CS 412 Data Mining (22Sp): Assignment 2

Xu Ke (UIN: 675776713 NetID: kex5)

Feb. 18, 2022

1 Problem 1. OLAP

We would like to build a data cube of the fruit data, and want to include the following dimensions and measures:

- 4 dimensions: (Variety, Location, Size, Maturity)
- 5 measures: (Price.mean, Price.standard deviation, Price.IQR, Fruit.num eq min, Price.sum of top5)

It indicates that the fruit price is a function of variety, location, size and maturity. Suppose we are aggregating from the Maturity 1-D cuboid to the apex cuboid. The meanings of these measures are explained below.

1.1 Show that Price.mean is an algebraic measure. What distributive measures do you need to use in calculating Price.mean? Here Price.mean denotes the mean value of all prices.

According to the definition of algebraic measure, if the result can be computed by an algebraic function with M arguments (where M is bounded integer), each of which is obtained by applying a distributive aggregate function. In this given case, we can easily calculate $Price.mean$ by $\frac{sum(Price)}{count(Price)}$, a fraction of two distributive measures $\boxed{sum(Price)}$ and $\boxed{count(Price)}$ obviously.

1.2 Show that Price.standard deviation is an algebraic measure. What distributive measures do you need in calculating Price.standard deviation? Here Price.standard deviation denotes the standard deviation of all prices.

According to the definition of algebraic measure, if the result can be computed by an algebraic function with M arguments (where M is bounded integer), each of which is obtained by applying a distributive aggregate function. In this given case, we can easily calculate $Price.standarddeviation$ by $\sqrt{\frac{sum(Price^2)}{count(Price)} - (\frac{sum(Price)}{count(Price)})^2}$, thus the distributive measures will be clearly $\boxed{sum(Price)}$, $\boxed{sum(Price^2)}$ and $\boxed{count(Price)}$.

1.3 Explain whether Price.IQR is an algebraic measure or holistic measure. Please justify your answer. Here Price.IQR means inter-quartile range of all prices.

According to the definition of holistic measure, if there is no constant bound on the storage size needed to describe a subaggregate. In the given case, above all, we can first partition the Price into two equal-size subset namely $Price_0$ and $Price_1$ where $Price_0$ contains the smaller price than $Price_1$, thus $Q3 = median(Price_1)$ and $Q1 = median(Price_0)$. And we know that $IQR = Q3 - Q1$ where Q3 and Q1 are both holistic measures, thus Price.IQR should also be $\boxed{holistic}$ measure.

1.4 Explain whether Fruit.num eq min is an algebraic measure or holistic measure. Please justify your answer. Here Fruit.num eq min means the number of fruits that have the lowest price. For example, given 5 fruits with prices: 10, 11, 23, 10, 23, then num eq min = 2.

According to the definition of algebraic measure, if the result can be computed by an algebraic function with M arguments (where M is bounded integer), each of which is obtained by applying a distributive aggregate function.

In this given case, we can easily calculate $Fruit.numeqmin$ by $count(min(Price))$ where count() and min() are both distributive measures, thus Fruit.num eq min should be an $\boxed{algebraic}$ measure.

1.5 Explain whether Price.sum of top5 is an algebraic measure or holistic measure. Please justify your answer. Here Price.sum of top5 means the total price of the fruits with the highest 5 price values. For example, given 6 fruits with prices: 10, 11, 23, 18, 16, 40, then sum of top5 = 11 + 23 + 18 + 16 + 40 = 108.

According to the definition of algebraic measure, if the result can be computed by an algebraic function with M arguments (where M is bounded integer), each of which is obtained by applying a distributive aggregate function. In this given case, we can calculate the sum of top5 by $sum(max_5(Price))$ where sum() is distributive and $max_5()$ is algebraic (Textbook Chapter 4 Page 27), thus Price.sum of top5 is an $\boxed{algebraic}$ measure.

2 Problem 2. Data Cube Concepts

Suppose we have a data cube with 9 dimensions. The base cuboid of this data cube contains two cells:
$(a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9) : 1$,$(a_1, b_2, a_3, b_4, a_5, b_6, a_7, b_8, a_9) : 1$
where $a_i \neq b_i$ for i = 2, 4, 6, 8. Assume each dimension contains no concept hierarchy.

2.1 How many cuboids are there in this data cube?

Since the data cube has 9 dimensions, thus the number of cuboids will be 2^9 that is $\boxed{512}$. (Lecture 3 Page 25)

2.2 Please list all the (nonempty) closed cells in this data cube.

According to the lecture notes (Lecture 3 Page 46), this data cube will have 3 closed cells:
$\boxed{(a_1, *, a_3, *, a_5, *, a_7, *, a_9) : 2, (a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9) : 1, (a_1, b_2, a_3, b_4, a_5, b_6, a_7, b_8, a_9) : 1}$.

2.3 How many (nonempty) aggregate cells are there in this data cube?

According to the lecture notes (Lecture 3 Page 44), the total number of aggregate cells will be the total number of cuboids exclude two base cell and the redundant cells, $2 \cdot (2^9 - 1) - 2^5$, which is $\boxed{990}$.

2.4 How many (nonempty) aggregate closed cells are there in this data cube? Please list them.

The only aggregate closed cell will be $\boxed{(a_1, *, a_3, *, a_5, *, a_7, *, a_9) : 2}$.

2.5 If we set minimum support = 2, how many (nonempty) aggregate cells are there in the corresponding iceberg cube?

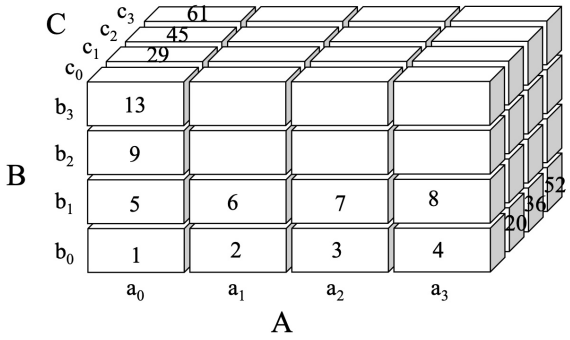
According to the lecture notes (Lecture 3 Page 45), the total number of iceberg cells will be, 2^5, which is $\boxed{32}$.

2.6 What are the differences among star schema, snowflake schema, or fact constellations for modeling the data warehouses? Which schema do you suggest to model this cube? Please justify your answer.

According to the definition in the lecture notes(Lecture 3 Page 19-21), star schema is a fact table in the middle connected to a set of dimension tables; snowflake schema is a refinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to snowflake; and a fact constellation is multiple fact tables share dimension tables, viewed as a collection of stars. I would like to suggest $\boxed{star}$ schema to model this cube since in this given case, we only have a fact table with 9 dimensions and there is no hierarchy.

3 Problem 3. Data Cube Computation

Assume our data is stored in a data cube with 3 dimensions A, B and C. We would like to do full cube computation using multi-way array aggregation. The lengths of dimensions A and B are 2000 and 400, respectively. The length of dimension C is an unknown value x (x > 0). We cut each dimension into quarters and get 64 chunks as follows.



3.1 If we follow the scan order 1-2-3-4-5-6-···, what is the memory requirement to compute the whole cube?

According to the lecture notes (Lecture 3 Page 52), in this given case the number of memory required to compute the whole cube will be the entire AB plane, one column of AC plane and one chunk of BC plane, that is:

$$2000 * \frac{x}{4}(AC) + 100 * \frac{x}{4}(BC) + 2000 * 400(AB) = \boxed{525x + 800000} \tag{1}$$

3.2 If we follow the scan order 1-5-9-13-2-6-···, what is the memory requirement to compute the whole cube?

According to the lecture notes (Lecture 3 Page 55), in this given case the number of memory required to compute the whole cube will be the entire AB plane, one column of BC plane and one chunk of AC plane, that is:

$$500 * \frac{x}{4}(AC) + 400 * \frac{x}{4}(BC) + 2000 * 400(AB) = \boxed{225x + 800000} \tag{2}$$

3.3 If we follow the scan order 1-17-33-49-5-21-···, what is the memory requirement to compute the whole cube?

According to the lecture notes (Lecture 3 Page 56), in this given case the number of memory required to compute the whole cube will be the entire BC plane, one column of AC plane and one chunk of AB plane, that is:

$$2000 * \frac{x}{4}(AC) + 400 * x(BC) + 500 * 100(AB) = \boxed{900x + 50000} \tag{3}$$

3.4 Given the scan orders 1-2-3-4-5-6-··· and 1-5-9-13-2-6-···, which one is more memory-saving? Please justify your answer.

As shown above, for the scan order 1-2-3-4-5-6-· · ·, the number of memory required will be 525x+800000 and for the scan order 1-5-9-13-2-6-· · ·, the number of memory required will be 225x+800000. And since we definitely have $x > 0$, $525x + 800000 > 225x + 800000$ will always hold, thus the scan order $\boxed{1 - 5 - 9 - 13 - 2 - 6 - \cdots}$ will be more memory-saving.

3.5 Besides the scan orders in (1)-(3), are there any other scan orders? If yes, please list all the other scan orders by specifying the first 10 chunks in their orders (you can assume the order always starts from chunk 1.). If not, please justify your answer.

$\boxed{Yes}$ Actually intuitively for a 3-D block we can figure out that we can scan from any one of three dimension to another one of dimension left, that is totally we should have $3 * 2 = 6$ scan orders, so in this given case, the other three scan orders will be:

$$\boxed{1 - 2 - 3 - 4 - 17 - 18 - 19 - 20 - 33 - 34 - \cdots} \tag{4}$$

$$\boxed{1 - 5 - 9 - 13 - 17 - 21 - 25 - 29 - 33 - 37 - \cdots} \tag{5}$$

$$\boxed{1 - 17 - 33 - 49 - 2 - 18 - 34 - 50 - 3 - 19 - \cdots} \tag{6}$$

4 Problem 4. Pattern Mining Concepts

Suppose that a transaction database has only three transactions:
$T_1 = \{a_{10}, a_{11}, \cdots, a_{20}\}$, $T_2 = \{a_{10}, a_{11}, \cdots, a_{30}\}$, $T_3 = \{a_1, a_2, \cdots, a_{30}\}$

4.1 If the minimum support threshold is 1, what are the number of the closed and maximal frequent itemsets in this database, respectively?

According to the lecture notes (Lecture 4 Page 14), a pattern X is closed if X is frequent, and there exists no super-pattern $X \in Y$ with the same support X. Thus, in this case we will have $\boxed{3}$ closed frequent itemsets, which are: $\{a_{10}, a_{11}, \cdots, a_{20}\}$:3, $\{a_{10}, a_{11}, \cdots, a_{30}\}$:2, $\{a_1, a_2, \cdots, a_{30}\}$:1.

According to the lecture notes (Lecture 4 Page 16), a pattern X is max-pattern if X is frequent and there exists no frequent super-pattern $X \in Y$. Thus, in this case we will have only $\boxed{1}$ maximal frequent itemset, which is: $\{a_1, a_2, \cdots, a_{30}\}$:1.

4.2 If the minimum support threshold is 2, what are the number of the closed and maximal frequent itemsets in this database, respectively?

According to the lecture notes (Lecture 4 Page 14), a pattern X is closed if X is frequent, and there exists no super-pattern $X \in Y$ with the same support X. Thus, in this case we will have $\boxed{2}$ closed frequent itemsets, which are: $\{a_{10}, a_{11}, \cdots, a_{20}\}$:3, $\{a_{10}, a_{11}, \cdots, a_{30}\}$:2.

According to the lecture notes (Lecture 4 Page 16), a pattern X is max-pattern if X is frequent and there exists no frequent super-pattern $X \in Y$. Thus, in this case we will have only $\boxed{1}$ maximal frequent itemset, which is: $\{a_{10}, a_{11}, \cdots, a_{30}\}$:2.

4.3 If the minimum support threshold is 3, what are the number of the closed and maximal frequent itemsets in this database, respectively?

According to the lecture notes (Lecture 4 Page 14), a pattern X is closed if X is frequent, and there exists no super-pattern $X \in Y$ with the same support X. Thus, in this case we will have $\boxed{1}$ closed frequent itemsets, which are: $\{a_{10}, a_{11}, \cdots, a_{20}\}$:3.

According to the lecture notes (Lecture 4 Page 16), a pattern X is max-pattern if X is frequent and there exists no frequent super-pattern $X \in Y$. Thus, in this case we will have only $\boxed{1}$ maximal frequent itemset, which is: $\{a_{10}, a_{11}, \cdots, a_{20}\}$:3.

5 Problem 5. Pattern Mining Methods

Given the transaction database below, please answer the following questions.

Tid	Items
1	B, C, E, F, G
2	A, B, C, F
3	A, B, C
4	E, F
5	A, B, G
6	B, C, D, E
7	A, B, C, D
8	A, C
9	A, B, G
10	A, D, E, F, G

5.1 Given an association rule B → C(s,c), what are its relative support s and confidence c?

According to the lecture notes (Lecture 4 Page 14) about the association rule, we will have:

$$s\{B, C\} = \frac{5}{10} = 50\% \tag{7}$$

$$c = \frac{sup(B, C)}{sup(B)} = \frac{5}{7} \tag{8}$$

5.2 Find all frequent itemsets using Apriori algorithm, when the minimum relative support is 0.5. Please show intermediate steps to get all credits.

According to lecture notes (Lecture 4 Page 22-27), we will need to do the following step by step and recursively:

5.2.1 Step 1: Scan to get candidate 1-itemset C_1

Itemset	sup
A	0.7
B	0.7
C	0.6
D	0.3
E	0.4
F	0.4
G	0.4

5.2.2 Step 2: Choose frequent 1-itemset F_1 with min-support

Itemset	sup
A	0.7
B	0.7
C	0.6

5.2.3 Step 3: Perform second scan to get candidate 2-itemset C_2

Itemset	sup
A,B	0.5
A,C	0.4
B,C	0.5

5.2.4 Step 4: Choose frequent 2-itemset F_2 with min-support

Itemset	sup
A,B	0.5
B,C	0.5

5.2.5 Step 5: Perform third scan to get candidate 3-itemset C_3

Itemset	sup
A,B,C	0.3

5.2.6 Step 6: Choose frequent 3-itemset F_3 with min-support

None, thus exit recursion. Thus, all frequent itemsets are $\boxed{\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}}$.

5.3 Construct the FP-tree from the transaction database above, when the minimum relative support is 0.5. Please show intermediate steps to get all credits. Note that you can use any software (hand drawing is also allowed) to draw the FP-tree and insert a screenshot of this FP-tree into the submitted file.

5.3.1 Step 1: Scan DB once and find single-item frequent pattern

Since in this case, min_support = 0.5, thus A: 0.7, B: 0.7, C: 0.6 are single-item frequent patterns.

5.3.2 Step 2: Sort frequent items in frequency descending order

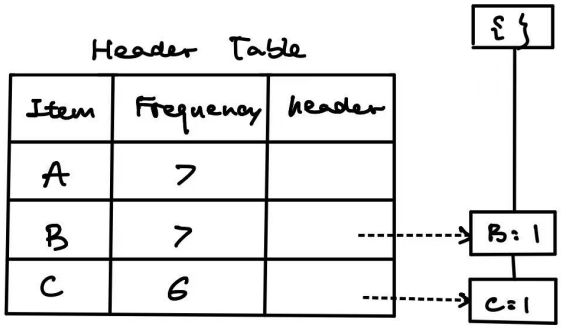
We will get F-list = A-B-C.

5.3.3 Step 3: Scan DB again, find the ordered frequent itemlist for each transaction

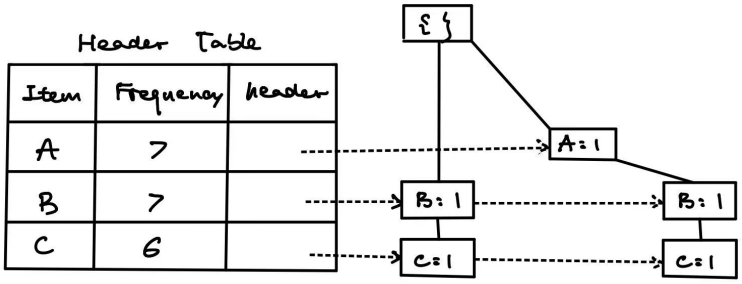
Tid	Items in the Transaction	Ordered, frequent itemlist
1	B,C,E,F,G	B,C
2	A,B,C,F	A,B,C
3	A,B,C	A,B,C
4	E,F	
5	A,B,G	A,B
6	B,C,D,E	B,C
7	A,B,C,D	A,B,C
8	A,C	A,C
9	A,B,G	A,B
10	A,D,E,F,G	A

5.3.4 Steep 4: For each transaction, insert the ordered frequent itemlist into an FP-tree, with shared sub-branches merged, counts accumulated

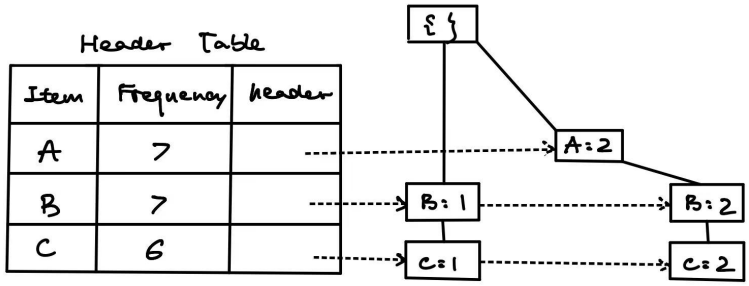
After inserting Tid 1, we will get:



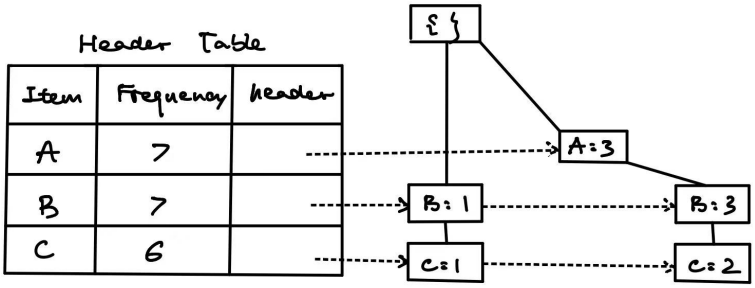
After inserting Tid 2, we will get:



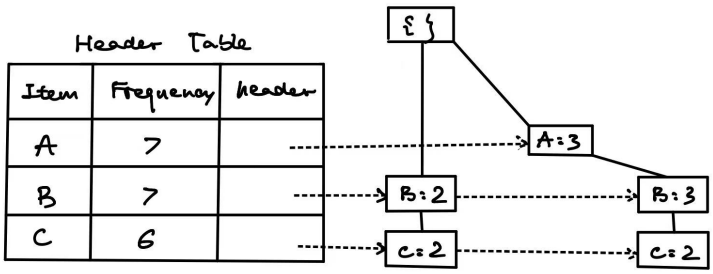
After inserting Tid 3 and Tid 4, we will get:



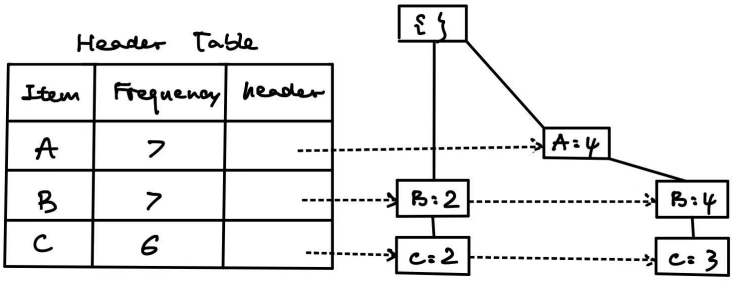
After inserting Tid 5, we will get:



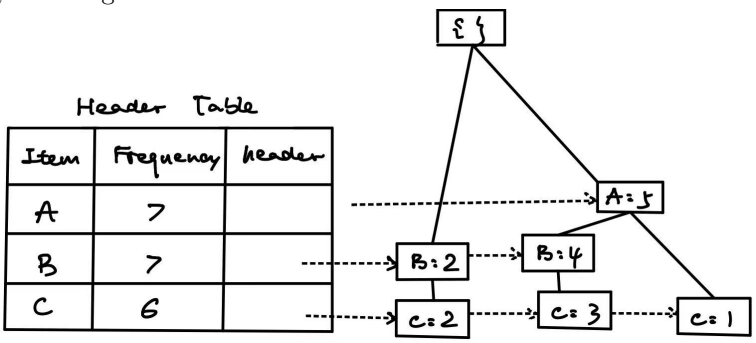
After inserting Tid 6, we will get:



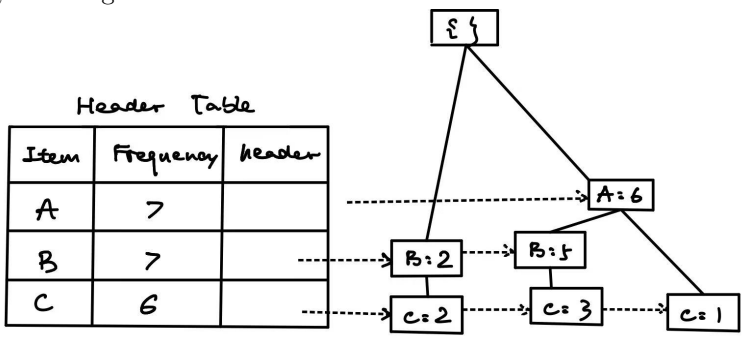
After inserting Tid 7, we will get:



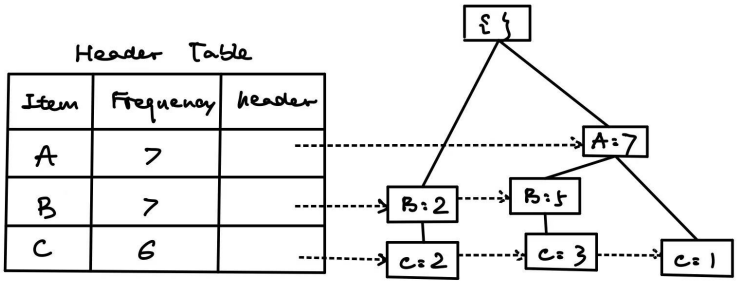
After inserting Tid 8, we will get:



After inserting Tid 9, we will get:



After inserting Tid 10, we will get:



which is the final answer.