# 실험 분석의 확장

가짜연구소 인과추론팀
온라인 통제 실험 연구자로 거듭나기

2024-06-18

양유승


# 분석 확장의 중요성

Causal-Lab

- "회사가 후기 성숙 단계로 나아감에 따라 데이터 분석 파이프라인을 실험 플랫폼의 일부로 통합하는 것은 해당 방법론을 견고하고, 일관적이며, 과학적이고, 신뢰가 높아지게 만듦
- 데이터 분석 파이프라인은 실험 플랫폼의 핵심!

# 데이터 처리 단계

Causal-Lab

## 데이터 처리: 원시 데이터를 계산 가능한 상태로 처리

- 1. 조인: 여러 데이터 소스를 통합
  - 클라이언트 측 계측(ex. 클릭, 호버, 페이지 로드 시간 등) 측 서버 측 계측(서버 응답 시간, 시스템 오류 등)을 공통 식별자를 사용해 결합
- 2. 정렬: 데이터를 특정 기준에 따라 정렬
  - 사용자 ID와 타임스탬프를 함께 사용해 정렬
- 3. 정제: 이상치 및 누락된 값 처리
  - 실제 사용자가 아닐 가능성이 있는 세션 제거
- 4. 데이터 보강: 유용한 측정 값을 제공하기 위해 일부 데이터를 추출하거나 보강
  - 사용자 에이전트로부터 브라우저 이름이나 버전을 가져오기도 하며, 날짜에서 요일을 추출할 수 있음
  - 또는 세션 중 이벤트 수나 총 세션 지속 시간을 추가할 수도 있음
  - 실험에 따라 특정 세션을 실험 결과 계산에 포함할지 여부를 주석으로 남길 수 있음.

# 데이터 처리 단계

Causal-Lab

## 봇 또는 이상행동의 예시 (ch.3)

### 1. 브라우저 리디렉션

- 리디렉션 메커니즘을 사용하여 A/B 테스트를 구현하는 것은 흔한 방법이지만, 이는 SRM(샘플 비율 불일치)을 일으킬 수 있음
- 봇은 리디렉션을 다르게 처리할 수 있음. 예를 들어, 일부 봇은 http-equiv="REFRESH" 메타 태그를 따라 리디렉션하지 않거나, 새로운 페이지로 태그하여 깊이 크롤링할 수 있음
- 이러한 리디렉션은 사용자들이 북마크하거나 친구들에게 링크를 공유하게 하여 실험의 무작위성을 훼손할 수 있음

### 2. 시간대 효과

- 이메일 캠페인을 A/B 테스트로 설정할 때, 각 변형 그룹에 동일한 수의 사용자가 랜덤하게 할당되었지만, 이메일 오픈율이 SRM을 보임
- 이는 이메일이 작업 시간 동안 대조군에 먼저 보내지고, 그 후에 실험군에 보내졌기 때문입니다. 이로 인해 각 그룹이 이메일을 받는 시간이 달라짐

# 데이터 계산

Causal-Lab

세그먼트 및 지표들을 계산하고, 실험 효과의 추정치(ex. 평균 또는 백분위 수 델타)와 통계적 유의성 정보(p값, 신뢰구간 등)을 포함해 각 실험의 요약 통계를 얻기 위한 결과를 집계

- 데이터 계산을 위한 2가지 유형의 아키텍처
  - 1. 사용자별 통계를 계산해 저장(모든 사용자의 페이지 뷰 수, 노출 수, 클릭 수 계산)한 후에, 이를 사용자를 실험에 매핑하는테이블과 결합
    - 실험뿐만 아니라 전체 비즈니스 보고에 사용자별 통계를 사용할 수 있다는 장점이 있음
  - 2. 사용자별 지표 계산을 실험 분석과 완전히 통합하는 것. 이 때, 사용자별 지표는 별도로 저장되지 않고 필요에 따라 계산됨
    - 실험 데이터 계산 파이프라인과 전체 비즈니스 보고 계산 파이프라인 간의 일관성을 보장하기 위해 추가적인 작업이 필요하다는 단점이 있음
    - 그러나, 실험 당 더 많은 유연성(하드웨어 및 저장 리소스를 절약할 수 있음)을 가질 수 있다는 장점이

# 데이터 계산

Causal-Lab

- 실험이 조직 전체에 걸쳐 확장됨에 따라 속도와 효율성이 중요해짐
- 빙, 링크드인 및 구글은 매일 테라 바이트 단위의 실험 데이트를 처리함. 세그먼트 및 지표의 수가 증가함에 따라 계산에 많은 리소스가 소모될 수 있음
- 더욱이, 실험 스코어 카드의 작성이 지연되면 의사결정을 지연시켜 큰 비용을 발생시킬 수 있으며, 실험이 보다 일반회되고, 혁신 싸이클(innovation cycle)에 필수적일 수록 지연에 대한 영향이 커짐
- 실험 플랫폼 초기에 빙, 구글 및 링크드인은 약 24시간 지연(ex. 월요일 데이터는 수요일 밤에 표시됨)으로 매일 실험 스코어 카드를 생성. 하지만 오늘날은 준실시간(near real-time)으로 처리가 가능해짐
  - 준실시간 처리는 더 간단한 지표와 계산만을 포함하며 심각한 문제(ex. 잘못 구성되거나 버그가 있는 실험)을 발견하는 데에 사용되기도 하며 데이터 처리가 이루어지지 않은 상태의 로그 데이터에 적용되기도 함
  - 이러한 준실시간 처리를 활용해 실험에 문제가 있는 경우, 알림 및 자동 실험 종료를 작동시킬 수도 있음

# 데이터 계산

Causal-Lab

속도와 효율성을 보장하기 위해 아래 사항이 권장됨

- 공통된 지표와 정의를 통해, 모두가 표준 어휘를 공유하고 동일한 데이터 직관을 확립하도록 함. 이를 통해 서로 다른 시스템들에 의해 생성되는 유사한 지표들의 차이를 매번 재조사하는 대신 제품에 관한 흥미로운 질문에 대해 토론할 수 있도록 함
  - Ex. Part_date는 주문완료시점 기준? 결제완료시점 기준?
  - 일관성을 위해 한 가지 구현만을 사용하거나, 실험 또는 지속적인 비교 메커니즘을 사용
- 변화를 미리 관리할 수 있도록 방안을 마련해야 함. 지표, OEC 및 세그먼트는 모두 변화하므로 변경 사항을 파악하고 적용하는 과정이 반복됨
  - 기존 지표의 정의를 변경하는 것이 추가 또는 삭제보다 어려운 경우가 많음
  - 과거 데이터를 다시 계산할 수 있는가? 만약 그렇다면 그것이 얼마나 오래된 데이터까지인가?

# 결과 요약 및 시각화

Causal-Lab

의사 결정자를 위해 주요 지표와 세그먼트를 시각적으로 요약하고 강조해야 함. 이를 위해,

- SRM과 같은 주요 테스트를 강조해서 결과의 신뢰성 여부를 명확히 나타낼 것.
    - Ex. 마이크로소프트의 실험플랫폼은 주요 테스트에 실패하면 스코어 카드를 표시하지 않음
- 흥미로운 세그먼트를 자동으로 강조 표시하는 등의 세그먼트 집중 분석을 통해 의사 결정이 올바른지 확인하고 제대로 작동하지 않는 세그먼트에 대한 제품을 개선할 수 있는 방법이 있는지 확인
- 실험에 트리거 조건이 있는 경우 트리거된 모집단에 대한 효과와 함께 전체 효과를 포함시킬 것

**예시를 통한 이해**

1. **예시 설정:**
    - 한 웹사이트에서 새로운 기능을 테스트하는 실험을 진행합니다.
    - 이 실험은 사용자가 특정 페이지에 도달했을 때(트리거 조건)만 적용됩니다.
2. **트리거 조건 적용 전:**
    - 전체 사용자는 100,000명입니다.
    - 그 중 10,000명이 특정 페이지에 도달하여 트리거 조건을 만족합니다.
3. **실험 결과 분석:**
    - **전체 효과:** 전체 100,000명 사용자에 대한 새로운 기능의 효과를 평가합니다.
        - 새로운 기능이 전체 사용자에게 어떤 영향을 미쳤는지 평가합니다.
    - **트리거된 모집단의 효과:** 트리거 조건을 만족한 10,000명 사용자에 대한 효과를 별도로 평가합니다.
        - 트리거된 모집단에서 새로운 기능이 어떤 영향을 미쳤는지 평가합니다.

# 결과 요약 및 시각화

Causal-Lab

- 건강한 의사 결정 프로세스를 구축하기 위해 플랫폼이 제공할 수 있는 두 가지 선택적 기능
  - 개인이 관심 있는 지표를 구독할 수 있으며 이러한 지표에 영향을 미치는 주요 실험의 요약본을 이메일(슬랙)으로 받아볼 수 있게 하라
  - 실험이 부정적인 효과를 보일 경우, 실험을 확대하기 전 플랫폼은 실험 담당자가 측정 담당자와 논의하는 승인 프로세스를 시작
- 시각화 도구는 제도적 기억(institutional memory)에 접근하기 위한 관문이기도 함
  - 제도적 기억이란 실험과 이를 통해 이루어진 변화의 역사로, 이를 통해 일반화된 패턴을 식별하고 실험 문화를 육성하며 향후 혁신을 개선하는 데에 사용됨
  - 제도적 기억의 구성요소
    - 중앙집중형 실험플랫폼: 모든 변화를 테스트하는 중앙 플랫폼을 통해 데이터 캡쳐 및 조직화 / 실험 소유자, 시작 날짜, 기간 등의 설명 및 스크린샷을 메타 정보로 캡쳐
    - 결과 요약: 각 실험이 다양한 메트릭에 미친 영향

# 결과 요약 및 시각화

Causal-Lab

조직이 실험 성숙도의 유지 및 성장 단계로 이동함에 따라 조직에서 사용하는 지표의 수는 수천 단위로 늘어남. 따라서 아래 기능을 사용.

- 지표를 계층 또는 기능별로 다른 그룹으로 분류
  - Ex. 링크드인은 지표를 1) 회사 전반적 2) 제품별 3) 기능별의 3가지 계층으로 분류 / 마이크로소프트는 지표를 1) 데이터 품질 2) OEC 3) 가드레일 4) 로컬 기능으로 그룹화
  - 시각화 도구는 서로 다른 지표 그룹을 분석할 수 있는 도구
- 다중 테스트 문제를 해결하려면, 표준 값 0.05보다 작은 p값 기준을 사용해 실험자가 가장 중요한 지표를 빠르게 필터링할 수 있도록 해야 함

# 결과 요약 및 시각화

Causal-Lab

## 다중 테스트 문제 (Multiple Testing Problem)

**정의**:
다중 테스트 문제는 여러 통계적 검정을 동시에 수행할 때 발생하는 문제로, 각 검정마다 독립적으로 유의수준을 설정하더라도 전체 실험에서 오류율이 증가하는 현상을 의미합니다.

### 다중 테스트 문제의 예

1. **유의수준 설정**:
   - 일반적으로 각 통계적 검정은 5%의 유의수준(알파 값)을 가지며, 이는 5%의 확률로 귀무가설이 참일 때 잘못 기각될 수 있음을 의미합니다.
2. **여러 검정 수행**:
   - 예를 들어, 20개의 독립적인 테스트를 수행한다고 가정합니다.
   - 각 테스트에서 귀무가설이 참일 때 잘못 기각될 확률은 5%입니다.
   - 20개의 테스트를 동시에 수행할 때, 하나 이상의 테스트에서 잘못된 결과가 나올 확률은 1 - (0.95)^20 ≈ 64%로 크게 증가합니다.

## 다중 테스트 문제의 해결 방법

1. **보정 방법 사용**:
   - 다중 테스트 문제를 해결하기 위해 다양한 보정 방법이 사용됩니다. 대표적인 보정 방법은 다음과 같습니다.
2. **본페로니 보정 (Bonferroni Correction)**:
   - 각 검정의 유의수준을 전체 테스트 수로 나누어 설정합니다.
   - 예를 들어, 20개의 테스트를 수행할 경우, 각 테스트의 유의수준을 0.05/20 = 0.0025로 설정하여 전체 오류율을 5% 이내로 유지합니다.
3. **홀름 보정 (Holm's Method)**:
   - 본페로니 보정의 확장으로, 검정의 p-값을 정렬하여 순차적으로 유의수준을 조정합니다.
   - 가장 작은 p-값부터 시작하여, 각 단계에서 유의수준을 (0.05/(남은 테스트 수))로 설정합니다.
4. **베니아미니-호크버그 보정 (Benjamini-Hochberg Procedure)**:
   - 거짓 발견율(FDR)을 제어하는 방법으로, 발견된 유의한 결과 중 거짓 발견의 비율을 제어합니다.
   - p-값을 정렬한 후, 각 p-값에 대해 (i/m) * Q 값과 비교하여 유의성을 판단합니다 (여기서 i는 현재 p-값의 순위, m은 전체 테스트 수, Q는 허용 가능한 거짓 발견율).

# 결과 요약 및 시각화

Causal-Lab

조직이 실험 성숙도의 유지 및 성장 단계로 이동함에 따라 조직에서 사용하는 지표의 수는 수천 단위로 늘어남. 따라서 아래 기능을 사용.

- 관심 지표를 설정해 예상치 못한 움직임이 발생했을 때 플랫폼이 이러한 지표를 자동으로 식별할 수 있도록 함
    - 식별 프로세스는 지표의 중요성, 통계적 유의성 및 거짓 양성 조정과 같은 여러 요소를 결합해 생성
- 연관 지표에 주목할 것. 예를 들어 CTR이 높은 것은 클릭이 증가했기 때문일지, 전체 페이지수가 감소했기 때문일지 살펴봐야 함.
    - 특히 수익과 같이 분산이 높은 지표는 작은 변화에도 크게 반응할 수 있어, 더욱 민감하고 분산이 낮은 지표(ex. 방문자 수, 구매 전환율 등)을 함께 분석해 더 올바른 의사결정을 할 수 있음