Homework 4

Recitation Problems

Problem 9.2.1: Three computers A, B, C have numerical featureslisted below

Feature	A	B	C
Processor Speed	3.06	2.68	2.92
Disk Size	500	320	640
Main-Memory Size	6	4	6

We may imagine these values as defining a vector for each computer; for instance, A's vector is [3.06, 500, 6]. We can compute the cosine distance between any two of the vectors, but if we do not scale the components, then the disk size will dominate and make differences in the other components essentially invisible.
Let us use 1 as the scale factor for processor speed, α for the disk size and β for the main memory size.
(a) In terms of α and β, compute the cosines of the angles between the vectors for each pair of the three computers.
(b)What are the angles between the vectors if $\alpha = \beta = 1$?
(c) What are the angles between the vectors if $\alpha = 0.01$ and $\beta = 0.5$?

Answer:
Cosine of the angle between two vector is represented as
$= \sum x.y / (|x|.\ |y|)$
$|x|$= length of vector x $=\sqrt{(\sum (x^2))}$
$|y|$= length of vector y $=\sqrt{(\sum (y^2))}$

a) After applying Scaling factor

Feature	A	B	C
Processor Speed	3.06(1)	2.68(1)	2.92(1)
Disk Size	500α	320α	640α
Main Memory size	6β	4β	6β

Cosine of the angle between A[3.06 500α 6β]and B[2.68 320α 4β]
$= 3.06*2.68+ 500\alpha* 320\alpha+ 6\beta* 4\beta/ (\sqrt{(3.06^2+500\alpha^2+6\beta^2)} * \sqrt{(2.68^2+320\alpha^2+4\beta^2)})$
$= 8.2008 + 160000\alpha^2 +24\beta^2/ \sqrt{9.3636 +250000\alpha^2+36\beta^2} * \sqrt{7.1824 +102400\alpha^2 +16\beta^2}$

Cosine of the angle between A [3.06 500α 6β] and C [2.92 640α 6β]
$= 8.9352+ 320000\alpha^2+ 36\beta^2/\sqrt{9.3636 +250000\alpha^2+36\beta^2} * \sqrt{8.5264+409600\alpha^2+36\beta^2}$

Cosine of the angle between B [2.68 320α 4β] and C [2.92 640α 6β]
$= 7.8256+204800\alpha^2+24\beta^2/ \sqrt{7.1824 +102400\alpha^2 +16\beta^2} * \sqrt{8.5264+409600\alpha^2+36\beta^2}$

b) the angles between the vectors if $\alpha = \beta = 1$

Cosine of the angle between A [3.06 500α 6β]and B[2.68 320α 4β]

$= 8.2008 + 160000\alpha^2 + 24\beta^2 / \surd 9.3636 + 250000\alpha^2 + 36\beta^2 * \surd 7.1824 + 102400\alpha^2 + 16\beta^2$

$= 8.2008 + 160000*1^2 + 24*1^2 / \surd 9.3636 + 250000*1^2 + 36*1^2 * \surd 7.1824 + 102400*1^2 + 16*1^2$

= 0.99999733

theta =cos-1(0.99999733)

= 0.132

Cosine of the angle between A[3.06 500α 6β] and C [2.92 640α 6β]

$= 8.9352 + 320000\alpha^2 + 36\beta^2 / \surd 9.3636 + 250000\alpha^2 + 36\beta^2 * \surd 8.5264 + 409600\alpha^2 + 36\beta^2$

$= 8.9352 + 320000*1^2 + 36*1^2 / \surd 9.3636 + 250000*1^2 + 36*1^2 * \surd 8.5264 + 409600*1^2 + 36*1^2$

theta =cos-1 (0.99999534)

= 0.175

Cosine of the angle between B[2.68 320α 4β] and C [2.92 640α 6β]

$= 7.8256 + 204800\alpha^2 + 24\beta^2 / \surd 7.1824 + 102400\alpha^2 + 16\beta^2 * \surd 8.5264 + 409600\alpha^2 + 36\beta^2$

$= 7.8256 + 204800*1^2 + 24*1^2 / \surd 7.1824 + 102400*1^2 + 16*1^2 * \surd 8.5264 + 409600*1^2 + 36*1^2$

theta =cos-1 (0.99998785)

= 0.282

c)the angles between the vectors if $\alpha = 0.01$ and $\beta = 0.5$

Cosine of the angle between A [3.06 500α 6β]and B[2.68 320α 4β]

$= 3.06*2.68 + 500\alpha * 320\alpha + 6\beta * 4\beta / \surd 3.06^2 + 500\alpha^2 + 6\beta^2 * \surd 2.68^2 + 320\alpha^2 + 4\beta^2$

$= 8.2008 + 160000*0.01^2 + 24*0.5^2 / \surd 9.3636 + 250000*0.01^2 + 36*0.5^2 * \surd 7.1824 + 102400*0.01^2 + 16*0.5^2$

theta =cos-1 (0.9908815)

=7.74

Cosine of the angle between A[3.06 500α 6β] and C [2.92 640α 6β]

$= 8.9352 + 320000*\alpha^2 + 36\beta^2 / \surd 9.3636 + 250000\alpha^2 + 36\beta^2 * \surd 8.5264 + 409600\alpha^2 + 36\beta^2$

$= 8.9352 + 320000*0.01^2 + 36*0.5^2 / \surd 9.3636 + 250000*0.01^2 + 36*0.5^2 * \surd 8.5264 + 409600*0.01^2 + 36*0.5^2$

theta =cos-1 (0.99155471)

= 7.45

Cosine of the angle between B[2.68 320α 4β] and C [2.92 640α 6β]

$= 7.8256 + 204800\alpha^2 + 24\beta^2 / \surd 7.1824 + 102400\alpha^2 + 16\beta^2 * \surd 8.5264 + 409600\alpha^2 + 36\beta^2$

$= 7.8256 + 204800*0.01^2 + 24*0.5^2 / \surd 7.1824 + 102400*0.01^2 + 16*0.5^2 * \surd 8.5264 + 409600*0.01^2 + 36*0.5^2$

theta = cos-1 (0.96917792)

= 14.26

(d) One fair way of selecting scale factors is to make each inversely proportional to the average value in its component. What would be the values of α and β, and what would be the angles between the vectors?

Answer-

Feature	A	B	C	Average of Compnents
Processor Speed	3.06	2.68	2.92	2.88
Disk Size	500	320	640	486.67
Main Memory size	6	4	6	5.33

Scale factor for processor speed =1/2.88 = 0.347

Scale factor for Disk size, α=1/486.67 = 0.002

Scale factor for Main memory size, β =1/5.33 = 0.187

Feature	A	B	C
Processor Speed	1.06	0.93	1.01
Disk Size	1	0.64	1.28
Main Memory size	1.12	0.75	1.12

Cosine of the angle between A[1.06 1 1.12]and B[0.93 0.64 0.75]

$= 1.06*0.93+ 1*0.64+ 1.12*0.75/ \sqrt{}1.06^2+1^2+1.12^2 * \sqrt{}\ 0.93^2+0.64^2+0.75^2$

theta = cos-1 (0.9898) =8.19

Cosine of the angle between A[1.06 1 1.12] and C [1.01 1.28 1.12]

$= 1.06*1.01+ 1*1.28+ 1.12*1.12/\sqrt{}1.06^2+1^2+1.12^2 * \sqrt{}1.01^2+1.28^2+1.12^2$

theta =cos-1 (0.9915) = 7.475

Cosine of the angle between B[0.93 0.64 0.75] and C [1.01 1.28 1.12]

$= 0.93*1.01+0.64*1.28+0.75*1.12/\sqrt{}\ 0.93^2+0.64\text{^}2+0.75^2 * \sqrt{}1.01^2+1.28^2+1.12^2$

theta =cos-1 (0.9692) = 14.257

Problem 9.2.3 : A certain user has rated the three computers of Exercise 9.2.1 as follows: A: 4 stars, B: 2 stars, C: 5 stars.

(a) Normalize the ratings for this user.

(b) Compute a user profile for the user, with components for processor speed, disk size, and main memory size, based on the data of Exercise 9.2.1.

Answer -

a) Normalized ratings for user X ,the average rating by a user X is 2
   Average = (4+2+5)/3 = 11/3
   A = 4 – (11/3) = (12-11)/3 = 1/3 = 0.333333
   B = 2-(11/3) = (6-11)/3 = -5/3 = -1.666666
   C = 5- (11/3) = (15-11)/3 = 4/3 = 1.33333

Feature	A	B	C
User ratings	0.34	-1.66	1.34

b)
Processor Speed
 = (3.06 * 0.34) – (2.68* 1.66) + (2.92*1.34)
= (1.02) -(4.46)+(3.89)
= 0.4467

Disk Size
= (500*0.34) – (320*1.66) +(640*1.34)
= 166.66-533.33+853.3
=486.667

Main Memory Size
=(6 * 0.34) – (4* 1.66) + (6*1.34)
= (2.04) -(6.64)+(8.04)
= 3.44

Exercise 9.3.1 : Figure 9.8 is a utility matrix, representing the ratings, on a 1–5 star scale, of eight items, a through h, by three users A, B, and C. Compute the following from the data of this matrix.
(a) Treating the utility matrix as boolean, compute the Jaccard distance between each pair of users.
(b) Repeat Part (a), but use the cosine distance.
(c)Treat ratings of 3, 4, and 5 as 1 and 1, 2, and blank as 0. Compute the Jaccard distance between each pair of users.
(d) Repeat Part (c), but use the cosine distance.
(e) Normalize the matrix by subtracting from each nonblank entry the average value for its user.
(f) Using the normalized matrix from Part (e), compute the cosine distance between each pair of users

	a	b	c	d	e	f	g	h
A	4	5		5	1		3	2
B		3	4	3	1	2	1	
C	2		1	3		4	5	3

Answer -
(a) Treating the utility matrix as boolean, compute the Jaccard distance between each pair of users.
Jaccard similarity between A and B = 4/8 =0.5

Jaccard similarity between A and C = 4/8 =0.5

Jaccard similarity between B and C = 4/8 =0.5

Sim(A,B) = Sim(A,C)=Sim(B,C)

(b) Repeat Part (a), but use the cosine distance.

Cosine Similarity between A and B = 0.601

Cosine Similarity between A and C = 0.615

Cosine Similarity between B and C = 0.5138

Sim(A,C)>Sim(A,B)>Sim(B,C)

(c)Treat ratings of 3, 4, and 5 as 1 and 1, 2, and blank as 0. Compute the Jaccard distance between each pair of users.

	a	b	c	d	e	f	g	h
A	1	1	0	1	0	0	1	0
B	0	1	1	1	0	0	0	0
C	0	0	0	1	0	1	1	1

Jaccard Similarity = $M_{11}/(M_{01}+M_{10}+M_{11})$
Jaccard Distance = $(M_{10}+ M_{01})/(M_{01}+M_{10}+M_{11})$ = 1-J

Jaccard Similiarity between A and B = 2/5 = 0.4
Jaccard Distance = 0.6

Jaccard Similiarity between A and C = 2/6 = 0.3333
Jaccard Distance = 0.666667

Jaccard Similiarity between B and C = 1/6 = 0.1666
Jaccard Distance = 0.833334

Sim(A,B)>Sim(A,C)>Sim(B,C)

(d) Repeat Part (c), but use the cosine distance.
Cosine Similarity = $\sum A_iB_i /(\surd A_i^2 \surd B_i^2)$
Cosine Distance = $\cos^{-1}$(Similarity)/π

Cosine Similarity between A and B = 0.5774
Cosine Distance = 0.4226

Cosine Similarity between A and C = 0.5
Cosine Distance = 0.5

Cosine Similarity between B and C = 0.2887
Cosine Distance = 0.7113

(e) Normalize the matrix by subtracting from each nonblank entry the average value for its user.

Normalizing the utility matrix

	a	b	c	d	e	f	g	h
A	0.66	1.66		1.66	1		-0.34	-1.34
B		0.66	1.66	0.66	-1.34	-0.34	-1.34	
C	-1		-2	0		1	2	0

(f) Using the normalized matrix from Part (e), compute the cosine distance between each pair of users

Cosine similarity between A and B = 0.584

Cosine similarity between A and C = -0.1154

Cosine similarity between B and C = -0.73955

Exercise 9.4.1 : Starting with the decomposition of Fig. 9.10, we may choose any of the 20 entries in U or V to optimize first. Perform this first optimization step assuming we choose: (a) u32 (b) v41.

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{bmatrix}$$

Answer –

Answer :

a) u32

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & x \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 \\ (x+1) & (x+1) & (x+1) & (x+1) & (x+ \\ 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{bmatrix}$$

$$(2-(x+1))^2 + (3-(x+1))^2 + (1-(x+1))^2 + (4-(x+1))^2$$

This simplifies :

$$(1-x)^2 + (2-x)^2 + (-x)^2 + (3-x)^2$$

minimizing the sum, so take derivative and set equal to 0,

$$2\left[(1-x) + (2-x) + (-x) + (3-x)\right] = 0$$

$\therefore 4x = 6$

$\therefore x = 6/4$;

$\therefore \boxed{x = 1.5}$

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1.5 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 \\ 2.5 & 2.5 & 2.5 & 2.5 & 2.5 \\ 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{bmatrix}$$

b) V_{41}

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & y & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 2 & 2 & y+1 & 2 \\ 2 & 2 & 2 & y+1 & 2 \\ 2 & 2 & 2 & y+1 & 2 \\ 2 & 2 & 2 & y+1 & 2 \\ 2 & 2 & 2 & y+1 & 2 \end{bmatrix}$$

$$= (y-3)^2 + (y-3)^2 + y^2 + (y-2)^2 + (y-3)^2$$

$$\frac{dy}{dx} = 2(y-3) + 2(y-3) + 2y + 2(y-2) + 2(y-3)$$

$$0 = 2\{(y-3) + (y-3) + y + (y-2) + (y-3)\}$$

$$0 = \cancel{2}\{5y - 11\}$$

$$5y - 11 = 0 \qquad y = 11/5 = 2.2$$

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 & 2.2 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 2 & 2 & 3.2 & 2 \\ 2 & 2 & 2 & 3.2 & 2 \\ 2 & 2 & 2 & 3.2 & 2 \\ 2 & 2 & 2 & 3.2 & 2 \\ 2 & 2 & 2 & 3.2 & 2 \end{bmatrix}$$

2.1 Problem 1

Load the Movielens 100k dataset (ml-100k.zip) into Python using Pandas dataframes. Build a user profile on unscaled data for both users 200 and 15, and calculate the cosine similarity and distance between the user's preferences and the item/movie 95. Which user would a recommender system suggest this movie to?

Solution:

```
import numpy as np
import pandas as pd
from sklearn import metrics
```

```
user_cols = ['user_id',
             'age',
             'gender',
             'occupation',
             'zip_code']
users = pd.read_csv(r'E:\ml-100k\u.user',
                    sep='|',
                    names=user_cols)
```

```
rating_cols = ['user_id',
               'movie_id',
               'rating',
               'timestamp']
ratings = pd.read_csv(r'E:\ml-100k\u.data',
                      sep='\t',
                      names=rating_cols)

r=pd.DataFrame(pd.read_table(r'E:\ml-100k\u.data',sep = '\t',names=rating_cols ))
```

```
item_cols = ['movie id',
             'movie title',
             'release date',
             'video release date',
             'IMDb URL',
             'Unknown',
             'Action',
             'Adventure',
             'Animation',
             'Childrens',
             'Comedy',
             'Crime',
             'Documentary',
             'Drama',
             'Fantasy',
             'FilmNoir',
             'Horror',
             'Musical',
             'Mystery',
             'Romance',
             'SciFi',
             'Thriller',
             'War',
             'Western']
items = pd.read_csv(r'E:\ml-100k\u.item',
                    sep='|',
                    names=item_cols,
                    encoding='latin-1')
i=pd.DataFrame(pd.read_table(r'E:\ml-100k\u.item',sep = '|',names=item_cols,encoding='latin-1' ))
```

```
user200 = r.loc[r['user_id'].isin([200])]
v=user200['movie_id']
v200 =pd.DataFrame(data=v)
#for row in  i,v1
```

```
v2= i.loc[i['movie id'].isin(v1['movie_id'])]
```

```
v3 = v2.iloc[:,5:24]
mean3 = v3.mean(axis=0)
```

```
user2 = r.loc[r['user_id'].isin([15])]
v4=user2['movie_id']
v4 =pd.DataFrame(data=v4)
v4
v5= i.loc[i['movie id'].isin(v4['movie_id'])]
v5
v5 = v5.iloc[:,5:24]
v5.head()
mean5 = v5.mean(axis=0)
```

```
item2 = items[94:95]
feat2 = item2.iloc[:,5:24]
feat2.head()
mean2 = feat2.mean()
```

```
mean200 = mean3.reshape(1,-1)
mean200
```

```
array([[ 0.        ,  0.36574074,  0.26388889,  0.07407407,  0.19444444,
         0.24074074,  0.0462963 ,  0.00462963,  0.28240741,  0.0462963 ,
         0.00925926,  0.0462963 ,  0.0787037 ,  0.01851852,  0.16666667,
         0.19907407,  0.23148148,  0.0787037 ,  0.01851852]])
```

```
mean15 = mean5.reshape(1,-1)
mean15
```

```
array([[ 0.        ,  0.21153846,  0.11538462,  0.01923077,  0.06730769,
         0.26923077,  0.05769231,  0.        ,  0.46153846,  0.02884615,
         0.01923077,  0.01923077,  0.01923077,  0.06730769,  0.24038462,
         0.11538462,  0.24038462,  0.09615385,  0.        ]])
```

```
mean95 = mean2.reshape(1,-1)
mean95
```

```
array([[ 0.,  0.,  0.,  1.,  1.,  1.,  0.,  0.,  0.,  0.,  0.,  0.,  1.,
         0.,  0.,  0.,  0.,  0.,  0.]])
```

```
result = metrics.pairwise.cosine_similarity(mean200,mean95)
result
```

```
array([[ 0.40572802]])
```

```
result2 = metrics.pairwise.cosine_similarity(mean15,mean95)
result2
```

```
array([[ 0.26612637]])
```

The problem uses recommendation system building based on prediction of the ratings. To get the prediction we used the mean average user ratings and found the similarity to the item/movie 95, we found that the similarity in ratings to movie 95 for both user 200 and 15.

User 200 similarity : 0.40572802
User 15 similarity : 0.26612637

Here user 200 has higher similarity which led the recommendation system choose user 200.

2.2 Problem 2

Load the Movielens 100k dataset (ml-100k.zip) into Python using Pandas dataframes. Convert the ratings data into a utility matrix representation, and the 10 most similar users for user 1 based on cosine similarity of the user ratings data. Based on the average of of the ratings for item 508 from the similar users, what is the expected rating for this item for user 1?

```
import numpy as np
import pandas as pd
import heapq

from sklearn import metrics


new_dict = dict()
cosine_similarity_list = []
similar_users_list = []


user_cols = ['user_id',
             'age',
             'gender',
             'occupation',
             'zip_code']
users = pd.read_csv('E:/ml-100k/u.user',
                    sep='|',
                    names=user_cols)
rating_cols = ['user_id',
               'movie_id',
               'rating',
               'timestamp']
ratings = pd.read_csv('E:/ml-100k/u.data',
                      sep='\t',
                      names=rating_cols)
```

```
item_cols = ['movie id',
             'movie title',
             'release date',
             'video release date',
             'IMDb URL',
             'unknown',
             'Action',
             'Adventure',
             'Animation',
             'Childrens',
             'Comedy',
             'Crime',
             'Documentary',
             'Drama',
             'Fantasy',
             'FilmNoir',
             'Horror',
             'Musical',
             'Mystery',
             'Romance',
             'SciFi',
             'Thriller',
             'War',
             'Western']
items = pd.read_csv('E:/ml-100k/u.item',
                    sep='|',
                    names=item_cols,
                    encoding='latin-1')

utility = ratings.pivot(index='user_id',
                        columns='movie_id',
                        values='rating')
```

```
user_means = utility.mean(axis=1)

utility_centered = utility - user_means
utility_centered = utility_centered.where((pd.notnull(utility_centered)), 0)

print("Utility")

print(utility_centered)
```

```
11      0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
12      0.000000  0.000000  0.000000  0.666667  0.000000  0.000000  0.000000
13     -0.610294 -0.709677  0.000000  0.666667 -1.874286  0.000000 -1.965261
14      0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  1.034739
15     -2.610294  0.000000  0.000000  0.000000  0.000000  0.000000 -2.965261
16      1.389706  0.000000  0.000000  0.666667  0.000000  0.000000  1.034739
17      0.389706  0.000000  0.000000  0.000000  0.000000  0.000000  0.034739
18      1.389706  0.000000  0.000000 -1.333333  0.000000  1.364929  0.000000
19      0.000000  0.000000  0.000000 -0.333333  0.000000  0.000000  0.000000
20     -0.610294  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
21      1.389706  0.000000  0.000000  0.000000 -0.874286  0.000000  1.034739
22      0.000000 -1.709677  0.000000  0.666667  0.000000  0.000000  0.000000
23      1.389706  0.000000  0.000000  0.000000  0.000000  0.000000  0.034739
24      0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.034739
25      1.389706  0.000000  0.000000  0.000000  0.000000  0.000000  0.034739
26     -0.610294  0.000000  0.000000  0.000000  0.000000  0.000000 -0.965261
27      0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
28      0.000000  0.000000  0.000000  0.000000  0.125714  0.000000  1.034739
29      0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000
30      0.000000 -0.709677  0.000000  0.000000  0.000000  0.000000  0.034739
```

```
user_1 = utility_centered[0:1]

for index, row in utility_centered.iterrows():
    np_row = np.array(row)
    np_row.resize(1, 1682)
    if index != 1:
        list = metrics.pairwise.cosine_similarity(user_1, np_row)
        cosine_similarity_list.append(list)
        new_dict.__setitem__(index, list)

similar_users_top_ten = heapq.nlargest(10, new_dict, key=new_dict.get)
user_columns = ['user_id' , '508']

data =0
for i in similar_users_top_ten:

        print("Item: ", i); print("Cosine Similarity ", new_dict.get(i));
        similar_users_list.append(utility_centered.loc[i, 508])
```

```
Item:  738
Cosine Similarity  [[ 0.29148679]]
Item:  592
Cosine Similarity  [[ 0.27840172]]
Item:  276
Cosine Similarity  [[ 0.26815054]]
Item:  267
Cosine Similarity  [[ 0.26476147]]
Item:  643
Cosine Similarity  [[ 0.2640026]]
Item:  757
Cosine Similarity  [[ 0.26236785]]
Item:  457
Cosine Similarity  [[ 0.26233704]]
Item:  606
Cosine Similarity  [[ 0.26084701]]
Item:  916
Cosine Similarity  [[ 0.25562438]]
Item:  44
Cosine Similarity  [[ 0.2529544]]
```

```
np_array_similar_users = np.array(similar_users_list)

print("Expected rating of user 1 for item 508:")
print(np_array_similar_users.mean())
```

```
Expected rating of user 1 for item 508:
0.268965517241
```

Hence, for User 1 ➔ item 508 ➔ expected Rating is 0.26896.