Ch5 Mining Frequent Patterns, Associations, and Correlations

Dr. Bernard Chen Ph.D.
Troy University

Outline

- Association Rules
- Misleading Rules
- Multi-level Association Rules

What Is Frequent Pattern Analysis?

- Frequent pattern: a pattern (a set of items, subsequences, substructures, etc.) that occurs frequently in a data set

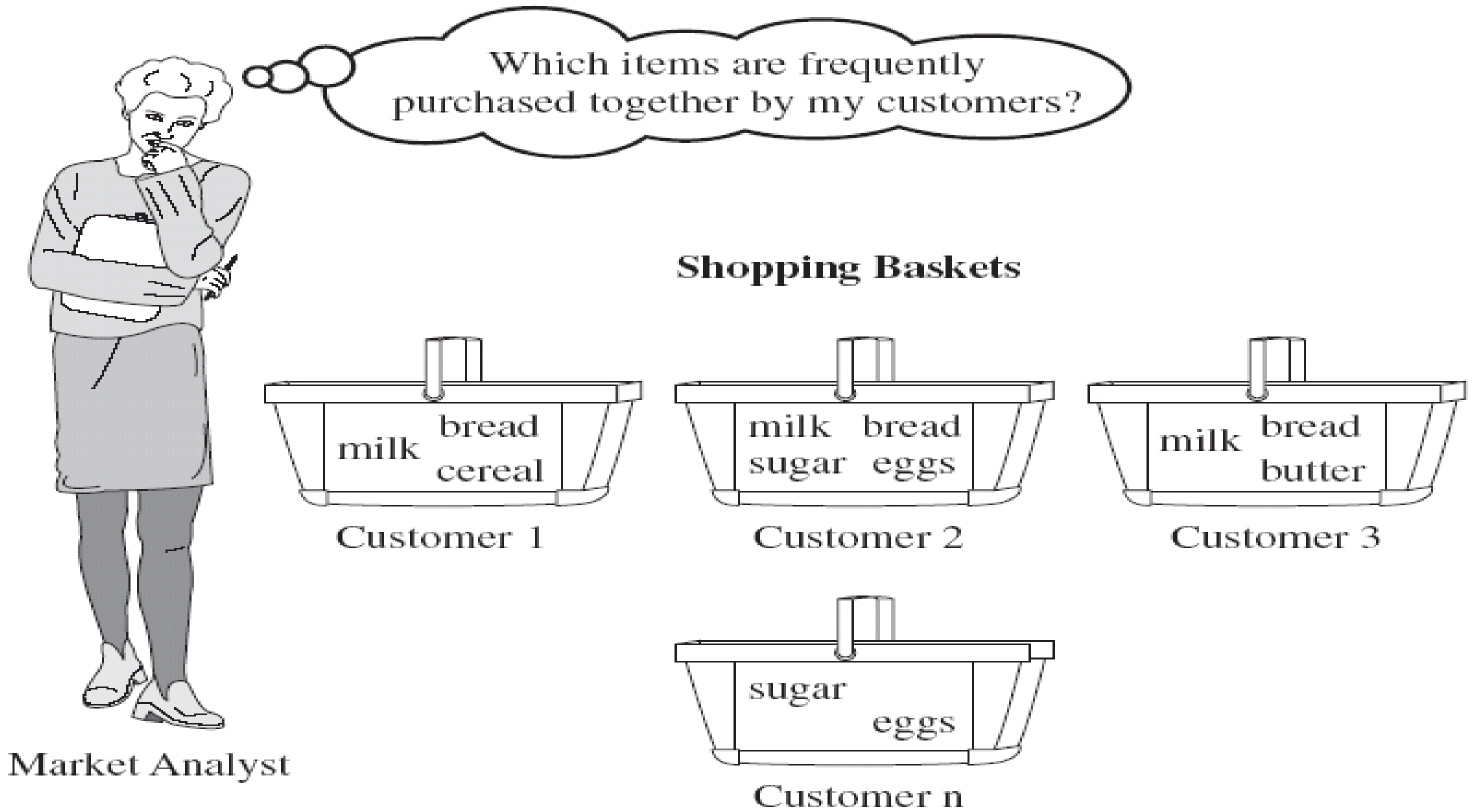
- First proposed by Agrawal, Imielinski, and Swami [AIS93] in the context of frequent itemsets and association rule mining

What Is Frequent Pattern Analysis?

- Motivation: Finding inherent regularities in data
  - What products were often purchased together? bread and milk?
  - What are the subsequent purchases after buying a PC?
  - What kinds of DNA are sensitive to this new drug?
  - Can we automatically classify web documents?

- Applications
  - Basket data analysis, cross-marketing, catalog design, sale campaign analysis, Web log (click stream) analysis, and DNA sequence analysis.

Association Rules



Association Rules

- support, *s*, probability that a transaction contains X ∪ Y

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

- confidence, *c*, conditional probability that a transaction having X also contains *Y*

$$Confidence(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$$

Association Rules

- Then user need to define:
  - Minimum Support
  - Minimum Confidence

Let us focus on SUPPORT

- support, *s*, probability that a transaction contains X ∪ Y

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

Association Rules

- Let’s have an example
  - T100 1,2,5
  - T200 2,4
  - T300 2,3
  - T400 1,2,4
  - T500 1,3
  - T600 2,3
  - T700 1,3
  - T800 1,2,3,5
  - T900 1,2,3

- Suppose our **minimum support** is 2/9

Association Rules

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

- Let's have an example
  - T100 1,2,5
  - T200 2,4
  - T300 2,3
  - T400 1,2,4
  - T500 1,3
  - T600 2,3
  - T700 1,3
  - T800 1,2,3,5
  - T900 1,2,3

- |T|=9

To calculate suppo

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

- Since the grocery has only 5 items, the basic way is to try all items:
- TWO item set
  - (1,2) (1,3) (1,4) (1,5)
  - (2,3) (2,4) (2,5)
  - (3,4) (3,5)
  - (4,5)

Support (1,2) = 4/9 (pass)

Support (2,3) = 4/9 (pass)

Support (3,5) = 1/9

- T100 1,2,5
- T200 2,4
- T300 2,3
- T400 1,2,4
- T500 1,3
- T600 2,3
- T700 1,3
- T800 1,2,3,5
- T900 1,2,3

To calculate suppo $Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$

- Since the grocery has only 5 items, the basic way is to try all items:
- THREE item set
  - (1,2,3) (1,2,4) (1,2,5) (1,3,4) (1,3,5) (1,4,5)
  - (2,3,4) (2,3,5) (2,4,5)
  - (3,4,5)

Support (1,2,3) = 2/9 (pass)

Support (2,3,4) = 0/9

Support (3,4,5) = 0/9

- T100 1,2,5
- T200 2,4
- T300 2,3
- T400 1,2,4
- T500 1,3
- T600 2,3
- T700 1,3
- T800 1,2,3,5
- T900 1,2,3

To calculate suppo $Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$

- Since the grocery has only 5 items, the basic way is to try all items:
- FOUR item set
  - (1,2,3,4) (1,2,3,5)
  - (2,3,4,5)

Support (1,2,3,4) = 0/9

Support (2,3,4,5) = 0/9

- T100 1,2,5
- T200 2,4
- T300 2,3
- T400 1,2,4
- T500 1,3
- T600 2,3
- T700 1,3
- T800 1,2,3,5
- T900 1,2,3

To calculate suppo

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

- Since the grocery has only 5 items, the basic way is to try all items:
- FIVE item set
  - (1,2,3,4,5)

Support (1,2,3,4,5) = 0/9

T100	1,2,5
T200	2,4
T300	2,3
T400	1,2,4
T500	1,3
T600	2,3
T700	1,3
T800	1,2,3,5
T900	1,2,3

To calculate suppo

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

- This is the situation based on our store has only **5** items
- In real grocery store like Kroger, they have **thousands** of items
- Is there a better way?

T100	1,2,5
T200	2,4
T300	2,3
T400	1,2,4
T500	1,3
T600	2,3
T700	1,3
T800	1,2,3,5
T900	1,2,3

To calculate suppo

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

- Of course, it's called the **Apriori**

- It proceeds by identifying the frequent individual items (the item that passes the minimum support) in the database and **extending** them to larger and larger item sets as long as those item sets appear sufficiently often in the database.

To calculate suppo

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

- Start with one item set

Scan D for count of each candidate

C_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Compare candidate support count with minimum support count

L_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

ALL Pass

Generate C_2 candidates from L_1

C_2

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

- T100 1,2,5
- T200 2,4
- T300 2,3
- T400 1,2,4
- T500 1,3
- T600 2,3
- T700 1,3
- T800 1,2,3,5
- T900 1,2,3

To calculate suppo $Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$

Generate C_2 candidates from L_1 →

C_2

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

Scan D for count of each candidate →

C_2

Itemset	Sup. count
{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

Compare candidate support count with minimum support count →

L_2

Itemset	Sup. count
{I1, I2}	4
{I1, I3}	4
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2

Generate C_3 candidates from L_2 →

C_3

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

- T100 1,2,5
- T200 2,4
- T300 2,3
- T400 1,2,4
- T500 1,3
- T600 2,3
- T700 1,3
- T800 1,2,3,5
- T900 1,2,3

To calculate suppo

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

Generate C_3 candidates from L_2 →

C_3

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

Scan D for count of each candidate →

C_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Compare candidate support count with minimum support count →

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

- → {I1, I2, I3, I5}
- Support {I1, I2, I3, I5} = 1/9
- No Five item set can be generated

- T100 1,2,5
- T200 2,4
- T300 2,3
- T400 1,2,4
- T500 1,3
- T600 2,3
- T700 1,3
- T800 1,2,3,5
- T900 1,2,3

Association Rules with Apriori

Minimum support=2/9

Minimum confidence=60%

Scan D for count of each candidate →

C_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Compare candidate support count with minimum support count →

L_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Generate C_2 candidates from L_1 →

C_2

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

Scan D for count of each candidate →

C_2

Itemset	Sup. count
{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

Compare candidate support count with minimum support count →

L_2

Itemset	Sup. count
{I1, I2}	4
{I1, I3}	4
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2

Generate C_3 candidates from L_2 →

C_3

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

Scan D for count of each candidate →

C_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Compare candidate support count with minimum support count →

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

The Apriori Algorithm

- Pseudo-code:

  C_k: Candidate itemset of size k
  L_k : frequent itemset of size k

  L_1 = {frequent items};
  for (k = 1; L_k !=∅; k++) **do begin**
  C_{k+1} = candidates generated from L_k;
  for each transaction t in database do
  increment the count of all candidates in C_{k+1} that are contained in t
  L_{k+1} = candidates in C_{k+1} with min_support
  end
  return $\cup_k L_k$;

Exercise

- A dataset has five transactions, let min-support=60%
- Find all frequent item sets using Apriori

TID	Items_bought
T1	M, O, N, K, E, Y
T2	D, O, N, K , E, Y
T3	M, A, K, E
T4	M, U, C, K ,Y
T5	C, O, O, K, I ,E

Association Rules

- support, *s*, probability that a transaction contains X ∪ Y

Precedent

$$Support(X \Rightarrow Y) = \frac{|X \cup Y|}{|T|}$$

Consequent

- confidence, *c*, conditional probability that a transaction having X also contains *Y*

Precedent

$$Confidence(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$$

Consequent

Association Rules

- Now, Let us take a look of CONFIDENCE
  - Suppose our **minimum confidence:60%**

- confidence, *c*, conditional probability that a transaction ... tains *Y*

$$Confidence(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$$

Strong Association Rule

- **Strong association rules** means the frequent rules that **also** pass the minimum confidence.

$$Confidence(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$$

- For example frequent rules: {I1,
  - Confidence(I1->I2)= 4/6 pass 60% (strong association rule!)
  - Confidence(I2->I1)= 4/7

T100	1,2,5
T200	2,4
T300	2,3
T400	1,2,4
T500	1,3
T600	2,3
T700	1,3
T800	1,2,3,5
T900	1,2,3

Strong Association Rule

- Confidence for 2 item set (1,2) (1,3) (1,5)
  - C(1->2)= |1 ∪ 2|/|1| = 4/6
  - C(2->1)= |1 ∪ 2|/|2| = 4/7
  - C(1->3)= |1 ∪ 3|/|1| = 4/6
  - C(3->1)= |1 ∪ 3|/|3| = 4/6
  - C(1->5)= |1 ∪ 5|/|1| = 2/6
  - C(5->1)= |1 ∪ 5|/|5| = 2/2 (100%

$$Confidence(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$$

- T100 1,2,5
- T200 2,4
- T300 2,3
- T400 1,2,4
- T500 1,3
- T600 2,3
- T700 1,3
- T800 1,2,3,5
- T900 1,2,3

Strong Association Rule

- Confidence for 2 item set (2,3) (2,4) (2,5)
  - C(2->3)= |2 ∪ 3|/|2| = 4/7
  - C(3->2)= |2 ∪ 3|/|3| = 4/6
  - C(2->4)= |2 ∪ 4|/|2| = 4/7
  - C(4->2)= |2 ∪ 4|/|4| = 2/2 (100%
  - C(2->5)= |2 ∪ 5|/|2| = 2/7
  - C(5->2)= |2 ∪ 5|/|5| = 2/2 (100%

$$Confidence(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$$

T100	1,2,5
T200	2,4
T300	2,3
T400	1,2,4
T500	1,3
T600	2,3
T700	1,3
T800	1,2,3,5
T900	1,2,3

Strong Association Rule

- Confidence for 3 itemset (1, 2, 5)
  - C(1->2,5)=2/6
  - C(2->1,5)=2/7
  - C(5->1,2)=2/2
  - C(1,2->5)=2/4
  - C(1,5->2)=2/2
  - C(2,5->1)=2/2

$$Confidence(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$$

T100	1,2,5
T200	2,4
T300	2,3
T400	1,2,4
T500	1,3
T600	2,3
T700	1,3
T800	1,2,3,5
T900	1,2,3

Strong Association Rule

- Confidence for 3 itemset (1, 2, 3)
  - C(1->2,3)=2/6
  - C(2->1,3)=2/7
  - C(3->1,2)=2/6
  - C(1,2->3)=2/4
  - C(1,3->2)=2/4
  - C(2,3->1)=2/4

$$Confidence(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$$

- T100 1,2,5
- T200 2,4
- T300 2,3
- T400 1,2,4
- T500 1,3
- T600 2,3
- T700 1,3
- T800 1,2,3,5
- T900 1,2,3

Exercise

- A dataset has five transactions, let min-support=60% and min_confidence=80%
- Find all **strong association rules**

TID	Items_bought
T1	M, O, N, K, E, Y
T2	D, O, N, K , E, Y
T3	M, A, K, E
T4	M, U, C, K ,Y
T5	C, O, O, K, I ,E

Association Rules with Apriori

K:5		KE:4		KE		
E:4		KM:3		KM		
M:3		KO:3		KO		
O:3	=>	KY:3	=>	KY	=>	KEO
Y:3		EM:2		EO		
		EO:3				
		EY:2				
		MO:1				
		MY:2				
		OY:2				

Outline

- Association Rules
- Misleading Rules
- Multi-level Association Rules

Example 5.8 Misleading "Strong" Association Rule

- Of the 10,000 transactions analyzed, the data show that
  - 6,000 of the customer included computer games,
  - while 7,500 include videos,
  - And 4,000 included both computer games and videos
    - $P(A \cup B)=40\%$

Misleading "Strong" Association Rule

- For this example:
  - Support (Game & Video) = 4,000 / 10,000 =40%
  - Confidence (Game => Video) = 4,000 / 6,000 = 66%
  - Suppose it pass our minimum support and confidence (30% , 60%, respectively)

Misleading "Strong" Association Rule

- However, the truth is : "computer games and videos are <u>negatively</u> associated"
- Which means the purchase of one of these items actually decreases the likelihood of purchasing the other.
- (How to get this conclusion??)

Misleading "Strong" Association Rule

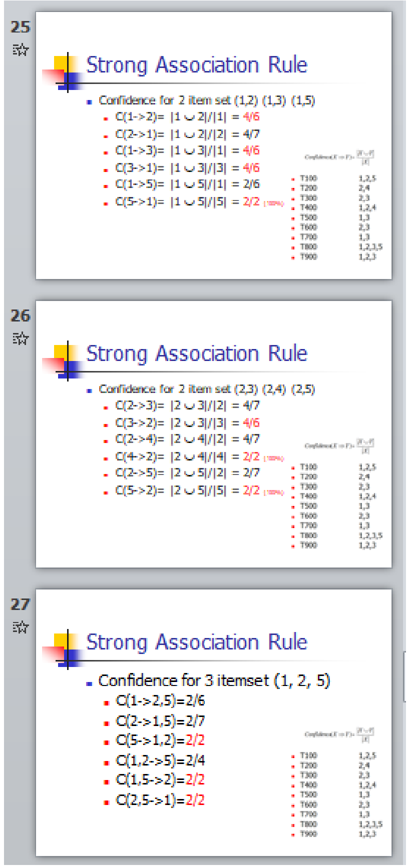
- Under the normal situation,
  - 60% of customers buy the game
  - 75% of customers buy the video
  - Therefore, it should have 60% * 75% = 45% of people buy both
  - That equals to 4,500 which is more than 4,000 (the actual value)

From Association Analysis to Correlation Analysis

- **Lift** is a simple correlation measure that is given as follows
  - The occurrence of itemset A is independent of the occurrence of itemset B if
    $P(A \cup B) = P(A)P(B)$
  - Otherwise, itemset A and B are **dependent** and **correlated** as events

- **Lift(A,B) = $P(A \cup B) / P(A)P(B)$**
  - If the value is less than 1, the occurrence of A is negatively correlated with the occurrence of B
  - If the value is greater than 1, then A and B are positively correlated
  - **$P(A \cup B) / P(A)P(B) = 0.4 / (0.6*0.75) = 0.4/0.45 < 1$**

Lift in our example



Scan D for count of each candidate →

C_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Compare candidate support count with minimum support count →

L_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Generate C_2 candidates from L_1 →

C_2

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

Scan D for count of each candidate →

C_2

Itemset	Sup. count
{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

Compare candidate support count with minimum support count →

L_2

Itemset	Sup. count
{I1, I2}	4
{I1, I3}	4
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2

Generate C_3 candidates from L_2 →

C_3

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

Scan D for count of each candidate →

C_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Compare candidate support count with minimum support count →

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Lift in our example

Scan D for count of each candidate →

C_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Compare candidate support count with minimum support count →

L_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Generate C_2 candidates from L_1 →

C_2

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

Scan D for count of each candidate →

C_2

Itemset	Sup. count
{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

Compare candidate support count with minimum support count →

L_2

Itemset	Sup. count
{I1, I2}	4
{I1, I3}	4
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2

Generate C_3 candidates from L_2 →

C_3

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

Scan D for count of each candidate →

C_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Compare candidate support count with minimum support count →

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Lift in our example

- C(1->2)

P(1,2)/P(1)*P(2)

= (4/9) / (6/9)*(7/9)

= 0.86

- C(1->3)

P(1,3)/P(1)*P(3)

= (4/9) / (6/9)*(6/9)

= 1.01

L_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

L_2

Itemset	Sup. count
{I1, I2}	4
{I1, I3}	4
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Lift in our example

- C(5->1,2)

P(5,1,2)/P(5)*P(1,2)

= (2/9) / (2/9)*(4/9)

= 2.25

- C(2,5->1)

P(2,5,1)/P(2,5)*P(1)

= (2/9) / (2/9)*(6/9)

= 1.5

L_1

Itemset	Sup. count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

L_2

Itemset	Sup. count
{I1, I2}	4
{I1, I3}	4
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

From Association Analysis to Correlation Analysis

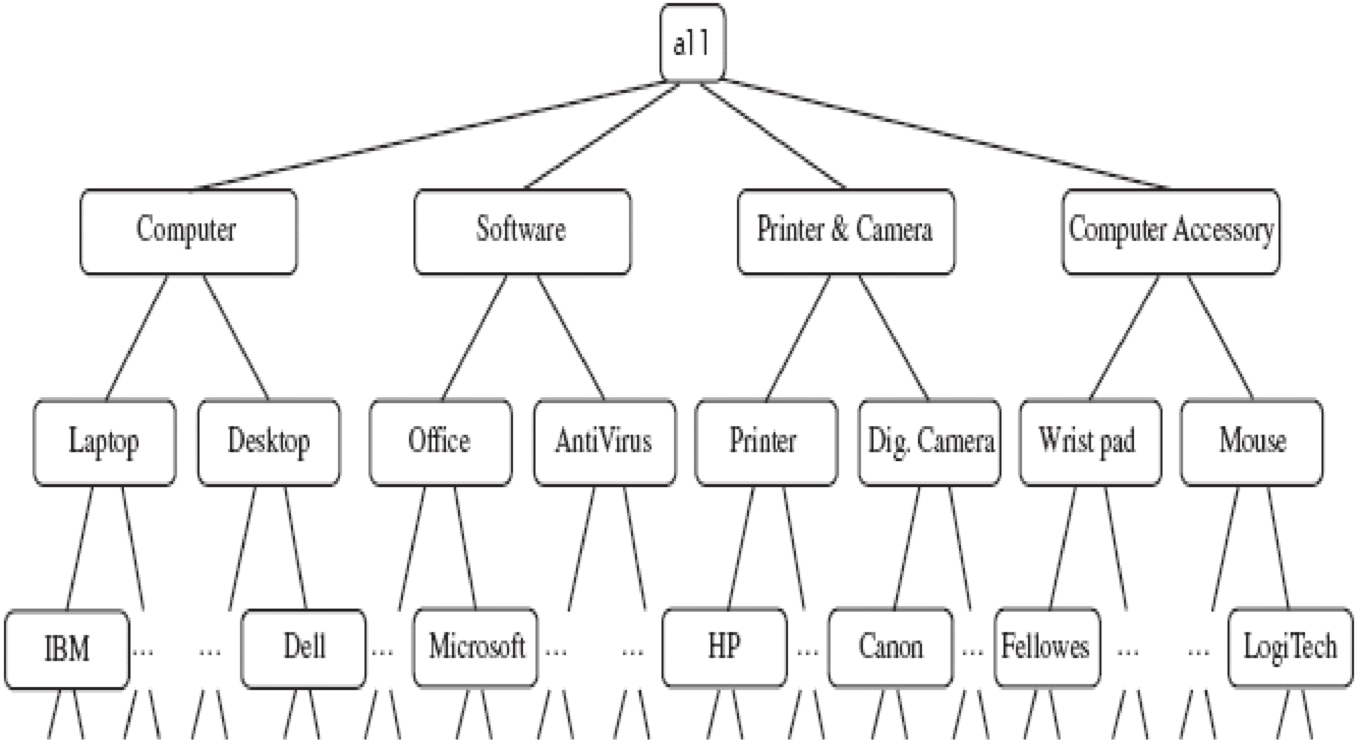
Strong Association Rules	Lift
C(1->2)	P(1,2)/P(1)*P(2) = (4/9) / (6/9)*(7/9) = 0.86
C(1->3)	P(1,3)/P(1)*P(3) = (4/9) / (6/9)*(6/9) = 1.01
C(3->1)	P(3,1)/P(3)*P(1) = (4/9) / (6/9)*(6/9) = 1.01
C(5->1)	P(5,1)/P(5)*P(1) = (2/9) / (2/9)*(6/9) = 1.49
C(3->2)	P(3,2)/P(3)*P(2) = (4/9) / (6/9)*(7/9) = 0.86
C(4->2)	P(4,2)/P(4)*P(2) = (2/9) / (2/9)*(7/9) = 1.29
C(5->2)	P(5,2)/P(5)*P(2) = (2/9) / (2/9)*(7/9) = 1.29
C(5->1,2)	P(5,1,2)/P(5)*P(1,2) = (2/9) / (2/9)*(4/9) = 2.25
C(1,5->2)	P(1,5,2)/P(1,5)*P(2) = (2/9) / (2/9)*(7/9) = 1.29
C(2,5->1)	P(2,5,1)/P(2,5)*P(1) = (2/9) / (2/9)*(6/9) = 1.5

Outline

- Association Rules
- Misleading Rules
- Multi-level Association Rules

Mining Multiple-Level Association Rules

- Items often form hierarchies



Mining Multiple-Level Association Rules

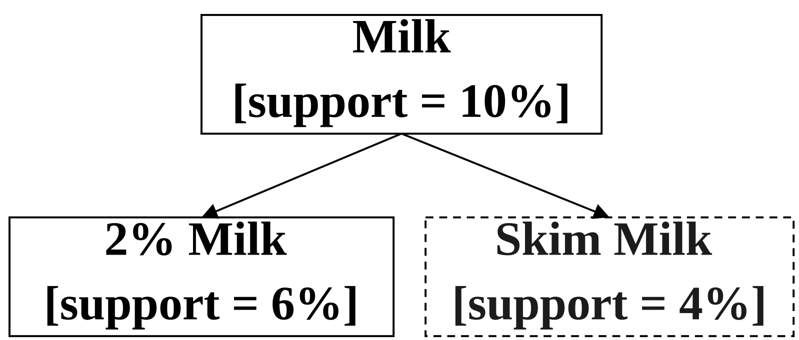
- Flexible support settings
  - Items at the lower level are expected to have lower support

uniform support

reduced support

Level 1
min_sup = 5%

Level 2
min_sup = 5%



Level 1
min_sup = 5%

Level 2
min_sup = 3%

Multi-level Association: Redundancy Filtering

- Some rules may be redundant due to "ancestor" relationships between items.
- Example
  - milk ⇒ wheat bread [support = 8%, confidence = 70%]
  - 2% milk ⇒ wheat bread [support = 2%, confidence = 72%]
- We say the first rule is an ancestor of the second rule.