

빅데이터 연구회 : KU-BIG

# MyAnimeList Dataset Recommendation

송민 우경민 이강현  
윤재경 정재원 이소담



# INDEX

프로젝트

추천 시스템  
주제 선정 이유

추천 시스템

추천 결과  
성능 평가

한계/참고

한계점  
참고자료

MyAnime Data  
Data EDA

데이터 분석



# 프로젝트 추천시스템



WATCHA PLAY

## 오만과 편견

예상 별점 5.0

평균 별점 4.1

## 오만과 편견

예상 별점 5.0 평균 별점 4.1 15세 · 에피소드 6개

왓차 회원들이 가장 보고 싶어하는 상위 5% 작품

17세기 영국 남부의 마을의 제인 베넷의 집은 그녀의 아버지가 죽고 나면 먼 친척들에게 넘어갈 위기에 처하고, 가족의 행복은 다섯 딸이 행복한 결혼을 하는 일에 달린다.

감독 시몬 랑턴

출연 제니퍼 엘, 콜린 퍼스, 엘리슨 스테드먼

개요 TV드라마 · 시대극 | 영국 | 1995년



+ 보고 싶어요

⊖ 관심없어요

이미 본 작품인가요? ☆☆☆☆☆

기본정보

상세정보

에피소드

비슷한 작품

NETFLIX 영화

## 우리의 계절은

80% 일치 2018 12 1시간 14분

총 74분 중 1분

세 개의 도시, 세 개의 이야기가 시작된다. 상실을 딛고, 그리움을 안고, 내일을 바라보는 청춘들. 생의 의미를 찾아 한 발 한 발 나아가는 그들의 이야기를 담는다.



+ 내가 평한 콘텐츠



콘텐츠 정보

예고편 및 다른 영상

비슷한 콘텐츠

상세 정보

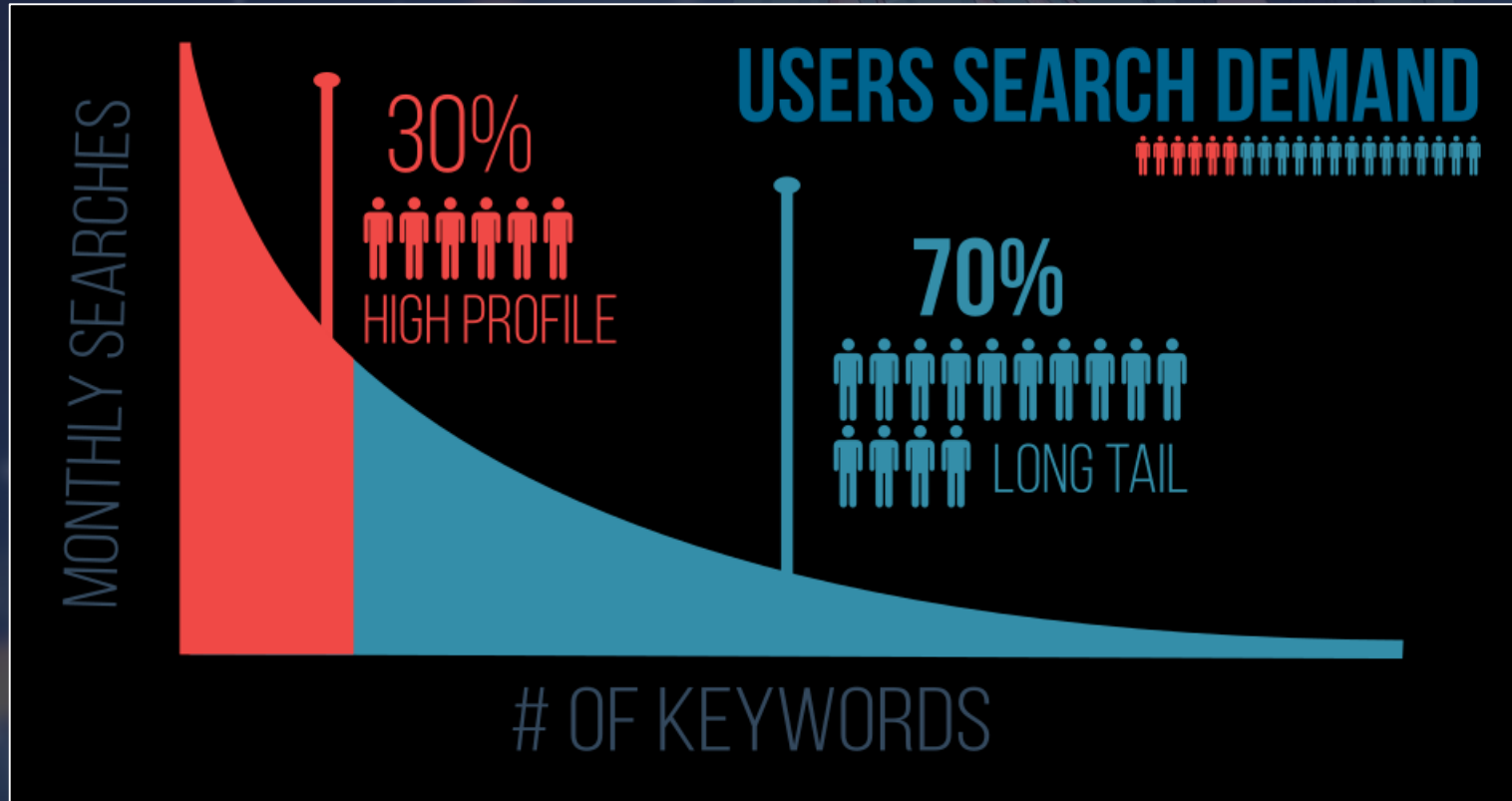
NETFLIX 영화

## 우리의 계절은

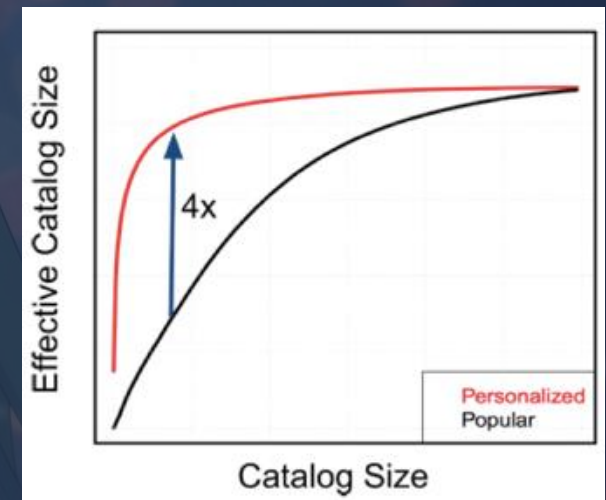
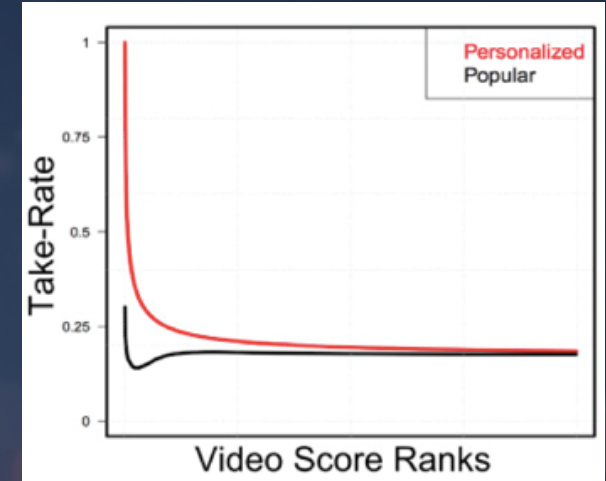
80% 일치 2018 12 1시간 14분

NETFLIX

# 프로젝트 주제 선정 이유



추천 시스템은 1년에 1.2조원의 가치





# INDEX

추천 시스템  
주제 선정 이유

추천 결과  
성능 평가

MyAnime Data  
Data EDA

한계점  
참고자료

데이터 분석

한계/참고



# 데이터 탐색

## MyAnime data

**Tsurune: Kazemai Koukou Kyuudoubu**Edit Anime Information



 Follow us on FB **MyAnimeList**  

[Details](#) [Videos](#) [Episodes](#) [Reviews](#) [Recommendations](#) [Stats](#) [Characters & Staff](#) [News](#) [Forum](#) [Clubs](#) [Pictures](#)

Top > Anime > Tsurune: Kazemai Koukou Kyuudo...

**SCORE**  
**7.60**  
17,481 users

Ranked **#1401** Popularity **#1782** Members **49,211**

Fall 2018 | TV | Kyoto Animation


[Add to List](#) [Select](#) Episodes: 0/13 

 [More videos](#)

\* Your list is public by default.


## MyAnimeList

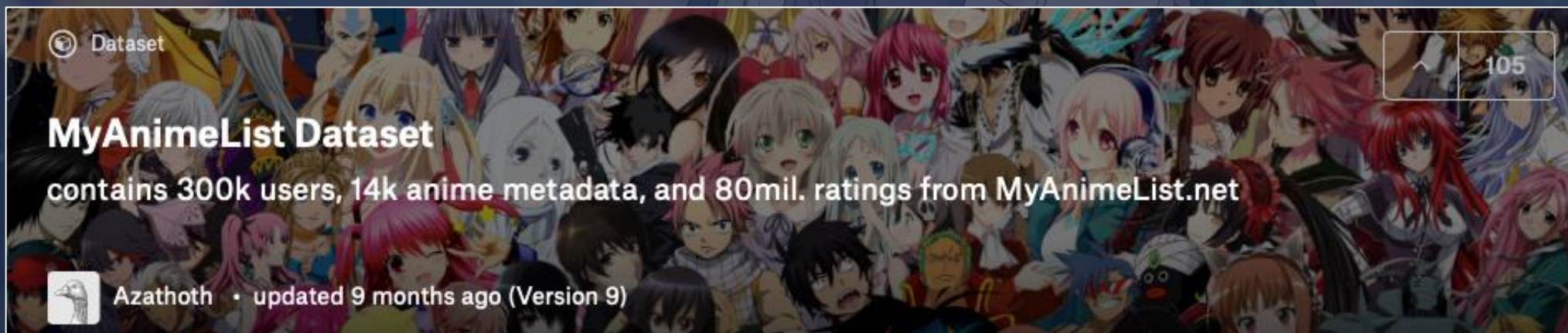
: an anime and manga social networking and social cataloging application website. The site provides its users with a list-like system to organize and score anime and manga.



 **Dataset**

## MyAnimeList Dataset

contains 300k users, 14k anime metadata, and 80mil. ratings from MyAnimeList.net

 **Azathoth** • updated 9 months ago (Version 9)



  105

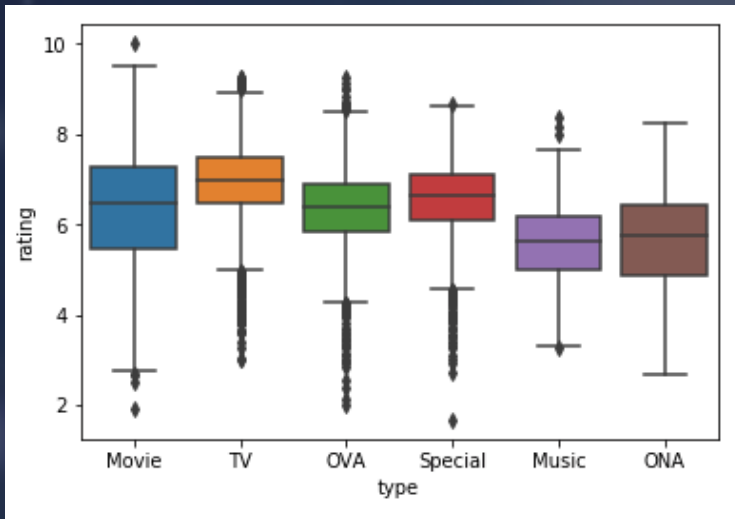
## Rating.csv

- user\_id
- anime\_id
- rating

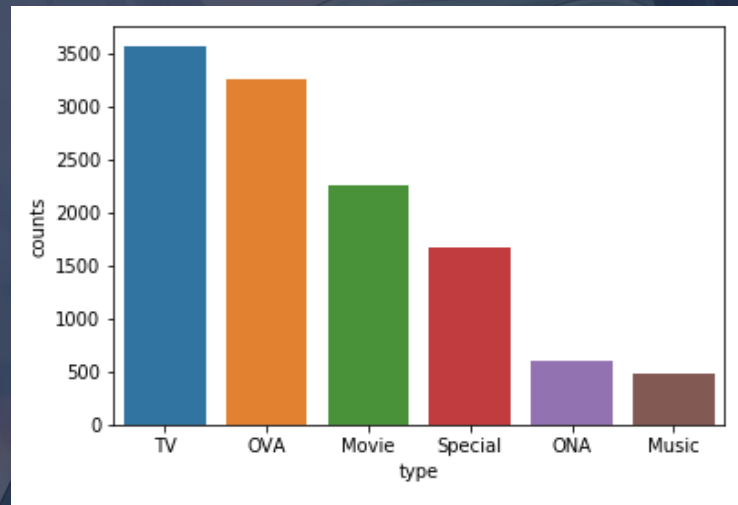
# 데이터 탐색

## Data EDA

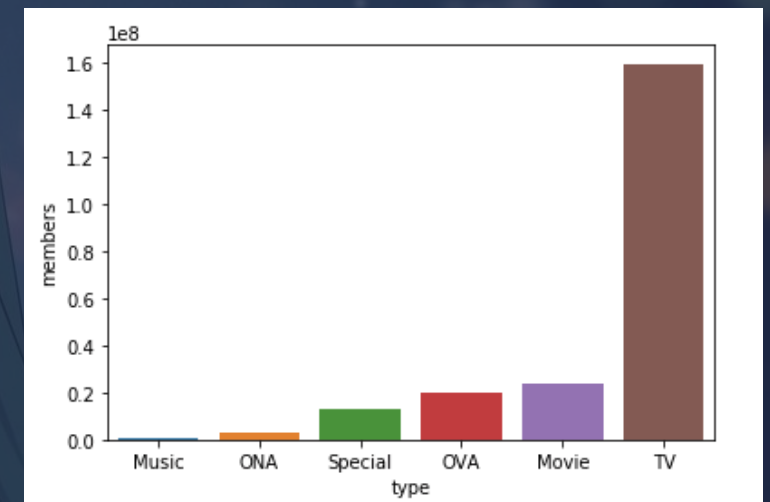
Rating - Type



Count - Type



Members - Type





# 데이터 탐색

## Data EDA

### WordCloud: Most Frequenctly Rated Anime

Naruto Movie 1: Dai Katsugeki!! Yuki Hime Shinobu Houjou Dattebayo!

Fullmetal Alchemist: The Conqueror of Shamballa

Dragon Ball Z Movie 08: Moetsukiro!! Nessen, Ressen, Chougekisen

Naruto Movie 2: Dai Gekitotsu! Maboroshi no Chiteiiseki Dattebayo!

Pokemon: Maboroshi no Pokemon Lugia Bakutan

Suzumiya Haruhi no Shoushitsu

Dragon Ball Z Movie 03: Chikyuu Marugoto Choukessen

Pokemon: Kesshoutou no Teiou Entei

Bleach Movie 3: Fade to Black - Kimi no Na wo Yobu

Dragon Ball Z Movie 05: Tobikkiri no Saikyou tai Saikyou

Tonari no Totoro

Summer Wars

Naruto: Shippuuden Movie 1

Dragon Ball Z Movie 01: Ora no Gohan wo Kaese!!

Pokemon: Mewtwo no Gyakushuu

Majo no Takkyuubin

	ANIME	SCORE
480	Fullmetal Alchemist: The Conqueror of Shamballa	3516
523	Tonari no Totoro	3572
5651	Summer Wars	3586
402	Naruto Movie 1: Dai Katsugeki!! Yuki Hime Shinobu Houjou Dattebayo!	3542
508	Pokemon: Mewtwo no Gyakushuu	3533
7511	Suzumiya Haruhi no Shoushitsu	3579
2472	Naruto: Shippuuden Movie 1	3465
1117	Pokemon: Maboroshi no Pokemon Lugia Bakutan	3442
806	Naruto Movie 2: Dai Gekitotsu! Maboroshi no Chiteiiseki Dattebayo!	3517
4535	Bleach Movie 3: Fade to Black - Kimi no Na wo Yobu	2738

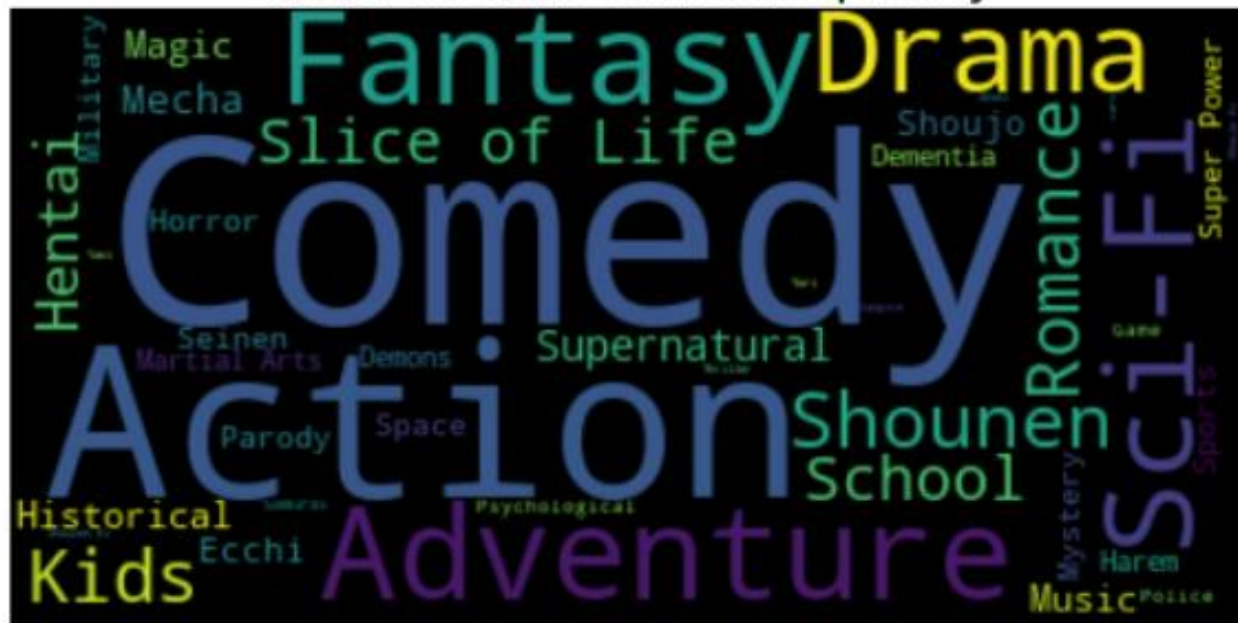


# 데이터 탐색

Data EDA

clustering

Word Cloud: Genre Frequency



cluster 1

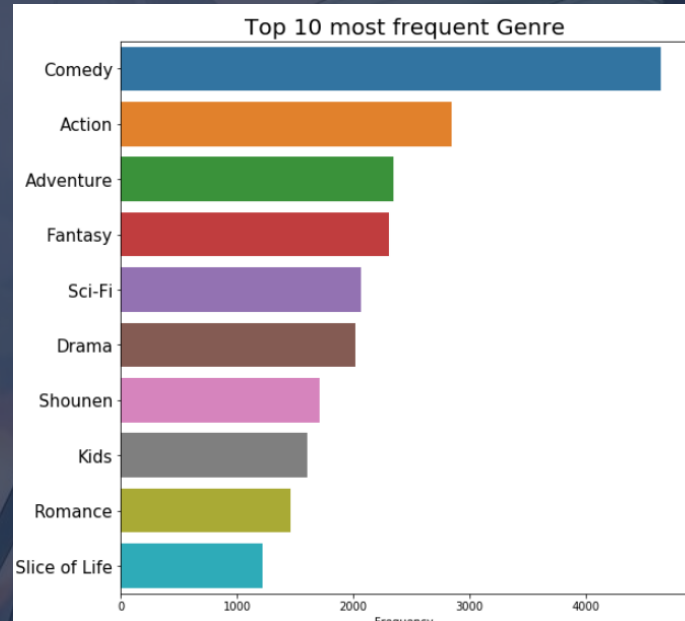


# 데이터 탐색

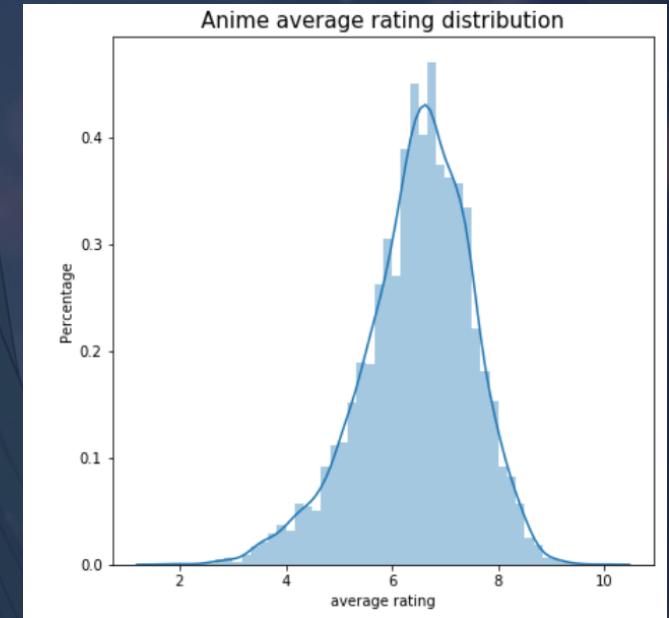
## Data EDA



Frequent Genre



Anime average rating





# INDEX

추천시스템

추천 결과  
성능 평가

한계/참고

한계점  
참고자료

MyAnime Data  
Data EDA

데이터 탐색

주제 선정 이유

추천 시스템  
주제 선정 이유

# 추천시스템

사용 알고리즘

- 협업 필터링

- Collaborative filtering

: Matrix Factorization





# 추천시스템

Python Coding



# GitHub

enflow Update README.md

Latest commit f722

<a href="#">.gitignore</a>	initial commint- implementing basic recommendation system for movie.
<a href="#">MF.py</a>	initial commint- implementing basic recommendation system for movie.
<a href="#">README.md</a>	Update README.md
<a href="#">data.py</a>	initial commint- implementing basic recommendation system for movie.
<a href="#">sim.py</a>	initial commint- implementing basic recommendation system for movie.

→ [https://github.com/KU-BIG/recommendation\\_system](https://github.com/KU-BIG/recommendation_system)

```
9 class MatrixFactorization():
```

```
11     def __init__(self, matrix, lr, reg_param, epochs):
```

```
13         self._matrix = matrix
```

```
14         self._row, self._column = matrix.shape
```

```
15         self._k = 5
```

```
16         self._lr = lr
```

```
17         self._reg_param = reg_param
```

```
18         self._epochs = epochs
```

```
21     def fit(self):
```

```
23         # initialize latent features
```

```
24         self._P = np.random.normal(size=(self._row, self._k))
```

```
25         self._Q = np.random.normal(size=(self._column, self._k))
```

```
27         # initialize biases
```

```
28         self._b_P = np.zeros(self._row)
```

```
29         self._b_Q = np.zeros(self._column)
```

```
30         self._b = np.mean(self._matrix[np.where(self._matrix != 0)])
```

```
32         #training
```

```
33         for epoch in range(self._epochs):
```

```
34             for i in range(self._row):
```

```
35                 for j in range(self._column):
```

```
36                     if self._matrix[i, j] > 0:
```

```
37                         self.gradient_descent(i, j, self._matrix[i, j])
```

```
39     def gradient_descent(self, i, j, rating):
```

```
41         # get error
```

```
42         prediction = self.get_prediction(i, j)
```

```
43         error = rating - prediction
```

# 추천시스템 추천 결과

```
#get anime name  
get_name(top10_rec_exmp)
```

## Recommendation for User 3 ##

['Gintama Movie: Kanketsu-hen - Yorozuya yo Eien Nare', 'Gintama: Yorinuki Gintama-san on Theater 2D', 'Tonari no Totoro', 'Gintama Movie: Shinyaku Benizakura-hen', 'Tengen Toppa Gurren Lagann Movie: Gurren-hen', 'Suzumiya Haruhi no Shoushitsu', 'Trigun: Badlands Rumble', 'One Piece Movie 4: Dead End no Bouken', 'One Piece Film: Strong World', 'Majo no Takkyuubin']

## Recommendation for User 19418 ##

10개의 Anime 추천 (Top 10)

['Gintama Movie: Kanketsu-hen - Yorozuya yo Eien Nare', 'Detective Conan Movie 08: Magician of the Silver Sky', 'Boruto: Naruto the Movie', 'Detective Conan Movie 07: Crossroad in the Ancient Capital', 'Gintama Movie: Shinyaku Benizakura-hen', 'Detective Conan Movie 03: The Last Wizard of the Century', 'Lupin III: Cagliostro no Shiro', 'Crayon Shin-chan Movie 09: Arashi wo Yobu Mouretsuo! Otona Teikoku no Gyakushuu', 'Mind Game', 'Detective Conan Movie 15: Quarter of Silence']

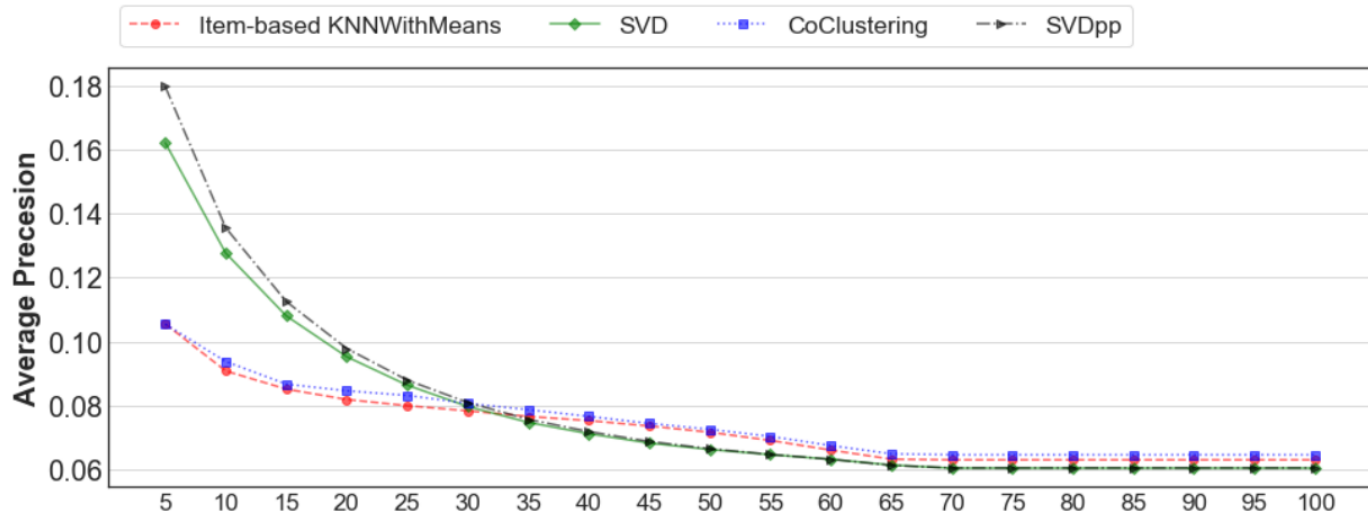




# 추천시스템

성능 평가

Precision & Recall

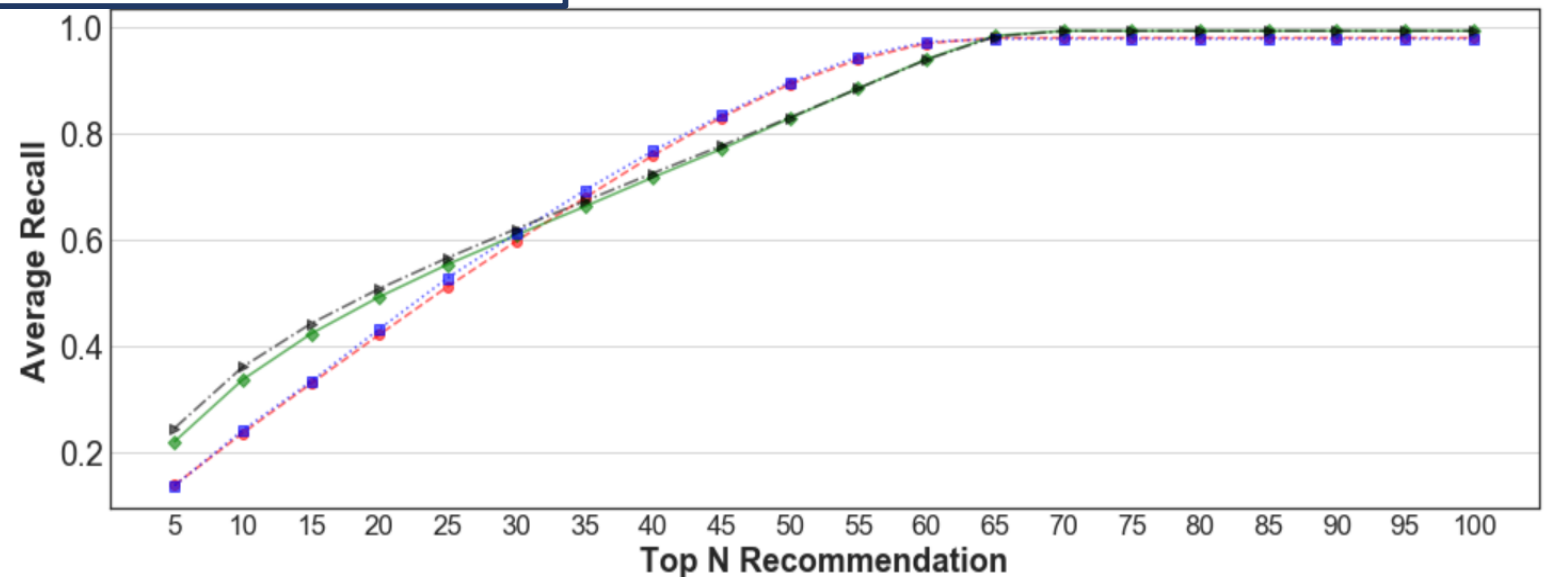


**Precision :**

a measure of exactness, determines the fraction of relevant items retrieved out of all items retrieved

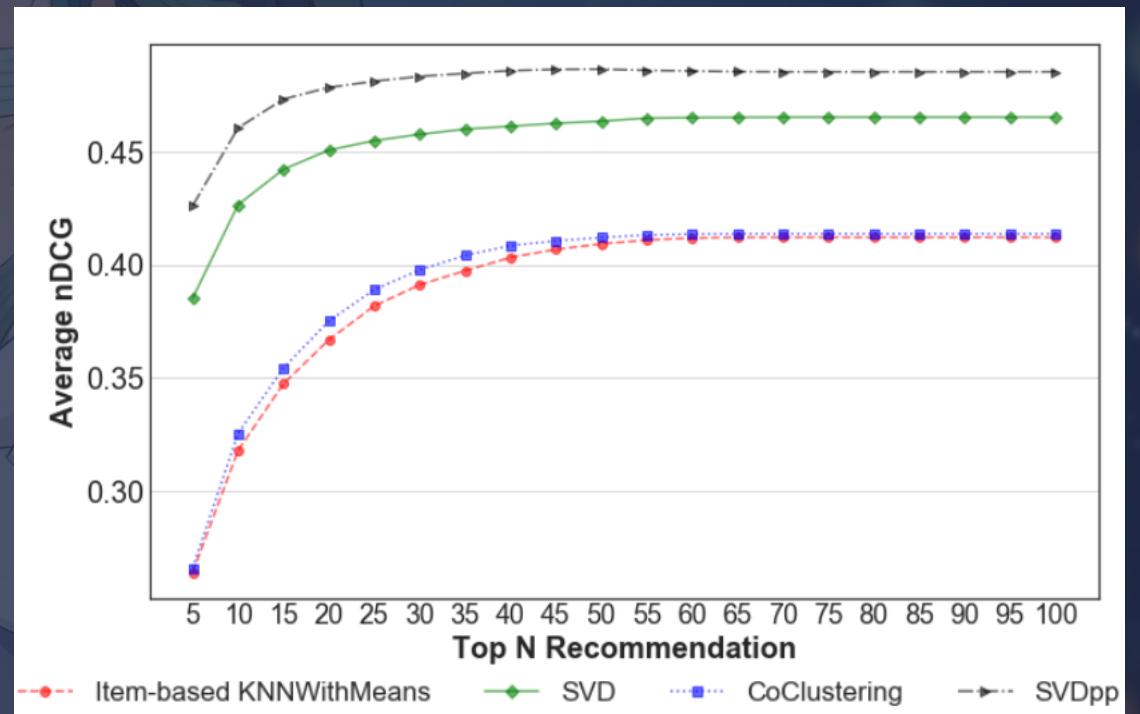
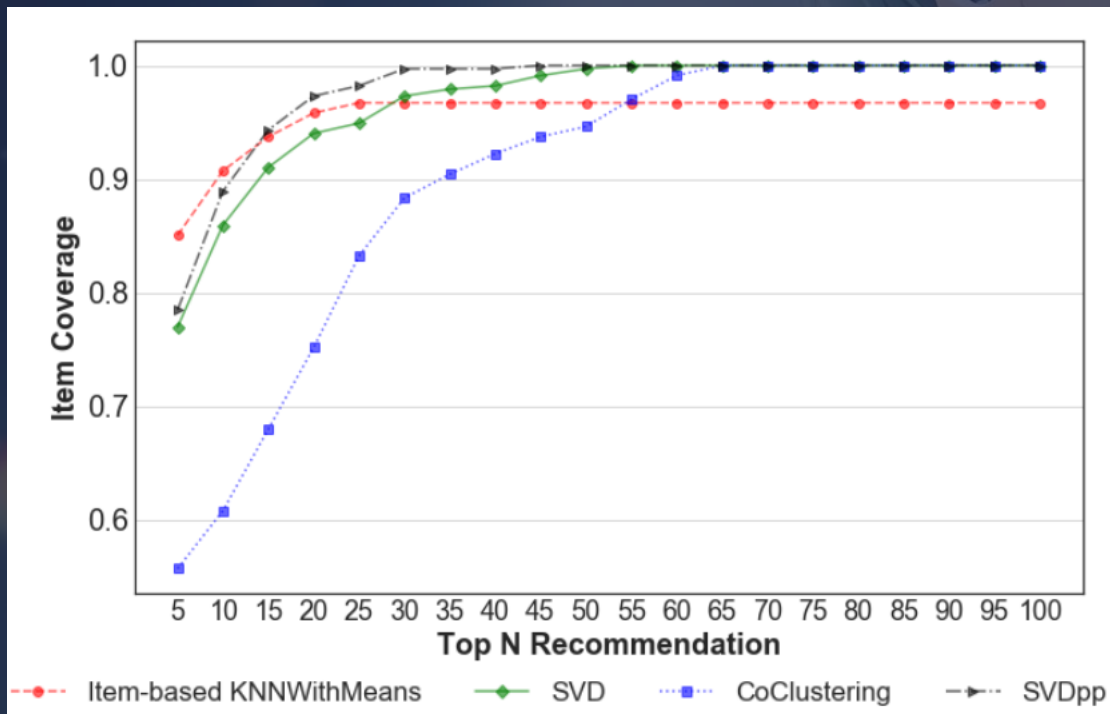
**Recall :**

a measure of completeness, determines the fraction of relevant items retrieved out of all relevant items



# 추천시스템 성능 평가

## Recommendation Quality Evaluation

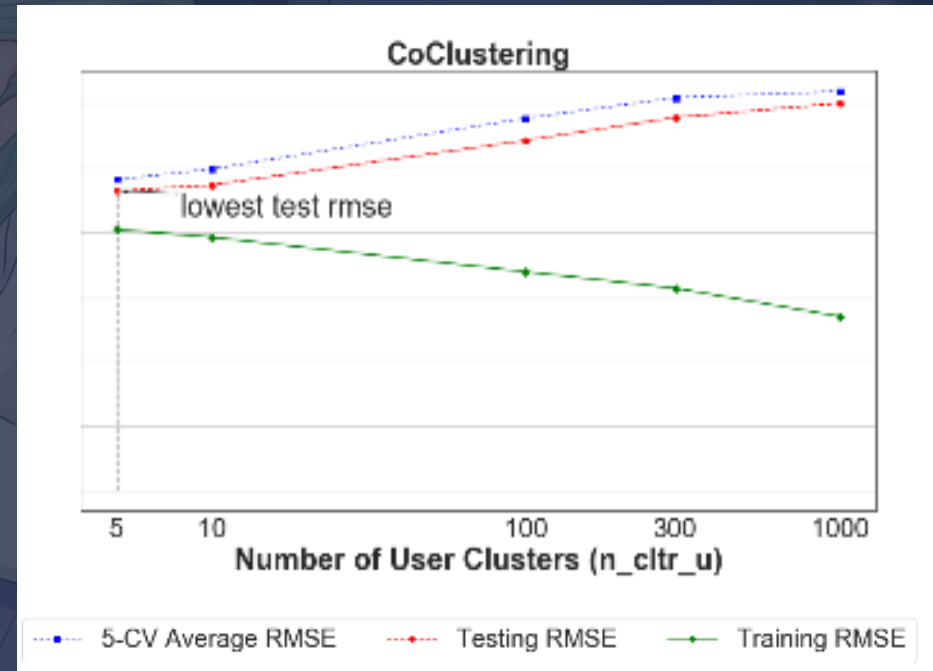
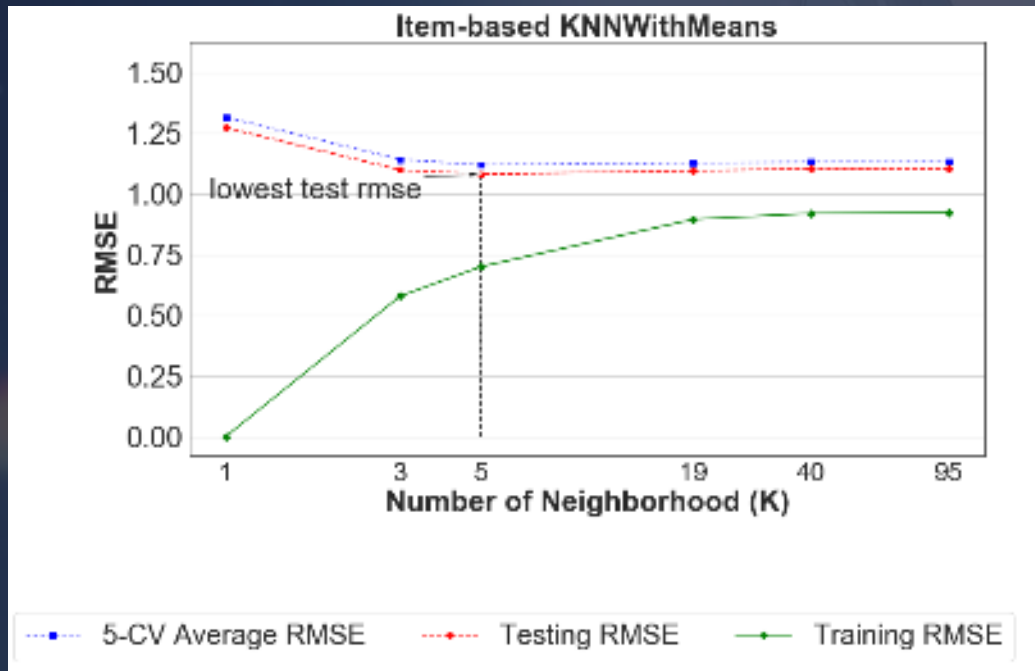




# 추천시스템 성능 평가

Knn

Co-clustering



# 추천시스템

Using R

Code

Output

```
install.packages("recommenderlab")
library(tidyverse)
library(recommenderlab)
library(stringr)
library(devtools)
data <- read.csv("rating.csv", header=T)
data_rating <- data[data$rating != -1,]
index <- sample(1:nrow(data_rating), 10000, replace=F)
data_rating <- data_rating[index,]

rating_df <- data_rating
rating_df <- rating_df %>%
  mutate(user_id = str_c("u", user_id),
         anime_id = str_c("m", anime_id))

rating_mat <- spread(rating_df, anime_id, rating) %>%
  remove_rownames() %>%
  column_to_rownames(var = "user_id")

rating_rrm <- as(as(rating_mat, "matrix"), "realRatingMatrix")

rating_rrm <- rating_rrm[rowCounts(rating_rrm) > 1,
                        colCounts(rating_rrm) > 0]

> print(object.size(rating_mat), units = "auto")
84.9 Mb
> print(object.size(rating_rrm), units = "auto")
302.4 kb

rating_eval <- evaluationScheme(rating_rrm, method="split", train=0.7, given=1)

ubcf_rmse <- Recommender(getData(rating_eval, "train"), method = "UBCF",
                        param=list(normalize = "center", method="Cosine", nn=5))

ubcf_pred <- predict(ubcf_rmse, getData(rating_eval, "known"), type="ratings")
calcPredictionAccuracy(ubcf_pred, getData(rating_eval, "unknown"))

ubcf_pred <- predict(object = ubcf_rmse, newdata = rating_rrm, n = 5)

recc_matrix <- sapply(ubcf_pred@items, function(x){
  colnames(rating_rrm)[x]
})
```

```
> recc_matrix[c(3,13,333)]
$`u10419`
[1] "m9253" "m29803" "m1" "m100" "m10012"

$u10903
[1] "m1943" "m26055" "m969" "m2449" "m1"

$u26192
[1] "m4985" "m10119" "m9260" "m1" "m100"
```

- Steins;Gate / Carnival Phantasm / Overload / Cowboy Bebop / Shin Shirayuki-hime  
Densetsu Pritear

- Paprika / JoJo no Kimyou na Bouken: Stardust Crusaders 2nd Season / Tsubasa  
Chronicle 2nd Season / Ghost in the Shell: Stand Alone Complex - The Laughing  
Man / Overload

- Mahou Shoujo Lyrical Nanoha: The Movie 1st / Seitokai Yakuindomo  
OVA / Kizumonogatari I: Tekketsu-hen / Overload / Cowboy Bebop



# INDEX

추천 시스템  
주제 선정 이유

추천 결과  
성능 평가

MyAnime Data  
Data EDA

한계점  
참고자료

# 참고

<https://www.youtube.com/playlist?list=PLwQdCtnouPk6rOKwGFs-KOzS8P4qIdJzY>

AI PM [5]: 추천시스템  
동영상 6개 · 조회수 245회 · 최종 업데이트: 2018. 12. 5.  
ALINA AI  
구독 143

- 1 AI PM [5.01]: 추천시스템 - 추천시스템 개요 ALINA AI 6:15
- 2 AI PM [5.02]: 추천시스템 - 내용기반추천 ALINA AI 15:42
- 3 AI PM [5.03]: 추천시스템 - 협업필터링 개요 ALINA AI 7:45
- 4 AI PM [5.04]: 추천시스템 - 협업필터링 실습1 ALINA AI 12:59
- 5 AI PM [5.05]: 추천시스템 - 협업필터링 실습2 ALINA AI 9:42
- 6 AI PM [5.06]: 추천시스템 - 모델기반 협업 필터링 및 추천 시스템 평가 방법 ALINA AI 14:54

[토크ON세미나] 빅데이터 이해 및 Jupyter Notebook 개요 1강 | T아카데미  
조회수 2,075회  
SKplanet Tacademy  
게시일: 2018. 4. 12.  
구독 1.3만

Python을 활용한 데이터분석 기초 | T아카데미  
SKplanet Tacademy - 1 / 3

- 1 [토크ON세미나] 빅데이터 이해 및 Jupyter Notebook 개요 1강 | T아카데미 SKplanet Tacademy 12:35
- 2 [토크ON세미나] Python을 통한 데이터 처리 (pandas) 2강 | T아카데미 SKplanet Tacademy 1:01:08
- 3 [토크ON세미나] Python을 통한 데이터 시각화 (matplotlib/seaborn) 3강 | T아카데미 SKplanet Tacademy 45:16

<https://www.youtube.com/watch?v=Ulik1Thn5YE&list=PL9mhQYIIEhdnNrVs9MVLramrRQ8jqis>

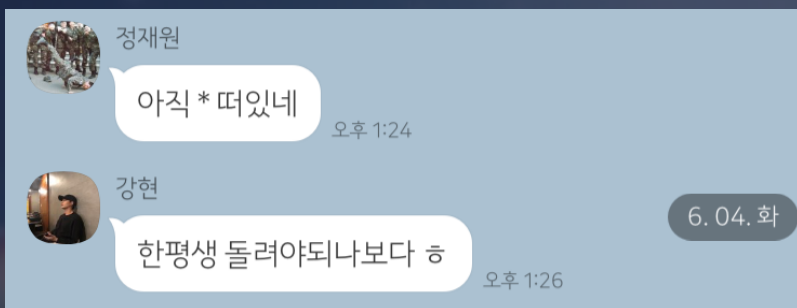
도움을 주신 많은 논문과,,  
블로거/엔지니어 분들 감사합니다^^,,

[https://github.com/KU-BIG/recommendation\\_system](https://github.com/KU-BIG/recommendation_system)

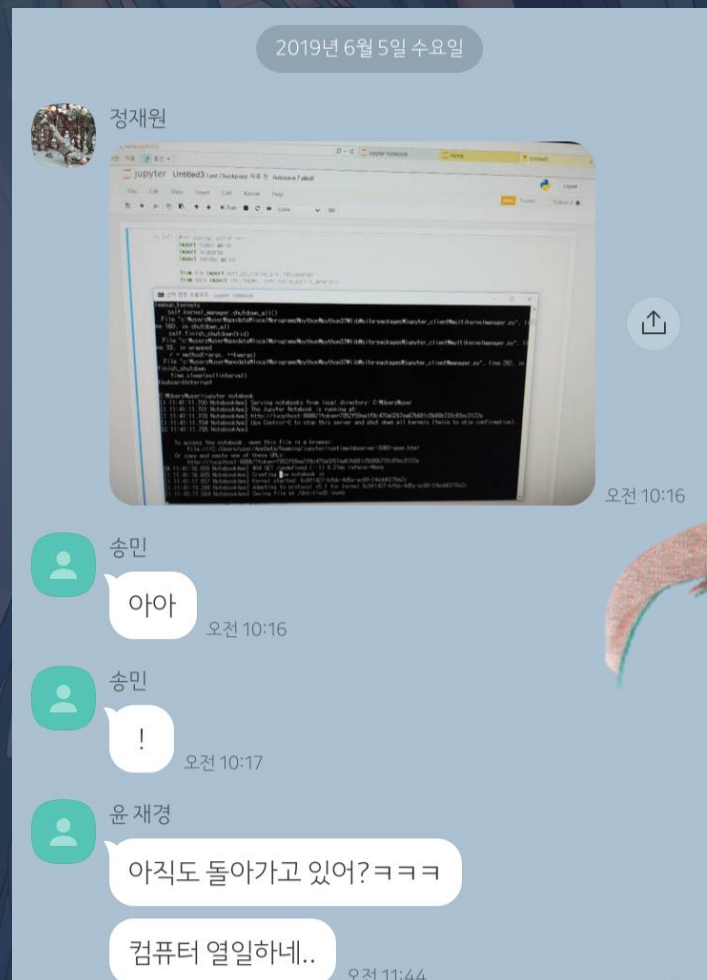


# 한계점

통컴실.. 열일..



시간이 너무 오래 걸려요.ㅠ





안녕하살범!  
3월!





첫 모임



=

마지막 모임





한학기 동안 모두 수고 많으셨습니다!