ME781 Course-project Technical report: Movie recommendation tool

P.Mithun Balram
180100081

Table of contents

1)Conceptual Design

2)Interface and Output visualization

3)Unit testing report

4)Model training & testing

Conceptual Design-Demographic filtering

The main idea of demographic filtering is recommending top movies in general depending on ratings,popularity,in each genre,of top actors,directors etc. We have the TMDB 5000 dataset which has information about ratings,popularity,cast,crew,genres of top 5000 english movies.Demographic filtering doesn’t involve any ML models it is all about choosing the movies with top ratings,popularity,followed by top 3 movies in each genre,movies of top actors,directors(Top actors,directors are those who have been a part of maximum number of movies).So,demographic filtering is useful when you know almost nothing about the viewer this typically the case with new viewers.

Conceptual Design-Content-based filtering

In content based filtering , we have a viewer and his watch history i.e. a list of movies he watched and the ratings he gave to each of them , our goal is to give personalized recommendations based on his interests that we must infer from his watch history and corresponding ratings.I have pre-processed each movie to get the popularity,ratings,genres,cast,crew then I attempt to calculate similarity between any pair of movies based on one-one similarity measure eg. fraction of actors in a movie that were a part of some other movie,similarly for genre,director etc. then take a weighted similarity over cast,director and genres.Now,having established similarity measure between every pair of movies, for each unrated movie we compute similarity we each rated movie,and then get a weighted similarity using corresponding ratings as weights.Now,we choose the top 20 similar unrated or unwatched movies among them and the we choose 10 from the 20 based on popularity and ImdB ratings.(Dataset is TMDB 5000 similar to the one used for demographic filtering)

One-one similarity measure explained

Say a movie X has cast->[A,B,C]

Say another movie Y has cate->[A,H,L,Z]

Similarity of Y to X is the fraction of cast of X that worked in movie Y,

I.e. one-one similarity(X,Y)=$|X \cap Y|/|X|$

Note:This isn't same as one-one similarity(Y,X) thus,it isn't commutative

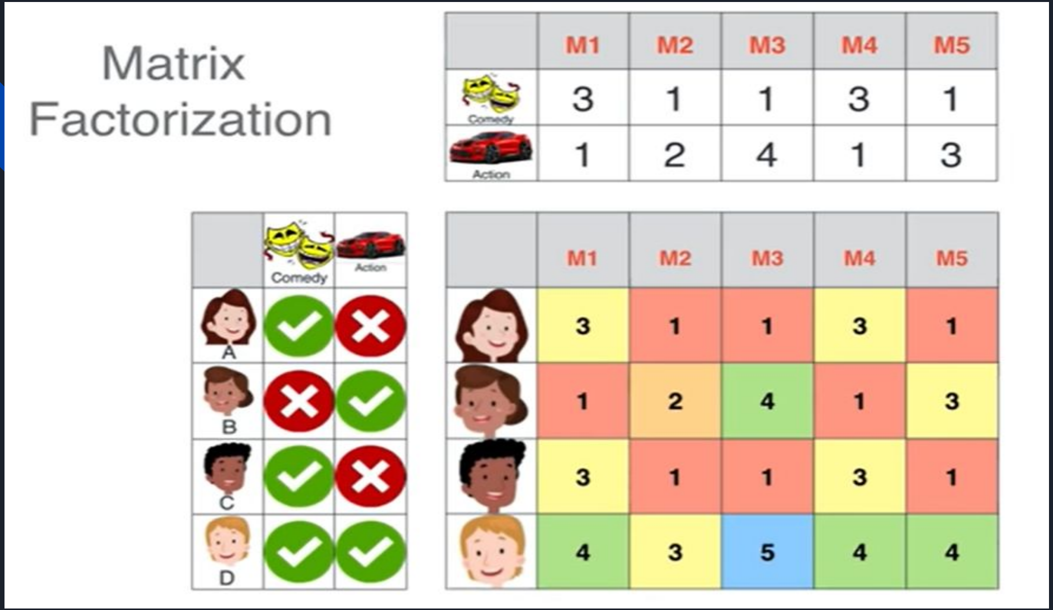
For each feature i.e. director,cast,genre we compute the one-one similarity and then take weighted similarity to get similarity between 2 movies(Take a look at the code of content-based filtering for a better idea)

Conceptual Design-Collaborative filtering

Collaborative filter works on the principle of finding similar viewers based on their watch history and ratings,finally recommend movies highly rated by a viewer to similar viewers.

We have the movieLens Dataset for collaborative filtering,this comprises of about 700 viewers and their ratings for the movies they watched.Let’s design a ratings matrix of viewer v/s movies where i,j th entry is the rating of the ith viewer to the jth movie(it is NA if the viewer hasn’t rated the movie).This is sparse matrix with only 2% entries filled,so we can perform SVD(Singular value Decomposition) such that the mean squared error with respect to the non-zero entries is minimized , now having performed SVD we can estimate each viewers rating to each movie and then recommend the top 10 movies with highest estimated rating to each viewer

Understanding Singular value decomposition



Matrix Factorization

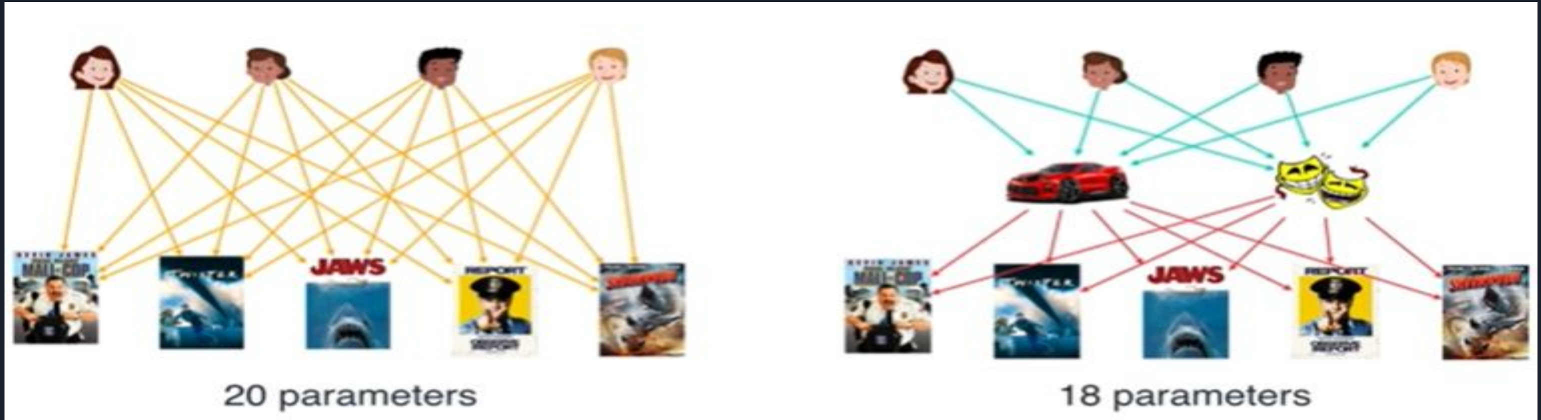
	M1	M2	M3	M4	M5
Comedy	3	1	1	3	1
Action	1	2	4	1	3

	Comedy	Action
A	✓	✗
B	✗	✓
C	✓	✗
D	✓	✓

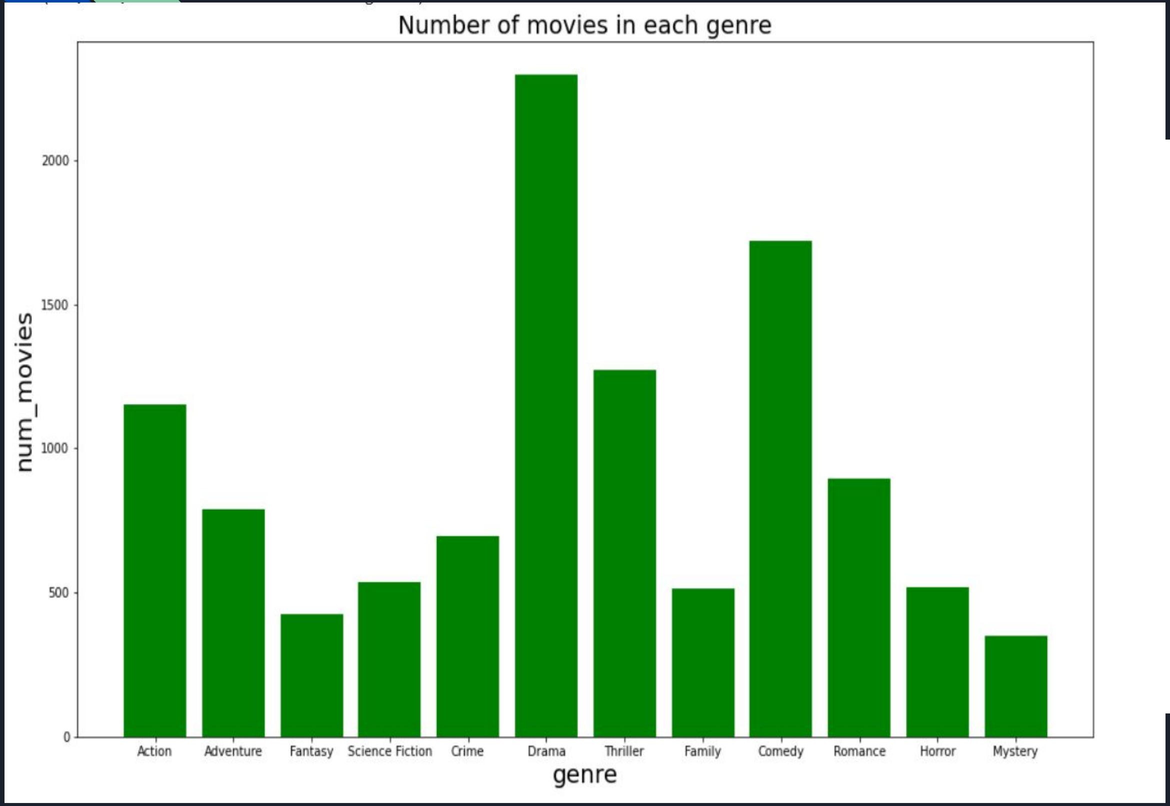
	M1	M2	M3	M4	M5
	3	1	1	3	1
	1	2	4	1	3
	3	1	1	3	1
	4	3	5	4	4

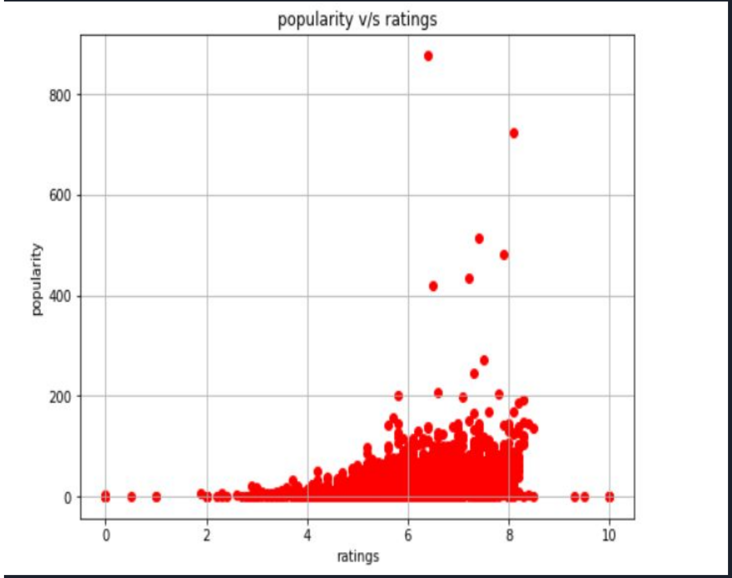
We can map a person to his interests say 1 part comedy and 2 parts action and say a movie has 2 parts comedy and 3 parts action then his estimated rating to the movies would be:$5*(\frac{1}{3}*\frac{2}{5}+\frac{2}{3}*\frac{3}{5})=2.67$

This factorization into intermediate features reduces the parameters and thus helps fill out the missing values i.e. estimate ratings



Demographic filtering visualization-TmDb-5000 data

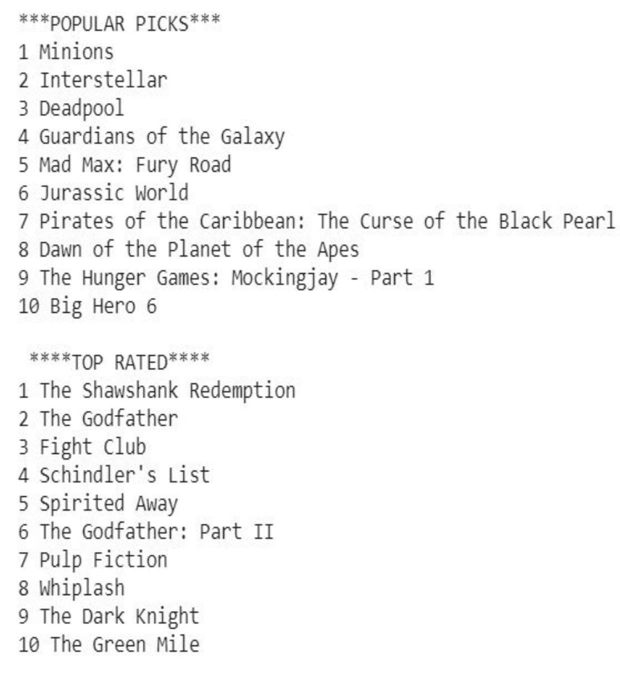




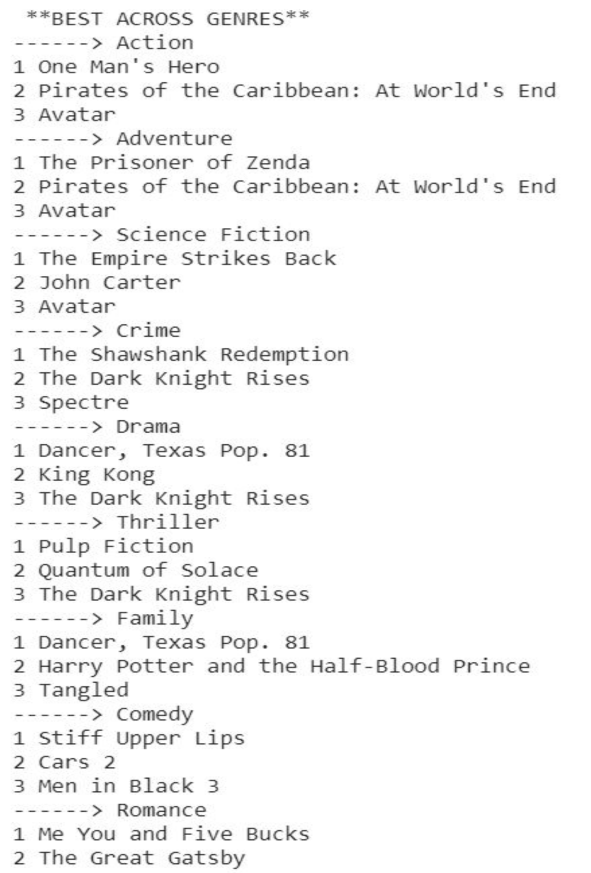
Popularity v/s rating correlation

Number of movies in each genre

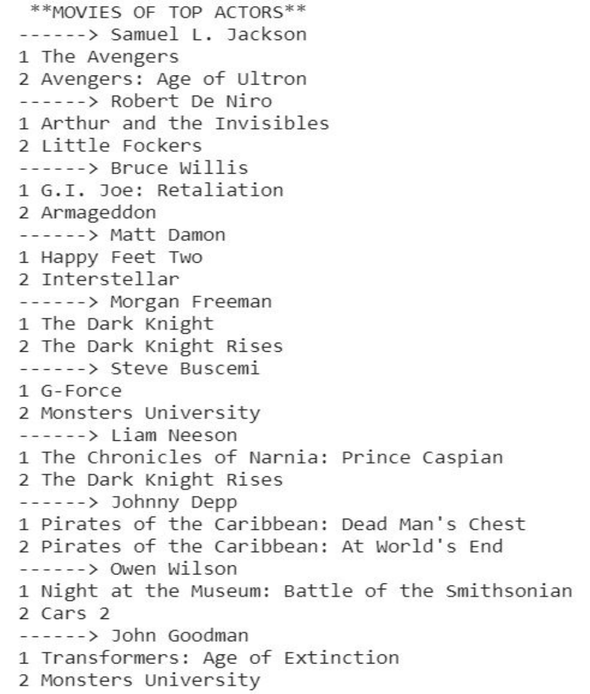
Demographic filtering output visualization



```
***POPULAR PICKS***
1 Minions
2 Interstellar
3 Deadpool
4 Guardians of the Galaxy
5 Mad Max: Fury Road
6 Jurassic World
7 Pirates of the Caribbean: The Curse of the Black Pearl
8 Dawn of the Planet of the Apes
9 The Hunger Games: Mockingjay - Part 1
10 Big Hero 6

 ****TOP RATED****
1 The Shawshank Redemption
2 The Godfather
3 Fight Club
4 Schindler's List
5 Spirited Away
6 The Godfather: Part II
7 Pulp Fiction
8 Whiplash
9 The Dark Knight
10 The Green Mile
```



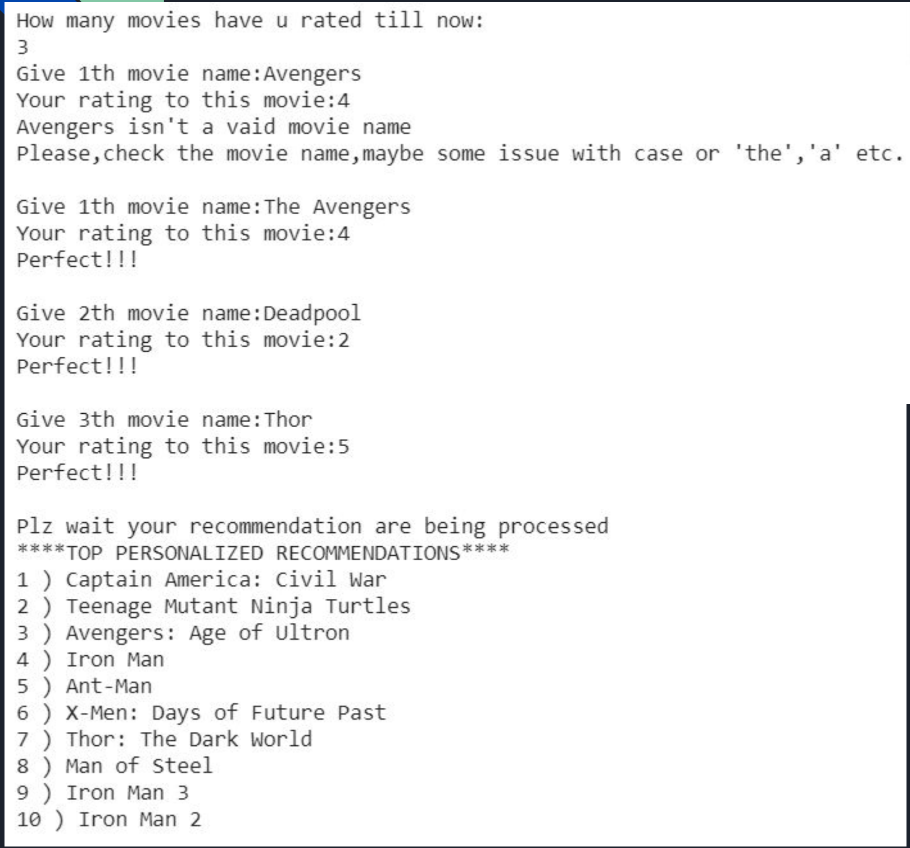
```
 **BEST ACROSS GENRES**
------> Action
1 One Man's Hero
2 Pirates of the Caribbean: At World's End
3 Avatar
------> Adventure
1 The Prisoner of Zenda
2 Pirates of the Caribbean: At World's End
3 Avatar
------> Science Fiction
1 The Empire Strikes Back
2 John Carter
3 Avatar
------> Crime
1 The Shawshank Redemption
2 The Dark Knight Rises
3 Spectre
------> Drama
1 Dancer, Texas Pop. 81
2 King Kong
3 The Dark Knight Rises
------> Thriller
1 Pulp Fiction
2 Quantum of Solace
3 The Dark Knight Rises
------> Family
1 Dancer, Texas Pop. 81
2 Harry Potter and the Half-Blood Prince
3 Tangled
------> Comedy
1 Stiff Upper Lips
2 Cars 2
3 Men in Black 3
------> Romance
1 Me You and Five Bucks
2 The Great Gatsby
```



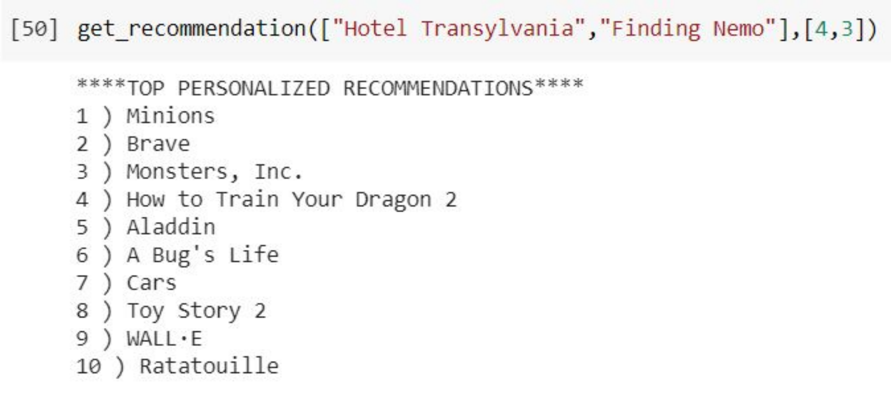
```
 **MOVIES OF TOP ACTORS**
------> Samuel L. Jackson
1 The Avengers
2 Avengers: Age of Ultron
------> Robert De Niro
1 Arthur and the Invisibles
2 Little Fockers
------> Bruce Willis
1 G.I. Joe: Retaliation
2 Armageddon
------> Matt Damon
1 Happy Feet Two
2 Interstellar
------> Morgan Freeman
1 The Dark Knight
2 The Dark Knight Rises
------> Steve Buscemi
1 G-Force
2 Monsters University
------> Liam Neeson
1 The Chronicles of Narnia: Prince Caspian
2 The Dark Knight Rises
------> Johnny Depp
1 Pirates of the Caribbean: Dead Man's Chest
2 Pirates of the Caribbean: At World's End
------> Owen Wilson
1 Night at the Museum: Battle of the Smithsonian
2 Cars 2
------> John Goodman
1 Transformers: Age of Extinction
2 Monsters University
```

The 3 screenshots show the outputs of the demographic filtering recommendations

Content based filtering output visualization



```
How many movies have u rated till now:
3
Give 1th movie name:Avengers
Your rating to this movie:4
Avengers isn't a vaid movie name
Please,check the movie name,maybe some issue with case or 'the','a' etc.

Give 1th movie name:The Avengers
Your rating to this movie:4
Perfect!!!

Give 2th movie name:Deadpool
Your rating to this movie:2
Perfect!!!

Give 3th movie name:Thor
Your rating to this movie:5
Perfect!!!

Plz wait your recommendation are being processed
****TOP PERSONALIZED RECOMMENDATIONS****
1 ) Captain America: Civil War
2 ) Teenage Mutant Ninja Turtles
3 ) Avengers: Age of Ultron
4 ) Iron Man
5 ) Ant-Man
6 ) X-Men: Days of Future Past
7 ) Thor: The Dark World
8 ) Man of Steel
9 ) Iron Man 3
10 ) Iron Man 2
```

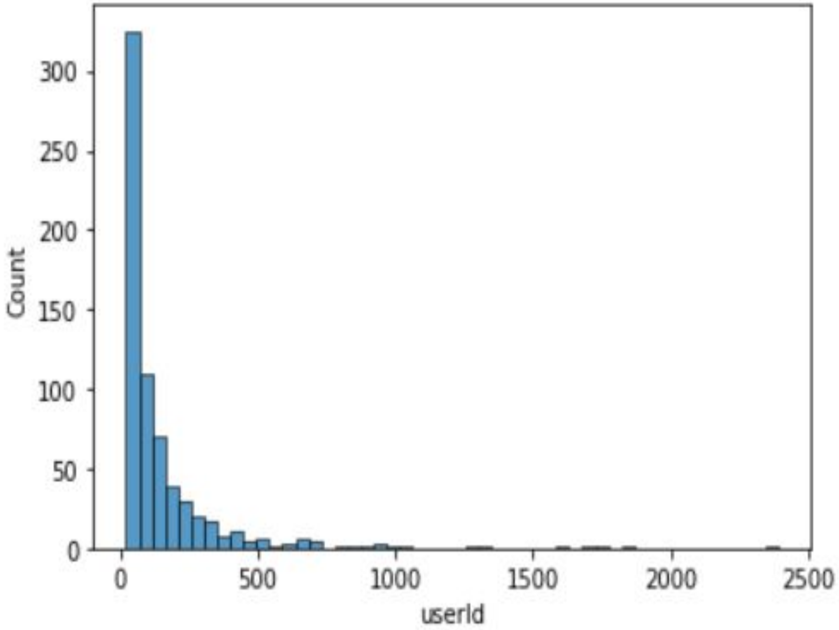


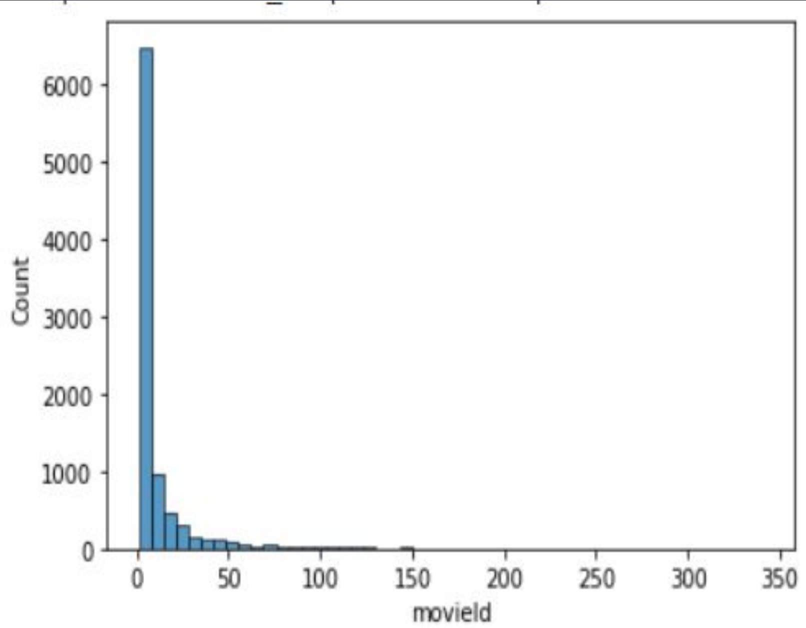
```
[50] get_recommendation(["Hotel Transylvania","Finding Nemo"],[4,3])

     ****TOP PERSONALIZED RECOMMENDATIONS****
     1 ) Minions
     2 ) Brave
     3 ) Monsters, Inc.
     4 ) How to Train Your Dragon 2
     5 ) Aladdin
     6 ) A Bug's Life
     7 ) Cars
     8 ) Toy Story 2
     9 ) WALL·E
     10 ) Ratatouille
```

Given,the watch history and corresponding ratings the above screenshots show the content-based recommendations.The right screenshot shows a viewer who loves Marvel movies and so are the recommendations.The left screenshot seems to be a kid and so cartoon movies have been recommended.

*Try testing for different watch history,check out the user manual for instructions to run the code

Collaborative filtering visualization:MovieLens data





Histogram:Number of ratings per user histogram

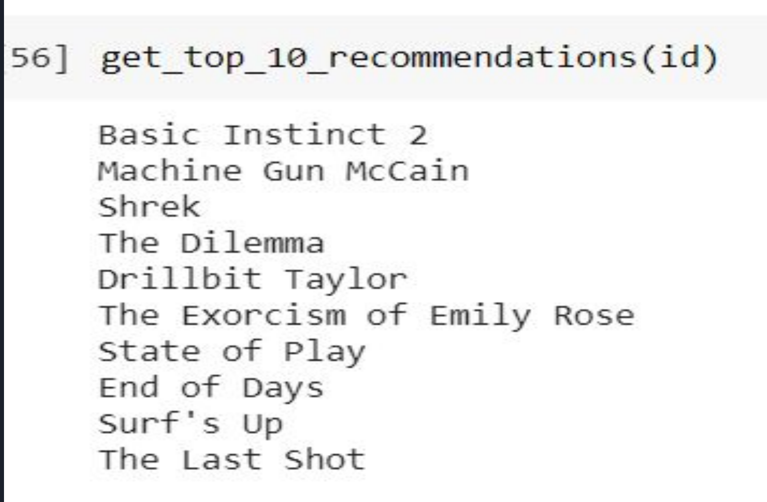
Histogram: Number of ratings per movies

The movieLens dataset has been used for collaborative filtering,the other 2 only require the TMDB dataset

Collaborative filtering output visualization

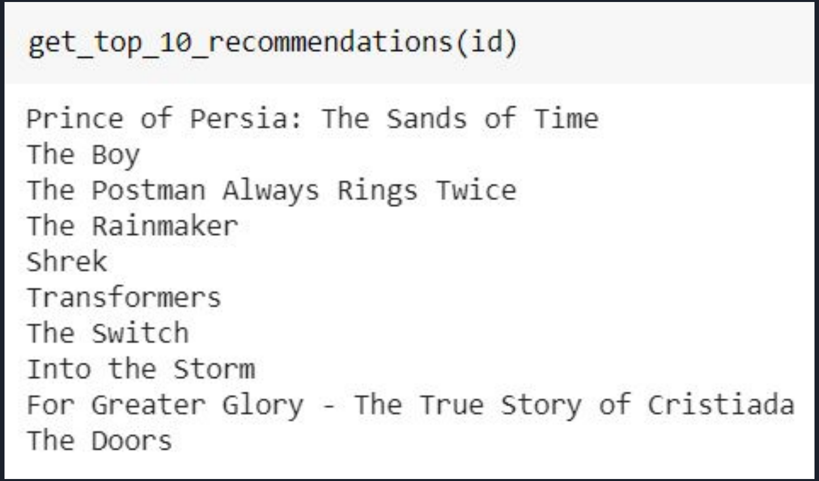
We obtain the estimated ratings for each user to unrated or unseen movies using SVD of the matrix as explained in conceptual design slide,then we recommend the movies with top 10 estimated ratings among unrated movies for each viewer,

For viewer Id->103:



```
56]  get_top_10_recommendations(id)

     Basic Instinct 2
     Machine Gun McCain
     Shrek
     The Dilemma
     Drillbit Taylor
     The Exorcism of Emily Rose
     State of Play
     End of Days
     Surf's Up
     The Last Shot
```

For viewer Id->348



```
get_top_10_recommendations(id)

Prince of Persia: The Sands of Time
The Boy
The Postman Always Rings Twice
The Rainmaker
Shrek
Transformers
The Switch
Into the Storm
For Greater Glory - The True Story of Cristiada
The Doors
```

Testing:Demographic filtering

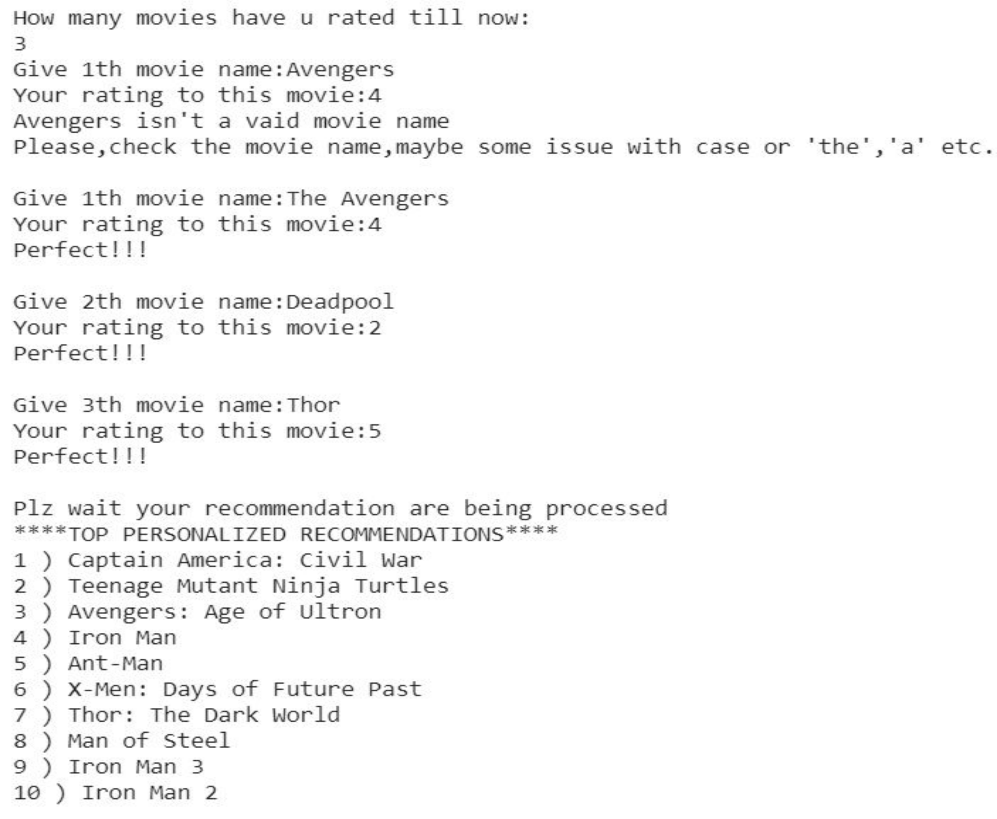
We can’t perform unittest as demographic recommendations don’t take any input but give the top picks across genres,actors,ratings etc.Thus,let’s manually check if the recommendations really make sense,

1)Minions,pirates of the carribean are truly popular,Shawshank redemption has the highest ImDb ratings followed by godfather 1&2 so it makes sense

2)Harry potter is a kids and family movies,pirates of carribean is action and adventure,the great gatsby is romantic so the genre based picks some alright.

3)Lets check out movies of top actors,Samuel Jackson has acted in Avengers,Johnny Depp has played Jack sparrow in pirates of the carribean and Morgan Freeman has been a part of the Batman franchise,so everything seems perfect

Unit testing:Content based filtering



```
How many movies have u rated till now:
3
Give 1th movie name:Avengers
Your rating to this movie:4
Avengers isn't a vaid movie name
Please,check the movie name,maybe some issue with case or 'the','a' etc.

Give 1th movie name:The Avengers
Your rating to this movie:4
Perfect!!!

Give 2th movie name:Deadpool
Your rating to this movie:2
Perfect!!!

Give 3th movie name:Thor
Your rating to this movie:5
Perfect!!!

Plz wait your recommendation are being processed
****TOP PERSONALIZED RECOMMENDATIONS****
1 ) Captain America: Civil War
2 ) Teenage Mutant Ninja Turtles
3 ) Avengers: Age of Ultron
4 ) Iron Man
5 ) Ant-Man
6 ) X-Men: Days of Future Past
7 ) Thor: The Dark World
8 ) Man of Steel
9 ) Iron Man 3
10 ) Iron Man 2
```

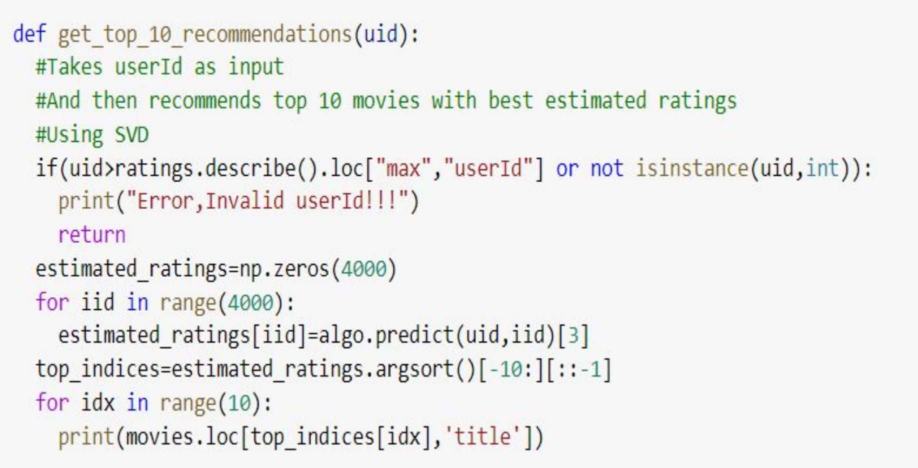
If the movie name is invalid the code flags an error while taking input.

For example:If the movies are [“Avengers”,”Iron man”] it will flag an error because the movie name is “The Avengers” and not “Avengers”.Similarly,the movie name is “Iron Man” as published by the movie makers not “Iron man” thus,the code keeps flagging an error until u get the correct name after a few iterations.

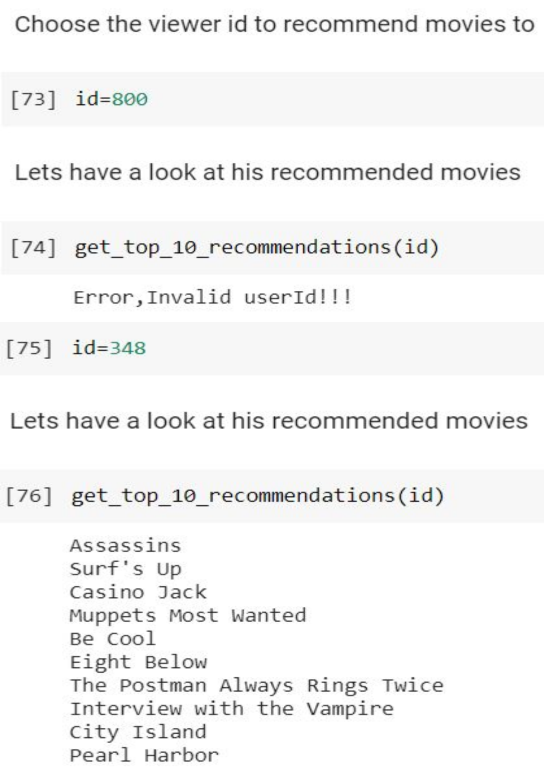
The recommendations don’t have an absolute answer so only correctness of movie name can be checked by unit test.

Unit testing:Collaborative filtering

Here,there are about 671 users in our MovieLens so if u give userId to be greater than 671 an error will be flagged by the program saying,"Error,Invalid userId!!!".Following is a screenshot showing the same in which the unit of the userId has been tested and error has been flagged



```python
def get_top_10_recommendations(uid):
  #Takes userId as input
  #And then recommends top 10 movies with best estimated ratings
  #Using SVD
  if(uid>ratings.describe().loc["max","userId"] or not isinstance(uid,int)):
    print("Error,Invalid userId!!!")
    return
  estimated_ratings=np.zeros(4000)
  for iid in range(4000):
    estimated_ratings[iid]=algo.predict(uid,iid)[3]
  top_indices=estimated_ratings.argsort()[-10:][::-1]
  for idx in range(10):
    print(movies.loc[top_indices[idx],'title'])
```



Choose the viewer id to recommend movies to

```
[73]  id=800
```

Lets have a look at his recommended movies

```
[74]  get_top_10_recommendations(id)

      Error,Invalid userId!!!

[75]  id=348
```

Lets have a look at his recommended movies

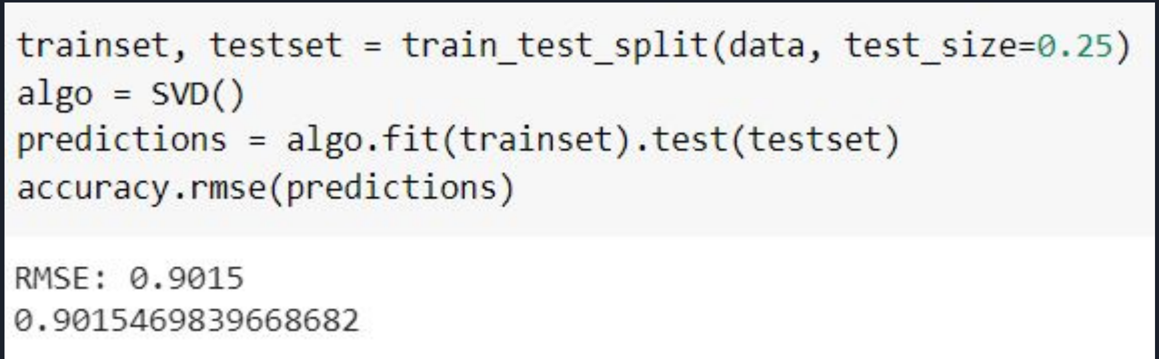
```
[76]  get_top_10_recommendations(id)

      Assassins
      Surf's Up
      Casino Jack
      Muppets Most Wanted
      Be Cool
      Eight Below
      The Postman Always Rings Twice
      Interview with the Vampire
      City Island
      Pearl Harbor
```

Model Training & testing

Recommendations systems are subjective things and models for content based filtering are based on similarity measures and can’t tested quantitavely , we can check out the recommendation and get a qualitative judgement of the prediction as content-based filtering isn’t a model based ML algorithm.

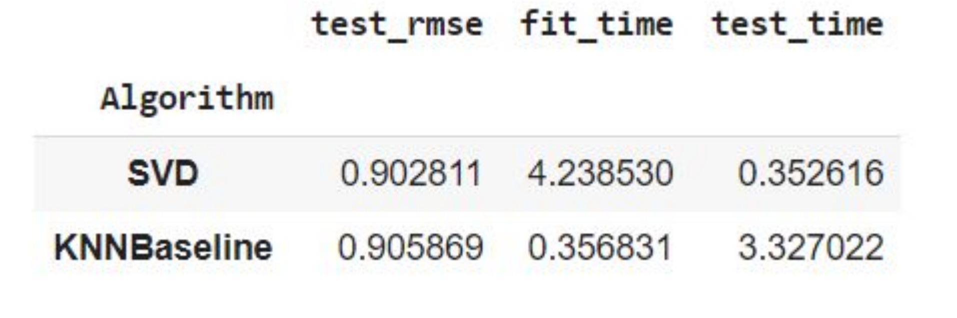
But content based filtering works on the Singular value decomposition of user-movie ratings matrix which finally estimated ratings of each user to movies he/she hasn’t rated we test the SVD algorithm by seperating 25% of the movieLens dataset into test set and then compare the estimated ratings with the test dataset after train the SVD algorithm.

Model Training & testing



```
trainset, testset = train_test_split(data, test_size=0.25)
algo = SVD()
predictions = algo.fit(trainset).test(testset)
accuracy.rmse(predictions)
```

```
RMSE: 0.9015
0.9015469839668682
```



	test_rmse	fit_time	test_time
Algorithm			
SVD	0.902811	4.238530	0.352616
KNNBaseline	0.905869	0.356831	3.327022

The training rmse is about 0.9 and even the test rmse is 0.9 for both SVD and KNN but SVD is marginally better.The SVD was trained on 75% of the ratings and tested on the remaining 25% ,this test rmse simple means that the estimated ratings are off by about 0.9 on an average which is not bad at all.