

### Program running instructions:

- 1) Run the program by running "python Recommender.py"
- 2) Inputs are hardcoded in the program in the main function().Line number 310
- 3) Currently the inputs are given as

```
datafile='u1.base'  
moviefile='u.item'  
userfile='u.user'  
occupationfile='u.occupation'  
testfile= 'u1.test'  
recommender = Recommender()  
distancemetric='euclidean' #euclidean,manhattan,lmax  
# recommender.naive(testfile,datafile)  
algo='parta' # parta or partb
```

You can change the training file by changing *datafile*, testfile by changing *testfile*, distance by *distancemetric*, and part a or part b of question by changing algo value as parta or partb respectively

- 4) For running the naïve strategy, Uncomment the code in line no 319, and comment line numbers 321 and 322.

I started the code by defining the naïve strategy for recommending the movie.

**Logic for** Naïve was that I defined a dictionary in which the key was movie id and values were a list of all the ratings given on that movie by all the users. Looping in test set, I just assigned the mean of the list to the corresponding movies.

### Pycharm output for Naïve approach:

---

```
Original Rating =4, Predicted Rating =3  
Original Rating =4, Predicted Rating =3  
Original Rating =4, Predicted Rating =3  
Original Rating =3, Predicted Rating =2  
Original Rating =3, Predicted Rating =3  
Original Rating =5, Predicted Rating =3
```

```
Mean Absolute Difference(MAD) value is -->0.94275
```

```
Percentage of movies rated correctly -->28.82%
```

### Fig: Code Screenshot Naïve approach

**PART A:** First, I generated distances from every user to all other user based on the ratings of the common movies they have seen. I made three separate functions which user can choose to calculate distance from as given and

calculated the observation based on all distances. The features getting subtracted for this part were the ratings.

**Observation:** 1) The program picks up and gives a better MAD (Mean Absolute Difference) value when I increase the value of k neighbors, i.e. the number of users which are similar to the given user based on the common movies they have seen.

The program gives about MAD value of about 1.06 when tested with value of k around 3-5 and gives around {0.8, 0.9} when tested on k = 40.

The program was tested on all the training and test files from u1 to u5.

- 2) The Euclidean distance performs a little better than Manhattan distance and Euclidean distance,

**PART B:** Since one feature is inadequate to accurately classify the result, so, in this step we are increasing the number of features to calculate distance.

In addition to the ratings of the users which have seen common movies, Users criteria {age, gender} are also taken. So, now combined vectors of {ratings, age, gender} are taken to calculate distance from the neighbors.

It also makes sense since the cluster of people of same age will have a consensus on the ratings they give for certain movies and same case is for the gender. Since Gender was given as M or F, we scaled it as 0 and 1 for calculating distance.

**Observation:** Though the program didn't show remarkable improvement but I am sure if k is peaked to an appropriate value and as more features are added, the program will perform better than part a of the problem.

Again the Euclidean distance performed better than rest two metrics.

**Part C:** On 100 M data sets, the program kept running for a long time because of 8000000 training rows and 200000 test rows. It generated numerous comparisons that's why took a huge amount of time.

The code is getting compiled correctly on the hulk server and tested the program with 100k dataset as well as 100M dataset. In this Unix absolute path importing os, was given as file input since the test and train files were in different directory and also the data was separated by (::) instead of space. So, the program needed a little bit of modification. I've created one different version of program for running on hulk servers named Recommender\_hulk.py

Here is the screenshot of program run on hulk.

Also, I have not used libraries like scipy which makes the program much faster.

### **Couple of observations and strategies for Million dataset:**

- 1) Since the program takes incredibly long time, understandably so, given the number of calculations, we must follow some strategy of parallelism in the code. We can use multithreading or map reduce in the code so that chunks of different calculations run on different threads of operating system.
- 2) Since, the dataset is huge, we should avoid creating multiple dictionaries and reading an input file more than once.

### **Part D:**

Possible tips to improve the accuracy of recommendation system:

- 1) Adding more relevant features and ignoring irrelevant features in predicting rating. It serves two purposes:
  - a) Like adding movie genre will help increase the accuracy since particular clusters of users have common taste of movie genre. So, it is relevant feature. But let's say a feature like zip-code might not be a good feature since users based on certain area might not say a specific thing about ratings.
  - b) Secondly, adding more irrelevant dimensions will lead to curse of dimensionality
- 2) Increasing the k value i.e the number of nearest neighbors to an optimum value.
- 3) Furthermore, we can use probabilistic techniques such as MCMC sampling, Markov chain or Viterbi sequences to predict the rating which might help given the amount of data set.
- 4) Feed forward Artificial Neural Networks with backpropagation can also be used to optimize the predicted values.

### **Screenshots of program run in various scenarios:**

- a) Train file: u1.base, Test file: u1.test, distance='Euclidean', part

a

```
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =2
Original Rating =1, Predicted Rating =4
Original Rating =3, Predicted Rating =3
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =2
Original Rating =5, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =5
Original Rating =4, Predicted Rating =2
Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =5
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =5
Original Rating =5, Predicted Rating =3
Mean Absolute Difference(MAD) value is -->0.8835
```

- b) Train file: u1.base, Test file: u1.test, distance=Manhattan, part a

```

Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =2
Original Rating =1, Predicted Rating =2
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =1
Original Rating =4, Predicted Rating =4
Original Rating =5, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =2
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =5
Original Rating =5, Predicted Rating =3
Mean Absolute Difference(MAD) value is -->0.905

```

c) Train file: u1.base, Test file: u1.test, distance=LMax, part a

```

Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =2
Original Rating =3, Predicted Rating =3
Original Rating =1, Predicted Rating =3
Original Rating =3, Predicted Rating =3
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =5
Original Rating =5, Predicted Rating =5
Original Rating =4, Predicted Rating =2
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =5
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =2
Original Rating =3, Predicted Rating =2
Original Rating =5, Predicted Rating =3
Mean Absolute Difference(MAD) value is -->0.9815

```

d) Train file: u2.base, Test file: u2.test, distance=Euclidean, part a

```

Original Rating =2, Predicted Rating =4
Original Rating =4, Predicted Rating =5
Original Rating =3, Predicted Rating =5
Original Rating =3, Predicted Rating =4
Original Rating =2, Predicted Rating =2
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =5
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =5
Mean Absolute Difference(MAD) value is -->0.8765

```

e) Train file: u2.base, Test file: u2.test, distance=Manhattan, part a

```

Original Rating =4, Predicted Rating =5
Original Rating =5, Predicted Rating =5
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =5
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =3
Original Rating =2, Predicted Rating =3
Original Rating =2, Predicted Rating =2
Original Rating =2, Predicted Rating =4
Original Rating =4, Predicted Rating =5
Original Rating =3, Predicted Rating =5
Original Rating =3, Predicted Rating =4
Original Rating =2, Predicted Rating =2
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =5
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =5

Original Rating =4, Predicted Rating =3
Mean Absolute Difference(MAD) value is -->0.8875
C:\Study\Sem2\Data Mining\assignments\Program>

```

f) Train file: u2.base, Test file: u2.test, distance= LMax, part a

```

Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =5
Original Rating =3, Predicted Rating =5
Original Rating =2, Predicted Rating =5
Original Rating =2, Predicted Rating =5
Original Rating =2, Predicted Rating =5
Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =5
Original Rating =2, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =5
Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =5

Original Rating =4, Predicted Rating =3
Mean Absolute Difference(MAD) value is -->0.9205

```

g) Train file: u3.base, Test file: u3.test, distance='Euclidean', part a

```

Original Rating =3, Predicted Rating =5
Original Rating =3, Predicted Rating =3
Original Rating =5, Predicted Rating =4
Original Rating =4, Predicted Rating =4
Original Rating =2, Predicted Rating =3
Original Rating =5, Predicted Rating =4

Original Rating =4, Predicted Rating =3
Original Rating =2, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =3
Original Rating =5, Predicted Rating =3

Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =3

Mean Absolute Difference(MAD) value is -->0.7555
C:\Study\Sem2\Data Mining\assignments\Program>

```

h) Train file: u3.base, Test file: u3.test, distance=Manhattan, part a

```

Original Rating =4, Predicted Rating =5
Original Rating =2, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =1
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =3
Original Rating =5, Predicted Rating =5

Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =3

Mean Absolute Difference(MAD) value is -->0.844
C:\Study\Sem2\Data Mining\assignments\Program>

```

- i) Train file: u3.base, Test file: u3.test, distance= LMax, part a

```

Original Rating =4, Predicted Rating =5
Original Rating =2, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =5, Predicted Rating =5

Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =3

Mean Absolute Difference(MAD) value is -->0.829

```

- j) Train file: u4.base, Test file: u4.test, distance='Euclidean', part a

```

Mean Absolute Difference(MAD) value is -->0.8445
C:\Study\Sem2\Data Mining\assignments\Program>

```

- k) Train file: u4.base, Test file: u4.test, distance=Manhattan, part a

```

Mean Absolute Difference(MAD) value is -->0.926
C:\Study\Sem2\Data Mining\assignments\Program>

```

- l) Train file: u4.base, Test file: u4.test, distance= LMax, part a

```

Mean Absolute Difference(MAD) value is -->0.984
C:\Study\Sem2\Data Mining\assignments\Program>

```

- m) Train file: u5.base, Test file: u5.test, distance='Euclidean', part a



```
Mean Absolute Difference(MAD) value is -->0.967
C:\Study\Sem2\Data Mining\assignments\Program>
```

- n) Train file: u5.base, Test file: u5.test, distance=Manhattan, part a

```
Mean Absolute Difference(MAD) value is -->0.999
C:\Study\Sem2\Data Mining\assignments\Program>
```

- o) Train file: u5.base, Test file: u5.test, distance= LMax, part a

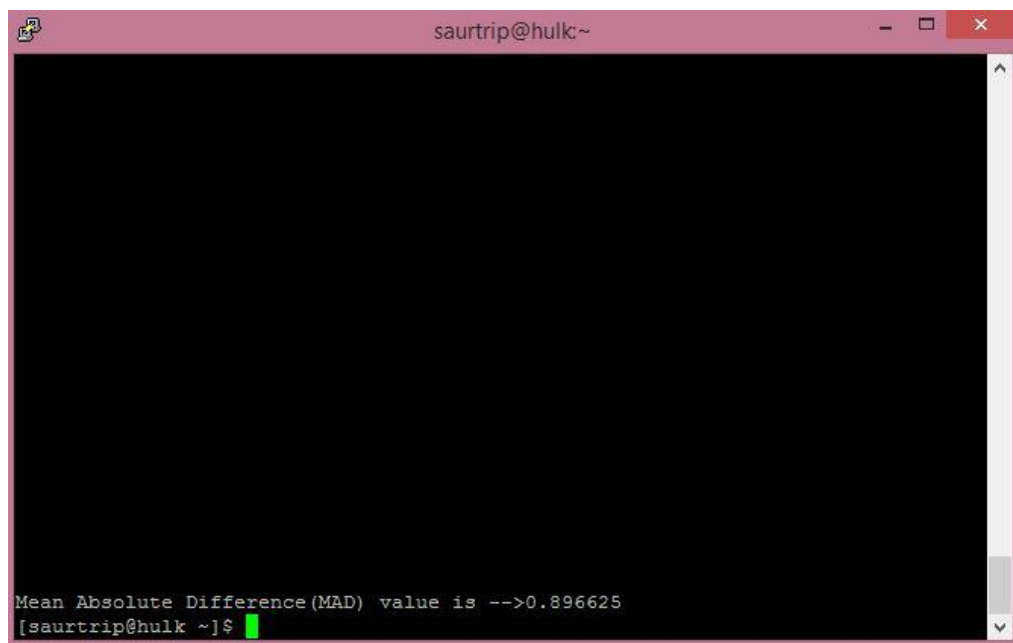
```
Mean Absolute Difference(MAD) value is -->0.926
C:\Study\Sem2\Data Mining\assignments\Program>
```

- p)

Part B:

```
Original Rating =1, Predicted Rating =2
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =1
Original Rating =4, Predicted Rating =4
Original Rating =5, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =4, Predicted Rating =2
Original Rating =4, Predicted Rating =4
Original Rating =3, Predicted Rating =4
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =4
Original Rating =4, Predicted Rating =3
Original Rating =3, Predicted Rating =3
Original Rating =3, Predicted Rating =5
Original Rating =5, Predicted Rating =3
Mean Absolute Difference(MAD) value is -->0.8885
C:\Study\Sem2\Data Mining\assignments\Program>
```

HULK server run:

A terminal window with a pink title bar and standard window controls. The title bar text is 'saurtrip@hulk:~'. The terminal area is black with white text. At the bottom, it shows the output of a command: 'Mean Absolute Difference(MAD) value is -->0.896625' followed by a new prompt line '[saurtrip@hulk ~]\$' with a green cursor.

```
saurtrip@hulk:~  
  
Mean Absolute Difference(MAD) value is -->0.896625  
[saurtrip@hulk ~]$
```