programmers

A/B Test 과정 살펴보기

A/B 테스트 시스템 구성
A/B 테스트 트래픽 나누기
A/B 테스트 분석

Grepp Inc

한기용

keeyonghan@hotmail.com

programmers

Contents

programmers

1장 퀴즈 풀이

1장 내용 숙지를 위해 퀴즈를 같이 리뷰해보자

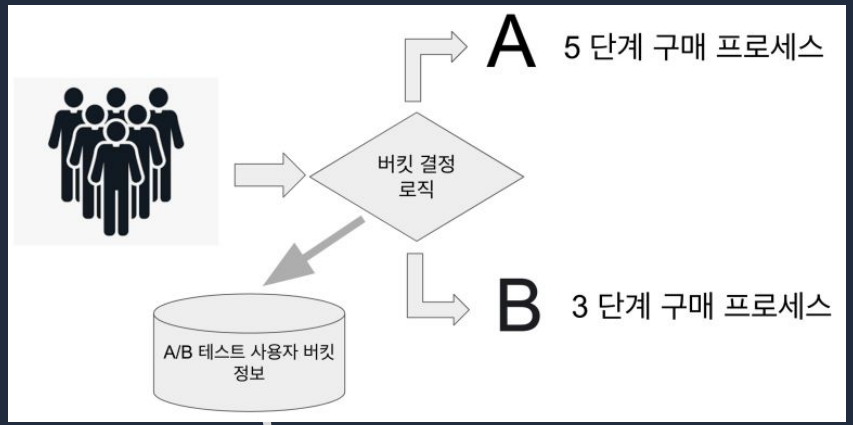
https://forms.gle/73pZdL3cimBLvyDA7

programmers

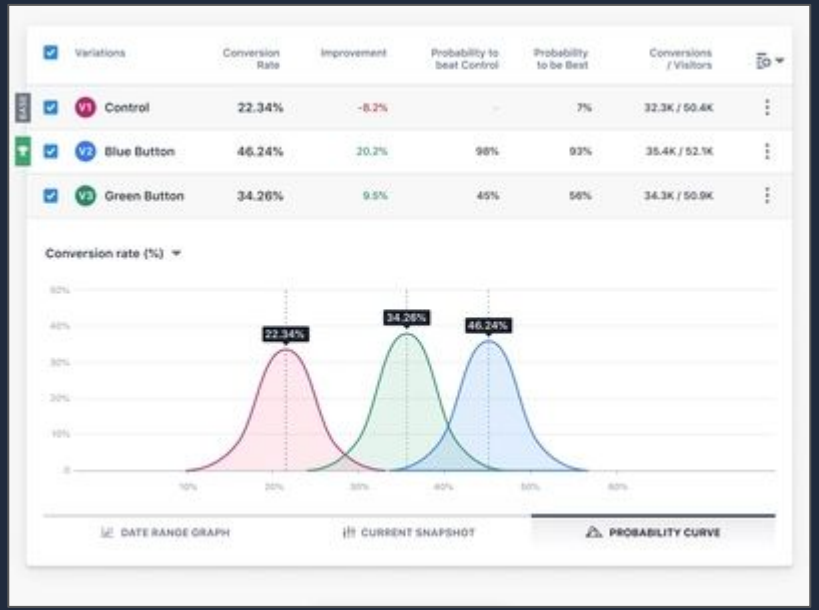
A/B 테스트 시스템 구성

A/B 테스트 시스템은 런타임 시스템과 분석 시스템 두 개로 구성되는데 이에 대해 살펴보자

A/B 테스트 시스템: 런타임 시스템 + 분석 시스템



+



Transform

Event log, revenue data, marketing data, …

구현 방법

- 직접 구현
- SaaS 사용
  - 많이 사용되는 서비스들
    - Optimizely
    - VWO
  - 대체적으로 Front End 관련 테스트를 하는데 유용
    - 보통 Javascript를 웹사이트에 임베딩하는 형태로 작동
    - 내부 데이터와 함께 지표를 계산하려면 연동작업이 더 필요

보통은 SaaS를 쓰다가 직접 구현하는 식으로 고도화됨

programmers

A/B 테스트 과정 전체적으로 살펴보기

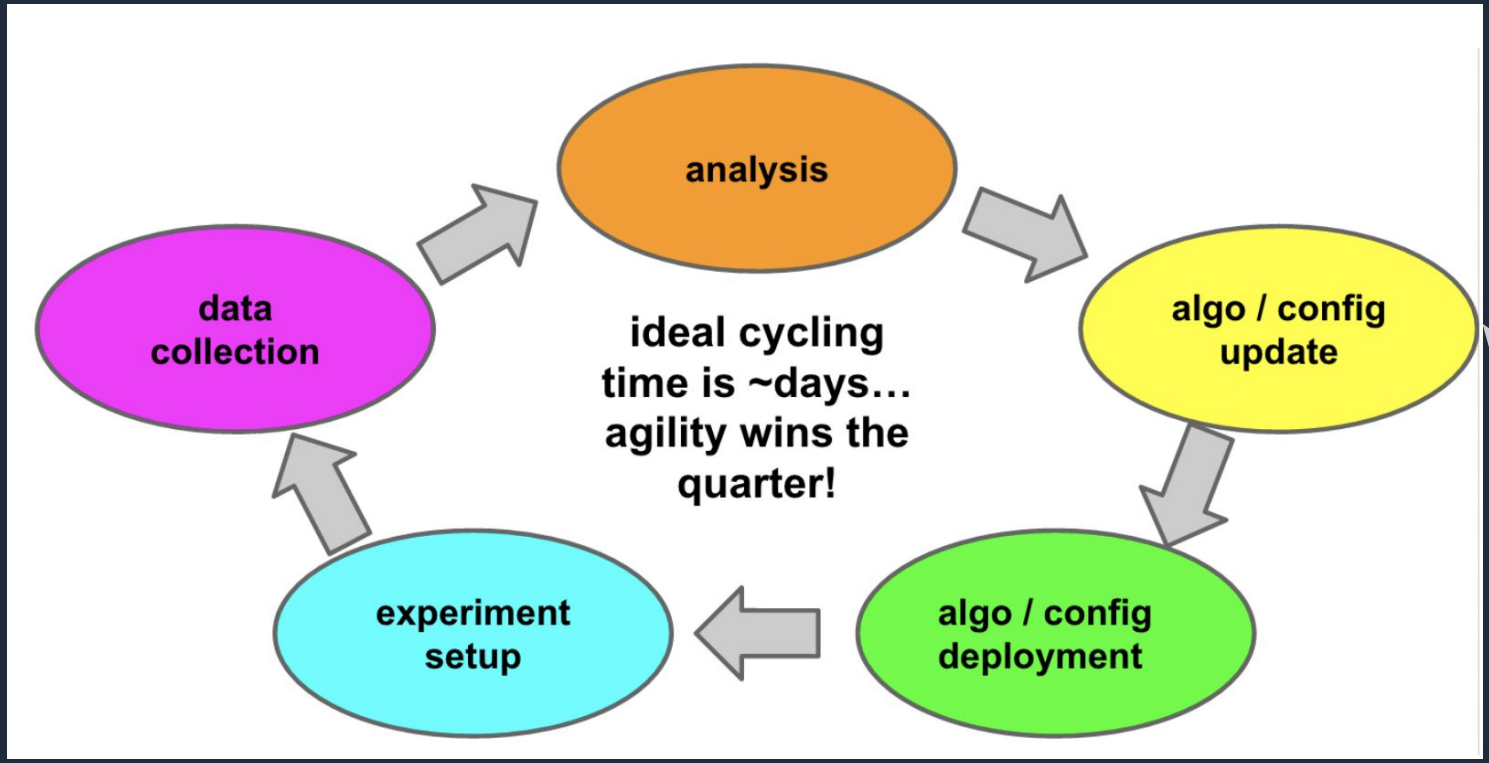
A/B 테스트 프로세스를 다시 한번 정리해보자

전체 과정

- A/B Test Proposal & Approval
  - One pager with hypothesis
  - Discussion with stakeholders (주간 AB 테스트 리뷰 미팅)
- Implementation and QA
- Rollout
- Iterations
  - 주간 리뷰 (주간 AB 테스트 리뷰 미팅)
  - 대시보드가 있으면 금상첨화 (Tableau, Looker, Power BI, 파이썬 노트북, …)
  - 여기서부터 속도가 중요해짐 -> Agile A/B Test
  - 결과가 좋으면 테스트 퍼센트를 증가 (Ramp-up)
    - 1% -> 5% -> 10% -> 25% -> 50% -> 75% -> 100%

A/B 테스트 과정 전체적으로 살펴보기

How Can You Speed Up an Iteration in AB Test?



programmers

유데미 추천엔진: A/B 테스트 과정

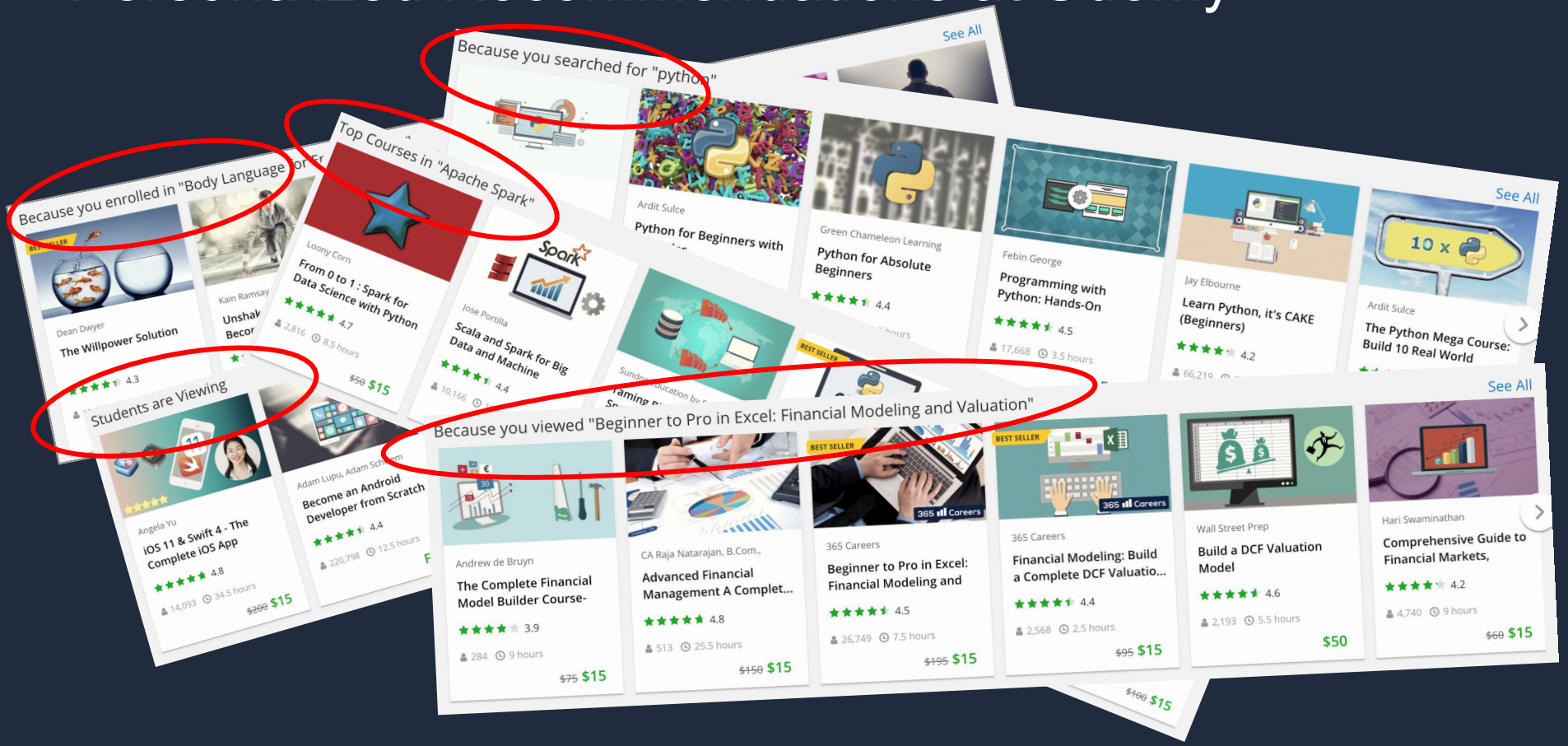
유데미에서 A/B 테스트 시스템을 구축하고 추천엔진 검증에 사용했던 경험을 공유해보자

타임라인: 총 10개월

- 먼저 A/B 테스트 시스템 구축
  - 기존 Optimizely를 걷어내기로 결정
  - 런타임 시스템 구축: 다른 엔지니어링 팀과 협업
  - A/B 테스트 분석 시스템 구축: 데이터 엔지니어들과 분석가들이 협업
- 다음으로 A/A 테스트 수행
  - 검증 용도
- 총 20번의 A/B 테스트를 통해 ML 기반 추천엔진 론치
  - 기존 추천 방식은 마케터 중심의 규칙 기반 추천
  - 12%의 매출 증대 확인. 더 중요한 부분은 모든 것이 자동화되었다는 점

유데미 추천엔진: A/B 테스트 과정

Personalized Recommendations at Udemy



Course Funnel + AB Bucketing Data

(Date, User, Course, impressions, clicks, purchase, consumption, nps)
+
(Date, User, AB Bucket Info)

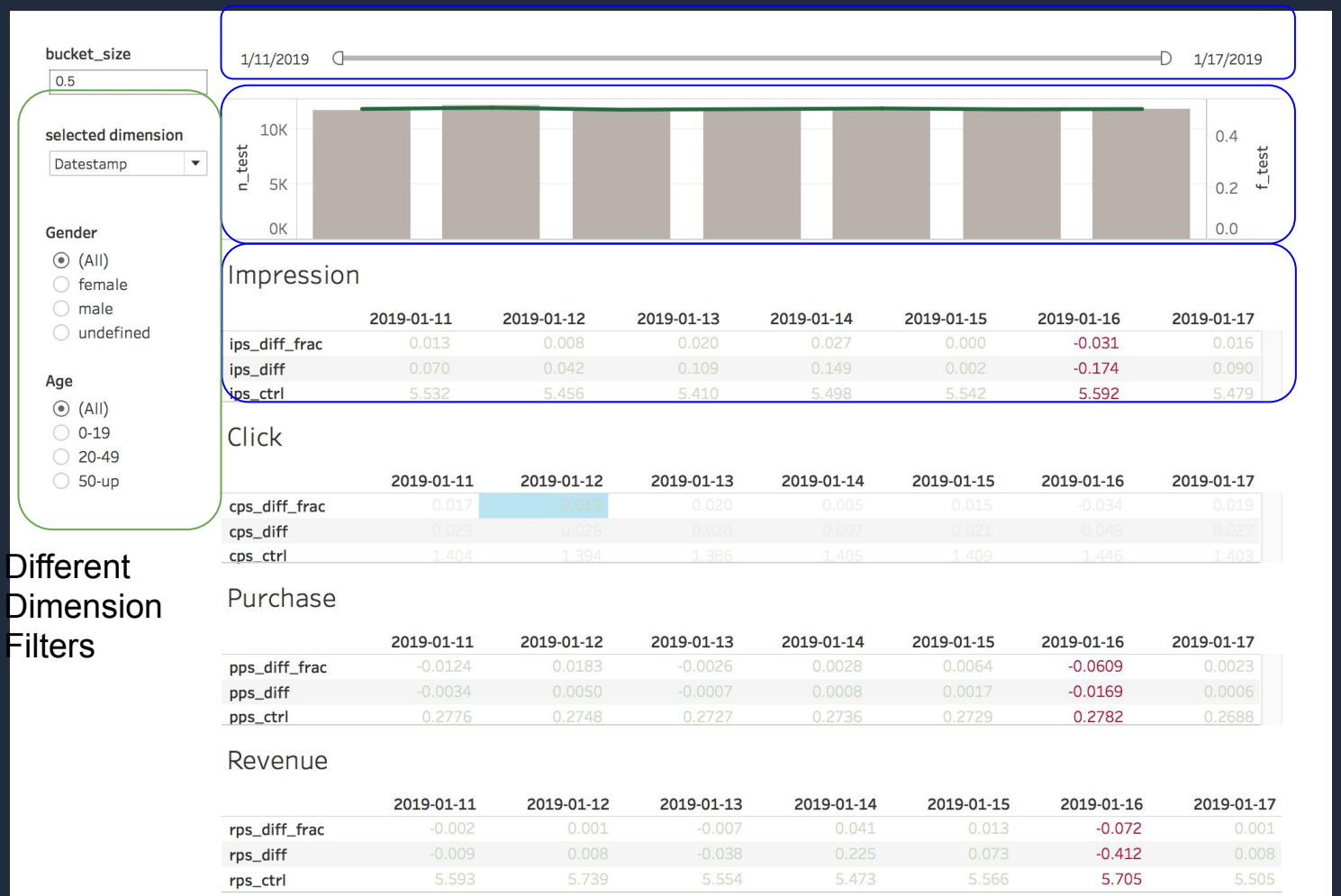
Impressions

Clicks

Enrollment (Revenue)

Consumption and NPS

A	B
10,000	10,500
500	**450**
15	**20**
11	12



Date Range Picker

Traffic Trend

Different Dimension Filters

Sample AB Test Dashboard in Tableau

programmers

Traffic을 A/B로 나누는 방법 이해하기

A/B 테스트 런타임 시스템의 기본은 트래픽을 A와 B로 나누는 것인데 이 과정에 대해 살펴보자

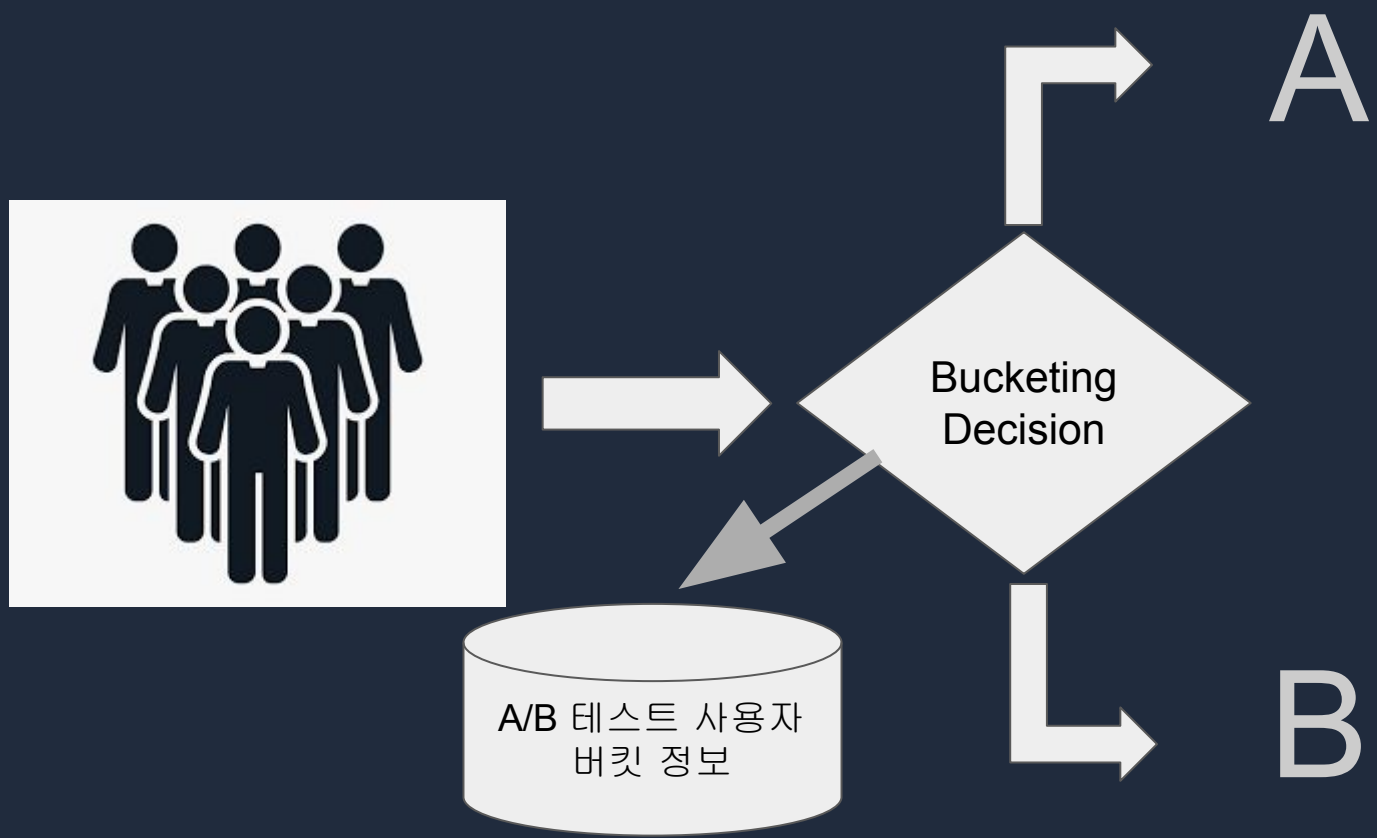
userid vs. deviceid

- A/B 테스트의 성격에 따라 userid를 사용할지 deviceid를 사용할지 결정
  - 로그인한 사용자에게만 하는 테스트인가? 신규 vs. 기존 vs. 모두
  - 모든 방문자에게만 하는 테스트인가?
- userid:
  - 보통 서비스에 사용자 등록이 되는 순간 부여되는 유일한 ID
- deviceid
  - 로그인과 관련없이 서비스 방문자에게 부여되는 ID로 보통 브라우저 쿠키를 이용해서 만들어짐
    - 브라우저 쿠키가 리셋되는 순간 다시 만들어짐
  - 이 ID는 사용자가 아닌 브라우저를 유일하게 지칭해줌
  - 한 userid가 여러 개의 deviceid를 가질 수 있고 한 deviceid에 다수의 userid가 나타날 수도 있다
  - 단순 크롤링/스크래핑을 하는 봇의 경우 쿠키 지원을 안 하기에 이 정보가 없음

크게 두 가지 방법이 존재

- 미리 모든 사용자를 A/B로 나누기
  - 로그인한 사용자를 대상으로 하는 경우 가능
  - 이 경우 다양한 각도에서 bias를 제거 가능
  - 하지만 비로그인 사용자를 대상으로 하는 A/B 테스트라면 이는 불가능
  - 또한 AB 테스트 중에 신규등록된 사용자에게도 적용 불가능
  - 넷플릭스가 사용하는 방법
- 사용자를 동적으로 A/B 테스트 진행 중에 나누기
  - 일반적으로 사용되는 방법
  - 로그인한 사용자이건 아니건 적용가능
  - 하지만 앞의 방법보다는 bias가 생길 가능성이 있음
    - 특히 interaction의 가능성이 있음

주어진 트래픽을 A 혹은 B로 나누기 (1)



주어진 트래픽을 A 혹은 B로 나누기 (2)

트래픽 (사용자ID 혹은 디바이스ID)을 임의의 숫자로 표현

간단한 나머지 (modulo) 연산 수행

나머지 = 0

A

나머지 = 1

B

Redshift 연결 정보

- Host: https://learnde.cduaw970ssvt.ap-northeast-2.redshift.amazonaws.com/
- Port number: 5439
- Database: dev
- ID:PW
  - 오늘 과정과 숙제: 다음 계정을 사용
    - ID:guest
    - Password:Guest1234

예제 데이터: Redshift에 있는 aa_example 테이블 사용

- 실제 구인 사이트에서 가져온 익명화된 로그 데이터
- 일별 구직자 + JD Funnel data

```
CREATE TABLE raw_data.aa_example (
    user_id int,
    date date,
    job_position_id int,
    clicked int,
    checkedout int,
    applied int
);
```

트래픽 나누기 - 간단한 파이썬 코딩으로 실험해보기 (1)

1. User ID 혹은 Device ID를 랜덤한 값으로 변경
    a. 보통 MD5을 사용
2. MD5으로 바뀐 값을 숫자로 변경
3. 2에서 나온 값에 Variant의 수(보통 2개)로 나머지 연산(modulo)을 수행
4. 3의 결과 값이 0이면 A, 1이면 B
5. 위를 함수로 구현

트래픽 나누기 - 간단한 파이썬 코딩으로 실험해보기 (2)

```
import hashlib


def split_userid(id, num_of_variants=2):
    """Given an id and the number of variants, returns a bucket number"""
    h = hashlib.md5(str(id).encode())
    return int(h.hexdigest(), 16) % num_of_variants


print(split_userid(100))   # 1
print(split_userid(101))   # 0
```

트래픽 나누기 - Redshift SQL로 실험해보기

```
SELECT MOD(STRTOL(LEFT(MD5(100),15), 16), 2);
SELECT MOD(STRTOL(LEFT(MD5(101),15), 16), 2);
```

- MD5 -> LEFT -> STRTOL -> MOD
  a. LEFT는 오버플로우를 막기 위해 사용
  b. STRTOL는 십육진수 문자열을 숫자로 바꾸기 위해 사용
  c. MOD는 나머지 계산으로 값을 최종적으로 0과 1로 바꾸기 위해 사용

실습

- (A/B 테스트) 실습

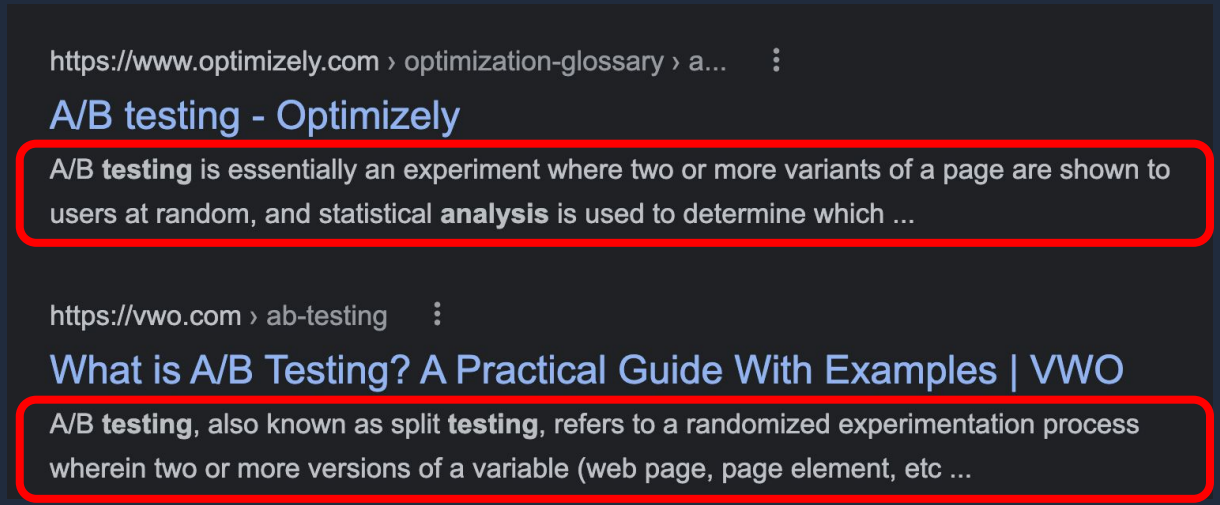
programmers

A/B 테스트 결과 분석이란?

A/B 테스트 분석은 생각보다 어렵다. 그 이유에 대해 알아보자

A/B 테스트 결과 분석은 과학이 아닌 예술

- 경험이 중요함
- 가설을 잘 세워야 배우는 것이 있음
  - 야후 때의 일화: 검색 결과 요약을 바꾸는 AB 테스트 수행
    - 검색결과 요약을 잘 만들면 클릭이 더 많이 생길까?



결과 분석은 객관적이고 공개되어야함

- AB 테스트 실험을 제안한 사람이 분석하는 것은 안 좋음 (wrong incentive)
- 다양한 사람들이 모인 자리에서 결과를 분석하고 여러 의견을 듣는 것이 좋음
- 이를 통해 경험 많은 사람들의 분석 방법을 다른 이들도 배울 수 있음
- 유데미 때의 방식
  - A/B 테스트 주간 미팅. 시니어 데이터 분석가들이 돌아가며 수행 (No라는 말을 잘 할 수 있어야 함)
  - 처음 40분은 기존 테스트들중 결과가 보이는 것들을 리뷰 (p-value가 유의미한 것들)
  - 마지막 20분은 새로운 A/B 테스트 제안들을 리뷰하고 스케줄

Outlier가 A/B 테스트에 미치는 영향

- 어느 서비스나 매출 등에 있어 큰 손들이 존재
  - 이런 큰 손들이 어느 버킷에 들어가느냐가 분석에 큰 영향을 끼침
  - 이런 사람들의 특징은 그냥 무조건 구매를 한다는 점 :)
  - 유데미에서는 이런 사용자들을 Whale user라고 부르고 분석에서 제외했음
  - A/B 테스트가 아니더라도 큰 손들의 retention/churn rate을 살펴볼 필요가 있음
- 봇 유저(scraper bot)가 한쪽으로 몰리는 경우
  - session/impression등에 큰 영향을 줄 수 있음
  - 야후/유데미 때 아주 큰 문제였음
  - 대략 40%의 트래픽까지 봇 트래픽일 수 있음. 음성 봇들은 정체를 밝히지 않음
- 항상 A/B 테스트 결과는 다양한 관점에서 바라봐야함

잘못된 가설

- Survivorship Bias
  - Subscription 서비스라면 Support call을 하는 사람들의 churn rate (이탈율)은 평균보다 높을까?
  - 가끔은 가설이 덜 중요한 문제를 해결하려는 경우가 있음. 그래서 가설에 대해 많은 질문을 하는 것이 중요
- 정말 중요한 지표를 보고 있는가?
  - 성공실패의 기준으로 더 중요한 지표가 있는지 항상 고민?
  - 무슨 테스트이건 기본으 음 -> 매출액이나 구매율이 대표적

programmers

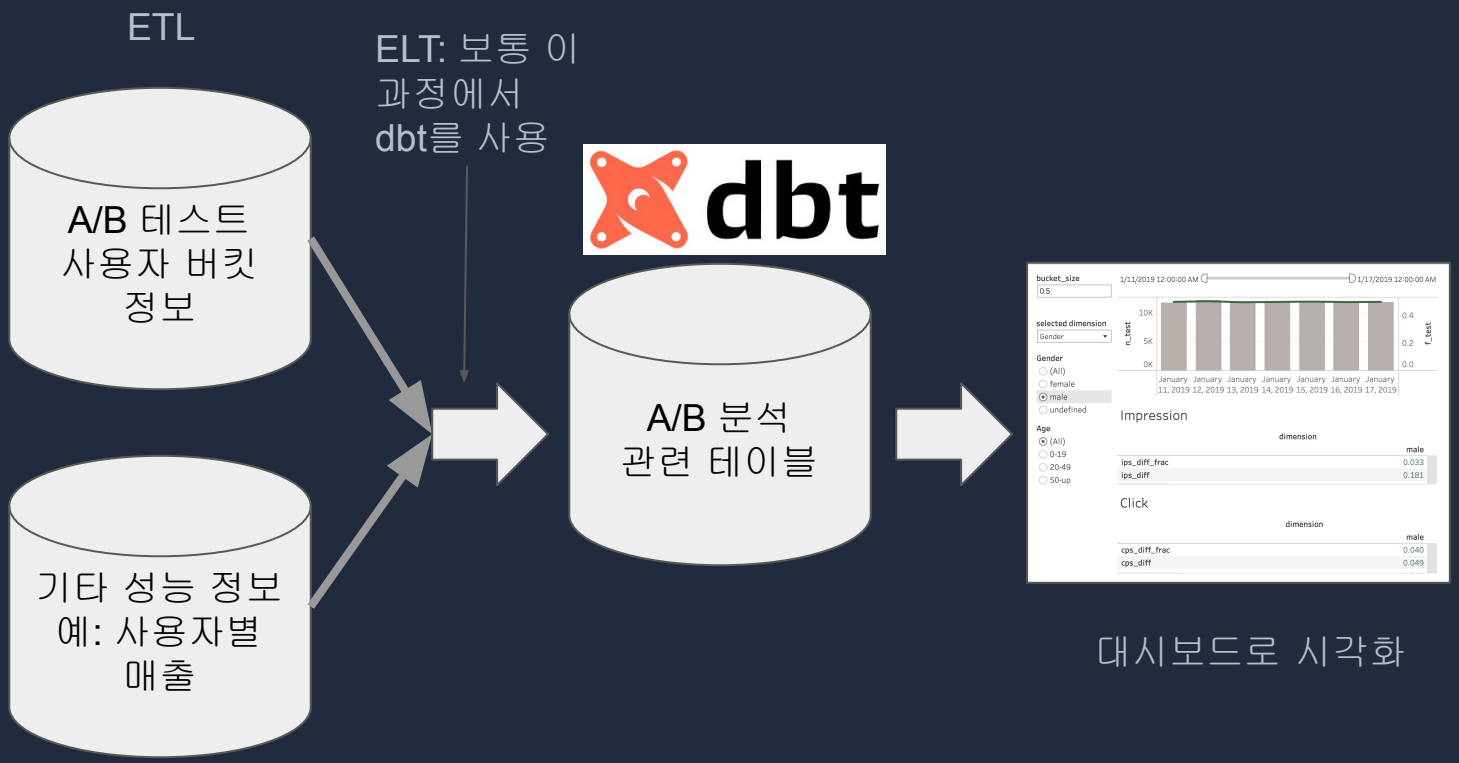
A/B 테스트 분석 시각화

A/B 테스트 분석을 대시보드로 옮겨보자

AB 테스트 분석 시각화 대시보드 요구 조건

- AB 테스트별로 다음 분석이 가능해야한다
  - AB 테스트 전체 기간에 걸쳐 키 지표가 비교 가능해야 한다
  - 일별로 키 지표의 비교가 가능해야 한다
  - 키 지표의 경우 통계적으로 유의미한지 무의미한지 표시가 되어야 한다 (Color coding)
  - 트래픽(사용자) 메타 데이터가 있다면 이를 바탕으로 필터링이 가능해야 한다
    - 성별
    - 나이
    - 지역
    - 신규 사용자 vs. 기존 사용자
    - Acquisition channel

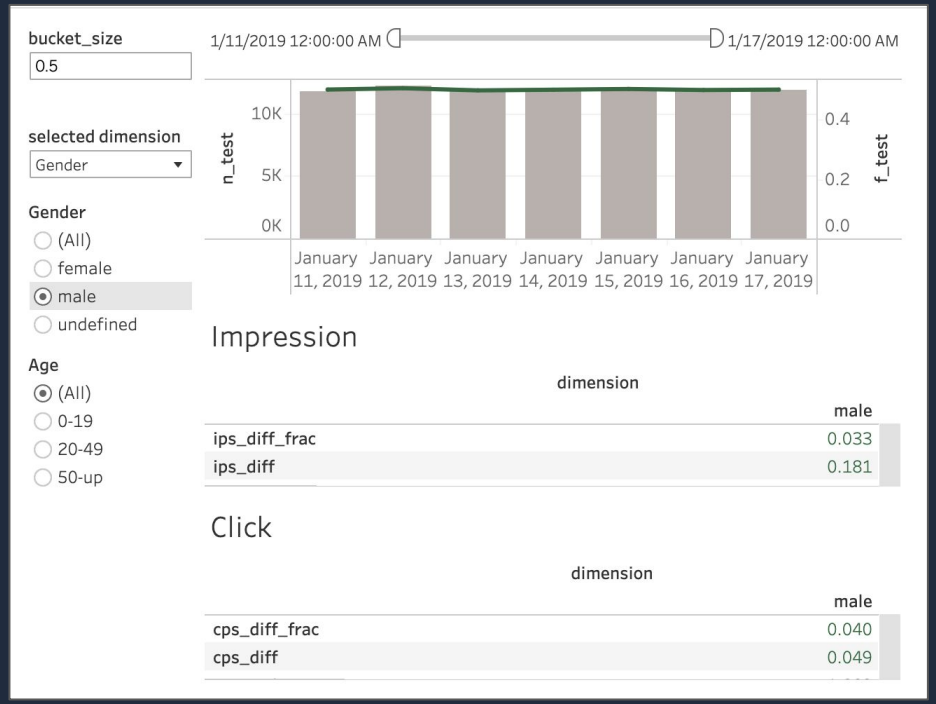
분석 데이터 전처리



대시보드로 시각화

Tableau Public 기반 A/B 테스트 대시보드

- https://public.tableau.com/app/profile/keeyong.han/viz/ABtestdashboard/Dashboard



Q & A