Tiffany Wong A20442087
CS422 - HW 9 Exercises 12/04/2021

1. Exercises

1.1 Tan, Ch. 5 (Association Analysis)

Questions 1, 2, 9 (a), 9 (b), 15

1. For each of the following questions, provide an example of an association rule from the market basket domain that satisfies the following conditions. Also, describe whether such rules are subjectively interesting.

Support is how often a rule is applicable for the provided data set.
Support, s(X → Y) = [σ(X ∩ Y)] / N

Confidence is how frequently an item in Y appear in transactions that contain X.
Confidence, c(X → Y) = [σ(X ∩ Y)] / [σ(X)]

a. A rule that has high support and high confidence.
Using Table 5.1 from Chapter 5, I get the following rules:
X = Bread
Y = MIlk

{Bread} → {Milk}
Support = ⅗ = 60%
Confidence = 34 = 75%

It’s not subjectively interesting because bread and milk are pretty common things to consume highly enough to be in every grocery store visit.

b. A rule that has reasonably high support but low confidence.
Using Table 5.1 from Chapter 5, I get the following rules:
X = Milk
Y = Cola

{Milk} → {Cola}
Support = ⅖ = 40%
Confidence = 2/4 = ½ = 50%

These kinds of rule can be interesting because both drinks are targeted to the same demographic, but their purpose is different.

c. A rule that has low support and low confidence.
Using Table 5.1 from Chapter 5, I get the following rules:
X = Bread
Y = Eggs

{Bread} → {Eggs}
Support = ⅕ = 20%
Confidence = ¼ = 25%

These kinds of rules are not subjectively interesting for the same reason part (a) was not subjectively interesting.

d. A rule that has low support and high confidence.
Using Table 5.1 from Chapter 5, I get the following rules:
X = Eggs
Y = Diapers

{Eggs} → {Diapers}
Support = ⅕ = 20%
Confidence = 1/1 = 100%

These kinds of rules are subjectively interesting because it means that people with babies still eat a lot of eggs.

2. Consider the data set shown in Table 5.20.

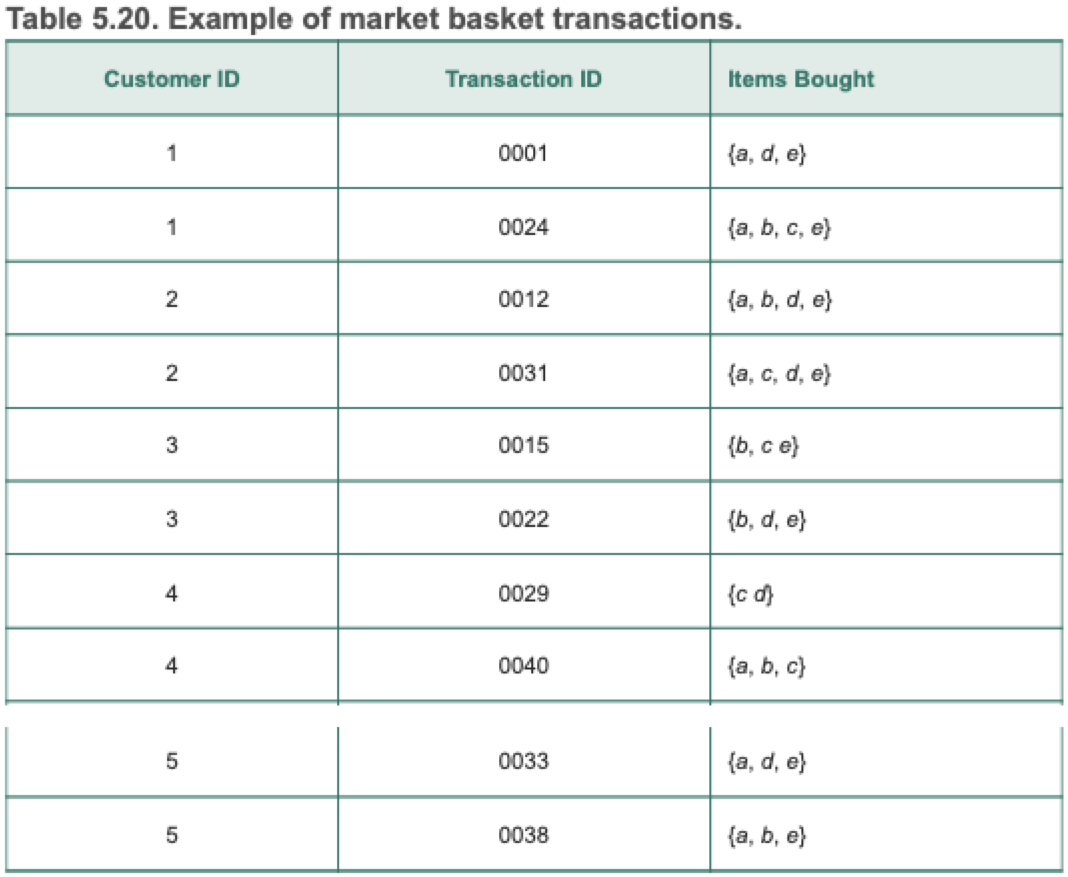


Table 5.20. Example of market basket transactions.

Customer ID	Transaction ID	Items Bought
1	0001	{a, d, e}
1	0024	{a, b, c, e}
2	0012	{a, b, d, e}
2	0031	{a, c, d, e}
3	0015	{b, c e}
3	0022	{b, d, e}
4	0029	{c d}
4	0040	{a, b, c}
5	0033	{a, d, e}
5	0038	{a, b, e}

a. Compute the support for itemsets {e}, {b, d}, and {b, d, e} by treating each transaction ID as a market basket.
Support for {e}: 0.8
Support for {b, d}: 0.2
Support for {b, d, e}: 0.2

b. Use the results in part (a) to compute the confidence for the association rules {b, d}→{e} and {e}→{b, d}. Is confidence a symmetric measure?
Confidence for {b, d} -> {e} = s({b, d, e}) / s({b, d}) = 1.0
Confidence for {e} -> {b, d} = s({b, d, e}) / s({e})= 0.25

Confidence is not a symmetric measure because it shows that if itemset {b, d} is purchased, we can say with confidence that itemset {e} will be purchased too, but if itemset {e} is purchased, it doesn't imply that itemset {b, d} will be purchased.

c. Repeat part (a) by treating each customer ID as a market basket. Each item should be treated as a binary variable (1 if an item appears in at least one transaction bought by the customer, and 0 otherwise).
Support for {e}: 0.8
Support for {b, d}: 1.0

Support for {b, d, e}: 0.8

d. Use the results in part (c) to compute the confidence for the association rules {b, d}→{e} and {e}→{b, d}.

Confidence for {b, d} -> {e} = s({b, d, e}) / s({b, d}) = 1.0
Confidence for {e} -> {b, d} = s({b, d, e}) / s({e})= 0.8

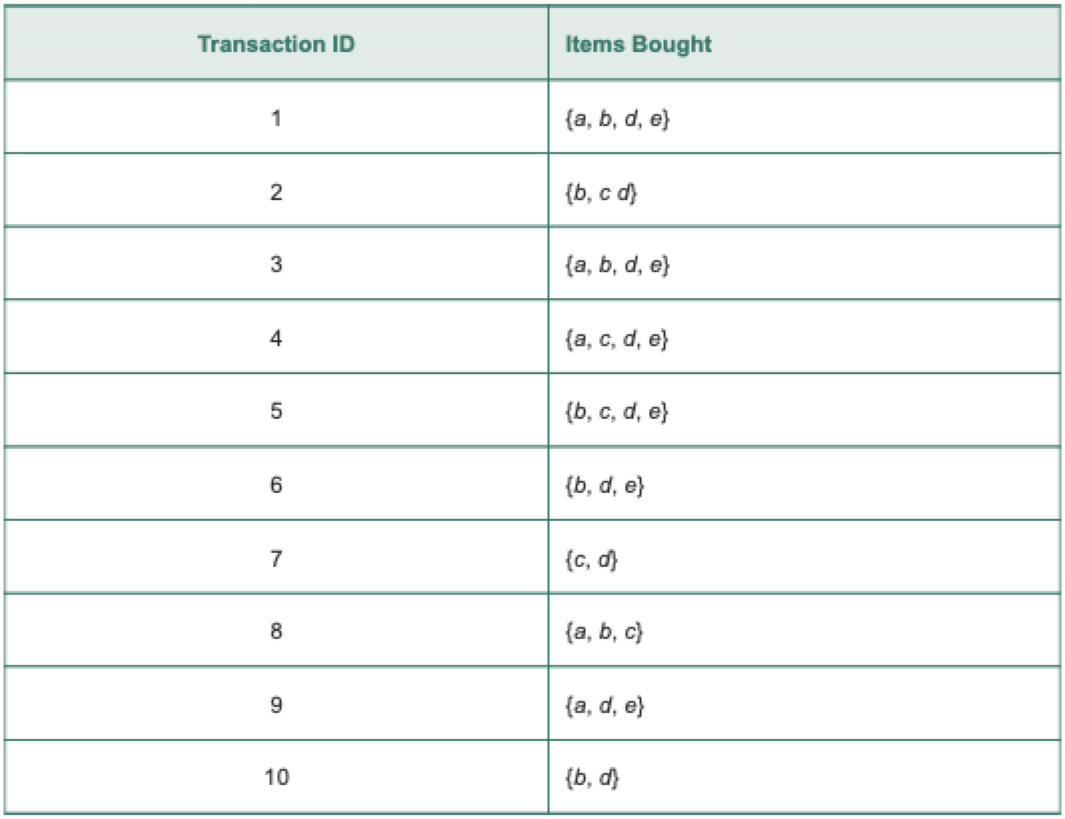
e. Suppose s1 and c1 are the support and confidence values of an association rule r when treating each transaction ID as a market basket. Also, let s2 and c2 be the support and confidence values of r when treating each customer ID as a market basket. Discuss whether there are any relationships between s1 and s2 or c1 and c2.

When we use the transactionID as a market basket, we're trying to predict the association between two itemsets for a given transaction. But with CustomerID, we're trying to predict the association between two itemsets from a customer's purchase history. It is dangerous to say that a high confidence/support in one thing will lead to high confidence/support in the other.

If a customer has previously bought a common item and then buys something else later in time, then it is very easy to show high support/confidence between those two items. However, if each customer had only purchased those items in a few transactions relative to their total transactions, the support/confidence would lower when using transactionID as a market basket.

9a. The Apriori algorithm uses a generate-and-count strategy for deriving frequent itemsets. Candidate itemsets of size k+1 are created by joining a pair of frequent itemsets of size k (this is known as the candidate generation step). A candidate is discarded if any one of its subsets is found to be infrequent during the candidate pruning step. Suppose the Apriori algorithm is applied to the data set shown in Table 5.22 with minsup=30%, i.e., any itemset occurring in less than 3 transactions is considered to be infrequent.

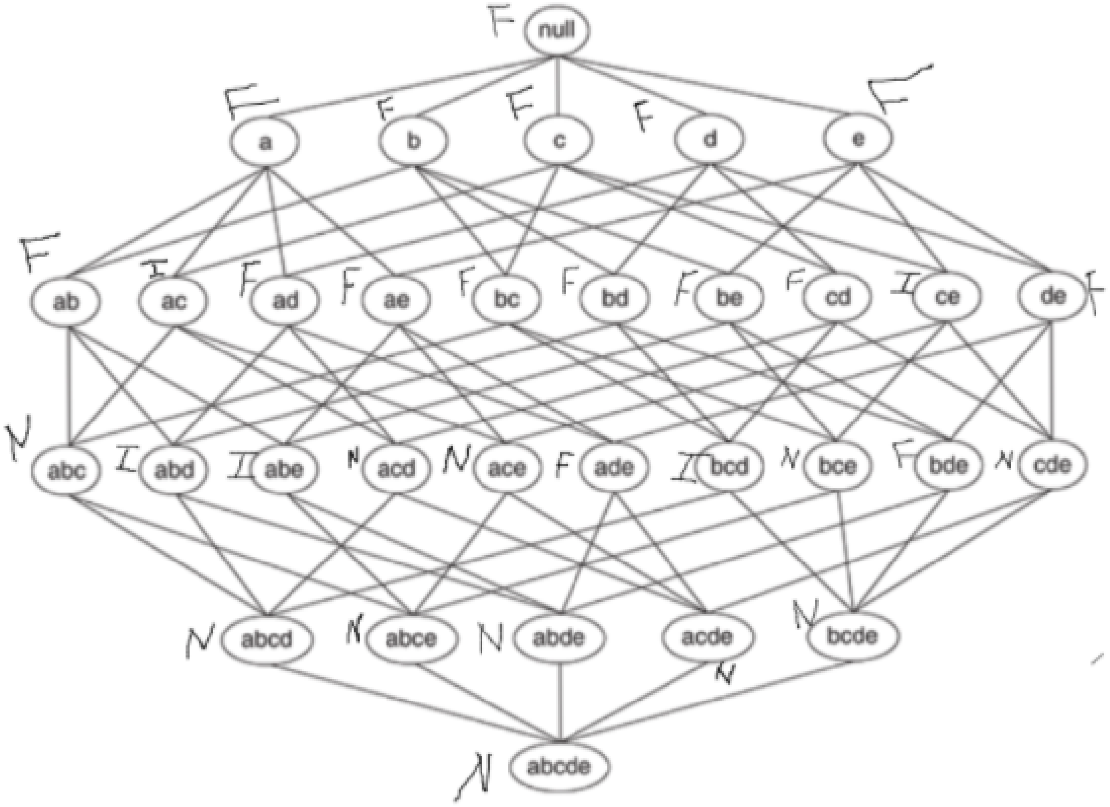
Table 5.22. Example of market basket transactions.



Transaction ID	Items Bought
1	{*a*, *b*, *d*, *e*}
2	{*b*, *c d*}
3	{*a*, *b*, *d*, *e*}
4	{*a*, *c*, *d*, *e*}
5	{*b*, *c*, *d*, *e*}
6	{*b*, *d*, *e*}
7	{*c*, *d*}
8	{*a*, *b*, *c*}
9	{*a*, *d*, *e*}
10	{*b*, *d*}

Draw an itemset lattice representing the data set given in Table 5.22 . Label each node in the lattice with the following letter(s):

- *N: If the itemset is not considered to be a candidate itemset by the Apriori algorithm. There are two reasons for an itemset not to be considered as a candidate itemset: (1) it is not generated at all during the candidate generation step, or (2) it is generated during the candidate generation step but is subsequently removed during the candidate pruning step because one of its subsets is found to be infrequent.*
- *F: If the candidate itemset is found to be frequent by the Apriori algorithm.*
- *I: If the candidate itemset is found to be infrequent after support counting.*



9b. What is the percentage of frequent itemsets (with respect to all itemsets in the lattice)?
16/32 = 50% of itemsets are frequent

15. Answer the following questions using the data sets shown in Figure 5.34. 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 minsup=10% (i.e., itemsets must be contained in at least 1000 transactions).

a. Which data set(s) will produce the most number of frequent itemsets?

Data set e will produce the most number of frequent itemsets because it's when there are dense columns because that means there are frequently recurring items. Datasets with overlapping columns will have a high number of frequent itemsets.

b. Which data set(s) will produce the fewest number of frequent itemsets?

Data set d will produce the fewest number of frequent itemsets because it's when there are essentially no columns because that means there are no frequent itemsets. Datasets with bare columns will have minimum support for their itemsets.

c. Which data set(s) will produce the longest frequent itemset?

Data set e will produce the longest frequent itemset because it's when there are many overlapping columns and data set e has many frequent 1-itemsets overlapping.

d. Which data set(s) will produce frequent itemsets with highest maximum support?

Data set b will produce frequent itemsets with highest maximum support because when item number is near 100, it shows that it occurs almost 8000 times and that is the highest support in all the data sets.

e. 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 20% to more than 70%)?

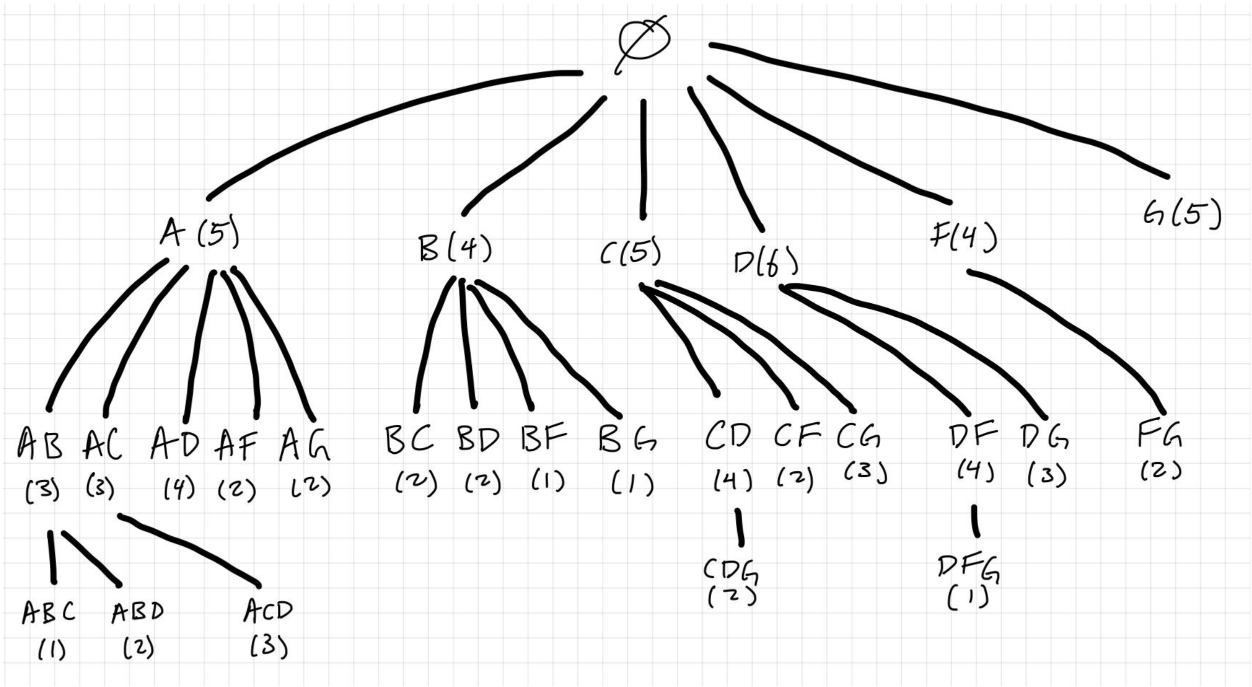
Data set e will produce frequent itemsets containing items with wide-varying support levels (i.e., items with mixed support, ranging from less than 20% to more than 70%) because it has frequent item columns with different lengths.

1.2 Zaki, Chapter 8 (Frequent Pattern Mining)
Questions 1(a), 4

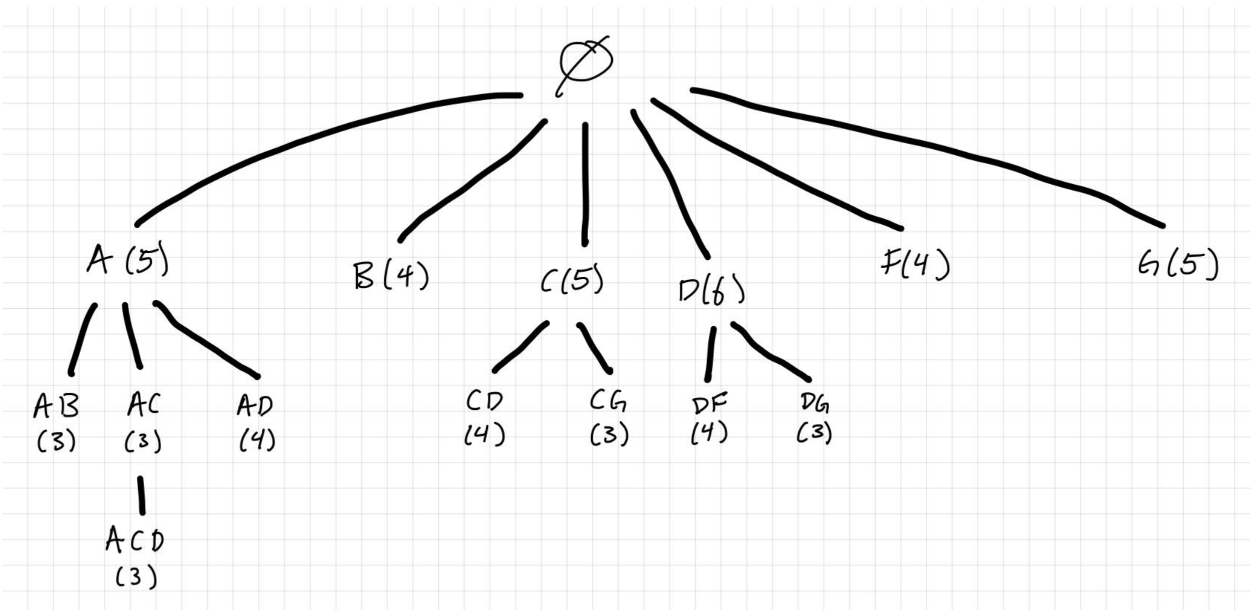
1a) Given the database in Table 8.2. Using minisup = ⅜, show how the Apriori algorithm enumerates all frequent patterns from this dataset.

Support	Itemset
6	D
5	A, C, G
4	B, F

Frequent pattern rough set

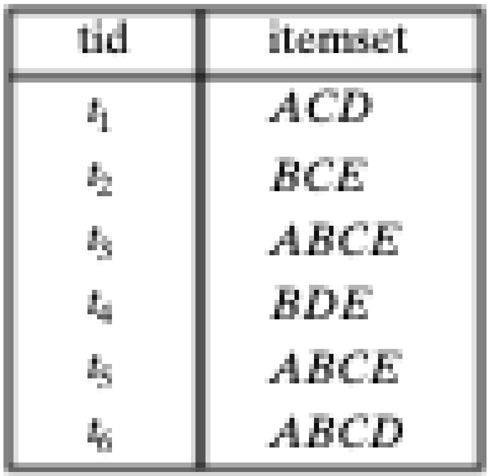


Frequent pattern final


∅
A (5)
B (4)
C (5)
D (6)
F (4)
G (5)
AB (3)
AC (3)
AD (4)
ACD (3)
CD (4)
CG (3)
DF (4)
DG (3)

4. Given the database in Table 8.4. Show all rules that one can generate from the set ABE.

Table 8.4. Dataset for Q4



tid	itemset
t_1	*ACD*
t_2	*BCE*
t_3	*ABCE*
t_4	*BDE*
t_5	*ABCE*
t_6	*ABCD*

sup(ABE) = 2

Rules	**conf(rule)**
AB → BE	2/4
B → AE	2/5
E → AB	4/5
AB → E	2/3
AE → B	2/2
BE → A	2/4
ABE → ∅	0
∅ → ABE	0