Data Analytics

Mrinal Das

Scenario

In a cutting machine suddenly you observe that friction sound has increased

Most likely vibration will also become higher soon

You may also expect to see defects in the cut pieces

Association

Association is a group where elements of that group are related to each other in some sense.

During running of some mixing machine or cutting machine, several patterns are related. For example, vibration and sound are almost directly proportional.

In some sense members of an association co-occur.

We want to find associations given the problem.

Observation Related to a Machine

Time	High Temp	High Vibration	High Pressure	Low Lubrication	High Speed	Bearing Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

Do you observe any association ?

Observation Related to a Machine

Time	High Temp	High Vibration	High Pressure	Low Lubrication	High Speed	Bearing Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

High Temp, High Vibration, Low Lubrication with Bearing Alarm

Goal

To find out associations

High Temp, Bearing Alarm

High Vibration, Bearing Alarm

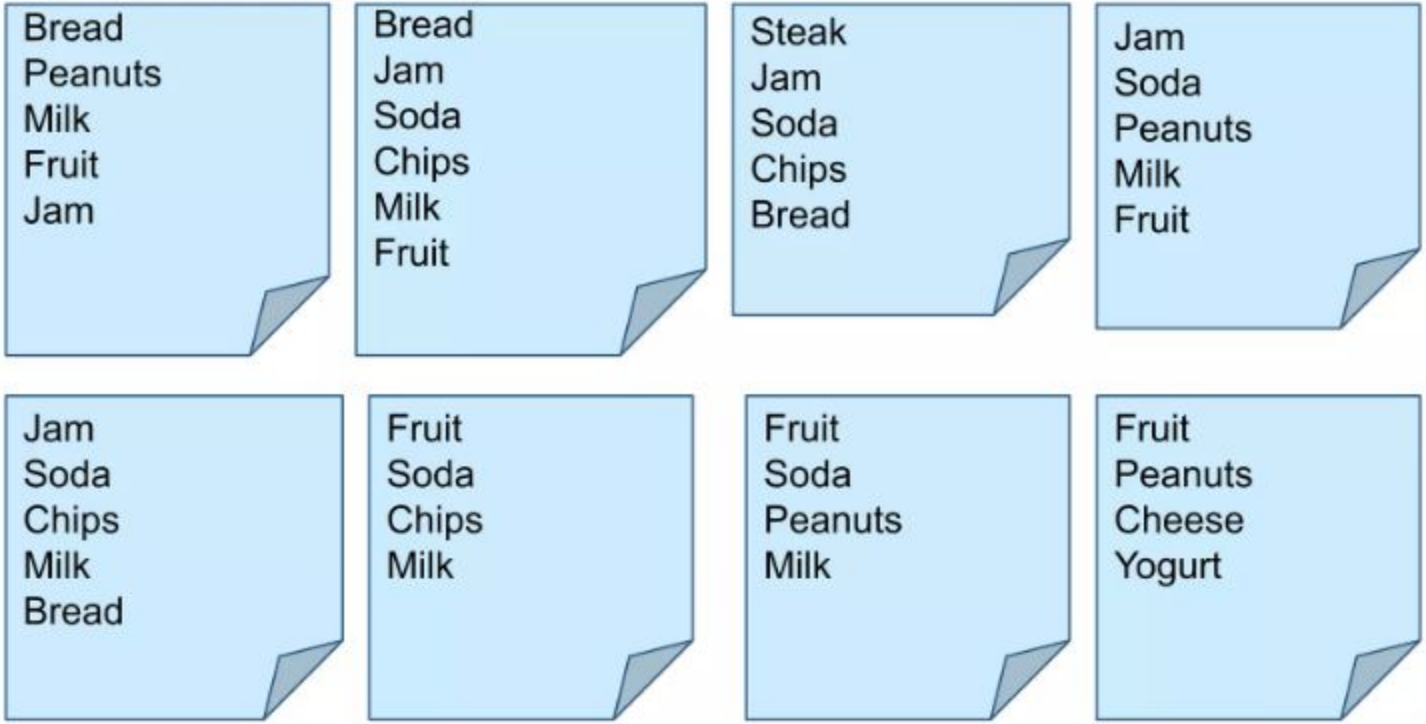
Then in turn, obtain rules such as

High Temp causes Bearing Alarm

High Vibration causes Bearing Alarm

Association in Super Markets

Example of Baskets



Basket in Manufacturing

A Basket need not be always a be Cart

A basket can be

a work order – a list of items serviced together

a log record – a list of defects found together

a composition – materials used in some product variety

A basket is a group of things that are related by some event

Dilemma of Supermarket Owner

- What to put on Sale
- How to give coupons
- How to arrange items in shelves

Again there is intuition vs data driven decision making

Can we use past transactions to analyse and prescribe ?

Case I

Suppose "diet coke" is in consequence

Can we find the rules ?

Can we find the items that boost the sale of diet coke ?

What can we do in the store to boost the sale of diet coke ?

Case II

Lets put bagels in antecedent

Can we find the impact If we stop selling bagels ?

Case III

Lets put sausage in antecedent

Mustard in consequence

Can we find out relevant rules ?

What items should be sold with sausages to increase sale of mustard ?

Rule

90% people who buy bread and butter also buy milk.

Bread and butter are antecedent

Milk is consequence

90% is confidence of the rule

Example of Association Rule

TID	Items
1	Bread, Peanuts, Milk, Fruit, Jam
2	Bread, Jam, Soda, Chips, Milk, Fruit
3	Steak, Jam, Soda, Chips, Bread
4	Jam, Soda, Peanuts, Milk, Fruit
5	Jam, Soda, Chips, Milk, Bread
6	Fruit, Soda, Chips, Milk
7	Fruit, Soda, Peanuts, Milk
8	Fruit, Peanuts, Cheese, Yogurt

Examples

$\{bread\} \Rightarrow \{milk\}$

$\{soda\} \Rightarrow \{chips\}$

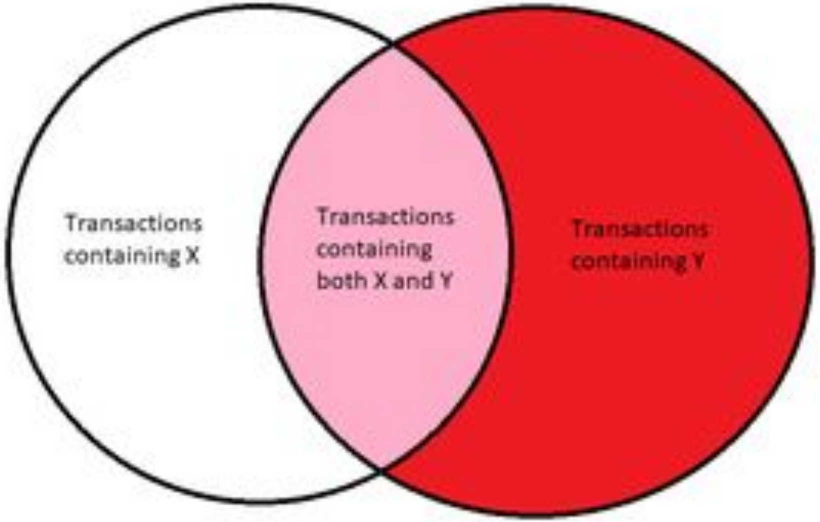
$\{bread\} \Rightarrow \{jam\}$

Association Rule X to Y implies that when X happens Y also happens

Items and Transactions

- $I = \{i_1, i_2, \ldots, i_m\}$: a set of *items*.
- Transaction t :
  - t a set of items, and $t \subseteq I$.
- Transaction Database T: a set of transactions $T = \{t_1, t_2, \ldots, t_n\}$.

Intersection



Support

$$\text{support} = P(A \cap B) = \frac{(\text{number of transactions containing } A \text{ and } B)}{(\text{total number of transactions})}$$

Example

- **Itemset**
  - A collection of one or more items, e.g., {milk, bread, jam}
  - k-itemset, an itemset that contains k items
- **Support count (σ)**
  - Frequency of occurrence of an itemset
  - σ({Milk, Bread}) = 3
    σ({Soda, Chips}) = 4
- **Support**
  - Fraction of transactions that contain an itemset
  - s({Milk, Bread}) = 3/8
    s({Soda, Chips}) = 4/8
- **Frequent Itemset**
  - An itemset whose support is greater than or equal to a minsup threshold

TID	Items
1	Bread, Peanuts, Milk, Fruit, Jam
2	Bread, Jam, Soda, Chips, Milk, Fruit
3	Steak, Jam, Soda, Chips, Bread
4	Jam, Soda, Peanuts, Milk, Fruit
5	Jam, Soda, Chips, Milk, Bread
6	Fruit, Soda, Chips, Milk
7	Fruit, Soda, Peanuts, Milk
8	Fruit, Peanuts, Cheese, Yogurt

Confidence

$$\operatorname{conf}(X \Rightarrow Y) = P(Y|X)$$

$$P(Y|X) = \frac{\operatorname{supp}(X \cup Y)}{\operatorname{supp}(X)}$$

$$= \frac{\operatorname{supp}(X \cup Y)}{\operatorname{supp}(X)} = \frac{\text{number of transactions containing } X \text{ and } Y}{\text{number of transactions containing } X}$$

- Implication of the form X $\Rightarrow$ Y, where X and Y are itemsets
- Example, {bread} $\Rightarrow$ {milk}
- Rule Evaluation Metrics, Suppor & Confidence
- Support (s)
  - Fraction of transactions that contain both X and Y

$$s = \frac{\sigma(\{\text{Bread}, \text{Milk}\})}{\#\text{ of transactions}} = 0.38$$

- Confidence (c)
  - Measures how often items in Y appear in transactions that contain X

$$c = \frac{\sigma(\{\text{Bread}, \text{Milk}\})}{\sigma(\{\text{Bread}\})} = 0.75$$

TID	Items
1	Bread, Peanuts, Milk, Fruit, Jam
2	Bread, Jam, Soda, Chips, Milk, Fruit
3	Steak, Jam, Soda, Chips, Bread
4	Jam, Soda, Peanuts, Milk, Fruit
5	Jam, Soda, Chips, Milk, Bread
6	Fruit, Soda, Chips, Milk
7	Fruit, Soda, Peanuts, Milk
8	Fruit, Peanuts, Cheese, Yogurt

- Support (s)
  - Fraction of transactions that contain both X and Y

$$s = \frac{\sigma(\{\text{Bread}, \text{Milk}\})}{\#\text{ of transactions}} = 0.38$$

- Confidence (c)
  - Measures how often items in Y appear in transactions that contain X

$$c = \frac{\sigma(\{\text{Bread}, \text{Milk}\})}{\sigma(\{\text{Bread}\})} = 0.75$$

Goal

- Given a set of transactions T, the goal of association rule mining is to find all rules having
  - support ≥ minsup threshold
  - confidence ≥ minconf threshold

- Brute-force approach:
  - List all possible association rules
  - Compute the support and confidence for each rule
  - Prune rules that fail the minsup and minconf thresholds

- Brute-force approach is computationally prohibitive!

- Frequent Itemset Generation
  - Generate all itemsets whose support $\geq$ minsup
- Rule Generation
  - Generate high confidence rules from frequent itemset
  - Each rule is a binary partitioning of a frequent itemset
- Frequent itemset generation is computationally expensive

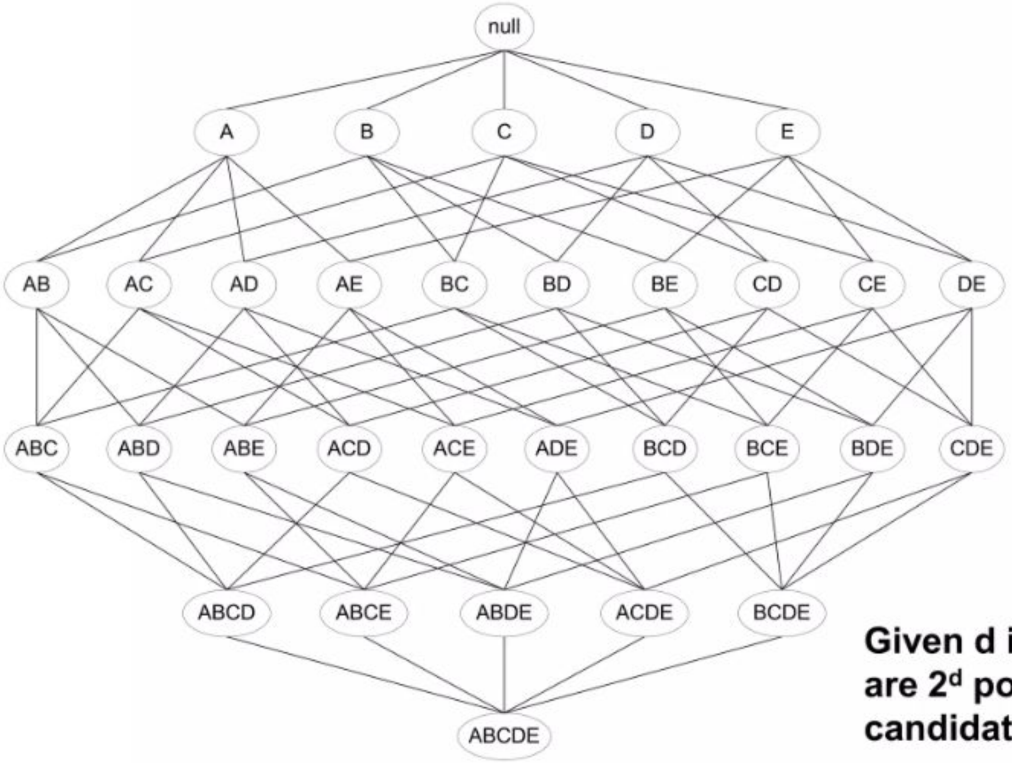
Example

TID	Items
1	Bread, Peanuts, Milk, Fruit, Jam
2	Bread, Jam, Soda, Chips, Milk, Fruit
3	Steak, Jam, Soda, Chips, Bread
4	Jam, Soda, Peanuts, Milk, Fruit
5	Jam, Soda, Chips, Milk, Bread
6	Fruit, Soda, Chips, Milk
7	Fruit, Soda, Peanuts, Milk
8	Fruit, Peanuts, Cheese, Yogurt

$\{Bread, Jam\} \Rightarrow \{Milk\}$
$\{Milk, Jam\} \Rightarrow \{Bread\}$
$\{Bread\} \Rightarrow \{Milk, Jam\}$
$\{Jam\} \Rightarrow \{Bread, Milk\}$
$\{Milk\} \Rightarrow \{Bread, Jam\}$

$\{Bread, Jam\} \Rightarrow \{Milk\}$ s=0.4 c=0.75
$\{Milk, Jam\} \Rightarrow \{Bread\}$ s=0.4 c=0.75
$\{Bread\} \Rightarrow \{Milk, Jam\}$ s=0.4 c=0.75
$\{Jam\} \Rightarrow \{Bread, Milk\}$ s=0.4 c=0.6
$\{Milk\} \Rightarrow \{Bread, Jam\}$ s=0.4 c=0.5

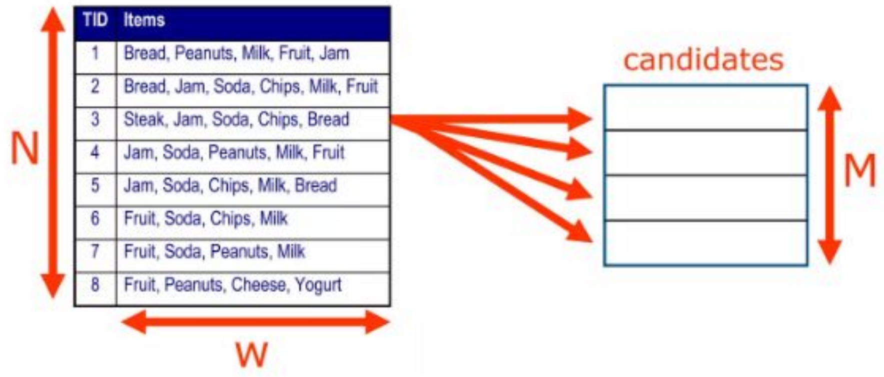
Frequent Itemset Generation



Given d items, there are 2^d possible candidate itemsets

Brute-force is Expensive

- Brute-force approach:
  - Each itemset in the lattice is a candidate frequent itemset
  - Count the support of each candidate by scanning the database



TID	Items
1	Bread, Peanuts, Milk, Fruit, Jam
2	Bread, Jam, Soda, Chips, Milk, Fruit
3	Steak, Jam, Soda, Chips, Bread
4	Jam, Soda, Peanuts, Milk, Fruit
5	Jam, Soda, Chips, Milk, Bread
6	Fruit, Soda, Chips, Milk
7	Fruit, Soda, Peanuts, Milk
8	Fruit, Peanuts, Cheese, Yogurt

  - Match each transaction against every candidate
  - Complexity ~ O(NMw) => Expensive since $M = 2^d$

Improve Efficiency

- Reduce the number of candidates (M)
  - Complete search: $M=2^d$
  - Use pruning techniques to reduce M
- Reduce the number of transactions (N)
  - Reduce size of N as the size of itemset increases
- Reduce the number of comparisons (NM)
  - Use efficient data structures to store the candidates or transactions
  - No need to match every candidate against every transaction

Reducing Candidates

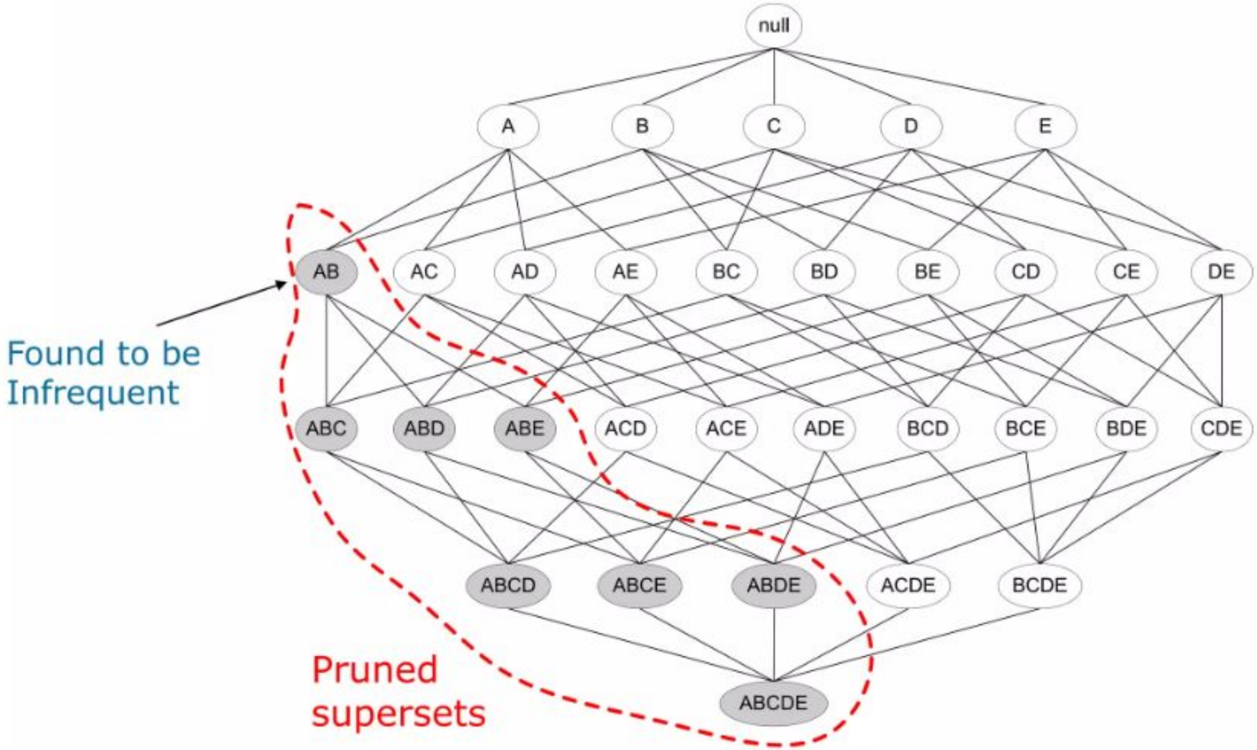
- Apriori principle
  - If an itemset is frequent, then all of its subsets must also be frequent

- Apriori principle holds due to the following property of the support measure:

$$\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$$

- Support of an itemset never exceeds the support of its subsets
- This is known as the anti-monotone property of support

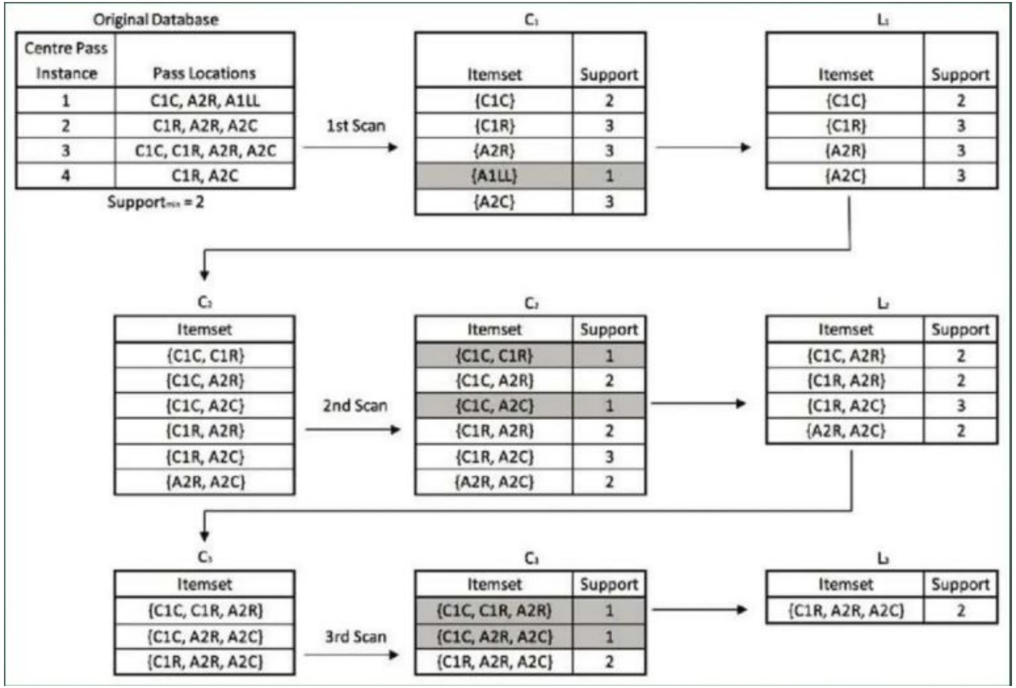
Illustration



Apriori Algorithm

- Let k=1
- Generate frequent itemsets of length 1
- Repeat until no new frequent itemsets are identified
  - Generate length (k+1) candidate itemsets from length k frequent itemsets
  - Prune candidate itemsets containing subsets of length k that are infrequent
  - Count the support of each candidate by scanning the DB
  - Eliminate candidates that are infrequent, leaving only those that are frequent

Example


Original Database
Centre Pass Instance | Pass Locations
1 | C1C, A2R, A1LL
2 | C1R, A2R, A2C
3 | C1C, C1R, A2R, A2C
4 | C1R, A2C
Support_min = 2
1st Scan
C1
Itemset | Support
{C1C} | 2
{C1R} | 3
{A2R} | 3
{A1LL} | 1
{A2C} | 3
L1
Itemset | Support
{C1C} | 2
{C1R} | 3
{A2R} | 3
{A2C} | 3
C2
Itemset
{C1C, C1R}
{C1C, A2R}
{C1C, A2C}
{C1R, A2R}
{C1R, A2C}
{A2R, A2C}
2nd Scan
C2
Itemset | Support
{C1C, C1R} | 1
{C1C, A2R} | 2
{C1C, A2C} | 1
{C1R, A2R} | 2
{C1R, A2C} | 3
{A2R, A2C} | 2
L2
Itemset | Support
{C1C, A2R} | 2
{C1R, A2R} | 2
{C1R, A2C} | 3
{A2R, A2C} | 2
C3
Itemset
{C1C, C1R, A2R}
{C1C, A2R, A2C}
{C1R, A2R, A2C}
3rd Scan
C3
Itemset | Support
{C1C, C1R, A2R} | 1
{C1C, A2R, A2C} | 1
{C1R, A2R, A2C} | 2
L3
Itemset | Support
{C1R, A2R, A2C} | 2

Rule Generation

- Given a frequent itemset L, find all non-empty subsets $f \subset L$ such that $f \rightarrow L - f$ satisfies the minimum confidence requirement
- If {A,B,C,D} is a frequent itemset, candidate rules:

  ABC $\rightarrow$D, ABD $\rightarrow$C, ACD $\rightarrow$B, BCD $\rightarrow$A, A $\rightarrow$BCD, B $\rightarrow$ACD, C $\rightarrow$ABD, D $\rightarrow$ABC, AB $\rightarrow$CD, AC $\rightarrow$ BD, AD $\rightarrow$ BC, BC $\rightarrow$AD, BD $\rightarrow$AC, CD $\rightarrow$AB

- If $|L| = k$, then there are $2^k - 2$ candidate association rules (ignoring $L \rightarrow \varnothing$ and $\varnothing \rightarrow L$)

Observation Related to a Machine

Time	High Temp	High Vibration	High Pressure	Low Lubrication	High Speed	Bearing Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

Computing Support

Time	High Temp	High Vibration	High Pressure	Low Lubrication	High Speed	Bearing Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

$$\boxed{Support(X) = \frac{\text{Number of transactions containing } X}{\text{Total number of transactions}}}$$

$$HighTemp + HighVibration + LowLubrication$$

$$Support = \frac{4}{10}$$

$$\boxed{Support = 40\%}$$

Computing Confidence

$$HighTemp + HighVibration + LowLubrication \rightarrow BearingAlarm.$$

Time	High Temp	High Vibration	High Pressure	Low Lubri cation	High Speed	Beari ng Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

$$\boxed{Confidence(X \rightarrow Y) = \frac{Support(X \cup Y)}{Support(X)}}$$

$$Support(X) = 40\%$$

$$Support(X, Y) = 40\%.$$

$$Confidence = \frac{40\%}{40\%}$$

$$\boxed{Confidence = 100\%}$$

Use Case

Why is this important in manufacturing ?

$$\boxed{HighTemp + HighVibration + LowLubrication \rightarrow BearingFailure}$$

Suppose a new machine is installed and nobody has much experience with it.

Using data analytics one can discover this rule

That helps to analyse the machine

More importantly, can signal early warning.

Apriori Algorithm - step 1

List items

```
HighTemp
HighVibration
HighPressure
LowLubrication
HighSpeed
BearingAlarm
QualityDefect
Downtime
```

Time	High Temp	High Vibration	High Pressure	Low Lubrication	High Speed	Bearing Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

If an itemset is infrequent, all of its supersets are infrequent.

Apriori Algorithm - step 2

Short-list frequent items

HighTemp	✓
HighVibration	✓
HighPressure	✓
LowLubrication	✓
HighSpeed	✓
BearingAlarm	✓
Downtime	✗

Time	High Temp	High Vibration	High Pressure	Low Lubri cation	High Speed	Beari ng Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

If an itemset is infrequent, all of its supersets are infrequent.

Apriori Algorithm - step 3

Find frequent pairs

```
HighTemp + HighVibration          ✓
HighTemp + LowLubrication         ✓
HighVibration + LowLubrication    ✓
HighPressure + HighSpeed          ✗
```

Time	High Temp	High Vibration	High Pressure	Low Lubrication	High Speed	Bearing Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

If an itemset is infrequent, all of its supersets are infrequent.

Apriori Algorithm - step 4

Create larger combination

$$HighTemp + HighVibration + LowLubrication.$$

Time	High Temp	High Vibration	High Pressure	Low Lubrication	High Speed	Bearing Alarm
T1	✓	✓		✓	✓	✓
T2	✓	✓		✓		✓
T3		✓	✓		✓	
T4	✓		✓		✓	
T5	✓	✓		✓	✓	✓
T6			✓			
T7	✓	✓			✓	✓
T8		✓	✓			
T9	✓			✓	✓	
T10	✓	✓		✓		✓

Continuous Values

Association Rule Mining works for discrete data

However continuous data are common in manufacturing

Time	Temperature	Vibration	Pressure
10:00	72.3	4.1	101
10:10	75.8	4.8	103
10:20	81.2	6.4	105

Discretize the data (a type of preprocessing)

$$Temperature > 80 \rightarrow HighTemperature$$

$$Vibration > 6 \rightarrow HighVibration$$

Diagnostic Analytics

Association does not imply causation

Design of Experiments can be used to refine association rules further.

Quiz 11

1. Write 2 things that you liked/understood today.
2. Write 2 things that you did not like/understand today.

3. Give one example or use case where association rule mining can be used. This need not be fully derived. Give basic idea and one example rule. Example from your direct experience will be preferred.