

Machine Learning

Association Rules

Claudio Sartori

DISI

Department of Computer Science and Engineering – University of Bologna, Italy

claudio.sartori@unibo.it

1	Introduction to Market Basket Analysis	2
●	Support and confidence	6
2	Frequent Itemset Generation	10
3	Rule Generation	26
4	Multidimensional association rules	52
5	Multilevel Association Rules	57

Association Rules – Discovering co-occurrences in a market basket

- Given a set of commercial transactions, find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction

- Example of Association Rules

- $\{\text{Diaper}\} \rightarrow \{\text{Beer}\},$
 - $\{\text{Bread, Milk}\} \rightarrow \{\text{Coke, Eggs}\},$
 - $\{\text{Beer, Bread}\} \rightarrow \{\text{Milk}\}$
 - Implication means co-occurrence, not causality!
 - The implication of Association Rules is different from that of logic (boolean): it can be true **with some level of truth**

<i>TID</i>	<i>Items</i>
1	Bread, Milk
2	Beer, Bread, Diaper, Eggs
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Bread, Coke, Diaper, Milk

Market Basket Transactions

Definition: Frequent Itemset

- **Itemset**
 - A collection of one or more items
 - Example: {Bread, Diaper, Milk}
- **k-itemset**
 - An itemset that contains k items
- **Support count (σ)**
 - Frequency of occurrence of an itemset
 - E.g. $\sigma(\{\text{Bread, Diaper, Milk}\}) = 2$
- **Support**
 - Fraction of transactions that contain an itemset
 - E.g. $\sigma(\{\text{Bread, Diaper, Milk}\}) = 2/5$
- **Frequent Itemset**
 - An itemset whose support is greater than or equal to a **minsup** threshold

<i>TID</i>	<i>Items</i>
1	Bread, Milk
2	Beer, Bread, Diaper, Eggs
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Bread, Coke, Diaper, Milk

Market Basket
Transactions

Definition: Association Rule

- Association Rule
 - An expression of the form $A \Rightarrow C$, where A and C are itemsets
 - A = Antecedent and
 C = Consequent
 - Example: $\{\text{Diaper, Milk}\} \rightarrow \{\text{Beer}\}$
- Rule Evaluation Metrics
 - Support (sup)
 - Fraction of the N transactions that contain both A and C
 - Confidence (conf)
 - Measures how often all the items in C appear in transactions that contain A

TID	Items
1	Bread, Milk
1	Beer, Bread, Diaper, Eggs
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Bread, Coke, Diaper, Milk

Market Basket Transactions

$$\text{sup} = \frac{\sigma(\text{Beer, Diaper, Milk})}{N} = \frac{2}{5} = 0.4$$

$$\text{conf} = \frac{\sigma(\text{Beer, Diaper, Milk})}{\sigma(\text{Milk, Diaper})}$$

Why support and confidence?

- Rules with low support can be generated by random associations
- Rules with low confidence are not really reliable
- Nevertheless a rule with relatively low support but high confidence can represent an uncommon but interesting phenomenon

Association Rule Mining Task

- Given a set of transactions N , the goal of association rule mining is to find all rules having
 - support \geq **minsup** threshold
 - confidence \geq **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

⇒ **Computationally prohibitive!**

Mining Association Rules

Example of Rules:

$$\{Diaper, Milk\} \Rightarrow \{Beer\} \quad (s = 0.4, c = 0.67)$$

$$\{Beer, Milk\} \Rightarrow \{Diaper\} \quad (s = 0.4, c = 1.0)$$

$$\{Beer, Diaper\} \Rightarrow \{Milk\} \quad (s = 0.4, c = 0.67)$$

$$\{Beer\} \Rightarrow \{Diaper, Milk\} \quad (s = 0.4, c = 0.67)$$

$$\{Diaper\} \Rightarrow \{Beer, Milk\} \quad (s = 0.4, c = 0.5)$$

$$\{Milk\} \Rightarrow \{Beer, Diaper\} \quad (s = 0.4, c = 0.5)$$

- All the rules above are binary partitions of the same itemset:
 $\{Beer, Diaper, Milk\}$
- Rules originating from the same itemset have identical support but can have different confidence

\Rightarrow we may decouple the support and confidence requirements

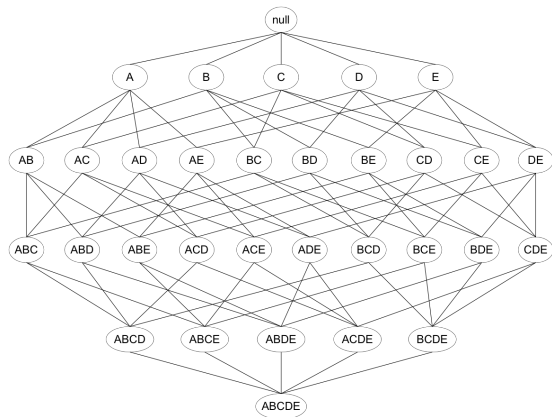
Mining Association Rules

- Two-step approach:
 - 1. Frequent Itemset Generation
 - Generate all itemsets whose support is greater than minsup
 - 2. Rule Generation
 - Generate high confidence rules from each frequent itemset, where each rule is a binary partitioning of a frequent itemset
- Frequent itemset generation is still computationally expensive

1	Introduction to Market Basket Analysis	2
2	Frequent Itemset Generation	10
•	The Apriori principle	16
•	The Apriori algorithm	18
3	Rule Generation	26
4	Multidimensional association rules	52
5	Multilevel Association Rules	57

Frequent Itemset Generation

Given D items, there are $M = 2^D$ possible candidate itemsets



Frequent Itemset Generation

Brute-force approach:

- Each itemset in the lattice is a **candidate** frequent itemset
- Count the support of each candidate by scanning the database
- Match each transaction against every candidate
- Complexity: $\mathcal{O}(NWM) \Rightarrow$ **Expensive**

<i>TID</i>	<i>Items</i>
1	Bread, Milk
1	Beer, Bread, Diaper, Eggs
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Bread, Coke, Diaper, Milk

\uparrow
 \approx
 \downarrow

← W →

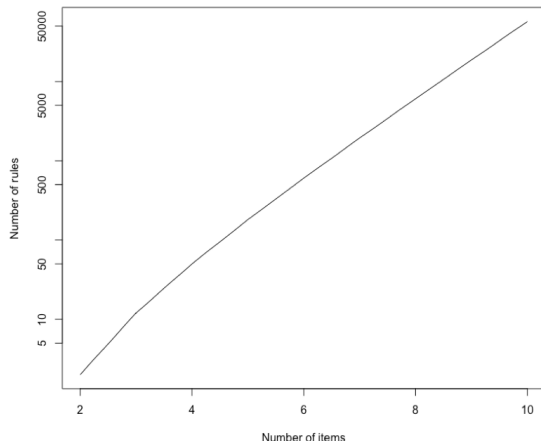
Brute Force - Computational Complexity

OPTIONAL

- Given D unique items:
 - Total number of itemsets = 2^D
 - Total number of possible association rules:

$$R = \sum_{k=1}^{D-1} \left(\binom{D}{k} \times \sum_{j=1}^{D-k} \binom{D-k}{j} \right)$$

$$= 3^D - 2^{D+1} + 1$$



Explanation of the formula

OPTIONAL

- count the number of ways to create an itemset that forms the left hand side of the rule
- for each size k itemset selected for the left-hand side, count the number of ways to choose the remaining $D - k$ items to form the right-hand side of the rule

Going deeper

- choose k of the D items for the left hand side of the rule, there are $\binom{D}{k}$ ways to do this
- there are $\binom{D-k}{i}$ ways to choose the right hand side of the rule, $1 \leq i \leq D - k$
- the double summation derives from the two points above
- the [binomial theorem](#) states that $\sum_{i=0}^n \binom{n}{i} x^i = (1 + x)^n$
- using the theorem for $x = 1$ and $x = 2$ leads to the final result (pay attention to the starting value of the summation)

Frequent Itemset Generation Strategies

- Reduce the **number of candidates** M
 - Complete search: $M = 2^D$
 - Use pruning techniques to reduce M
- 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

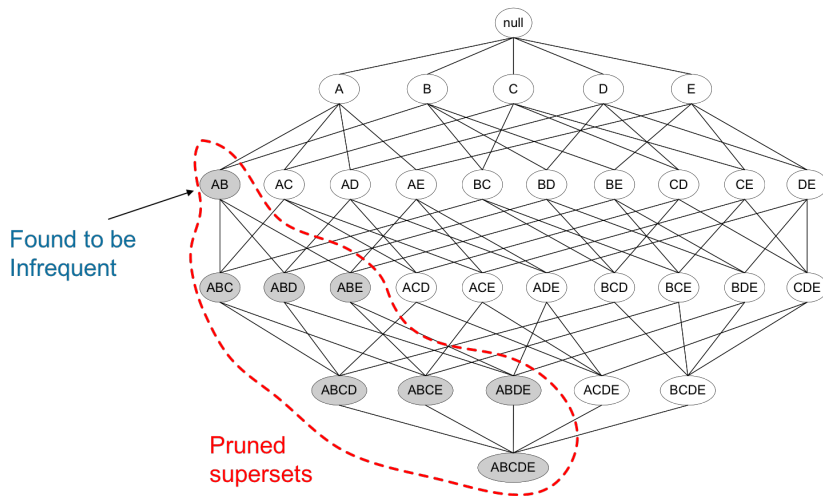
Reducing Number of Candidates

- **Apriori principle**
 - If an itemset is frequent, then all of its subsets must also be frequent
- It holds due to the following property of the support measure:

$$\forall X, Y : (X \subseteq Y) \Rightarrow \text{sup}(X) \geq \text{sup}(Y)$$

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

Pruning strategy



Apriori algorithm - Candidate generation

Definitions

C_k : candidate itemsets of size k

L_k : frequent itemsets of size k

$subset_k(c)$: set of the subsets of c with k elements

Candidate generation – Join Step

- Let L_k be represented as a table with k columns where each row is a frequent itemset
- Let the items in each row of L_k be in lexicographic order
- C_{k+1} is generated by a self join of L_k

```
INSERT INTO  $C_{k+1}$ 
SELECT p.item1, p.item2, ..., p.itemk, q.itemk
FROM  $L_k$  AS p,  $L_k$  AS q
WHERE p.item1 = q.item1 AND ... AND p.itemk-1 = q.itemk-1
      AND p.itemk < q.itemk;
```

Candidate generation – Prune Step

Each $(k + 1)$ -itemset which includes a k -itemset which is not in L_k is deleted from C_{k+1}

Algorithm 1: Prune Candidates

```
foreach  $c \in C_k$  do
  foreach  $s \in \text{subset}_{k-1}(c)$  do
    if  $s \notin L_{k-1}$  then
      delete  $c$  from  $C_k$ 
return  $C_k$ 
```

Frequent itemset generation

Algorithm 2: Apriori Algorithm

$L_1 \leftarrow$ frequent 1-itemsets

$k \leftarrow 1$

while $L_k \neq \emptyset$ **do**

$C_{k+1} \leftarrow$ candidates generated from L_k

foreach t transaction in database **do**

 increment candidate count in C_{k+1} for candidates found in t

$L_{k+1} \leftarrow \{c \in C_{k+1} : \text{sup}(c) \geq \text{minsup}\}$

$k \leftarrow k + 1$

return k, L_k

Pruning example – minsup=3

C_1

Item	Count
Beer	3
Bread	4
Coke	2
Diaper	4
Eggs	1
Milk	4

The support of {Coke}
and {Eggs} is below
minsup, therefore
they do not generate
 C_2 candidates



C_2

Item	Count
Beer,Bread	2
Beer,Diaper	3
Beer,Milk	2
Bread,Diaper	3
Bread,Milk	3
Diaper,Milk	3

No C_3 candidate will
include {Beer, Bread}
or {Beer, Milk}



C_3

Item	Count
Bread,Diaper,Milk	2

Number of itemsets to evaluate:

No pruning = $\binom{6}{1} + \binom{6}{2} + \binom{6}{3} = 41$

Support based pruning = 13

Origin of the name Apriori

- Level-wise computation
 - the level is the cardinality of the itemsets under evaluation
- The evaluations at level k use the **prior knowledge** acquired for the previous levels to reduce the search space

Factors Affecting Complexity I

- Choice of minimum support threshold
 - lowering support threshold results in a greater number of frequent itemsets
 - this may reduce pruning and increase the maximum length of frequent itemsets
 - the number of complete reads of the dataset is given by the maximum length of frequent itemsets plus one
- Dimensionality (number of items) of the data set
 - more space is needed to store support count of each item
 - if number of frequent items also increases, both computation and I/O costs may also increase

Factors Affecting Complexity II

- Size of database
 - since Apriori makes multiple passes, run time of algorithm may increase with number of transactions
- Average transaction width
 - transaction width increases with denser data sets
 - This may increase max length of frequent itemsets and traversals of data structures (number of subsets in a transaction increases with its width)

1	Introduction to Market Basket Analysis	2
2	Frequent Itemset Generation	10
3	Rule Generation	26
●	Pattern evaluation	32
4	Multidimensional association rules	52
5	Multilevel Association Rules	57

Confidence

From [Agrawal et al.(1993)Agrawal, Imieliński, and Swami]

- The confidence of a rule can be computed from the supports
⇒ for confidence based pruning of rules it is sufficient to know the supports of frequent itemsets

$$\text{conf}(A \Rightarrow C) = \frac{\text{sup}(A \Rightarrow C)}{\text{sup}(A)}$$

Rule Generation I

Give a frequent itemset L

- find all the non-empty subsets $f \in L$ such that the confidence of rule $f \Rightarrow (L - f)$ is not less than the minimum confidence (set by the experiment designer)
 - from $\{Beer, Diaper, Milk\}$ the possible rules are
 $Beer, Diaper \Rightarrow Milk$, $Beer \Rightarrow Diaper, Milk$,
 $Beer, Milk \Rightarrow Diaper$, $Milk \Rightarrow Beer, Diaper$,
 $Diaper, Milk \Rightarrow Beer$, $Diaper \Rightarrow Beer, Milk$
- if $|L| = k$ then there are $2^k - 2$ candidate rules
 - $L \Rightarrow \emptyset$ and $\emptyset \Rightarrow L$ can be ignored

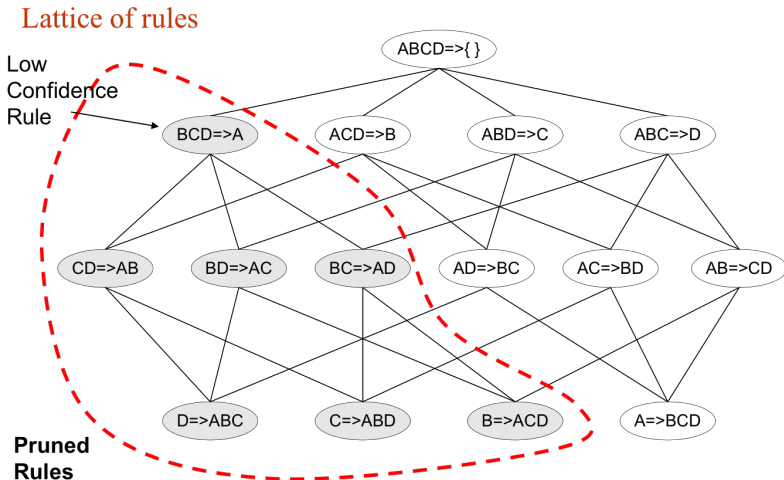
Rule Generation II

- How to efficiently generate rules from frequent itemsets?
 - In general, confidence does not have an anti-monotone property
 - $conf(ABC \rightarrow D)$ can be larger or smaller than $conf(AB \rightarrow D)$
 - But let us consider rules generated from the same itemset
 - e.g., $i = \{A, B, C, D\} \in L$:

$$conf(ABC \rightarrow D) \geqslant conf(AB \rightarrow CD) \geqslant conf(A \rightarrow BCD)$$

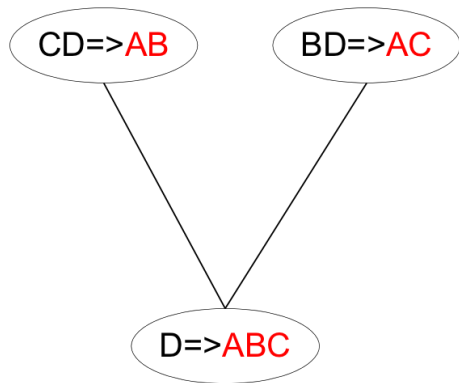
- Confidence of rules generated from the same itemset is anti-monotone w.r.t. the number of items on the RHS of the rule
 - i.e. it decreases when we move an item from the left hand to the right hand

Rule Pruning



Rule Generation in Apriori

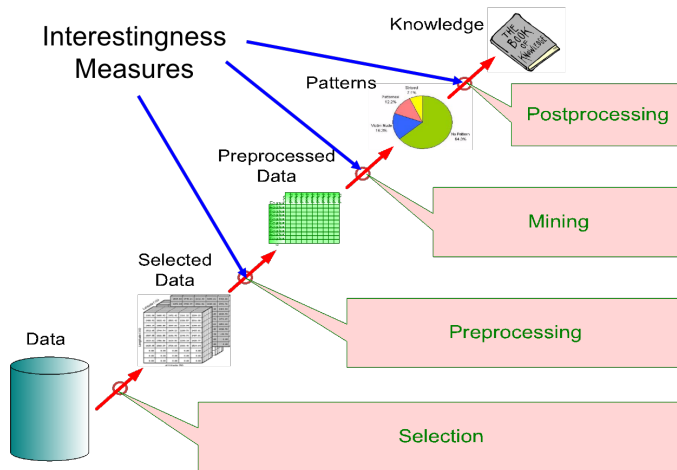
- Candidate rule is generated by merging two rules that share the same prefix in the rule consequent
- $\text{join}(\text{CD} \Rightarrow \text{AB}, \text{BD} \Rightarrow \text{AC})$ would produce the candidate rule $\text{D} \Rightarrow \text{ABC}$
- Prune rule $\text{D} \Rightarrow \text{ABC}$ if its subset $\text{AD} \Rightarrow \text{BC}$ does not have high confidence



Pattern Evaluation

- Association rule algorithms tend to produce too many rules
 - many of them are uninteresting or redundant
 - Redundant if $\{A, B, C\} \Rightarrow \{D\}$ and $\{A, B\} \Rightarrow \{D\}$ have same support and confidence
- Interestingness measures can be used to prune/rank the derived patterns
- In the original formulation of association rules, support and confidence are the only measures used

Application of Interestingness Measure



Computing Interestingness Measures

- Given a rule $A \Rightarrow C$, the information needed to compute rule interestingness can be obtained from a contingency table
- The elements of the contingency table are the basis for most of the interestingness measures

	C	\overline{C}	
A	f_{11}	f_{10}	f_{1+}
\overline{A}	f_{01}	f_{00}	f_{0+}
	f_{+1}	f_{+0}	

Drawback of Confidence

- $conf(Tea \Rightarrow Coffee) = \frac{sup(Tea, Coffee)}{sup(Tea)} = \frac{15}{20} = 0.75$
 - fairly high
- $\Pr(Coffee) = 0.9$ and $\Pr(Coffee | \overline{Tea}) = \frac{75}{80} = 0.9375$
 - despite the high confidence of $Tea \Rightarrow Coffee$, the absence of Tea increases the probability of $Coffee$
 - for this rule the confidence is misleading

	<i>Coffee</i>	\overline{Coffee}	
<i>Tea</i>	15	5	20
\overline{Tea}	75	5	80
	90	10	100

$Tea \Rightarrow Coffee$

Statistical Independence

- Population of 1000 students
 - 600 students know how to swim (S)
 - 700 students know how to bike (B)
 - 420 students know how to swim and bike (S,B)
- $\Pr(S \wedge B) = \frac{420}{1000} = 0.42$
- $\Pr(S) * P(B) = 0.6 * 0.7 = 0.42$
- $\Pr(S \wedge B) = P(S) * P(B) \Rightarrow$ Statistical independence
- $\Pr(S \wedge B) > P(S) * P(B) \Rightarrow$ Positively correlated
- $\Pr(S \wedge B) < P(S) * P(B) \Rightarrow$ Negatively correlated

Statistical-based Measures I

Measures that take into account the deviation from statistical independence

$$\textit{lift}(A \Rightarrow C) = \frac{\textit{conf}(A \Rightarrow C)}{\textit{sup}(C)} = \frac{\mathbf{Pr}(A, C)}{\mathbf{Pr}(A) \mathbf{Pr}(C)}$$

- **lift** evaluates to 1 for independence
- insensitive to rule direction
- it is the ratio of true cases w.r.t. independence

Statistical-based Measures II

Measures that take into account the deviation from statistical independence

$$\begin{aligned}leve(A \Rightarrow C) &= \mathbf{Pr}(A, C) - \mathbf{Pr}(A) * \mathbf{Pr}(C) \\ &= sup(A \cup C) - sup(A)sup(C)\end{aligned}$$

- **leverage** evaluates to 0 for independence
- insensitive to rule direction
- it is the number of additional cases w.r.t. independence

Statistical-based Measures III

Measures that take into account the deviation from statistical independence

$$\text{conv}(A \Rightarrow C) = \frac{1 - \text{sup}(C)}{1 - \text{conf}(A \Rightarrow C)} = \frac{\mathbf{Pr}(A) (1 - \mathbf{Pr}(C))}{\mathbf{Pr}(A) - \mathbf{Pr}(A, C)}$$

- **conviction** is infinite if the rule is always true
- sensitive to rule direction
- it is the ratio of the expected frequency that A occurs without C (that is to say, the frequency that the rule makes an incorrect prediction) if A and C were independent divided by the observed frequency of incorrect predictions
- also called **novelty**

Intuition about Measures

- higher support \Rightarrow rule applies to more records
- higher confidence \Rightarrow chance that the rule is true for some record is higher
- higher lift \Rightarrow chance that the rule is just a coincidence is lower
- higher conviction \Rightarrow the rule is violated less often than it would be if the antecedent and the consequent were independent

Example of page 35 – Interestingness measures

Tea \Rightarrow *Coffee*

$$\text{conf} = \frac{0.15}{0.20} = 0.75$$

in a 0 to 1 scale it is apparently high

$$\text{lift} = \frac{0.15}{0.90 * 0.20} = 0.83$$

is less than 1, therefore not interesting

$$\text{leve} = 0.15 - 0.90 * 0.20 = -0.03$$

is less than 0, therefore not interesting

$$\text{conv} = \frac{1-0.9}{1-0.75} = 0.4$$

is low, remembering that absolute truth gives

infinite

Comparison of measures

	C1	$\overline{C1}$	
A1	88	5	93
$\overline{A1}$	5	2	7
	93	7	100

Rule ($A1 \Rightarrow C1$)

$$conf = 0.88/0.93 = 0.946$$

$$lift = 0.88/(0.93 * 0.93) = 1.017$$

$$leve = 0.88 - 0.93 * 0.93 = 0.015$$

$$conv = (1 - 0.93)/(1 - 0.946) = 1.302$$

A high confidence rule can have small lift if both sides are very frequent

	C2	$\overline{C2}$	
A2	2	5	7
$\overline{A2}$	5	88	93
	7	93	100

Rule ($A2 \Rightarrow C2$)

$$conf = 0.02/0.07 = 0.286$$

$$lift = 0.02/(0.07 * 0.07) = 4.082$$

$$leve = 0.02 - 0.07 * 0.07 = 0.015$$

$$conv = (1 - 0.07)/(1 - 0.286) = 1.302$$

A low confidence rule can have high lift if both sides are very infrequent

Lift: Relative Association Strength

Practical meaning:

- how many times the presence of the consequent is inflated if the antecedent is present
- Measures **departure from statistical independence**

Key limitation:

- Inflated for rare items
- No notion of absolute business impact

Leverage: Absolute Incremental Effect

$$\text{Leverage}(A \Rightarrow B) = P(A \cap B) - P(A)P(B)$$

Practical meaning:

- 4% of all transactions contain the pair **because of the association**
- Measures **incremental volume**, not correlation

Business interpretation:

- In 1M transactions, this rule explains 40,000 extra joint purchases

Conviction: Rule Reliability

$$\text{Conviction}(A \Rightarrow B) = \frac{P(A)P(\neg B)}{P(A \cap \neg B)}$$

Practical meaning:

- How strongly the rule **avoids being wrong**
- Penalizes cases where A occurs but B does not

Visual Intuition: Confidence vs Conviction

Confidence: $P(B|A)$

- Looks only at **successes**
- Ignores how often the rule fails

Conviction:

$$\frac{\text{expected failures under independence}}{\text{observed failures}}$$

Intuition

- Confidence: “When I recommend, how often am I right?”
- Conviction: “How surprising is it when the recommendation fails?”

High confidence + low conviction \Rightarrow frequent false positives.

When Lift Is Misleading (Counterexample 1)

Assume:

$$P(A) = 0.01, \quad P(B) = 0.01, \quad P(A \cap B) = 0.005$$

Lift:

$$\text{Lift} = \frac{0.005}{0.0001} = 50$$

Problem:

- Extremely high lift
- Only 0.5% of transactions affected

Business reality:

- Rule is statistically strong but commercially irrelevant

When Lift Is Misleading (Counterexample 2)

Assume:

$$P(A) = 0.80, \quad P(B) = 0.70, \quad P(A \cap B) = 0.60$$

Lift:

$$\text{Lift} \approx 1.07$$

Observation:

- Lift suggests weak association
- Yet 60% of users satisfy the rule

Insight:

- Lift undervalues rules involving popular items

Mapping Metrics to Recommender KPIs

Metric	Measures	Related KPI
Lift	Relative predictiveness	CTR uplift
Leverage	Incremental volume	Revenue lift
Conviction	Error avoidance	Precision@K
Confidence	Success frequency	Hit Rate
Support	Coverage	Catalog utilization

Practical Guidance

- Use **lift** for discovery and filtering
- Use **leverage** for revenue prioritization
- Use **conviction** for recommendation reliability

Rule of Thumb

High lift \wedge high leverage \wedge high conviction \Rightarrow production-ready rule

Conclusion on measures

- There are lots of measures proposed in the literature, beyond the four presented here
- Confidence is usually the base tool
- Other measures can be used to test the results given by confidence and for additional filtering
- Use **lift** for discovery and filtering
- Use **leverage** for revenue prioritization
- Use **conviction** for recommendation reliability

Rule of Thumb

High lift \wedge high leverage \wedge high conviction \Rightarrow production-ready rule

1	Introduction to Market Basket Analysis	2
2	Frequent Itemset Generation	10
3	Rule Generation	26
4	Multidimensional association rules	52
•	Equivalence mono/multi	55
5	Multilevel Association Rules	57

Multidimensional association rules

Let's consider a dataset deriving from sensors measuring the concentration of air pollutants

<i>TID</i>	<i>CO</i>	<i>Tin_Oxide</i>	<i>Titanium</i>
1	high	medium	high
2	medium	low	medium
3	medium	high	low
4	low	medium	medium

- Look for rules such as *CO = high and Tin Oxide = high then Titanium = high* (support 0.25 and confidence 1)
- This can be used for example, if one of the sensor is not available, to guess its qualitative value given the others
- Useful for a qualitative analysis, in substitution of regression

Comparison mono- vs multi-dimensional

- **Mono-dimensional** (intra-attribute)
 - event: **transaction**
 - event description:
 - items A, B, and C are together in a transaction
- **Multi-dimensional** (inter-attribute)
 - event: **tuple**
 - event description:
 - attribute A has value a, attribute B has value b and attribute C has value c in a tuple

Equivalence mono/multi-dimensional

Multi-dimensional

Schema: (TID,CO,Tin_Oxide,Titanium)

- 1, high, medium, high
- 2, medium, low, medium



- 1, {CO/high, Tin_Oxide/medium, Titanium/high}
- 2, {CO/medium, Tin_Oxide/low, Titanium/medium}

Schema: (TID,a?,b?,c?,d?)

- 1, yes, yes, no, no
- 2, yes, no, yes, no



- 1, {a, b}
- 2, {a, c}

Quantitative attributes

<i>TID</i>	<i>CO</i>	<i>Tin Oxide</i>	<i>Titanium</i>
1	2.6	1360	1046
2	2.0	1292	955
3	2.2	1402	939
4	1.6	1376	948

- Too many distinct values for the multi/mono transformation
- Most software packages for association rules discovery do not deal with quantitative attributes

⇒ discretization

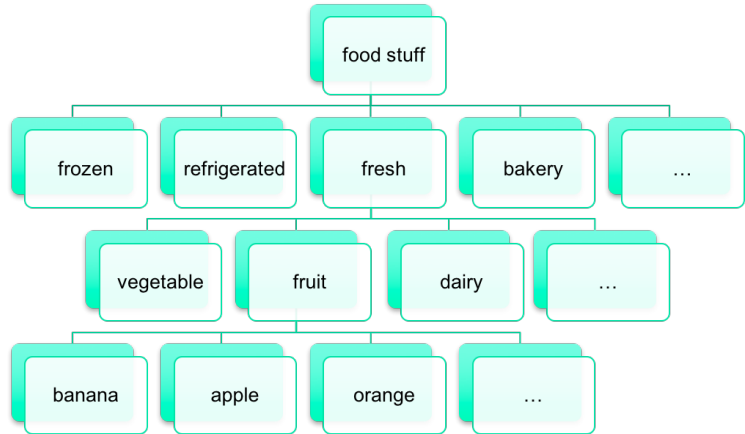
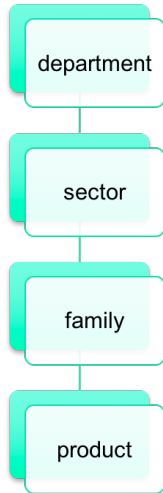
- possibly **equifrequency** or with **mono-dimensional clustering**, for optimal covering of the original value domains
- discretisation leads to a dataset like that of page 53
- Association rules can involve items at different qualitative levels

1	Introduction to Market Basket Analysis	2
2	Frequent Itemset Generation	10
3	Rule Generation	26
4	Multidimensional association rules	52
5	Multilevel Association Rules	57
●	Support and Confidence in Multilevel AR	60

Multilevel Association Rules

- A real MBA database can include tens of thousands of distinct items
- Frequently it is necessary to find a tradeoff between general and detailed reasoning
 - choose the right level of abstraction
- A common **background knowledge** is the organization of the items into a hierarchy of concepts
 - it can be easily coded in the transactions
 - it can help the choice of the right level of abstraction

Concept Hierarchy



Support in Multilevel AR

- From specialized to general

$(\text{apple} \Rightarrow \text{milk}) \rightarrow (\text{fruit} \Rightarrow \text{dairy})$

- the support of rules increases, in general
- new rules can become interesting

- From general to specialized

$(\text{fruit} \Rightarrow \text{dairy}) \rightarrow (\text{apple} \Rightarrow \text{milk})$

- the support of rules decreases, in general
- the support of rules can go under the threshold

Confidence in Multilevel AR

- A level change can influence the confidence in any direction
- If the specialized rule has (approximately) the same confidence as the general one, then it is **redundant**

Example

Low-fat milk is a subclass of milk

- 1000 transactions, 80 with milk and bread, 114 with milk, 20 with low-fat milk and bread, 28 with low-fat milk
 - a) $\text{milk} \Rightarrow \text{bread}$ (support = 8%, confidence = 70%)
 - b) $\text{low-fat milk} \Rightarrow \text{bread}$ (support = 2%, confidence = 71%)
 - rule b) has almost the same confidence as rule a)
 - rule b) is a descendant of rule a)
- \Rightarrow rule b) is **redundant**

Mining Multilevel Association Rules

- Look for frequent itemsets at each level of abstraction, top down
 - Each level requires a new run of the rule discovery algorithm
- Decrease the support threshold in lower levels

Bibliography I

- Rakesh Agrawal, Tomasz Imieliński, and Arun Swami.
Mining association rules between sets of items in large databases.
[SIGMOD Rec.](#), 22(2):207–216, June 1993.
ISSN 0163-5808.
doi: 10.1145/170036.170072.
URL <http://doi.acm.org/10.1145/170036.170072>.