CS412 HW#3
Tianqi Wu

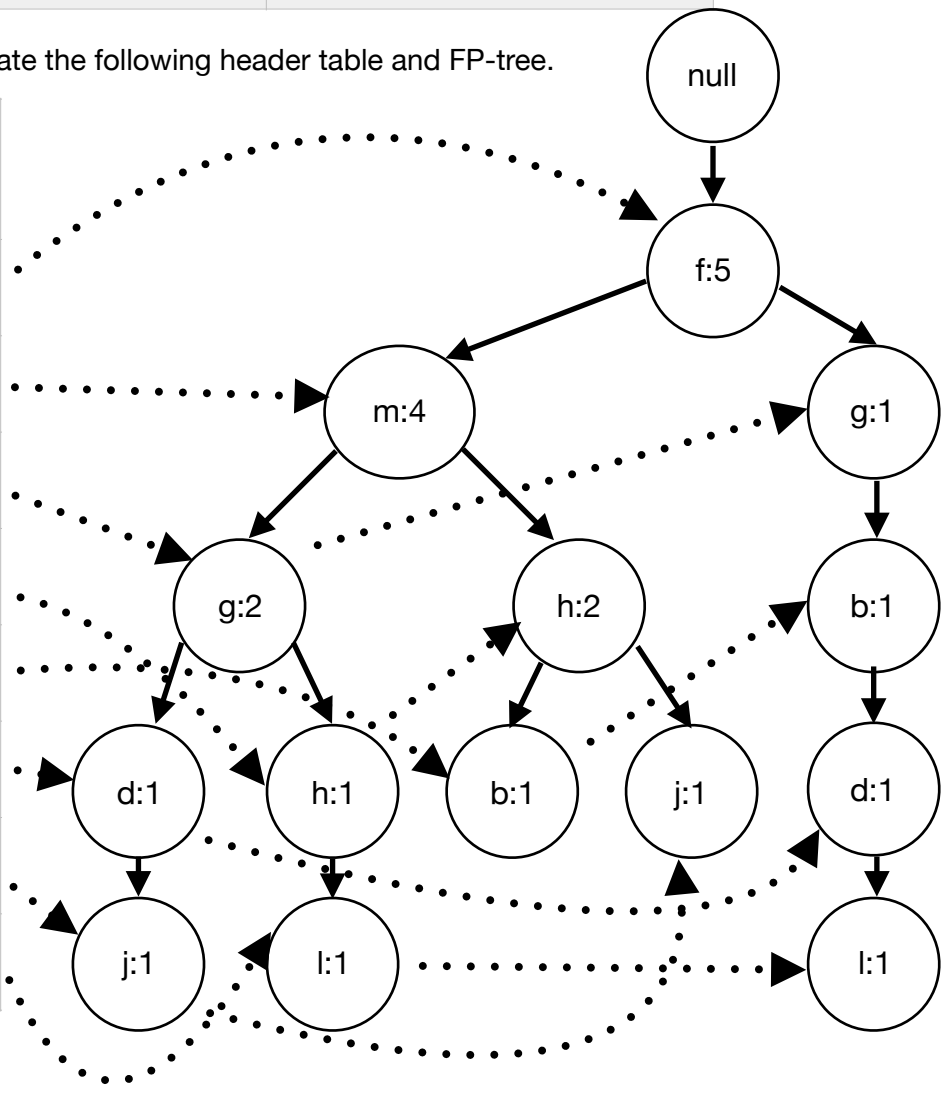
1.(1) First, count the frequency of each item. Since the minimum support = 0.4 and there are 5 transactions in total, we only keep the items with frequency of at least 2 (= 5*0.4).
Ordered list: f(5), m(4), g(3), h(3), b(2), d(2), j(2), l(2).
where the number in ( ) refers to the frequency of the item.

(2) After ordering the frequent items from raw transaction database based on the list generated.

TransactionID	Items
1	f,g,b,d,l
2	f,m,g,h,l
3	f,m,h,b
4	f,m,h,j
5	f,m,g,d,j

Then, we can generate the following header table and FP-tree.

Item	Count	Node-link
f	5	
m	4	
g	3	
h	3	
b	2	
d	2	
j	2	
l	2	



1.(3) First, start from the lowest count. We have the order j(2), b(2), h(3), m(4).
Then, the following table can be generated based on the information from the above FP-tree.

Item	Conditional Pattern Base	Conditional FP-tree	Frequent Pattern
j	{{f,m,g,d:1}, {f,m,h:1}}	{f:2, m:2}	{f,j: 2}, {m,j: 2}, {f,m,j :2}
b	{{f,m,h: 1},{f,g: 1}}	{f: 2}	{f,b: 2}
h	{{f,m,g: 1},{f,m: 2}}	{f: 3, m:3}	{f,h: 3}, {m,h: 3}, {f,m,h 3}
m	{f: 4}	{f: 4}	{f,m: 4}

1.(4) Ordering the items in each transaction by their frequency before constructing the FP-tree is necessary. Otherwise, some higher frequency items can be missed when generating the conditional pattern base since it may not even reach the item in the FP-tree generated from unordered transaction dataset. Ordering the items minimizes the number of scans needed to correctly find all the frequent patterns.

1.(5) Closed: {f,m,j} , {f,b}, {f,m,h}, {f,m}
Since those patterns have no super-pattern with the same supports.

Maximum: {f,m,j}, {f,b}, {f,m,h}
Since those patterns have no frequent super-itemset.

1.(6) {f,h} ⟶ {f,h,m} with support (40%) and confidence (100%)

{m,j} ⟶ {f,m,j} with support (40%) and confidence (100%)