Homework#2

ACS577 Knowledge Discovery and Data Mining, Summer II 2023

Due: July 12

- For your homework2 submission, prepare a single file, *YourLastName_FirstName*_ACS575_HW2.zip
- Organize the submission file with Part I and Part II and clearly number your answer with the question number.
- For each problem solving question, give your answer and also show the steps of computation to get the answer, if any.

Part I. Problem Solving

Q1 – Q5. Consider the transaction dataset below. Assume a lexicographic ordering of items.

TID	Items
1	{M, O, N, K, E, Y}
2	{D, O, N, K, E, Y}
3	{M, A, K, E}
4	{U, C, K, Y}
5	{C, O, K, I, E}

Q1. Show the procedure to find all frequent itemsets from the above transaction dataset using **Apriori** algorithm. Assume the *min_support* threshold is 0.6 (*min_support_count*=3).
For each level (*k* = 1, 2, …) process of Apriori, show **(1) the candidate itemsets generated**, **(2) the candidate itemsets after Apriori pruning**, and **(3) the frequent itemsets** .

Q2. Show the procedure to find all frequent itemsets from the above transaction dataset using **FP-growth** (with FP-tree) algorithm. Assume the *min_support* threshold is 0.6 (*min_support_count*=3).

Show **(1) F-list**, **(2) the transaction data with ordered frequent items**, **(3) FP-tree**, **(4) Conditional pattern bases, (5) Conditional FP-tree per each pattern base**, and **(6) the frequent itemsets generated from each conditional FP-tree.**

Q3. Among the frequent items found from either Q1 or Q2, find **(1) all maximal frequent itemsets** and **(2) all closed frequent itemsets** when we use the same the *min_support* threshold is 0.6 (*min_support_count*=3).

Q4. From the result from either Q1 or Q2, we know that {E, K, O} is a frequent itemset. **Find all strong rules generated with the itemset {E, K, O}** when the confidence threshold, *min_confidence,* is 0.7. Present each rule with its support and its confidence.

Q5. Compute the interesting measures of each rule **(1)** {K} → {Y} and **(2)** {Y} → {K} using (i) ***Support,*** (ii) ***Confidence***, (iii) ***Lift***, (iv) ***Leverage***, (v) ***Conviction***

Q6. Answer the following questions using the following data sets.

Note that each data set contains 1000 items and 10,000 transactions. Dark cells indicate the presence of items and white cells indicate the absence of items. We will apply the *Apriori* algorithm to extract frequent itemsets with *min_support* = 0.1 (i.e., itemsets must be contained in at least 1000 transactions)?

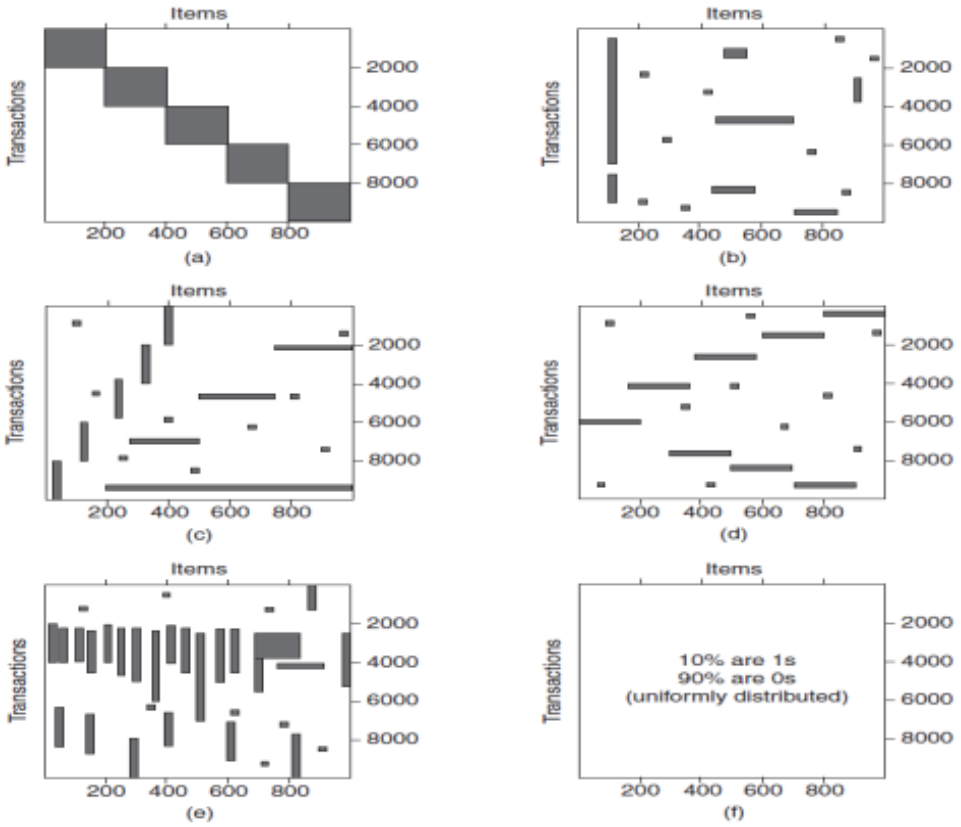
(1) Which data set(s) will produce the most number of frequent itemsets? Explain your answer.

(2) Which data set(s) will produce the fewest number of frequent itemsets? Explain your answer.

(3) Which data set(s) will produce the longest frequent itemset? Explain your answer.

(4) Which data set(s) will produce frequent itemsets with highest maximum support? Explain your answer.

(5) Which data set(s) will produce frequent itemsets containing items with wide-varying support levels (i.e., items with mixed support, ranging from less than 0.2 to more than 0.7). Explain your answer.



Q7. Find all the frequent subsequences with a support threshold of 50% (*min_support* =0.5) given the sequence database shown in a table below. Assume that there are no timing constraints imposed on the sequences.

Sensor	Timestamp	Events
S1	1	A, B
	2	C
	3	D, E
	4	C
S2	1	A, B
	2	C, D
	3	E
S3	1	B
	2	A
	3	B
	4	D, E
S4	1	C
	2	D, E
	3	C
	4	E
S5	1	B
	2	A
	3	B, C
	4	A, D

Part II. Hands-on practice

P1. The purpose of this practice is to get familiar with R scripts for association rule mining. Follow "Ch 9. Association Rule" tutorial from Zhao, "R and Data Mining". The tutorial document is attached.

The tutorial uses the *titanic* dataset, https://www.kaggle.com/competitions/titanic/data . The tutorial uses reconstructed titanic data (*titanic.raw.rdata*) where each row represents a person. The data is also available from https://github.com/ethen8181/machine-learning/blob/master/association_rule/R/titanic.raw.rdata

Submit: (1) Your program codes, e.g., the R script codes in the tutorial document (Ch 9.2 – Ch 9.6)
(2) A proof to show the successful execution of your codes, e.g., screen shots on running, and the output.

Alternatively, you can show the same results using Python or any other programming language you prefer.

P2 – P3. Download a program which implements Apriori algorithm from https://borgelt.net/apriori.html.

apriori.exe (268 kb) Windows console executable

You can run the program file in Command prompt. To see available parameters for running the program, run it as

```
c:\> apriori
```

P2. Download a mushroom dataset from UCI repository, https://archive.ics.uci.edu/dataset/73/mushroom
If you unzip the *mushroom.zip*, you can see two main files, *agaricus-lepiota.data* and *agaricus-lepiota.names*. Review the data file and also the data description.

Using the apriori program downloaded, generate the following frequent item sets from the mushroom data with a support threshold of **85%** (*min_support*=0.85). Suppose we are interested in patterns of **size 2 to size 5.** Submit your answer for each question.

(1) Generate **frequent itemsets** with the condition given.

For example, `apriori -ts -s85 -m2 -n5 agaricus-lepiota.data frequent_0.85.output`

(2) Generate **closed itemsets** with the condition given.

(3) Find **maximal itemsets from closed itemsets in (2)** by hand**.**

(4) To increase the readability of the output patterns, rewrite the maximal itemsets (from (3)) with presenting each item with the original attribute and its value, e.g., A frequent itemset {*c*, *f*, *p*} is presented with {cap-shape=*conical*, cap-surface=*fibrous*, cap-color=*pink*}. You need to refer to the data description for that.

P3. Download an adult dataset from UCI repository, https://archive.ics.uci.edu/dataset/2/adult
If you unzip the *adult.zip*, you can see two main files, *adult.data* and *adult.names*. Review the data file and also the data description.

(1) Using any programming langue or tool you prefer, convert the *adult.data* to a file named *adult_transaction.data* as follows:

- Include all attributes except **fnlwgt.**
- Convert the continuous attributes **age**, **capital-gain**, **capital-loss**, and **hours-per-week** to categorical attributes using the following rules;

```
if      (age  <   30) "young";
else if (age  <   50) "middle-aged";
else if (age  <   70) "senior";
else                  "old";

if      (capital-gain <=   0) "none ";
else if (capital-gain < 2000) "small ";
else if (capital-gain < 5000) "medium ";
else                          "high ";

if      (capital-loss <=   0) "none ";
else if (capital-loss < 2000) "small ";
else if (capital-loss < 5000) "medium ";
else                          "high ";

if      (hours-per-week <=  25) "half-time";
else if (hours-per-week <=  40) "full-time";
else if (hours-per-week <=  60) "overtime";
else                            "too-many";
```

Refer to the attached sample data, *adult_transaction_sample.data*. Submit the *adult_transaction.data* you generated.

(2) Using the apriori program (used in P2), find **maximal itemsets** with support threshold **60%** from *adult_transaction.data*. Submit the result.

(3) Using the apriori program, find **all association rules** with support threshold **60%** and confidence threshold **90%** from *adult_transaction.data*. Submit the result.