OLD DOMINION UNIVERSITY

CS 495: Introduction to Web Science
Instructor: Micheal L. Nelson, Ph.D
Fall 2014 Thursdays 4:20pm – 7:10pm ECSB 2120

Assignment # 8
Joseph Elder UIN: 00844802

Honor Pledge
I pledge to support the Honor System of Old Dominion University. I will refrain from any form of academic dishonesty or deception, such as cheating or plagiarism. I am aware that as a member of the academic community it is my responsibility to turn in all suspected violations of the Honor Code. I will report to a hearing if summoned

November 13, 2014

1 Assignment 8

1.1 Question 1

The goal of this project is to use the basic recommendation principles we have learned for user-collected data. You will modify the code given to you which performs movie recommendations from the MovieLense data sets.

The MovieLense data sets were collected by the GroupLens Research Project at the University of Minnesota during the seven-month period from September 19th, 1997 through April 22nd, 1998. It is available for download from http://www.grouplens.org/node/73

There are three files which we will use:

1. u.data: 100,000 ratings by 943 users on 1,682 movies. Each user has rated at least 20 movies. Users and items are numbered consecutively from 1. The data is randomly ordered. This is a tab separated list of

user id | item id | rating | timestamp

The time stamps are unix seconds since 1/1/1970 UTC.

Example:

196	242	3	881250949
186	302	3	891717742
22	377	1	878887116
244	51	2	880606923
166	346	1	886397596
298	474	4	884182806
115	265	2	881171488

2. u.item: Information about the 1,682 movies. This is a tab separated list of

movie id | movie title | release date | video release date | IMDb URL | unknown | Action | Adventure | Animation |Children's | Comedy | Crime | Documentary | Drama | Fantasy | Film-Noir | Horror | Musical | Mystery | Romance | Sci-Fi | Thriller | War | Western |

The last 19 fields are the genres, a 1 indicates the movie is of that genre, a 0 indicates it is not; movies can be in several genres at once. The movie ids are the ones used in the u.data data set.

Example:

161|Top Gun (1986)|01-Jan-1986||http://us.imdb.com/M/title-exact?Top%20Gun%20(1986)|0|1|0|0|0|0|0|0|0|0|0|0|0|1|0|0|0|0
162|On Golden Pond (1981)|01-Jan-1981||http://us.imdb.com/M/title-exact?On%20Golden%20Pond%20(1981)|0|0|0|0|0|0|0|0|1|0|0|0|0|0|0|0|0|0
163|Return of the Pink Panther, The (1974)|01-Jan-1974||http://us.imdb.com/M/title-exact?Return%20of%20the%20Pink%20Panther,%20The%20(1974)|0|0|0|0|0|1|0|0|0|0|0|0|0|0| 0|0|0|0|0

3. u.user: Demographic information about the users. This is a tab separated list of:

user id | age | gender | occupation | zip code

The user ids are the ones used in the u.data data set.

Example:
1|24|M|technician|85711
2|53|F|other|94043
3|23|M|writer|32067
4|24|M|technician|43537
5|33|F|other|15213

The code for reading from the u.data and u.item files and creating recommendations is described in the book Programming Collective Intelligence (check email for more details). You are to modify recommendations.py to answer the following questions. Each question your program answers correctly will award you 10 points. You must have the question answered completely correct; partial credit will only be awarded if your answer is very close to the correct one.

1.1.1 The Problem

What 5 movies have the highest average ratings? Show the movies and their ratings sorted by their average ratings.

1.1.2 The Solution

Python version 2.7.6 is used to answer all questions for this assignment. The file 'A8.py' contains the source code of the solution and generated two files. The first file 'u.critics' contains the processed information used to get answers. The file 'u.answers' will contain all answers to questions 1 through 10. To find the 5 movies with the highest average ratings, all movie ratings listed in 'u.data' must be processed. Python dictionaries are used to keep track of the data associated with each rating. Calculating the average for each movie requires the sum of ratings divided by the number of ratings. The total sum of all ratings for each movie id is stored in the totals dictionary. The number of ratings for each movie is stored in the count dictionary. For each movie id, which serves as the key for both dictionaries, the value of totals is divided by the count returning the average. The average for each movie id is stored in another dictionary called ratings. The results are reported in 'u.answers'. Since there was more than a five way tie for movies with a rating of 5.0 all movies with a rating of 5.0 are listed.

2.1.1 The Problem

What 5 movies received the most ratings? Show the movies and the number of ratings sorted by number of ratings.

2.1.2 The Solution

To find the 5 movies with the highest number of ratings the count dictionary is used from question 1. Since the count is required for calculating the average for each movie id, the total number of ratings for each movie id

already exists in that dictionary. It is reported in 'u.answers'.

3.1.1 The Problem

What 5 movies were rated the highest on average by women? Show
the movies and their ratings sorted by ratings.

3.1.2 The Solution

In order to find the movies with the highest average ratings by women first we must get all of the ids of women only. This is done by processing 'u.user' and collecting only the ids of users which are female designated an attribute being F as opposed to M. These ids are put into a list to be used soon. The file 'u.data' is then processed again, however only the ratings done by ids contained in the female ids list are collected. The average rating is found the same way as in question 1 using dictionaries for total, count, and ratings or average. Since there was more than a five way tie for movies with a rating of 5.0 all movies with a rating of 5.0 are listed. The results can be found in 'u.answers'.

4.1.1 The Problem

What 5 movies were rated the highest on average by men? Show
the movies and their ratings sorted by ratings.

4.1.2 The Solution

To find the movies with the highest average ratings by men, a very similar process is used as in question 3. A list of all male ids is generated by processing 'u.user'. The only difference is when collecting ratings from 'u.data' only the ratings done by users with ids contained in the male ids list. Since there was more than a five way tie for movies with a rating of 5.0 all movies with a rating of 5.0 are listed. The results can be found in 'u.answers'.

5.1.1 The Problem

What movie received ratings most like Top Gun? Which movie
received ratings that were least like Top Gun (negative correlation)?

5.1.2 The Solution

To find the movie with ratings most and least like Top Gun, the Euclidean Distance is taken by two dimensions, number of ratings and rating for each movie. These values already exist in the dictionaries created for question 1. By using the code supplied in the Collective Intelligence textbook the Euclidean distance is calculated for each movie and stored in a dictionary. Using the existing values and a similar sorting algorithm to the one used in

question 1 the most correlated and least correlated movies are found for this particular data set and my algorithm, to be Star Wars and The Lion King respectively. This can also be found in 'u.answers'.

6.1.1 The Problem

Which 5 raters rated the most films? Show the raters' IDs and the number of films each rated.

6.1.2 The Solution

To determine which 5 raters rated the most films, all ratings must be processed again. A dictionary is used to keep track of users and their number of ratings. The key for the dictionary is the user id where the value is their number of ratings. For every rating in 'u.data' the ratings count for the appropriate user id is incremented. The results can be found in 'u.answers'.

9.1.1 The Problem

What movie was rated highest on average by men over 40? By men under 40?

9.1.2 The Solution

To determine the movies rated the highest by men over and under 40, it must first be determined which users are men and which are women. The gender of user ids has already been determined for questions 3 and 4 so two lists already exist which contain the user ids of all men and women respectively. Each id in the list of men is then checked for age to be added to one of two additional lists. If age is 39 or below the user id will be placed in the men under 40 list if age is greater than or equal to 40 that user id is placed in the men over 40 list. To find the highest average movie ratings for each category, the data from 'u.data' is processed again in a similar method to questions 1,3,and 4. If the user id from 'u.data' is in the over or under list then ratings for that user will be evaluated. Two separate totals, count, and ratings dictionaries exist for men under 40 and above 40 so averages can be calculated for both categories. Since there was more than a five way tie for movies with a rating of 5.0 in both categories, all movies with a rating of 5.0 are listed. The results can be found in 'u.answers'.

10.1.1 The Problem

What movie was rated highest on average by women over 40? By women under 40?

10.1.2 The Solution

Highest average rating for women above and below the age of 40 is determined

using the same method as for men. Instead of using the list containing all male ids the list containing all female ids is used to calculate the ages of all female users. Two lists are created, one for women younger than 40 one for those who are older than 40. Two separate versions of the count, totals, and ratings dictionaries are necessary to compute highest average movie ratings. Since there was more than a five way tie for movies with a rating of 5.0 all movies with a rating of 5.0 are listed. The results can be found in 'u.answers'.

References

1. https://docs.python.org/2/