programmers

# A/B 테스트 분석 시각화 (1)

## 대시보드 툴들에 대해서 알아보고 시각화 방법에 대해 살펴보자

Grepp Inc

한기용

keeyonghan@hotmail.com

programmers

# Contents

programmers

# 다양한 시각화 툴 소개

어떤 대시보드들이 있는지 알아보자

# 어떤 툴들이 존재하나?

- Excel, Google Spreadsheet: 사실상 가장 많이 쓰이는 시각화 툴
- Looker (구글)
- Tableau (세일즈포스)
- Power BI (마이크로소프트)
- Apache Superset (오픈소스)
- Mode Analytics, ReDash
- Google Studio, AWS Quicksight
- Python: 데이터 특성 분석(EDA: Exploratory Data Analysis)에 더 적합

# Tableau

- 2002년 미국 캘리포니아 마운틴뷰에서 시작하여 2013년 상장
- 세일즈포스가 2019년 6월에 $15.7B에 인수함
- 특징
  - 다양한 제품군 보유. 일부는 사용이 무료
  - 제대로 배우려면 시간이 꽤 필요하지만 강력한 대시
  - Looker가 뜨기 전까지 오랫동안 마켓 리더로 군림

# 어떤 시각화 툴을 선택할 것인가?

- Looker 혹은 Tableau가 가장 많이 사용되는 추세
  - 두 툴 모두 처음 배우는데 시간이 필요함
  - Tableau의 가격이 더 싸고 투명하며 무료 버전도 존재해서 공부가 가능
- 중요한 포인트는 셀프서비스 대시보드를 만드는 것
  - 안 그러면 매번 사람의 노동이 필요해짐
    - 60-70%의 질문을 셀프서비스 대시보드로 할 수 있다면 대성공
  - 이런 측면에서는 Looker가 더 좋은 선택이지만 가격이 상당히 비쌈

programmers

# 좋은 지표란?

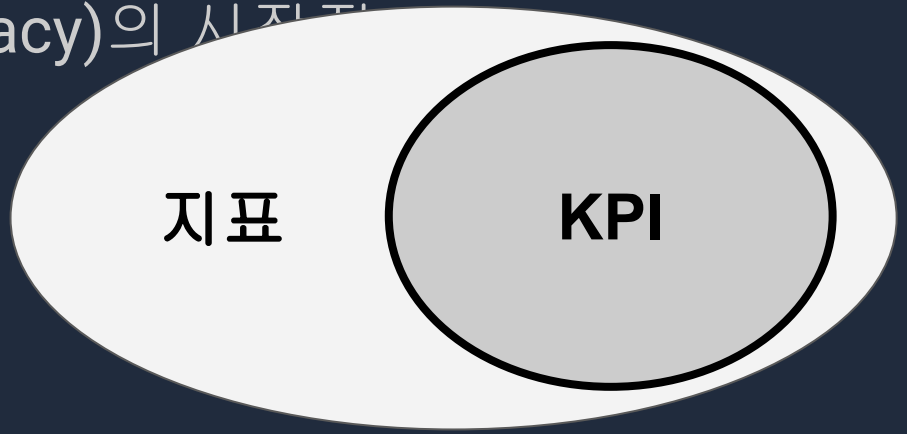
어떤 지표들이 좋은 지표인지 알아보자

# KPI(Key Performance Indicator)란?

- 조직내에서 달성하고자 하는 중요한 목표
  - 보통 정량적인 숫자가 선호됨
  - 예를 들면 매출액 혹은 유료 회원의 수/비율 (정의가 *중요*함)
- KPI의 수는 적을수록 좋음
- 잘 정의된 KPI -> 현재 상황을 알고 더 나은 계획 가능
  - 정량적이기에 시간에 따른 성과를 추적하는 것이 가능
  - OKR(Objectives and Key Results)과 같은 목표 설정 프레임웍의 중요한 포인트

# 지표(Metrics)란?

- 지표와 KPI의 차이점은 중요도
  - KPI는 회사에서 중요한 지표. 즉 지표가 더 큰 개념
- 팀/개인별로 중요한 성과 목표를 정량적으로 갖는 것이 중요
- 데이터 문해력(Data Literacy)의 시작점



# 좋은 지표의 특성

- 3A (Accessible, Actionable, Auditable)
- 쉽게 볼 수 있어야 함 (Accessible)
  - 지표를 보는 것이 쉬어야함 -> 시각화툴이 바로 여기서 도움이 됨
- 실행가능한 통찰력이 제공되어야 함 (Actionable)
  - 지표 등락의 의미가 분명해야함
- 감사가 가능해야 함 (Auditable)
  - 지표 계산이 제대로 되었는지 검증이 가능해야함
  - 데이터 기반이어야 가능

# Next Dashboard Fallacy

- 기존 지표 기반 결정을 못하고 대시보드를 계속해서 만드는 현상
  - 의사결정 장애의 일종 :)
- 지표의 수는 적을수록 좋고 따라서 대시보드의 수도 적을수록 좋음
- 비슷한 것으로 Next Feature Fallacy가 있음

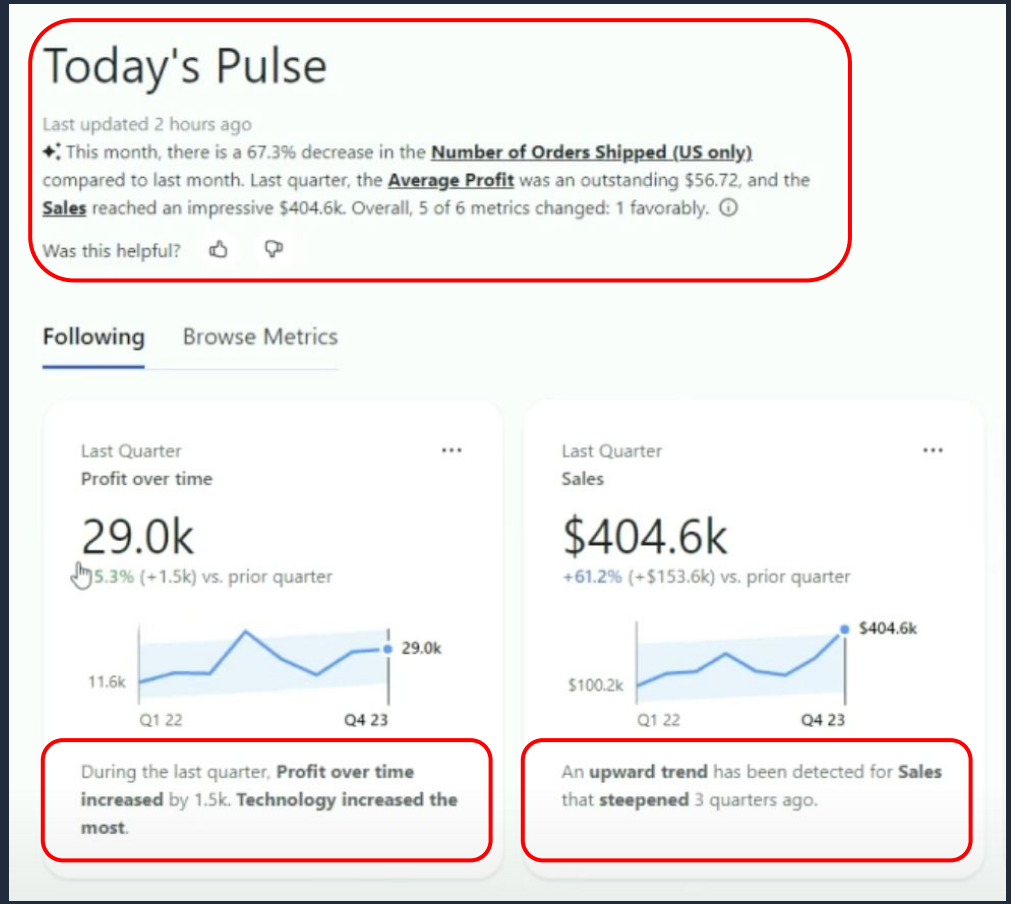
programmers

# 용어 설명

대시보드(Tableau)에서 많이 사용되는 용어들을 살펴보자

# Tableau 제품군

- Tableau Desktop
- Tableau Server
- Tableau Cloud
- Tableau Public
- Tableau Mobile
- Tableau Prep
- Tableau AI and Tableau Pulse

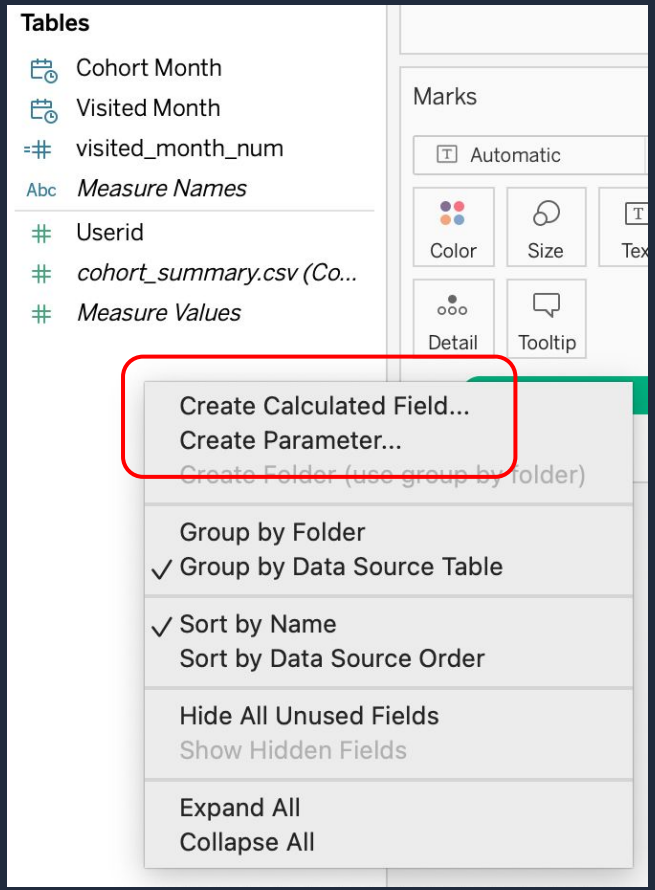


# 비슷한 용어 설명

- Workbook
  - Tableau 프로젝트 파일을 부르는 이름
- Data Source
  - 대시보드를 구성하는데 필요한 원천 데이터
- Worksheet
  - 개별 차트
- Dashboard
  - 개별 차트들의 집합
  - 대시보드라고 부를 수 있음

# Calculated Fields & Parameter

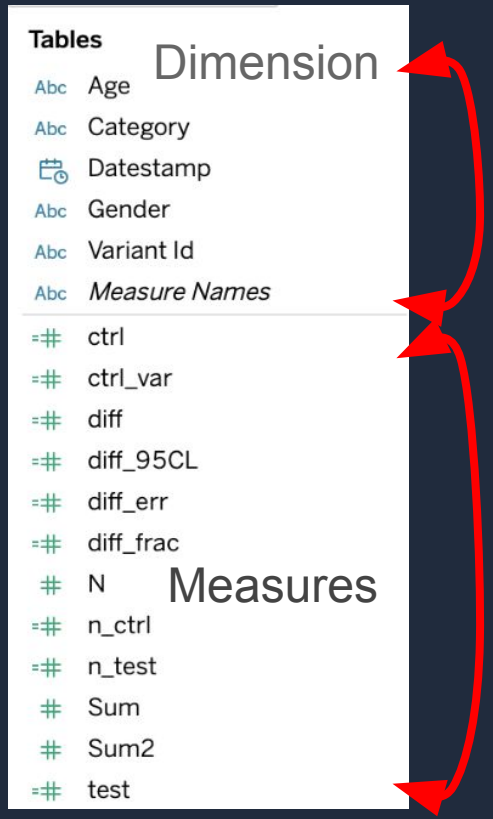
- Calculated Fields
  - 데이터 원본의 기존 필드를 바탕으로 만드는 새 필드
  - 수식과 함수 사용 가능
  - 이를 이용해 Z-score 계산을 하게 됨!!!
- Parameter
  - **동적**인 값을 가질 수 있는 변수
  - Calculated Fields나 필터 등에서 사용 가능

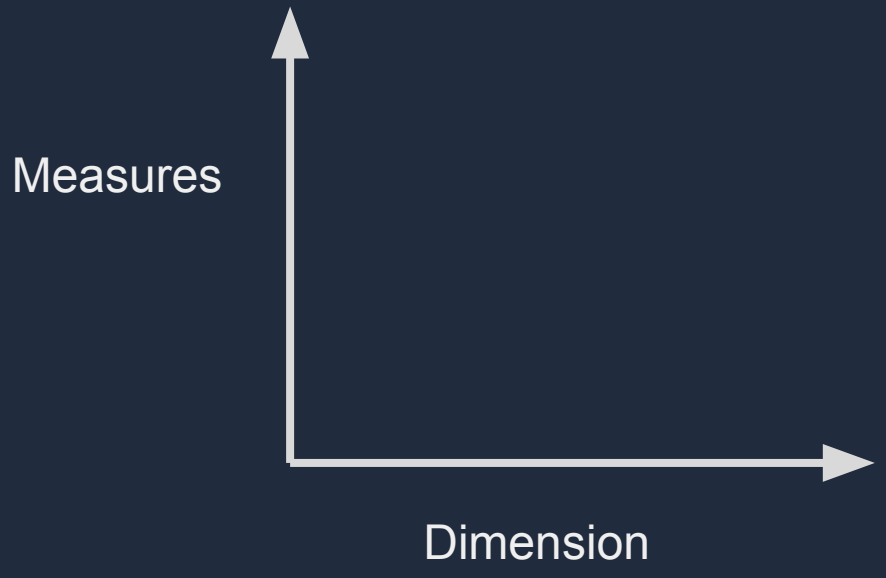


# Measures와 Dimensions의 정의

- Measures:
  - Metrics (Revenue, Purchase, Click, Impression, …)
  - 숫자, 값 (정량적)
- Dimensions:
  - Categorical Breakdown of Measures (Mostly User or Product Properties)
    - Gender, Age, Datestamp, Variant
    - Mobile or Desktop
      - Mobile: iOS vs. Android
    - Browser Type
  - 보통 사용자나 상품에 관한 메타 데이터 (정성적)

# Tableau에서 Measures와 Dimensions



programmers

# OLAP 큐브

미리 계산을 다 해놓음으로써 대시보드 응답속도를
개선해주는 OLAP 큐브에 대해 알아보자

# (다시 보기) AB 테스트 분석 시각화 대시보드 요구 조건

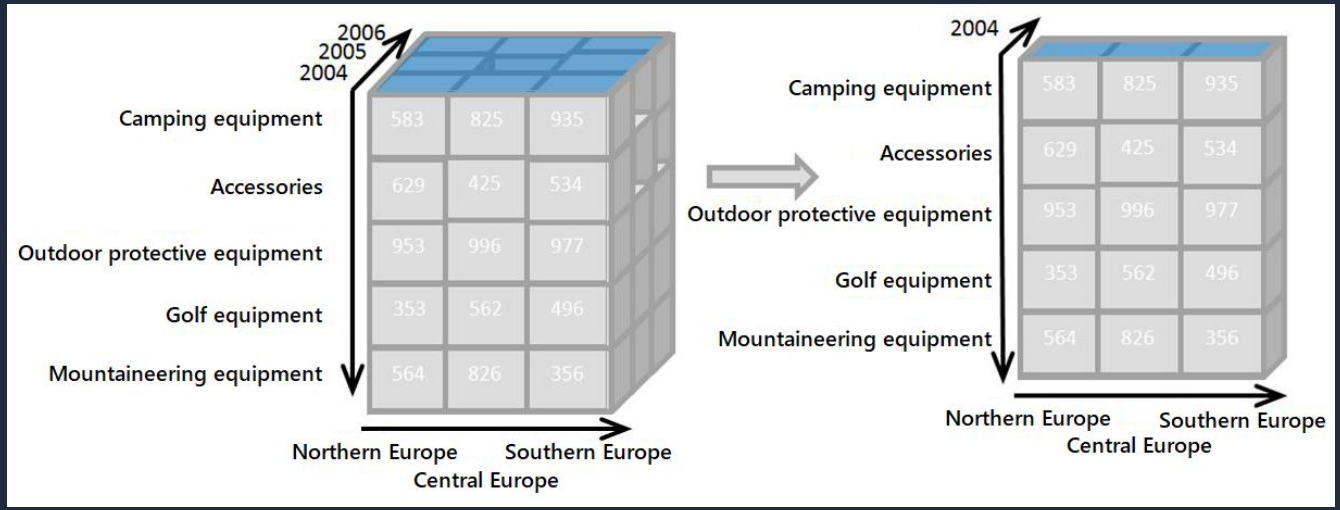
- AB 테스트별로 다음 분석이 가능해야한다
  - AB 테스트 전체 기간에 걸쳐 키 지표가 비교 가능해야 한다
  - 일별로 키 지표의 비교가 가능해야 한다 (trend)
  - 키 지표의 경우 통계적으로 유의미한지 무의미한지 표시가 되어야 한다 (Color coding)
  - 트래픽(사용자) 메타 데이터가 있다면 이를 바탕으로 필터링이 가능해야 한다
    - 성별
    - 나이
    - 지역
    - 신규 사용자 vs. 기존 사용자
    - Acquisition channel
    - 위 정보를 통해 새 기능의 부분적인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

# (다시 보기) 여기서 어려운 점은?

- 선택된 필터에 따라 z-score 계산이 이뤄져야 한다는 점
  - 지표, 날짜, 데모그래픽 조건 (성별과 나이)
- 먼저 선택된 필터에 맞춰 raw data 수집이 이뤄져야함
  - 아니면 모든 가능한 조합에 대해 미리 수집을 해놓고 필터 선택에 따라 지표들을 aggregate
  - 이는 어떤 대시보드를 사용하느냐에 따라 다름

# OLAP 큐브란?

- 미리 모든 조합에 대해 지표 데이터를 수집한 것
  - 그걸 바탕으로 시각화를 수행
  - 장점: 속도가 빠름 (데이터를 매번 읽어올 필요가 없음)
  - 단점: 필터가 변경될 때마다 데이터 수집 방법을 바꾸어야함



# 가상 데이터로 OLAP 큐브 만들어 보기

- 이 시점부터는 session 기반으로 계산
- Variant/Date/Age/Gender별(Dimension)로 아래(Measure)를 계산
  - Session 수, Impression, Click, Purchased, Revenue
- 최종 two sample t-test를 수행하려면 위 measure별로 다음 세 가지를 계산
  - 크기 (n)
  - 값의 합산
  - 값의 제곱의 합산 (이는 나중에 분산 계산을 위함)
  - 이 세 가지를 가지고 (단순화된) two sample t-test 계산이 가능

$$d = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

$$S_1^2 = \frac{\sum_{i=1}^{n} (x_i - x_1)^2}{n_1}$$

$$= \frac{\sum_{i=1}^{n} (x_i^2 - 2x_i x_1 + x_1^2)}{n_1}$$

$$= \frac{\sum_{i=1}^{n} x_i^2}{n_1} - 2x_1 \cdot x_1 + x_1^2$$

$$= \frac{\sum_{i=1}^{n} x_i^2}{n_1} - x_1^2$$

# OLAP 큐브 계산 SQL

이 계산을 모든 조합에 해두고 사용하는 것이 Tableau의 접근방식
이 SQL을 필터가 바뀔 때마다 보내는 것이 다른 대시보드들의 일반적인 방식

```
SELECT datestamp, variant_id, age, gender,
    'impression' category,
    count(1) n,    -- number of sessions
    sum(num_of_items) sum,
sum(num_of_items*num_of_items) sum2
FROM keeyong.analytics_variant_user_daily vds
GROUP BY 1, 2, 3, 4, 5

UNION

SELECT datestamp, variant_id, age, gender,
    'click' category,
    count(1) n,    -- number of sessions
    sum(num_of_clicks) sum,
sum(num_of_clicks*num_of_clicks) sum2
FROM keeyong.analytics_variant_user_daily vds
GROUP BY 1, 2, 3, 4, 5

UNION

SELECT datestamp, variant_id, age, gender,
    'purchase' category,
    count(1) n,    -- number of sessions
    sum(num_of_purchases) sum,
sum(num_of_purchases*num_of_purchases) sum2
FROM keeyong.analytics_variant_user_daily vds
GROUP BY 1, 2, 3, 4

UNION

SELECT datestamp, variant_id, age, gender,
    'revenue' category,
    count(1) n,    -- number of sessions
    sum(revenue) sum, sum(revenue*revenue) sum2
FROM keeyong.analytics_variant_user_daily vds
GROUP BY 1, 2, 3, 4;
```

# sessions_hypercube.csv

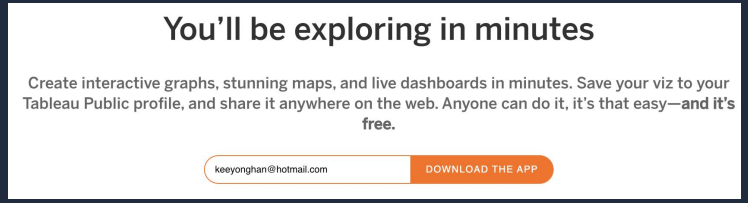
- datestamp (그룹키 1): 2019-01-11 to 2019-01-17
- variant (그룹키 2): “test” or “control”
- age (그룹키 3): “0-19”, “20-49” or “50-up”
- gender (그룹키 4): “male”, “female”, “undefined”
- n: 위의 그룹 조합에 소속된 세션의 수
- category: impression, click, purchase, revenue의 중의 하나가 됨
- sum, sum2: 위 조합의 세션에서 발생한 행동의 총합과 제곱의 총합
    - 예를 들어 category가 impression이라면 sum은 해당 세션들에 속한 impression들의 총합이 되고 sum2는 impression 제곱의 총합이 됨

programmers

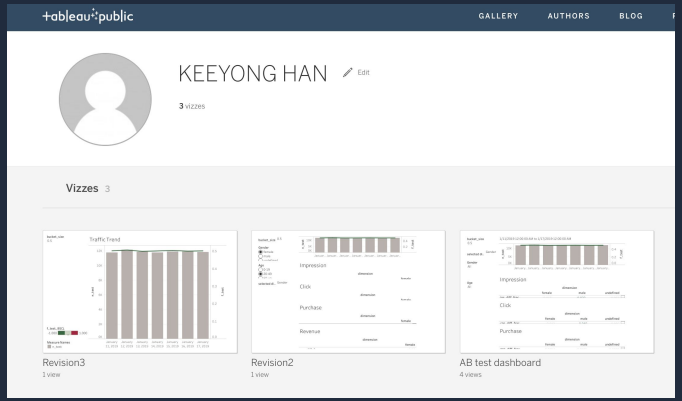
# Tableau Public 다운로드하기

다른 상용제품에 비해 Tableau는 무료로 사용할 방법을 제공한다

# 설치




- Visit https://public.tableau.com/en-us/s/
- Create your account
  - Click “Sign-in”
  - https://public.tableau.com/s/download
- Download the App
  - Compared to Tableau Desktop, it has limited data source support
    - Only local files are supported as data sources

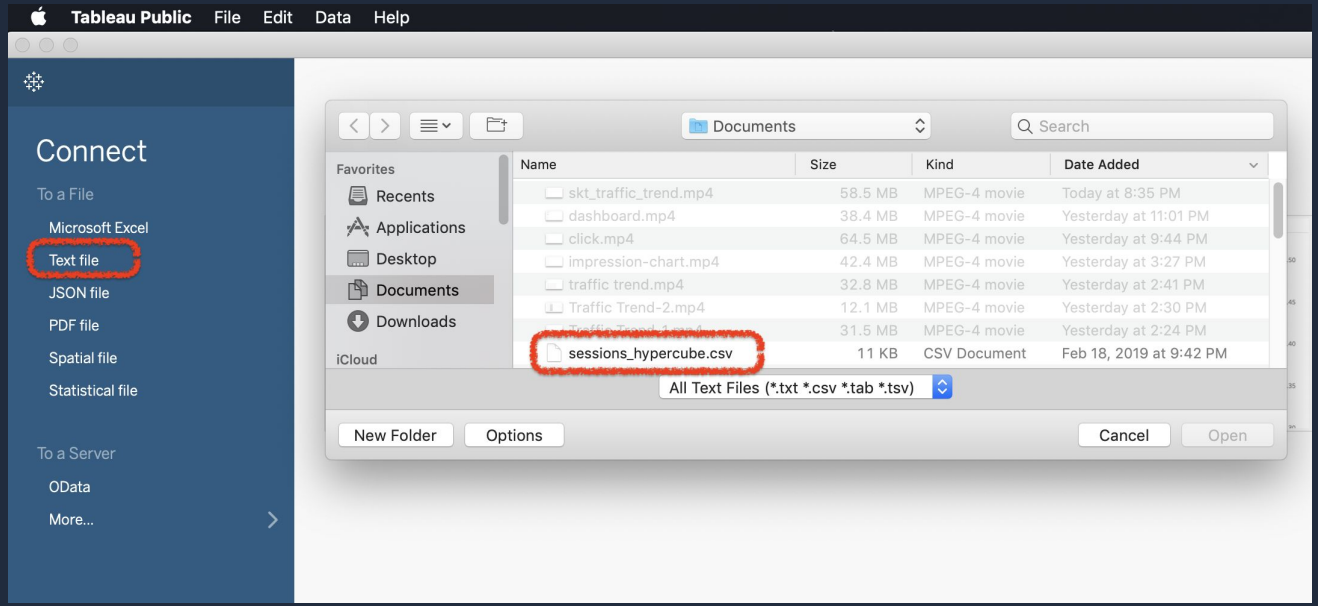


programmers

# Tableau Public으로 CSV 파일 로드하기

# Loading the hypercube CSV file

- Tableau Public은 로컬 파일 밖에 지원하지 않음
- 앞서 SQL의 내용이 있는 sessions_hypercube.csv을 다운로드 받아 저장
- 이를 Tableau Public으로 업로드



programmers

# Z-score 계산과 실습 부분은 다음 슬라이드에서 설명드리겠습니다