CS550: Massive Data Mining
Homework 1

Due 11:59pm Monday, February 20, 2023
Please see the homework file for late policy

Submission Instructions

Honor Code Students may have discussions about the homework with peers. However, each student must write down their solutions independently to show they understand the solution well enough in order to reconstruct it by themselves. Students should clearly mention the names of all the other students with whom they have discussions about the homework. Directly using the code or solutions obtained from the web or from others is considered an honor code violation. We check all the submissions for plagiarism and take the honor code seriously, and we hope students to do the same.

Discussions (People with whom you discussed ideas used in your answers): Abhishek Nayak, Anjali Menghwani

On-line or hardcopy documents used as part of your answers:

I acknowledge and accept the Honor Code.

*(Signed) Vikram Sahai Saxena*_____

If you are not printing this document out, please type your initials above.

Answer to Question 1

1. The source code is included in the submission as question_1.py.
2. Algorithm to tackle the problem:
   (a) The data is first read from the file 'soc-LiveJournal1Adj.txt'.
   (b) Next, the data is split into user ID and a list of friend IDs using 'generate_user_friend_tuples' function. The function "generate_user_friend_pairs' then takes a tuple containing a user ID and a list of friend IDs, and generates pairs of the form (user1, user2) along with a flag indicating whether the users are direct friends (having value 0) or indirect friends (having value 1). This is further used to identify mutual friends and filter the data.
   (c) All pairs with a value of 0, i.e., all pairs where the users are already friends are filtered out.
   (d) Then, list of recommended friends is generated for each user, by grouping the pairs by user ID.
   (e) Finally, every user is mapped in decreasing order of the number of mutual friends.
3. Recommendations for the users with following user IDs: 924, 8941, 8942, 9019, 9020, 9021, 9022, 9990, 9992, 9993 :

```
924 ['439', '2409', '6995', '11860', '15416', '43748', '45881']
8941 ['8943', '8944', '8940']
8942 ['8939', '8940', '8943', '8944']
9019 ['9022', '317', '9023']
9020 ['9021', '9016', '9017', '9022', '317', '9023']
9021 ['9020', '9016', '9017', '9022', '317', '9023']
9022 ['9019', '9020', '9021', '317', '9016', '9017', '9023']
9990 ['13134', '13478', '13877', '34299', '34485', '34642', '37941']
9992 ['9987', '9989', '35667', '9991']
9993 ['9991', '13134', '13478', '13877', '34299', '34485', '34642', '37941']
```

Answer to Question 2(a)

A drawback of using Confidence is that it ignores Pr(B). This is a drawback because, if B occurs in all the baskets, having A in any basket doesn't increase the chance of B in that basket, as the presence of B is not affected by the presence if A in a basket.

Lift and conviction do not suffer from this drawback as

$$lift(A \rightarrow B) = \frac{conf(A \rightarrow B)}{S(B)}$$

and

$$conv(A \rightarrow B) = \frac{1 - S(B)}{1 - conf(A \rightarrow B)}$$

where, both of the above are affected by the presence of B.
So, the value of lift is lowered when B occurs in all the baskets, and the value of lift is maximized when B occurs in only those baskets having A.
The value of conviction would be one, if B occurs in only those baskets that have A. Also, the conviction value will not be zero, if B occurs in each basket.

Answer to Question 2(b)

1. Confidence is not a symmetrical measure. This can be shown by the following counter example:
   Consider sets: {A,B}, {A,C}, {A,B,C}

$$Conf(A \rightarrow B) = \frac{Pr(A,B)}{Pr(A)} = \frac{2}{3}$$

$$Conf(B \rightarrow A) = \frac{Pr(A,B)}{Pr(B)} = 1$$

2. Lift is a symmetrical measure. This can be shown by the following proof:

$$Lift(A \rightarrow B) = \frac{Conf(A \rightarrow B)}{S(B)} = \frac{Pr(B \mid A)}{S(B)} = \frac{Pr(A,B)}{Pr(A)Pr(B)}$$

$$Lift(B \rightarrow A) = \frac{Conf(B \rightarrow A)}{S(B)} = \frac{Pr(A \mid B)}{S(B)} = \frac{Pr(A,B)}{Pr(A)Pr(B)}$$

3. Conviction is not a symmetrical measure. This can be shown by the following counter example:
   Consider sets: {D}, {A,B}, {A,C}, {A,B,C}

$$conv(A \rightarrow B) = \frac{1 - S(B)}{1 - conf(A \rightarrow B)} = \frac{1 - \frac{1}{2}}{1 - \frac{2}{3}} = \frac{3}{2}$$

$$conv(B \rightarrow A) = \frac{1 - S(A)}{1 - conf(B \rightarrow A)} = \frac{1 - \frac{3}{4}}{1 - 1} = \infty$$

Answer to Question 2(c)

1. Confidence will have a maximum value of 1 because it is a probability measure. So, if rules occur together, the confidence will be maximal and will be equal to 1. It can be shown to be desirable from the following example:
   Consider baskets: {A,B}, {A,B}, {C, D}.

$$Pr(B \mid A) = 1 \Rightarrow conf(A \rightarrow B) = 1$$

2. Lift value can change based on the denominator, making it not a desirable property. This can be shown by the following counter example:
   Consider baskets: {A,B}, {A,B}, {C, D}.

$$lift(B \rightarrow A) = \frac{1}{\frac{2}{3}} = \frac{3}{2}$$

$$lift(C \rightarrow D) = \frac{1}{\frac{1}{3}} = 3$$

   Both the above lifts are 100% rules, but they have different lift scores, one better than the other.

3. Conviction value becomes infinity if there are perfect implications, making it desirable. This can be shown by the following counter example:
   Consider baskets: {A,B}, {A,B}, {C, D}.

$$conv(A \rightarrow B) = \frac{1 - \frac{2}{3}}{1 - 1} = \infty$$

Answer to Question 2(d)

Confidence Scores:

```
[(('DAI93865',), 'FRO40251', 1.0),
 (('GRO85051',), 'FRO40251', 0.999176276771005),
 (('GRO38636',), 'FRO40251', 0.9906542056074766),
 (('ELE12951',), 'FRO40251', 0.9905660377358491),
 (('DAI88079',), 'FRO40251', 0.9867256637168141)]
```

Answer to Question 2(e)

Confidence Scores:

```
[(('DAI23334', 'ELE92920'), 'DAI62779', 1.0),
 (('DAI31081', 'GRO85051'), 'FRO40251', 1.0),
 (('DAI55911', 'GRO85051'), 'FRO40251', 1.0),
 (('DAI62779', 'DAI88079'), 'FRO40251', 1.0),
 (('DAI75645', 'GRO85051'), 'FRO40251', 1.0)]
```

Answer to Question 3(a)

Total number of columns having n rows and m 1's = $\binom{n}{m}$
Number of columns that have first k rows as 0 out of n rows = $\binom{n-k}{m}$

Therefore, the probability we get "don't know" as the min-hash value for this column is:

$$\frac{\binom{n-k}{m}}{\binom{n}{m}} = \frac{(n-k)!(n-m)!m!}{m!(n-k-m)!n!} = \frac{(n-k)}{n}\frac{(n-k-1)}{(n-1)}\cdots\frac{(n-k-m+1)}{(n-m+1)} < \left(\frac{n-k}{n}\right)^m$$

Answer to Question 3(b)

Given:

Eq(1): For large x, $\left(1-\frac{1}{x}\right)^{x} \approx \frac{1}{e} \Rightarrow \left(1-\frac{1}{x}\right)^{10x} \approx \frac{1}{e^{10}}$

Eq(2): Probability we get "don't know" $= \left(1-\frac{k}{n}\right)^{m}$

Comparing Eq(1) and Eq(2); we get:

$m = \frac{10n}{k} \Rightarrow k = \frac{10n}{m}$

Therefore, the smallest value of k that will assure this probability we get "don't know" is at most e^{-10} is $\frac{10n}{m}$

Answer to Question 3(c)

Consider two columns:

Column 1: $[0, 1, 1, 1]^T$
Column 2: $[0, 1, 0, 0]^T$

Jaccard similarity of above two columns = $\frac{1}{3}$.
If the first two rows are used for a random cyclic permutation, both columns' min-hash values will be similar. But, if we consider the bottom two rows from, then both the columns will have different min-hash values.
Therefore, the probability that a random cyclic permutation gives the same min-hash value for both the columns is $\frac{1}{2}$.
Hence, for two columns, the probability (over cyclic permutations only) that their min-hash values agree is not the same as their Jaccard similarity..