LAST WEEK

- Euclidean distance
- Minkowski distance
- Simple Matching and Jaccard Coefficient
- Cosine similarity
- Correlation measures

0, 1

$$\frac{f_{00}+f_{11}}{\# f}$$

$$\frac{f_{11}}{f_{01}+f_{10}+f_{11}}$$

$x_1=(1,2)$ $\quad x_2=(4,3)$

3 $\quad$ 1

$$E=\sqrt{(4-1)^2+(3-2)^2}=\sqrt{10}$$

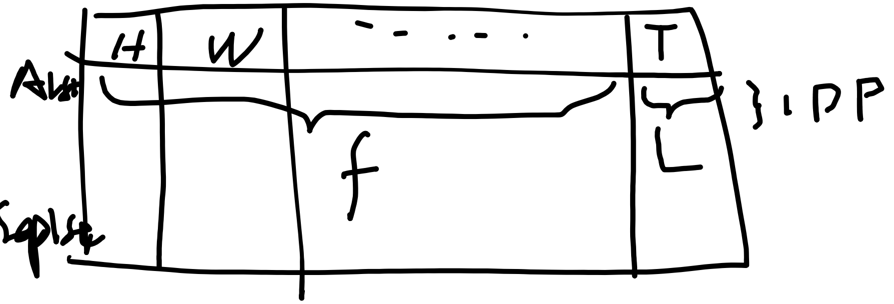
$$=\left((4-1)^2+(3-2)^2\right)^{\frac{1}{2}}$$

$$M=\left(\qquad\right)^{\frac{1}{r}}$$

① $r=1$ $\quad L_1=M_{r=1}=3+1=4$

② $r=2$ $\quad L_2=E$

③ $r=\infty$, $\quad L_{max}=L_{\infty}=\max(3,1)=3$



ASSOCIATION RULE MINING

BEIYU LIN

ASSOCIATION RULE MINING

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

Market transactions

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

Example of Association Rules

{Beer} $\rightarrow$ {Eggs},
{Milk, Bread} $\rightarrow$ {Diaper, Beer},

REVIEW: SET AND SUBSET

- {a, b, c, d} ⇔ a set (there are one or more than one items)
- Subset ⇔ possible combinations of the items in a set
  - Possible sets:
  - {a}, {b}, {c}, {d}, {a, b}, {a, c}, {a, d}, {b, c}, {b, d}, {c, d} {a, b, c}, {a, b, d}, {a, c, d}, {b, c, d}, {a, b, c, d}, {}
  - What is the total number of the subset:

6

$2^4 = 2^{\#item} = C_4^0 + C_4^1 + C_4^2 + C_4^3 + C_4^4$

Non-Empty : $2^4 - 1 = 2^{\#of\ item} - 1$

subset:

① Empty $\{\} = C_4^0$

② 1 item $\{a\}, \{b\}, \{c\}, \{d\} = C_4^1$

③ 2 item $C_4^2 = \frac{4!}{2!\,2!} = \frac{4\times 3\times \cancel{2!}}{\cancel{2!}\times 2!} = 6$

③ 3 item $C_4^3 = 4$

④ 4 item $C_4^4 = 1$

ASSOCIATION RULE MINING

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

- **Itemset (set / subset)** : order ×
  - A collection of one or more items
    - Example: {Milk, Bread, Diaper}
  - k-itemset
    - An itemset that contains k items
- **Support count (σ)**
  - Frequency of occurrence of an itemset
    3-itemset
  - E.g. σ({Milk, Bread,Diaper}) = 2

\# of itemset as a subset in dataset

Support $= \dfrac{\sigma}{\text{size of Dataset}} = \dfrac{2}{5}$

Fraction of transactions that contain an itemset

E.g. s({Milk, Bread, Diaper}) = 2/5

Frequent Itemset

An itemset whose support is greater than or equal to a *minsup* threshold

= min σ

$minsup = 1 \le \sigma(\{M, B, D\}) = 2$

DEFINITION: ASSOCIATION RULE

- Association Rule : conf of $X \to Y = \frac{\# X \cup Y}{\# X}$
  - An implication expression of the form X → Y, where X and Y are itemsets k-itemset set | subset
  - Example:
    $\underbrace{\{\text{Milk, Diaper}\}}_{X\text{ 2-itemset}} \to \underbrace{\{\text{Beer}\}}_{Y\text{ 1-itemset}}$
- Rule Evaluation Metrics
  - Support (s) = {M, D, B}
    - Fraction of transactions that contain both X and Y
  - Confidence (c)
    - Measures how often items in Y {B} appear in transactions that contain X = {M, D}

$c = \frac{\# \{M, D, B\}}{\# \{M, D\}} = \frac{X \cup Y}{X} = \frac{2}{3}$

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

$S\{M, D, B\} = \frac{2}{5}$

Example:

$$\{\text{Milk}, \text{Diaper}\} \Rightarrow \{\text{Beer}\}$$

$$s = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$ =# of itemset / total # transac

$$c = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{\sigma(\text{Milk}, \text{Diaper})} = \frac{2}{3} = 0.67$$ = # of itemset of X and Y / # of

ASSOCIATION RULE MINING TASK

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

$X \to Y$

$\{B, M\} \to \{D\}$

- 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 expensive / prohibitive!

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

6 item

$X \to Y$

$X = 1, 2, 3, \dots, 6$

{a,b,c,d} ➔ # of subset:

X: 1 item $C_6^1 * [C_6^1 + \dots + C_6^6]$

2 item $C_6^2 *$

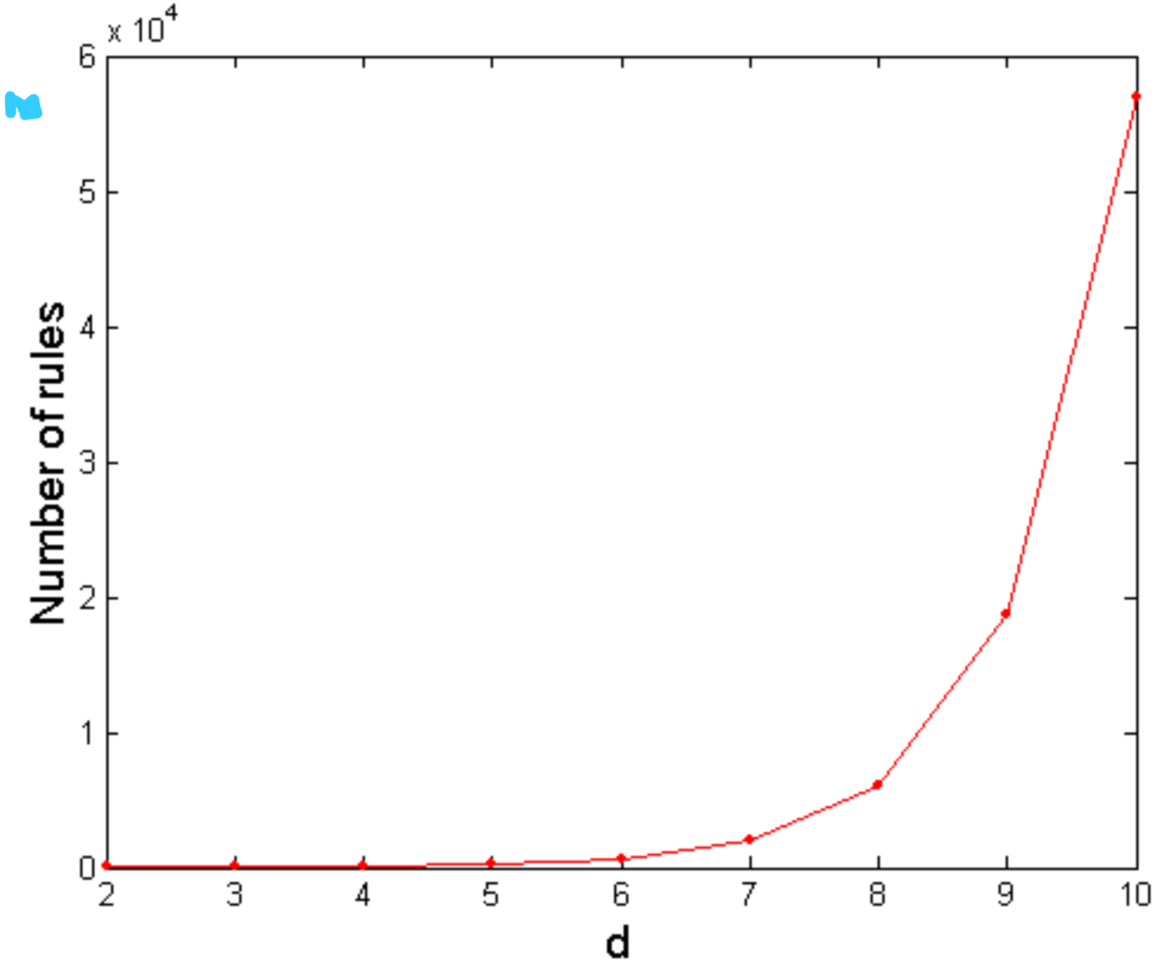
COMPUTATIONAL COMPLEXITY

- Given d unique items:
  - Total number of itemsets = 2^d d = # of item
  - Total number of possible association rules:

$$R = \sum_{k=1}^{d-1}\left[\binom{d}{k} \times \sum_{j=1}^{d-k}\binom{d-k}{j}\right]$$
$$= 3^d - 2^{d+1} + 1$$

X → Y

If d=6, R = 602 rules



MINING ASSOCIATION RULES

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

Example of Rules:

$X \rightarrow Y$ $\quad s \quad c$

{Milk,Diaper} → {Beer} (s=0.4, c=0.67)
{Milk,Beer} → {Diaper} (s=0.4, c=1.0)
{Diaper,Beer} → {Milk} (s=0.4, c=0.67)
{Beer} → {Milk,Diaper} (s=0.4, c=0.67)
{Diaper} → {Milk,Beer} (s=0.4, c=0.5)
{Milk} → {Diaper,Beer} (s=0.4, c=0.5)

{a, b, c}
={b, c, a}
={c, a, b}
={c,b,a}

$s = \frac{X \cup Y = \#(M,D,B)}{5} = \frac{2}{5} = .4 \qquad c = \frac{\#(M,D,B)}{\#(M)} = \frac{X \cup Y}{X} = \frac{2}{4} = .5$

Observations:

- All the above rules are binary partitions of the same itemset: {Milk, Diaper, Beer}
- Rules originating from the same itemset have identical support but can have different confidence
- Thus, we may decouple the support and confidence requirements

MINING ASSOCIATION RULES

- **Two-step approach:**
  1. Frequent Itemset Generation
     - Generate all itemsets whose support $\geq$ 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**

FREQUENT ITEMSET GENERATION

① all poss itemsets: $2^d = 2^5 = 32$

0

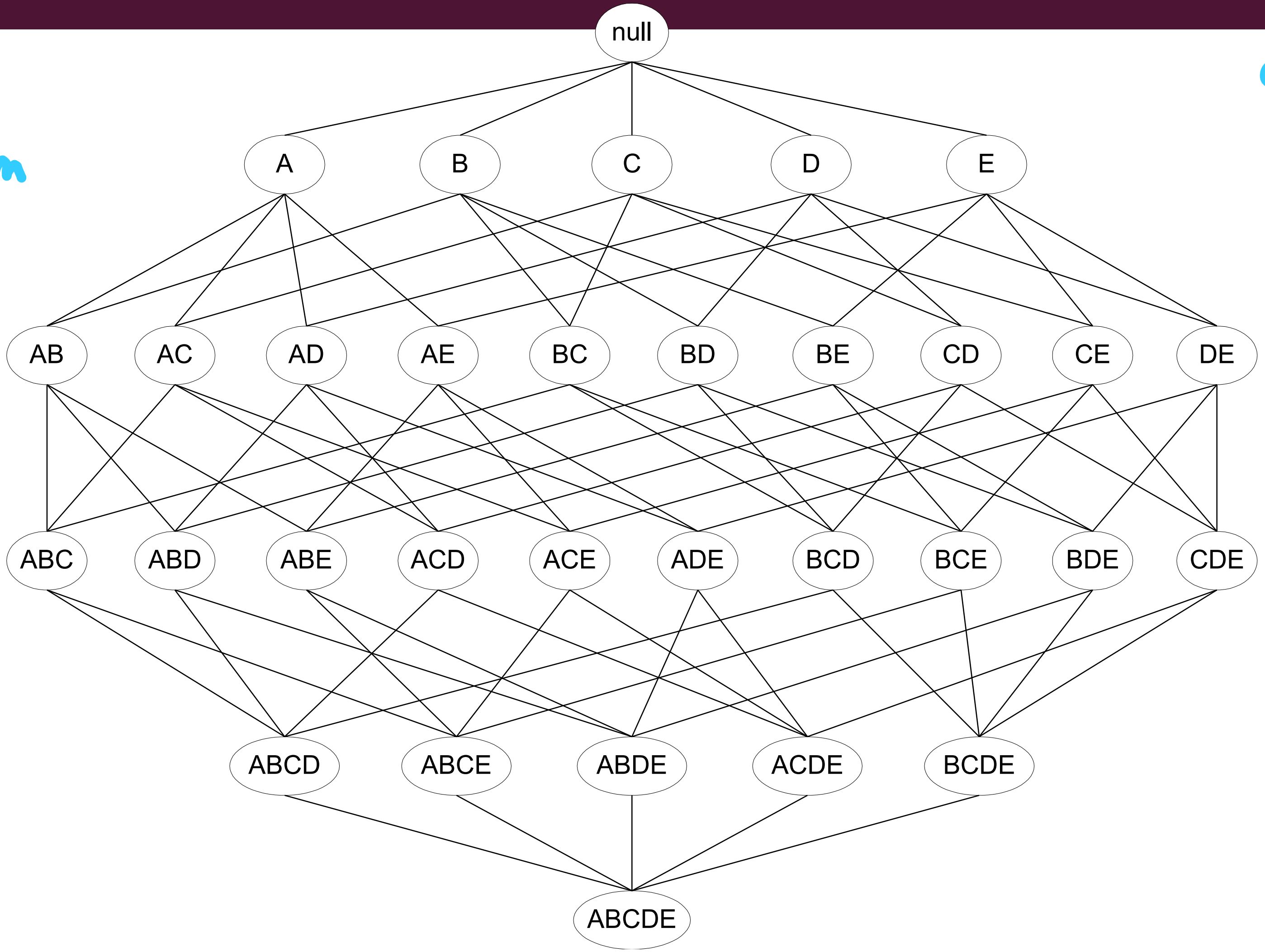
1 item

2

3

4

5



Given d items, there are 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

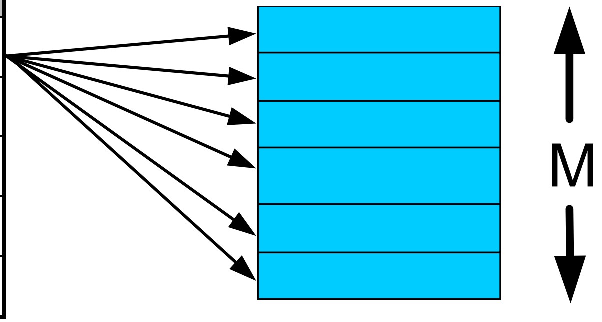
Transactions

N

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

w

List of Candidates



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

FREQUENT ITEMSET GENERATION STRATEGIES

= # of itemsets $= 2^d$

- 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
  - Used by DHP and vertical-based mining algorithms
- 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

Transactions

N

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

w

List of Candidates