CSC 571-485B / SENG 480A: Summer 2017
Assignment 4
Due: June 28, 2017, 11:55 AM

1. (6 pt) Implement the TODO parts in MedlineAnalysis_post.java.

Submit all your source code and your outputs.

2. (6 pt) Here is a collection of twelve baskets. Each contains three of the six items 1 through 6.

{1, 2, 3} {2, 3, 4} {3, 4, 5} {4, 5, 6}
{1, 3, 5} {2, 4, 6} {1, 3, 4} {2, 4, 5}
{3, 5, 6} {1, 2, 4} {2, 3, 5} {3, 4, 6}

Suppose the support threshold is 4. On the first pass of the PCY Algorithm we use a hash table with 11 buckets, and the set {i, j} is hashed to bucket i × j mod 11.
a) By any method, compute the support for each item and each pair of items.

1 (4), 2 (6), 3 (8), 4 (8), 5 (6), 6 (4)

1 2 (2), 1 3 (3), 1 4 (2), 1 5 (1)

2 3 (3), 2 4 (4), 2 5 (2), 2 6 (1)

3 4 (4), 3 5 (4), 3 6 (2)

4 5 (3), 4 6 (3)

5 6 (2)

b) Which pairs hash to which buckets? Which buckets are frequent?

	B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10
Pairs		2 6, 3 4,	1 2, 4 6,	1 3,	1 4, 3 5,	1 5,	2 3,	3 6,	2 4, 5 6,	4 5,	2 5,
Counts	0	5	5	3	6	1	3	2	6	3	2

Frequent buckets are: B1, B2, B4, B8.

c) Which pairs are counted on the second pass of the PCY Algorithm?
2 6, 3 4, 1 2, 4 6, 1 4, 3 5, 2 4, 5 6

Suppose now we run the Multistage Algorithm. The first pass is the same previously, and for the second pass, we hash pairs to nine buckets, using the hash function that hashes

{i, j} to bucket i + j mod 9.
d) Determine the counts of the buckets on the second pass. Does the second pass reduce the set of candidate pairs?

	B0	B1	B2	B3	B4	B5	B6	B7	B8
Pairs		4 6,	5 6,	1 2,		1 4,	2 4,	3 4,	2 6, 3 5,
Counts	0	3	2	2	0	2	4	4	5

Only 2 4, 3 4, 2 6, 3 5 survive after the second pass.
Yes, the second pass reduces the set of candidate pairs.

Note that all items are frequent, so the only reason a pair would not be hashed on the second pass is if it hashed to an infrequent bucket on the first pass.

Suppose now we run the Multihash Algorithm. We shall use two hash tables with five buckets each. For one, the set {i, j}, is hashed to bucket 2i+3j +4 mod 5, and for the other, the set is hashed to i + 4j mod 5. Since these hash functions are not symmetric in i and j, order the items so that i < j when evaluating each hash function.
e) Determine the counts of each of the 10 buckets.

	B0	B1	B2	B3	B4
Pairs	1 3, 2 4, 3 5, 4 6,	1 5, 2 6	1 2, 2 3, 3 4, 4 5, 5 6,	1 4, 2 5, 3 6,	
Counts	14	2	14	6	0

	B0	B1	B2	B3	B4
Pairs		1 5, 2 6,	1 4, 2 5, 3 6,	1 3, 2 4, 3 5, 4 6,	1 2, 2 3, 3 4, 4 5, 5 6,
Counts	0	2	6	14	14

f) How large does the support threshold have to be for the Multihash Algorithm to eliminate more pairs than the PCY Algorithm would in this example?

In this particular example, for any support threshold, the PCY algorithm doesn't do worse than the Multihash Algorithm.

3. (**3 pts**) Apply Toivonen's Algorithm to the data of the previous exercise with a support threshold of 4. Take as the sample the first row of baskets:
{1, 2, 3}, {2, 3, 4}, {3, 4, 5}, and {4, 5, 6}, i.e., one-third of the file.
Our scaled-down support threshold will be 1.

a) What are the itemsets frequent in the sample?
1, 2, 3, 4, 5, 6
1 2, 1 3, 2 3, 2 4, 3 4, 3 5, 4 5, 4 6, 5 6
1 2 3, 2 3 4, 3 4 5, 4 5 6

b) What is the negative border?

1 4, 1 5, 1 6, 2 5, 2 6, 3 6

c) What is the outcome of the pass through the full dataset?

1 (4), 2 (6), 3 (8), 4 (8), 5 (6), 6 (4)
1 2 (2), 1 3 (3), 2 3 (3), 2 4 (4), 3 4 (4), 3 5 (4), 4 5 (3), 4 6 (3), 5 6 (2)
1 2 3 (1), 2 3 4 (1), 3 4 5 (1), 4 5 6 (1)

1 4 (2), 1 5 (0), 1 6 (0), 2 5 (2), 2 6 (1), 3 6 (2)

Are any of the itemsets in the negative border frequent in the whole?

No.
So, we don't need another run of the algorithm and the frequent itemsets are:
1 (4), 2 (6), 3 (8), 4 (8), 5 (6), 6 (4)
2 4 (4), 3 4 (4), 3 5 (4)

4. (**4 pts**) Consider the example of slide 32 of similarity.pdf

a. What is the probability that we miss a pair of similar columns when the (Jaccard) similarity threshold is 60%? The values of *b* and *r* are as in the example: 20 and 5, respectively.

Probability C_1, C_2 agree on one particular band:
$(0.6)^5 = .07776$.
Probability C_1, C_2 do *not* agree on any of the 20 bands:
(1-.07776)^20 ~ .20 i.e., we miss a lot

b. How should we change *b* and *r* such that the probability of missing a pair of similar columns (for a 60% threshold) is about 1/3000?

Let b=33, r=3
Probability C_1, C_2 agree on one particular band:
$(0.6)^3 = .216$
Probability C_1, C_2 do *not* agree on any of the 50 bands:
(1-.216)^33 » .00032
i.e. we miss about 1/3000th of the 60%-similar column pairs.