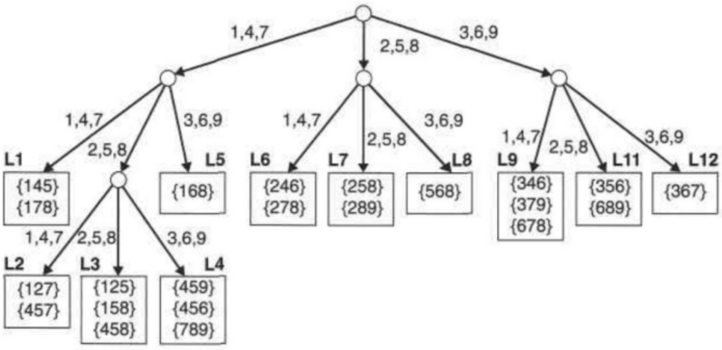
10215501435-杨茜雅-数据挖掘 Quiz

- The Apriori algorithm uses a hash tree data structure to efficiently count the support of candidate itemsets. Consider the hash tree for candidate 3-itemsets shown below.



(a) Given a transaction that contains items {1,3,4,5,8}, which of the hash tree leaf nodes will be visited when finding the candidates of the transaction?

(b) Use the visited leaf nodes in part (a) to determine the candidate itemsets that are contained in the transaction {1,3,4,5,8}.

题目 1:

Apriori 算法使用哈希树数据结构有效地计数候选项集的支持度。请考虑下面显示的候选 3-项集的哈希树。

(a) 给定一个包含物品 {1,3,4,5,8} 的交易记录，当寻找该交易的候选项集时，哪些哈希树的叶节点将会被访问?

(b) 使用（a）部分中访问的叶节点来确定包含在交易记录 {1,3,4,5,8} 中的候选项集。

题目答案:

(a) L1 L3 L5 L9 L11

(b) {145}，{158}，{458}

解答:

a)

首先，生成交易{1,3,4,5,8}所有可能的 3-项集:

{1,3,4} {1,3,5} {1,3,8} {1,4,5} {1,4,8} {1,5,8} {3,4,5} {3,4,8} {3,5,8} {4,5,8}

然后，我们根据哈希树的分支逻辑来检查这些 3-项集会访问哪些叶节点。

{1,3,4} 将会走到 L5
{1,3,5} 将会走到 L5
{1,3,8} 将会走到 L5
{1,4,5} 将会走到 L1
{1,4,8} 将会走到 L1
{1,5,8} 将会走到 L3
{3,4,5} 将会走到 L9
{3,4,8} 将会走到 L9
{3,5,8} 将会走到 L11
{4,5,8} 将会走到 L3

所以，会访问到 **L1 L3 L5 L9 L11** 这些节点。

(b) 接下来，我们需要检查在步骤(a)中确定的叶节点（L1, L3, L5, L9, L11）来找出哪些候选项集实际包含在交易{1,3,4,5,8}中。

由图可知：

L1 包含 **{145}** 和 {178}

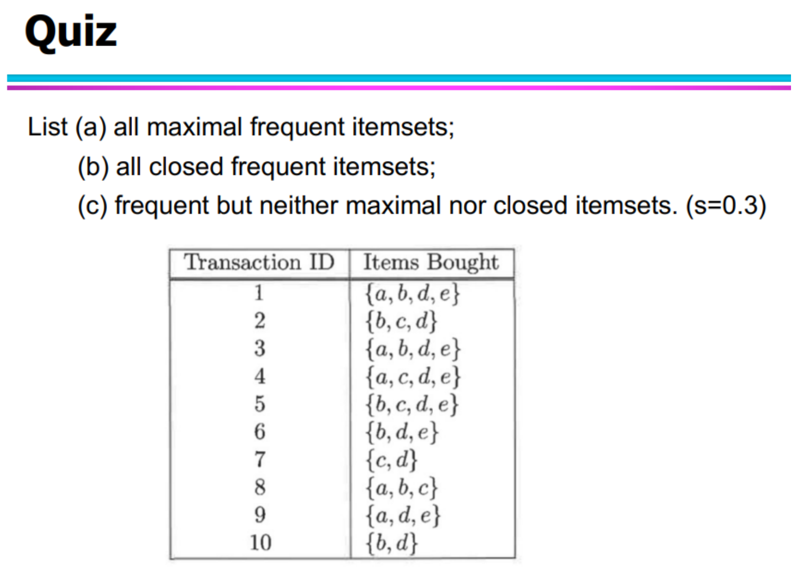
L3 包含 {125}、**{158}** 和 **{458}**

L5 包含 {168}

L9 包含 {346}、{379} 和 {678}

L11 包含 {356}、{689}

所以最终的答案是**{145}，{158}，{458}**。



Quiz

List (a) all maximal frequent itemsets;

(b) all closed frequent itemsets;

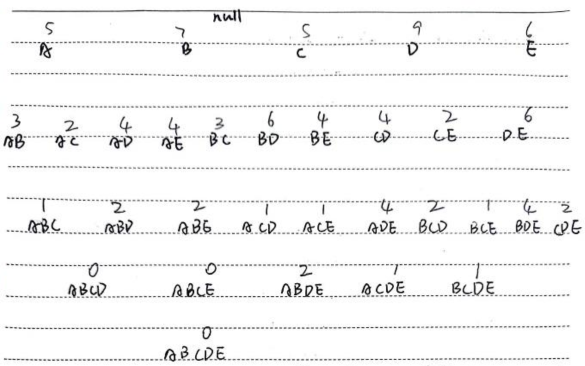
(c) frequent but neither maximal nor closed itemsets. (s=0.3)

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	$\{b, d\}$

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}	a b c d
7	{c,d}	
8	{a.b.c}	
9	{a,d,e}	
10	{b,d}	

Timeset - Support					
		ab	3	abc	1
a	5	ac	2	abd	2
b	7	ad	4	abe	2
		ae	4	acd	1
c	5	bc	3	ace	1
d	9	bd	6	ade	4
e	6	be	4	bcd	2
		dc	4	bce	1
		de	6	cde	2
		ce	2	bde	4

abcd	0	abcde	0
abce	0		
abde	2		
acde	1		
bcde	1		



S=0.3 => 任何一个项集在少于3个事务中出现 X fre

f: ~~a~~. ~~b~~、~~c~~、~~d~~、e、~~ab~~、ad、ae、~~bc~~、[illegible]、~~bd~~、be
~~cd~~、~~de~~、~~ade~~、~~bde~~

close: a、b、c、d、ab、bc、bd、cd、de、ade、bde

maximal: ab、bc、cd、ade、bde

(a): ab、bc、cd、ade、bde

(b): a、b、c、d、ab、bc、bd、cd、de、ade、bde

(c): e、ad、ae、be