제 9회 한국 커뮤니티 데이

KCD 2020

"커뮤니티의, 커뮤니티에 의한, 커뮤니티를 위한
대한민국 최대 규모의 커뮤니티 소통의 장"

주최 커뮤니티

Codeigniter 한국사용자포럼 · MongoDB Korea · OKKY · TAEYO.NET ASP & ASP.NET · K KOREA · k KOREA · C++ · ASP.NET · Azure · ubuntu · {SLiPP} · DATA BREAK · OSS개발자포럼 · 자바카페. · 한국 Azure 사용자그룹 · 30 · JBUG Korea · KR · TS · DevOps · BACKEND. · 모닝클래스 · WOMEN WHO CODE SEOUL · fast.ai · 한국리눅스커널 개발자모임 · TIZEN · IT Infra Engineer · tidyverse

| 제 9회 한국 커뮤니티 데이 온라인 | 2020년 11월 7일(토) |

nipa 정보통신산업진흥원 National IT Industry Promotion Agency · Microsoft · 미노베이션아카데미 · elastic · LINE · MySQL · CUBRID · eBrain · boneis CONVERGENCE · Festa! · 한빛미디어 Hanbit Media, Inc. · Jpub 제이펍 · 志&禪 지앤선 · 길벗

Korea 제 9회 한국 커뮤니티 데이
Community Day 2020

Most Common Deadly Mistakes in Data Analysis

데이터뽀개기 이상열

DATA BREAK

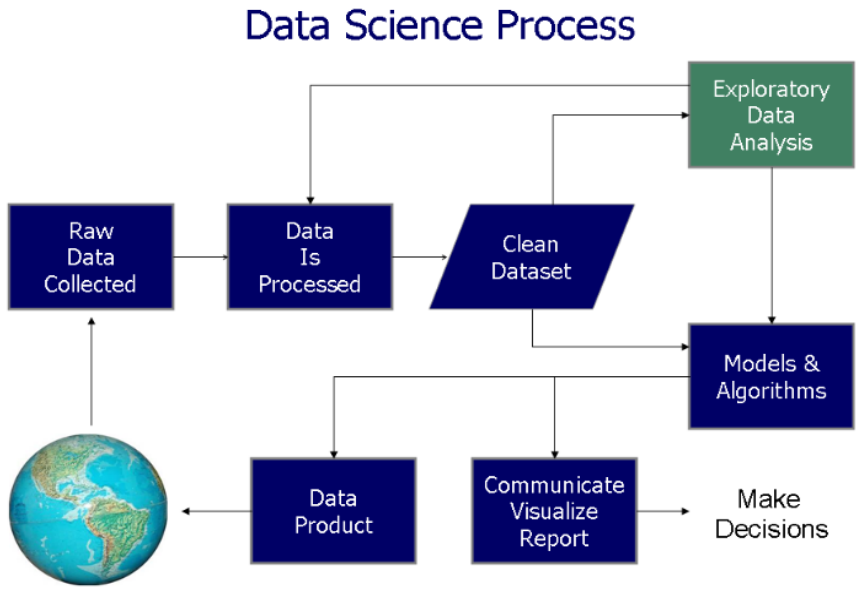
목차

About

게임과 이커머스 도메인 커리어를 거치면서 데이터 추출/리포트부터 머신러닝 개발까지 데이터 분석가가 할 수 있는 다양한 업무를 즐기면서 하고 있습니다.

이상열 syleeie@gmail.com / https://github.com/syleeie2310

데이터 분석



"The goal of a model is to provide a simple low-dimensional summary of a dataset"

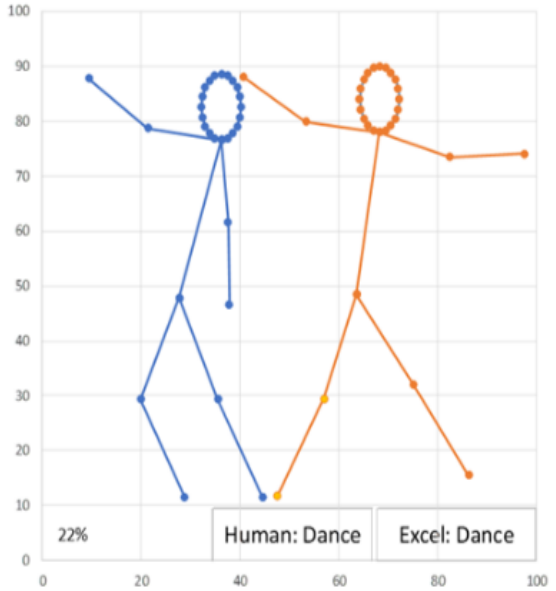
Hadley Wickham

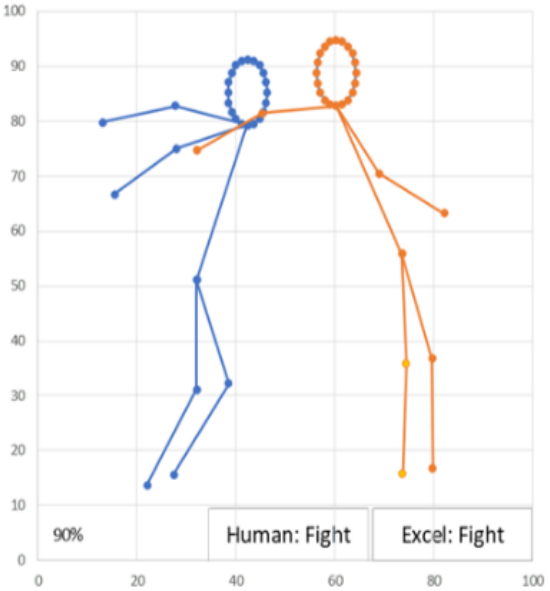
Data Driven 서비스는 시나리오 기반으로 비즈니스 모델의 개입하여 최적화하고 플랫폼 기능을 데이터(추천, 예측, 시뮬레이션 등)로 제공하는 것

데이터 분석에서 얻어지는 인사이트는 가설 기반으로 **"고객의 경험"**을 **"숫자"**로 표현하는 것

데이터 분석

데이터 분석가에게 요구하는 것들





데이터 분석

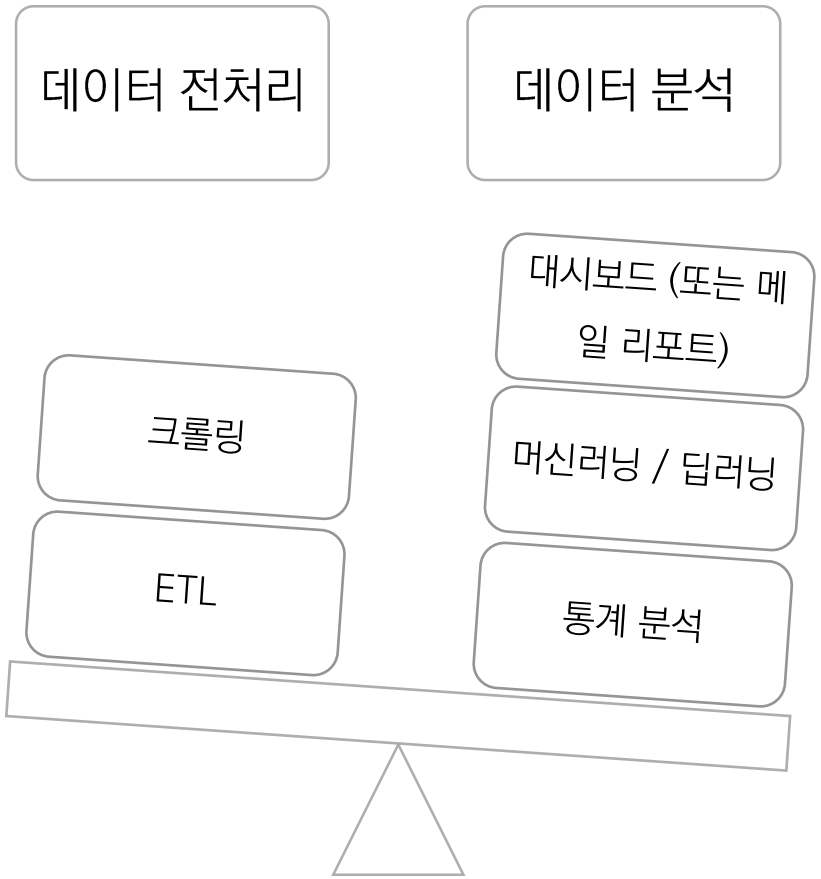
데이터 분석가에게 요구하는 것들

흐름의 변화

(2012년) RDB => Hadoop Ecosystem 로 전환

(2016년) Cloud 에서 Data Lake가 구축된 환경

(2020년) COVID-19 and X Analytics



Hadoop Ecosystem : Hadoop (클러스터에서 동작하는 분산 응용 프로그램을 지원하는 자바 소프트웨어 프레임워크)에서 상호작용하는 프로젝트들 (ex, HDFS,MapReduce, Hive, …)
Data Lake : 대규모의 원시 데이터 세트를 기본 형식으로 저장하는 데이터 레퍼지토리
X Analytics : 텍스트 분석, 비디어 분석, 오디오 분석 등과 같은 다양한 비구조화 컨텐츠에 대한 데이터 분석

데이터 분석

Deadly Mistakes

여러가지 실수

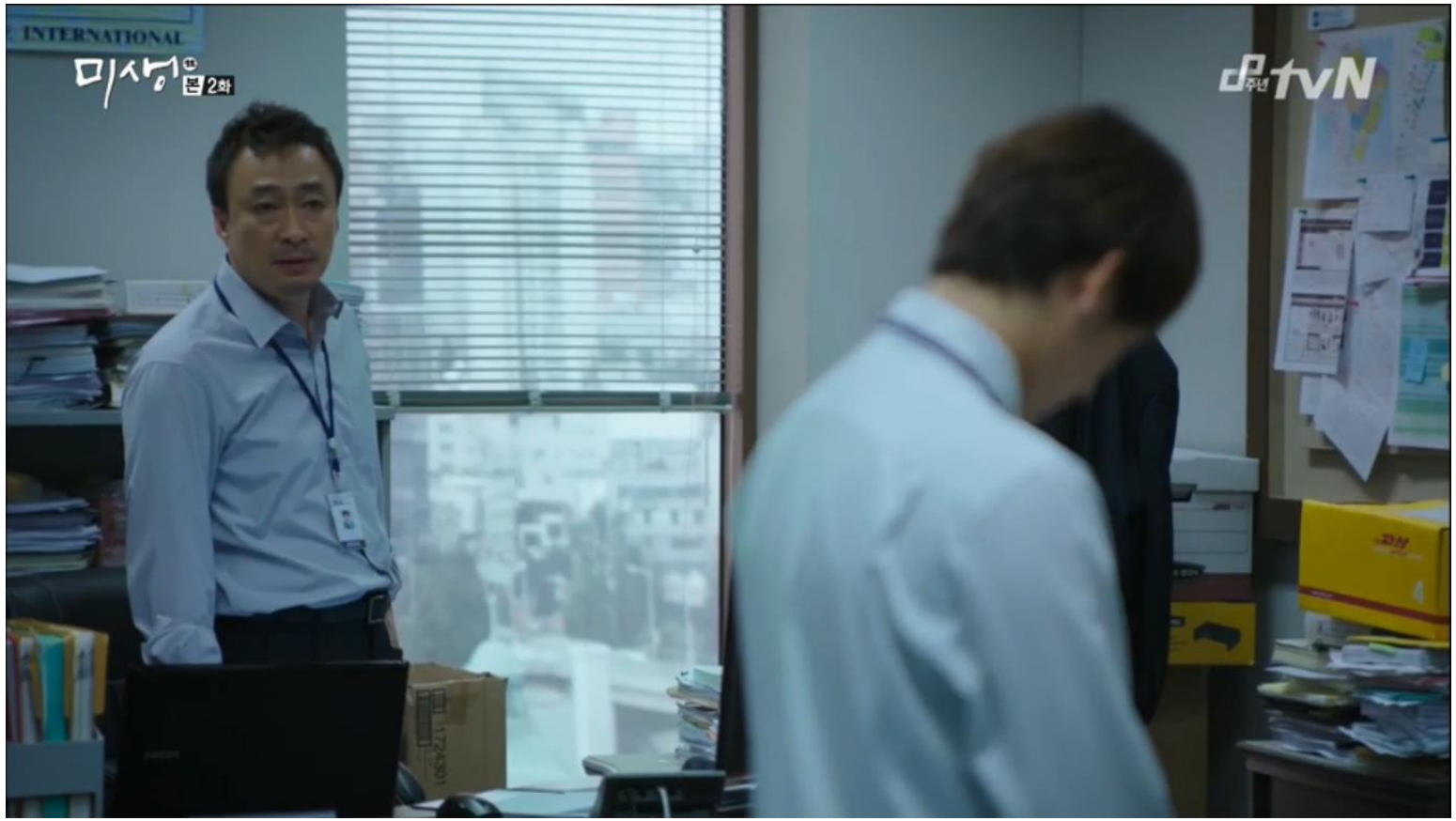
데이터 전처리 하면서 발생하는 실수들
- 데이터 조회
- Join
- 중복

분석 과정에서 발생하는 실수들
- 중앙값 / 평균
- 카테고리 / NULL 처리
- 정규화 또는 표준화
- 축에 대한 설정 (시각화)

사람마다 실수는 종류가 *"다양"*하고 *"다를 수"* 있기 때문에

이 발표에서는 *"개인의 경험"*에 대해서 이야기합니다.

데이터 전처리



팀장님

장그래
새로 런칭한
A게임
평균 플레이시
간 확인해

Junior

네, 접속 날짜
별로 확인해
보겠습니다.

데이터 전처리

데이터 조회

```
SELECT * FROM LOGIN_TABLE
```

SQL 문법 (LOGIN_TABLE 전체를 조회합니다)

분석을 하기 위해서 나는 로그 정보(데이터)를 보고싶다!

근데, 로딩이 끝나지 않음...

데이터 전처리

데이터 조회

```
SELECT * FROM LOGIN_TABLE
```

(MySQL Workbench) 30초 커넥션 타임아웃, Limit 갯수 제한이 있습니다. 결과를 로딩하다가 팅길 수 있습니다.

(MSSQL) 격리 수준을 설정하지 않으면,
조회하는 중 해당 테이블이 트랜잭션(Insert, Delete 등) 이 실행되지 않아 한소리 들을 수 있습니다.
* Hints 사용

(Hive) Partition 을 조건으로 걸지 않으면 무한 로딩이 생길 수 있습니다. (Full Scan)

모든 데이터를 눈으로 확인해볼 필요는 없기 때문에, head (limit)를 찍는 습관을 갖는 것이 중요합니다.
(Tips) 특정 Editor 사용할 때에는, limit 조회 쿼리를 단축키로 설정하면 좋습니다.

데이터 전처리

데이터 조인

헐!! 데이터가 뻥튀기 되었네!!!

라고 말하는 순간...

데이터 조인에서 발생하는 문제

데이터 전처리

SQL JOIN

여러 테이블의 레코드를 조합

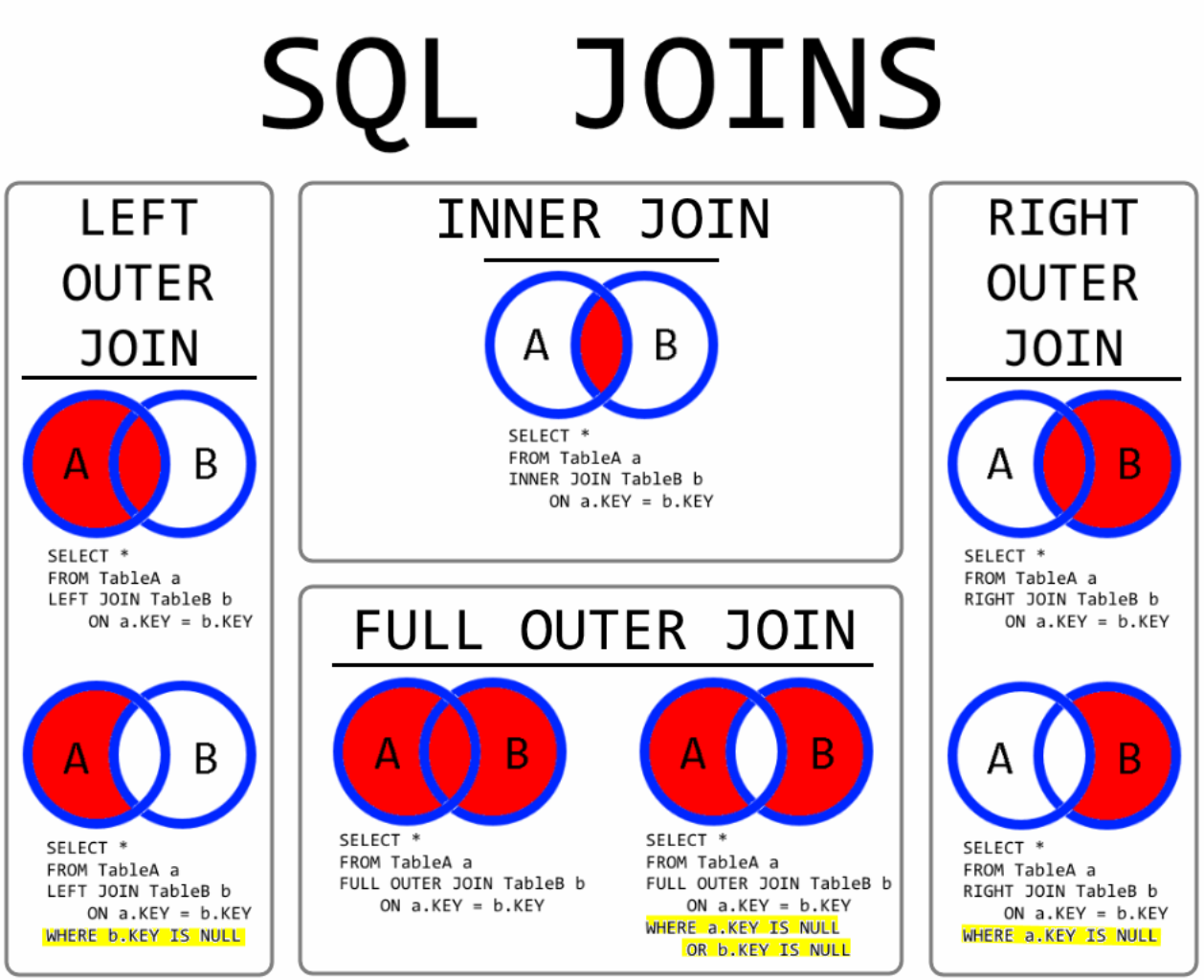
발생하는 문제들

1. 조인 키의 조건 실수

2. 테이블 간의 분포를 확인
(B 데이터가 클 경우,
조인하는 키 확인 필요)

3. JOIN (ON vs WHERE)

ON : Join을 하기 전의 필터링
WHERE : Join을 한 후의 필터링



데이터 전처리

SQL JOIN

Q) 접속 유저(일별)의 gold 보유량의 평균값을 알고 싶다면?

login

UserID	login_dt	Level
10001	2019-01-01	10
10001	2019-01-15	15
10001	2019-01-30	15
10002	2019-01-14	11
10009	2019-01-30	20

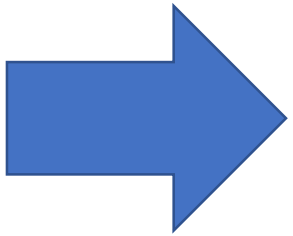
status

UserID	std_dt	Gold
10001	2019-01-01	100
10001	2019-01-15	200
10001	2019-01-30	150
10002	2019-01-14	140
10009	2019-01-30	100

데이터 전처리

SQL JOIN

Ex)
```
select a.*, b.gold
from login as a
inner join status as b
on a.UserID = b.UserID
and a.login_dt = b.std_dt
```



UserID	login_dt	Level	gold
10001	2019-01-01	10	100
10001	2019-01-15	15	200
10001	2019-01-30	15	150
10002	2019-01-14	11	140
10009	2019-01-30	20	100

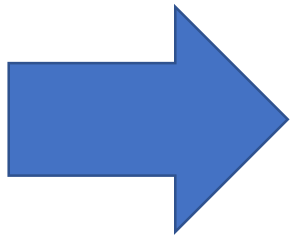
일별 gold 평균 계산

근데 대상되는 key 중에 날짜 조건을 깜박하고 join 조건에 넣지 않으면 어떻게 될까?

데이터 전처리

SQL JOIN

Ex)
select a.UserId, a.login_dt,
b.std_dt, a.level, b.gold
from login as a
inner join status as b
on a.UserID = b.UserID

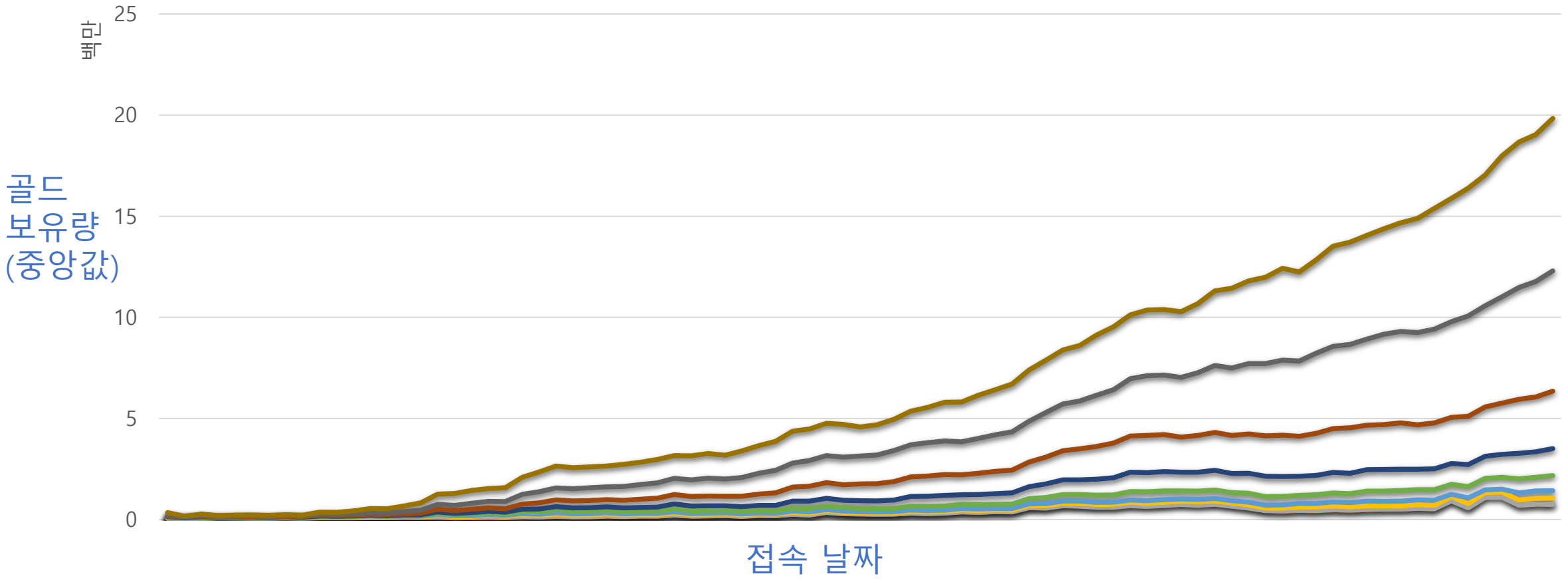


뻥튀기된 데이터

UserID	login_dt	std_dt	Level	gold
10001	2019-01-01	2019-01-01	10	100
10001	2019-01-15	2019-01-01	15	100
10001	2019-01-30	2019-01-01	15	100
10001	2019-01-01	2019-01-15	10	200
10001	2019-01-15	2019-01-15	15	200
10001	2019-01-30	2019-01-15	15	200
10001	2019-01-01	2019-01-30	10	150
10001	2019-01-15	2019-01-30	15	150
10001	2019-01-30	2019-01-30	15	150
10002	2019-01-14	2019-01-14	11	140
10009	2019-01-30	2019-01-30	20	100

데이터가 뻥튀기가 되면서, 통계량이 예상한 것과 다르게 동작할 수 있음
(조인했을 때, 테이블 Dimension 를 확인하는 습관 필요)

데이터 전처리



일별/레벨 구간별 골드 중앙값을 계산했는데 다음과 같이 snowfall이 되었음 (중복계산 이슈)

데이터 전처리

중복 제거

로그 데이터를 다루다보면 발생하는 문제 (의도와 다르게 종종 발생)

Login Log

UserID	login_time	level	gold	crystal	...	Ip
10001	2019-01-01 10:00:00 11:24:11	10	150	0		
10001	2019-01-01 10:00:00 11:24:11	10	150	0		
10001	2019-01-01 10:00:00 11:24:11	10	150	0		

ex) 로그인 로그의 약 5%가 중복된 데이터가 들어옴

로그 파싱의 문제가 발생하거나, 클라이언트에서 로그를 다량으로 보내거나, 서버가 잘못 수집되었을 때

데이터 전처리

중복 제거

"데이터가 중복이 없겠지" 라고 생각하는 순간 다시 또 뻥튀기의 문제를 발견할 수 있음..
(해결책) 모든 행의 distinct 로 처리하면 중복이 없어지긴 하지만, 오래 걸림

데이터 전처리

중복 제거

어뷰징 유저(Abusing) 가 게임 Action을 아래와 같이 의도적으로 부풀리는 경우도 있음

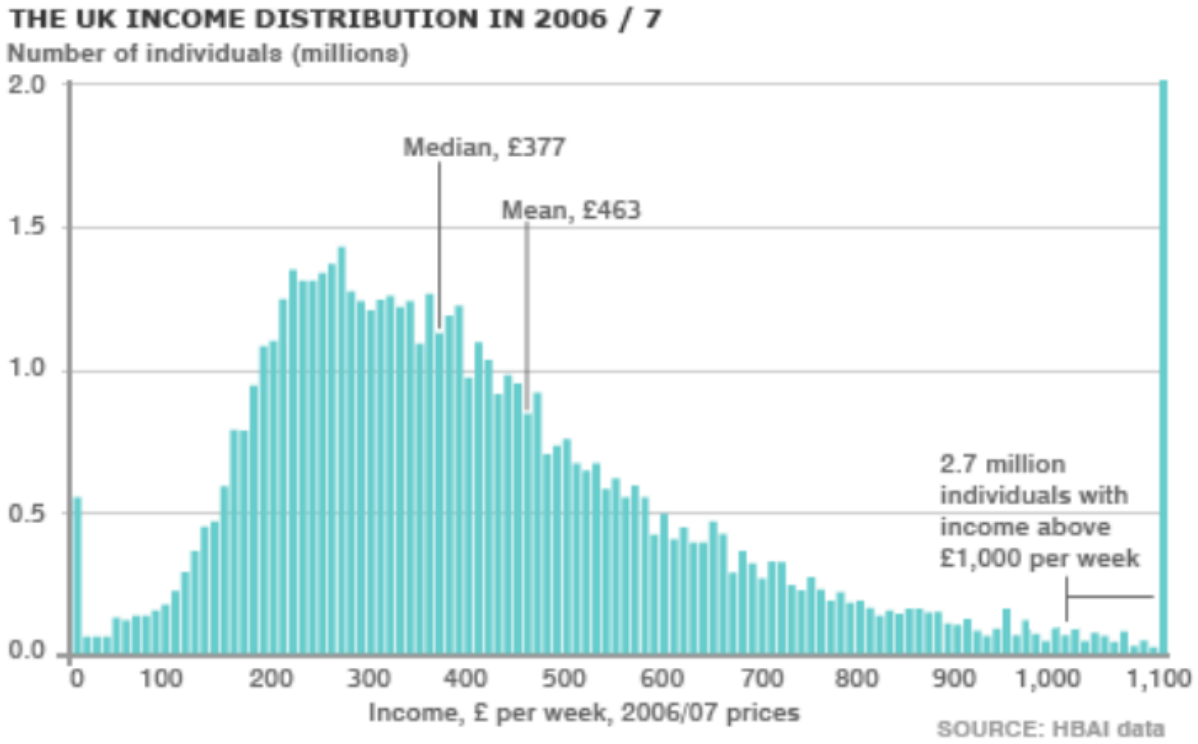
유저 1명이
하루에 대략 350만건
(초당 약 40건을
직접 생성해서 전송)

Logdetailname	count
보상 수령	1,850,363
티켓(플레이시작시 필요한 재화) 발송	1,423,755
아이템 삭제	82,627
아이템 판매	75,043
재화교환	49,291
게임시작	12,331
게임종료	12,322
유물 보상 획득	9,362
로그인	7,186
유물 사용	4,171
아이템 사용	3,259
아이템 성장 (레벨업)	3,206
미션(업적) 달성 후 보상 획득	3,042
CDN 패치 오류	2,985
출석체크	995
아이템 뽑기	95
길드 출석	10
아이템 합성	5
랭킹 보상 획득	5

데이터 분석

데이터 분석

중앙값 / 평균



평균은 반드시 중앙 근처에 있는 것이 아니기 때문에
(데이터 분포에 따라서 대표값끼리 차이가 있음)

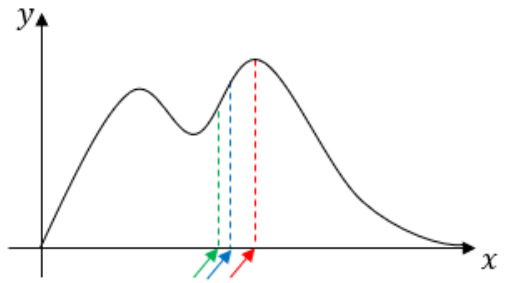
데이터 분석가 면접에 가장 많이 물어보는 단골 질문 (통계학의 대표값은 어떤 것이 있고, 어떨 때 사용하는지?)

데이터 분석

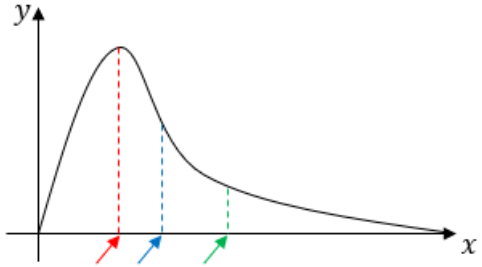
중앙값 / 평균

최빈값 : 표본에서 가장 자주 발생한 값
중앙값 : 표본에서 중앙에 위치하는 값 혹은 그 이하가 표본의 절반을 차지하는 값
평균 : 표본을 모두 더한 후 표본의 수로 나눈 값

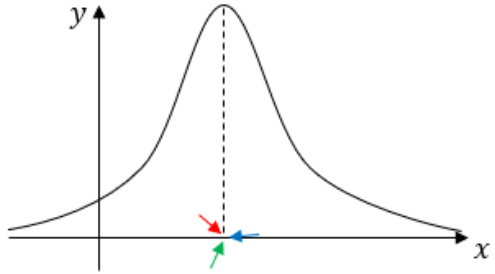
(1) 단봉형이 아닌 분포



(2) 단봉이면서 오른쪽으로 꼬리가 긴 분포



(3) 단봉이면서 대칭인 분포



출처(https://freshrimpsushi.tistory.com/740)

데이터 분석

카테고리 변수 처리

Training Data

캐릭터	갯수
전사	100
마법사	120
궁수	60
팔라딘	100
싸울아비	60

Test Data

캐릭터	갯수
전사	60
마법사	110
궁수	50
팔라딘	110
싸울아비	65
NPC	1

학습 데이터는 없는데
테스트 데이터에 추가되었을 때

ML(Machine Learning)을 개발할 때 카테고리 변수들의 새로운 level(수준)이 추가되었을 때 Error가 발생할 수 있음

Hint) 정의되지 않은 알 수 없는 값을 “etc(또는 기타)” 로 코딩하는 것이 일반적인 관행

=〉 특정 라이브러리(spark.mllib.feature)에서는 option(setHandleInvalid(‘skip’)) 으로 Control

데이터 분석

NULL 처리

NULL (초기화 되지 않은 Empty 값) vs NA (Not Available, 할당하지 않은 값, 결측치) (Python 에서는 NaN으로 표시)

```
import pandas as pd
import numpy as np

df = pd.DataFrame([[1, np.nan, 2, np.nan],
                   [3, np.nan, 4, np.nan],
                   [5, 6, 7, np.nan],
                   [np.nan, np.nan, np.nan, np.nan]],
                   columns = list('abcd'))
```

```
print(df)
```

```
     a    b    c   d
0  1.0  NaN  2.0 NaN
1  3.0  NaN  4.0 NaN
2  5.0  6.0  7.0 NaN
3  NaN  NaN  NaN NaN
```

일반적으로 NULL은 Python의 경우 (NoneType) 으로 분리

1. 결측치 또한 Level(수준)으로 처리하거나
2. 결측치를 새로운 값으로 채워넣는 방식을 선택

데이터 분석

정규화 또는 표준화

정규화(normalization) vs 표준화(standardization)

표준화 = (요소값 – 표본 평균) / 표준편차

정규화(MinMax) = (요소값–최소값) / (최대값–최소값)
(정규화는 다양한 방식이 존재)

정규화는 값의 범위를 0~1 사이의 값으로 바꾸는 것. 데이터의 범주를 바꾸는 작업

표준화는 값의 범위를 평균 0, 분산 1이 되도록 바꾸는 것, 표준 정규분포로 변환하는 것과 같음, Z-score(표준 점수)

ML 개발할 때 Feature 를 어떤 방식으로 전처리 할 지는 **“데이터”와 “모델”**에 따라서 일부 차이가 있음

데이터 분석

정규화 또는 표준화

정규화(normalization) vs 표준화(standardization)

관측치	X	Y	Z
전사	3000	1	1
마법사	2000	2	0.5
궁수	1000	3	0

The distance between X and Y =

$$\sqrt[2]{(X_{p1} - X_{p2})^2 + (Y_{p1} - Z_{p2})^2}$$

$$= \sqrt[2]{(3000 - 2000)^2 + (1 - 2)^2}$$

= 1000.0005

The distance between Z and Y =

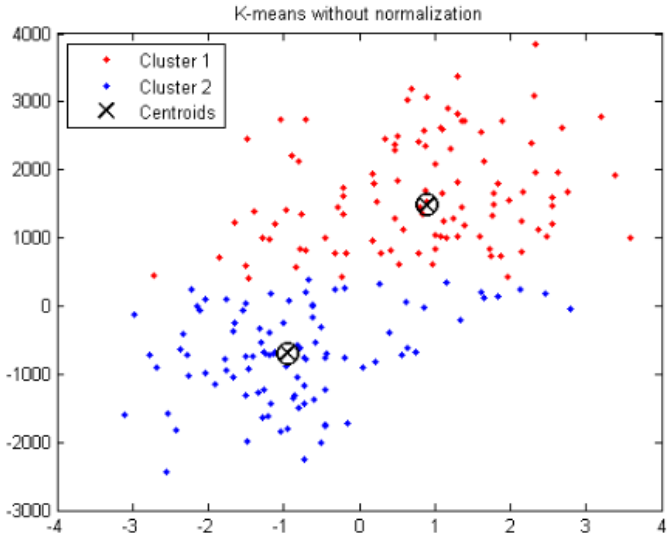
$$\sqrt[2]{(Z_{p1} - Z_{p2})^2 + (Y_{p1} - Z_{p2})^2}$$

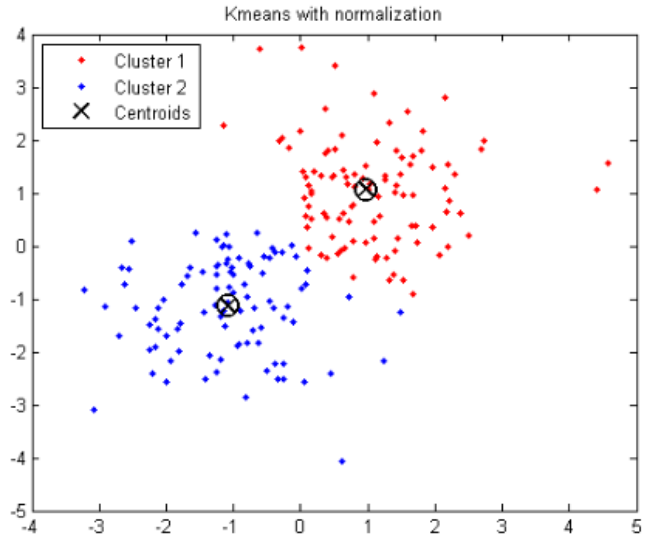
$$= \sqrt[2]{(1 - 0.5)^2 + (1 - 2)^2}$$

= 1.11803

Ex) Feature가 위도, 경도와 같이 잘 정의된 의미를 가질 경우에는 크기를 조정하면 안됨

Kmeans





데이터 분석

정규화 또는 표준화

정규화(normalization) vs **표준화(standardization)**

Sonar Dataset(Dimension : 208/60)
금속 실린더에 반사된 초음파 신호와 원통형 모양인 암석에서 반사된 신호를 구별하는 분류(classifier) 작업

Name	Sklearn_Class
LR	LogisticRegression
LDA	LinearDiscriminantAnalysis
KNN	KNeighborsClassifier
CART	DecisionTreeClassifier
NB	GaussianNB
SVM	SVC
RF	RandomForestClassifier
MLP	MLPClassifier

훈련 데이터의 10-fold 무작위 교차 검증에 대한 정확도 점수

model scaler	CART	KNN	LDA	LR	MLP	NB	RF	SVM
	0.735	0.753	0.699	0.754	0.778	0.657	0.71	0.608
MaxAbsScaler	0.735	0.808	0.699	0.778	0.767	0.657	0.71	0.705
MinMaxScaler	0.735	**0.813**	0.699	0.778	0.767	0.657	0.71	0.711
Normalizer	0.716	0.765	0.692	0.699	0.724	0.662	**0.723**	0.524
PowerTransformer-Yeo-Johnson	0.729	0.813	**0.747**	0.76	**0.839**	**0.752**	0.71	0.814
QuantileTransformer-Normal	0.735	0.783	0.694	0.718	0.808	0.752	0.717	0.832
QuantileTransformer-Uniform	**0.74**	0.789	0.742	**0.808**	0.808	0.752	0.717	0.772
RobustScaler	0.735	0.76	0.699	0.735	0.808	0.657	0.71	0.776
StandardScaler	0.735	0.796	0.699	0.742	0.82	0.657	0.71	**0.849**

StandardScaler(Z-score Standardization)

Ex) SVM에서는 특성(Feature)의 동일한 크기를 맞추기 위해 표준화를 통한 스케일링이 필수

데이터 분석

정규화 또는 표준화

많이 하는 실수 (scikit-learn Normalizer 클래스 / spark.mllib Normalizer)

Normalizer 클래스는 정규화 이지만 **"Column 기반"**이 아닌 **"Row 기반"**으로 정규화 하는 기술 (p-norm 방식)

```
In [4]: from sklearn.preprocessing import Normalizer

        X = [[4, 1, 2, 2],
             [1, 3, 9, 3],
             [5, 7, 5, 1]]

        transformer = Normalizer().fit(X)

        transformer

Out[4]: Normalizer()

In [5]: transformer.transform(X)

Out[5]: array([[0.8, 0.2, 0.4, 0.4],
               [0.1, 0.3, 0.9, 0.3],
               [0.5, 0.7, 0.5, 0.1]])
```

Default는 l2 norm 방식

표준화 vs 정규화만 기억하고, 많이 사용하지만
unit vector를 만들기 때문에 의도와 다르게 동작할 수 있음

p-norm [edit]

Main article: L^p space

Let $p \geq 1$ be a real number. The p-norm (also called ℓ_p-norm) of vector $\mathbf{x} = (x_1, \ldots, x_n)$ is

$$\|\mathbf{x}\|_p := \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}.$$ [7]

For $p = 1$, we get the taxicab norm,[6] for $p = 2$, we get the Euclidean norm, and as p approaches ∞ the p-norm approaches the infinity norm or maximum norm:

$$\|\mathbf{x}\|_\infty := \max_i |x_i|.$$

The p-norm is related to the generalized mean or power mean.

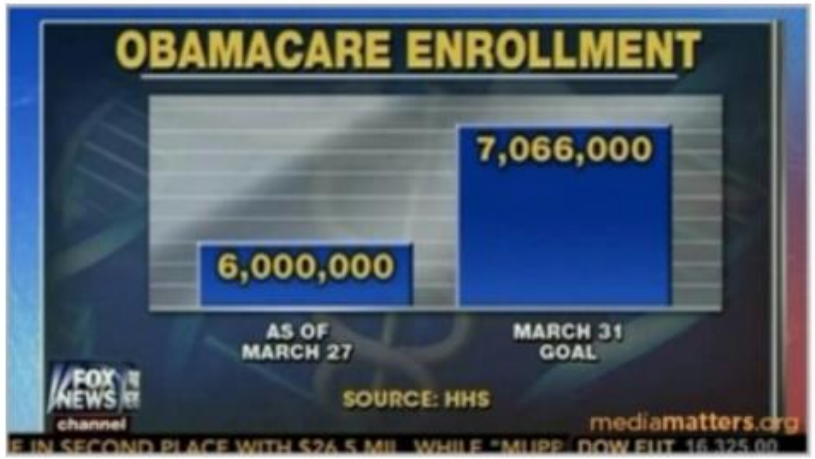
데이터 분석

축에 대한 설정 (시각화)

<u>이 문제는 기준이 다를 수 있기 때문에 “논란”이 있을 수 있습니다.</u>

가설을 확인하기 위해 데이터 분석을 하고, 의도에 따라서 다양하게 데이터는 해석될 수 있음

특히 시각화 영역이 가장 오해를 불러올 수 있고, 잘못된 시각화의 정보로 많은 사람들의 질타를 받기도 합니다.



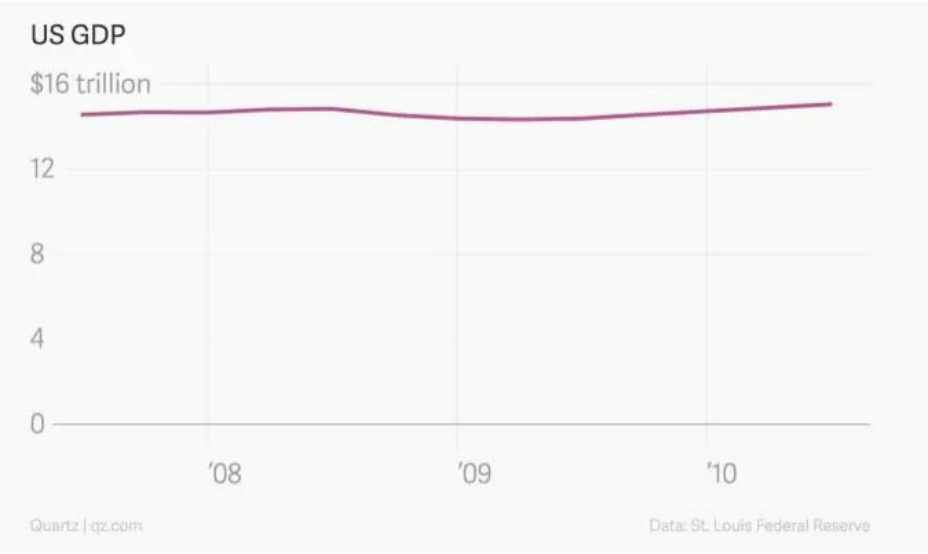
Cropped Axes : 오바마 케어 등록 현황 그래프인데, 왼쪽 그래프는 약 3배 정도 차이가 나는 것으로 보이지만 사실 막대차트의 시작 값을 0으로 하지 않았기 때문에 발생한 문제

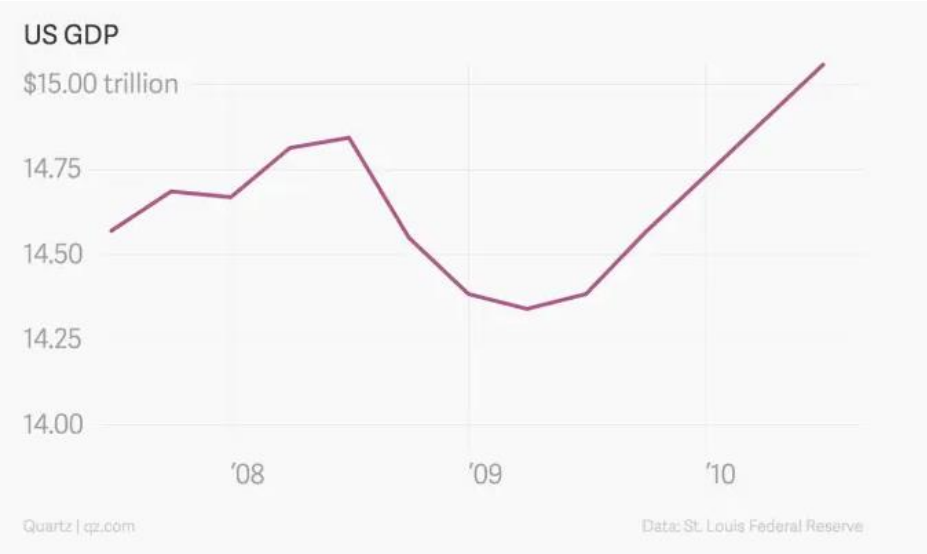
데이터 분석

축에 대한 설정 (시각화)

그렇다고, 모든 차트의 시작값(최소값) 설정을 꼭 0으로 해야하는지?
(Cropped Axes 가 잘못되었는가, y축이 항상 0부터 시작하지 않으면 거짓말인가??)

[그렇지 않다. 차트는 정보를 전달하고 요점을 만들어야 함](#) (Quartz : 미국의 디지털 뉴스 발행 서비스)

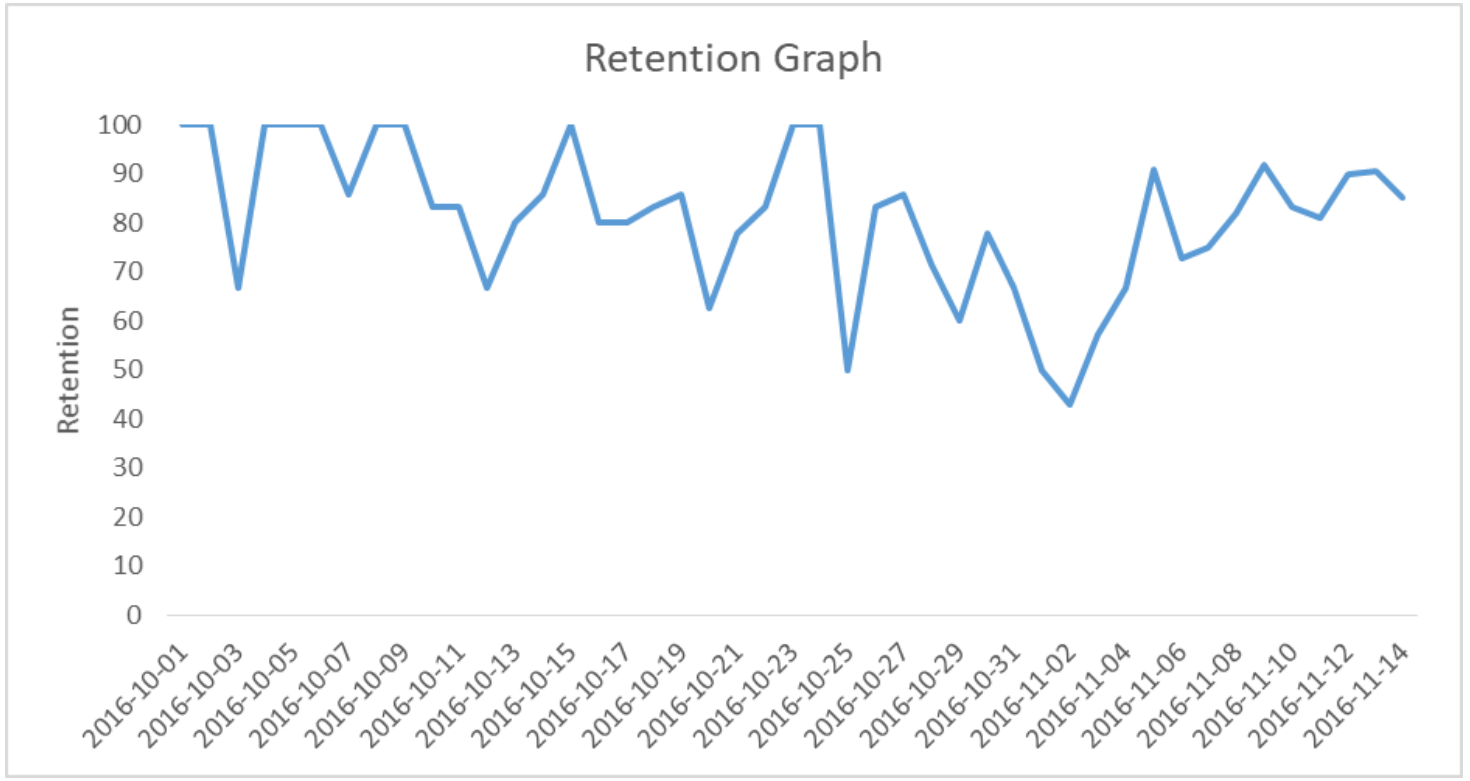




같은 데이터 (y축을 자를 경우, 미국 GDP(불황의 저점)을 확인하기 쉬움)

데이터 분석

축에 대한 설정 (시각화)



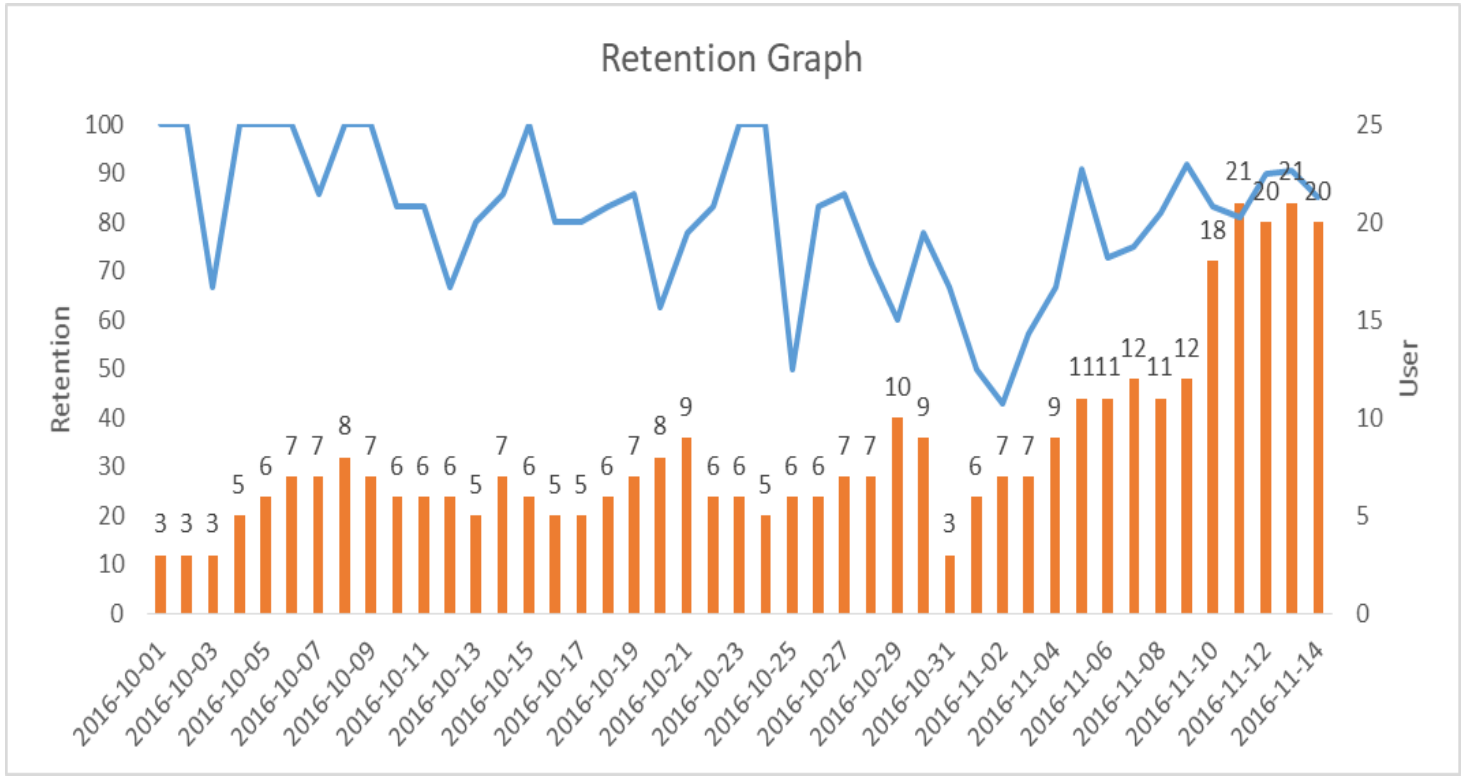
시계열의 트렌드를

확인하기 위해 데이터를 보니

그래프 변곡점이 너무 심함

데이터 분석

축에 대한 설정 (시각화)



모수가 적으면 (ex, User)

접속한 유저 3명 중에
2명만 나가도 33%가 될 수 있어
비율만 보면 데이터를 오해할 수 있음

다음의 방법처럼

1. 모수를 같이 표시
2. 다차원 집단 분석을 할 때 30개 미만일 경우에 분석 대상에서 제외할 필요도 있음

데이터 분석

Q&A는?

현장(온라인)으로 질문주시거나 아래의 email로 보내주시면

답변드리겠습니다.!

syleeie@gmail.com