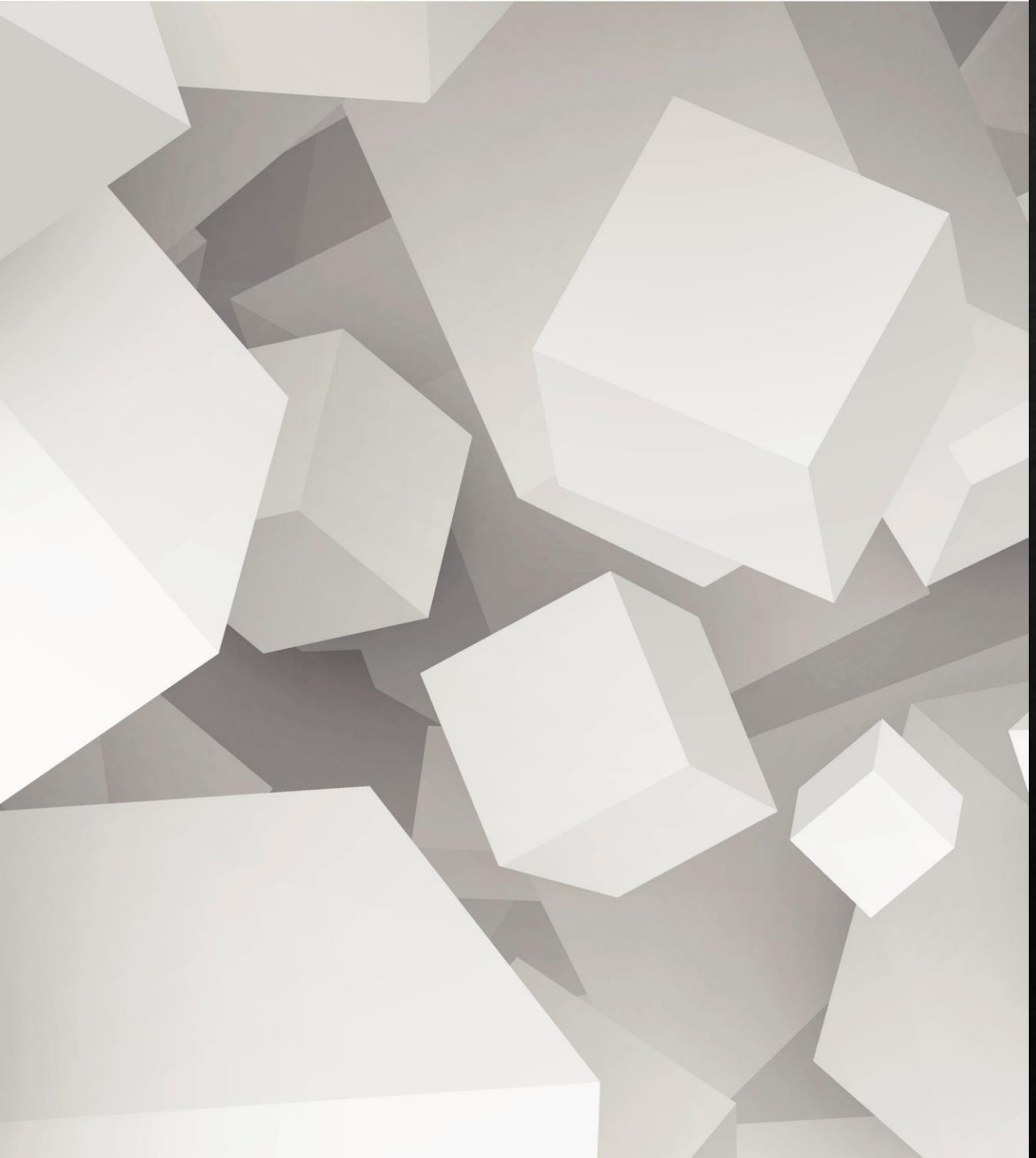
# Data Mining Final Project : Anime Recommendation System

Vinay Chandra Makineni (vimakin@iu.edu)

Navakanth Boyina (nboyina@iu.edu)

Varun Kumar Tangudu (vtangudu@iu.edu)



# Contents

# Team Profile

Vinay Chandra Makineni – vimakin@iu.edu

Navakanth Boyina – nboyina@iu.edu

Varun  Kumar Tangudu – vtangudu@iu.edu

# Abstract

While watching the anime many people want to watch the best story content/genre animes to get best experience and to not get disappointment after watching the animes. But without taking risk we may not know whether the anime is good to watch or not. Unfortunately, No one knows the best anime to watch without the user ratings and most watched opinion by the Users.

Anime Recommendation helps to broaden intrusion for the users by providing a positive and negative votes to recommend the anime based on genre, to choose best anime titles. In order to make sure the anime recommendation system makes use of a variety of alternative data~including user rating and members who watched the anime~to predict the ranking of the anime based on the genre. We perform exploratory data analysis on the given datasets and train the data using clustering algorithms and different recommendation models. Based on some specific metrics, we then decide which model is the best among all.

# Problem Statement

Everyone who wants to watch a new anime series or shows they are confused to decide which anime to start with based on their genre/category.

So, by using Anime recommendation system they can get recommendations based on the following :

| Genre | User rating / Votes |
| --- | --- |

We would like to solve this problem using Anime recommendation system

# Tasks Done in Project

## Data Pre-processing

> We took anime/rating datasets from Kaggle and then we have pre-processed the data by removing null/empty values and removing/filtering the data.

> We then cleaned the data and then removed the duplicate values from the dataset.

## EDA

> Generate charts

> Correlation plots

> Drop Columns

# Data Pre-processing

- We have removed all the punctuations and all unwanted characters, numbers, etc. from the column ‘name’.

data cleaning

```
def text_clean(text):
    text = re.sub(r'&quot;', '', text)
    text = re.sub(r'.hack//', '', text)
    text = re.sub(r'I&#039;', 'I\'', text)
    text = re.sub(r'&amp;', 'and', text)
    text = re.sub(r'Â°', '',text)
    text = re.sub(r'&#039;', '', text)
    text = re.sub(r'A&#039;s', '', text)
    return text
```

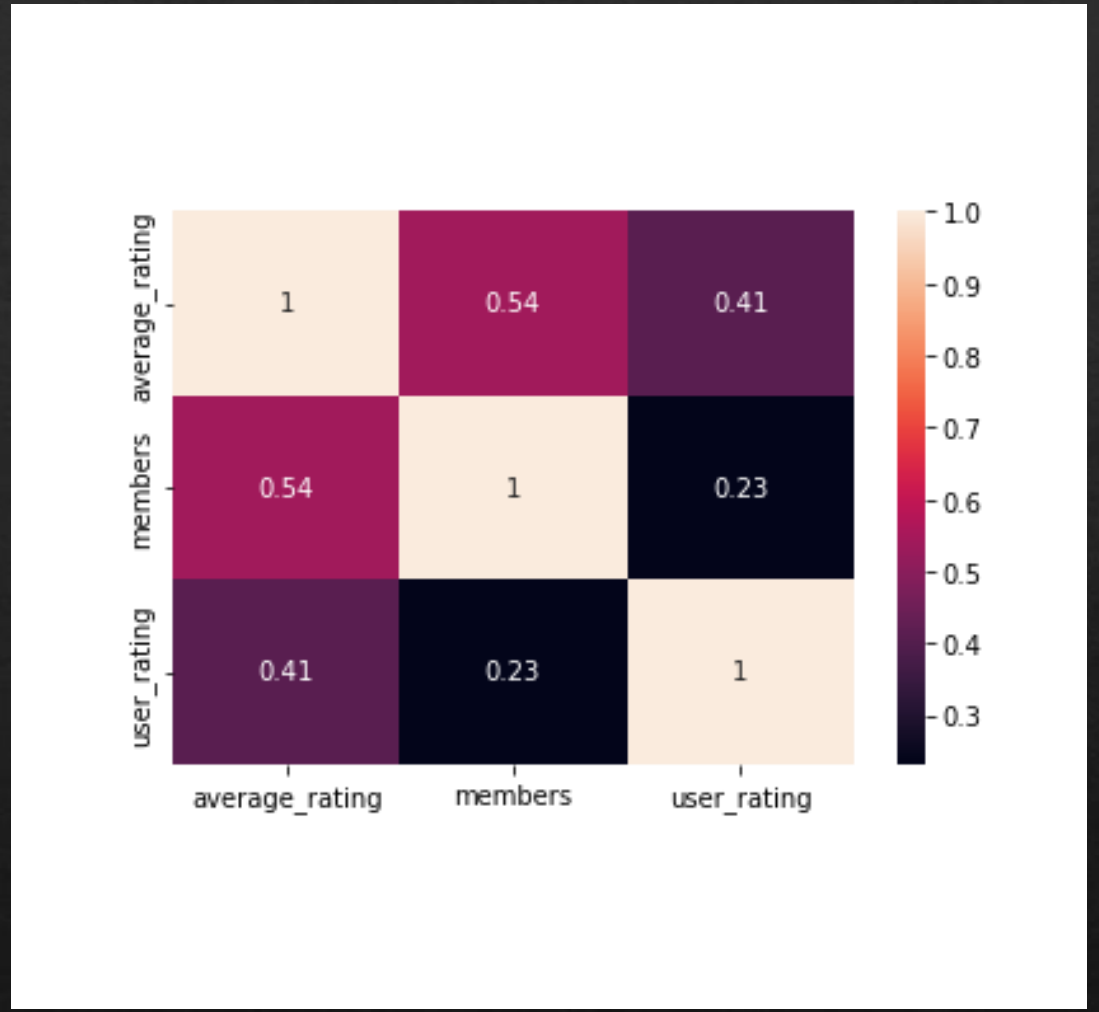
```
anime['name']=anime['name'].apply(text_clean)
```

```
anime['name']
```

```
0                                                Kimi no Na wa.
1                          Fullmetal Alchemist: Brotherhood
2                                                 Gintama°
3                                               Steins;Gate
4                                                    Gintama
                                  ...
12289          Toushindai My Lover: Minami tai Mecha-Minami
12290                                            Under World
12291                            Violence Gekiga David no Hoshi
12292    Violence Gekiga Shin David no Hoshi: Inma Dens...
12293                          Yasuji no Pornorama: Yacchimae!!
Name: name, Length: 12294, dtype: object
```

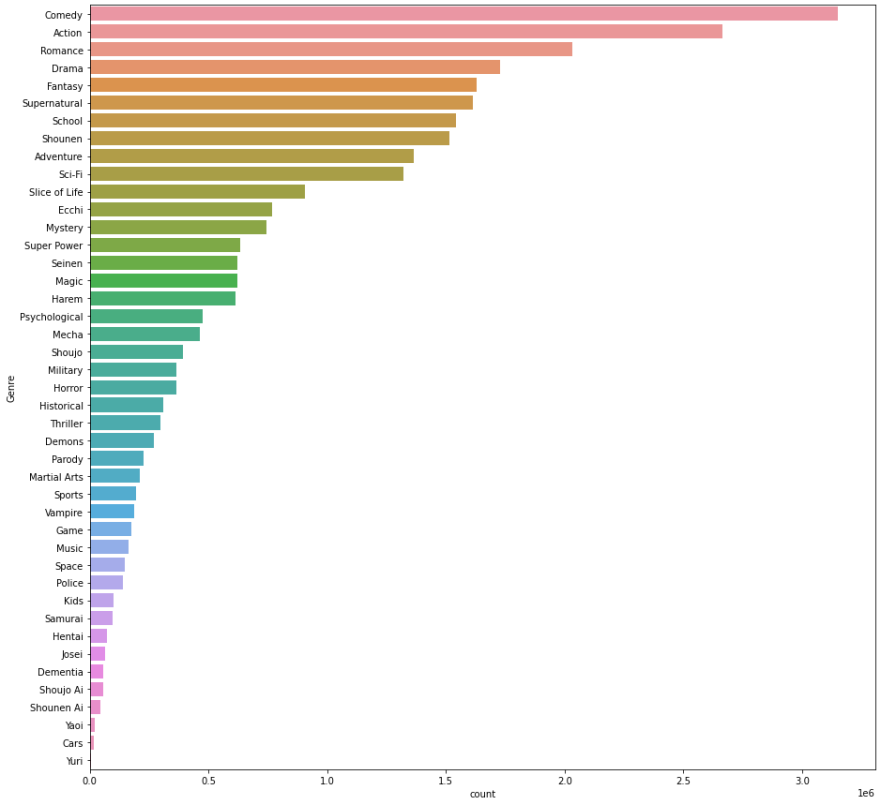
# EDA

- Correlation
  - We have done the correlation for user rating and members
  - There is no strong correlation between any attributes.



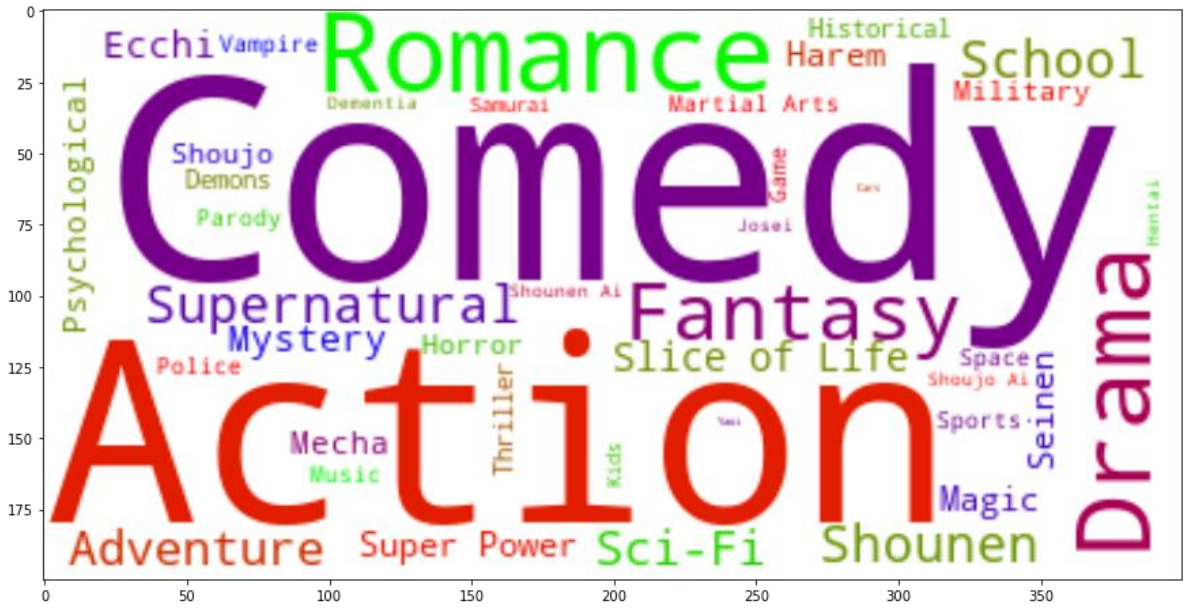
# EDA

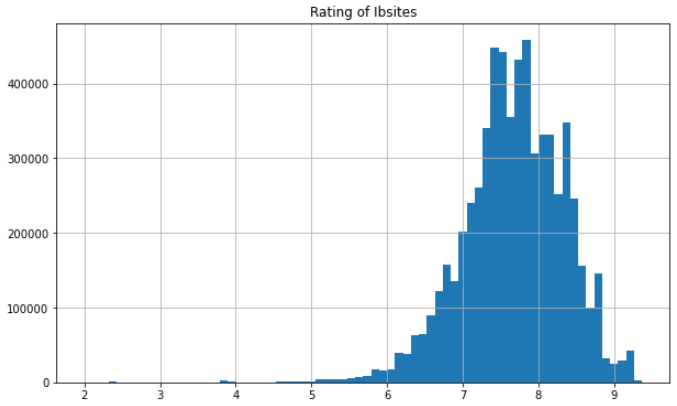
- The bar plot indicates the Count of animes Vs Genres, In the descending order.
- It gives a clear idea on count of animes based on genres.
- We have observed that Comedy has highest count and Yuri has lowest count

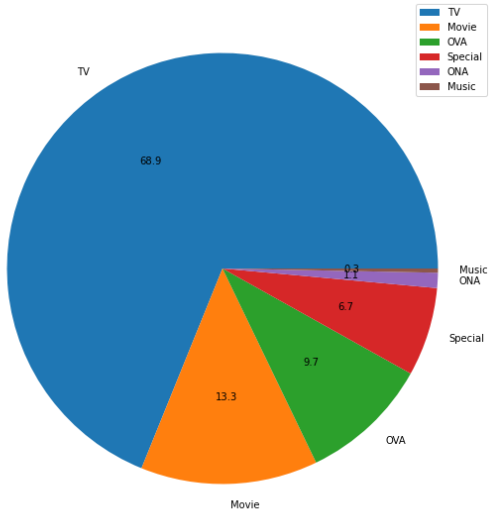


# EDA

- The following word cloud shows the anime genres based on count in the dataset.
- We have observed that Comedy, Action, Romance, Drama, Fantasy are having high volume than others.





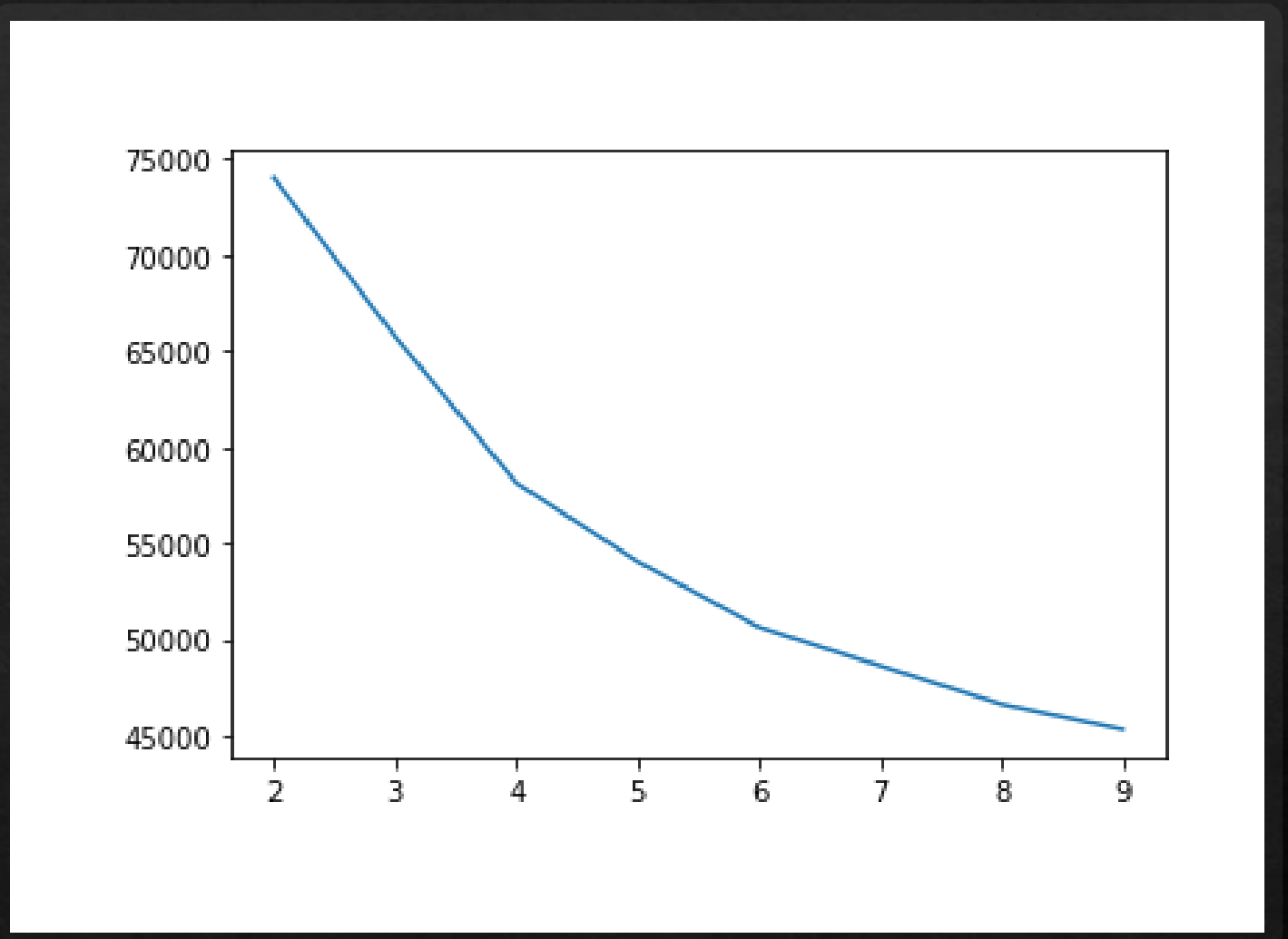




# EDA

- The histogram represents the average rating of animes based on the member ratings.
  - we can see that most ratings are from 6 to 10
- The Pie chart represents the percentage of animes based on anime ‘types’.
  - We have observed that TV has more percentage than other ‘types’, which is 68.9% and we have seen ONA and Music has 0.3%,1.1% respectively.

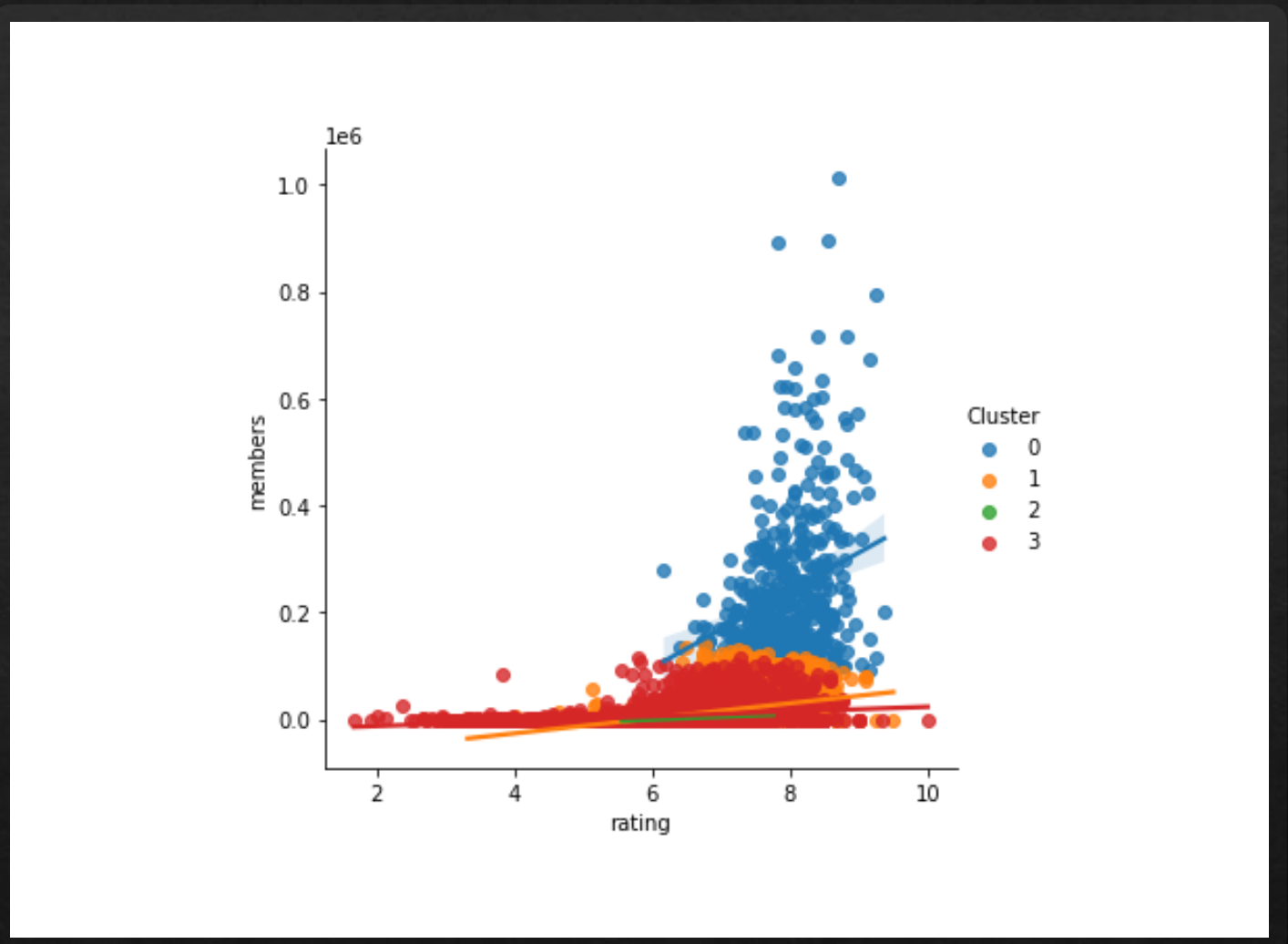
# Clustering

- We have taken K-means clustering and we can see an elbow in the plot for K = 4 from the graph above. The SSD decreases significantly before this elbow point and steadily after K = 4.
- So, we have taken 4 clusters.



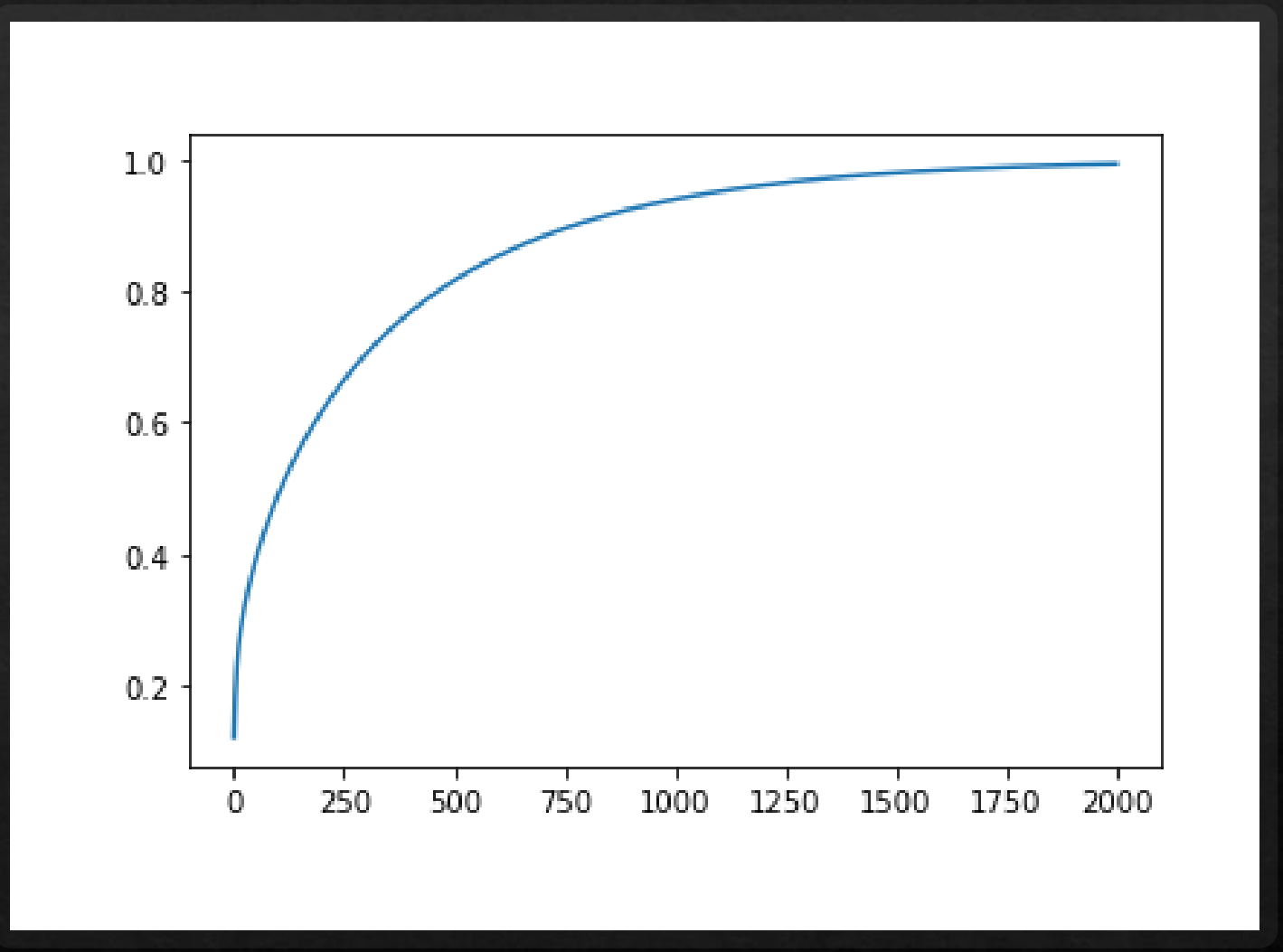
# Clustering

- Rating vs Members clustering
- So, K=4 we have 4 clusters based on that, the Sum of squared differences (SSD) decreases significantly after k=4
- The legend shows the label numbers of clusters



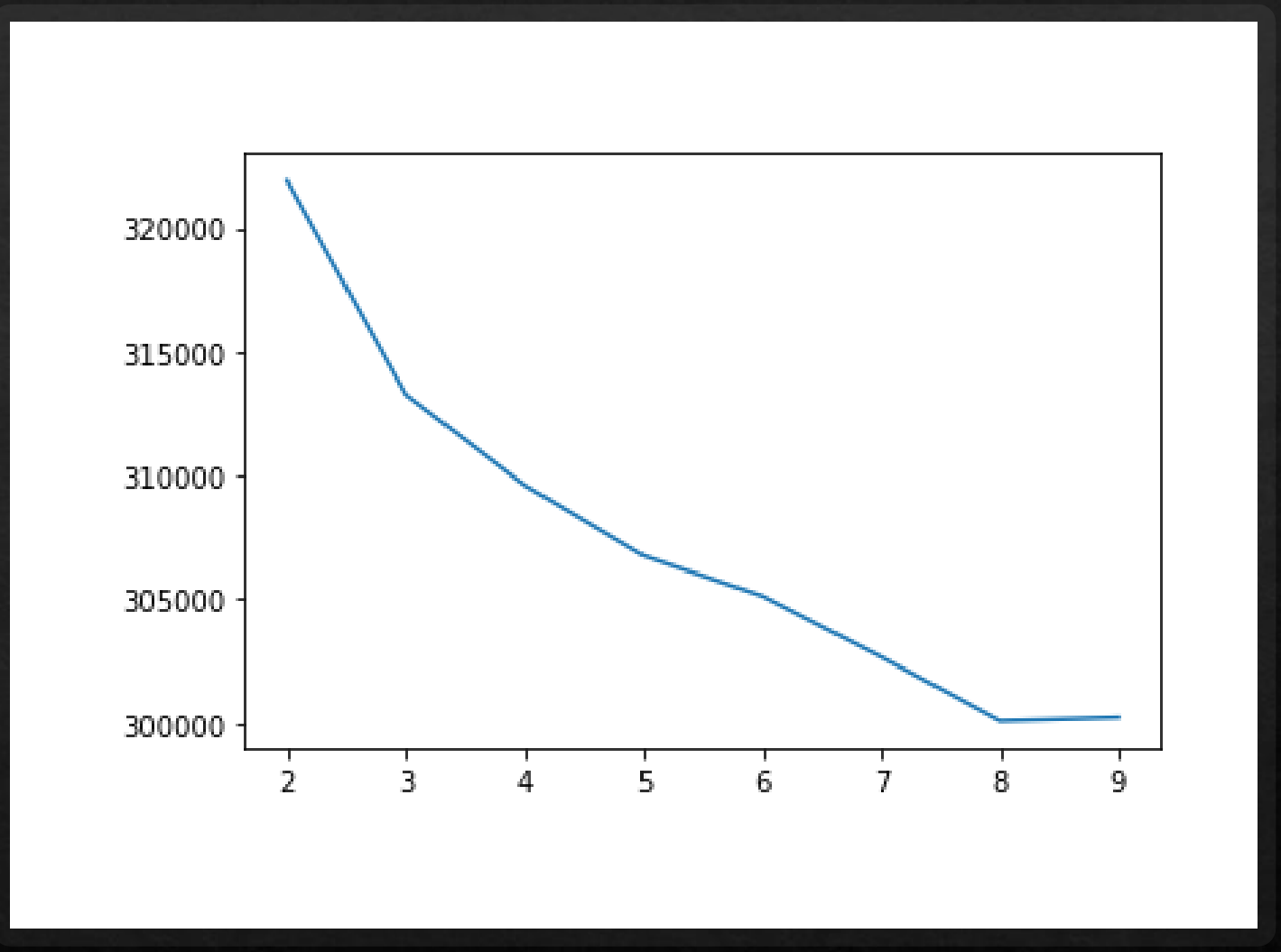
# Dimensionality Reduction using PCA

- The variance of the principal component is 1500 around one.
- So, we  take n_components value as 1500 and apply PCA again



# Clustering after applying PCA

- After PCA the elbow point is at 3.
- From point  3 the values gradually decreases
- So, we have taken n_clusters = 3
- We can also observe the cluster values for each cluster separately in the notebook clearly



# Content Based Recommendation

- This model suggest similar items based on a particular item. This system uses item metadata, such as genre, director, description, actors, etc. for movies, to make these recommendations. The general idea behind these recommender systems is that if a person likes a particular item, he or she will also like an item that is similar to it. And to recommend that, it will make use of the user's past item metadata.
- Cosine Similarity
  - You will be using the cosine similarity to calculate a numeric quantity that denotes the similarity between two movies. You use the cosine similarity score since it is independent of magnitude and is relatively easy and fast to calculate (especially when used in conjunction with TF-IDF scores, which will be explained later).

# Collaborative filter Recommendation

- This model is widely used, and they try to predict the rating or preference that a user would give an item-based on past ratings and preferences of other users.
- Collaborative filters do not require item metadata like its content-based counterparts.
- Types of collaborative Filtering
  - memory based (user based, item based)
  - Model based
  - Hybrid

# Results : Content Based Recommendation

- We have observed that the outputs for anime title ‘Gintama’ we have suggestions related to the title with same different types related to the anime title with same genre.
- But for anime title ‘Fullmetal Alchemist: Brotherhood’ we have suggestions with different types and titles with same genre.

```
recommend_content('Gintama').head()
```

| name | similar_val |
|---|---|
| Gintama° | 1.000000 |
| Gintama | 1.000000 |
| Gintama Movie: Kanketsu-hen - Yorozuya yo Eien Nare | 0.798927 |
| Gintama Movie: Shinyaku Benizakura-hen | 0.798927 |
| Gintama: Shinyaku Benizakura-hen | 0.778725 |

```
recommend_content('Fullmetal Alchemist: Brotherhood')
```

| name | similar_val |
|---|---|
| Fullmetal Alchemist: Brotherhood | 1.000000 |
| Fullmetal Alchemist: The Sacred Star of Milos | 0.525534 |
| Fullmetal Alchemist: Brotherhood Specials | 0.505807 |
| Fullmetal Alchemist | 0.486249 |
| Arion | 0.393410 |
| ... | ... |
| Kaitou Tenshi Twin Angel: Kyun Kyun☆Tokimeki Paradise!! | 0.000000 |
| Himegoto | 0.000000 |
| Girlfriend (Kari) | 0.000000 |
| Forsaken | 0.000000 |
| Yasuji no Pornorama: Yacchimae!! | 0.000000 |

11823 rows × 1 columns

# Results : Collaborative filter recommendation using memory based

```
recommender2=recommender.copy()
for i in range(0, 6):
    animes=pd.DataFrame(rating_anime_pivot.index).reset_index()
    animes=animes.rename(columns={'index':f'anime{i}'})
    recommender2=pd.merge(recommender2,animes,on=[f'anime{i}'],how='left')
    recommender2=recommender2.drop(f'anime{i}',axis=1)
    recommender2=recommender2.rename(columns={'name':f'anime{i}'})
```

```
recommender2.head()
```

| | anime0 | anime1 | anime2 | anime3 | anime4 | anime5 |
|---|---|---|---|---|---|---|
| 0 | 0 | Oosouji | Rusty Nail | Decorator | Doudou | Nebula feat. Hatsune Miku |
| 1 | 001 | Gakuen Senki Muryou | Ki Renka | Suna Asobi | Ame no Bus Stop-hen | Minihams no Ai no Uta |
| 2 | 009 Re:Cyborg | Cyborg 009 | Terra e... | Mardock Scramble: The First Compression | Arve Rezzle: Kikaijikake no Yousei-tachi | Toaru Hikuushi e no Tsuioku |
| 3 | 009-1 | 009-1: RandB | AIKa | Choujuushin Gravion | Gun Frontier | AIKa Zero |
| 4 | 009-1: RandB | 009-1 | Glass no Kantai: La Legende du Vent de lUniver... | G-Taste (2010) | AIKa Zero | AIKa |

- The table shows the 5 animes that are like each other.

# Conclusion & Future work

- We have done content-based recommendation model and memory-based model in collaborative filter recommendation. Further we are going to implement further Model based, Hybrid based Recommendation models and Association Analysis(based on type and genre)
- After we get the future results of all these recommendation models mentioned above, we then choose best Anime Recommendation Model.

# Reference links


- https://www.kdnuggets.com/2020/07/building-content-based-book-recommendation-engine.html#:~:text=Content%2Dbased%20recommendation%20system,products%20based%20on%20their%20descriptions
- https://www.analyticsvidhya.com/blog/2019/08/5-applications-singular-value-decomposition-svd-data-science/
- https://towardsdatascience.com/how-does-collaborative-filtering-work-da56ea94e331
- https://www.researchgate.net/publication/342690182_Collaborative_Recommendation_System_in_Users_of_Anime_Films