Answer 5:

Movie Rating Prediction System:

As per the requirement, I am predicting rating for the movies that users didn't see. Then I calculated Mean Absolute Difference (MAD) between predicted and actual values present in test.

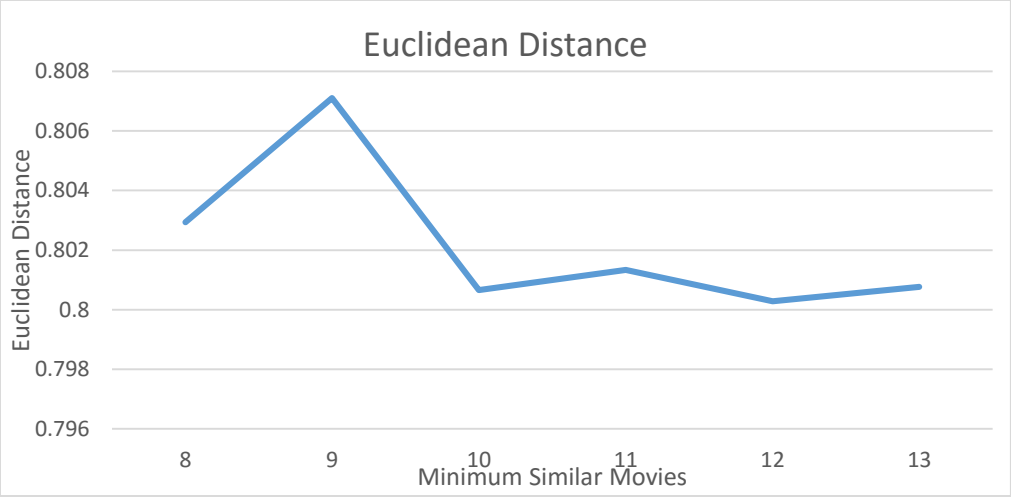
When I calculated average predictions by naïve algorithm then I am getting these results:

MAD	AverageCase
U1	0.826267027
U2	0.819109898
U3	0.810524945
U4	0.810343964
U5	0.815340613
Sum	4.081586447
Average	0.816317289

In order to perform well, MAD needs to be lesser than "Average Predictions":

To predict ratings I am finding similar users and then taking average of similar user's ratings. To find similar users I am considering a threshold (minimum number of movies that should be seen in common) that allows me to filter users and select appropriate users. I tried different threshold values and fixed 12 as I was getting best results at same. Here is the graph for same:

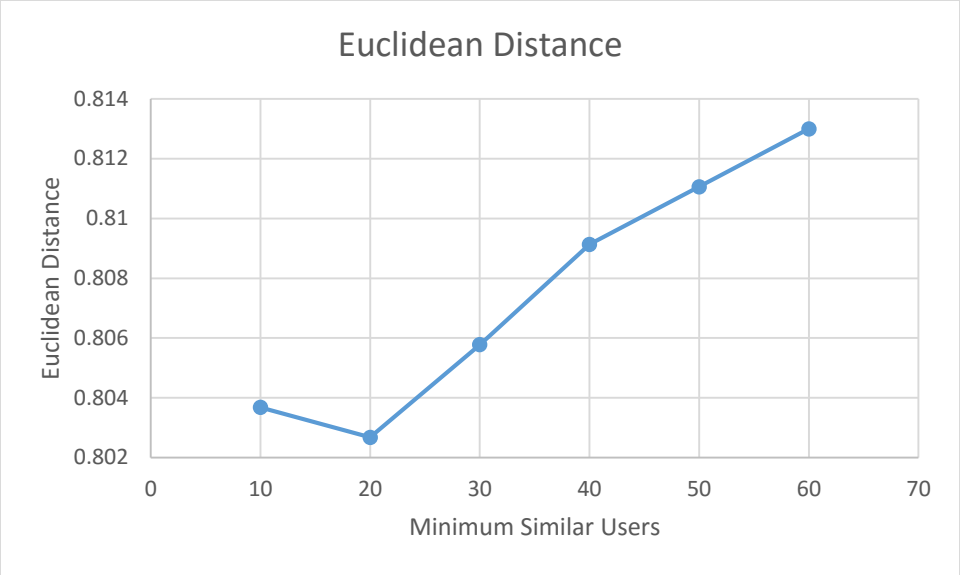
Minimum Similar Movies	Euclidean Distance
8	0.802935196
9	0.807105369
10	0.800655549
11	0.801329627
12	0.800277945
13	0.800761719



Another consideration that I was required to finalize is how many similar users I should find. For that I again tried multiple values and found out that results are best at 20. Here is the graph for same. Here no of similar movies is fixed t0 15.

Minimum similar movie	15

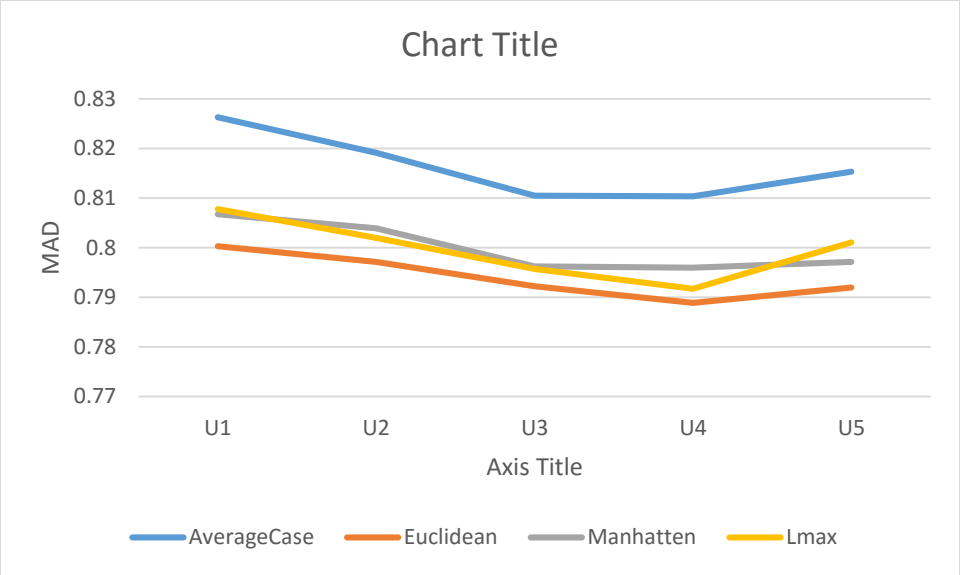
Minimum Users Similar	Euclidean Distance
10	0.803679287
20	0.802679287
30	0.80577524
40	0.809132612
50	0.811059696
60	0.812997296



I did all my experiments on Euclidean distance for U1 data set.

After fixing these values I tries 3 distance matrix that was told us to implement. I calculated distances between all the similar users and then I took 20 users with least distance. If I couldn't find 20 similar users than I am considering all the users to find average and then predict. I figured out that results are better when "Euclidean Distance" is applied. Here are the values and graph for the same:

MAD	AverageCase	Euclidean	Manhatten	Lmax
U1	0.826267027	0.800277945	0.806735777	0.807797
U2	0.819109898	0.797158886	0.803896013	0.801975
U3	0.810524945	0.79226658	0.796256261	0.795718
U4	0.810343964	0.788891504	0.795951034	0.791735
U5	0.815340613	0.79196658	0.797169405	0.801089
Sum	4.081586447	3.970561494	4.00000849	3.998315
Average	0.816317289	0.794112299	0.800001698	0.799663



(b)

Going forward I used Euclidean distance as it was best in previous case. While calculating I also incorporated age, gender, occupation and genre of dissimilar movies seen.

I figured out that minimum and maximum age in the data set is 7 and 73 respectively. So I scaled down age of all the users from 7-73 to 1-5 for normalizing as our movie ratings reside. I am taking difference in the age as an extra parameter along with ratings to find distance. Users with high age difference will be at greater distance.

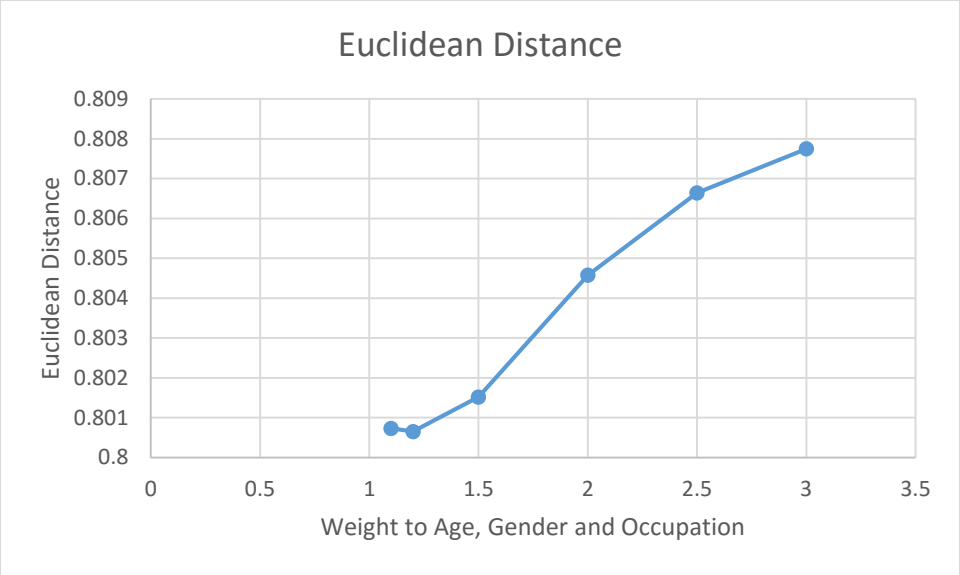
For gender and occupation, I am checking if considered users are of same gender and same applies to occupation. If users are of different gender and occupation then distance between them increase. I gave 3 as the basic difference in either case of different gender or occupation.

I thought that age, gender and occupation can be given more consideration while calculating distance so tried to give weights to these values. I tried different weights and fixed that weight of 1.1 is best. Here is the table and graph:

Minimum Similar User	20	Minumum Similar Movies	12

Weight to Age, Gender and Occupation	Euclidean Distance
1.1	0.80073117
1.2	0.800651042
1.5	0.801517428
2	0.80457482
2.5	0.806643129

3	0.807744892



Going further, to calculate distance I also incorporated genre of movies which are not common between users. For example:

User 1 saw: {1,2,3,4,5}

Users 2 saw: {2,3,7,8,9}

Similar Movies: {2,3}

Not common Movies Seen by 1: {1,4,5}

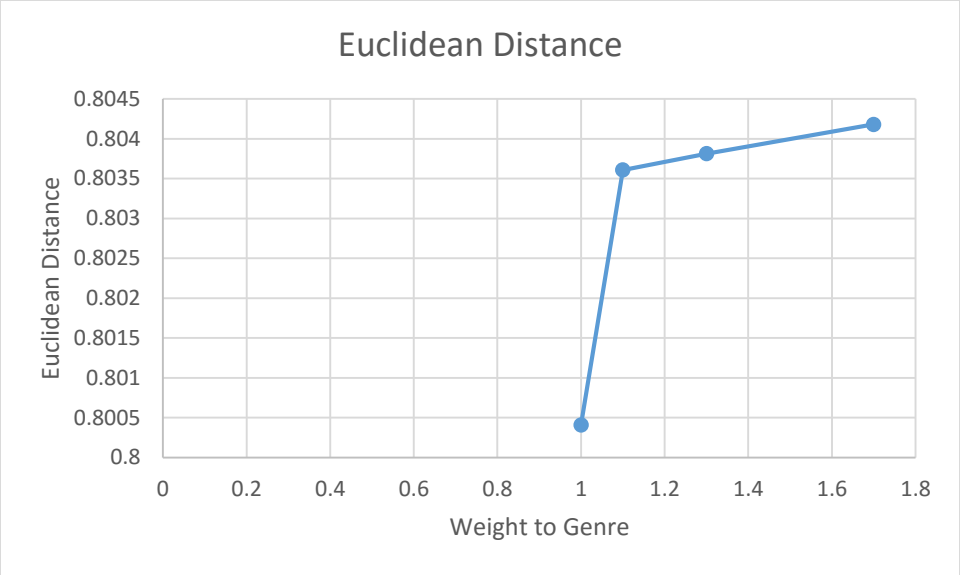
Not common Movies seen by 2: {7,8,9}

Now 2 and 3 , which are common, we are have already considered them to calculate distance so I thought to use other movies too which are not common.

I found top genres from “Not common” movies for both users and then I figured out that how many are common. If less genres are common that means more difference and vice-versa. I also tried different weights and fixed that weight of 1 is best. Here is the table and graph:

Everthing before this fixed

Weight to similar Genre	Euclidean Distance
1	0.800408153
1.1	0.803608273
1.3	0.803813602
1.7	0.804179187



Even after applying these parameters unfortunately I didn't get any significant improvement. Though I got some results better. Here is the comparison of Euclidean distance in part a and b of the question:

MAD	Euclidean in part b	Euclidean in part a
U1	0.800345553	0.800277945
U2	0.796565318	0.797158886
U3	0.790350631	0.79226658
U4	0.787802533	0.788891504
U5	0.796588359	0.79196658
Sum	3.971652394	3.970561494
Average	0.794330479	0.794112299