



## Data Mining Unit 2 - Unit 2

Data Mining And Analytics (SRM Institute of Science and Technology)

# Data Mining

## Unit-2

### \* Mining frequent Patterns :-

\* Frequent patterns are patterns that appear frequently in a dataset. Set of all such item is called frequent itemset.

\* It was first proposed by Agarwal, Imielinski and Swami.

\* Sequential frequent pattern :- Buying a digital camera and then memory card if it occurs frequently is called frequent sequential pattern.

### \* Market Basket Analysis :-

Frequent itemset mining leads to discovery of association and correlation among itemset in large transactional dataset.

The process of analyzing customer habbit by finding association between items and customer is placed in the shopping basket.

$C_1 \rightarrow$  Milk, Bread, Cereals

$C_2 \rightarrow$  Milk, Bread, Sugar, Eggs

$C_3 \rightarrow$  Milk, Bread, butter

$C_4 \rightarrow$  Sugar, Egg

Itemset  $\rightarrow \{\text{Milk, Bread, Cereal, Sugar, Egg, Butter}\}$

Frequent Itemset  $\rightarrow \{\text{Milk, Bread, Sugar, Egg}\}$

- \* Association between any two item is represented in the form of association rules.  
For example, the information that customer who ~~bought~~ buy computer also buy antivirus at same time is represented in association rule as  
 $\text{Computer} \rightarrow \text{Antivirus}$  [Support = 2%, Confidence = 60%]

In General:  $A \rightarrow B$  [Support, Confidence]

$$\begin{aligned} \text{Support} &= A \cup B && [\text{A \& B together}] \\ \text{Confidence} &= A/B && [\text{Buyed A also buyed B}] \end{aligned}$$

\* The frequent itemset of k item is denoted by  $L_k$ .

\* Threshold :- Minimum Support & Confidence above which association can be said interesting.

### \* Formula

Let  $I = \{I_1, I_2, \dots, I_n\}$  be a set of items. Let  $D$  be a set of database transaction where each transaction  $T$  is a set of items such that  $T \subseteq I$ . Each transaction is associated with an identifier TID. Let  $A$  be set of items where  $A$

transaction  $T$  is said to contain  $A$  iff  $A \subseteq T$ .

An association rule is implication of form  
 $A \rightarrow B$  where  $A \subseteq I$ ,  $B \subseteq I$ ,  $A \cap B = \emptyset$

$$\text{Support}(A \rightarrow B) = P(A \cup B)$$

$$\text{Confidence}(A \rightarrow B) = P(A|B).$$

Rules that satisfy both minimum support & confidence are called strong.

### \* Steps:-

Association rule mining can be viewed as 2 step process :-

- (i) Find all frequent subset with min-support count.
- (ii) Generate Association rule from frequent itemset.

### \* Problem

Drawback :- It often generate huge number of itemset satisfying minimum support threshold.  
For example, a frequent set of length 100 contains

$\binom{100}{1}$  frequent itemset of 1 item

$\binom{\frac{100}{2}}{2}$  frequent itemset of 2 item.

$$\Rightarrow \binom{100}{1} + \binom{100}{2} + \binom{100}{3} + \dots + \binom{\frac{100}{2}}{100} = 2^{100} - 1 \approx 1.22 \times 10^{30}$$

### \* Solution.

There is too huge number of itemset for any computer to compute or store. To overcome this difficulty we introduce the concept of frequent closed frequent itemset and maximal frequent itemset.

- \* An itemset  $X$  is closed in a dataset  $S$  if there exist no proper super-itemset  $Y$  such that  $Y$  contains same support count as  $X$ .
- \* An itemset  $X$  is closed frequent itemset in a set  $S$  if  $X$  is both closed & frequent.
- \* An itemset  $X$  is a maximal frequent itemset in a set  $S$  if  $X$  is frequent and there exist no super-itemset  $Y$  such that  $X \subset Y$  and  $Y$  is frequent in  $S$ .

### \* APRIORI Algorithm :- Finding frequent itemset Using Candidate Generation

TID	List of Items
T <sub>100</sub>	I <sub>1</sub> , I <sub>2</sub> , I <sub>5</sub>
T <sub>200</sub>	I <sub>2</sub> , I <sub>4</sub>
T <sub>300</sub>	I <sub>2</sub> , I <sub>5</sub>
T <sub>400</sub>	I <sub>1</sub> , I <sub>2</sub> , I <sub>4</sub>
T <sub>500</sub>	I <sub>1</sub> , I <sub>3</sub>
T <sub>600</sub>	I <sub>2</sub> , I <sub>5</sub>
T <sub>700</sub>	I <sub>1</sub> , I <sub>5</sub>
T <sub>800</sub>	I <sub>1</sub> , I <sub>2</sub> , I <sub>3</sub> , I <sub>5</sub>
T <sub>900</sub>	I <sub>1</sub> , I <sub>2</sub> , I <sub>3</sub>

Minimum-Support = 2

$\Rightarrow C_1$

Itemset	Support-Count
$\{I_1\}$	6
$\{I_2\}$	7
$\{I_3\}$	6
$\{I_4\}$	2
$\{I_5\}$	2

$L_1$

Itemset	Support-Count
$\{I_1\}$	6
$\{I_2\}$	7
$\{I_3\}$	6
$\{I_4\}$	2
$\{I_5\}$	2

Now, we will generate  $C_2$  by joining  $L_k \bowtie L_k$   
 This Join is possible when we have  $k-1$  item in common.

$C_2$  :- Itemset

Itemset	Support-Count
$\{I_1, I_2\}$	4
$\{I_2, I_3\}$	4
$\{I_1, I_4\}$	1
$\{I_1, I_5\}$	2
$\{I_2, I_3\}$	4
$\{I_2, I_4\}$	2
$\{I_2, I_5\}$	2
$\{I_3, I_4\}$	0
$\{I_3, I_5\}$	1
$\{I_4, I_5\}$	0

Itemset

Itemset	Support-Count
$\{I_1, I_2\}$	4
$\{I_1, I_3\}$	4
$\{I_1, I_5\}$	2
$\{I_2, I_3\}$	4
$\{I_2, I_4\}$	2
$\{I_2, I_5\}$	2
$\{I_3, I_4\}$	2
$\{I_3, I_5\}$	2
$\{I_4, I_5\}$	2

Key pruning  
is performed

(Removal of  
set whose  
Sup-Count < min-Sup)

Now, we generate  $C_3$  with  $L_2 \bowtie L_2$ .

$C_3$  :-

Itemset	Support-Count
$\{I_1, I_2, I_3\}$	2
$\{I_1, I_2, I_5\}$	2
$\{I_1, I_2, I_4\}$	1
$\{I_1, I_3, I_5\}$	1
$\{I_2, I_3, I_4\}$	0
$\{I_2, I_3, I_5\}$	1
$\{I_2, I_4, I_5\}$	0

on joining

$L_3$  :-

Itemset	Support-Count
$\{I_1, I_2, I_3\}$	2
$\{I_1, I_2, I_5\}$	2

On again performing  $L_3 \bowtie L_3$  to get  $C_4$ .

$C_4$ :-	Itemset	Support-Count
	$\{I_1, I_2, I_3, I_5\}$	1

$L_4$  :- — —

Q Why it is called Apriori?  
 $\Rightarrow$  Because we are utilizing prior information at each step.

Apriori Property :- All non-empty subset of a frequent itemset must also be frequent.

This method is pruning & property is Apriori property. This property belongs to special category of property called antimonotone in sense that if if a set cannot pass a test, all its subsets can also not pass it!

Problem :- We basically have to scan entire database again and again. (frequent scanning).

### Generating Association Rules :-

$$\text{Confidence} = P(B|A) = \frac{\text{Support}(A \cup B)}{\text{Support}(A)}$$

Q Suppose the data contains frequent itemset  $I = \{I_1, I_2, I_3\}$ . What are the association rules that can be generated from  $I$ ? Threshold = 75%.

$$\Rightarrow X \{I_1, I_2\} \rightarrow \{I_3\} = \frac{\text{Support}(A \cup B)}{\text{Support}(A)} = \frac{2}{4} = 50\%$$

$$\{I_1, I_3\} \rightarrow \{I_2\} = \frac{2}{2} = 100\%$$

$$\{I_2, I_3\} \rightarrow \{I_1\} = \frac{2}{2} = 100\%$$

$$X \{I_1\} \rightarrow \{I_2, I_3\} = 2/6 = 33.33\%$$

$$X \{I_2\} \rightarrow \{I_1, I_3\} = 2/7 \approx 29\%$$

$$\{I_3\} \rightarrow \{I_1, I_2\} = 2/2 = 100\%.$$

$\Rightarrow$  2<sup>nd</sup>, 3<sup>rd</sup> & 6<sup>th</sup> rule are output.

# Improving the efficiency of Apriori Algo:

1. Hashing technique :- A hash based technique can be used to reduce the size of candidate k-sets.

For example :-

$$I_1 \rightarrow 1$$

$$I_2 \rightarrow 2$$

$$I_3 \rightarrow 3$$

$$I_4 \rightarrow 4$$

$$I_5 \rightarrow 5$$

For hash function,  $h(u, y) = (\text{order of } u) + (\text{ord of } y) \pmod N$

Index	0	1	2	3	4	5	6
Count	1	2	3	2	3	3	1
Content	$\{I_4, I_5\}$	$\{I_1, I_2\}$ $\{I_2, I_3\}$	$\{I_1, I_2\}$ $\{I_3, I_5\}$	$\{I_4, I_5\}$ $\{I_1, I_3\}$	$\{I_3, I_5\}$ $\{I_1, I_2\}$	$\{I_1, I_2\}$ $\{I_2, I_3\}$ $\{I_3, I_4\}$	$\{I_1, I_2\}$ $\{I_2, I_3\}$ $\{I_3, I_4\}$ $\{I_4, I_5\}$

$$h(I_1, I_2) = (10 \times (1) + (2)) \pmod 7$$

$$= 12 \pmod 7$$

$$= 5$$

If min support is 3. Then items at index 0, 1, 3, 6 can be removed and only items at index 2, 4, 5 will be considered.

2. Transaction Reduction :- It meant reducing the no. of total transaction scanned in future iteration.

$$I_1 \rightarrow 6, \quad I_2 \rightarrow 7, \quad I_3 \rightarrow 6, \quad I_4 \rightarrow 2, \quad I_5 \rightarrow 1$$

for minors of 3, we can remove  $I_4$  &  $I_5$  by  
transaction reduction.

3 Partitioning :- It involves partitioning the data to find candidate items.

It consists of 2 phases.

- ① Partition of data & find local frequent pattern

② Consolidate global frequent pattern.

4

Sampling :- It involves mining on a subset of a given data. The basic idea is to pick a sample from the data and search for frequent itemset in sample instead of searching entire data.

Size of Sample should be that it ~~should~~ can be stored in memory. Because if this does not happen then we will have to store in secondary memory in which time wastage will be more.

## FP Growth Algorithm :-

$$T_1 : \{ E, K, M, N, O, Y \}$$

$$T_2 : \{ D, E, K, N, O, Y \}$$

$$T_3 : \{ A, E, K, M \}$$

$$T_4 : \{ C, K, M, U, Y \}$$

$$T_5 : \{ C, E, I, K, O, O \}$$

Min Supp=3

Step 1 → Generate the frequency Table.

Item	Frequency
A	1
C	2
D	1
E	4
I	1
K	5
M	3
N	2
O	3
R	1
U	1
Y	3

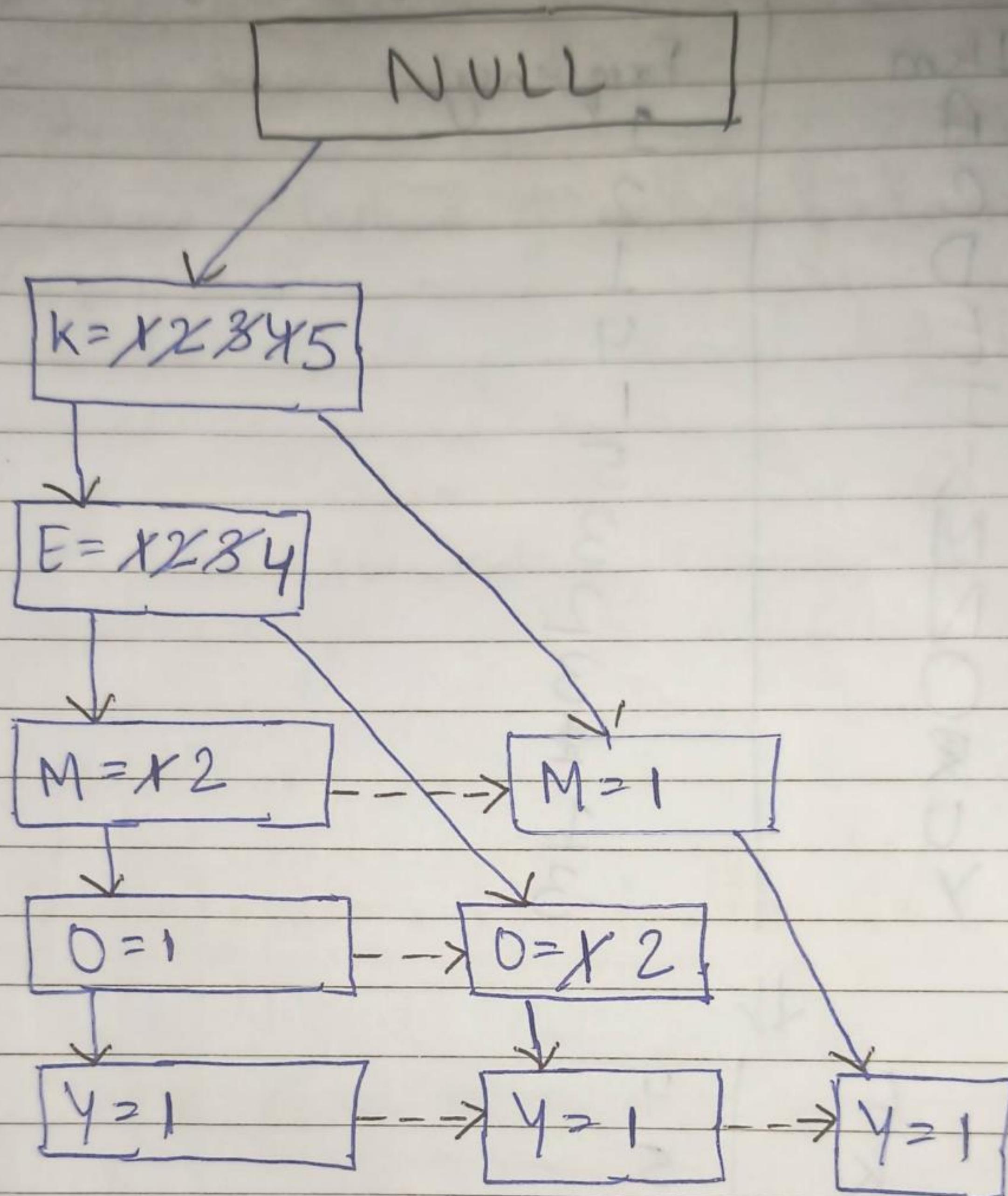
↓  
↓  
↓

Item	Frequency
E	4
K	5
M	3
O	3
Y	3

Ordered  $\{K=5, E=4, M=3, O=3, Y=3\}$

TID	Item Bought	(Ordered) frequent Item
T <sub>1</sub>	{E, K, M, N, O, Y}	{K, E, M, O, Y}
T <sub>2</sub>	{D, E, K, N, O, Y}	{K, E, O, Y}
T <sub>3</sub>	{A, E, K, M}	{K, E, M}
T <sub>4</sub>	{C, K, M, U, Y}	{K, M, Y}
T <sub>5</sub>	{C, E, I, K, O, D}	{K, E, O}

Step 2 : Draw frequent Pattern Tree



Condition Pattern Base :-

$$Y = \{\{K, E, M, O: 1\}, \{K, E, O: 1\}, \{K, M: 1\}\}$$

$$O = \{\{K, E, M: 1\}, \{K, E: 2\}\}$$

$$M = \{\{K, E: 2\}, \{K: 1\}\}$$

$$E = \{\{K: 4\}\}$$

$$K = \{\}\}$$

## Conditional frequent Pattern Tree

Y: {K: 3}

O: {K, E: 3}

M: {K: 3}

E: {K: 4}

Frequent Patterns Generated :-

Y: {K, Y: 3}

O: {K, O: 3}, {E, O: 3}, {K, E, O: 2}

M: {K, M: 3}

E: {K, E: 4}

Benefits :- (i) No need to have Candidate Selection after each iteration.

(ii) Complexity Reduce :- No need to scan whole database which was taking  $O(n^2)$  time. we just need to scan tree which uses Binary Search  $O(\log n)$  time.

\* Mining Frequent itemset using Vertical Data format :-

Consider TID and Data set Table of Apriori example. That table was based on flat Transaction. Vertical Data involves Table according to items.

Items	TID SJ
I <sub>1</sub>	T <sub>100</sub> , T <sub>400</sub> , T <sub>500</sub> , T <sub>700</sub> , T <sub>800</sub> , T <sub>900</sub>
I <sub>2</sub>	T <sub>100</sub> , T <sub>200</sub> , T <sub>300</sub> , T <sub>400</sub> , T <sub>600</sub> , T <sub>800</sub> , T <sub>900</sub>
I <sub>3</sub>	T <sub>300</sub> , T <sub>500</sub> , T <sub>600</sub> , T <sub>700</sub> , T <sub>800</sub> , T <sub>900</sub>
I <sub>4</sub>	T <sub>200</sub> , T <sub>400</sub>
I <sub>5</sub>	T <sub>100</sub> T <sub>800</sub>

ItemSet	TID SJ	Count
{I <sub>1</sub> , I <sub>2</sub> }	T <sub>100</sub> T <sub>900</sub> T <sub>800</sub> T <sub>900</sub>	4
{I <sub>1</sub> , I <sub>3</sub> }	T <sub>500</sub> T <sub>700</sub> T <sub>800</sub> T <sub>900</sub>	4
x {I <sub>1</sub> , I <sub>4</sub> }	T <sub>400</sub>	1
{I <sub>1</sub> , I <sub>5</sub> }	T <sub>100</sub> T <sub>800</sub>	2
{I <sub>2</sub> , I <sub>3</sub> }	T <sub>300</sub> T <sub>600</sub> T <sub>800</sub> T <sub>900</sub>	4
{I <sub>2</sub> , I <sub>4</sub> }	T <sub>200</sub> T <sub>400</sub>	2
{I <sub>2</sub> , I <sub>5</sub> }	T <sub>100</sub> T <sub>800</sub>	2
x {I <sub>3</sub> , I <sub>4</sub> }		0
x {I <sub>3</sub> , I <sub>5</sub> }	T <sub>800</sub>	1
x {I <sub>4</sub> , I <sub>5</sub> }		0

Itemset	TID	Count
{I, I <sub>2</sub> I <sub>3</sub> }	T <sub>800</sub> , T <sub>900</sub>	2
{I, I <sub>2</sub> I <sub>5</sub> }	T <sub>800</sub> , T <sub>900</sub>	2
X {I <sub>1</sub> , I <sub>2</sub> , I <sub>4</sub> }		
{I <sub>1</sub> I <sub>3</sub> I <sub>5</sub> }		
{I <sub>2</sub> I <sub>3</sub> I <sub>4</sub> }		
{I <sub>1</sub> I <sub>3</sub> I <sub>4</sub> }		
{I <sub>2</sub> I <sub>4</sub> I <sub>5</sub> }		
{I}		
X {I, I <sub>2</sub> I <sub>3</sub> I <sub>5</sub> }		0

## Support, Confidence :- Strong vs Weak Rule

\* Strong  $\rightarrow$  If Support & Confidence both greater than min\_sup & min\_confidence respectively.

\* Weak  $\rightarrow$  Otherwise

<u>Q</u>	10,000 transaction
6,000	Computer game purchase
7,500	Video game purchase
4,000	Both game <sup>for</sup> & video

$$\text{min\_sup} = 30\%$$

$$\text{min\_confidence} = 60\%$$

$Sol \rightarrow \text{buy}(x, \text{game}) \rightarrow \text{buy}(x, \text{video}).$

$$\text{Support} = P(A \cup B)$$

$$= \frac{4000}{10000} \times 100\%$$

$$= 40\%.$$

$$\text{Confidence} = P(A|B) = \frac{\text{Support}(A \cup B)}{\text{Support}(A)}$$

$$= \frac{4000}{6000} \times 100\%$$

$$= 66.66\%$$

Both greater than its respective threshold

$\Rightarrow$  Strong rule.

However this rule is misleading because probability of purchasing video is 75% which is even larger than confidence (66%). In fact, they ~~are~~ are negatively associated because purchase of one item actually decreases the likelihood of purchasing other.

So, to solve this problem of misleading there must be some other factor in rules alongside support & confidence.

And this is where the concept of correlation came up which is called Correlation rules of form

Now, Correlation can be find in multiple ways.

$A \rightarrow B$  [Support, Confidence, Correlation]

1) Lift :- Lift is simple correlation measure that is given as follows :-

$$\text{Lift} = \frac{P(A \cup B)}{P(A) P(B)}$$

If value of lift  $< 1$   $\rightarrow$  negatively correlated

If value of lift  $= 1$   $\rightarrow$  Event are independent

If value of lift  $> 1$   $\rightarrow$  positively correlated.

$\downarrow$   
Occurrence of one  $\Rightarrow$  occurrence of other.

For above example,

$$\text{lift(game, video)} = \frac{\frac{4000}{10000}}{\frac{6000}{10000} \times \frac{2000}{10000}} = 0.89$$

Lift  $< 1$ .

So, there is negative correlation between them.

## 2) Correlation analysis using $\chi^2$ test

	video	<u>video</u>	
game	4000	2000	6000
<u>game</u>	3500	500	4000
	7500	2500	10000

$$\text{Expected (4000)} = \frac{6000 \times 7500}{10000} = 4500$$

$$E(2000) = \frac{2000 \times 6000}{10000} = 1200$$

$$E(3500) = \frac{7500 \times 4000}{10000} = 3000$$

$$E(500) = \frac{2500 \times 4000}{10000} = 1000$$

	game	video	<u>video</u>
<del>video game</del>	4000 (4500)	2000 (1500)	
game	3500 (3000)	500 (1000)	

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

$$\begin{aligned}
 &= \frac{500^2}{4500} + \frac{500^2}{3000} + \frac{500^2}{1500} + \frac{500^2}{1000} \\
 &= 55.55 + 166.67 + 250 + 83.33 \\
 &= 555.55 \\
 &= 555.55
 \end{aligned}$$

Because value of  $\chi^2$  is greater than 1 and observed value of slot (game, video) = 4000 which is less than expected value of 4500. So, it is negatively correlated.

$$\begin{aligned}
 \underline{3. AU\text{-Confidence}} &= \frac{\text{Support}(A \cup B)}{\max(\text{Support}(A), \text{Support}(B))} \\
 &= \min(P(B|A), P(A|B))
 \end{aligned}$$

Kulczyński Measure :-

$$\frac{1}{2} (P(A|B) + P(B|A)).$$

$$\underline{4. Cosine Measure} : \frac{\text{Support}(A \cup B)}{\sqrt{\text{Support}(A) \times \text{Support}(B)}}$$

$$\begin{aligned}
 \text{Max Confidence} &= \max(P(B|A), P(A|B)) \\
 &= \frac{\text{Support}(A \cup B)}{\min(\text{Support}(A), \text{Support}(B))}
 \end{aligned}$$

for our Example of game & video :-

$$\text{all-cos}(\text{game}, \text{v.video}) = \frac{4000}{7500}$$
$$= 0.53$$

$$\text{Cosine} = \frac{4000}{\sqrt{7500 \times 6000}}$$

$$= 0.596$$

$$= 0.60$$

Based on note that it is greater than 0.5  
So, it is +vely correlated in allcos & cosine.  
Since it was -vely correlated in Lift &  
 $\chi^2$ . So, good way is to first find it  
by all-cos & cosine. Then if they  
are weakly correlated then only other  
analysis can be done.