Recitation

Assignment 6

Exercise 9.2.1

a) w.r.t α *and* β

Cosine angle between A and B:

$$cos\theta 1 = \frac{((3.06*1)(2.68*1))+((500*\alpha)(320*\alpha))+((6*\beta)(4*\beta))}{\left(\sqrt{(3.06)^2+(500\alpha)^2+(6\beta)^2}\right)*\left(\sqrt{(2.68)^2+(320\alpha)^2+(4\beta)^2}\right)} = \frac{8.2008+\left(160000\alpha^2\right)+24\beta^2}{\left(\sqrt{9.3636+250000\alpha^2+36\beta^2}\right)*(\sqrt{7.1824+102400\alpha^2+16\beta^2})}$$

Cosine angle between B and C:

$$cos\theta 2 = \frac{7.8256+\left(204800\alpha^2\right)+24\beta^2}{\left(\sqrt{7.1824+102400\alpha^2+16\beta^2}\right)*(\sqrt{8.5264+409600\alpha^2+36\beta^2})}$$

Cosine angle between A and C

$$cos\theta 3 = \frac{8.9352+\left(320000\alpha^2\right)+(36\beta^2)}{(\sqrt{9.3636+250000\alpha^2+36\beta^2})(\sqrt{8.5264+409600\alpha^2+36\beta^2})}$$

b) For $\alpha = 1$ *and* $\beta = 1$

Cosine angle between A and B:

$$cos\theta 1 = \frac{8.2008+160000+24}{\left(\sqrt{9.3636+250000+36}\right)*(\sqrt{7.1824+102400+16})} = 1$$

So, the angle between A and B, $\theta 1 = 0$

Cosine angle between B and C:

$$cos\theta 2 = \frac{7.8256+204800+24}{\left(\sqrt{7.1824+102400+16}\right)*(\sqrt{8.5264+409600+36})} = 1$$

So, the angle between B and C, $\theta 2 = 0$

Cosine angle between A and C:

$cos\theta3 = \frac{8.9352+320000+36}{(\sqrt{9.3636+250000+36})*(\sqrt{8.5264+409600+36})} = 1$

$So,\ the\ angle\ between\ A\ and\ C,\ \theta3 = 0$

C) For $\alpha = 0.01\ and\ \beta = 0.5$, putting this value in the formula derived in (a) we get:

Cosine angle between A and B:

$cos\theta1 = 0.99$

$So,\ the\ angle\ between\ A\ and\ B,\ \theta1 = 8.1096$

Cosine angle between B and C

$cos\theta2 = 0.96$

$So,\ the\ angle\ between\ B\ and\ C,\ \theta2 = 16.2602$

Cosine angle between A and C

$cos\theta3 = 0.99$

$So,\ the\ angle\ between\ A\ and\ C,\ \theta3 = 8.1096$

d) $when\ \alpha\ and\ \beta\ inversely\ proportional\ to\ the\ average\ of\ their\ respective\ components.$

$\alpha = \frac{3}{500+320+640} = \frac{3}{1460}$

$\beta = \frac{3}{6+4+6} = \frac{3}{16}$

Cosine angle between A and B:

$$cos\theta1 = \frac{8.2008+\left(160000(\frac{3}{1460})^2\right)+24(\frac{3}{16})^2}{\left(\sqrt{9.3636+250000(\frac{3}{1460})^2+36(\frac{3}{16})^2}\right)*\sqrt{7.1824+102400(\frac{3}{1460})^2+16(\frac{3}{16})^2}} = 0.99$$

$So,\ the\ angle\ between\ A\ and\ B,\ \theta1 = 8.1096$

Similarly,

Cosine angle between B and C

$cos\theta2 = 0.99$

$So,\ the\ angle\ between\ B\ and\ C,\ \theta2 = 8.1$

Cosine angle between A and C

$cos\theta 3 = 0.99$

$So,\ the\ angle\ between\ A\ and\ C,\ \theta 3 = 8.1$

Exercise 9.2.3

a)

Average rating (μ) = $\frac{(4+2+5)}{3} = \frac{11}{3}$

Normalised rating for user A = $4 - \frac{11}{3} = \frac{1}{3}$

Normalised rating for user B = $2 - \frac{11}{3} = \frac{-5}{3}$

Normalised rating for user C = $5 - \frac{11}{3} = \frac{4}{3}$

b)

Value for main memory size = $\left(6 * \frac{1}{3}\right) + \left(4 * \frac{-5}{3}\right) + \left(6 * \frac{4}{3}\right) = 3.3333$

Value of the Processor speed = $\left(3.06 * \frac{1}{3}\right) + \left(2.68 * \frac{-5}{3}\right) + \left(2.92 * \frac{4}{3}\right) = 0.4467$

Value of disk size = $\left(500 * \frac{1}{3}\right) + \left(320 * \frac{-5}{3}\right) + \left(640 * \frac{4}{3}\right) = 486.667$

Exercise 9.3.1

a) Jaccard distance can be understood as follows:

Jaccard Distance = $\frac{Total\ Number\ of\ features\ for\ which\ both\ users\ have\ rated}{Net\ Total\ features\ rated\ by\ both\ the\ users\ (count\ each\ features\ just\ once)}$

thus,

Jaccard distance (A, B) = $\frac{4}{8} = \frac{1}{2}$

Jaccard distance (B, C) = $\frac{4}{8} = \frac{1}{2}$

Jaccard distance (A, C) = $\frac{4}{8} = \frac{1}{2}$

b) Cosine distance can be understood as follows:

Cosine Distance (X, Y) = 1 - Cosine Similarity(X,Y) i.e. Cosine Distance (X, Y) = 1-

$$\frac{\sum(Only\ product\ of\ feature\ ratings\ from\ X\ and\ Y\ where\ both\ users\ have\ rated\ for\ same\ feature)}{|X|*|Y|}$$

thus,

Cosine Distance (A, B) = 1 - $\frac{(5*3)+(5*3)+(1*1)+(3*1)}{\left(\sqrt{4^2+5^2+5^2+1^2+3^2+2^2}\right)*\sqrt{3^2+4^2+3^2+1^2+2^2+1^2}} = 1 - 0.601 = 0.399$

Cosine Distance (B, C) = 1 - $\frac{(4*1)+(3*3)+(2*4)+(1*5)}{\left(\sqrt{2^2+1^2+3^2+4^2+5^2+3^2}\right)*\sqrt{3^2+4^2+3^2+1^2+2^2+1^2}} = 1 - 0.514 = 0.486$

Cosine Distance (A, C) = 1 - $\frac{(4*2)+(5*3)+(3*5)+(2*3)}{\left(\sqrt{4^2+5^2+5^2+3^2+2^2+1^2}\right)*\sqrt{2^2+1^2+3^2+4^2+5^2+3^2}} = 1 - 0.615 = 0.385$

c) The updated utility matrix will looks like this:

	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

$Jaccard\ Distance\ (X, Y) = 1 - Jaccard\ Similarity(X, Y)$

Where,

$Jaccard\ Similarity(X, Y) = \frac{a}{a+b+c}$

a = the number of attributes that equal 1 for both objects X and Y

b = the number of attributes that equal 0 for objects X but equal 1 for object Y

c = the number of attributes that equal 1 for objects X but equal 0 for object Y

d = the number of attributes that equal 0 for both objects X and Y

thus,

Jaccard distance (A, B) = 1 $- \frac{2}{5} = 0.6$

Jaccard distance (B, C) = 1 $- \frac{1}{6} = 0.833$

Jaccard distance (A, C) = 1 $- \frac{2}{6} = 0.667$

d) Cosine Distance (X, Y) = 1-

$$\frac{\sum(Only\ product\ of\ feature\ ratings\ from\ X\ and\ Y\ where\ both\ users\ have\ rated\ for\ same\ feature)}{|X|*|Y|}$$

thus,

Cosine Distance (A, B) = 1 - $\frac{(1*1)+(1*1)}{\left(\sqrt{1^2+2^2+2^2+1^2+2^2}\right)*\sqrt{1^2+1^2+1^2+1^2+2^2+1^2}} = 1 - 0.5773 = 0.4227$

Cosine Distance (B, C) = 1 - $\frac{(1*1)+(1*1)+(2*2)+(1*2)}{\left(\sqrt{1^2+1^2+1^2+1^2+2^2+1^2}\right)*\sqrt{2^2+1^2+1^2+2^2+2^2+1^2}} = 1 - 0.2886 = 0.7114$

Cosine Distance (A, C) = 1 - $\frac{(1*2)+(2*1)+(1*2)+(2*1)}{\left(\sqrt{1^2+2^2+2^2+1^2+1^2+2^2}\right)*\sqrt{2^2+1^2+1^2+2^2+2^2+1^2}} = 1 - 0.5 = 0.5$

e) The updated normalized matrix will looks like this:

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**
A	2/3	5/3	0	5/3	-7/3	0	-1/3	-4/3
B	0	2/3	5/3	2/3	-4/3	-1/3	-4/3	0
C	-1	0	-2	0	0	1	2	0

f)

$$cosine\ Distance(A,B) = 1 - \frac{\left(\frac{5}{3}*\frac{2}{3}\right)+\left(\frac{5}{3}*\frac{2}{3}\right)+\left(-\frac{7}{3}*-\frac{4}{3}\right)+(-\frac{1}{3}*-\frac{4}{3})}{\sqrt{\left(\frac{2}{3}\right)^2+\left(\frac{5}{3}\right)^2+\left(\frac{2}{3}\right)^2+\left(-\frac{4}{3}\right)^2+\left(-\frac{1}{3}\right)^2+\left(-\frac{4}{3}\right)^2}\sqrt{\left(\frac{2}{3}\right)^2+\left(\frac{5}{3}\right)^2+\left(\frac{5}{3}\right)^2+\left(-\frac{7}{3}\right)^2+\left(-\frac{1}{3}\right)^2+\left(-\frac{4}{3}\right)^2}}$$

$= 1 - 0.584 = 0.416$

$$cosine\ Distance(B,C) = 1 - \frac{\left(\frac{5}{3}*\frac{-2}{1}\right)+\left(\frac{2}{3}*0\right)+\left(-\frac{1}{3}*1\right)+(-\frac{4}{3}*2)}{\sqrt{\left(\frac{2}{3}\right)^2+\left(\frac{5}{3}\right)^2+\left(\frac{2}{3}\right)^2+\left(-\frac{4}{3}\right)^2+\left(-\frac{1}{3}\right)^2+\left(-\frac{4}{3}\right)^2}\sqrt{(-1)^2+(-2)^2+(0)^2+(1)^2+(2)^2+(0)^2}}$$

$= 1 - (-0.739)$= 1.739

$$cosine\ Distance(A,B) = 1 - \frac{\left(\frac{2}{3}*-1\right)+\left(\frac{5}{3}*0\right)+\left(-\frac{1}{3}*2\right)+(0*-\frac{4}{3})}{\sqrt{(-1)^2+(-2)^2+(0)^2+(1)^2+(2)^2+(0)^2}\sqrt{\left(\frac{2}{3}\right)^2+\left(\frac{5}{3}\right)^2+\left(\frac{5}{3}\right)^2+\left(-\frac{7}{3}\right)^2+\left(-\frac{1}{3}\right)^2+\left(-\frac{4}{3}\right)^2}}$$

$= 1 - (-0.115)$= 1.115

Exercise 9.4.1

a) u_{32}:

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

The third user's contribution is as follows: $(x - 1)^2 + (x - 2)^2 + x^2 + (x - 3)^2$

By differentiating and equating to 0, then solving for x, we can find the minimum value for x:

$2 * ((x - 1) + (x - 2) + x + (x - 3)) = 0$

so we get x=1.5

So after first step,

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

b) V_{41}

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

The fourth column's contribution is as follows:

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

By differentiating and equating to 0, then solving for x, we can find the minimum value for y:

$2 * ((y - 3) + (y - 3) + y + (y - 2) + (y - 3)) = 0$

so we get y=2.2

After first step,

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

Practicum Problems

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import metrics
import math
```

Problem 1

Load data into dataframe

In [2]:
```
columns = [['user id','age','gender','occupation','zip code'],
           ['movie id','movie title','release date','video release date','IMDb URL','unknown','Action','Adventure
           ['user id','movie id','rating','timestamp']]

user_data = pd.read_csv('./data/u.user', sep='|', names = columns[0], encoding='latin-1')
item_data = pd.read_csv('./data/u.item', sep='|', names = columns[1], encoding='latin-1')
ratings_data = pd.read_csv('./data/u.data', sep='\t', names = columns[2], encoding='latin-1')
```

Visualize and understand data

In [3]:
```
print(user_data.shape, item_data.shape, ratings_data.shape)
```

```
(943, 5) (1682, 24) (100000, 4)
```

In [4]:
```
user_data.head()
```

Out[4]:

	user id	age	gender	occupation	zip code
0	1	24	M	technician	85711
1	2	53	F	other	94043
2	3	23	M	writer	32067
3	4	24	M	technician	43537
4	5	33	F	other	15213

In [5]:
```
item_data.head()
```

Out[5]:

	movie id	movie title	release date	video release date	IMDb URL	unknown	Action	Adventure	Animation	Childrens	...	Fantasy	Film-Noir	Horror
0	1	Toy Story (1995)	01-Jan-1995	NaN	http://us.imdb.com/M/title-exact?Toy%20Story%2...	0	0	0	1	1	...	0	0	0
1	2	GoldenEye (1995)	01-Jan-1995	NaN	http://us.imdb.com/M/title-exact?GoldenEye%20(...	0	1	1	0	0	...	0	0	0
2	3	Four Rooms (1995)	01-Jan-1995	NaN	http://us.imdb.com/M/title-exact?Four%20Rooms%...	0	0	0	0	0	...	0	0	0
3	4	Get Shorty (1995)	01-Jan-1995	NaN	http://us.imdb.com/M/title-exact?Get%20Shorty%...	0	1	0	0	0	...	0	0	0
4	5	Copycat (1995)	01-Jan-1995	NaN	http://us.imdb.com/M/title-exact?Copycat%20(1995)	0	0	0	0	0	...	0	0	0

5 rows × 24 columns

In [6]:
```
ratings_data.head()
```

Out[6]:

	user id	movie id	rating	timestamp
0	196	242	3	881250949
1	186	302	3	891717742
2	22	377	1	878887116
3	244	51	2	880606923
4	166	346	1	886397596

Make a utility matrix and center the data

In [7]:
```
utility_matrix = ratings_data.pivot(index='user id',columns='movie id',values='rating')
user_means = utility_matrix.mean(axis=1)
utility_centered = utility_matrix - user_means
utility_centered = utility_centered.where((pd.notnull(utility_centered)),0)
utility_centered
```

Out[7]:

1	2	3	4	5	6	7	8	9	10	...	1673	1674	1675	1676	1677

user id																
1	1.389706	-0.709677	1.203704	-1.333333	0.125714	1.364929	0.034739	-2.79661	0.727273	-1.206522	...	0.0	0.0	0.0	0.0	0.0
2	0.389706	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.000000	-2.206522	...	0.0	0.0	0.0	0.0	0.0
3	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.000000	0.000000	...	0.0	0.0	0.0	0.0	0.0
4	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.000000	0.000000	...	0.0	0.0	0.0	0.0	0.0
5	0.389706	-0.709677	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.000000	0.000000	...	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
939	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.727273	0.000000	...	0.0	0.0	0.0	0.0	0.0
940	0.000000	0.000000	0.000000	-2.333333	0.000000	0.000000	0.034739	1.20339	-1.272727	0.000000	...	0.0	0.0	0.0	0.0	0.0
941	1.389706	0.000000	0.000000	0.000000	0.000000	0.000000	0.034739	0.00000	0.000000	0.000000	...	0.0	0.0	0.0	0.0	0.0
942	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.000000	0.000000	...	0.0	0.0	0.0	0.0	0.0
943	0.000000	1.290323	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	-1.272727	0.000000	...	0.0	0.0	0.0	0.0	0.0

943 rows × 1682 columns

Save user 15 and 200

In [9]:

```
user_200 = user_data.iloc[199]
user_15 = user_data.iloc[14]
print(user_200)
print('\n',user_15)
```

```
user id              200
age                   40
gender                 M
occupation    programmer
zip code           93402
Name: 199, dtype: object

 user id            15
age                49
gender              F
occupation    educator
zip code         97301
Name: 14, dtype: object
```

Save item 95

In [10]:

```
item_95 = item_data[94:95]
print(item_95)
```

```
    movie id     movie title release date  video release date  \
94        95  Aladdin (1992)  01-Jan-1992                 NaN   

                                             IMDb URL  unknown  Action  \
94  http://us.imdb.com/M/title-exact?Aladdin%20(1992)        0       0   

    Adventure  Animation  Childrens  ...  Fantasy  Film-Noir  Horror  Musical  \
94          0          1          1  ...        0          0       0        1   

    Mystery  Romance  Sci-Fi  Thriller  War  Western  
94        0        0       0         0    0        0  

[1 rows x 24 columns]
```

Select Features for our Item 95

In [11]:

```
feat_select = item_95.iloc[:, 5:24]
feat_select.head()
```

Out[11]:

	unknown	Action	Adventure	Animation	Childrens	Comedy	Crime	Documentary	Drama	Fantasy	Film-Noir	Horror	Musical	Mystery	Roma
94	0	0	0	1	1	1	0	0	0	0	0	0	1	0	

Save item profile

In [12]:

```
item_profile = item_data.iloc[:,5:24]
item_profile.head()
```

Out[12]:

	unknown	Action	Adventure	Animation	Childrens	Comedy	Crime	Documentary	Drama	Fantasy	Film-Noir	Horror	Musical	Mystery	Roman
0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	
1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	1	0	0	0	1	0	0	1	0	0	0	0	0	

4	0	0	0	0	0	0	1	0	1	0	0	0	0	0

Generate User Profile by taking dot product of Item Profile and Utility Matrix

In [13]:
```
user_profile = np.dot(utility_centered,item_profile)
user_profile_200 = user_profile[199]
user_profile_15 = user_profile[14]
print("\nUser Profile:\n", user_profile)
print("\nUser Profile (200):\n", user_profile_200)
print("\nUser Profile (15):\n", user_profile_15)
```

```
User Profile:
 [[  0.06486486 -19.04561424 -23.97711089 ...   0.30691865   5.61156716
   -0.39044204]
 [  0.           0.16987665   2.03863023 ...  -0.51419764  -0.45999396
    0.        ]
 [  0.          -9.90027817   1.20553194 ... -24.45159726  -1.38051487
    0.        ]
 ...
 [  0.           2.68707838   3.3375375  ...   1.02512049   3.50804598
    0.        ]
 [  0.           9.95250865  12.35307555 ...   4.44379932  10.76957541
    2.60729178]
 [  0.           2.49861816  -7.48438072 ...   7.74176264   1.45769492
   -9.91893414]]

User Profile (200):
 [ 0.         53.68782567 35.4495237   7.6489686  14.16588957  8.80920192
  0.20447963 -1.78787879 34.25995913  3.6691485   2.07481203  8.48019395
  8.71830943  2.46827733 25.27766511 36.71256459 22.5206827  13.54465351
  1.98478731]

User Profile (15):
 [  0.         -18.26962989  -5.63815687  -5.52457983 -14.10394259
 -25.84333442  -6.31283599   0.         -24.23614665  -1.76218535
  -0.43397683  -6.48214286  -2.84021164  -6.19720898  -1.75310808
 -10.45977148 -31.15379592  -0.33176457   0.        ]
```

Cosine Similarity

In [14]:
```
cosine_sim = metrics.pairwise.cosine_similarity(user_profile,feat_select)
print("User 15")
print("Cosine Similarity:", cosine_sim[14])
print("Cosine Distance:", 1-cosine_sim[14])
print("\nUser 200")
print("Cosine Similarity:", cosine_sim[199])
print("Cosine Distance:", 1-cosine_sim[199])
```

```
User 15
Cosine Similarity: [-0.43632073]
Cosine Distance: [1.43632073]

User 200
Cosine Similarity: [0.21328933]
Cosine Distance: [0.78671067]
```

In conclusion, the system is more likely to recommend Movie 95 to User 200 as their distance to the recommendation axis is smaller than User 15.

Problem 2

Utility Matrix generated and centered in first part will be used further.

Save the Users from Centered Utility Matrix

In [15]:
```
user_1 = utility_centered.iloc[:1,]
user_other = utility_centered.iloc[1:,]
```

Find Top 10 Cosine Similar Users to the User 1:

In [16]:
```
cosine_sim = metrics.pairwise.cosine_similarity(user_1,user_other)
index = np.argpartition(cosine_sim, -10, axis=1)[:, -10:]
print(index)
```

```
[[ 42 274 641 590 755 265 604 455 914 736]]
```

Calculate Expected Rating

In [17]:
```
c,ratings = 0, [0,0,0,0,0,0,0,0,0,0]

for i in range(10):
    ratings[i] = user_other[508][index[0][i]]
    if ratings[i] != 0.0:
        c += 1
```

```
total = math.fsum(ratings)
mean = total / c
print("In Conclusion, the expected rating for the item for user 1 is: ",mean)
```

```
In Conclusion, the expected rating for the item for user 1 is:  0.1724137931034484
```