

# Introduction

The purpose of this assignment is to compare a User-based Recommendation System to an Item-based Recommendation System. The data used is based on a historical movie review dataset. The goal is to determine which system is a better recommendation system along with what parameters should be taken into account.

The User-based Recommendation System is based on using the Pearson's Correlation Coefficient for identifying similarities between users and using a corresponding equation that makes use of the average rating of the user to predict a rating for an item.

The Item-based Recommendation System is based on using the Adjusted Cosine Similarity for identifying similarities between items and using a corresponding equation that makes use of just similarity scores to predict a rating for an item.

The other variables being taken into account are the neighbourhood size on which similarities will be considered and a threshold that only takes into account certain values of similarities. Different combinations of these variables along with the two different systems would yield different results. The goal is to find the appropriate combination of system and variables that would result in the least amount of errors by calculating the Mean Absolute Error (MAE). Other factors to take into account is how feasible the size of neighbourhood used as well as the time taken to perform the calculation.

To create these tests, a strategy has to be considered to compare actual ratings from the dataset to a calculated value. In this assignment, the leave-one out cross validation strategy is used to compare calculated predicted ratings to actual ratings when the actual ratings is not a zero. This is done by doing the same calculated predicted rating to non-zero ratings as if their value is a zero then comparing it to the actual value. This would calculate the MAE.

Performing several tests, the lowest MAE values can be considered and the appropriate system and variables used will be compared to determine what is best for this particular dataset.

# User-based Recommendation System

User-based Recommendation system is used to predict the rating a user would rate by identifying similar users based on their ratings. After we collect all the users and their corresponding rating information, we focus on finding similar users to the target user for the item we are predicting. The similarity calculation considers only items that have been rated (non-zero) by both the target user and the other users. It is part of the leave-one out cross validation strategy where we exclude one rating from a user and predict it. The similarity calculation involves computing similarity scores using Pearson's Correlation Coefficient(PCC). It aims to measure the linear correlation between the two users, with scores ranging from -1(strong negative correlation) to +1(strong positive correlation). The formula used for PCC is

$$sim(a, b) = \frac{\sum_{p \in P} (r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P} (r_{a,p} - \bar{r}_a)^2 (r_{b,p} - \bar{r}_b)^2}}, P \text{ is set of products that both user } a \text{ and user } b \text{ have reviewed}$$

Since we are implementing the leave-one out cross validation, we have to exclude the current item while calculating the average( $\bar{r}_a$ ).

After measuring all the similarity scores, we begin to predict the ratings for the items that already have known ratings. The prediction formula is calculated as:

$$pred(a, p) = \bar{r}_a + \frac{\sum_{b \in N} sim(a, b)(r_{b,p} - \bar{r}_b)}{\sum_{b \in N} sim(a, b)}$$

If the denominator of the prediction value is 0, we assume the predicted score to be the average rating score of the user excluding the current rating that we are predicting. Another potential issue may arise when the number of available neighbours is less than our desired size(denoted as X). We will take the maximum number of available neighbours until it reaches size X. After we get all the predicted values, we evaluate the accuracy of our algorithm by comparing each predicted value to its original rating for each item and computing the Mean Absolute Error(MAE) to validate our leave-one-out cross validation strategy.

The runtime of the user-based recommendation system is approximately 90s for this dataset. The runtime was minimized by calculating the prediction after we computed all the similarity pairs between the target user and the rest of the users.

There are four cases to consider when we predict the rating since the assignment requires us to implement using either the top-K neighbour or similarity threshold, with or without negative correlations.

### Case 1: Top-k results, no threshold

We take in all possible similarity scores, sort them in descending order and select the top k scores. That being said, it returns the top k scores no matter whether the top k scores are positive or negative. For example, if we have a list of similarity scores [0.5,0.2,-0.3,-0.1] with k=3 it returns [0.5,0.2,-0.3]. The following are the results when k = 5 and k =50. The MAE increases when k=50. All of the results are tabulated in the Comparison of Results section.

```
User-based, no similarity threshold, top 5 neighbours
Total predictions: 14464
Total under predictions (< 1): 59
Total over predictions (> 5): 324
Number of cases with no valid neighbours: 0
Average neighbours used: 5
MAE = 0.6925020741150519
timerMAE: 1:29.406 (m:ss.mmm)
```

```
User-based, no similarity threshold, top 50 neighbours
Total predictions: 14464
Total under predictions (< 1): 239
Total over predictions (> 5): 719
Number of cases with no valid neighbours: 0
Average neighbours used: 19.05337389380531
MAE = 0.7285156250000091
timerMAE: 1:30.879 (m:ss.mmm)
```

### Case 2: Top-k results, only positives

We only consider positive similarity scores and sort them in descending order. For example, if we have a list of similarity scores [0.5,0.2,-0.3,-0.1] with k=3, it returns [0.5,0.2]. The following are the results when k = 5 and k = 50. The MAE decreases when k = 50. All of the results are tabulated in the Comparison of Results section.

```
User-based, has similarity threshold of 0, top 5 neighbours
Total predictions: 14464
Total under predictions (< 1): 17
Total over predictions (> 5): 288
Number of cases with no valid neighbours: 16
Average neighbours used: 4.902170907079646
MAE = 0.6840459070796526
timerMAE: 1:29.038 (m:ss.mmm)
```

```
User-based, has similarity threshold of 0, top 50 neighbours
Total predictions: 14464
Total under predictions (< 1): 9
Total over predictions (> 5): 188
Number of cases with no valid neighbours: 16
Average neighbours used: 15.25172842920354
MAE = 0.6628989214601831
timerMAE: 1:28.777 (m:ss.mmm)
```

### Case 3: Top-k results with absolute negative comparison, threshold = 0.5

We consider the absolute value of similarity scores that are above the threshold. The reason for taking the absolute value is that we are also taking into account the negative similarity scores. For example, if we have a list of similarity scores [0.5,0.2,-0.3,-0.1] with k=3, it returns [0.5]. The following are the results when k = 5 and k =50. The MAE decreases when k = 50. All of the results are tabulated in the Comparison of Results section.

```
User-based, has similarity threshold of 0.5, top 5 neighbours
Total predictions: 14464
Total under predictions (< 1): 372
Total over predictions (> 5): 979
Number of cases with no valid neighbours: 4293
Average neighbours used: 1.8722345132743363
MAE = 0.8206450497787763
timerMAE: 1:29.376 (m:ss.mmm)
```

```
User-based, has similarity threshold of 0.5, top 50 neighbours
Total predictions: 14464
Total under predictions (< 1): 370
Total over predictions (> 5): 934
Number of cases with no valid neighbours: 4293
Average neighbours used: 2.712458517699115
MAE = 0.8071999446902776
timerMAE: 1:27.408 (m:ss.mmm)
```

## Case 4: Top-k results with absolute negative comparison

We consider top-k based on the absolute value of correlation. That being said, we are taking the absolute value of the similarity scores and sorting them in descending order. For example, if we have a list of similarity scores [0.5,0.2,-0.3,-0.1] with k=3, it returns [0.5,-0.3,0.2] The following are the results when k = 5 and k =50. The MAE decreases when k = 50. All of the results are tabulated in the Comparision of Results section.

User-based, no similarity threshold, top 5 neighbours	User-based, no similarity threshold, top 50 neighbours
Total predictions: 14464	Total predictions: 14464
Total under predictions (< 1): 407	Total under predictions (< 1): 240
Total over predictions (> 5): 1077	Total over predictions (> 5): 769
Number of cases with no valid neighbours: 0	Number of cases with no valid neighbours: 0
Average neighbours used: 5	Average neighbours used: 19.05337389380531
MAE = 0.8048250829646075	MAE = 0.7301154590708061
timerMAE: 1:29.204 (m:ss.mmm)	timerMAE: 1:29.279 (m:ss.mmm)

## Item-based Recommendation System

The Item-based Recommendation System compares the similarity between items rated by users. This system compares the item to be predicted with another item that has been rated by all applicable users. The equation used to implement this system is the Adjusted Cosine Similarity, where  $U$  is the set of users,  $r$  is the rating of a specific user and  $\bar{r}$  is the average rating of said user:

$$\text{sim}(\bar{a}, \bar{b}) = \frac{\sum_{u \in U} (r_{u,a} - \bar{r}_u) \cdot (r_{u,b} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u,a} - \bar{r}_u)^2} \cdot \sqrt{\sum_{u \in U} (r_{u,b} - \bar{r}_u)^2}}$$

Since the implementation is considering the leave-one out cross validation, it will be excluded when calculating the average scores.

When a similarity score is done being calculated for a specific item whose rating is to be predicted then the prediction calculation can be done by using the following formula:

$$\text{pred}(u, p) = \frac{\sum_{i \in \text{ratedItems}(u)} \text{sim}(i, p) \cdot r_{u,i}}{\sum_{i \in \text{ratedItems}(u)} \text{sim}(i, p)}$$

where  $u$  is the user whose predicted rating is being calculated for a particular item and item  $i$  that they have rated to compare ratings.

During the prediction calculation, the similarity scores previously calculated are used. Depending on the similarity scores considered, the result could vary. These variations include the number of similarity neighbours being considered as well as the threshold for the similarity scores. Also, need to take into account the sum of the denominator should not be 0. When this

happens, the average rating score of that user excluding the rating that is currently being predicted for.

To determine how well the algorithm runs in terms of correctness, the leave-one out cross validation strategy is used. This is implemented by performing the predicted calculation for every rating and not just the rating that has a 0 (which would be normally predicted for). This way, those non-zero values have a calculated predicted rating to be compared to so the accuracy of the calculation can be determined.

The runtime for this code is below 60 seconds for this dataset. The runtime was minimized by calculating the predicted rating immediately after calculating the similarity score. This way, it is not looping to the same user and item to be calculated for. The average is not necessary to be calculated unless the denominator in the predicted rating is 0 thus the average is only calculated in those edge cases. This can only happen when negative similarities are considered.

The following cases show some of the results obtained from the tests, the other results will be attached in Appendix B.

### Case 1: Top-k results, no threshold

Case 1 takes in all the top  $k$  results regardless of sign. For example, in the similarity scores of  $[0.8, 0.1, -0.2, -0.4, -0.7]$  with  $k = 3$  then the values  $[0.8, 0.1, -0.2]$  are taken into account. The following are the results when  $k = 5$  and  $k = 50$ . All of the results are tabulated in the Comparison of Results section.

```
Item-based, no similarity threshold, top 5 neighbours
Total predictions: 14464
Total under predictions (< 1): 11
Total over predictions (> 5): 5
Number of cases with no valid neighbours: 0
Average neighbours used: 5
MAE = 0.7588491520899074
timerMAE: 48.178s
```

```
Item-based, no similarity threshold, top 50 neighbours
Total predictions: 14464
Total under predictions (< 1): 239
Total over predictions (> 5): 546
Number of cases with no valid neighbours: 0
Average neighbours used: 47.34554756637168
MAE = 0.7452255032817119
timerMAE: 48.196s
```

### Case 2: Top-k results, only positives

Case 2 takes in all the top  $k$  results that are positive, if the number of positive values is less than  $k$  then only take those values even if  $k$  is not reached. For example, in the similarity scores of  $[0.8, 0.1, -0.2, -0.4, -0.7]$  with  $k = 3$  then the values  $[0.8, 0.1]$  are taken into account. The following are the results when  $k = 5$  and  $k = 50$ . All of the results are tabulated in the Comparison of Results section.

```
Item-based, has similarity threshold of 0, top 5 neighbours
Total predictions: 14464
Total under predictions (< 1): 11
Total over predictions (> 5): 2
Number of cases with no valid neighbours: 1
Average neighbours used: 4.995022123893805
MAE = 0.7587033409734395
timerMAE: 50.527s
```

```
Item-based, has similarity threshold of 0, top 50 neighbours
Total predictions: 14464
Total under predictions (< 1): 0
Total over predictions (> 5): 0
Number of cases with no valid neighbours: 1
Average neighbours used: 43.026963495575224
MAE = 0.6756329287887215
timerMAE: 58.327s
```

### Case 3: Top-k results with absolute negative comparison, threshold = 0.5

Case 3 takes in all the top  $k$  results that are absolute above the threshold, if the number of accepted values is less than  $k$  then only take those values even if  $k$  is not reached. For example, in the similarity scores of  $[0.8, 0.1, -0.2, -0.4, -0.7]$  with  $k = 3$  then the values  $[0.8, -0.7]$  are taken into account. The following are the results when  $k = 5$  and  $k = 50$ . All of the results are tabulated in the Comparison of Results section.

```
Item-based, has similarity threshold of 0.5, top 5 neighbours
Total predictions: 14464
Total under predictions (< 1): 1057
Total over predictions (> 5): 1612
Number of cases with no valid neighbours: 20
Average neighbours used: 4.950221238938053
MAE = 1.1508176602032034
timerMAE: 49.615s
```

```
Item-based, has similarity threshold of 0.5, top 50 neighbours
Total predictions: 14464
Total under predictions (< 1): 1606
Total over predictions (> 5): 2293
Number of cases with no valid neighbours: 118
Average neighbours used: 39.41088219026549
MAE = 1.2106827334171013
timerMAE: 49.880s
```

### Case 4: Top-k results with absolute negative comparison

Case 4 takes in all the top  $k$  results that are absolute, taking into account more negative numbers. For example, in the similarity scores of  $[0.8, 0.1, -0.2, -0.4, -0.7]$  with  $k = 3$  then the values  $[0.8, -0.7, -0.6]$  are taken into account. The following are the results when  $k = 5$  and  $k = 50$ . All of the results are tabulated in the Comparison of Results section.

```
Item-based, no similarity threshold, top 5 neighbours
Total predictions: 14464
Total under predictions (< 1): 1053
Total over predictions (> 5): 1629
Number of cases with no valid neighbours: 0
Average neighbours used: 5
MAE = 1.1529594751896532
timerMAE: 47.515s
```

```
Item-based, no similarity threshold, top 50 neighbours
Total predictions: 14464
Total under predictions (< 1): 1606
Total over predictions (> 5): 2364
Number of cases with no valid neighbours: 98
Average neighbours used: 47.006775442477874
MAE = 1.2025526117922147
timerMAE: 48.769s
```

## Comparison of Results

### User-based Recommendation System

MAE Values for set conditions (round to three decimal place)

<u>Case #/K</u>	2	5	25	50	100
Case 1	0.751	0.693	0.716	0.729	0.730
Case 2	0.751	0.684	0.663	0.663	0.663
Case 3	0.855	0.821	0.801	0.807	0.807
Case 4	0.918	0.805	0.730	0.730	0.730

Case 1: It suggests the MAE is decreasing until it reaches its optimal and lowest MAE of 0.693 when  $k = 5$ . After  $k = 5$ , there is a slight increase in MAE.

Case 2 and 4: MAE keeps decreasing and remains constant when  $k = 25$ . It suggests adding more neighbours does not impact the accuracy of our predictions.

Case 3: MAE is the highest among other cases. It keeps decreasing until  $k = 25$ . There is a slight increase in MAE after it reaches its lowest MAE of 0.801 at  $k = 25$ .

It shows the negative correlations (case 3 and case 4) do not improve the accuracy of the recommendation system, but rather the opposite. MAE from case 3 and case 4 are higher compared to the other two cases where they exclude the negative correlations.

Based on the table above, it is confident to conclude that the threshold-based is more accurate for this data under the user-based recommendation system because it has a lower MAE than the top- $k$  method when we compare Case 2 (threshold method without negative correlation) with Case 1 (top- $k$  without negative correlation). Therefore, Case 2 is the ideal threshold method among the four cases. MAE keeps decreasing when the neighbourhood size is increasing. The runtime of the lowest MAE is 89.107s when  $k = 25$  (see Appendix A- Case 2).

## Item-based Recommendation System

MAE Values for set conditions (round to three decimal place)

<b>Case #/K</b>	<b>2</b>	<b>5</b>	<b>25</b>	<b>50</b>	<b>100</b>
<b>Case 1</b>	0.831	0.759	0.718	0.745	0.824
<b>Case 2</b>	0.831	0.759	0.690	0.676	0.668
<b>Case 3</b>	0.991	1.151	1.197	1.211	1.259
<b>Case 4</b>	0.993	1.153	1.191	1.203	1.254

Case 1 MAE values dip at around  $k = 25$  and is higher when  $k = 2$  and  $k = 100$ . This suggests that using case 1, no threshold and take everything into account, would be ideal to use at  $k = 25$ .

Case 2 MAE values keep lowering from  $k = 2$  to  $k = 100$ . This suggests that the higher the  $k$  value, the lower the MAE is when only taking positive values into account.

Case 3 and Case 4 values keep increasing, the difference in these 2 cases over the previous 2 is the inclusion of negative values. The system has higher discrepancies between predicted ratings compared to the actual ratings ( $> 1$  difference) which resulted in higher MAE. Since the predicted rating is based on just the sum of similarity adjusted ratings over the sum of similarities, the discrepancies will be more noticeable compared to the User-based Recommendation System where the predicted rating is calculated differently and has a starting

value of the user's average rating. These two cases will not be considered since they all result in higher MAE values.

This means that Case 1 and Case 2 are in consideration for the optimal variables for the Item-based Recommendation System where Case 1 is taking all top- $k$  similarities while Case 2 is taking only the positive top- $k$  similarities, even if  $k$  is not reached. From the results, at lower  $k$  values, the results are equivalent but as the value of  $k$  increases, it starts to shift in favour of Case 2 where at higher values of  $k$ , the MAE is lower.

Based off the analysis of the resulting data from the tests, for the Item-based Recommendation System, Case 2 where only positive  $k$  neighbours will be considered should be the ideal threshold. As for the value of  $k$ , it can be seen that the higher the  $k$ , the lower the MAE is which is ideal. The runtime when  $k = 100$  (lowest MAE) is 48.295s (see Appendix B - Case 2). For systems that is time sensitive and contains higher number of data to be considered, such as the real-time online movie recommendation system then the Item-based Recommendation System, with only taking up to  $k$  neighbours that are positive and at higher  $k$  values create an ideal combination.

## Conclusion

Based on the result from the User-based Recommendation System and Item-based Recommendation System, we can conclude that including negative similarities while predicting ratings does not improve the MAE of the recommendation system. Even though a high negative similarity means it has a strong negative correlation with the predicted item of the target user, which means if the user has a high rating, the target user will have a low rating, and vice versa. The increase in MAE compared to Case 1 and Case 2 where they exclude negative similarity shows the negative similarities do not contribute positively to the prediction. It decreases the accuracy of the predictions.

When we compare the four cases from the user-based recommendation system and Item-based Recommendation System, it is evident that Case 2(threshold method without negative similarity scores) is the optimal parameter since it computes the lowest MAE among the other cases for both recommendation systems. In the Item-based Recommendation System, as the neighbourhood size increases, the lower MAE we can compute. For the User-based Recommendation System, MAE decreases until  $k = 25$  and slightly increases afterwards, though the increase is less than 0.001.

The User-based nearest neighbour recommendation is more accurate for this data because all MAE from the User-based Recommendation System is lower than in the Item-based Recommendation System(see the tables from the Comparison section). The lower MAE shows that the user-based recommendation system has a higher overall accuracy than the item-based recommendation system. However, if our goal is to look for a recommendation system with a faster runtime, then an item-based recommendation system is the ideal solution because it



takes approximately 50s to calculate the result but it takes approximately 90s to run under a user-based recommendation system.

Therefore, it is convincing that a User-based Recommendation System with a threshold method without negative similarity scores is the best recommendation solution for the given dataset. If we are given fewer reviews, a User-based Recommendation System with a threshold method without negative similarity scores is still the best method because the fewer the neighbourhood size, it is still the lowest MAE compared to other cases in a User-based and Item-based Recommendation System. However, if we are given more reviews, an Item-based Recommendation System with a threshold method without negative similarity scores might be a better solution because the MAE in Item-based keeps decreasing but stays consistent in the User-based Recommendation System (with less than 0.001 increase). The Item-based Recommendation System is also a better solution if we aim to have a faster runtime.

Based on the above analysis and knowledge of both algorithms, it is sufficient to conclude that the Item-based Recommendation System is a better solution in a real-time online movie recommendation system because it requires significantly less time to get the result than the User-based Recommendation System system. Its MAE also keeps decreasing as we have greater number of reviews and it is no doubt that we have an enormously large reviews dataset in real life.

## Appendix

The following images are the other test results

### A. User-based Recommendation System

Case 1:

```
User-based, no similarity threshold, top 2 neighbours
Total predictions: 14464
Total under predictions (< 1): 65
Total over predictions (> 5): 583
Number of cases with no valid neighbours: 1
Average neighbours used: 1.9998617256637168
MAE = 0.7511144911504438
timerMAE: 1:29.830 (m:ss.mmm)
```

```
User-based, no similarity threshold, top 25 neighbours
Total predictions: 14464
Total under predictions (< 1): 207
Total over predictions (> 5): 570
Number of cases with no valid neighbours: 0
Average neighbours used: 15.634056969026549
MAE = 0.7164304480088587
timerMAE: 1:28.671 (m:ss.mmm)
```

```
User-based, no similarity threshold, top 100 neighbours
Total predictions: 14464
Total under predictions (< 1): 239
Total over predictions (> 5): 770
Number of cases with no valid neighbours: 0
Average neighbours used: 19.640763274336283
MAE = 0.7297967367256728
timerMAE: 1:30.137 (m:ss.mmm)
```

## Case 2:

```
User-based, has similarity threshold of 0, top 2 neighbours
Total predictions: 14464
Total under predictions (< 1): 62
Total over predictions (> 5): 578
Number of cases with no valid neighbours: 16
Average neighbours used: 1.9956443584070795
MAE = 0.7508476216814172
timerMAE: 1:28.996 (m:ss.mmm)
```

```
User-based, has similarity threshold of 0, top 25 neighbours
Total predictions: 14464
Total under predictions (< 1): 9
Total over predictions (> 5): 193
Number of cases with no valid neighbours: 16
Average neighbours used: 13.594441371681416
MAE = 0.6625988661504478
timerMAE: 1:29.107 (m:ss.mmm)
```

```
User-based, has similarity threshold of 0, top 100 neighbours
Total predictions: 14464
Total under predictions (< 1): 9
Total over predictions (> 5): 188
Number of cases with no valid neighbours: 16
Average neighbours used: 15.348727876106194
MAE = 0.6629258849557581
timerMAE: 1:28.474 (m:ss.mmm)
```

## Case 3:

```
User-based, has similarity threshold of 0.5, top 2 neighbours
Total predictions: 14464
Total under predictions (< 1): 573
Total over predictions (> 5): 1012
Number of cases with no valid neighbours: 4648
Average neighbours used: 1.1108268805309736
MAE = 0.8548527378318782
timerMAE: 1:29.062 (m:ss.mmm)
```

```
User-based, has similarity threshold of 0.5, top 25 neighbours
Total predictions: 14464
Total under predictions (< 1): 370
Total over predictions (> 5): 935
Number of cases with no valid neighbours: 4293
Average neighbours used: 2.6753318584070795
MAE = 0.8072462665929323
timerMAE: 1:27.657 (m:ss.mmm)
```

```
User-based, has similarity threshold of 0.5, top 100 neighbours
Total predictions: 14464
Total under predictions (< 1): 370
Total over predictions (> 5): 934
Number of cases with no valid neighbours: 4293
Average neighbours used: 2.712458517699115
MAE = 0.8071999446902776
timerMAE: 1:29.917 (m:ss.mmm)
```

### Case 4:

```
User-based, no similarity threshold, top 2 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 882  
Total over predictions (> 5): 1459  
Number of cases with no valid neighbours: 361  
Average neighbours used: 1.9500829646017699  
MAE = 0.9179030696902669  
timerMAE: 1:28.263 (m:ss.mmm)
```

```
User-based, no similarity threshold, top 25 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 240  
Total over predictions (> 5): 795  
Number of cases with no valid neighbours: 0  
Average neighbours used: 15.634056969026549  
MAE = 0.730427267699124  
timerMAE: 1:29.610 (m:ss.mmm)
```

```
User-based, no similarity threshold, top 100 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 239  
Total over predictions (> 5): 770  
Number of cases with no valid neighbours: 0  
Average neighbours used: 19.640763274336283  
MAE = 0.7297967367256728  
timerMAE: 1:29.185 (m:ss.mmm)
```

## B. Item-based Recommendation System

### Case 1:

```
Item-based, no similarity threshold, top 2 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 46  
Total over predictions (> 5): 49  
Number of cases with no valid neighbours: 1  
Average neighbours used: 1.9998617256637168  
MAE = 0.8313086423304121  
timerMAE: 52.313s
```

```
Item-based, no similarity threshold, top 25 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 91  
Total over predictions (> 5): 202  
Number of cases with no valid neighbours: 0  
Average neighbours used: 24.766316371681416  
MAE = 0.7179971459951789  
timerMAE: 49.948s
```

```
Item-based, no similarity threshold, top 100 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 547  
Total over predictions (> 5): 1089  
Number of cases with no valid neighbours: 0  
Average neighbours used: 85.16924778761062  
MAE = 0.8241049865860782  
timerMAE: 50.220s
```

## Case 2:

```
Item-based, has similarity threshold of 0, top 2 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 46  
Total over predictions (> 5): 49  
Number of cases with no valid neighbours: 1  
Average neighbours used: 1.9997234513274336  
MAE = 0.8313086423304121  
timerMAE: 51.031s
```

```
Item-based, has similarity threshold of 0, top 25 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 0  
Total over predictions (> 5): 0  
Number of cases with no valid neighbours: 1  
Average neighbours used: 23.78774889380531  
MAE = 0.6902297899877501  
timerMAE: 52.399s
```

```
Item-based, has similarity threshold of 0, top 100 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 0  
Total over predictions (> 5): 0  
Number of cases with no valid neighbours: 1  
Average neighbours used: 70.47358960176992  
MAE = 0.6677063220950935  
timerMAE: 48.295s
```

## Case 3:

```
Item-based, has similarity threshold of 0.5, top 2 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 902  
Total over predictions (> 5): 1133  
Number of cases with no valid neighbours: 2946  
Average neighbours used: 1.5893943584070795  
MAE = 0.9907439581540379  
timerMAE: 47.726s
```

```
Item-based, has similarity threshold of 0.5, top 25 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 1498  
Total over predictions (> 5): 2060  
Number of cases with no valid neighbours: 20  
Average neighbours used: 22.53166482300885  
MAE = 1.1968905543256225  
timerMAE: 52.179s
```

```
Item-based, has similarity threshold of 0.5, top 100 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 1813  
Total over predictions (> 5): 2650  
Number of cases with no valid neighbours: 64  
Average neighbours used: 64.57957688053098  
MAE = 1.2585016757368743  
timerMAE: 51.206s
```

## Case 4:

```
Item-based, no similarity threshold, top 2 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 912  
Total over predictions (> 5): 1139  
Number of cases with no valid neighbours: 2927  
Average neighbours used: 1.595271017699115  
MAE = 0.9934064972847232  
timerMAE: 51.856s
```

```
Item-based, no similarity threshold, top 25 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 1501  
Total over predictions (> 5): 2106  
Number of cases with no valid neighbours: 0  
Average neighbours used: 24.766316371681416  
MAE = 1.191498813722769  
timerMAE: 51.104s
```

```
Item-based, no similarity threshold, top 100 neighbours  
Total predictions: 14464  
Total under predictions (< 1): 1806  
Total over predictions (> 5): 2773  
Number of cases with no valid neighbours: 44  
Average neighbours used: 84.86504424778761  
MAE = 1.2537284985245833  
timerMAE: 53.645s
```