数据挖掘和大数据分析

Outline

① Review Association Rules

② Apriori Algorithm ★

③ Lift & Association Rule Example ★

投票 最多可选1项

设置

Which knowledge is used in this example?

A Association Rules of DM

B Classification of DM

C Regression of DM

Frequently Bought Together

Price For All Three: $166.83

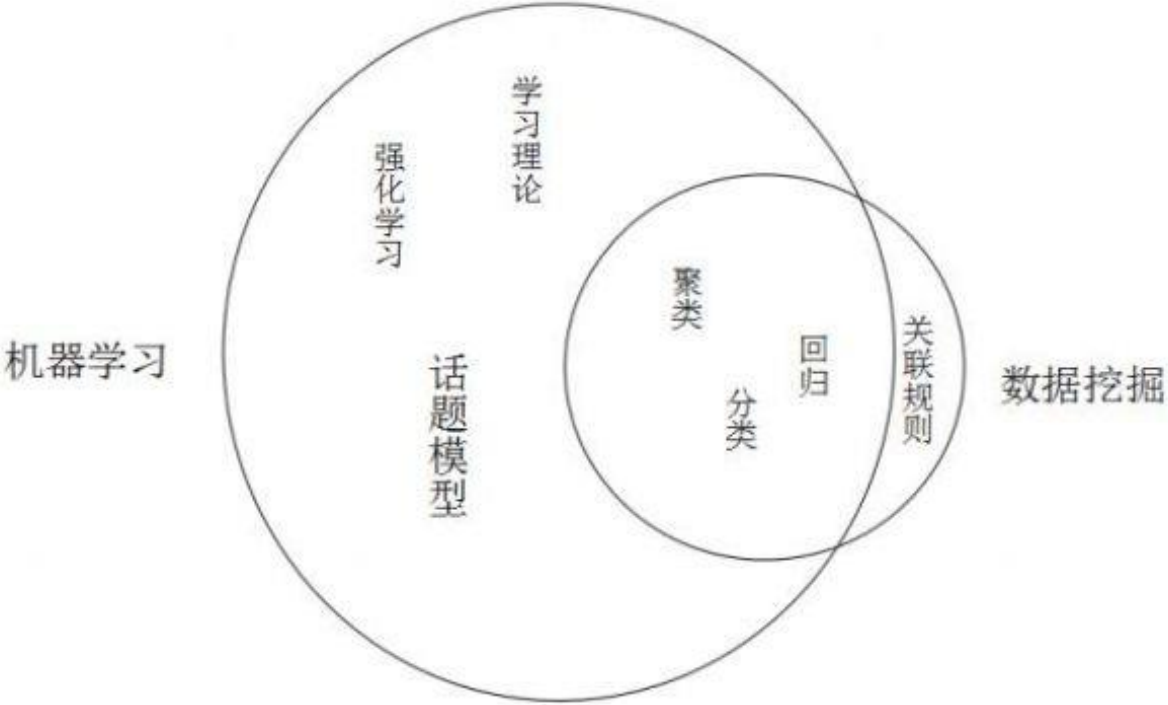
Add all three to Cart | Add all three to Wish List

Show availability and shipping details

- **This item:** Data Mining: Practical Machine Learning Tools and Techniques, Second Edition (Morgan Kaufmann Series in Data Management Systems) by Eibe Frank
- The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics) by Robert Tibshirani

提交

Review Association Rules



Review Association Rules

Frequent Itemsets

Association Rules

单选题 1分

设置

Support(Bread → Milk) = ?

Confidence(Bread → Milk) = ?

A 1/8 & 1/3

B 2/8 & 2/3

C 2/8 & 1/3

D 2/8 & 2/4

Transactions	Items
1	Bread, Jelly, Peanut, Butter
2	Bread, Butter
3	Bread, Jelly
4	Bread, Milk, Butter
5	Chips, Milk
6	Bread, Chips
7	Bread, Milk
8	Chips, Jelly

提交

单选题 1分

设置

Support(Milk →Bread) = ?

Confidence(Milk →Bread) = ?

A 1/8 & 1/3

B 2/8 & 2/3

C 2/8 & 1/3

D 3/8 & 1/3

Transactions	Items
1	Bread, Jelly, Peanut, Butter
2	Bread, Butter
3	Bread, Jelly
4	Bread, Milk, Butter
5	Chips, Milk
6	Bread, Chips
7	Bread, Milk
8	Chips, Jelly

提交

Review Association Rules

Transactions	Items
1	Bread, Jelly, Peanut, Butter
2	Bread, Butter
3	Bread, Jelly
4	Bread, Milk, Butter
5	Chips, Milk
6	Bread, Chips
7	Bread, Milk
8	Chips, Jelly

Itemset	Support
Bread	6/8
Butter	3/8
Chips	2/8
Jelly	3/8
Milk	3/8
Peanut	1/8

Bread → Milk

Support: 2/8

Confidence: 1/3

Milk → Bread

Support: 2/8

Confidence: 2/3

Searching for rules in the form of: Bread → Butter

Review Association Rules

- **Support and Confidence are bounded by thresholds:**
  - Minimum support σ
  - Minimum confidence Φ
- A frequent (large) itemset is an itemset with support larger than σ.
- **A strong rule is a rule that is frequent and its confidence is higher than Φ.**
- **Association Rule Problem**
  - Given **I, D, σ and Φ**, to find all strong rules in the form of **$X \rightarrow Y$.**
- **The number of all possible association rules is huge.**
  - Brute force strategy is infeasible.
  - A smart way is to find frequent itemsets first.

DATA ANALYTICS:

DATA MINING AND BIG DATA

—— Mining Rules 2

Apriori Algorithm

- **For each *k*, we construct two sets of *k -sets* (sets of size *k*):**
  - C_k = *candidate* k -sets = those that might be frequent sets (support $\geq s$) based on information from the pass for $k-1$.
  - L_k = the set of truly frequent k -sets.

Apriori Algorithm

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

Scan D →

C_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

Support > 1

L_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{5}	3

C_2

itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

Scan D ←

C_2

itemset	sup
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

L_2

itemset	sup
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

C_3

itemset
{2 3 5}

Scan D →

L_3

itemset	sup
{2 3 5}	2

Note: {1,2,3} {1,2,5} and {1,3,5} not in C_3

A_Rule	Con
2->{3,5}	66.7%
3->{3,5}	66.7%
5->{2,3}	66.7%
{3,5}->2	1
{2,5}->3	66.7%
{2,3}->5	1

>50%

Apriori Algorithm

（T1，T2，T3，T4，T5，T6，T7，T8，T9）

{{I1， I2， I5},{I2， I4},{I2， I3},{I1， I2， I4},{I1， I3},{I2，

I3},{I1， I3},{I1， I2， I3， I5},{I1， I2， I3}}

Apriori Algorithm

C1

Items	Support
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Min_Support = 2

Apriori Algorithm

C2

Items	Support
{I1, I2}	4
{I1, I3}	4
{I1, I4}	1
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2
{I3, I4}	0
{I3, I5}	1
{I4, I5}	0

Min_Support = 2

投票 最多可选1项

设置

Do you finish the task?

A Yes

B No

提交

Apriori Algorithm

C3

Items	Support
{I1, I2, I3}	2
{I1, I2, I5}	2
{I1, I3, I5}	1
{I1, I2, I4}	1

Min_Support = 2

C4

Items	Support
{I1, I2, I3, I5}	1

Apriori Algorithm

C_k: Candidate itemset of size k

L_k: Frequent itemset of size k

$L_1 \leftarrow \{frequent\ items\}$

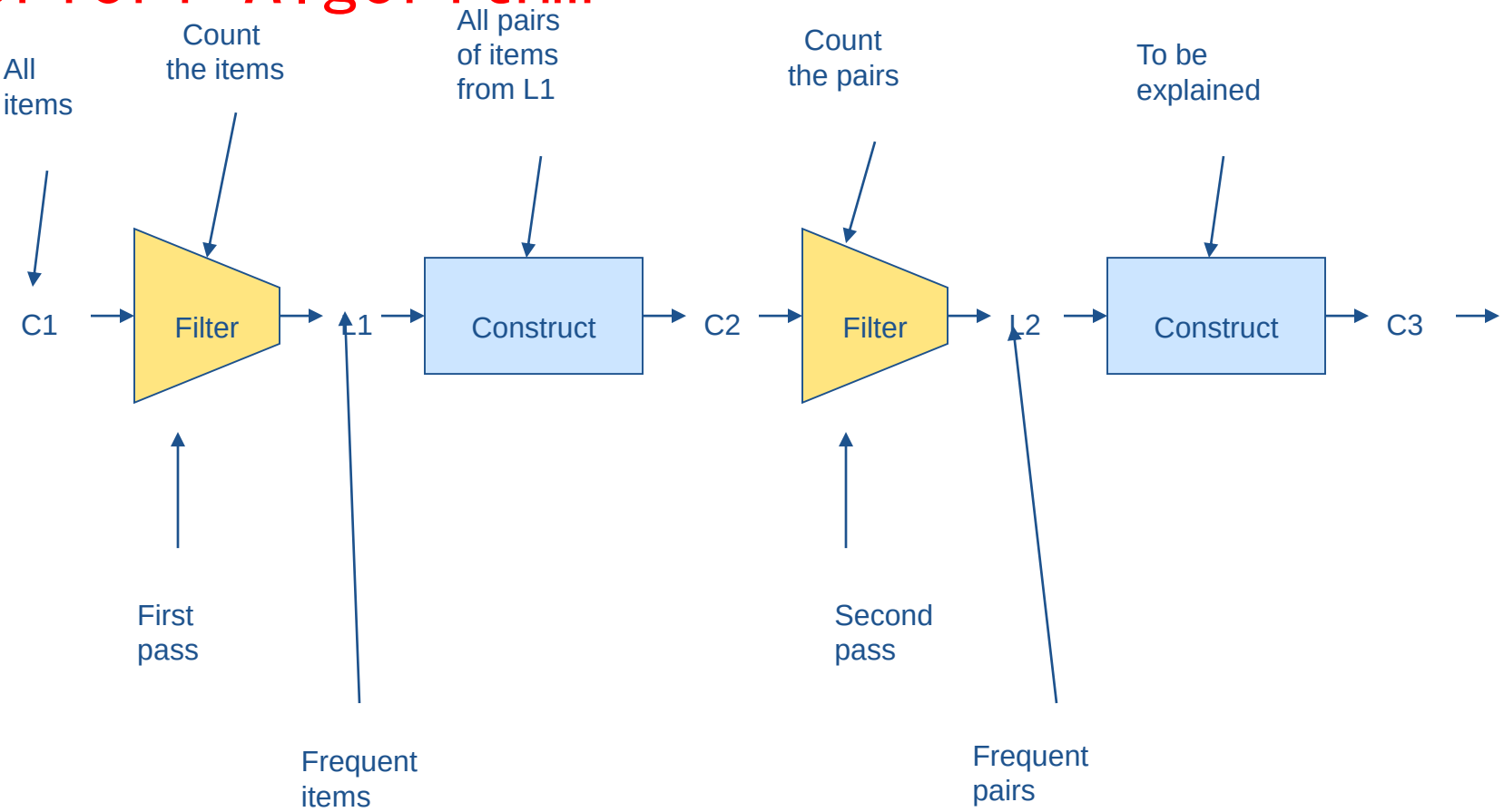
for (k=1; $L_k \neq \emptyset$; k++)

- $C_{k+1} \leftarrow candidate(L_k)$ — candidates
- for each transaction t
  - $Q \leftarrow \{c \mid c \in C_{k+1} \wedge c \subseteq t\}$
  - $count[c] \leftarrow count[c] + 1, \quad \forall c \in Q$ — counting
- end for
- $L_{k+1} \leftarrow \{c \mid c \in C_{k+1} \wedge count[c] / N \geq \sigma\}$ — filtering

end for

return $\bigcup_k L_k$

Apriori Algorithm



Apriori Algorithm

```
from akapriori import Apriori
dataset =
[("ariocot","apple","cherry","plum","banana"),("strawberry","plum","cherry"),("persimm
on","peach","banana","apple"),("kiwi
fruit","apple","pear"),("cherry","pear","banana"),("watermelon","apple"),("plum","banan
a"),("pear","peach","cherry","banana","apricot"),"pineapple","apple","plum",("banana",
"plum","peach"),("grape","cherry"),("mandarin","plum"),("melon","apple","persimmon",
"plum"),("peach","cherry","apple"),("apple","mandarin","plum","persimmon")]

rules = apriori(dataset,support = 0.05,confidence = 0.3,lift = 2)

rules_sorted = sorted(rules,key=lambda x:[x[4],x[3],x[2]],reverse = True)
```

投票 最多可选1项

设置

Do you finish the task?

A Yes

B No

提交

Lift

Lift(A => B)>1

$$Lift\ (A \Rightarrow B) = \frac{Support(A \Rightarrow B)}{Support(A) * Support(B)}$$

Lift(A => B) <=1

Lift(A => B) =1

Lift

篮子	商品1	商品2	商品3
1	香草威化	香蕉	狗粮
2	香蕉	面包	酸奶
3	香蕉	苹果	酸奶
4	香草威化	香蕉	生奶油
5	面包	香草威化	酸奶
6	牛奶	面包	香蕉
7	香草威化	苹果	香蕉
8	酸奶	苹果	香草威化
9	香草威化	香蕉	牛奶
10	香蕉	面包	花生酱

Lift （Vanilla Wafer => Banana） =

Lift （Banana => Vanilla Wafer） =

Confidence （Vanilla Wafer => Banana） = 4/6 = 67%

Confidence （Banana => Vanilla Wafer） = 4/8 = 50%

Association Rule Example

Association Rule Example

投票　最多可选2项

设置

关于抽查作业，你的想法?

A　需要不停补作业，老师很烦人

B　督促学生按时完成作业，不至于积累很多

C　没有什么意思，该抄还是抄

D　作业又不计分，没有必要检查

提交

贵在坚持！