IBM-312 Group Assignment
Group 18

Archit Isham (20312008) Uddipt Gupta (20312038)

Manas Misra (20312019) Samyak Wanjarwadkar (20312030)

November 29, 2022

Figure 1: Animes

1 Motivation of the Problem

Anime is a hand-drawn computer animation originating from Japan which has gained a cult following around the world, especially seeing a massive boom in following in the past decade. The Japan External Trade Organization has valued the industry's overseas sales to 18 billion (5.2 billion for the US alone) in 2004. This has definitely grown and has the potential to grow even further, especially during this time in the world.
Since the anime culture boom has given rise to lots of new industries, it also gives rise to loads of new opportunities. As the variety of anime increase, we get back to the age old problem of human indecisiveness, what to choose and what to discard. In an attempt to solve this problem, we try to predict the anime a viewer will like based on their reviews of the anime they have seen in the past.
Therefore, in this assignment, we try to recommend to the user, new animes based on their review of the anime they have seen before using their MAL(my-anime-list) profile.

2 Snapshot of the Data

The project uses 2 data files, namely anime.csv data file and rating.csv data file.
The anime data file consists of 6 columns - anime id, Name, Genres, type (movie/series), Episodes and Score (average rating by users).
The rating anime files contains 3 columns - anime id, user id and rating.

This files tells us about the rating given by each user to each anime.

Here is link to the dataset used : G-18-datasets.

The data has been taken from : Dataset Source

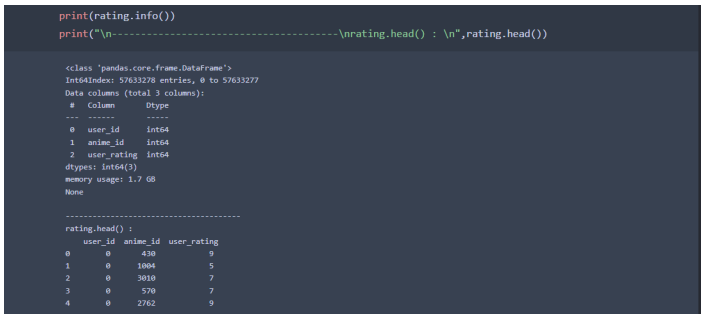
We merge the 2 data files using JOIN clause taking anime id as foreign key. The merged data set is the final data set which we work on. The final data set, is represented in tabular form, denoted by pivot table. The pivot table has 9076 columns, each representing a unique anime. Each row represents a user, and the data values in the table are the ratings given by the user to the anime.

```
In [143]: anime = pd.read_csv('anime.csv')
          rating = pd.read_csv('rating.csv')

In [144]: anime = anime.rename(columns={"MAL_ID": "anime_id"})
          anime = anime[['anime_id','Name','Score','Genres','Type','Episodes']]
          anime.head()
```

	anime_id	Name	Score	Genres	Type	Episodes
0	1	Cowboy Bebop	8.78	Action, Adventure, Comedy, Drama, Sci-Fi, Space	TV	26
1	5	Cowboy Bebop: Tengoku no Tobira	8.39	Action, Drama, Mystery, Sci-Fi, Space	Movie	1
2	6	Trigun	8.24	Action, Sci-Fi, Adventure, Comedy, Drama, Shounen	TV	26
3	7	Witch Hunter Robin	7.27	Action, Mystery, Police, Supernatural, Drama, ...	TV	26
4	8	Bouken Ou Beet	6.98	Adventure, Fantasy, Shounen, Supernatural	TV	52

Figure 2: original anime data snapshot



```
print(rating.info())
print("\n--------------------------------------\nrating.head() : \n",rating.head())
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 57633278 entries, 0 to 57633277
Data columns (total 3 columns):
 #   Column       Dtype
---  ------       -----
 0   user_id      int64
 1   anime_id     int64
 2   user_rating  int64
dtypes: int64(3)
memory usage: 1.7 GB
None

--------------------------------------
rating.head() :
    user_id  anime_id  user_rating
0        0       430            9
1        0      1004            5
2        0      3010            7
3        0       570            7
4        0      2762            9
```

Figure 3: rating data snapshot

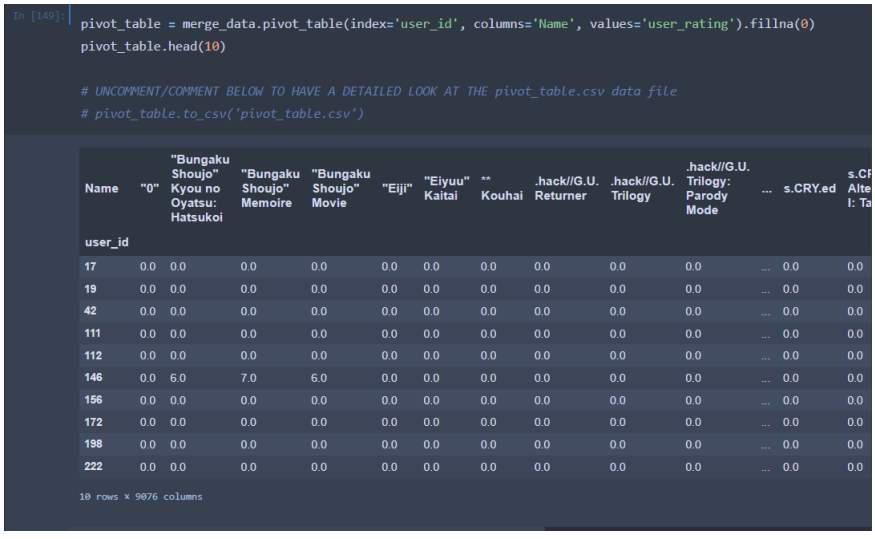
3 Methodology

3.1 Principal Component Analysis

The pivot table data set is a large data set. Analysis of data using a data set of this size would be very time consuming, which would end making the recommendation system very slow.
The first task in the process is to build a recommendation system to represent data in more memory and time efficient way. We decided to use PCA for dimension reduction for this.
Using PCA, we reduce the dimension of the data set to 3 dimensions. **This means that each of the new column now represents a combination of anime(principal anime). Since each of the columns are mutually orthogonal to each other, each of these 3 'principal' anime are different from each other as they can be, ideally representing a wide range of anime, and explaining a lot of variance of the original data. Each user is being represented in terms of his rating of the 3 principal anime.**

3.2 K-Means Clustering

Users can be assumed to belong to distinct clusters, each of the cluster having their own tastes in terms of length, genre and ratings.



```
In [149]: pivot_table = merge_data.pivot_table(index='user_id', columns='Name', values='user_rating').fillna(0)
          pivot_table.head(10)

          # UNCOMMENT/COMMENT BELOW TO HAVE A DETAILED LOOK AT THE pivot_table.csv data file
          # pivot_table.to_csv('pivot_table.csv')
```

Name	"0"	"Bungaku Shoujo" Kyou no Oyatsu: Hatsukoi	"Bungaku Shoujo" Memoire	"Bungaku Shoujo" Movie	"Eiji"	"Eiyuu" Kaitai	** Kouhai	.hack//G.U. Returner	.hack//G.U. Trilogy	.hack//G.U. Trilogy: Parody Mode	...	s.CRY.ed	s.CR Alte I: Ta
user_id													
17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
42	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
111	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
112	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
146	0.0	6.0	7.0	6.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
156	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
172	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
198	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
222	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0

10 rows x 9076 columns

Figure 4: pivot table

To recommend the user new anime, we need to group the users in different clusters. We use K-means Clustering for this task. Using both elbow method and Davies Bouldin index we find the optimal number of clusters for this system is 5.
Note that the clustering operation is being performed on the reduced dimension data set which we obtained from PCA.

3.2.1 Why not GMM?

Another way of Clustering the data set could have been GMM (Gaussian Mixture Modelling). GMM results in each point having a set of probabilities representing the chances of that point belonging to the set of clusters.
However, we have decided against using GMM in this model. Each anime generally spans multiple genres, hence, when performing clustering using Gaussian boundaries, the probability of lying in each cluster would be significant. Using GMM will therefore not give us very useful classifications.
K-Means clustering provides with concrete boundaries, and hence is a better clustering algorithm for this project.

4 Analysis

Now we have the users in the clusters. The final step is to identify and classify the taste of the cluster.

4.1 What does a cluster signify

Let us partition the pivot table into 5 tables, each table representing all the users belonging to that particular cluster. We also assume that the tastes of the users belonging to the same cluster are similar, hence mean of ratings would be a good measure of the popularity of a given anime in that cluster.
We now have the average ratings of all the anime for each cluster. A simple sort would reveal the favourite anime of that cluster.

4.2 Wordcloud Anaylsis

To illustrate what each cluster represents, we made the wordcloud of genre each cluster to represent what genres are contained insides each cluster. We notice that our wordclouds are very distinct from each other representing different tastes in anime. Although some of the words like

"Action" are common in each wordcloud, since most animes have action as a genre . However we notice from the wordclouds, that they are significantly different from each other. Hence we have succeeded in obtaining 5 different clusters of anime based on users' taste.

5 Result

MAL(myanimelist) does not expose the user id publicly. Hence making an end to end product is difficult to privacy issues. However, for demonstration purpose, we pick a random used id form the data set we already have.
The PCA followed by K means clustering algorithm will assign this user id to the cluster it should belong to. From the merged dataset, we also obtain the animes that the user has already watched.
Since we have partitions of the pivot table representing the charm of anime in the cluster, the algorithm would simply recommend the top animes from this list which have not yet been seen by the user.

Link to the Final Project : IBM312 G18 Project Folder.

Figure 5: Recommendation list to the user