# 구글 앱스토어
# Rating 예측 모델링

# 개요

| | |
|---|---|
| 주제 | 구글 앱스토어 앱의 Rating(평가점수) 예측 모델링 |
| 활용 방안 | 앱 업데이트 시 Rating의 변화 시뮬레이션<br>앱 시장에서 특정 앱의 포지션 파악 등 |
| 활용 데이터 | kaggle의 googleplaystore.csv 데이터 (13개 컬럼, 10,841건)<br>https://www.kaggle.com/lava18/google-play-store-apps |
| 문제 정의 | 타겟 변수인 Rating을 나머지 변수를 활용하여 예측<br>이 때 Rating은 연속적인 수치형 변수 => 회귀 모델링 문제 |
| 분석 과정 | 데이터 전처리 & 탐색 -> 모델 생성 & 검증 -> 최적화 & 최종 모델 선택 |
| 분석 환경 | Python 언어 사용, Jupyter Notebook 환경에서 작업 |
| 활용 패키지 | numpy, math 연산<br>pandas 데이터 조작<br>random 난수 생성<br>scikit-learn, lightgbm, keras 모델링<br>bayes_opt, functools 최적화, 변수 고정<br>matplotlib.pyplot, seaborn 시각화 |

# 데이터 전처리 & 탐색

## 전체 데이터 점검

### 1. 잘못 읽어들인 데이터

| | App | Category | Rating | Reviews | Size | Installs | Type | Price | Content Rating | Genres | Last Updated | Current Ver | Android Ver |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Life Made WI-Fi Touchscreen Photo Frame | | 1.9 | 19.0 | 3.0M | 1,000+ | Free | 0 | Everyone | NaN | February 11, 2018 | 1.0.19 | 4.0 and up | NaN |

| | App | Category | Rating | Reviews | Size | Installs | Type | Price | Content Rating | Genres | Last Updated | Current Ver | Android Ver |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Life Made WI-Fi Touchscreen Photo Frame | | LIFESTYLE | 1.9 | 19 | 3.0M | 1,000+ | Free | 0 | Everyone | Lifestyle | February 11, 2018 | 1.0.19 | 4.0 and up |

-> 열이 밀려있는 경우 복원, 결측치는 검색하여 채워넣음

### 2. 중복 데이터

| App | Category | Rating | Reviews | Size | Installs | Type | Price | Content Rating | Genres | Last Updated | Current Ver | Android Ver |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Box | BUSINESS | 4.2 | 159872 | Varies with device | 10,000,000+ | Free | 0 | Everyone | Business | July 31, 2018 | Varies with device | Varies with device |
| Box | BUSINESS | 4.2 | 159872 | Varies with device | 10,000,000+ | Free | 0 | Everyone | Business | July 31, 2018 | Varies with device | Varies with device |
| Box | BUSINESS | 4.2 | 159872 | Varies with device | 10,000,000+ | Free | 0 | Everyone | Business | July 31, 2018 | Varies with device | Varies with device |

-> 10,841건 중 중복 데이터 1,181건 삭제

## 타겟 변수 Rating

1. **결측치** -> 행 삭제 처리
2. **데이터 타입** -> float로 변환

| | App | Category | Rating | Reviews |
|---|---|---|---|---|
| 23 | Mcqueen Coloring pages | ART_AND_DESIGN | NaN | 61 |
| 1 | Coloring book moana | ART_AND_DESIGN | 3.9 | 967 |
| 2 | U Launcher Lite – FREE Live Cool Themes, Hide ... | ART_AND_DESIGN | 4.7 | 87510 |

# 데이터 전처리 & 탐색

## Category 변수

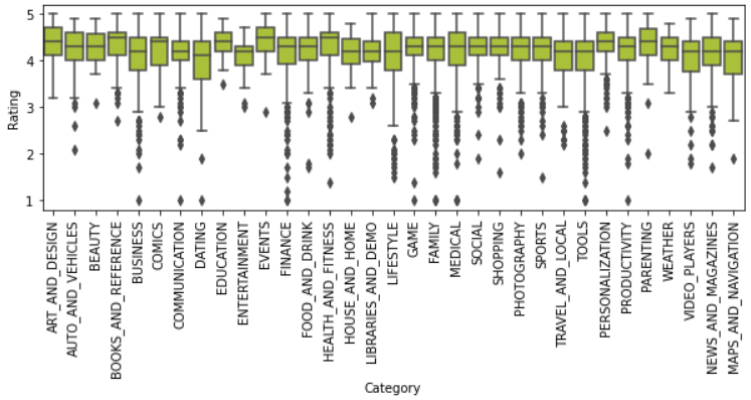
### 1. 범주형 변수 -> 더미 변수

| | Category |
|---|---|
| 0 | ART_AND_DESIGN |
| 1 | ART_AND_DESIGN |
| 2 | ART_AND_DESIGN |
| 3 | ART_AND_DESIGN |
| 4 | ART_AND_DESIGN |

⇨

| ART_AND_DESIGN | WEATHER | FAMILY |
|---|---|---|
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |

### 2. 분포 파악

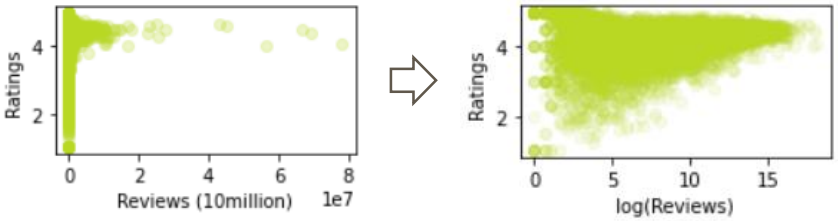



-> 약간의 개별 분포 차이가 있어
설명변수로 활용해볼 가치가 있음

## Reviews 변수

### 1. 데이터 타입 -> int로 변환

### 2. 분포 파악 -> log(x)로 변환




## Size 변수

### 1. 단위 혼재 -> k(킬로바이트) 단위로 통일

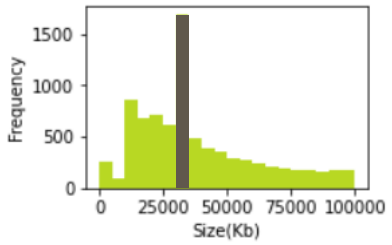
### 2. 결측치 -> 중앙값으로 대체

| | Size |
|---|---|
| 37 | Varies with device |
| 600 | 77M |
| 5783 | 976k |

⇨

| Size |
|---|
| 33000 |
| 77000 |
| 976 |

← 중앙값



# 데이터 전처리 & 탐색

## Installs 변수

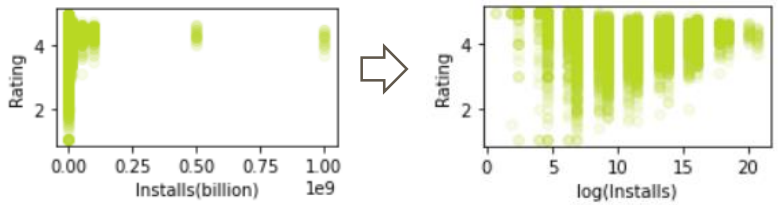
### 1. 데이터 타입 -> int로 변환

```
array(['10,000+', '500,000+', '5,000,000+', '50,000,000+', '100,000+',
       '50,000+', '1,000,000+', '10,000,000+', '5,000+', '100,000,000+',
       '1,000,000,000+', '1,000+', '500,000,000+', '100+', '500+', '10+',
       '5+', '50+', '1+'], dtype=object)
```

⇩

```
array([      10000,      500000,     5000000,    50000000,      100000,
            50000,     1000000,    10000000,        5000,   100000000,
       1000000000,        1000,   500000000,         100,         500,
               10,           5,          50,           1])
```

### 2. 분포 파악 -> log(x)로 변환

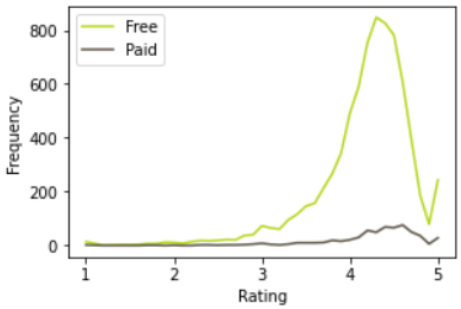



## Type 변수

### 1. 범주형 변수

-> 더미 변수

### 2. 분포 파악




## Price 변수

### 1. 데이터 타입 -> float로 변환

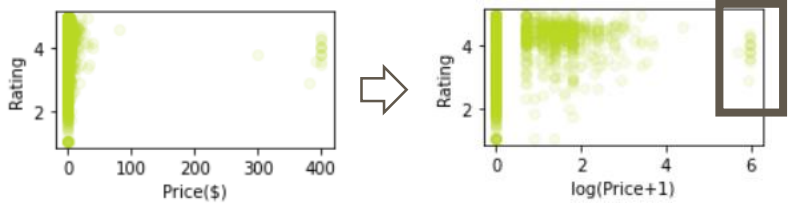
```
array(['0', '$4.99', '$3.99', '$6.99', '$7.99', '$5.99', '$2.99', '$3.49',
       '$1.99', '$9.99', '$7.49', '$0.99', '$9.00', '$5.49', '$10.00',
       '$24.99', '$11.99', '$79.99', '$16.99', '$14.99', '$29.99',
```

⇩

```
array([  0.  ,   4.99,   3.99,   6.99,   7.99,   5.99,   2.99,   3.49,
         1.99,   9.99,   7.49,   0.99,   9.  ,   5.49,  10.  ,  24.99,
        11.99,  79.99,  16.99,  14.99,  29.99,  12.99,   2.49,  10.99,
```

### 2. 분포 파악 -> log(x)로 변환

이상치




### 3. 이상치 -> 행 삭제 처리

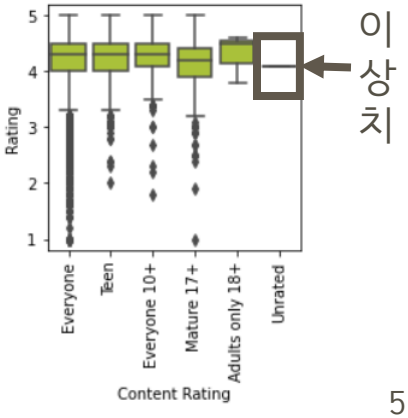
## Content Rating 변수

### 1. 범주형 변수

-> 더미 변수

### 2. 분포 파악

### 3. 이상치 -> 1건 삭제



# 데이터 전처리 & 탐색

## Genres 변수

**1. 제1정규화**

-> ';'로 구분

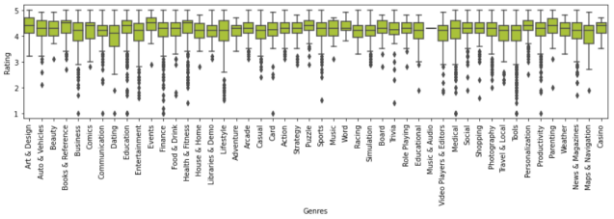
**2. 범주형 변수**

-> 더미 변수

| | Genres |
|---|---|
| 0 | Art & Design |
| 1 | Art & Design;Pretend Play |
| 2 | Art & Design |
| 3 | Art & Design |
| 4 | Art & Design;Creativity |

| Art & Design | Pretend Play | Creativity |
|---|---|---|
| 1 | 0 | 0 |
| 1 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |

**3. 분포 파악**




## Android Ver 변수

**1. 데이터 타입**

-> 주번호, 부번호

까지 사용

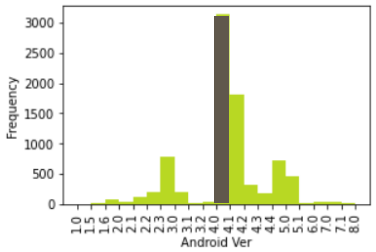
**2. 범주형 변수**

-> 더미 변수

**3. 결측치**

-> 최빈값으로 대체

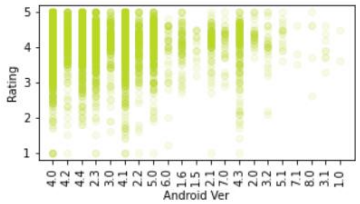
**4. 분포 파악**




최빈값

| Android Ver |
|---|
| 4.0.3 and up |
| 5.0 - 8.0 |
| Varies with device |
| NaN |

| 4.0 | 5.0 |
|---|---|
| 1 | 0 |
| 0 | 1 |
| 1 | 0 |
| 1 | 0 |




## Last Updated 변수

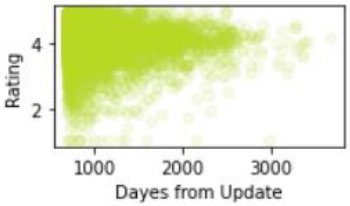
**1. 데이터 타입**

-> 마지막 업데이트일부터

2020-06-18까지 경과 일수

| Last Updated | Days from Update |
|---|---|
| January 7, 2018 | 893 |
| January 15, 2018 | 885 |
| August 1, 2018 | 687 |
| June 8, 2018 | 741 |
| June 20, 2018 | 729 |

**2. 분포 파악**



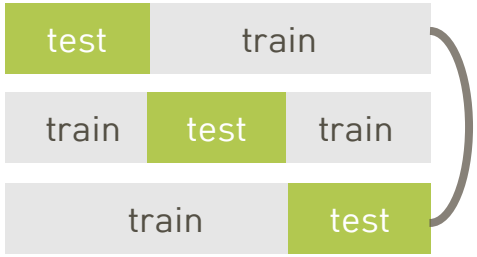
# 모델 생성 & 검증

## train, test set 분리

train (학습) | test (검증)

약 2:1로 구성

## k-fold cross validation




score의 평균
-> 우연성 최소화

## 평가 지표

예측값과 실제값의 잔차

- (절댓값의 평균) ... MAE
- √(제곱의 평균) ... RMSE

## Baseline 모델링

1. 다중 선형 회귀 모델 (sklearn.LinearRegression 활용)
2. 의사 결정 나무 모델 (lightGBM.LGBMRegressor 활용)
3. 신경망 모델 (keras.Sequential 활용)


# 최적화 & 최종 모델 선택

## hyperparameter 최적화

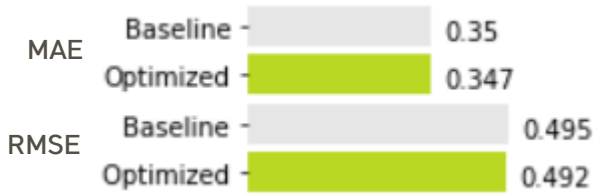
머신러닝 모델 생성 시

사용자 설정값인 hyperparameter의

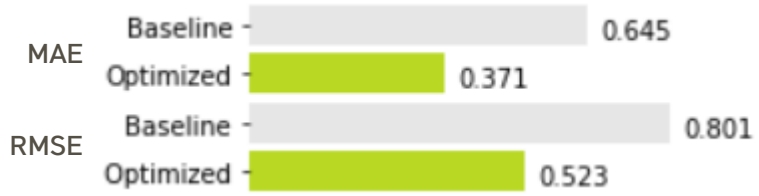
최적점을 찾아 성능 향상

### ② 의사 결정 나무 모델

bayes_opt.BayesianOptimization 활용
5회 랜덤 샘플링 후 분포 추정
45회 반복해서 최적화




### ③ 신경망 모델




## 앙상블

**1. Sampling**

같은 알고리즘을 적용하되

데이터의 컬럼과 레코드를 샘플링하여

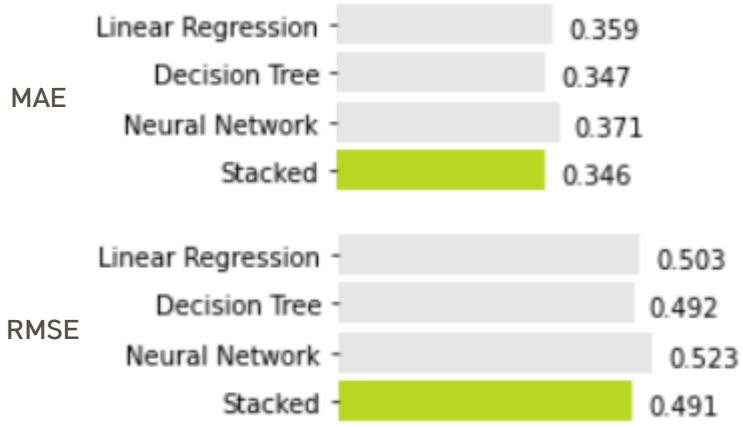
독립성을 높인 여러 모델 생성

**2. Stacking**

다른 알고리즘을 사용하는 여러 모델 생성

여러 모델에서 얻은 예측값들을

설명변수로 두고 최종 모델링



# 결론

## 최종 성능

MAE : 0.346 / RMSE : 0.491

## 의사 결정 나무 모델에서의 주요 변수 (중요도 top5)


## 선형 회귀 모델에서의 계수

Reviews : 0.16
Days from Update : -2e-4
Size : -1e-7
Installs : -0.14
Price : -0.02

⇨ 선형 회귀 모델 생성 시
스케일링을 하지 않아 계수의 절대값이 매우 작다.
부호를 통해 Rating과의 음/양의 상관관계를 알 수 있다.

## 더 해볼 것

모든 변수 활용 / 파생 변수 생성

변수 선택 / 잔차가 큰 특정 구간이 있는지 확인

- 끝 -