Data Mining:

2. Assoziationsanalyse

D) Usual Data

Continuous and Categorial Attributes

How to apply association analysis formulation to non-asymmetric / non-binary data ?

Session Id	Country	Session Length (sec)	Number of Web Pages viewed	Gender	Browser Type	Buy
1	USA	982	8	Male	IE	No
2	China	811	10	Female	Netscape	No
3	USA	2125	45	Female	Mozilla	Yes
4	Germany	596	4	Male	IE	Yes
5	Australia	123	9	Male	Mozilla	No
…	…	…	…	…	…	…

Example of Association Rule:

$$\{\text{Number of Pages} \in [5,10) \wedge (\text{Browser=Mozilla})\} \rightarrow \{\text{Buy = No}\}$$

Handling Categorial Attributes

- Transform categorial attribute into asymmetric binary variables

- Introduce a new “item” for each distinct attribute-value pair
  - Example: replace attribute ‘Browser Type’ with
    - Browser Type = Internet Explorer
    - Browser Type = Mozilla

Handling Categorial Attributes

- Potential Issues
  - What if attribute has many possible values
    - Example: attribute country has more than 200 possible values
    - Many of the attribute values may have very low support
      - Potential solution: Aggregate the low-support attribute values
  - What if distribution of attribute values is highly skewed
    - Example: 95% of the visitors have Buy = No
    - Most of the items will be associated with (Buy=No) item
      - Potential solution: drop the highly frequent items
      - Or apply multi-support technique
  - Avoid generating sets with >1 item of same attribute

Handling Continuous Attributes

- Different kinds of rules:
  - $Age \in [21,35) \wedge Salary \in [70k,120k) \rightarrow Buy$
  - $Salary \in [70k,120k) \wedge Buy \rightarrow Age: \mu=28, \sigma=4$

μ mean, σ standard deviation

- Different methods:
  - Discretization-based
  - Statistics-based
  - Non-discretization based
    - min-Apriori

Discretization Issues

- Use discretization, e.g. unsupervised methods like equal-width/-depth binning or clustering
- Size of the discretized intervals affect support & confidence

  {Refund = No, (Income = $51,250)} $\rightarrow$ {Cheat = No}

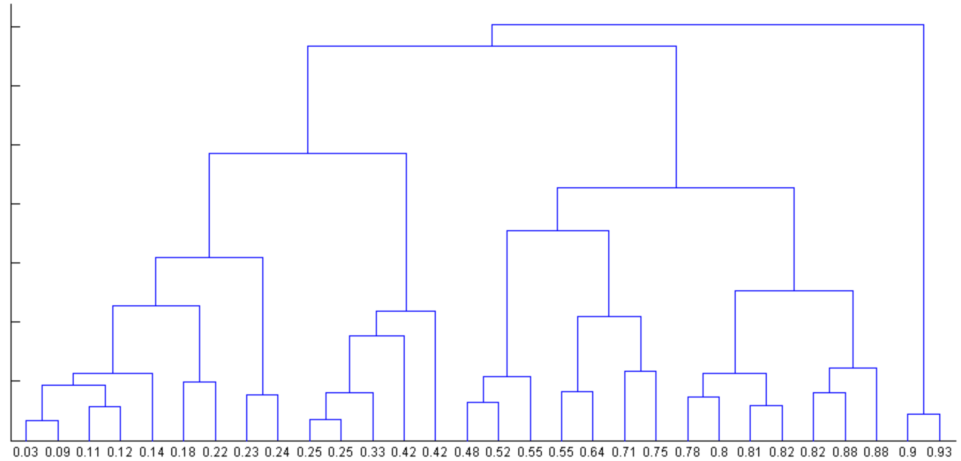
  {Refund = No, (60K $\leq$ Income $\leq$ 80K)} $\rightarrow$ {Cheat = No}

  {Refund = No, (0K $\leq$ Income $\leq$ 1B)} $\rightarrow$ {Cheat = No}

  – If intervals too small
    - may not have enough support
  – If intervals too large
    - may not have enough confidence
- Potential solution?: use all possible intervals

Discretization Issues

- Execution time
  - If intervals contain n values, there are on average $O(n^2)$ possible ranges



- Too many rules, many redundant rules

{Refund = No, (Income = $51,250)} → {Cheat = No}

{Refund = No, (51K ≤ Income ≤ 52K)} → {Cheat = No}

{Refund = No, (50K ≤ Income ≤ 60K)} → {Cheat = No}

A Discretization Approach [see Srikant & Agrawal]

- Preprocess the data
  - Discretize attribute using equi-depth partitioning
    - Use *partial completeness measure* to determine how much information is lost and thus to determine the number of partitions
    - Merge adjacent intervals as long as support is less than max-support
- Apply existing association rule mining algorithms
- Determine interesting rules in the output

Statistics-based Methods

- Example: Browser=Mozilla $\wedge$ Buy=Yes $\rightarrow$ Age: μ=23
- Rule consequent consists of a continuous variable, characterized by their statistics
  - mean, median, standard deviation, etc.
- Approach:
  - Withhold the target variable from the rest of the data
  - Apply existing frequent itemset generation on the rest of the data
  - For each frequent itemset, compute the descriptive statistics for the corresponding target variable
  - A frequent itemset becomes a rule by introducing the target variable as rule consequent
  - Apply statistical tests to determine interestingness of the rule:

    The statistics for the segment of population covered by the rule vs the statistics for the segment of population not covered by the rule must differ significantly.

Non-discretization-based: Min-Apriori [Han et al]

- Consider text mining, in particular finding word associations in text documents
- Input: **Document-word matrix *D***

In this example, W1 and W2 tend to appear together in a document.

TID	W1	W2	W3	W4	W5
D1	2	2	0	0	1
D2	0	0	1	2	2
D3	2	3	0	0	0
D4	0	0	1	0	1
D5	1	1	1	0	2

- Data contains only continuous attributes of the same “type”
  - Here, frequency of words in a document
- Potential solutions ?
  - Convert into 0/1 matrix and then apply existing algorithms
    - loses word frequency information and depends on 0/1 threshold
  - Discretize, e.g. ‘data’∈[21,24] and ‘mining’ ∈[32,36]
    - users don’t want associations between word frequency intervals, but between words

Min-Apriori

- How to determine the support of a word ?
  - If we simply sum up its frequencies, support count will be greater than total number of documents!
    - Normalize the word vectors – e.g., by dividing each word frequency by the sum of word frequencies across all documents
    - Each word has a support equals to 1.0

TID	W1	W2	W3	W4	W5
D1	2	2	0	0	1
D2	0	0	1	2	2
D3	2	3	0	0	0
D4	0	0	1	0	1
D5	1	1	1	0	2

Normalize

TID	W1	W2	W3	W4	W5
D1	0,40	0,33	0,00	0,00	0,17
D2	0,00	0,00	0,33	1,00	0,33
D3	0,40	0,50	0,00	0,00	0,00
D4	0,00	0,00	0,33	0,00	0,17
D5	0,20	0,17	0,33	0,00	0,33

Min-Apriori

- How to determine the support of a word association *C* ?
- New definition of support:

$$\sup(C) = \sum_{i \in T} \min_{j \in C} D(i, j)$$

D:

TID	W1	W2	W3	W4	W5
D1	0,40	0,33	0,00	0,00	0,17
D2	0,00	0,00	0,33	1,00	0,33
D3	0,40	0,50	0,00	0,00	0,00
D4	0,00	0,00	0,33	0,00	0,17
D5	0,20	0,17	0,33	0,00	0,33

sup(W1,W2,W3)
= 0 + 0 + 0 + 0 + 0,17
= 0.17
sup (W1,W2)
= 0,33+0+0,4+0+0,17
= 0,9
sup(W1) = 1

- Summarizing the averages would be useless (always =1)

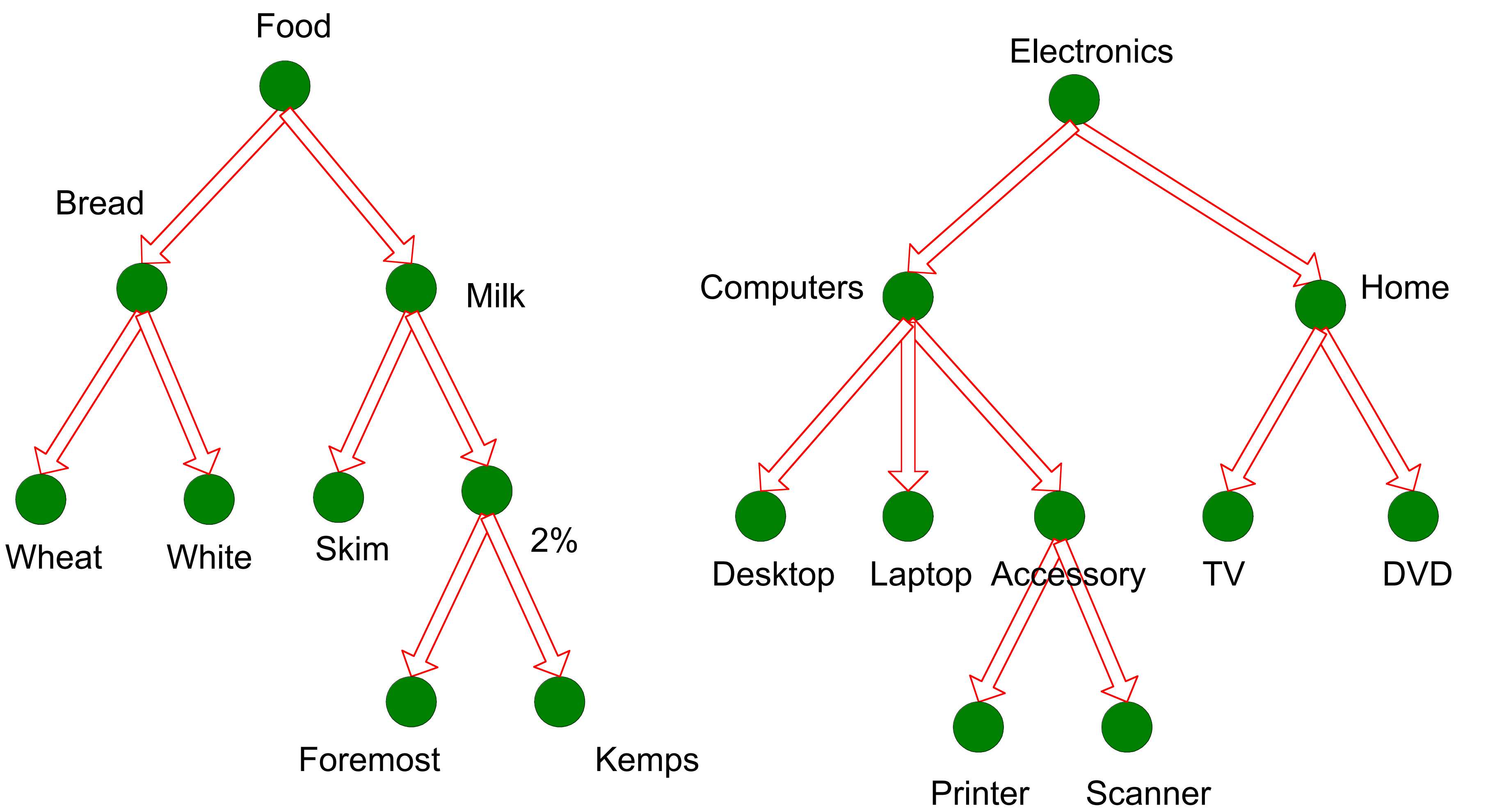
Min-Apriori

- Support has the usual anti-monotone property !

- The standard Apriori algorithm can be applied using the new support definition.

Multi-level Association Rules

based on a concept hierarchy like



Multi-level Association Rules

- Why should we incorporate concept hierarchy?

  – Rules at lower levels may not have enough support to appear in any frequent itemsets

  – Rules at lower levels of the hierarchy are overly specific
    - e.g., skim milk → white bread, 2% milk → wheat bread,
      skim milk → wheat bread, etc.

    are indicating an association between milk and bread

Multi-level Association Rules

- How do support and confidence vary as we traverse the concept hierarchy?
  - If X is the parent item for both X1 and X2, then $\sigma(X) \leq \sigma(X1) + \sigma(X2)$
  - If $\sigma(X1 \cup Y1) \geq$ minsup,
    and X is parent of X1, Y is parent of Y1
    then $\sigma(X \cup Y1) \geq$ minsup, $\sigma(X1 \cup Y) \geq$ minsup
    $\sigma(X \cup Y) \geq$ minsup
  - If conf$(X1 \Rightarrow Y1) \geq$ minconf,
    then conf$(X1 \Rightarrow Y) \geq$ minconf

Multi-level Association Rules

- Approach 1:
  - Extend current association rule formulation by augmenting each transaction with higher level items

Original Transaction:	{skim milk, wheat bread}
Augmented Transaction:	{skim milk, wheat bread, milk, bread, food}

- Issues:
  - Items that reside at higher levels have much higher support counts
    - if support threshold is low, too many frequent patterns involving items from the higher levels
    - if support threshold is too high, only high-level patterns are generated
  - Increased dimensionality of the data
  - Redundant rules (but redundant itemsets can be easily discovered)

Multi-level Association Rules

- Approach 2:
  - Generate frequent patterns at highest level first
  - Then, generate frequent patterns at the next highest level, and so on

- Issues:
  - I/O requirements will increase dramatically because we need to perform more passes over the data
  - May miss some potentially interesting cross-level association patterns