Regole Associative
Concetti di base

Introduzione

- ***Data mining*** è il processo di estrazione di conoscenza e di informazioni utili da grandi quantità di dati memorizzati nei database

- ***Regole associative****:* descrivono relazioni di associazione rilevanti tra gli attributi del dataset.

Mining di regole associative

- Dato un insieme di transazioni, trovare le regole che segnalano la presenza di un elemento sulla base della presenza di altri elementi nella transazione

Transazioni del carrello della spesa

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

Alcuni esempi di regole:

{ Diaper } → { Beer },
{ Milk, Bread } → { Eggs, Coke },
{ Beer, Bread } → { Milk },

L'implicazione indica co-occorrenza e non causalità!

ATTENZIONE: gli item sono modellati da variabili binarie asimme-triche. Un item è presente oppure è assente nella transazione; la sua presenza è considerata un evento più importante della sua assenza

Applicazioni: collezione di documenti

- **Dataset: Insieme di documenti**
- **Un documento è trattato come un insieme di parole chiave**
  - doc1: Student, Teach, School
  - doc2: Student, School
  - doc3: Teach, School, City, Game
  - doc4: Baseball, Basketball
  - doc5: Basketball, Player, Spectator
  - doc6: Baseball, Coach, Game, Team
  - doc7: Basketball, Team, City, Game

Altri domini applicativi

- Dati Relationali

  { x.diagnosis=Heart, x.sex=Male } → { x.age>50 } [0.4 , 0.7]

- Dati Object-Oriented

  { s.hobbies = { sport , art } } → { s.age()=Young } [0.5 , 0.8]

Itemset Frequenti

- **Itemset**
  - ✓ **Una collezione di uno o più elementi**
    - Esempio: {Milk, Bread, Diaper}
  - ✓ **k-itemset**
    - Un itemset che contiene k-elementi
- **Support count (σ)**
  - ✓ Numero di istanze dell’itemset nell’insieme di transazioni
  - – Esempio; σ({Milk, Bread,Diaper}) = 2
- **Supporto**
  - ✓ Frazione delle transazioni che contiene l’itemset : $s(X) = \sigma(X)/N$
  - – Esempio: s({Milk, Bread, Diaper})= 2/5
- **Frequent Itemset**
  - ✓ Un itemset il cui supporto è maggiore o uguale a una soglia minsup

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

Association Rule

- Regole associative
  - Una implicazione ha la forma **X → Y**, dove X e Y sono itemset
  - Esempio:
    {Milk, Diaper} → {Beer}

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

- Metriche per la valutazione
  - **Supporto (s)**
    - Frazione delle transazioni che includono X e Y :
    - $\boldsymbol{s}(\boldsymbol{X} \cup \boldsymbol{Y}) = \boldsymbol{\sigma}(\boldsymbol{X} \cup \boldsymbol{Y})/\boldsymbol{N}$
  - **Confidenza (c)**
    - Misura quante volte gli elementi in Y appaiono in transazioni che contengono X
    - $\mathbf{c}(\boldsymbol{X} \cup \boldsymbol{Y}) = \boldsymbol{\sigma}(\boldsymbol{X} \cup \boldsymbol{Y})/\boldsymbol{\sigma}(\boldsymbol{X})$

Esempio:

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

$$s = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{\sigma(\text{Milk}, \text{Diaper})} = \frac{2}{3} = 0.67$$

Formulazione del problema

- $I = \{ i_1 , i_2 , \ldots , i_d \}$ è un insieme di elementi (items)
- D è un insieme di transazioni T
- Ciascuna transazione T è un insieme di items (sottoinsieme di I)
- TID è un identificatore di una transazione T
- Problema:

  Generare tutte le regole associative che hanno un supporto e una confidenza superiori ai valori di soglia specificati dall'utente (*minsup* e *minconf*)

Scoperta di regole associative

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

Esempio di regole

{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)

Osservazione:

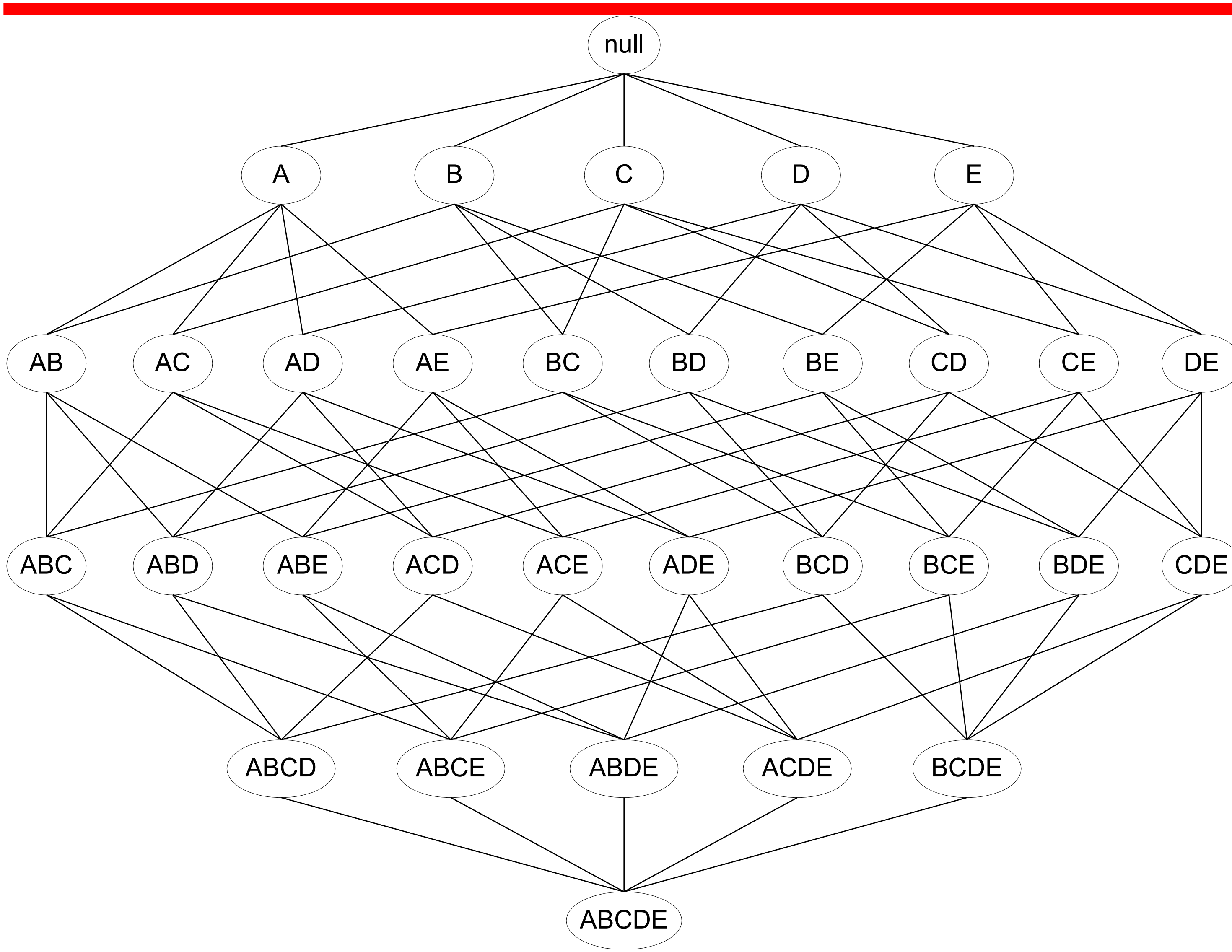
- Tutte le regole sono partizioni binarie dello stesso itemset:
  {Milk, Diaper, Beer}
- Le regole basate sullo stesso itemset hanno sempre il medesimo *supporto* ma possono avere valori di *confidenza* diversi

Scoperta di regole associative

- Approccio in due fasi:
  1. Generazione degli Itemset frequenti
     - Generare tutti gli itemset con supporto ≥ minsup
  2. Generazione delle regole
     - Per ogni itemset frequente massimale **L**, e per ogni sottoinsieme non vuoto ***f*** *di* ***L***, genera una regola **f → (L-f)** se la sua confidenza è maggiore della confidenza minima.

- La generazione degli itemset frequenti è comunque un problema computazionalmente complesso

Generazione di Itemset Frequenti



Dati d elmenti ci sono $\sum_{k=0}^{d}\binom{d}{k} = 2^d$ possibili item-set candidati

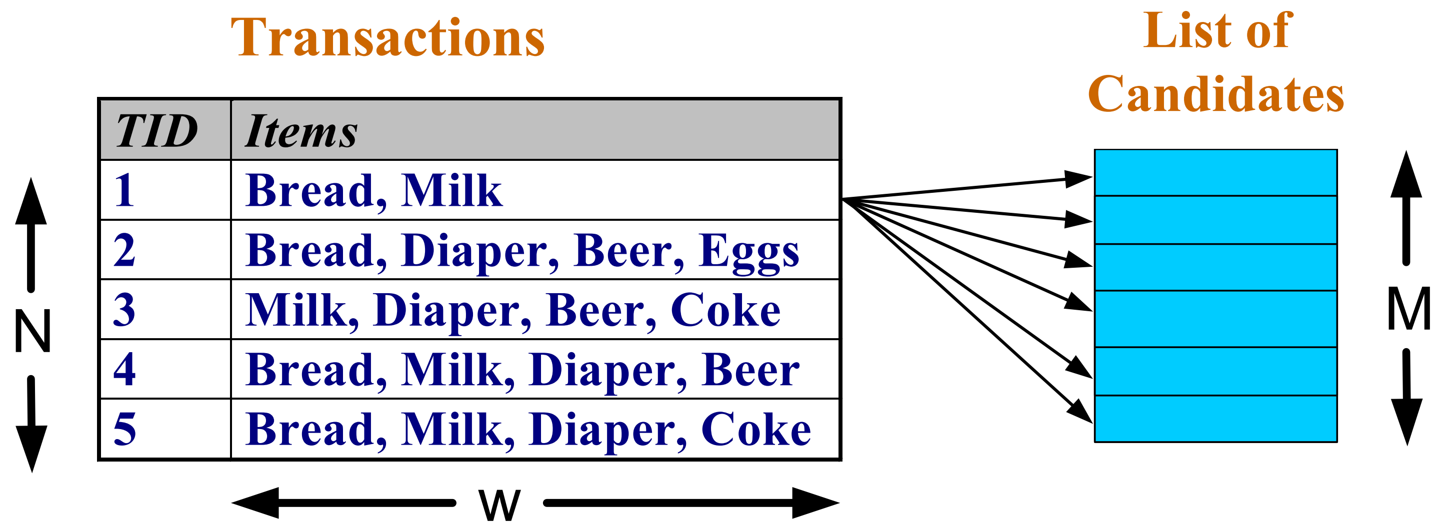
Generazione di Itemset Frequenti

- Approccio di forza bruta (naïve):
  - ✓ Elenca tutte le possibili regole associative
  - ✓ Per ogni regola calcola il supporto e la confidenza
  - ✓ Elimina le regole che non superano le soglie per *minsup* e *minconf*

Computazionalmente proibitivo!!

Generazione di Itemset Frequenti

- Approccio di forza bruta:
  - ✓ Ogni itemset nel reticolo con cardinalità diversa da 0, 1, d e d-1 è candidato a essere frequente
  - ✓ Calcola il supporto dei candidati scorrendo il database



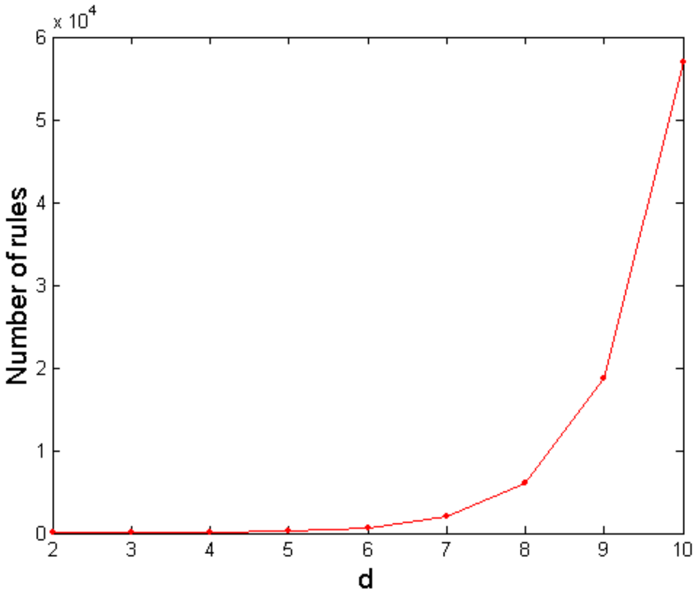
TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

- ✓ Confronta ogni transazione con ogni candidato

- Complessità ~ O(N M w) => Esponenziale poiché **$M = 2^d$** !!!

Complessità Computazionale

- Dati d distinti item:
  - ✓ Numero totale di itemset = 2^d (#candidati = $2^d - 2(d+1)$)
  - ✓ Il numero totale di regole associative *R* è:



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

Esempio

- $\boldsymbol{d = 4 \Rightarrow R = 3^4 - 2^5 + 1 = 50}$
- $\boldsymbol{d = 6 \Rightarrow R = 3^6 - 2^7 + 1 = 602}$

I limiti delle due sommatorie potrebbero essere ottimizzati (e.g. sostituendo **d** con **min{d,w}**, dove w è la lunghezza massima di una transazione)

Strategie per generare Itemset Frequenti

- Ridurre il numero di candidati (M)
  - Ricerca completa: $M=2^d$
  - Usa tecniche di pruning per ridurre M
- Ridurre il numero di transazioni (N)
  - Ridurre la dimensione di N all'aumentare della dimensione del set di elementi
- Ridurre il numero delle confronti (NM)
  - Utilizzare strutture dati efficienti per memorizzare i candidati o le transazioni
    Non c'è bisogno di fare matching tra tutti i candidati e tutte le transazione

Riduzione del numero di candidati

- **Principio Apriori :**
  - ✓ **Se un itemset è frequente, allora lo sono anche tutti i suoi sottoinsiemi.**

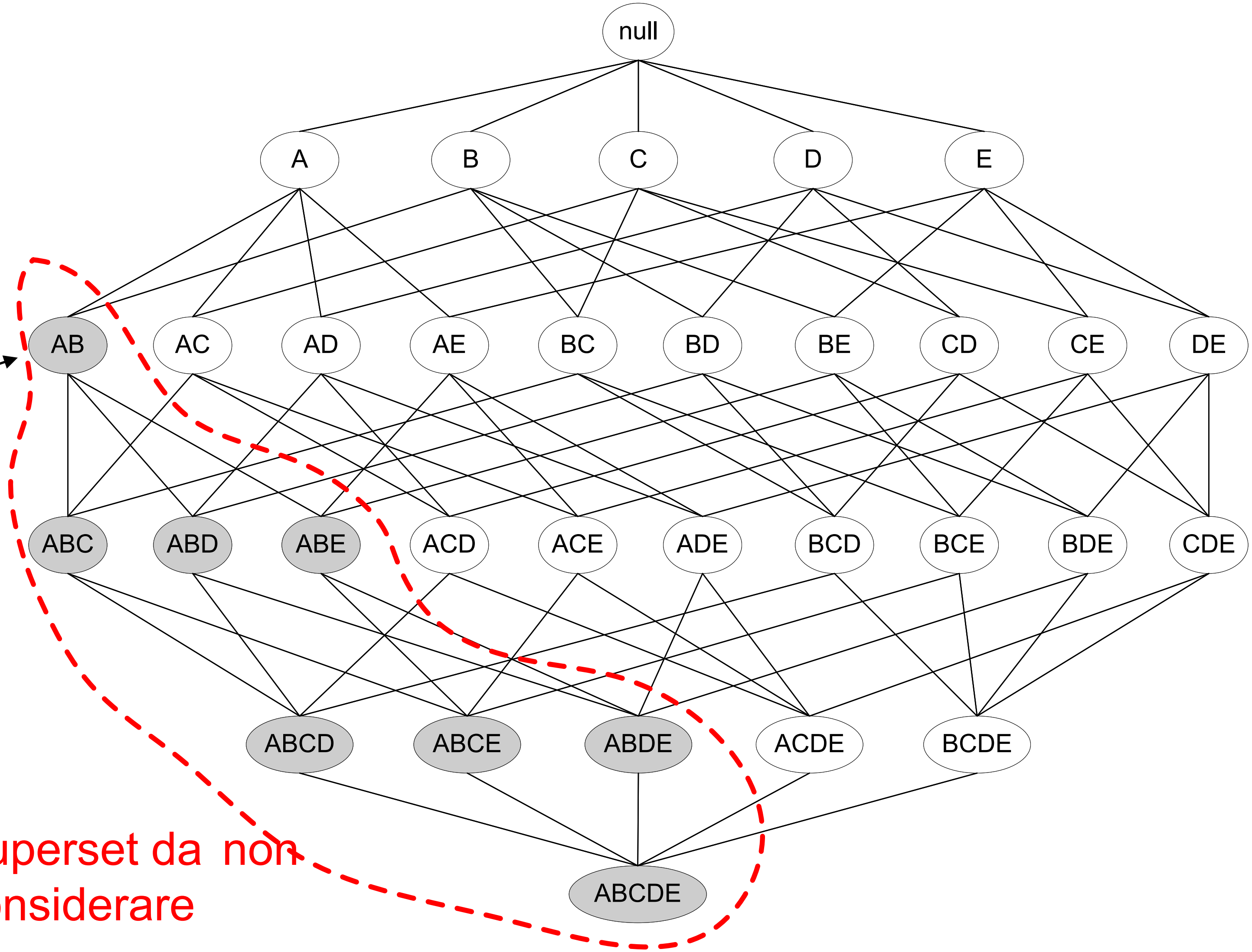
- Il principio Apriori è dovuto alla seguente proprietà del supporto :

$$\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$$

  - ✓ Aumentando il numero di items il supporto diminuisce
  - ✓ Questa è nota come proprietà anti-monotona del supporto

Riduzione del numero di candidati

Identificato come non frequente



Superset da non considerare

Applicazione del Principio Apriori

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

Items (1-itemsets)

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

Minimum Support = 3

Considerando tutti i candidati:

$$\binom{6}{1} + \binom{6}{2} + \binom{6}{3} = 6 + 15 + 20 = 41$$

Applicazione del Principio Apriori

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

Items (1-itemsets)

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

Minimum Support = 3

Considerando tutti i candidati:

$$\binom{6}{1} + \binom{6}{2} + \binom{6}{3} = 6 + 15 + 20 = 41$$

Applicando il pruning support-based:

$$\binom{6}{1} + \binom{4}{2} + \binom{4}{3} = 6 + 6 + 4 = 16$$

Applicazione del Principio Apriori

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

Items (1-itemsets)

Itemset
{Bread,Milk}
{Bread, Beer }
{Bread,Diaper}
{Beer, Milk}
{Diaper, Milk}
{Beer,Diaper}

Coppie (2-itemsets)

(Non è necessario generare gli itemset candidati che contengono Coke o Eggs)

Minimum Support = 3

Considerando tutti i candidati:

$$\binom{6}{1} + \binom{6}{2} + \binom{6}{3} = 6 + 15 + 20 = 41$$

Applicando il support-based pruning :

$$\binom{6}{1} + \binom{4}{2} + \binom{4}{3} = 6 + 6 + 4 = 16$$

Applicazione del Principio Apriori

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

Items (1-itemsets)

Itemset	Count
{Bread,Milk}	**3**
{Beer, Bread}	**2**
{Bread,Diaper}	**3**
{Beer,Milk}	**2**
{Diaper,Milk}	**3**
{Beer,Diaper}	**3**

Coppie (2-itemsets)

(Non è necessario generare gli itemset candidati che contengono Coke o Eggs)

Minimum Support = 3

Considerando tutti i candidati:

$$\binom{6}{1} + \binom{6}{2} + \binom{6}{3} = 6 + 15 + 20 = 41$$

Applicando il support-based pruning :

$$\binom{6}{1} + \binom{4}{2} + \binom{4}{3} = 6 + 6 + 4 = 16$$

Illustrating Apriori Principle

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

Items (1-itemsets)

Itemset	Count
{Bread,Milk}	3
{Bread,Beer}	2
{Bread,Diaper}	3
{Milk,Beer}	2
{Milk,Diaper}	3
{Beer,Diaper}	3

Coppie (2-itemsets)

(Non è necessario generare gli itemset candidati che contengono Coke o Eggs)

Minimum Support = 3

Considerando tutti i candidati:

$$\binom{6}{1} + \binom{6}{2} + \binom{6}{3} = 6 + 15 + 20 = 41$$

Applicando il support-based pruning:

$$\binom{6}{1} + \binom{4}{2} + \binom{4}{3} = 6 + 6 + 4 = 16$$

Triple (3-itemsets)

Itemset	Count
{ Beer, Diaper, Milk}	2
{ Beer,Bread, Diaper}	2
{Bread, Diaper, Milk}	2
{Beer, Bread, Milk}	1

Algoritmo Apriori

- F_k: insieme dei k-itemsets frequenti
- L_k: insieme dei k-itemsets candidati

- Algoritmo:
  - Sia k=1
  - Genera F_1 = {1-itemsets frequente}
  - Repeat until F_k is empty
    - **Generazione dei Candidati**: Genera L_{k+1} da F_k
    - **Pruning dei Candidati**: Effettua il pruning degli itemset candidati in L_{k+1} contenenti sottoinsiemi di lunghezza k che sono non frequenti
    - **Conteggio del Supporto**: Conta il support di ciascun itemset candidato in L_{k+1} attraverso una scansione del DB
    - **Eliminazione dei Candidati**: Elimina i candidati in L_{k+1} che sono non frequenti, llasciando solo quelli che sono frequenti => F_{k+1}

Generazione Candidati: Forza Bruta (naïve)

Items

Item
Beer
Bread
Cola
Diapers
Milk
Eggs

Candidate Generation

Itemset
{Beer, Bread, Cola}
{Beer, Bread, Diapers}
{Beer, Bread, Milk}
{Beer, Bread, Eggs}
{Beer, Cola, Diapers}
{Beer, Cola, Milk}
{Beer, Cola, Eggs}
{Beer, Diapers, Milk}
{Beer, Diapers, Eggs}
{Beer, Milk, Eggs}
{Bread, Cola, Diapers}
{Bread, Cola, Milk}
{Bread, Cola, Eggs}
{Bread, Diapers, Milk}
{Bread, Diapers, Eggs}
{Bread, Milk, Eggs}
{Cola, Diapers, Milk}
{Cola, Diapers, Eggs}
{Cola, Milk, Eggs}
{Diapers, Milk, Eggs}

Frequent 2-itemset

Itemset
{Beer, Diapers}
{Bread, Diapers}
{Bread, Milk}
{Diapers, Milk}

Candidate Pruning

Itemset
{Bread, Diapers, Milk}

Figure 6.6. A brute-force method for generating candidate 3-itemsets.

Generazione dei Candidati: Metodo $F_{k-1} \times F_1$

Frequent 2-itemset

Itemset
{Beer, Diapers}
{Bread, Diapers}
{Bread, Milk}
{Diapers, Milk}

Frequent 1-itemset

Item
Beer
Bread
Diapers
Milk

Candidate Generation

Itemset
{Beer, Diapers, Bread}
{Beer, Diapers, Milk}
{Bread, Diapers, Milk}
{Bread, Milk, Beer}

Candidate Pruning

Itemset
{Bread, Diapers, Milk}

Figure 6.7. Generating and pruning candidate k-itemsets by merging a frequent $(k-1)$-itemset with a frequent item. Note that some of the candidates are unnecessary because their subsets are infrequent.

Generazione dei Candidati: Metodo $F_{k-1} \times F_{k-1}$

- Effettua la fusione di 2 (k-1)-itemset frequenti se i loro (k-2)-prefissi sono identici

Esempio

- F_3 = { ABC, ABD, ABE, ACD, BCD, BDE, CDE }
  - Merge(**AB**C, **AB**D) = **AB**CD
  - Merge(**AB**C, **AB**E) = **AB**CE
  - Merge(**AB**D, **AB**E) = **AB**DE

  - Non effettua Merge(**A**BD, **A**CD) perché condividono solo un prefisso di lunghezza 1 invece che di lunghezza 2

Pruning dei candidati

- Sia F_3 = { ABC, ABD, ABE, ACD, BCD, BDE, CDE } l'insieme di 3-itemset frequenti

- L_4 = { ABCD, ABCE, ABDE } è l'insieme generato di 4-itemsets frequenti (lucido precedente)

- Pruning dei candidati
  - Taglia ABCE perché ACE e BCE sono infrequenti
  - Taglia ABDE perché ADE è infrequente

- Dopo aver effettuato il pruning: L_4 = { ABCD }

Generazione deie Candidati: Metodo F_{k-1} x F_{k-1}

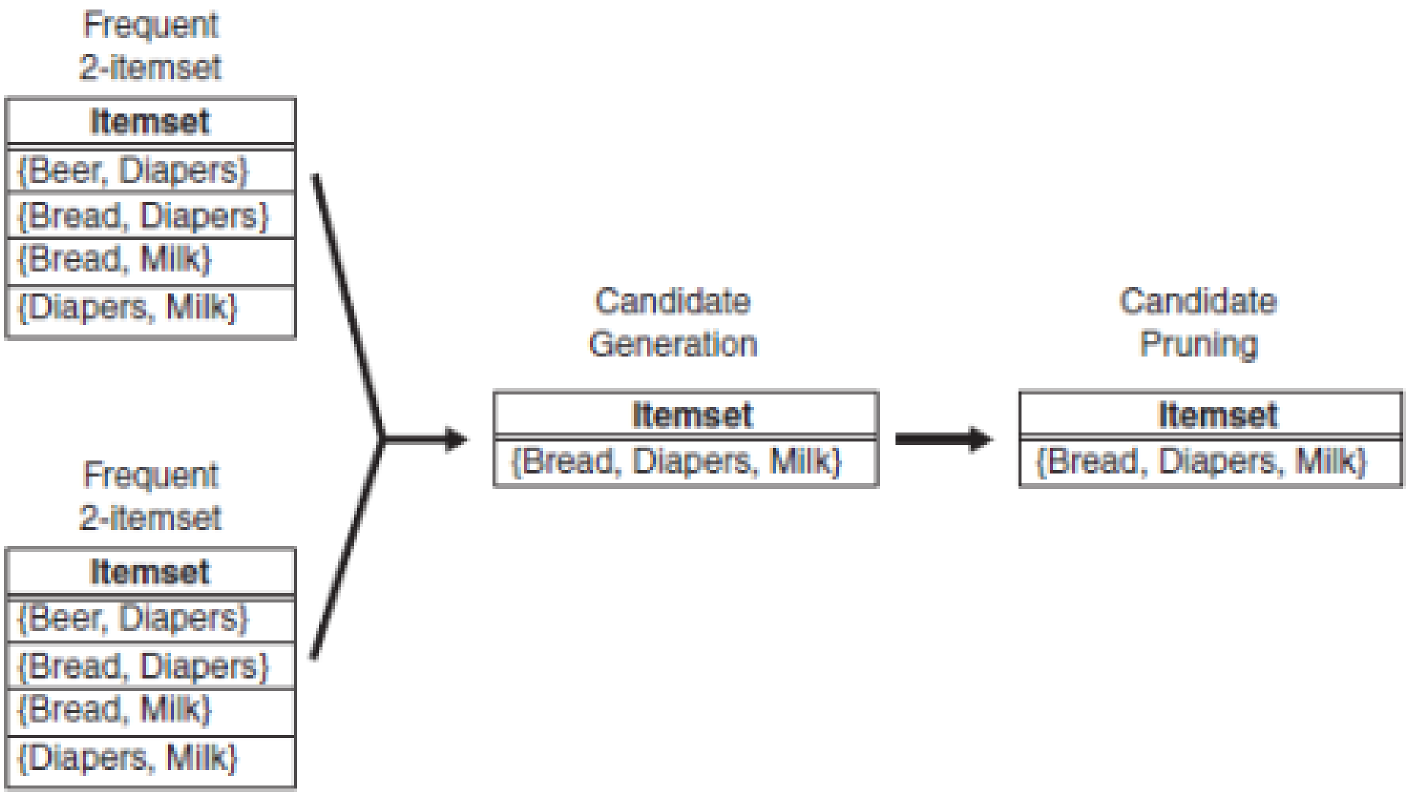


Figure 6.8. Generating and pruning candidate k-itemsets by merging pairs of frequent $(k-1)$-itemsets.

Metodo alternativo F_{k-1} x F_{k-1}

- Effettua la fusione di 2 (k-1)-itemset frequenti se il(k-2)-suffisso del primo coincide con il (k-2)-prefisso del secondo.

Esempio

- F_3 = { ABC, ABD, ABE, ACD, BCD, BDE, CDE}
  - Merge(A**BC**, **BC**D) = A**BC**D
  - Merge(A**BD**, **BD**E) = A**BD**E
  - Merge(A**CD**, **CD**E) = A**CD**E
  - Merge(B**CD**, **CD**E) = B**CD**E

Pruning dei Candidati - Metodo alternativo $F_{k-1} \times F_{k-1}$

- Sia F_3 = { ABC, ABD, ABE, ACD, BCD, BDE,CDE } l'insieme di 3-itemset frequenti

- L_4 = { ABCD, ABDE, ACDE, BCDE } è l'insieme generato di 4-itemset candidati (lucido precedente)

- Pruning dei candidati
  - Taglia ABDE perché ADE è infrequente
  - Taglia ACDE perché ACE e ADE sono infrequenti
  - Taglia BCDE perché BCE è infrequente

- Dopo aver effettuato il pruning: L_4 = { ABCD }.

Principio Apriori

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

Items (1-itemsets)

Itemset	Count
{Bread,Milk}	**3**
{Bread,Beer}	**2**
{Bread,Diaper}	**3**
{Milk,Beer}	**2**
{Milk,Diaper}	**3**
{Beer,Diaper}	**3**

Coppie (2-itemsets)

(NoN vengono generati CANDIDATI CONTENENTI Coke o Eggs)

Minimum Support = 3

Triple (3-itemsets)

Itemset	Count
{Bread, Diaper, Milk}	**2**

Considerando tutti i candidati:

$$\binom{6}{1} + \binom{6}{2} + \binom{6}{3} = 6 + 15 + 20 = 41$$

Applicando il support-based pruning:

$$\binom{6}{1} + \binom{4}{2} + 1 = 6 + 6 + 1 = 13$$

Calcolo del Supporto degli Itemset Candidati

- Effettua la scansione delle transazioni per determinare il supporto di ciascun itemset candidato
  – Bisogna confrontare ciascun itemset candidato con ciascuna transazione. Operazione costosa.

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

Itemset
{ Beer, Diaper, Milk}
{ Beer,Bread,Diaper}
{Bread, Diaper, Milk}
{ Beer, Bread, Milk}

Generazione delle regole

- Dato un itemset frequente L, genera tutti I sottoinsiemi $f \subset L$ tali che $f \to L - f$ soddisfi la soglia del supporto minimo.
  - Se L = {A,B,C,D} è un itemset frequente, le regole candidate sono :

$ABC \to D$,	$ABD \to C$,	$ACD \to B$,	$BCD \to A$,
$A \to BCD$,	$B \to ACD$,	$C \to ABD$,	$D \to ABC$
$AB \to CD$,	$AC \to BD$,	$AD \to BC$,	$BC \to AD$,
$BD \to AC$,	$CD \to AB$,		

- Se $|L| = k$, allora ci sono $2^k - 2$ regole associative candidate (ignorando $L \to \emptyset$ e $\emptyset \to L$)

Generazione delle regole

- In generale, la confidenza non gode della proprietà di anti-monotonicità

  $c(ABC \rightarrow D)$ può essere maggiore o minore di $c(AB \rightarrow D)$

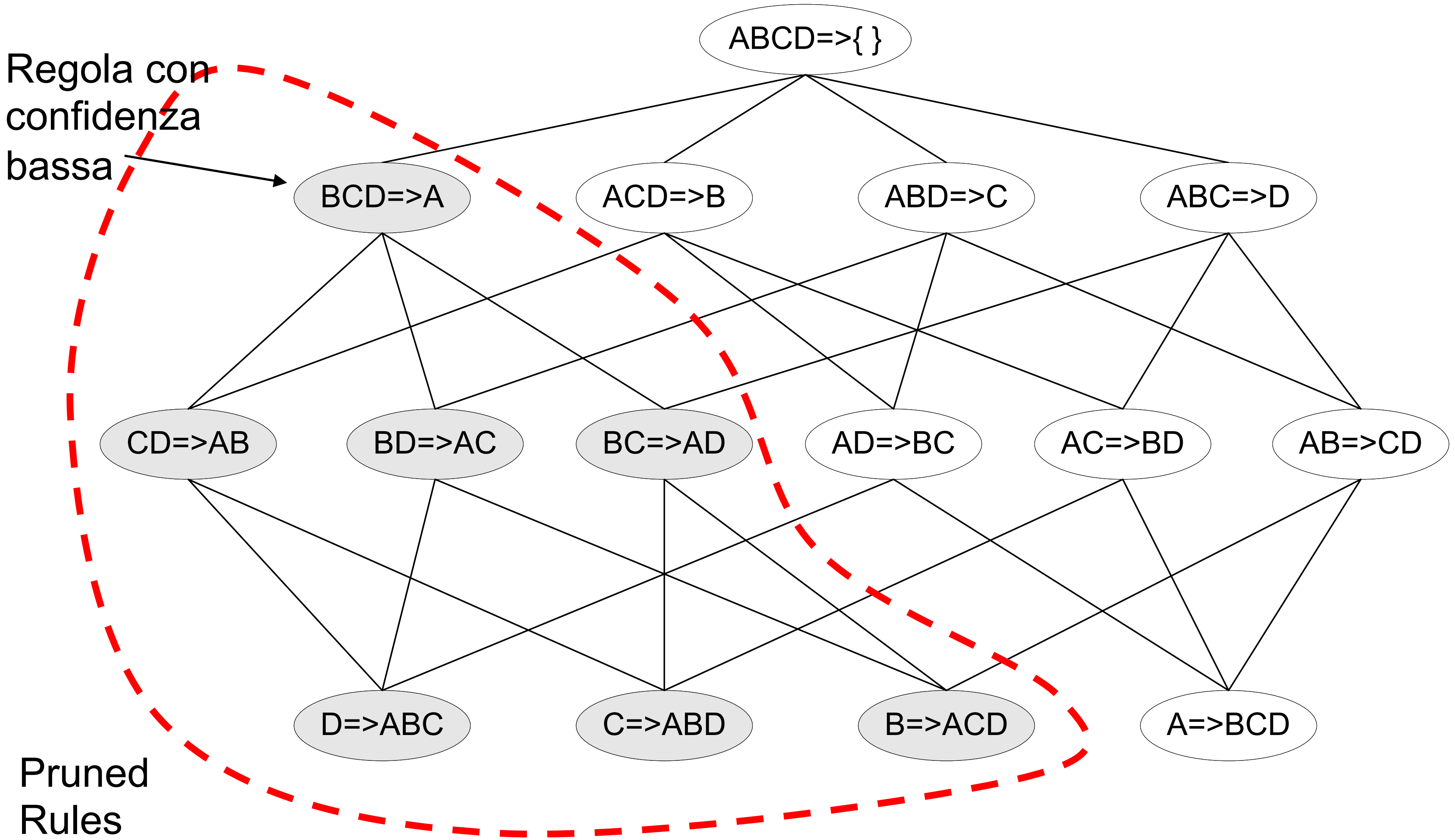
- La confidenza di regole generate dallo stesso itemset soddisfa una proprietà di anti-monotonicità
  - E.g., per l'itemset frequente {A,B,C,D}

    $$c(ABC \rightarrow D) \geq c(AB \rightarrow CD) \geq c(A \rightarrow BCD)$$

- La confidenza è **anti-monotona** rispetto al numero di **item nella testa** (e **monotona** rispetto al numero di **atomi nel corpo**)

Generazione delle regole in Apriori

Lattice of rules



Regole Associative
Algoritmi e Complessità

Fattori che influenzano la complessità di Apriori

- Scelta della soglia minima di supporto
  - abbassando la soglia di supporto si ottengono più itemset frequenti
  - Può crescere il numero di itemset candidati e la lunghezza massima degli itemset frequenti
- Dimensionalità (numero di items) del dataset
  - è necessario maggiore spazio per memorizzare i *support count*
  - La crescita degli itemset frequenti comporta maggiore complessità
- Dimensione del database
  - poiché Apriori effettua più passaggi, il tempo di esecuzione dell'algoritmo può aumentare con il numero di transazioni
- Larghezza media delle transazioni
  - Comporta l'aumento della lunghezza massima degli itemset frequenti e delle visite dell'hash-tree (punto successivo)
- Strutture di memorizzazione
  - L'uso di hash-tree permette una minore complessità

Calcolo del Supporto degli Itemset Candidati

- Effettua la scansione delle transazioni per determinare il supporto di ciascun itemset candidato
  - Bisogna confrontare ciascun itemset candidato con ciascuna transazione. Operazione costosa.

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

Itemset
{ Beer, Diaper, Milk}
{ Beer,Bread,Diaper}
{Bread, Diaper, Milk}
{ Beer, Bread, Milk}

Calcolo del Supporto degli Itemset Candidati

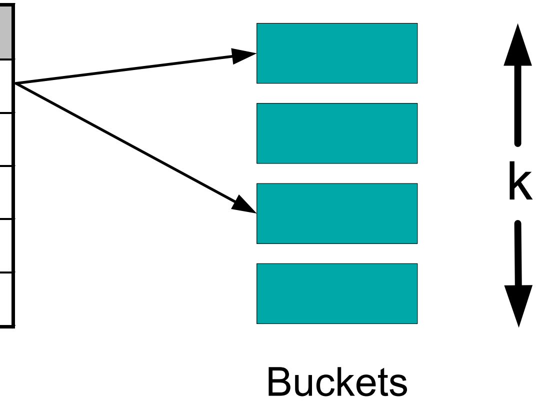
- Per ridurre il numero di confronti, gli itemset candidati sono memorizzati in una struttura hash.
  - Invece di controntare ogni transazione con ogni itemset candidato, confronta la transazione con gli itemset candidati memorizzati nel bucket corrispondente della struttura hash.

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

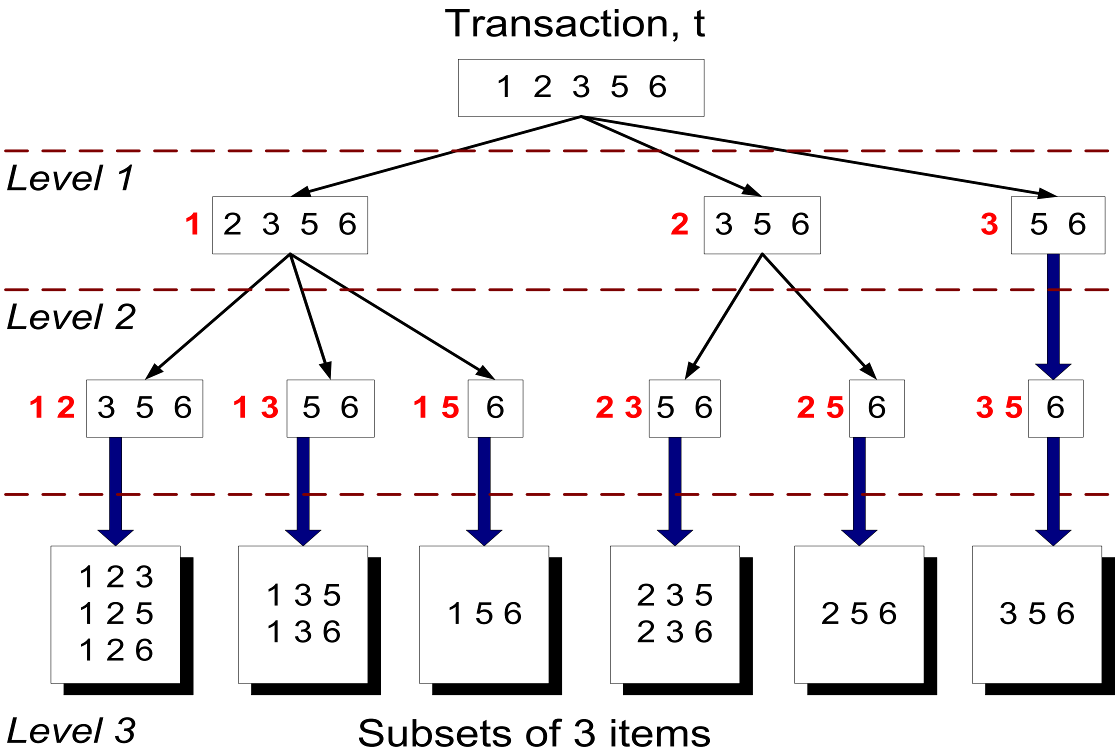
Hash Structure



Calcolo del Supporto: Esempio

Si supponga di averee 15 itemset candidati di lunghezza 3:
{1 4 5}, {1 2 4}, {4 5 7}, {1 2 5}, {4 5 8}, {1 5 9}, {1 3 6}, {2 3 4}, {5 6 7}, {3 4 5}, {3 5 6}, {3 5 7}, {6 8 9}, {3 6 7}, {3 6 8}

Quanti di questi itemset sono supportati dalla transazione (1,2,3,5,6)?



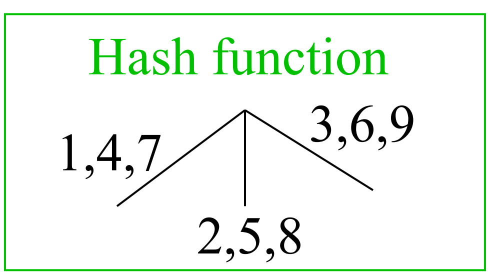
Conteggio del Supporto con Hash Tree

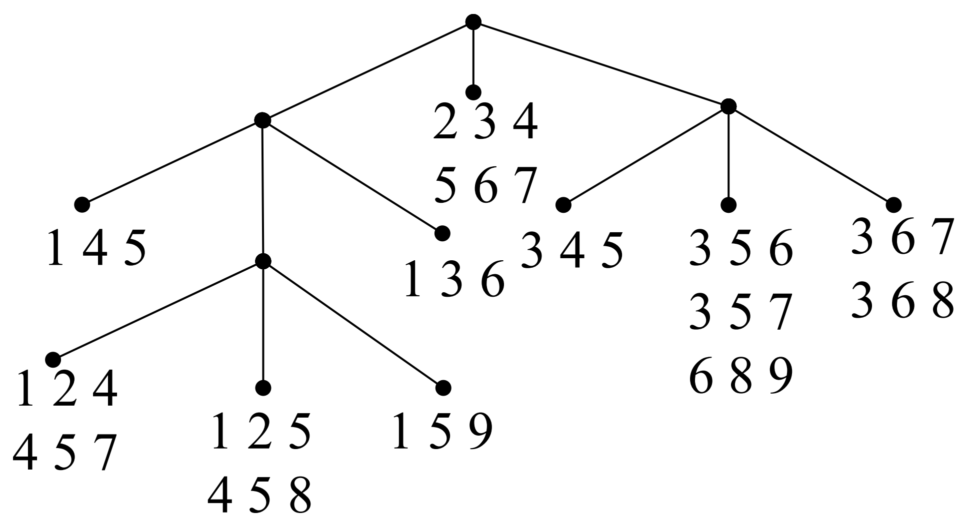
Si supponga di averee 15 itemset candidati di lunghezza 3:
{1 4 5}, {1 2 4}, {4 5 7}, {1 2 5}, {4 5 8}, {1 5 9}, {1 3 6}, {2 3 4}, {5 6 7}, {3 4 5}, {3 5 6}, {3 5 7}, {6 8 9}, {3 6 7}, {3 6 8}

Sono necessari:

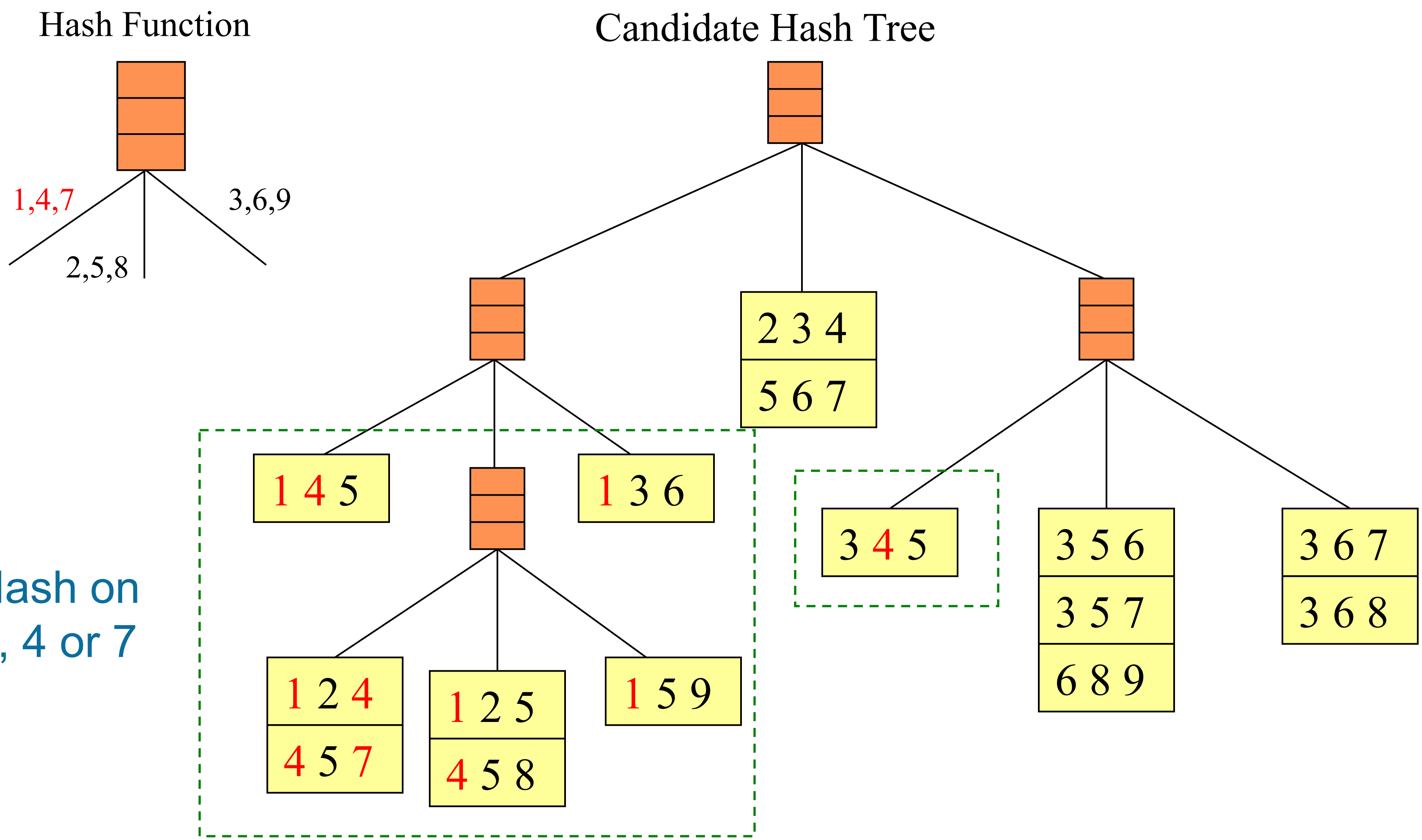
• Funzione Hash (e.g. h(x) = x%3)

• Dimensione nodi foglia (max): numero di itemset memorizzati in un nodo foglia (se il numero di itemset candidati supera max il nodo spezzato in 2)



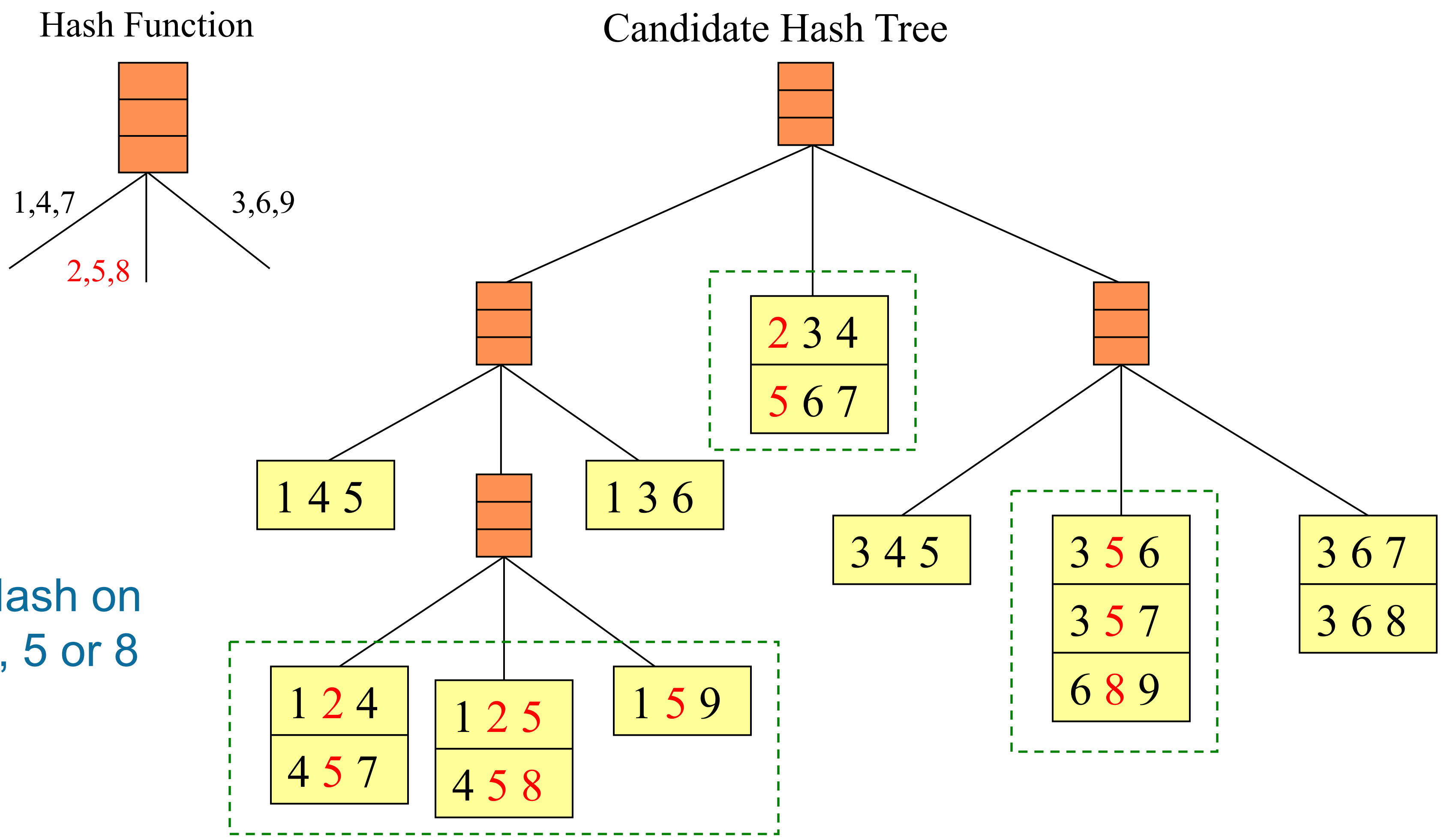


Conteggio del Supporto con Hash Tree



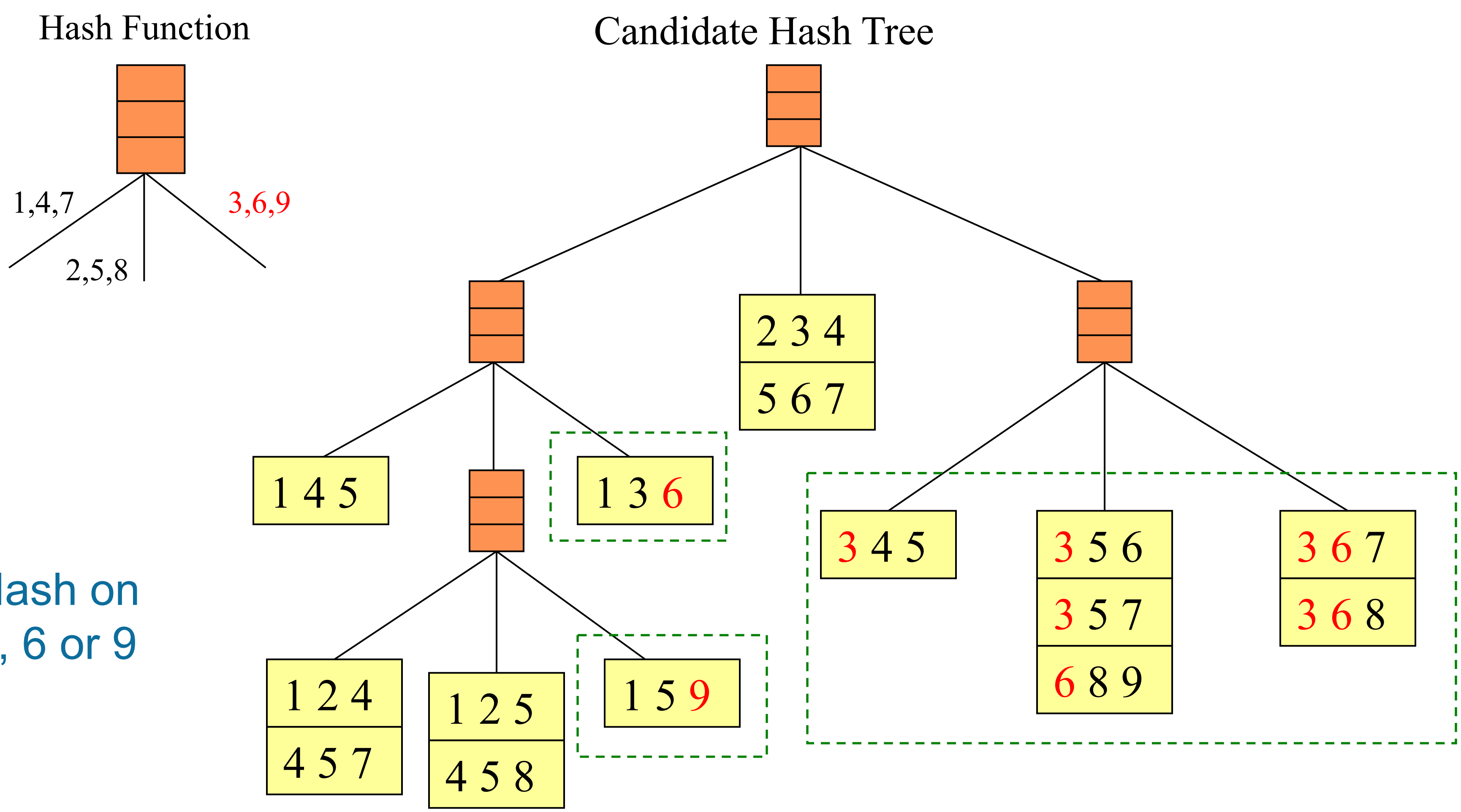
Hash on
1, 4 or 7

Conteggio del Supporto con Hash Tree



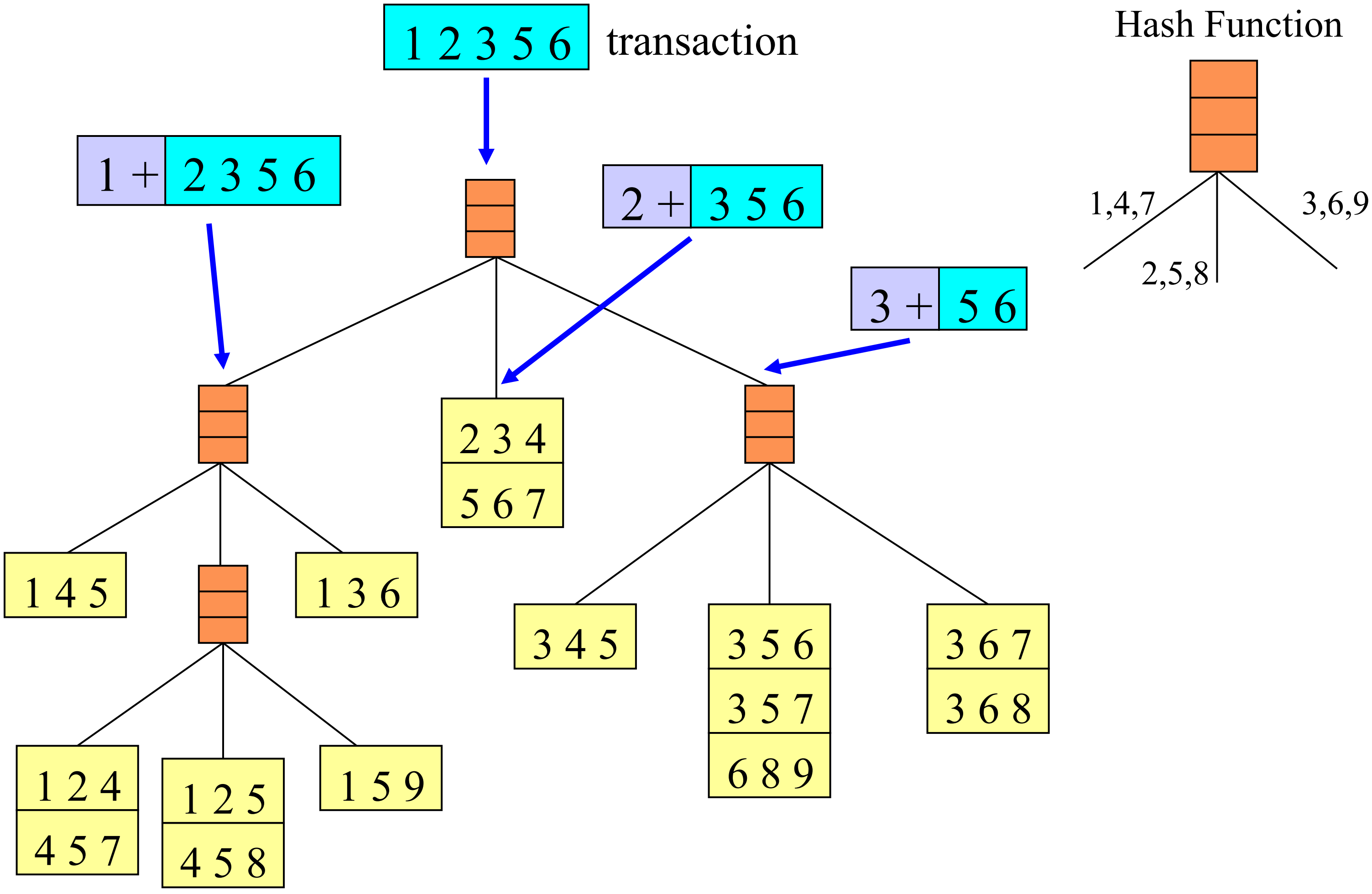
Hash on
2, 5 or 8

Conteggio del Supporto con Hash Tree

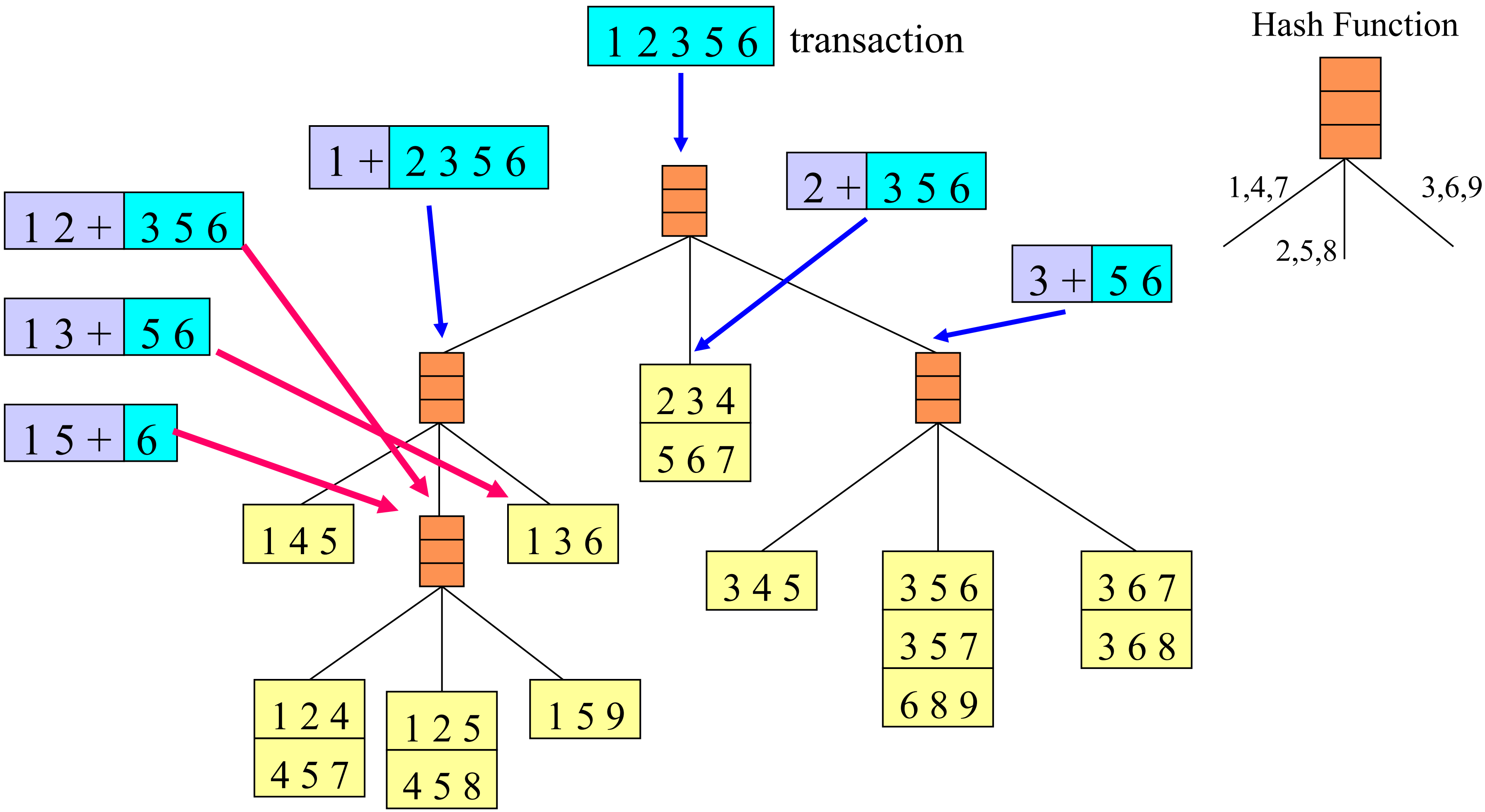


Hash on
3, 6 or 9

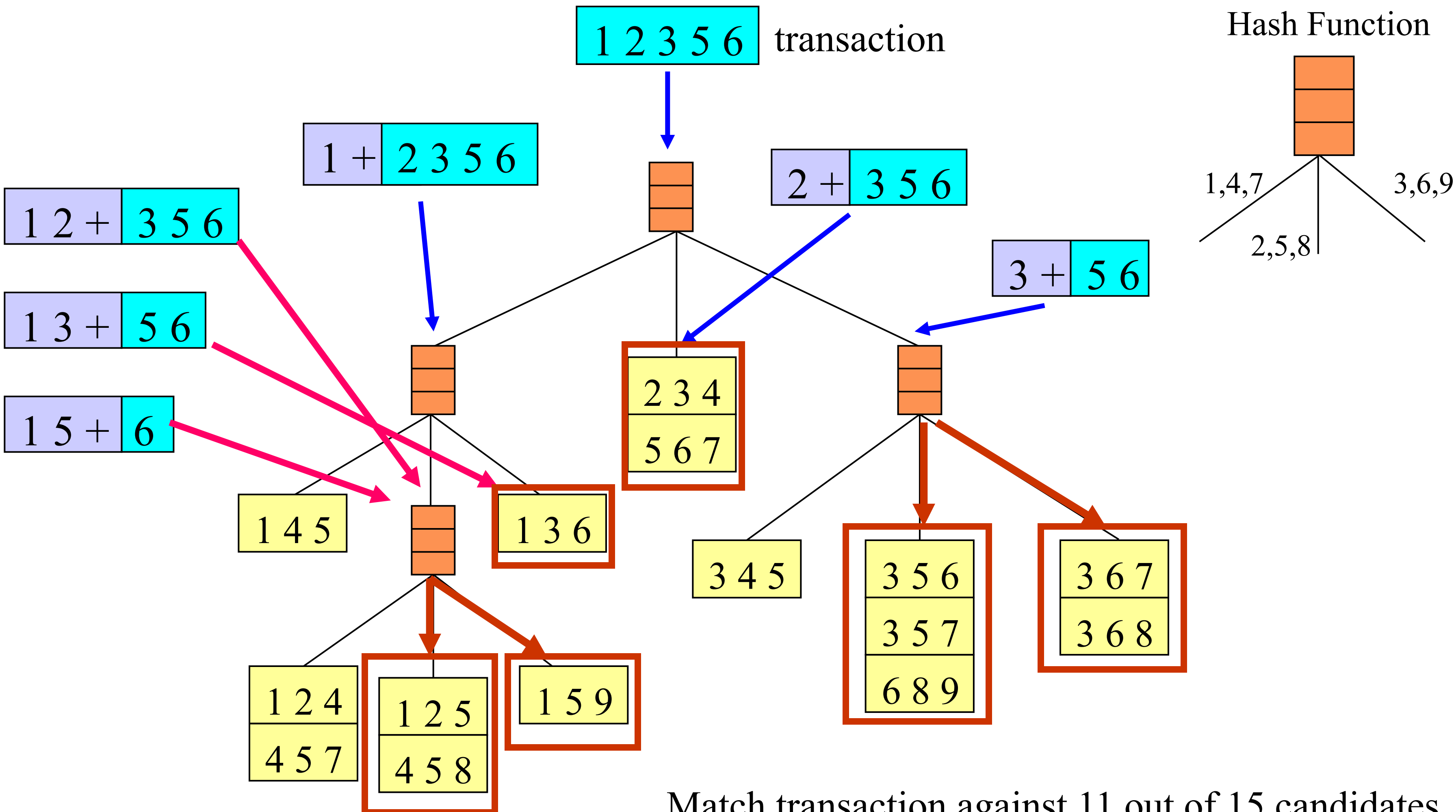
Conteggio del Supporto con Hash Tree



Conteggio del Supporto con Hash Tree

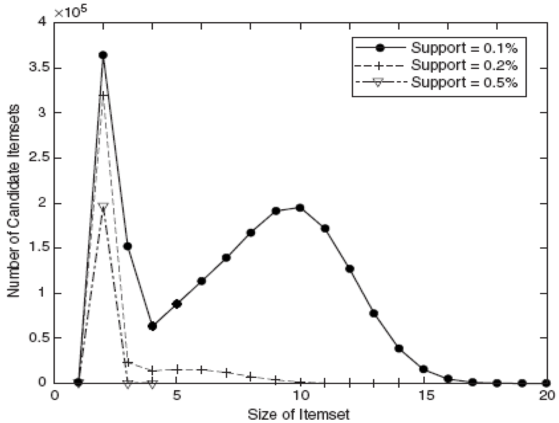


Conteggio del Supporto con Hash Tree

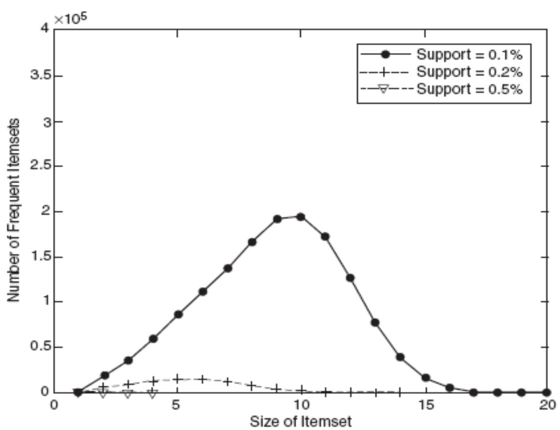


Match transaction against 11 out of 15 candidates

Fattori che influenzano la complessità di Apriori

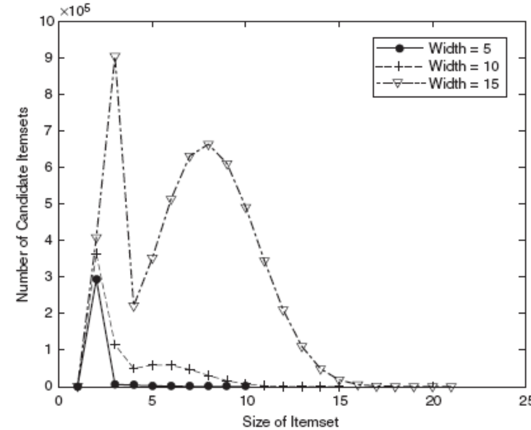


(a) Number of candidate itemsets.

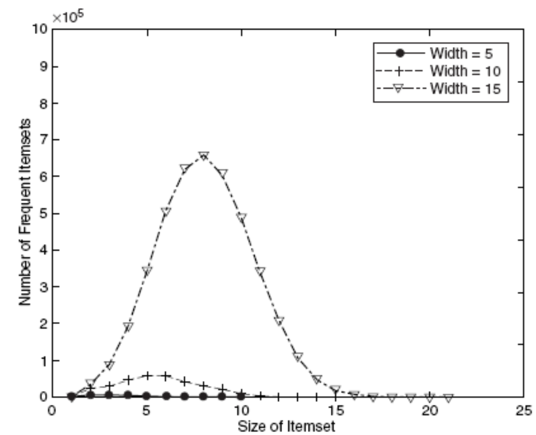


(b) Number of frequent itemsets.

Figure 6.13. Effect of support threshold on the number of candidate and frequent itemsets.



(a) Number of candidate itemsets.



(b) Number of Frequent Itemsets.

Figure 6.14. Effect of average transaction width on the number of candidate and frequent itemsets.

Representatione Compatta degli Itemset Frequenti

- Alcuni itemset sono ridondanti perché hanno un supporto uguale a quello di alcuni loro superset

TID	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1

- Numero di Itemset $= 3 \times \sum_{k=0}^{10} \binom{10}{k}$

- E’ recessaria una rappresentazione “compatta” (cosiderare solo quelli significativi)

Itemset Frequenti Massimali

Un itemset frequente è massimale se nessunio dei suoi immediati superset è frequente.

Itemset Frequenti Massimali - Esempio

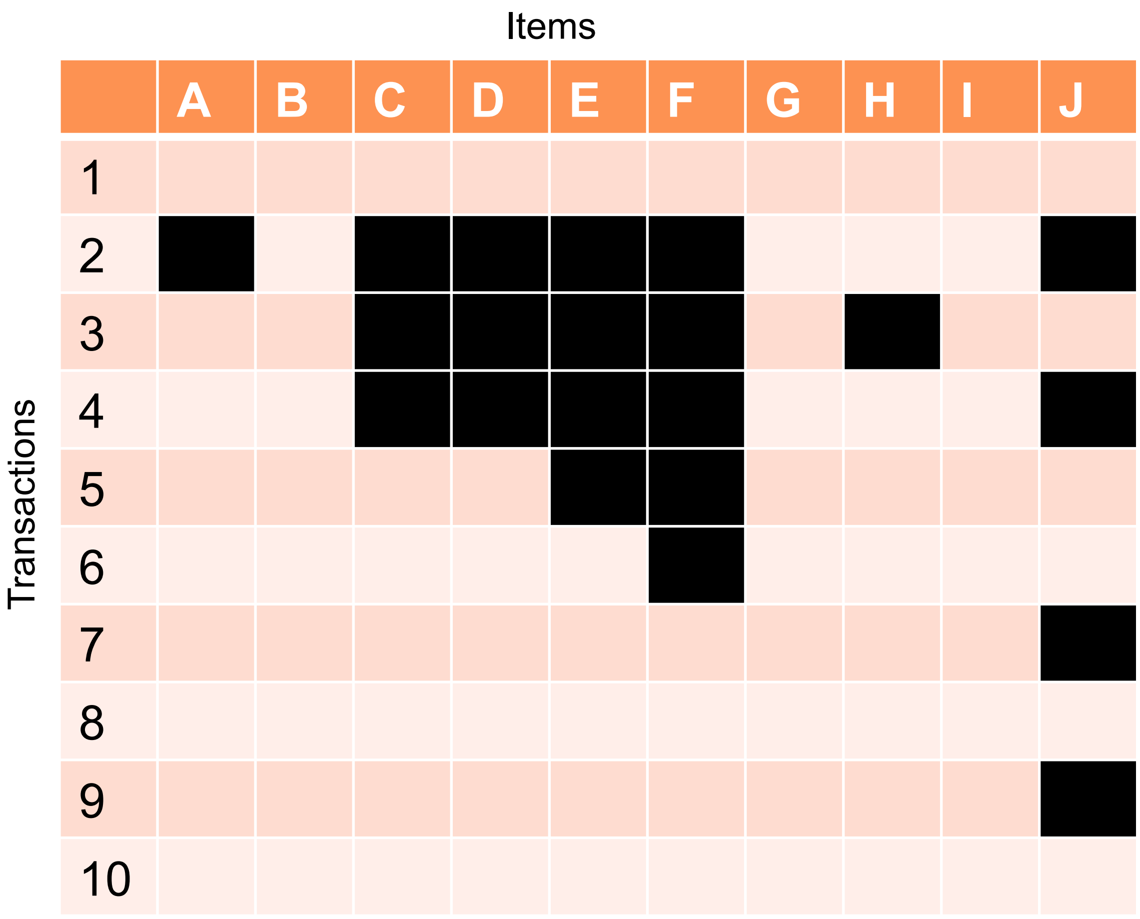
Items

Transactions

	A	B	C	D	E	F	G	H	I	J
1										
2	■		■	■	■	■				■
3			■	■	■	■		■		
4			■	■	■	■				■
5					■	■				
6						■				
7										■
8										
9										■
10										

Support threshold (by count) : 5
Frequent itemsets: ?

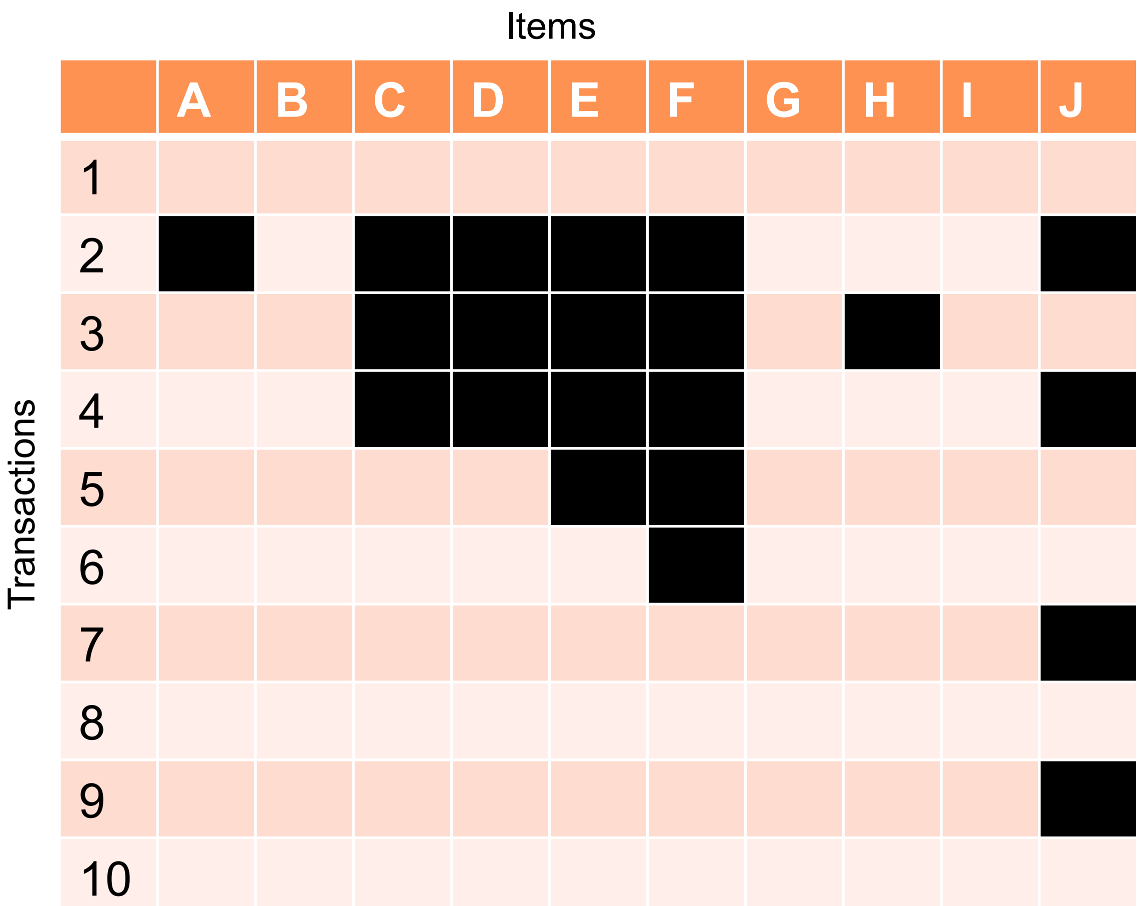
Itemset Frequenti Massimali - Esempio



Support threshold (by count) : 5

Frequent itemsets: {F}

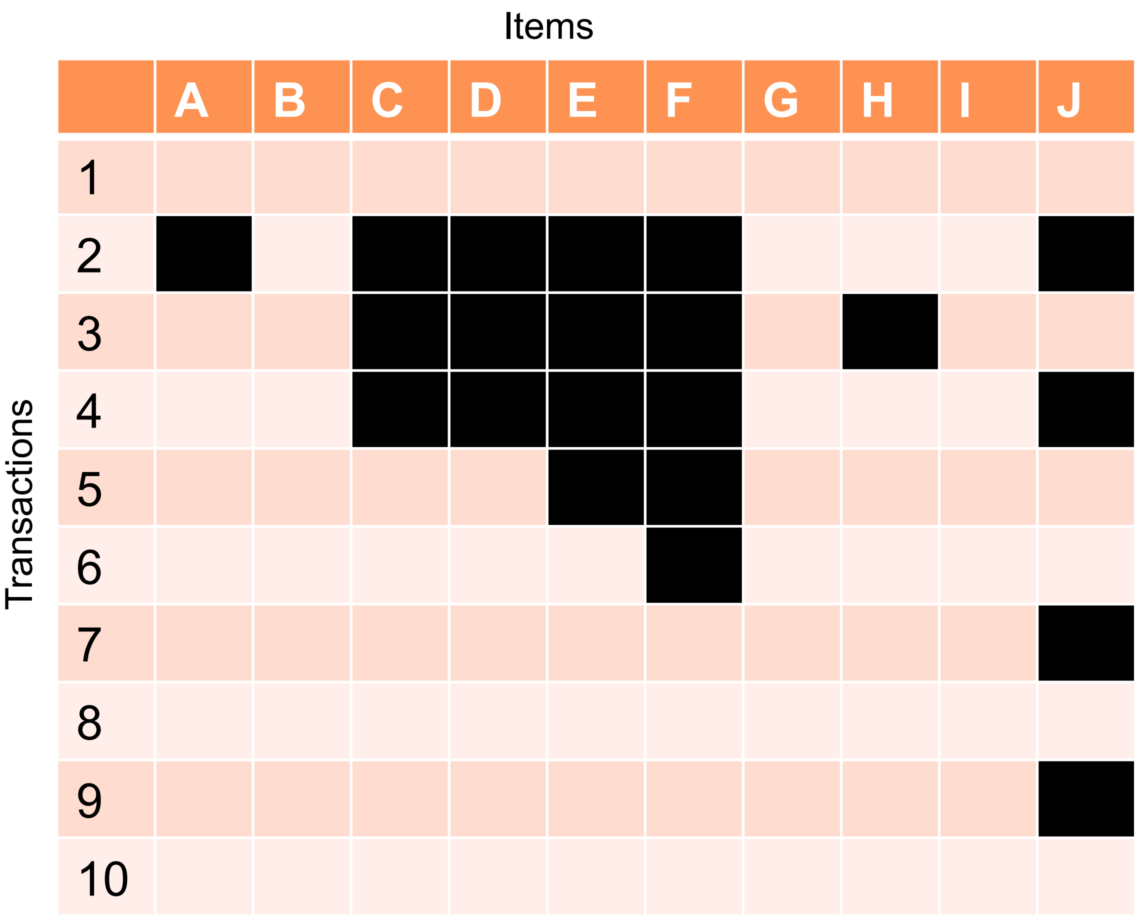
Itemset Frequenti Massimali - Esempio



Support threshold (by count) : 5
Frequent itemsets: {F}

Support threshold (by count): 4
Frequent itemsets: ?

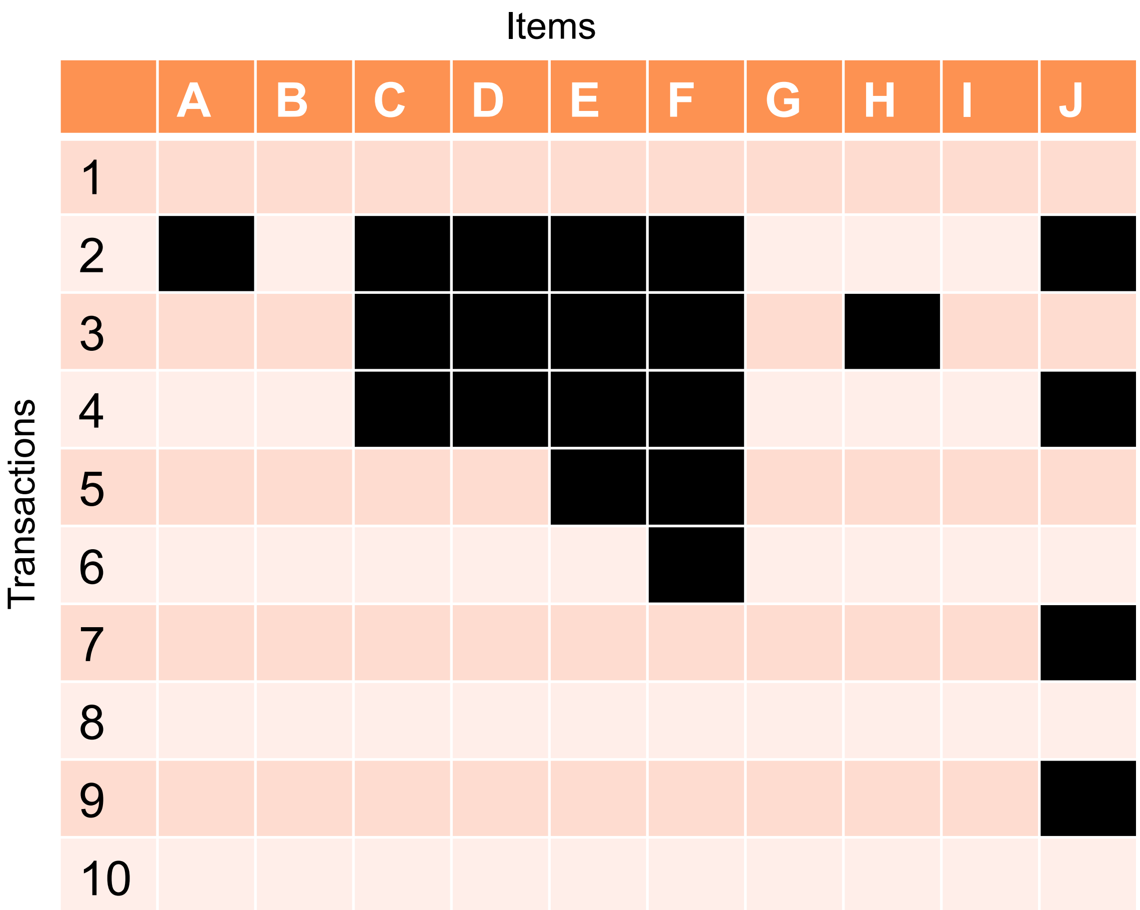
Itemset Frequenti Massimali - Esempio



Support threshold (by count) : 5
Frequent itemsets: {F}

Support threshold (by count): 4
Frequent itemsets: {E}, {F}, {E,F}, {J}

Itemset Frequenti Massimali - Esempio

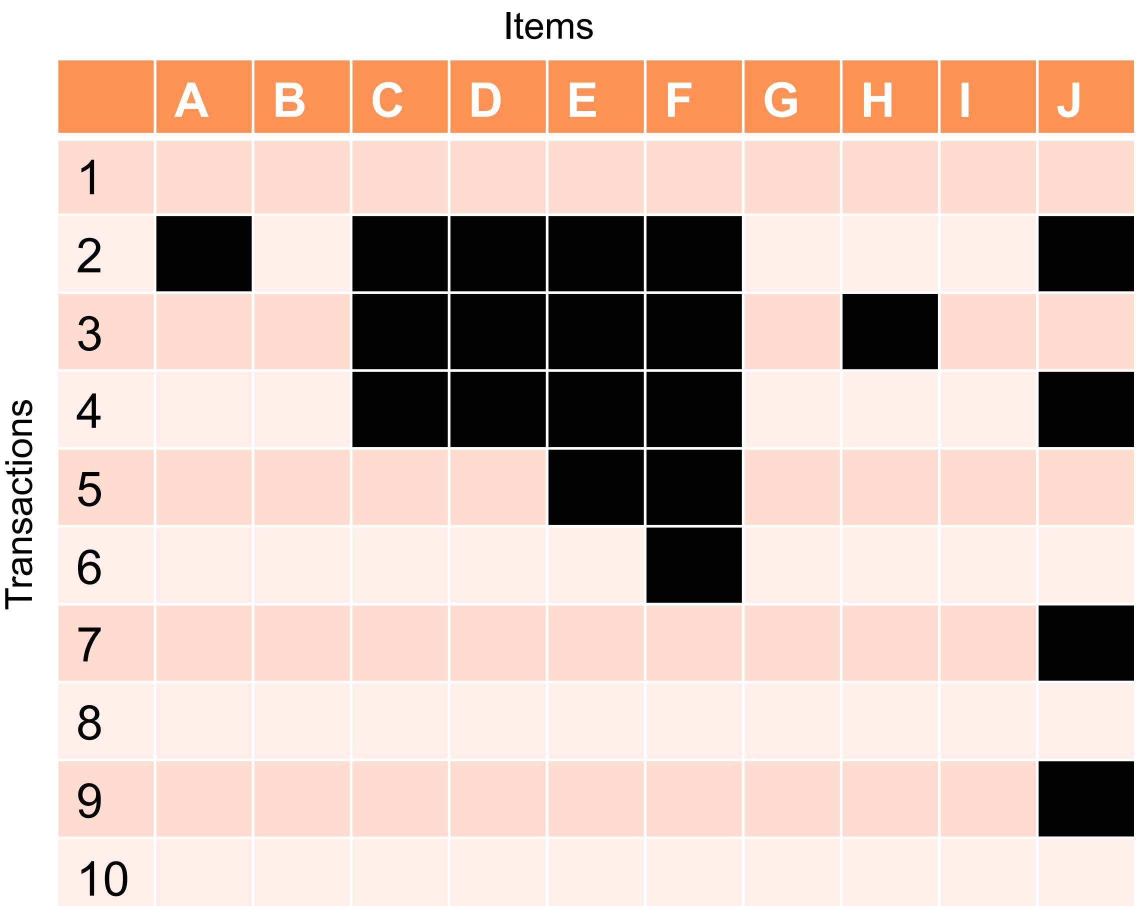


Support threshold (by count) : 5
Frequent itemsets: {F}

Support threshold (by count): 4
Frequent itemsets: {E}, {F}, {E,F}, {J}

Support threshold (by count): 3
Frequent itemsets: ?

Itemset Frequenti Massimali - Esempio

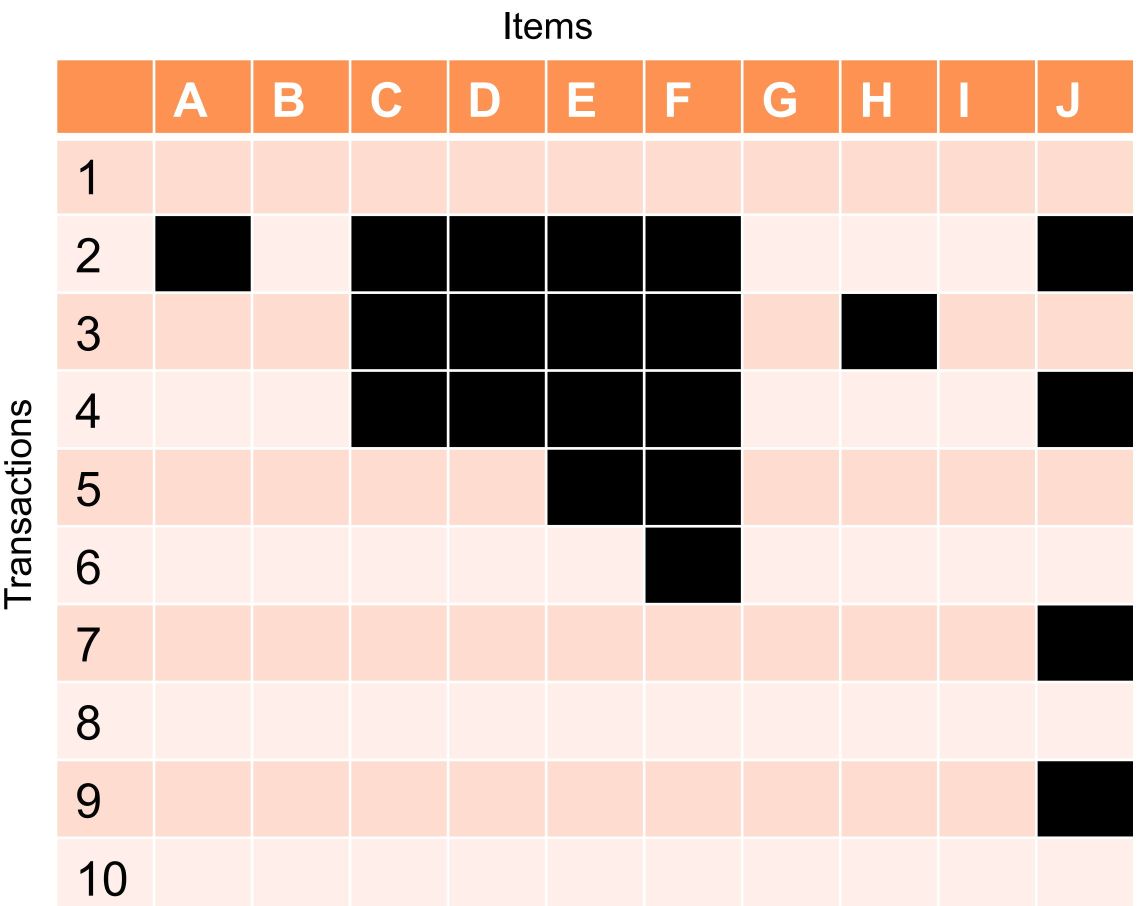


Support threshold (by count) : 5
Frequent itemsets: {F}

Support threshold (by count): 4
Frequent itemsets: {E}, {F}, {E,F}, {J}

Support threshold (by count): 3
Frequent itemsets:
All subsets of {C,D,E,F} + {J}

Itemset Frequenti Massimali - Esempio

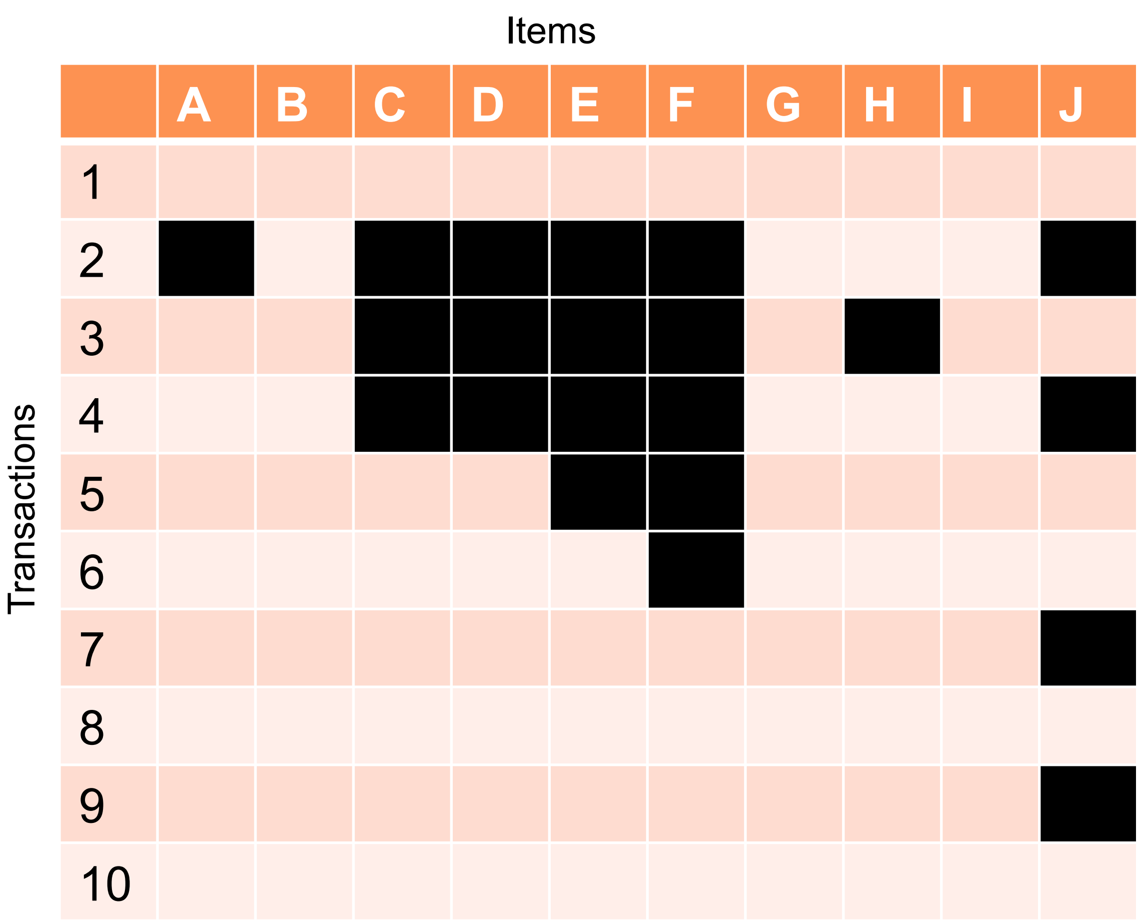


Support threshold (by count) : 5
Frequent itemsets: {F}
Maximal itemsets: ?

Support threshold (by count): 4
Frequent itemsets: {E}, {F}, {E,F}, {J}
Maximal itemsets: ?

Support threshold (by count): 3
Frequent itemsets:
All subsets of {C,D,E,F} + {J}
Maximal itemsets: ?

Itemset Frequenti Massimali - Esempio

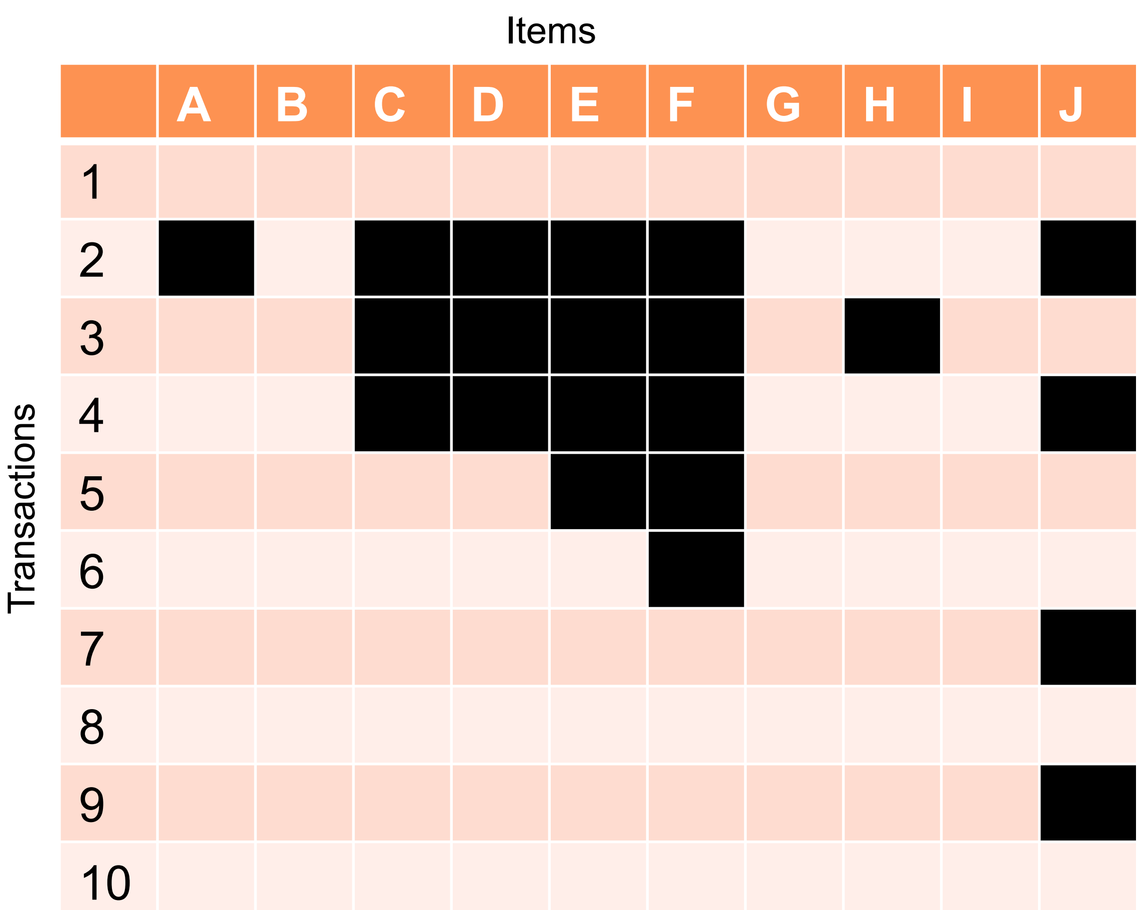


Support threshold (by count) : 5
Frequent itemsets: {F}
Maximal itemsets: {F}

Support threshold (by count): 4
Frequent itemsets: {E}, {F}, {E,F}, {J}
Maximal itemsets: ?

Support threshold (by count): 3
Frequent itemsets:
All subsets of {C,D,E,F} + {J}
Maximal itemsets: ?

Itemset Frequenti Massimali - Esempio

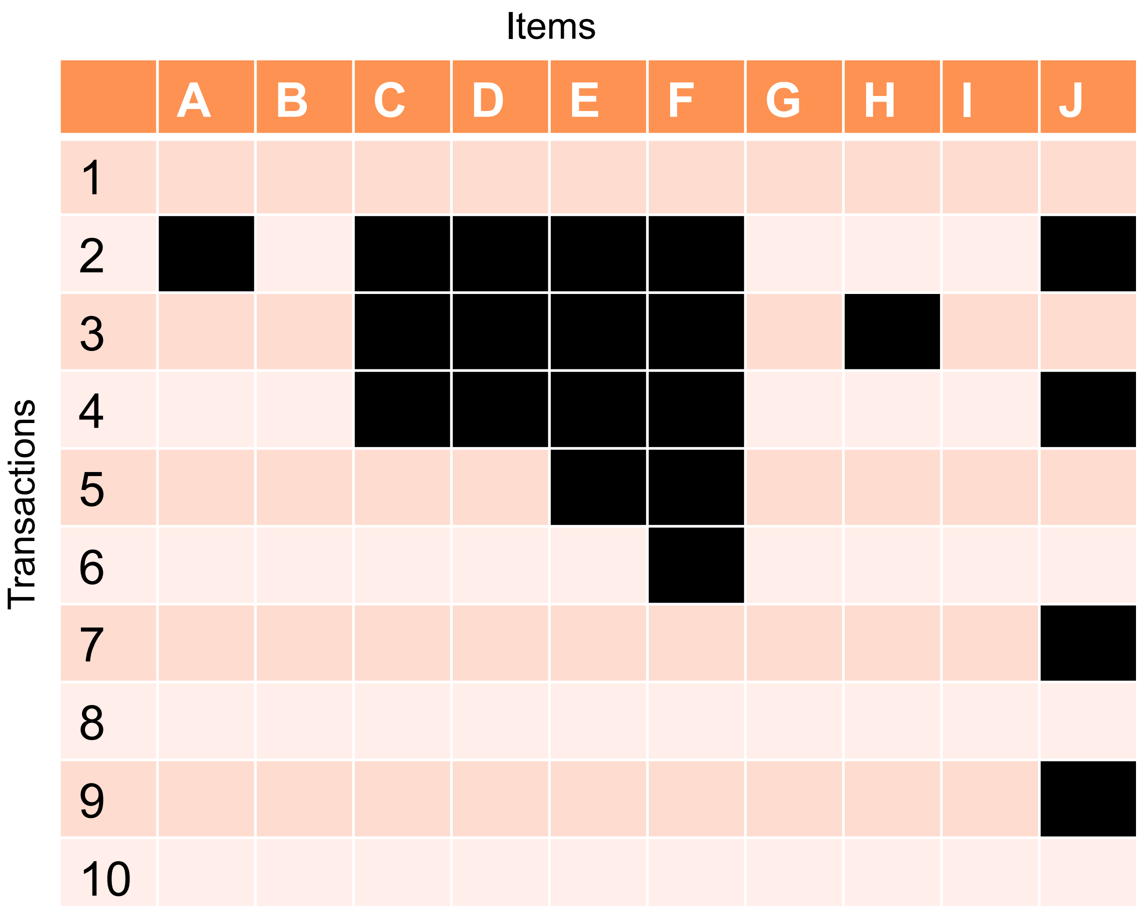


Support threshold (by count) : 5
Frequent itemsets: {F}
Maximal itemsets: {F}

Support threshold (by count): 4
Frequent itemsets: {E}, {F}, {E,F}, {J}
Maximal itemsets: {E,F}, {J}

Support threshold (by count): 3
Frequent itemsets:
All subsets of {C,D,E,F} + {J}
Maximal itemsets: ?

Itemset Frequenti Massimali - Esempio

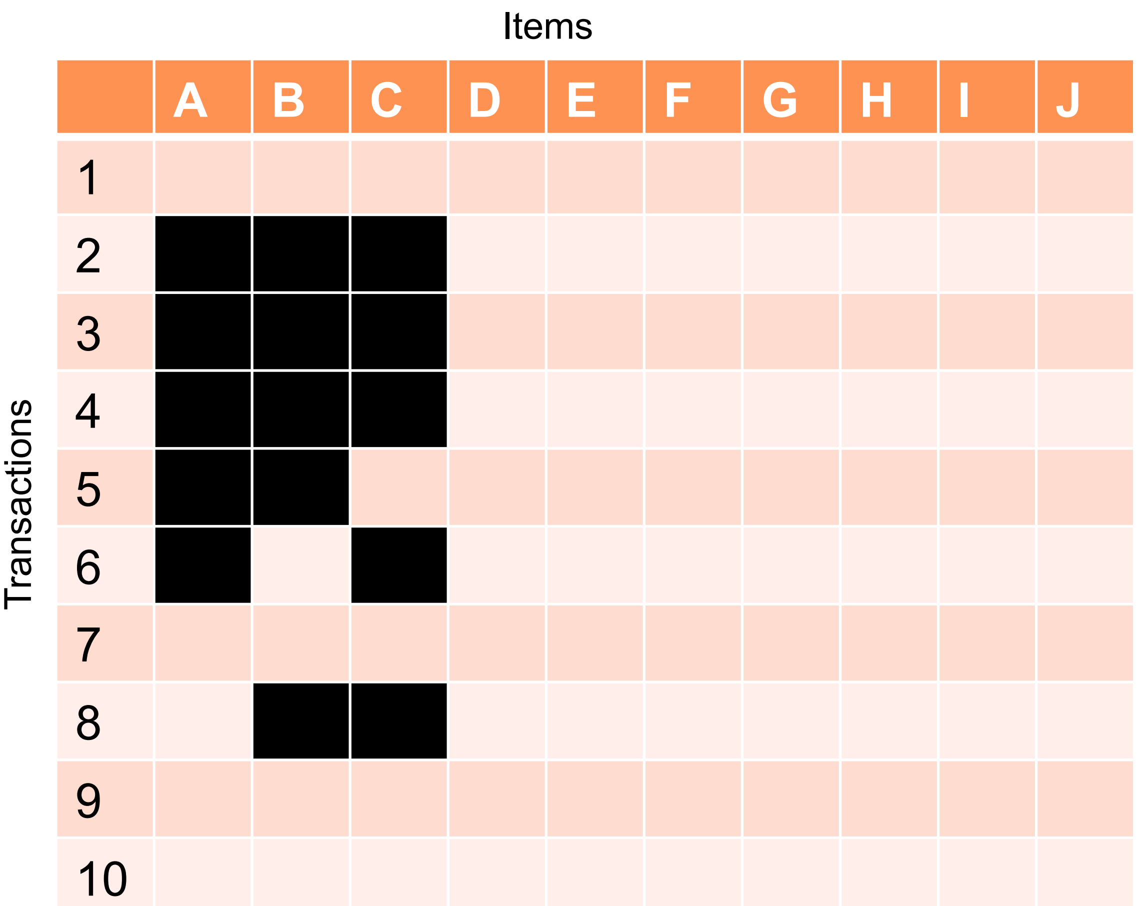


Support threshold (by count) : 5
Frequent itemsets: {F}
Maximal itemsets: {F}

Support threshold (by count): 4
Frequent itemsets: {E}, {F}, {E,F}, {J}
Maximal itemsets: {E,F}, {J}

Support threshold (by count): 3
Frequent itemsets:
All subsets of {C,D,E,F} + {J}
Maximal itemsets:
{C,D,E,F}, {J}

Itemset Frequenti Massimali - Esempio



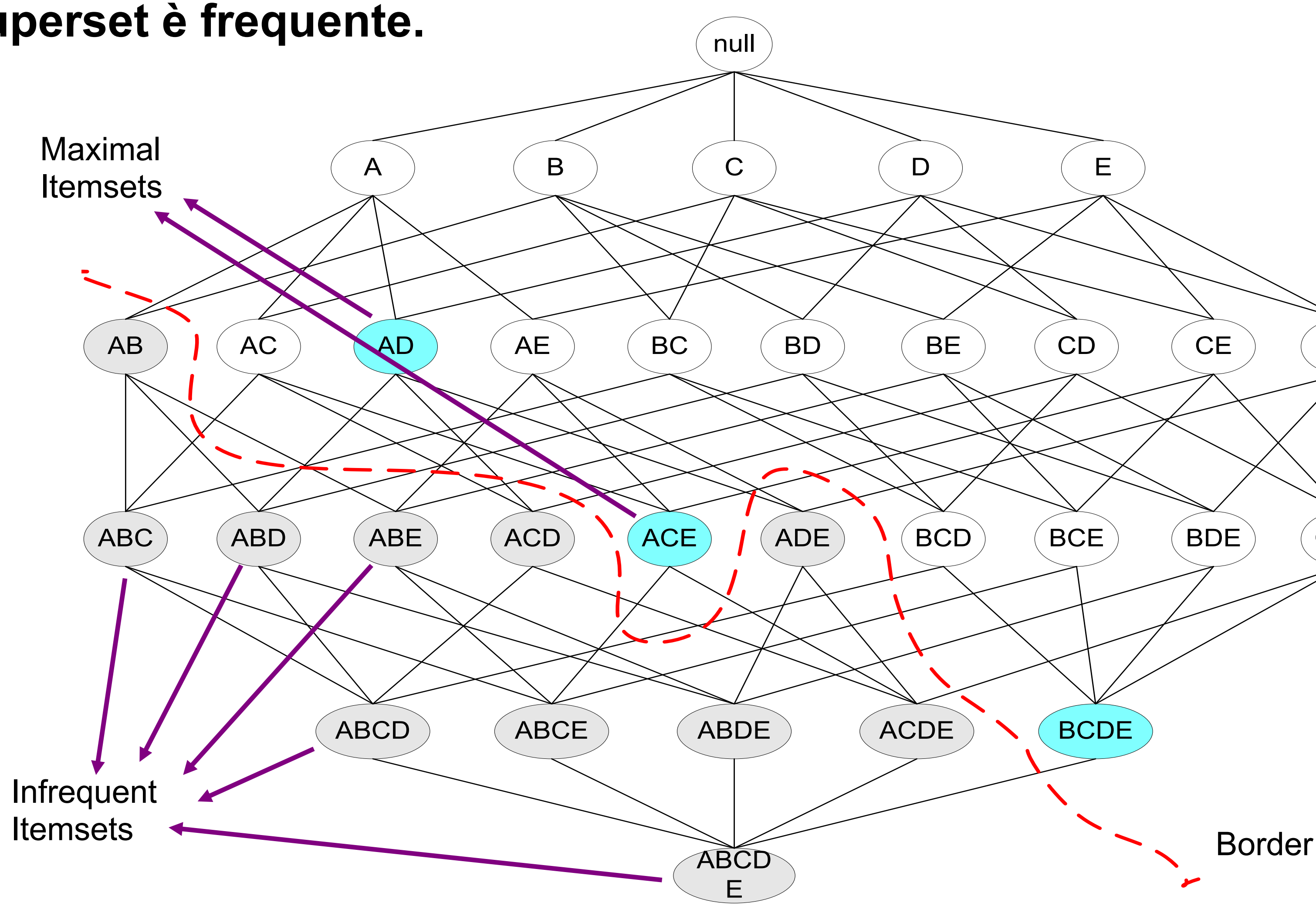
Support threshold (by count) : 5
Maximal itemsets: {A}, {B}, {C}

Support threshold (by count): 4
Maximal itemsets: {A,B}, {A,C},{B,C}

Support threshold (by count): 3
Maximal itemsets: {A,B,C}

Itemset Frequenti Massimali

Un itemset frequente è massimale se nessunio dei suoi immediati superset è frequente.



Itemset Chiusi

- Un itemset X è chiuso se nessuno dei suoi immediati superset ha lo stesso supporto di X.
- X è non-chiuso se almeno uno dei suoi immediati superset ha lo stesso support di X.

Itemset Chiusi

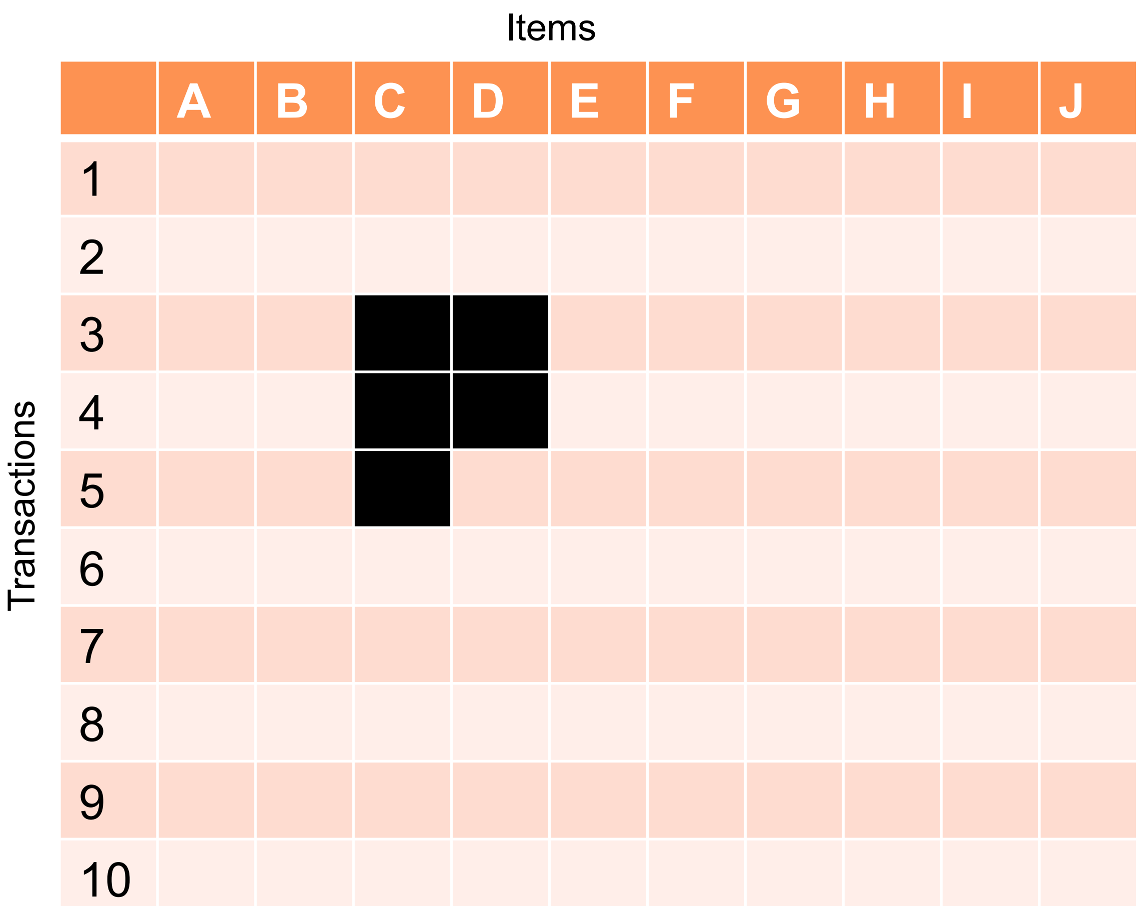
- Un itemset X è chiuso se nessuno dei suoi immediati superset ha lo stesso supporto di X.
- X è non-chiuso se almeno uno dei suoi immediati superset ha lo stesso support di X.

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,B,C,D}
4	{A,B,D}
5	{A,B,C,D}

Itemset	Support
{A}	4
{B}	5
{C}	3
{D}	4
{A,B}	4
{A,C}	2
{A,D}	3
{B,C}	3
{B,D}	4
{C,D}	3

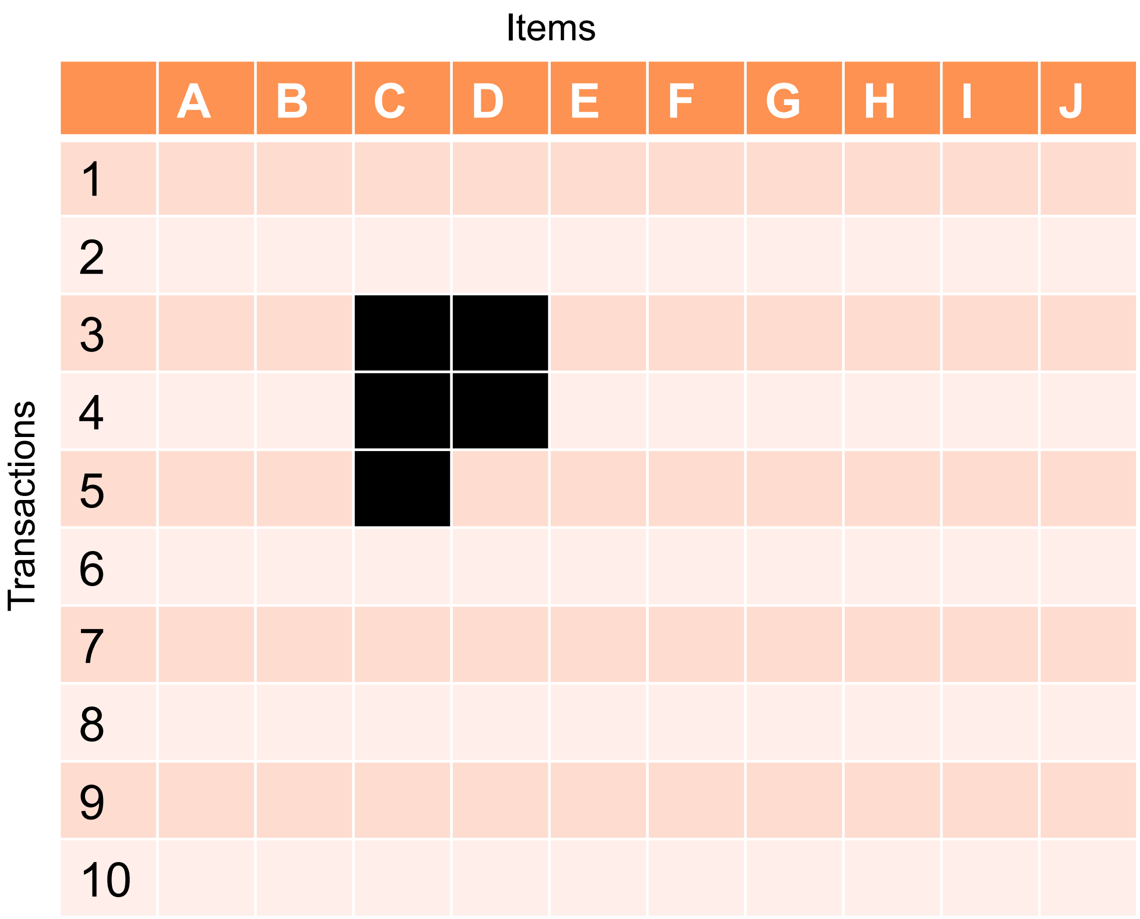
Itemset	Support
{A,B,C}	2
{A,B,D}	3
{A,C,D}	2
{B,C,D}	2
{A,B,C,D}	2

Esempio 1



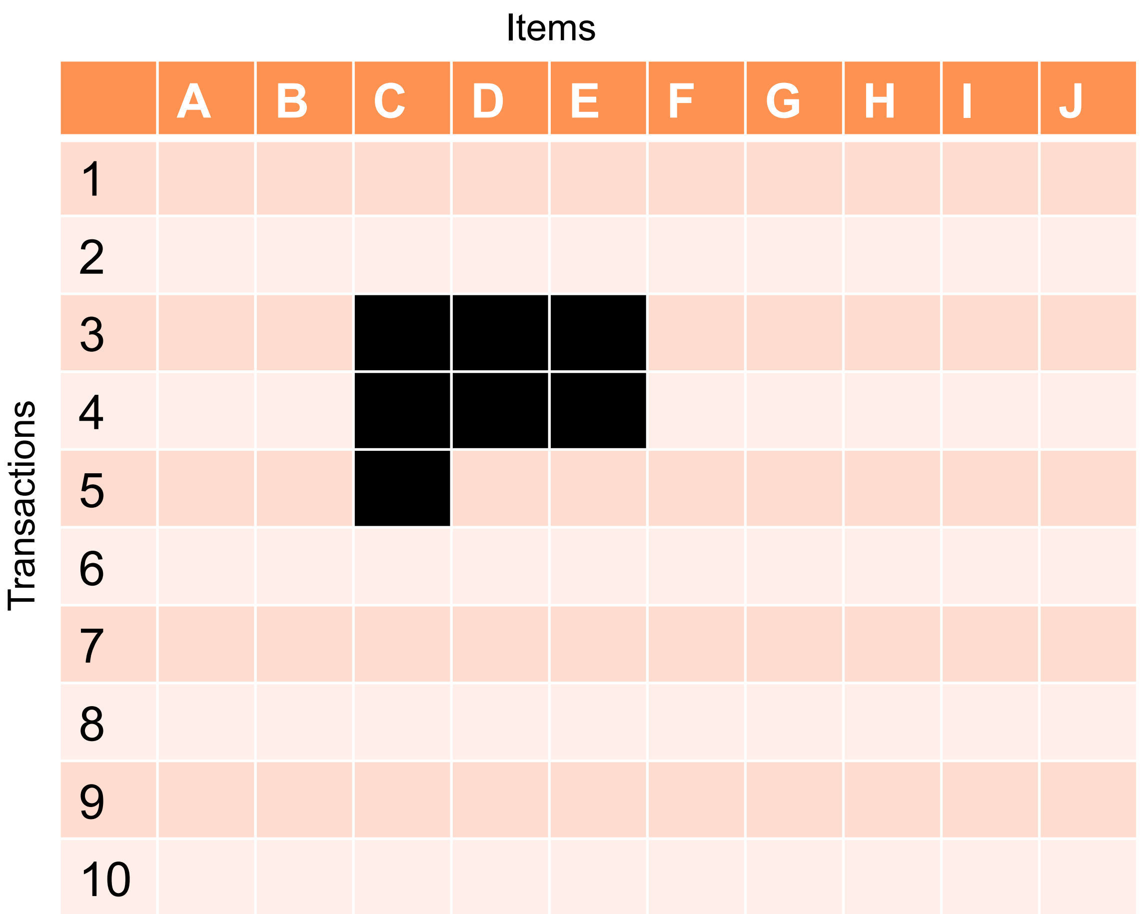
Itemsets	Support (counts)	Closed itemsets
{C}	3	
{D}	2	
{C,D}	2	

Esempio 1



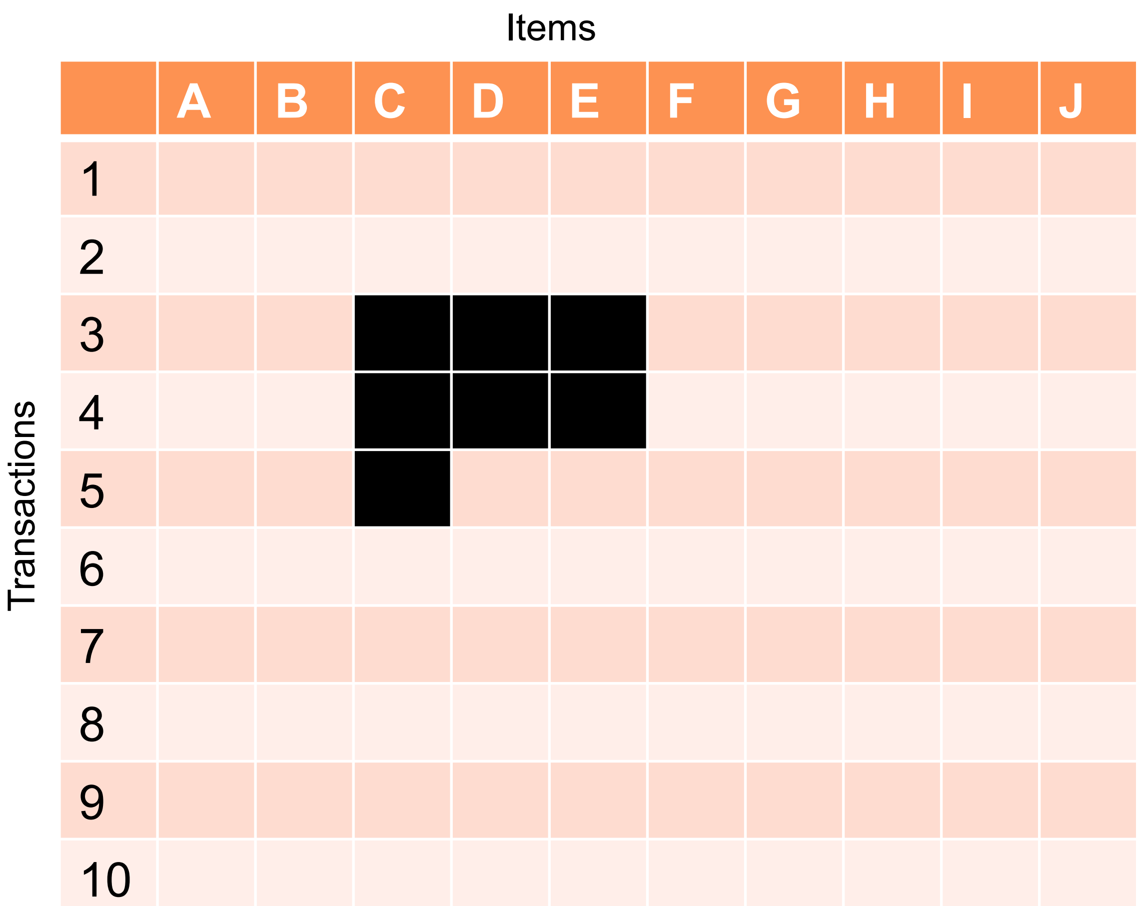
Itemsets	Support (counts)	Closed itemsets
{C}	**3**	✓
{D}	2	
{C,D}	**2**	✓

Esempio 2



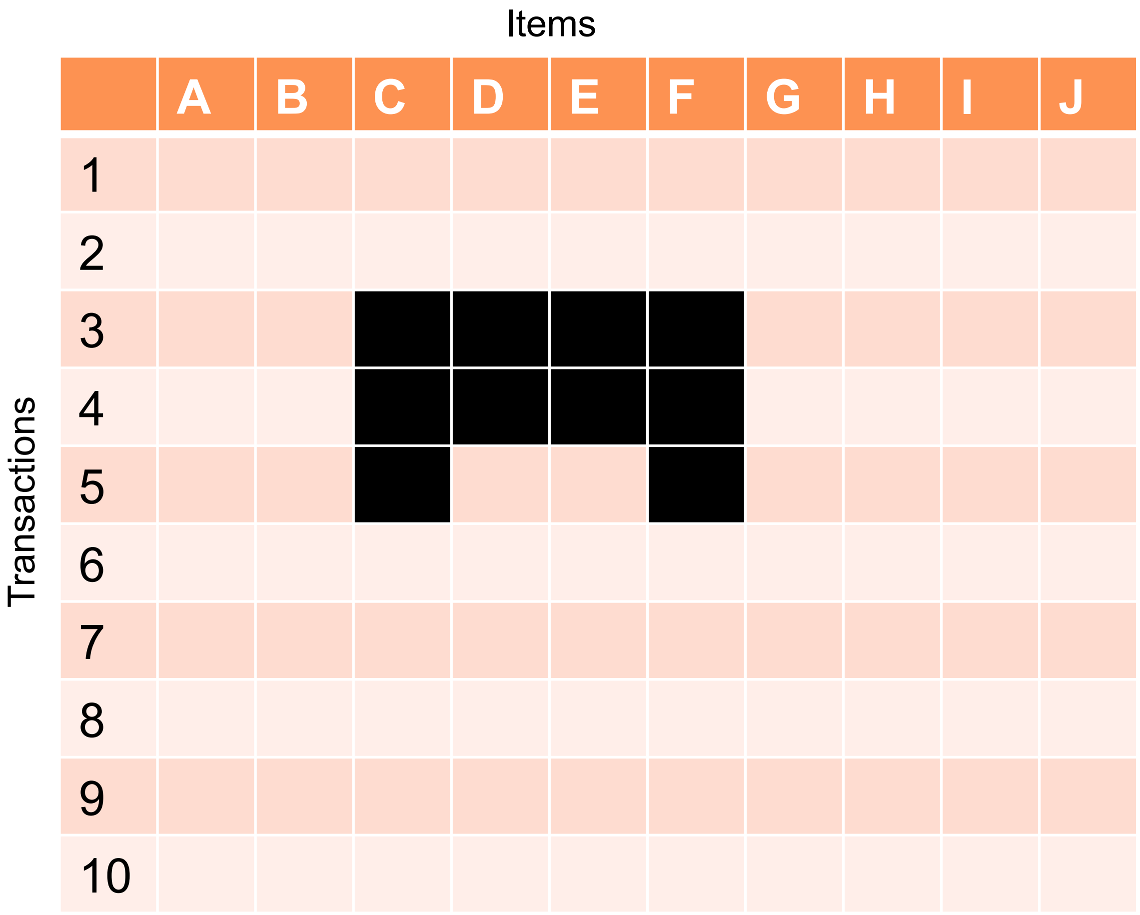
Itemsets	Support (counts)	Closed itemsets
{C}	3	
{D}	2	
{E}	2	
{C,D}	2	
{C,E}	2	
{D,E}	2	
{C,D,E}	2	

Esempio 2



Itemsets	Support (counts)	Closed itemsets
{C}	**3**	✓
{D}	2	
{E}	2	
{C,D}	2	
{C,E}	2	
{D,E}	2	
{C,D,E}	**2**	✓

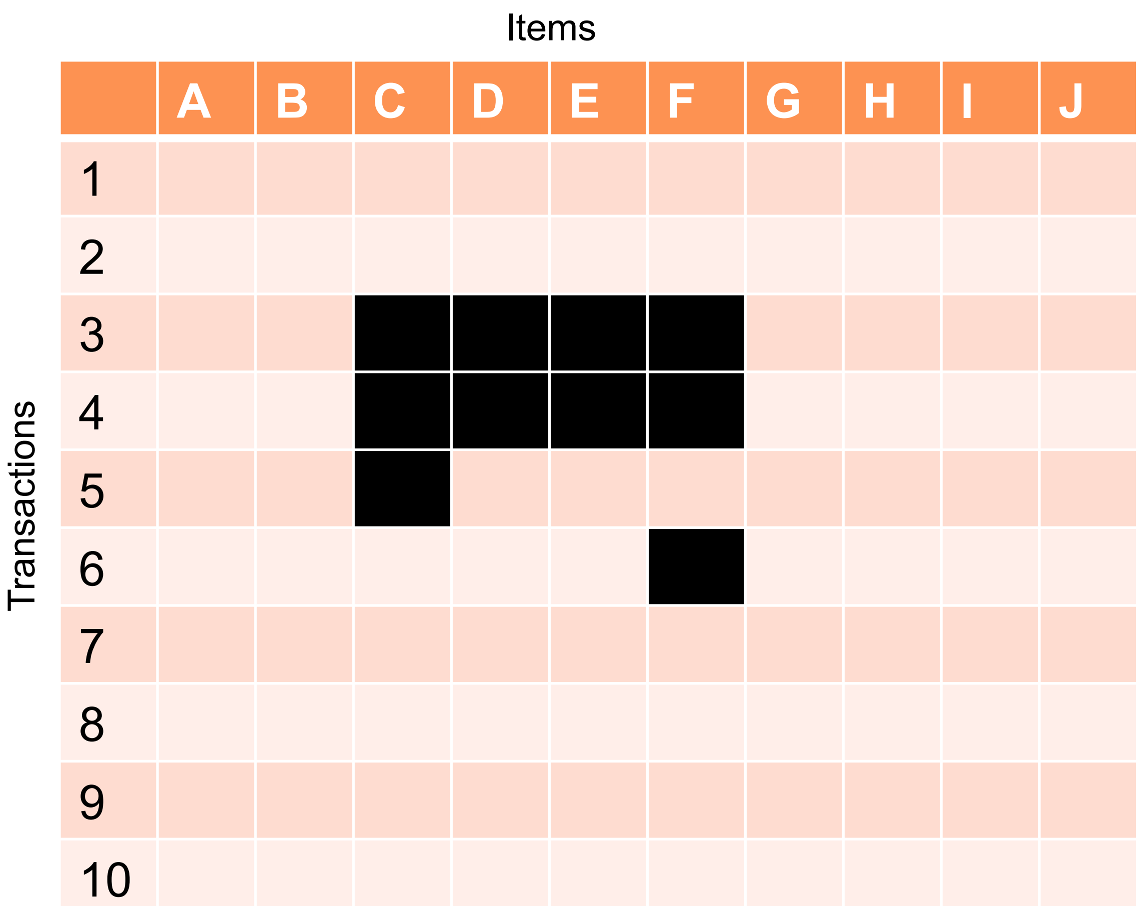
Esempio 3



Closed itemsets:

{C,D,E,F}, {C,F}

Esempio 4

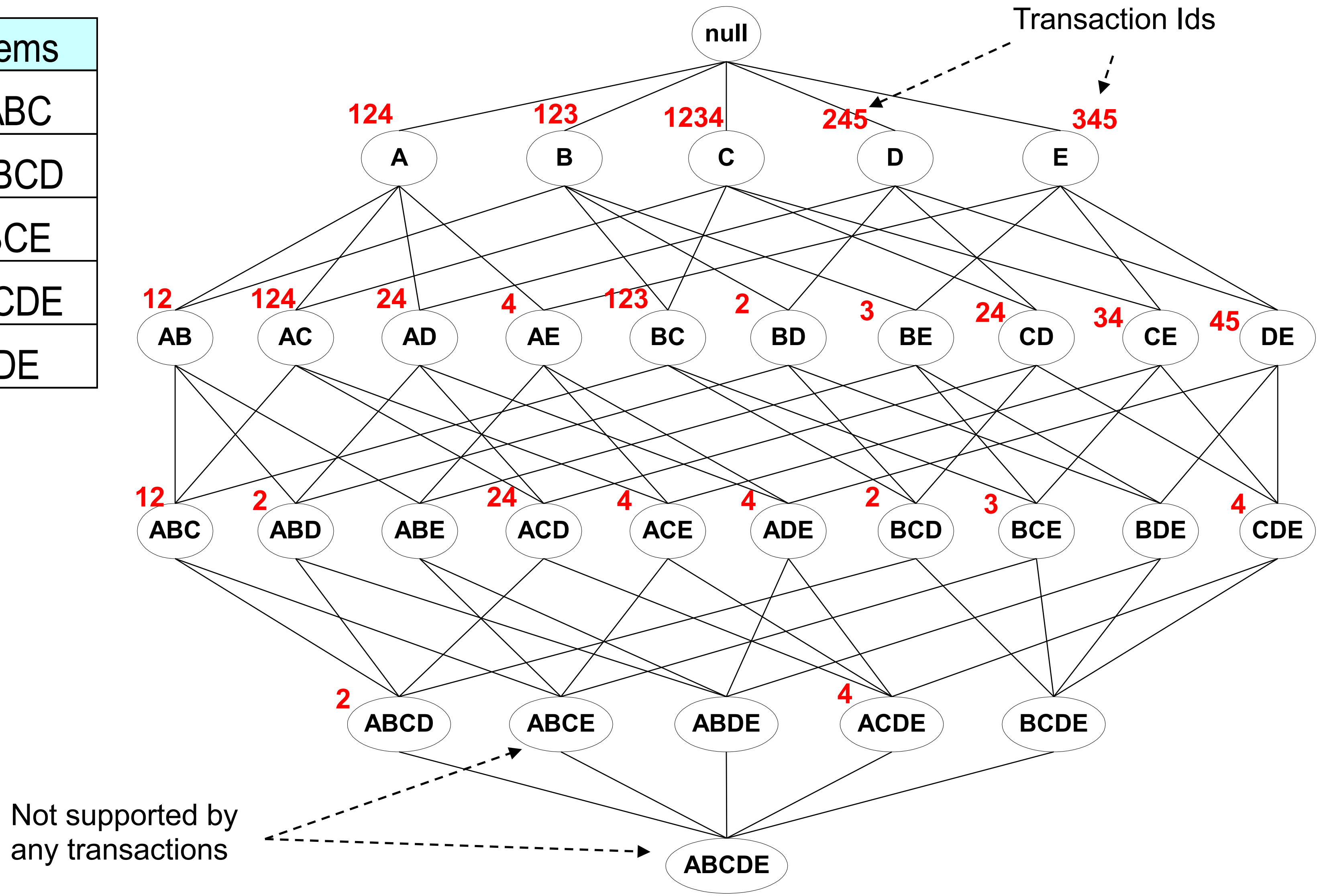


Closed itemsets:

{C,D,E,F}, {C}, {F}

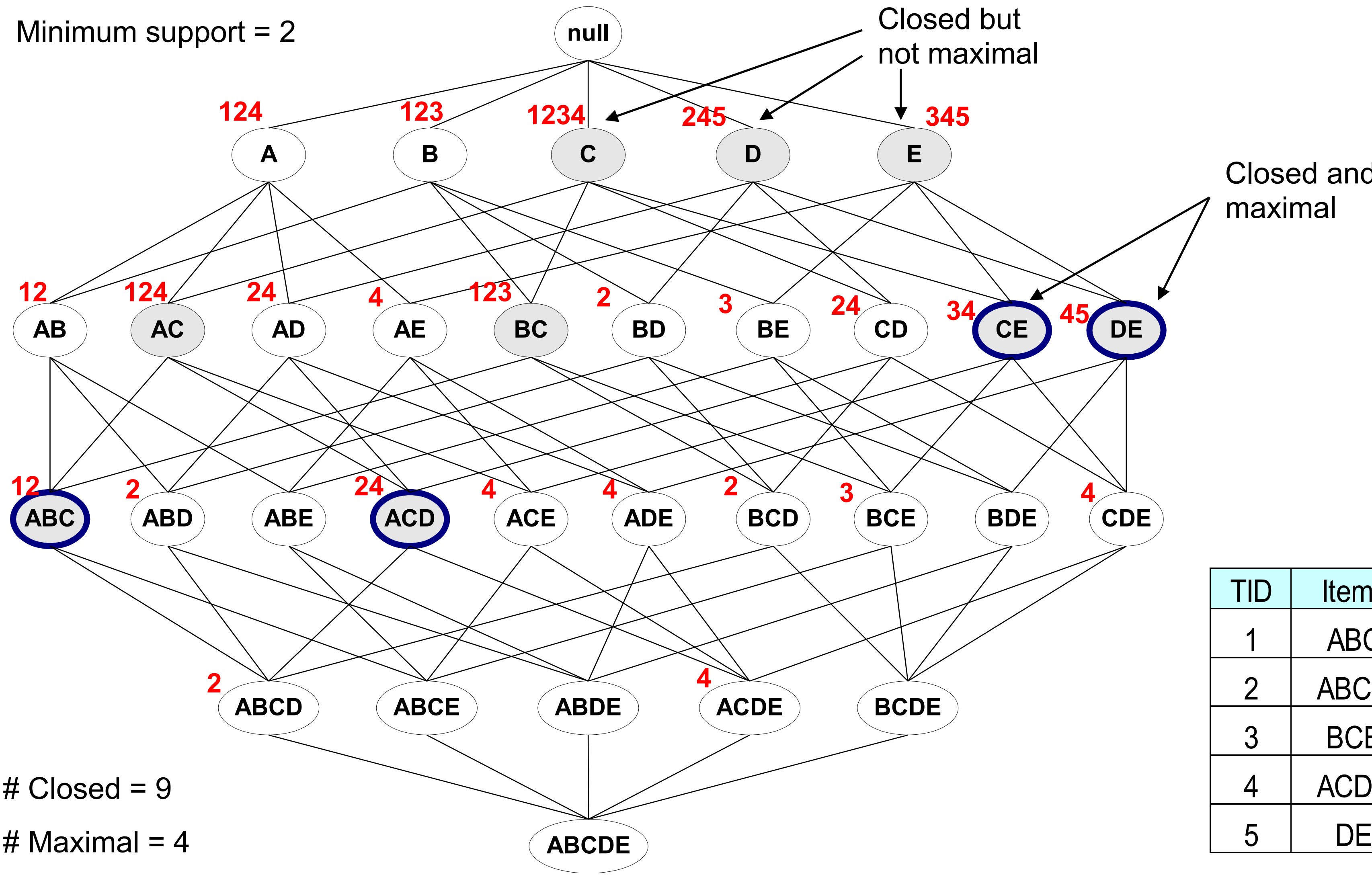
Itemset Frequenti Chiusi e Massimali

TID	Items
1	ABC
2	ABCD
3	BCE
4	ACDE
5	DE



Itemset Frequenti Chiusi e Massimali

Minimum support = 2

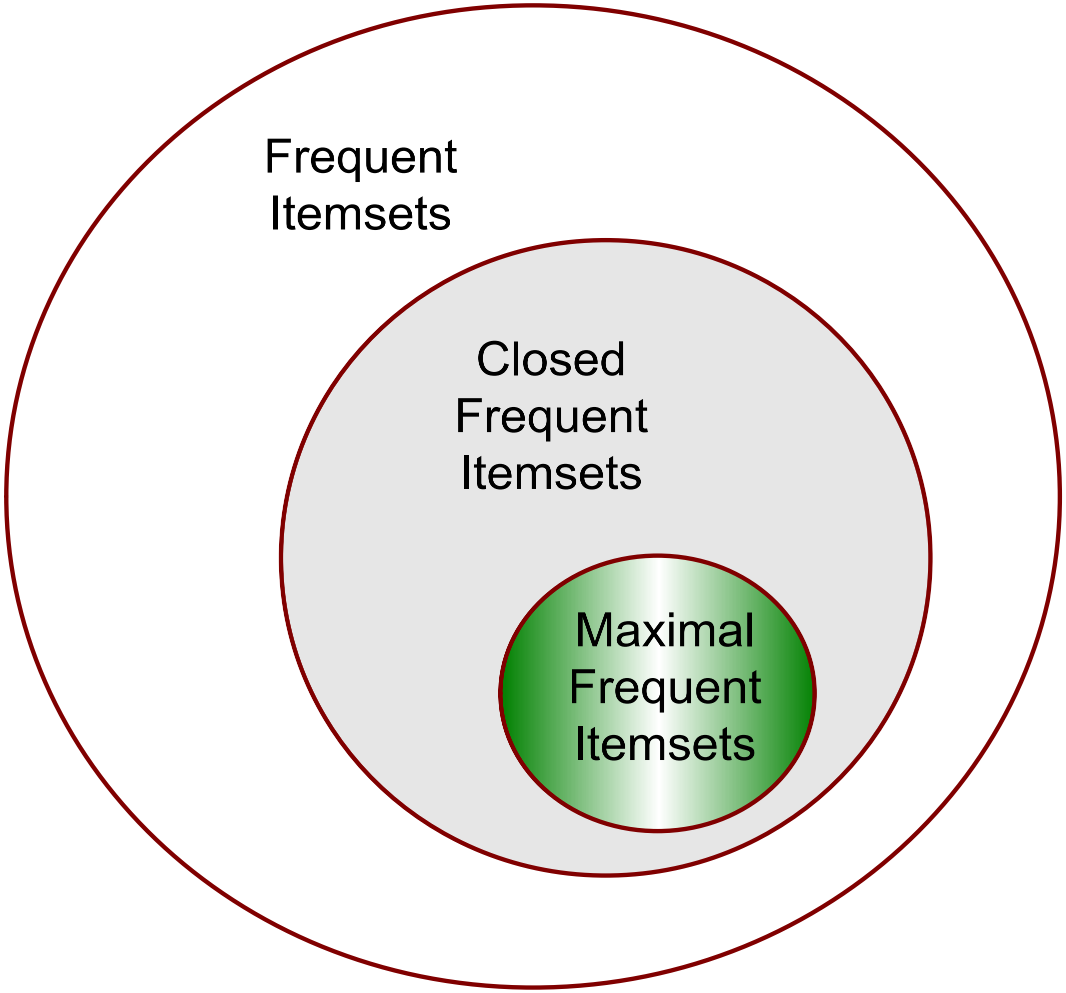


Closed = 9

Maximal = 4

TID	Items
1	ABC
2	ABCD
3	BCE
4	ACDE
5	DE

Itemset Chiusi e Massimali



Valutazione delle regole

- Gli algoritmi per le regole associative tendono a produrre molte regole *inutili*
  - ✓ Molte di esse non sono interessanti (ovvie), altre possono essere **ridondanti**
    - C'è ridondanza se $\{A,B,C\} \rightarrow \{D\}$ e $\{A,B\} \rightarrow \{D\}$ hanno lo stesso supporto e confidenza

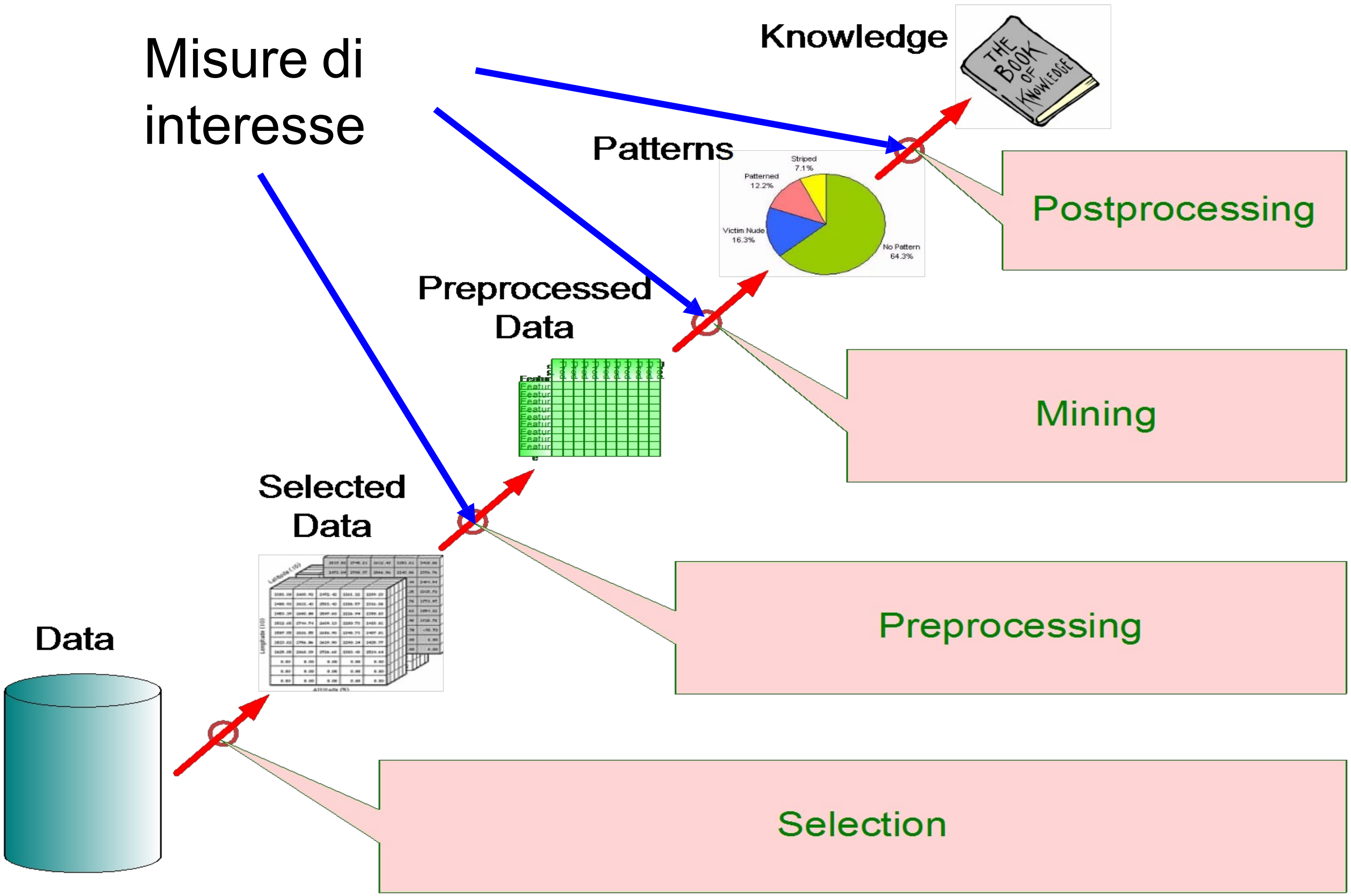
- L'utilizzo di **criteri/misure di interesse** possono permettere di eliminare/ordinare le regole associative fin qui costruite

- Si noti che sino a ora le uniche misure di interesse utilizzate sono il *supporto* e la *confidenza*

Misure di interesse

- **Misure oggettive:** danno priorità alle regole sulla base di criteri statistici calcolati a partire dai dati
  - ✓ Esistono molte formule a questo scopo, ognuna con i suoi pro e contro.
- **Misure soggettive:** danno priorità alle regole sulla base di criteri definiti dall'utente
  - ✓ Un pattern è interessante se contraddice le attese dell'utente
    - Un pattern è interessante se l'utente è interessato a svolgere qualche attività o prendere qualche decisione relativamente agli elementi che lo compongono.
    - In questo caso si dice che il pattern è *actionable* (Silberschatz & Tuzhilin).Per esempio se sono intenzionato a fare una campagna promozionale sulla birra, sarò particolarmente interessato a tutte le regole associative che includono la birra

Quando applicare le misure di interesse?



Dataset con supporto non omogeneo

- Molti dataset presentano gruppi di item con supporto molto elevato assieme ad altri con supporto molto limitato

- Esempio:
  - ✓ Una grande catena commerciale vende prodotti con prezzo da 1€ a 10.000€. Il numero di transazioni che includono prodotti con prezzo ridotto è di molto superiore a quelle con prezzo elevato. Tuttavia le associazioni tra questi ultimi sono di interesse per l'azienda.

- Difficile stabilire un opportuno valore di minsup

Dataset con supporto non omogeneo

- Fissare minsup per questi dataset può essere difficile
  - ✓ Una soglia troppo alta non permette di catturare le associazioni tra item con supporto limitato
  - ✓ Una soglia troppo bassa crea i seguenti problemi
    - Tempi di esecuzione elevati
    - Numero elevato di regole restituite (diverse delle quali non utili)
    - **Pattern cross-support**

- E' utile definire ulteriori **misure di interesse (interestin-gness measure)** per:
  - avere un'idea di quanto interessante è una regola, e
  - assegnare un rank o potare pattern

  Nella formulazione originale, il supporto e la confidenza sono le uniche misure utilizzate

Misure di interesse delle regole

- Data X → Y o {X,Y}, le informazioni necessarie per calcolare l'interesse possono essere derivate dalla **tabella di contigenza**

	Y	$\overline{\text{Y}}$	
X	s_{XY}	$s_{X\overline{Y}}$	s_X
$\overline{\text{X}}$	$s_{\overline{X}Y}$	$s_{\overline{X}\overline{Y}}$	$s_{\overline{X}}$
	s_Y	$s_{\overline{Y}}$	N

Tabella di

Contigenza

s_{XY}: *supporto di* X *e* Y
$s_{X\overline{Y}}$: *supporto di* X *e* $\overline{Y}$
$s_{\overline{X}Y}$: *supporto di* $\overline{X}$ *e* Y
$s_{\overline{X}\overline{Y}}$: *supporto di* $\overline{X}$ *e* $\overline{Y}$

Usata per definire diverse misure

supporto (s_{XY}/N), confidenza ($s_{XY}/\ s_X$), Gini, entropia, ecc.

Misura di confidenza

Custo	Tea	Coffee	…
C1	0	1	…
C2	1	0	…
C3	1	1	…
C4	1	0	…
…			

	$Coffee$	$\overline{Coffee}$	
Tea	15	5	20
$\overline{Tea}$	75	5	80
	90	10	100

Regola Associativa: Tea $\rightarrow$ Coffee

Confidenza $c(Tea \rightarrow Coffee) = P(Coffee \mid Tea) = \frac{15}{20} = 0.75$

Confidenza > 50%, cioé, tra le persone che bevono tè, è molto più probabile che bevano anche caffe (rispetto a quelle che bevono tè e non bevono caffe). **La regola sembra essere ragionevole**

Misura di confidenza

	$Coffee$	$\overline{Coffee}$	
Tea	15	5	20
$\overline{Tea}$	75	5	80
	90	10	100

Regola Associativa: Tea $\rightarrow$ Coffee

Confidenza $\boldsymbol{c(Tea \rightarrow Coffee) = P(Coffee \,|\, Tea) = \frac{15}{20} = 0.75}$

ma P(Coffee) = **0.9**, il che significa che una persona che beve tè ha una minore probabilità che beva caffe

Si noti anche che $\boldsymbol{c(\overline{Tea} \rightarrow Coffe) = P(Coffee \,|\overline{Tea}) = \frac{75}{80} = 0.9375}$

Quindi, **la confidenza non è una misura adeguata**

Misure per Regole Associative

- Che tipo di misura vogliamo?
  - La confidenza $c(X \rightarrow Y)$ dovrebbe essere sufficientemente alta per assicurare che una persona che compra X molto probabilmente comprerà Y
  - Nell'esempio precedente $c(tea \rightarrow coffee) = 0{,}75$, ma $s(coffee) = 0{,}9$

- **$c(X \rightarrow Y) > s(Y)$**
  - Altrimenti la regola potrebbe essere fuorviante poiché la presenza di X riduce la possibilità che anche Y sia presente nella stessa transazione

- Ci sono misure che catturano questo aspetto?
  - Si, molte!

Independenza Statistica

- La condizione

$$c(X \rightarrow Y) > s(Y)$$

  è equivalente a:
  - $P(Y|X) > P(Y)$
  - $P(X,Y) > P(X) \times P(Y)$

Se $P(X,Y) > P(X) \times P(Y)$: X & Y sono correlate positivamente

Se $P(X,Y) < P(X) \times P(Y)$: X & Y sono correlate negativamente

Misure che considerano l'indipendenza statistica

$$Lift = \frac{P(Y \mid X)}{P(Y)}$$

$$Interest = \frac{P(X,Y)}{P(X)P(Y)}$$

lift e usata per le regole

mentre

interest è usata per gli itemset

$$PS = P(X,Y) - P(X)P(Y)$$

$$\phi - coefficient = \frac{P(X,Y) - P(X)P(Y)}{\sqrt{P(X)[1-P(X)]P(Y)[1-P(Y)]}}$$

Exampio: Lift/Interest

	$Coffee$	$\overline{Coffee}$	
Tea	15	5	20
$\overline{Tea}$	75	5	80
	90	10	100

Association Rule: Tea $\rightarrow$ Coffee

Confidenza c(Tea→Coffee) = **P(Coffee|Tea) = 0.75**

ma **P(Coffee) = 0.9**

$\Rightarrow$ **Lift = 0.75/0.9= 0.8333** (< 1, quindi è negativamente associata)

$\Rightarrow$ **Interest = 0,15/(0,9*0,2) = 0,8333**

E' sufficiente usare interest/lift per il pruning?

MISURA Lift/Interest: Esempio

	Y	$\overline{Y}$	
X	10	0	10
$\overline{X}$	0	90	90
	10	90	100

	Y	$\overline{Y}$	
X	90	0	90
$\overline{X}$	0	10	10
	90	10	100

$$Lift = \frac{0.1}{(0.1)(0.1)} = 10$$

$$Lift = \frac{0.9}{(0.9)(0.9)} = 1.11$$

Statistical independence:

If P(X,Y) = P(X) P(Y) => Lift = 1

There are lots of measures proposed in the literature

#	Measure	Formula
1	ϕ-coefficient	$\frac{P(A,B)-P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$
2	Goodman-Kruskal's (λ)	$\frac{\sum_j \max_k P(A_j,B_k)+\sum_k \max_j P(A_j,B_k)-\max_j P(A_j)-\max_k P(B_k)}{2-\max_j P(A_j)-\max_k P(B_k)}$
3	Odds ratio (α)	$\frac{P(A,B)P(\overline{A},\overline{B})}{P(A,\overline{B})P(\overline{A},B)}$
4	Yule's Q	$\frac{P(A,B)P(\overline{AB})-P(A,\overline{B})P(\overline{A},B)}{P(A,B)P(\overline{AB})+P(A,\overline{B})P(\overline{A},B)} = \frac{\alpha-1}{\alpha+1}$
5	Yule's Y	$\frac{\sqrt{P(A,B)P(\overline{AB})}-\sqrt{P(A,\overline{B})P(\overline{A},B)}}{\sqrt{P(A,B)P(\overline{AB})}+\sqrt{P(A,\overline{B})P(\overline{A},B)}} = \frac{\sqrt{\alpha}-1}{\sqrt{\alpha}+1}$
6	Kappa (κ)	$\frac{P(A,B)+P(\overline{A},\overline{B})-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A)P(B)-P(\overline{A})P(\overline{B})}$
7	Mutual Information (M)	$\frac{\sum_i \sum_j P(A_i,B_j)\log\frac{P(A_i,B_j)}{P(A_i)P(B_j)}}{\min(-\sum_i P(A_i)\log P(A_i), -\sum_j P(B_j)\log P(B_j))}$
8	J-Measure (J)	$\max\Big(P(A,B)\log(\frac{P(B\|A)}{P(B)}) + P(A\overline{B})\log(\frac{P(\overline{B}\|A)}{P(\overline{B})}),$ $P(A,B)\log(\frac{P(A\|B)}{P(A)}) + P(\overline{A}B)\log(\frac{P(\overline{A}\|B)}{P(\overline{A})})\Big)$
9	Gini index (G)	$\max\Big(P(A)[P(B\|A)^2+P(\overline{B}\|A)^2]+P(\overline{A})[P(B\|\overline{A})^2+P(\overline{B}\|\overline{A})^2]$ $-P(B)^2-P(\overline{B})^2,$ $P(B)[P(A\|B)^2+P(\overline{A}\|B)^2]+P(\overline{B})[P(A\|\overline{B})^2+P(\overline{A}\|\overline{B})^2]$ $-P(A)^2-P(\overline{A})^2\Big)$
10	Support (s)	$P(A,B)$
11	Confidence (c)	$\max(P(B\|A), P(A\|B))$
12	Laplace (L)	$\max\Big(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2}\Big)$
13	Conviction (V)	$\max\Big(\frac{P(A)P(\overline{B})}{P(A\overline{B})}, \frac{P(B)P(\overline{A})}{P(B\overline{A})}\Big)$
14	Interest (I)	$\frac{P(A,B)}{P(A)P(B)}$
15	cosine (IS)	$\frac{P(A,B)}{\sqrt{P(A)P(B)}}$
16	Piatetsky-Shapiro's (PS)	$P(A,B)-P(A)P(B)$
17	Certainty factor (F)	$\max\Big(\frac{P(B\|A)-P(B)}{1-P(B)}, \frac{P(A\|B)-P(A)}{1-P(A)}\Big)$
18	Added Value (AV)	$\max(P(B\|A)-P(B), P(A\|B)-P(A))$
19	Collective strength (S)	$\frac{P(A,B)+P(\overline{AB})}{P(A)P(B)+P(\overline{A})P(\overline{B})} \times \frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{AB})}$
20	Jaccard (ζ)	$\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$
21	Klosgen (K)	$\sqrt{P(A,B)}\max(P(B\|A)-P(B), P(A\|B)-P(A))$

Misure a confronto: Esempio

10 esempi di tabelle di contigenza :

Example	s_{11}	s_{10}	s_{01}	s_{00}
E1	8123	83	424	1370
E2	8330	2	622	1046
E3	9481	94	127	298
E4	3954	3080	5	2961
E5	2886	1363	1320	4431
E6	1500	2000	500	6000
E7	4000	2000	1000	3000
E8	4000	2000	2000	2000
E9	1720	7121	5	1154
E10	61	2483	4	7452

Rankings degli esempi usando diverse misure

#	ϕ	λ	α	Q	Y	κ	M	J	G	s	c	L	V	I	IS	PS	F	AV	S	ζ	K
E1	1	1	3	3	3	1	2	2	1	3	5	5	4	6	2	2	4	6	1	2	5
E2	2	2	1	1	1	2	1	3	2	2	1	1	1	8	3	5	1	8	2	3	6
E3	3	3	4	4	4	3	3	8	7	1	4	4	6	10	1	8	6	10	3	1	10
E4	4	7	2	2	2	5	4	1	3	6	2	2	2	4	4	1	2	3	4	5	1
E5	5	4	8	8	8	4	7	5	4	7	9	9	9	3	6	3	9	4	5	6	3
E6	6	6	7	7	7	7	6	4	6	9	8	8	7	2	8	6	7	2	7	8	2
E7	7	5	9	9	9	6	8	6	5	4	7	7	8	5	5	4	8	5	6	4	4
E8	8	9	10	10	10	8	10	10	8	4	10	10	10	9	7	7	10	9	8	7	9
E9	9	9	5	5	5	9	9	7	9	8	3	3	3	7	9	9	3	7	9	9	8
E10	10	8	6	6	6	10	5	9	10	10	6	6	5	1	10	10	5	1	10	10	7

Proprietà - Permutazione di variabili

	B	$\overline{B}$
A	p	q
$\overline{A}$	r	s

	A	$\overline{A}$
B	p	r
$\overline{B}$	q	s

M(A,B) = M(B,A)?

Misure simmetriche:

- support, lift, collective strength, cosine, Jaccard, etc

Misure non simmetriche:

- confidence, conviction, Laplace, J-measure, etc

Proprietà - Variazione di scala

Esempio Livello-Sesso (Mosteller, 1968):

	Female	Male	
High	2	3	5
Low	1	4	5
	3	7	10

	Female	Male	
High	4	30	34
Low	2	40	42
	6	70	76

2x　　10x

Mosteller:

- L'associazione sottostante deve essere indipendente dal numero relativo di studenti maschi e femmine nei campioni

Esempio: ϕ-Coefficient

- ϕ-coefficient è analogo al coefficiente di correlazione per variabili continue

	Y	$\overline{Y}$	
X	60	10	70
$\overline{X}$	10	20	30
	70	30	100

	Y	$\overline{Y}$	
X	20	10	30
$\overline{X}$	10	60	70
	30	70	100

$$\phi = \frac{0.6 - 0.7 \times 0.7}{\sqrt{0.7 \times 0.3 \times 0.7 \times 0.3}} = 0.5238$$

$$\phi = \frac{0.2 - 0.3 \times 0.3}{\sqrt{0.7 \times 0.3 \times 0.7 \times 0.3}} = 0.5238$$

ϕ Coefficient è lo stesso oer entrambe le tabelle

Proprietà - Addizione di casi nulli

	B	$\overline{B}$
A	p	q
$\overline{A}$	r	s

	B	$\overline{B}$
A	p	q
$\overline{A}$	r	s + k

Misure invarianti:

- support, cosine, Jaccard, etc

Misure non-invarianti:

- correlation, Gini, mutual information, odds ratio, etc

Proprietà di alcune misure

Symbol	Measure	Inversion	Null Addition	Scaling
ϕ	ϕ-coefficient	Yes	No	No
α	odds ratio	Yes	No	Yes
κ	Cohen's	Yes	No	No
I	Interest	No	No	No
IS	Cosine	No	Yes	No
PS	Piatetsky-Shapiro's	Yes	No	No
S	Collective strength	Yes	No	No
ζ	Jaccard	No	Yes	No
h	All-confidence	No	No	No
s	Support	No	No	No

Paradosso di Simpson

Buy HDTV	Buy Exercise Machine		
	Yes	No	
Yes	99	81	180
No	54	66	120
	153	147	300

$$c(\{\text{HDTV} = \text{Yes}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 99/180 = 55\%$$

$$c(\{\text{HDTV} = \text{No}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 54/120 = 45\%$$

➔ Clienti che acquistano *HDTV* molto probabilmente compreranno *exercise machines*

Paradosso di Simpson

Customer Group	Buy HDTV	Buy Exercise Machine		Total
		Yes	No	
College Students	Yes	1	9	10
	No	4	30	34
Working Adult	Yes	98	72	170
	No	50	36	86

College students:

$$c(\{\text{HDTV} = \text{Yes}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 1/10 = 10\%$$

$$c(\{\text{HDTV} = \text{No}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 4/34 = 11.8\%$$

Working adults:

$$c(\{\text{HDTV} = \text{Yes}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 98/170 = 57.7\%$$

$$c(\{\text{HDTV} = \text{No}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 50/86 = 58.1\%$$

Simpson’s Paradox

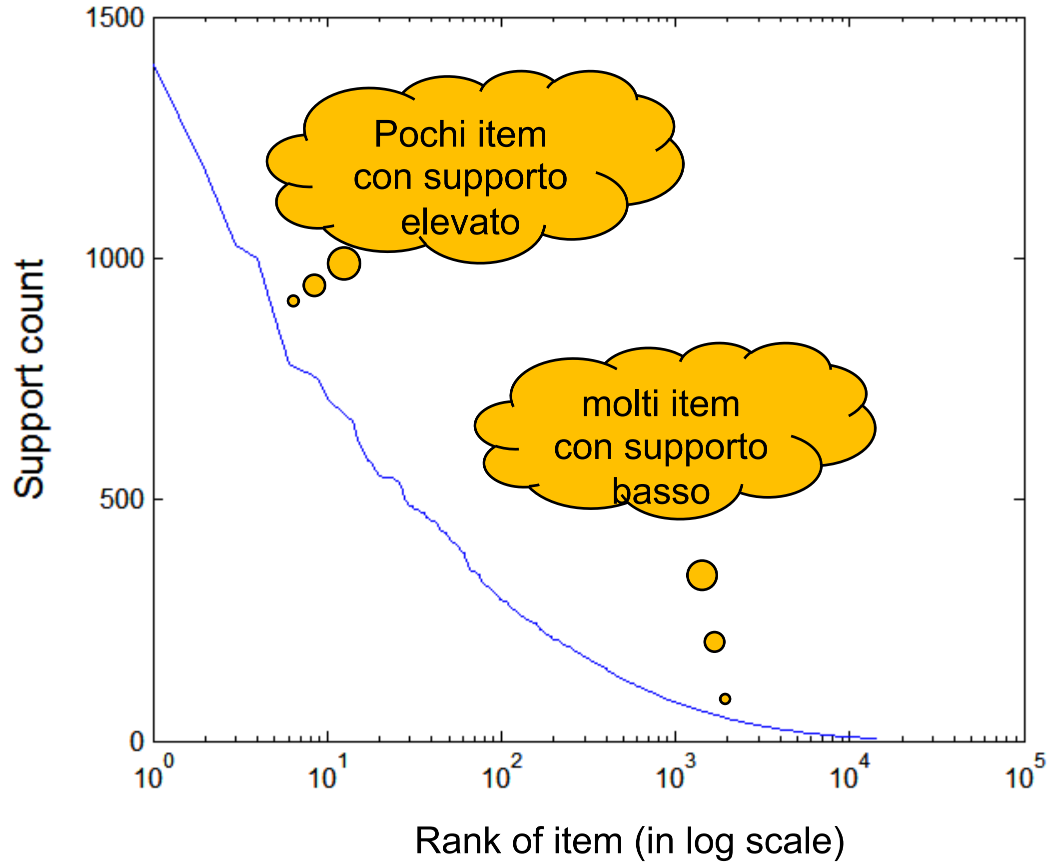
- Alcune relazioni osservate possono essere influenzate da altri fattori (variabili nascoste) che creano confusione
  - Le variabili nascoste possono causare che alcune relazioni scompaiono o invertono la dipendenza

- E’ opportuno fissare una “stratificazione” per evitare la generazione di pattern anomali

Effetti della distribuzione del supporto

- In molti dataset reali la distribuzione del supporto è distorta

Difficile stabilire **minsup**:
- Elevato ➔Perdita di rari, ma interes-santi itemset (e.g., {caviale, vodka})
- Basso ➔ calcolo costoso con un numero di itemset (troppo) elevato



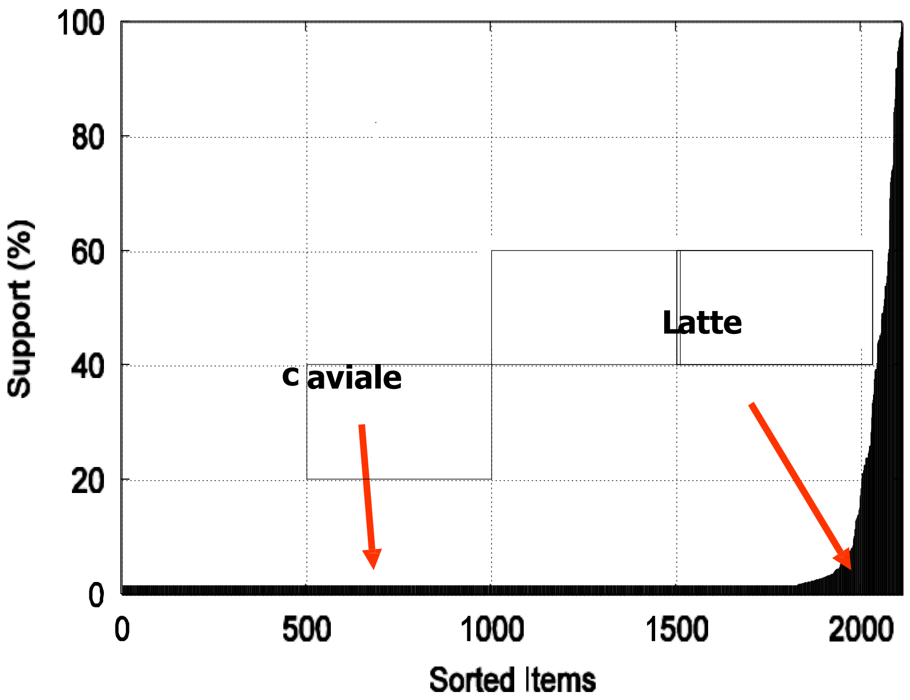
Dataset con supporto non omogeneo

- Fissando la soglia *thr*=0.15 gli temset {*p*,*q*,*r*}, {*p*,*r*}, {*p*,*q*} risultano essere cross-support
- Le regole associative relative sono di scarso interesse anche se presentano confidenze elevate
  - ✓ c({*q*} → {*p*})=4/5=80%
- Sebbene tali pattern possano essere eliminati con un valore elevato per minsup (es. minsup=20%) il rischio è di perdere pattern che determinano regole di maggiore interesse
  - ✓ s({*p*,*q*})=4/30=13.3%
  - ✓ s({*q*,*r*})=5/30=16.7%
- In questa situazione supporto e confidenza non catturano adeguatamente la correlazione/affinità tra tutti gli elementi dell'itemset
  - ✓ c({*p*} →{*q*})=4/25=16% ha una confidenza molto bassa

p	q	r
0	1	1
1	1	1
1	1	1
1	1	1
1	1	1
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Cross-Support Patterns

La misura di supporto potrebbe far perdere pattern rari ma interessanti (e.g. { caviale, vodka })
Sarebbe opportuno anche avere una misura per pattern rari (cross support)



Tuttavia, è possible trovare anche pattern con grado di supporto diversi (**cross-support pattern**):

- Esempio: {caviale, latte}

Come evitare tali pattern?

Cross Support

- Dato un itemset, $X = \{x_1, x_2, \dots, x_d\}$, con d items, possiamo definire una misura di **cross support**, r, per X

$$r(X) = \frac{\mathbf{min}\{s(x_1), s(x_2), \dots, s(x_d)\}}{\mathbf{max}\{s(x_1), s(x_2), \dots, s(x_d)\}}$$

dove $s(x_i)$ è il supporto dell'item x_i

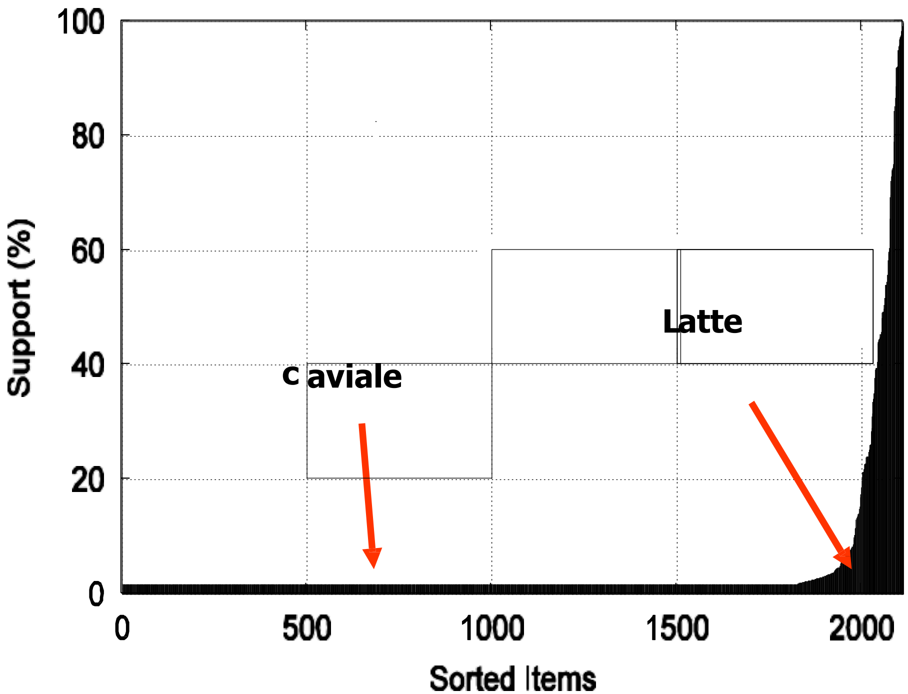
  – Possiamo usare $r(X)$ per potare cross support patterns, ma non possiamo evitarli

H-Confidence

- Per evitare pattern i cui item hanno supporti molto diversi, è utile definire una nuova misura per la loro valutazione (**h-confidence** o **all-confidence)**

- Specificatamente, dato un itemset $X = \{x_1, x_2, \dots, x_d\}$
  - *h-confidence* è la confidenza minima di tutte le regole associative derivate da X
  - **hconf(X) = min{ c($X_1 \rightarrow X_2$) | $X_1 \subset X$ e $X_2 = X - X_1$ },**

Cross-Support Pattern



Calcolo della confidenza

Si noti che: **c(caviar→milk)** è elevata,

mentre **c(milk→caviar)** è molto bassa

Quindi, **min(c(caviar→milk), c(milk→caviar))**

è anche molto basso

H-Confidence ...

- Dato un itemset $X = \{x_1, x_2, \dots, x_d\}$ qual'è la regola con confidenza minima che possiamo ottenere da X?

  Si ricordi che
  - c($X_1 \rightarrow X_2$) = s($X_1 \cup X_2$) / s(X_1)
  - Il numeratore è costante: s($X_1 \cup X_2$) = s(X)
  - Poiché hconf(X) = min{ c($X_1 \rightarrow X_2$) | $X_1 \subset X$ *e* $X_2 = X - X_1$}
  - Per trovare la regola con confidenza minima, dobbiamo trovare la variabile X_1 con supporto massimo
  - Si considerano solo regole dove X_1 è un singoletto, i.e., $\{x_1\} \rightarrow X - \{x_1\}$, $\{x_2\} \rightarrow X - \{x_2\}$, …, or $\{x_d\} \rightarrow X - \{x_d\}$

$$\text{hconf}(X) = \min\left\{\frac{s(X)}{s(x_1)}, \frac{s(X)}{s(x_2)}, \dots, \frac{s(X)}{s(x_d)}\right\} = \frac{s(X)}{\max\{s(x_1), s(x_2), \dots, s(x_d)\}}$$

Cross-Support e H-confidence

- Per la proprietà di anti-monotonicità del supporto

$$s(X) \leq \min\{s(x_1), s(x_2), \dots, s(x_d)\}$$

- Quindi, possiamo derivare la seguente relazione tra h-confidence and cross-support di un itemset

$$\begin{aligned}\mathrm{hconf}(X) &= \frac{s(X)}{\max\{s(x_1), s(x_2), \dots, s(x_d)\}} \\ &\leq \frac{\min\{s(x_1), s(x_2), \dots, s(x_d)\}}{\max\{s(x_1), s(x_2), \dots, s(x_d)\}} \\ &= r(X)\end{aligned}$$

Quindi, $\mathbf{hconf}(\boldsymbol{X}) \leq \boldsymbol{r}(\boldsymbol{X})$

Cross-Support e h-confidence …

- Poiché, $\mathrm{hconf}(X) \leq r(X)$, possiamo eliminare i pattern cross con *h-confidence* < h_c, (soglia fissata dall'utente)

- Si noti che

$$0 \leq \mathrm{hconf}(X) \leq r(X) \leq 1$$

- Ciascun itemset con una *h-confidence* che superi la soglia h_c, è detto **hyperclique**
- *h-confidence* può essere usata in sostituzione o in congiunzione con il supporto.

Proprietà degli Hyperclique

- Gli hyperclique sono itemset, ma non necessaria-mente itemset frequenti
  - Utilizzati per trovare pattern con basso supporto

- **H-confidence è anti-monotona**

- E' possible definire **hypercliques chiusi** e **massimali** in termini di h-confidence
  - Un hyperclique X è chiuso se nessuno dei suoi immediati superset ha la stessa h-confidence di X
  - Un hyperclique X è massimale se nessuno dei suoi immediati superset è un hyperclique.

Proprietà e applicazioni degli Hyperclique

- Vantaggi derivanti dall'uso di h-confidence:

1. Una elevata *h-confidence* implica una **stretta relazione tra tutti gli articoli del pattern**
2. **Eliminazione dei cross-support pattern** come {caviale, latte}
3. Pattern con basso supporto e elevata h-confidence possono essere identificati in modo efficiente

- Usati per trovare gruppi fortemente coerenti di item anche in contesti diversi
  – Parole che si occorrono insieme nei documenti
  – Proteine in una rete di interazione proteica