 했다는 뜻이다

학습용 데이터와 테스트용 데이터를 나눴다.

전체 유저의 98.3%가 churn = 1이므로 모델 학습할 때 class_weight="balanced"를 넣어주었다. (소수 클래스를 무시하지 못하게 가중치 보정)

```
train: (5492, 18) test: (1374, 18)
y mean: 0.9831051558403728
```

카테고리 분석

▼ 취합

◉ 프로젝트 목표

- 본 프로젝트는 **Retail Rocket** 이커머스 사용자 행동 로그 데이터를 활용하여 고객의 구매 여정을 구조적으로 분석하고, 구매 퍼널 내 이탈이 발생하는 지점과 그 원인을 데이터 기반으로 파악하는 것을 목표로 한다.
- 이를 위해
- 퍼널 분석을 통해 구매 여정의 병목 구간을 확인하고
 - 행동 로그 기반 이탈(Early Churn) 분석 및 예측으로 초기 식별 가능성을 검증하며
 - 코호트 분석과 RFM-클러스터링을 통해 구매 이후 고객 구조를 이해하고
 - 최종적으로 고객 세그먼트별 전략 수립의 근거를 마련한다.

1 문제 정의

(✖ EDA-문제 인식: 박정우)

Retail Rocket 데이터셋은 약 275만 건의 사용자 이벤트 로그로 구성되어 있으며, 이벤트 유형은 `view`, `addtocart`, `transaction` 세 가지이다.

초기 데이터 탐색 결과

- 전체 이벤트의 약 96.6%가 조회(view)
 - 구매(transaction)는 약 0.8%에 불과하였다.
- 이는 단순히 구매가 적은 문제가 아니라, 구매 여정 전반에서 대규모 이탈이 발생하고 있을 가능성을 시사한다.
- 이에 따라 본 프로젝트는 다음 질문에서 출발하였다.
- 사용자는 구매 퍼널의 어느 단계에서 가장 많이 이탈하는가?
 - 이탈은 무작위 현상인가, 특정 행동 패턴을 가진 사용자 집단의 특성인가?
 - 행동 로그만으로도 이탈을 사전에 식별할 수 있는가?
 - 그렇다면 누구에게, 언제, 무엇을 제안해야 하는가?

2 분석 및 모델링

2-1. 데이터 구조 이해 및 전처리

(✖ 전처리-데이터 검증: 유정하 / 박정우)

- `event` 컬럼을 **category** 타입으로 변환하여 메모리 효율 개선
 - `timestamp` 를 `datetime64` 로 변환
 - `transaction` 이벤트 중 `transactionid` 누락 여부 검증 → 이상 없음
- 데이터 품질에 큰 문제는 없으며, 구매 퍼널 분석 및 사용자 단위 분석이 가능한 상태임을 확인하였다.

2-2. 퍼널 분석 – 이벤트 기준

(✖ 가설 1 검증: 황선희)

이벤트 기준 구매 퍼널 분석 결과:

- View → Add to Cart 전환율: 약 2.6%
- Add to Cart → Transaction 전환율: 약 32.4%
- View → Transaction 전환율: 약 0.83%

특히 **View** 단계에서 약 97%가 이탈하며,

퍼널 내 가장 큰 사용자 감소가 이 구간에서 발생하였다.

👉 이는 초기 탐색 단계(View → Cart)가 핵심 병목 구간임을 의미한다.

2-3. 퍼널 분석 – 유저 기준 재확인

(✖ 유저 기준 퍼널: 황선희, 유정하)

이벤트 수가 아닌 고유 사용자 기준으로 퍼널을 재구성한 결과:

- View 경험 사용자: 1,404,179명
- Add to Cart 경험 사용자: 37,722명
- 구매 경험 사용자: 11,719명

유저 기준 전환율 역시

- View → Cart: 2.69%
- View → Transaction: 0.83%

→ 이벤트 기준과 동일하게

대부분의 사용자가 퍼널 초기 단계에서 이탈함을 재확인하였다.

2-4. 행동 세그먼트 분석

(✖ 가설 2 검증: 황선희)

사용자를 행동 기준으로 다음과 같이 분류하였다.

- **View-only**: 조회만 하고 이탈
- **Cart-only**: 장바구니까지 도달
- **Buyer**: 구매 완료

분석 결과,

- View-only: 97.2%
- Cart-only: 1.9%

- Buyer: 0.8%
- 👉 피널 이탈은 일부 특이 사용자 문제가 아니라, 대다수 사용자에게 공통적으로 나타나는 구조적 현상임을 확인했다.

2-5. 조기 이탈(Early Churn) 정의 및 행동 차이

(📌 가설 3 검증: 활성화)

조기 이탈(Early Churn)을

| View만 수행하고 Cart-Transaction으로 이어지지 않은 사용자로 정의하였다.

이탈/비이탈 사용자 비교 결과:

- 평균 조회 수
 - 비이탈: **8.49회**
 - 조기 이탈: **1.71회**
- 평균 활동 기간
 - 비이탈: **8.13일**
 - 조기 이탈: **3.18일**
- recency 차이는 크지 않음

👉 구매 여부는 “최근 방문”보다 탐색의 깊이와 지속성과 더 밀접한 관련이 있음을 시사한다.

2-6. 이탈 예측 모델

(📌 로지스틱 회귀: 활성화)

`view 수, recency, active_days` 를 변수로

로지스틱 회귀 모델을 학습한 결과:

- Accuracy: **0.70**
- 이탈 고객 Recall: **0.88**

👉 단순한 행동 로그만으로도 조기 이탈 고객을 상당 수준 사전에 식별 가능함을 확인하였다.

2-7. 구매 이후 분석 – 리텐션과 고객 가치

(📌 코호트-RFM 분석: 이윤화)

(1) 코호트 리텐션 분석

- 첫 구매 이후 1주차 재구매율 1.88%
- 2주차 1.12%, 4주차 0.46%
- 장기 리텐션은 거의 0%에 수렴

→ 대부분의 고객은 회성 구매에 그친다.

(2) 신규 vs 기존 유저 비교

- 신규 유저 첫날 View → Buy: **0.60%**
- 기존 유저 View → Buy: **2.08%**

→ 첫 방문에서 전환率이 일어나지 않으면

이후 구매로 이어질 가능성도 매우 낮다.

2-8. RFM 분석 및 K-Means 클러스터링

(📌 고객 세분화: 이윤화)

RFM 지표 기반 K-Means 결과, 4개 군집 도출:

- Cluster 0 (45.1%): 최근 1회 구매자 → 2차 구매 유도 핵심 타겟
- Cluster 1 (48.7%): 오래된 1회 구매자 → 강한 리액티비언 필요
- Cluster 3 (5.5%): 단복 구매자 → 개인화-객단가 상승 전략
- Cluster 2 (0.7%): VIP 고객 → 이탈 방지 최우선 집단

▼ [보충] RFM 지표 분포 및 클러스터링 설계 근거

본 절에서는 RFM 분석 및 K-Means 클러스터링 결과에 대한 지표 분포 특성, 전처리 방식, 군집 수 선택 근거, 그리고 클러스터 해석의 타당성을 설명한다.

1 RFM 지표 요약 및 분포 특성

	R	F	M
count	11,719	11,719	11,719
mean	70.85	1.51	1.92
std	39.09	7.34	8.85
min	1.00	1.00	1.00
25%	38.00	1.00	1.00
50%	71.00	1.00	1.00
75%	105.00	1.00	1.00
max	138.00	502.00	559.00

• R (Recency)

- 평균 70.85일로, 최근 구매 시점은 사용자별로 비교적 고르게 분포

• F(Frequency)

- 중앙값과 75% 분위수가 모두 1로, 대부분의 고객이 1회 구매
- 극소수 고객이 매우 높은 구매 빈도를 보이며 분포가 심하게 치우침

• M (Monetary)

- 대부분의 고객은 소량 구매
- 일부 고객이 매우 큰 거래 규모를 보임

👉 전체적으로 **“대다수는 1회 구매자, 소수의 충성 고객이 매출을 견인하는 구조”**임을 확인할 수 있다.

2 R(Recency)에 로그 변환을 적용하지 않은 이유

R 지표의 분포를 히스토그램으로 확인한 결과,

특정 값에 과도하게 치우치지 않고 비교적 고르게 분포되어 있었다.

- 로그 변환은 주로 극단적인 왜도(skewness)를 완화하기 위한 처리
 - R은 분포 특성상 로그 변환의 필요성이 낮다고 판단
- 👉 따라서 R은 원값을 유지하고,
F와 M만 로그 변환을 적용하는 방식이 데이터 특성에 더 적합하다고 판단하였다.

3 F, M 지표 로그 변환 및 표준화 이유

- F, M 지표는 **소수의 극단값(outlier)**이 존재
 - 로그 변환을 통해 분포의 왜도를 완화하지 않으면
K-Means 군집화 시 거리 계산이 일부 사용자에 과도하게 영향을 받을 수 있음
- 전처리 과정:
- F, M에 로그 변환 적용
 - R, log(F), log(M)을 StandardScaler로 표준화
 - 각 지표가 군집 형성에 동일한 중요도로 반영되도록 조정

4 군집 수(k)를 4로 선택한 근거

- Elbow Method**
 - k=4 이후부터 inertia 감소폭이 원만해짐
 - Silhouette Score**
 - k=4, 5에서 비교적 높은 값
 - k=8이 다소 높게 나타났으나, 해석 및 운영 복잡도가 급격히 증가
- 👉 본 프로젝트의 목적이
***“해석 가능한 고객 세그먼트 도출 및 전략 수립”**임을 고려하여
설명 가능성과 실무 활용성을 기준으로 k=4를 최종 선택하였다.

5 클러스터별 상세 해석 및 타당성 검토

◆ Cluster 1 (48.7%, 5,706명)

- R 평균 104일 → 마지막 구매 시점이 오래됨
- F=1, M=1 → 대부분 1회 구매
- 휴면/이탈 위험이 가장 큰 집단
- 전체의 절반 가까이를 차지

👉 강한 리액티비티션 전략 필요

(재방문 쿠폰, 무료배송, 관심 카테고리 기반 일정)

◆ Cluster 0 (45.1%, 5,286명)

- R 평균 36일 → 비교적 최근 구매
- F=1, M=1 → 신규 1회 구매자
- 향후 활성 가능성성이 가장 높은 집단

👉 첫 구매 후 7~14일 내 2차 구매 유도가 핵심

(추천, 쿠폰, 재구매 리마인드)

◆ Cluster 3 (5.5%, 643명)

- F 평균 2, M 평균 6 (중庸값도 상대적으로 높음)
- 반복 구매 성향이 확인되는 집단

👉 가격 할인보다는

개인화 추천, 멤버십, 세트 제안 등으로 LTV 확대 전략 적합

◆ Cluster 2 (0.7%, 84명)

- F 평균 54, M 평균 67
- 평균뿐 아니라 종양값도 매우 높음
- 소수지만 실질적인 VIP 고객군

👉 집단 크기가 작아 평균 과장 가능성은 있으나,

중앙값 기준으로도 고가치 고객임이 확인됨

👉 이탈 방지 최우선 집단

(VIP 전용 혜택, 우선 배송/CS, 리워드 강화)

✓ 정리

- RFM 지표 분포 특성과 전처리, 군집 수 선택 근거를 종합할 때
본 클러스터링 결과는 데이터 특성과 분석 목적에 부합하는 합리적인 세분화로 판단된다.
또한 각 군집은 명확히 구분되는 행동·가치 특성을 가지며,
후속 CRM 및 마케팅 전략 수립에 직접적으로 활용 가능하다.

3 인사이트 제안 (분석 흐름 기반)

본 프로젝트의 분석 결과를 종합하면, 구매 전환 저하는 결제 단계의 문제가 아니라 구매 여정 초반에서 ‘대부분의 사용자가 피널에 진입하지 못하는 구조’에서 비롯된다. 또한 구매 이후에도 고객 가치가 크게 분화되어 전환·리텐션을 올리려면 세그먼트별로 다른 개인이 필요함을 확인하였다.

3-1. 핵심 병목은 View → Cart: “전환을 막는 구간”이 아니라 “진입이 안 되는 구간”

- 이벤트/유저 기준 모두 View → Cart 전환율이 약 2~3% 수준으로 낮았고,
결과적으로 약 97%가 조회 단계에서 이탈하는 구조가 관찰되었다.
- 반면 Cart까지 도달한 사용자는 Cart → 구매 전환이 약 31%로 상대적으로 높아,
구매 실패의 중심 원인이 “결제 실패 부족”이라기보다 Cart 이전 단계에서의 대규모 이탈임을 시사한다.

✓ 액션 방향(초기 탐색 단계 개입)

- 목표는 “결제 버튼 개선”보다 ‘장바구니까지 가는 사람 수’를 늘리는 것
- 따라서 View 구간에서

- 탐색을 이어가게 만드는 추천/정렬/카테고리 전입 유도
- 구매 의도가 있는 사용자를 빠르게 Cart 행동으로 유도하는 CTA(장바구니 담기)/바로구매 유도
- 같은 '퍼널 진입 장치' 강화가 우선 과제로 도출된다.

3-2. 이탈 사용자는 "저관여·단발 탐색" 패턴으로 식별된다

- 사용자 행동 세그먼트에서 View-only가 약 97%로 대부분을 차지한다.
- 조기 이탈(Early Churn)과 비이탈을 비교하면
 - 비이탈 사용자는 평균 조회 수가 약 8.5회
 - 조기 이탈 사용자는 평균 1.7회
 - 활동 기간도 비이탈이 더 길어 '탐색이 누적되는가'가 전환의 분기점으로 나타났다.
- 즉, "구매 의도"는 초기부터 드러나며, 상당수 사용자는 탐색을 1~2회 수준에서 종료한다.
- ✓ 액션 방향(저관여 사용자에 대한 마찰 최소화 + 탐색 지속 유도)**
- View-only 다수는 "바로 구매"가 아니라 '더 보게 만드는 장치'가 필요
 - 개인화 추천, 인기/유사 상품 묶음, 최근 본 상품 리마인드
 - 탐색 흐름 끊김을 줄이는 리스트/상세 페이지 구성
- 반대로 구매 의도가 빠르게 감지되는 사용자에게는
 - 혜택 노출(배송/쿠폰)이나 "장바구니 담기" 유도 등
- Cart 행동을 앞당기는 자극이 더 효과적일 가능성이 높다.

3-3. 조기 이탈은 예측 가능하며, "선제 개입 타이밍"을 만들 수 있다

- view, recency, active_days 만으로 학습한 로지스틱 회귀에서 이탈 고객 Recall 0.88 수준으로, 단순 행동 지표만으로도 이탈 가능성성이 높은 사용자를 선별할 수 있음을 확인했다.
- 이는 "이탈은 뒤늦게 알게 되는 결과"가 아니라, 초기 행동 로그에서 이미 구분되는 패턴이라는 점을 의미한다.
- ✓ 액션 방향(조기 이탈 타겟팅 / CRM 트리거 설계)**
- 실무적으로는 예측 모델을 "정교한 AI"로 보기보다 초기 행동 기준 트리거(예: 조회 1~2회, 활동 1일 이하 등)를 만들어 즉시 개입 가능한 를 기반 운영으로도 연결 가능
- 개입은 "전환 유도"뿐 아니라 다음 방문을 유도하는 리마인드/관심상품 알림 같은 재유입 장치로 설계될 수 있다.

3-4. 구매 이후 고객 가치는 크게 분화되며, 동일한 CRM은 비효율적이다

- 코호트 분석에서 첫 구매 이후 재구매율이 1주차 1.88% → 2주차 1.12% → 4주차 0.46%로 급감해
- 고객이 첫 구매 이후 빠르게 이탈하는 구조가 확인되었다.
- 신규 유저 첫날 전환(view→buy)이 0.60%, 기존 유저는 2.08%로 첫 경험(첫날/첫구매)이 이후 성과에 큰 영향을 주는 패턴이 나타났다.
- 또한 RFM 기반 클러스터링에서
 - 최근 1회 구매자(활성 가능)
 - 오래된 1회 구매자(휴면 위험)
 - 반복 구매자
 - VIP(극소수 핵심 매출)
- 고객 구조가 분명히 나뉘었다.

✓ 액션 방향(세그먼트별 CRM 차등 운영)

- 최근 1회 구매자(Cluster 0): 첫 구매 후 7~14일 내 2차 구매 유도가 핵심
- 오래된 1회 구매자(Cluster 1): 강한 리액티비에션(쿠폰/무료배송/재입고 알림)
- 반복 구매자(Cluster 3): 혜택보다 개인화-세트 추천-멤버십으로 LTV 확대
- VIP(Cluster 2): 이탈 방지가 최우선(전용 혜택/우선 CS/리워드)

✓ 결론: "누구에게, 언제, 무엇을"의 방향

- 언제: 구매 여정의 시작(초회 직후) + 첫 구매 후 1~2주
 - 누구에게:
 - 탐색이 얇은 조기 이탈 위험군(View-only 중심)
 - 첫 구매자/1회 구매자(활성 전환 가능성)
 - 반복 구매자/VIP(유지·성장 대상)
 - 무엇:
 - 초기에는 "탐색 지속"과 "Cart 진입"을 유도하는 장치
 - 구매 후에는 "2차 구매"를 만드는 CRM 트리거
 - 장기적으로는 고객 가치별 차등 전략
- ▼ 다음 액션 제안

1) 초기 탐색 단계 개입 전략

① 목표

상품을 본 고객이 장바구니까지 가게 만들기

액션 1: 문제상품 PDP(상세페이지) 집중 개선

대상:

- view ≥ 500
- view_to_cart < 1%

실행:

- 썸네일 교체 (첫 클릭 결정)
- 가격/월인정보 상단 배치
- 리뷰/평점 강조
- 배송비 정보 upfront 제공

☞ KPI: view_to_cart +2~3%p 상승

액션 2: "관심상품" 즉시 혜택 제공 실험

예:

- “지금 담으면 무료배송”
- “첫 장바구니 쿠폰”

☞ 탐색 고객은 결제보다 장바구니 진입이 목표

액션 3: 효자상품(구매전환 높은데 노출 적음) 노출 확대

예: cart_to_buy > 0.5 인 상품

- 메인 추천 슬롯 배치
- 검색 상단 고정
- 광고 확장

☞ ROI가 가장 빠르게 나오는 액션

2) 조기 이탈 고객 타겟팅

목표

“이탈 고객 다시 데려오기”

액션 1: View-only 고객 리타겟팅 캠페인

세그먼트:

- view O
- addtocart X
- transaction X

전략:

- 24시간 내 리마인드 광고
- “방금 본 상품 할인중” 메시지

☞ KPI: 재방문율 + 구매전환 상승

액션 2: 이탈 고객 행동 기반 추천 메시지

예:

- 비슷한 상품 추천
- 가격대 낮은 대안 제시

“이 상품이 부담스러웠다면 이런 상품은 어때요?”

액션 3: 조기 이탈 상품군 광고 타겟 재정비

문제상품이 광고로 유입되는 경우

- 타겟 mismatch
- 클릭만 유도하는 키워드 가능

→ 캠페인 구조 수정 필요

3) 구매자 세그먼트별 차별화 CRM 전략

목표

“같은 쿠폰 뿌리지 말고, 단계별로 다르게”

세그먼트 CRM 액션 매핑

고객 유형	조건	CRM 액션
탐색 고객	view만 있음	인기상품 추천 + 첫구매 혜택
장바구니 고객	cart 있음, 구매 없음	무료배송/쿠폰 + 긴급 메시지
구매 고객	transaction 있음	재구매 유도 + VIP 혜택
이탈 고객	7일 이상 미구매	win-back 캠페인

액션 1: 장바구니 이탈 방지 메시지 자동화

- “장바구니 상품 품절 임박”
- “오늘까지 10% 할인”

☞ cart_to_buy 개선 효과 기대

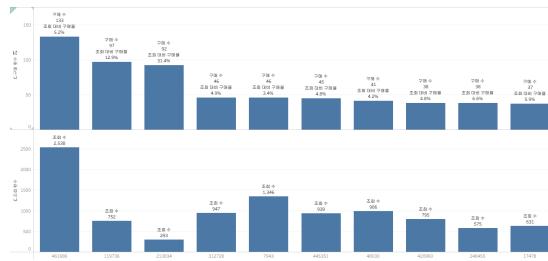
액션 2: 구매 고객 리텐션 프로그램

- 재구매 쿠폰
- 추천인 이벤트
- 리뷰 작성 리워드

☞ LTV 증가 기대

▼ 아이템별

1) 구매 수 대비 조회 수와 구매율



Item 461686은 압도적인 핵심 매출 상품

- View: 2,538
- Purchase: 133
- 👉 구매량도 1위
- 👉 조회량도 1위

📌 의미

이 상품은

- 노출도 잘 되고
- 구매도 잘 되는

완전 매인 헤자 상품

액션

- 추천 영역 고정
- 광고 예산 우선 배치
- 연관상품 묶음 판매(번들링)
- 리뷰/재구매 유도 CRM

조회 대비 구매 효율이 좋은 상품이 존재

- 213834
- View: 293
- Purchase: 92 92/293=31.4%

보는 사람의 3명 중 1명이 구매

📌 의미

이 상품은 "노출만 부족한 숨은 헤자"

액션

- 메인 페이지 노출 확대
- 검색 상단 배치
- 광고 확장하면 ROI 최고

조회는 높은데 구매는 낮은 상품군 존재

예:

- 7943
 - View: 1,346
 - Purchase: 46

전환율:

$$46/1346 = 3.4\%$$

👉 관심은 많은데 구매로 연결 안 됨

📌 의미

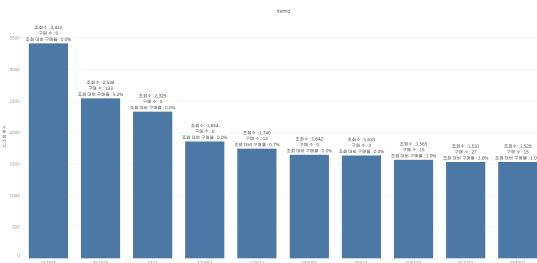
"문제 상품 후보"

- 가격 부담
- 상세페이지 문제
- 경쟁상품 대비 매력 부족

✓ 액션

- PPD 개선 A/B 테스트
- 쿠폰 적용 실험
- 배송비/옵션 UX 점검

2) 조회수 대비 구매 수와 구매율



"상위 조회 상품 대부분이 구매로 전환되지 못하고 있으며, 구매는 소수 상품에 집중되어 있다."

① '조회 낭비' 상품이 명확히 존재

예:

- **187946**
 - 조회수 3,410
 - 구매수 0
 - 구매율 0.0%
- **5411 / 370653 / 96924**
 - 조회수 상위권
 - 구매 천황 0%

👉 노출은 최상위인데 매출 기여는 전혀 없음

✖ 해석

- 탐색 단계(View)에서 구조적으로 막힘
- PDP/가격/옵션/품질/광고 타겟 이슈 가능성 큼

② 구매를 실제로 만들어내는 상품은 극소수

예:

- **461686**
 - 조회수 2,538
 - 구매수 133
 - 구매율 5.2%
- 👉 사실상 매출을 혼자 견인
- ✖ 해석
 - 현재 구매 구조가 건강한 분산형이 아니라
 - 단일 히트 상품 의존 구조

③ '구매율은 낮지만 가능성 있는 상품' 존재

예:

- **257040 (1.8%)**
- **309778 / 384302 (1.0%)**

👉 완전 실패 상품은 아님

👉 조금만 개선해도 매출 증가 가능

Action 1. 조회 낭비 상품 즉시 점검

대상:

- 구매율 0% + 조회수 상위

액션:

- 상세페이지(PDP) 품질 점검
- 가격/배송 경쟁력 비교
- 옵션 선택 불가/품질 여부 확인
- 광고 유입 상품이면 타겟 재설계

✓ Action 2. 핵심 매출 상품 보호 & 확장

대상:

- 461686

액션:

- 메인/추천 영역 고정
- 연관상품 크로스셀
- 재구매 CRM 우선 적용

✓ Action 3. "숨은 효자" 육성

대상:

- 구매율 1% 이상 상품군

액션:

- 노출 확대 A/B 테스트
- 장바구니 쿠폰
- 무료배송 실험

Retail Rocket 이커머스 구매 패널 이탈 분석.pptx

	Recency					Frequency					Monetary				
	count	mean	median	min	max	count	mean	median	min	max	count	mean			
segment2															
2	541	62.517560	61.0	1	138	541	10.970425	3.0	1	502	541	15.63031			
1	3498	117.439680	117.5	89	138	3498	1.064608	1.0	1	2	3498	1.278159			
0	3572	25.153415	25.0	1	53	3572	1.052352	1.0	1	2	3572	1.271837			
3	4108	72.046738	72.0	49	95	4108	1.035540	1.0	1	2	4108	1.213973			

view 100 이상

	segment2	categoryid	view	addtocart	transaction	view_to_cart	cart_to_buy	<u>view_to_buy</u>
1759	2.0	808	105	15	14	0.142857	0.933333	0.133333
1392	2.0	57	111	25	18	0.225225	0.720000	0.162162
1603	2.0	491	150	45	36	0.300000	0.800000	0.240000
1387	2.0	48	113	36	28	0.318584	0.777778	0.247788

카트에 담은 게 50 이상

	itemid	view_users	cart_users	buy_users	view_to_cart	<u>cart_to_buy</u>	view_to_buy
14707	29196	624	96	13	0.153846	0.135417	0.020833
157492	312728	602	114	22	0.189369	0.192982	0.036545
206467	409804	453	119	28	0.262693	0.235294	0.061810
23414	46232	469	56	15	0.119403	0.267857	0.031983
161235	320130	846	81	24	0.095745	0.296296	0.028369
129504	257040	979	67	22	0.068437	0.328358	0.022472
232496	461686	1379	251	120	0.182016	0.478088	0.087020
24339	48030	628	74	37	0.117834	0.500000	0.058917

	itemid	categoryid	parentid	grandparentid
0	29196	1265	1136.0	61.0
1	312728	1098	897.0	61.0
2	409804	1191	61.0	140.0
3	46232	1317	897.0	61.0
4	320130	1483	561.0	395.0
5	257040	683	1606.0	1684.0
6	461686	1037	402.0	143.0
7	48030	1219	121.0	540.0

	grandparentid	items	carts	buys	cart_to_buy_weighted
6	NaN	5	308	179	0.581169
0	61.0	3	266	50	0.187970
2	143.0	1	251	120	0.478088
1	140.0	1	119	28	0.235294
3	395.0	1	81	24	0.296296
4	540.0	1	74	37	0.500000
5	1684.0	1	67	22	0.328358

상위 카테고리 61은 장바구니 유입이 적지 않은데 구매 전환이 낮아 최우선 개선 타겟

61 내부에서 아이템별로 '배송비/재고/정책' 공통 이슈가 있는지 먼저 점검해야