

## DM\_user\_window — 설계노트 (Version v1 scope)

### 1, 목적(Why)

유저 가입일(signup\_date) 기준으로 초기 Activation(14/30일)과 장기 활동량(180일), 주문/매출(14/30/180일)을 한 테이블에 표준화하여 downstream 분석(LTV/Retention/Consistency)의 공통 베이스로 사용한다.

초기 행동 패턴이 장기 성과를 설명하는가 분석에서 Activation/Control 변수를 일관되게 제공한다.

### 2, Grain

1 user\_id = 1 row

여기서 기준일은 signup\_date 잡고, 모든 window 지표는 signup\_date를 기준으로 계산된다.

### 3, Input tables & Join key

- users - 기준 테이블 (signup\_date/유저 속성), key - user\_id
- events - 14/30 퍼널 도달 여부(reach), 180 이벤트량, key - user\_id
- sessions - 180 세션량, 30 promo 세션량, key - user\_id
- orders - 구매여부/주문건수(정합성: purchase 1건 = orders 1건), key - user\_id
- order\_items - revenue(line\_amount 합) 계산용, key - order\_id
- 참고(스캔 최적화): 이 프로젝트에서 적용될 time window 가 (0~180) 일 이니까 sign\_up date 기준 bounds(min\_signup, max\_signup)를 계산해 events/sessions/orders의 전체 스캔을 줄인 후, 최종 집계에서 유저별 window 조건을 다시 적용한다.

### 4, Partition / Clustering

- PARTITION BY : signup\_date 코호트/기간 분석이 핵심("가입 월별" slice)
- CLUSTER BY : user\_id 후속 DM/분석에서 user join이 대부분

### 5, Window 정의

- 기준일: signup\_date
- 전처리 (global filter, 스캔 절감 목적)

- $\text{min\_signup} \leq \text{date} < \text{max\_signup} + 180\text{일}$  범위로 events/sessions/orders를 먼저 필터
- 유저별 window(최종 집계 기준)
  - 0-14d:  $\text{signup\_date} \leq \text{date} < \text{signup\_date} + 14\text{일}$
  - 0-30d:  $\text{signup\_date} \leq \text{date} < \text{signup\_date} + 30\text{일}$
  - 0-180d:  $\text{signup\_date} \leq \text{date} < \text{signup\_date} + 180\text{일}$  - 장기 volume control window
- 경계 규칙: 시작일 포함, 종료일 미포함

## 6, Main Features and 계산 로직

### A. 사용자 속성 (users에서 직접)

- signup\_date, user\_type, device, region, marketing\_source, anomaly\_flag

### B. 퍼널 도달(이벤트 기반 boolean)

- has\_view / has\_click / has\_add\_to\_cart / has\_checkout\_14d/30d
- 로직: 해당 window 내 event\_type이 1건이라도 있으면 TRUE (MAX(IF(...)) 기반)

### C. 구매 도달(주문 기반 boolean)

- has\_purchase\_14d, has\_purchase\_30d
- 로직:  $\text{orders\_14d} > 0, \text{orders\_30d} > 0$

### D. 주문/매출(주문 기반)

- orders\_14d/30d/180d , revenue\_14d/30d/180d
  - Logic : window 내 distinct order\_id count
  - Logic : order\_items(line\_amount)를 order\_id 단위로 합산(주문별 매출) 후, window 조건으로 집계

### E. 장기 활동량(control) - volume controls

- session\_cnt\_180d: 세션 수 (COUNTIF) event\_cnt\_180d: 이벤트 수 (COUNTIF)

## 7. Sanity checks

1. PK 유일성: row 수 = distinct user\_id 수
2. Window 단조성:
  - $orders_{14d} \leq orders_{30d} \leq orders_{180d}$
  - $revenue_{14d} \leq revenue_{30d} \leq revenue_{180d}$
3. 플래그 정합성:
  - $has\_purchase_{14d} == (orders_{14d} > 0)$
  - $has\_purchase_{30d} == (orders_{30d} > 0)$
4. 값 범위:
  - count 지표(orders/session/event/promo)는 0 이상
  - revenue 지표는 0 이상
5. 매출 조인 중복 점검:
  - order\_rev\_filt가 order\_id당 1행인지 확인(중복 발생 시 revenue가 과대계상될 수 있음)
  - 샘플 유저 몇 명을 뽑아 orders\_와 revenue\_의 비정상 급증 여부 확인

## 8. 이 DM이 꼭 필요한지

필요 - 초기 **Activation(14/30일) + 장기 활동량(180일) + 유저 속성**을 한 테이블로 표준화하여 **downstream** 분석에서 반복 조인/중복 정의를 줄이는 베이스 역할을 한다.

Out of scope

F. 프로모 영향(초기 30일) \* Didn't use it in v 1.0