Mineritul regulilor de asociere

1 Concepte de bază

Pattern-urile frecvente sunt acele mulţimi de itemuri, subsecvente sau substructuri care apar în mode frecvent în setul de date. O subsecventă se referă la o anumită ordine a înregistrărilor din baza de date iar o substructură se referă la diferite forme structurale ale bazei de date.

În general mineritul regulilor de asociere poate fi văzut ca un proces care se face în 2 paşi:

1. **Găsirea tuturor itemseturilor frecvente –**fiecare dintre aceste itemseturi vor apărea ca şi frecvenţă de cel puțin valoare predeterminată pentru suportul minim (pragul minim de apariţie);

2. **Generarea regulilor de asociere puternice din itemseturile frecvente -** aceste reguli trebuie să satisfacă suportul minim şi încrederea minimă.

În plus măsurarea gradului de interes poate fi aplicată pentru descoperirea de legături de corelare între itemurile asociate. Toate performanţele regulilor de asociere sunt determinate în primul pas.

1. Mineritul pattern-urilor frecvente

Mineritul patternurilor frecvente pot fi clasificate în diferite feluri, bazându-ne pe următoarele criterii:

- **Bazat pe complexitatea regulilor care voi fi extrase** - putem procesa setul complet de articole (itemseturi) frecvente, setul restrâns de articole şi setul de articole frecvente de dimensiune maximă specificându-se un prag de suport minim.

- **Bazat pe gradul de abstractizare implicat în setul de reguli** – unele metode de minerit a regulilor de asociere pot găsi reguli la diferite nivele de abstractizare. Ne referim la setul de reguli de minerit ca fiind compus din **reguli de asociere pe mai multe nivele.** Dacă în schimb regulile pentru un set dat nu fac referire la articole sau atribute pe diferite nivele de abstractizare atunci setul conţine **reguli de asociere pe un singur nivel**.

- **Bazat pe numărul de dimensiuni al datelor implicate în regulă –** dacă în regula de asociere articolul sau atributul referă doar o singură dimensiune se

numesc **reguli de asociere unidimensionale** (ex.: buys(X,"computer") => buys(X,"antivirus_software"), altfel ele se numesc **reguli de asociere multidimensionale** (ca de exemplu) $age(X,"30...39") \wedge income(X,"42K...48K") \Rightarrow buys(X, high\_resolution\_TV)$

- **bazate pe tipurile de valori cu care se lucrează în regulă -** dacă regula implică asocierile dintre prezenţa şi absenţa articolelor, aceasta este **regla de asociere booleană**. Asocierile descoperite pot fi analizate ulterior pentru a descoperi corelaţiile statistice ducând la **regulile de corelaţie.** Dacă regula descrie o asociere între cantitatea articolelor sau atributelor, aceasta este o **regulă de asociere cantitativă**. În aceste reguli, valorile cantitative pentru articole sau atribute sunt partiţionate în intervale.
- **Bazat pe tipurile de patternuri care urmează să fie minerite –** mineritul asocierii poate fi extins la analiza corelaţiilor, unde se poate identifica absenţa sau prezenţa articolelor şi eventuale corelaţii care există între articolele respective.

1.1 Algoritmul APRIORI pentru extragerea mulţimilor de item-uri frecvente

Este un algoritm propus de R. Agrawal şi R. Srikant în 1994 pentru mineritul itemseturilor frecvente pentru regulile de asociere booleene. Numele vine din faptul că algoritmul utilizează *cunoştinţe apriori* despre proprietățile itemseturilor frecvente. Proprietatea APRIORI este *„Toate subseturile nevide ale unui itemset frecvent trebuie de asemenea să fie frecvente"*.

Proprietatea apriori se bazează pe următoarea observaţie: dacă un itemset *I* nu satisface pragul de suport minim atunci itemsetul *I* nu este frecvent. Dacă un item A este adăugat la itemsetul *I* atunci itemsetul rezultat nu poate apare mai frecvent decât *I*. Aşadar, mulţimea $I \cup A$ nu este nici ea mai frecventă. Această proprietate aparţine unei categorii speciale de proprietăţi numite **antimonotone** în sensul că dacă un set nu poate trece un anumit test, toate superseturile vor pica acel test de asemenea.

Algoritmul Apriori este format dintr-o succesiune de etape fiecare etapă fiind formată din 2 acţiuni, o acţiune de alăturare a itemurilor şi o acţiune de eliminare a itemseturilor. În acţiunea de alăturare (join) a itemurilor un set de candidaţi este generat prin alăturarea membrilor de pe nivelul imediat anterior. În acţiunea de eliminare (prune) itemseturile rezultate sunt verificate dacă respectă pragul pentru suportul minim şi sunt eliminate acele itemseturi care nu îndeplinesc pragul minim de număr de apariţii.

Considerăm următorul exemplu în care avem o tabelă cu lista de tranzacţii dintr-o bază de date D.

TID	List of item_IDs
T100	I1,I2,I5
T200	I2,I4
T300	I2,I3
T400	I1,I2,I4
T500	I1,I3
T600	I2,I3
T700	I1,I3
T800	I1,I2,I3,I5
T900	I1,I2,I3

|D|=9 deoarece sunt 9 tranzacţii în B

1.2 Paşi pentru algoritmul Apriori

1. La prima iteraţie a algoritmului fiecare item este un membru din setul de candidaţi C1 de 1-itemset. Algoritmul numără apariţii pentru fiecare item

2. Presupunem că pragul pentru suportul minim de tranzacţie necesar numărării este 2. setul de 1-itemseturi frecvente poate fi determinat. Considerăm doar candidaţii care satisfac suportul minim.

3. Dezvoltăm setul de 2-itemseturi frecvente, L2, algoritmul utilizează L1xL1 pentru generarea candidaţilor, C2

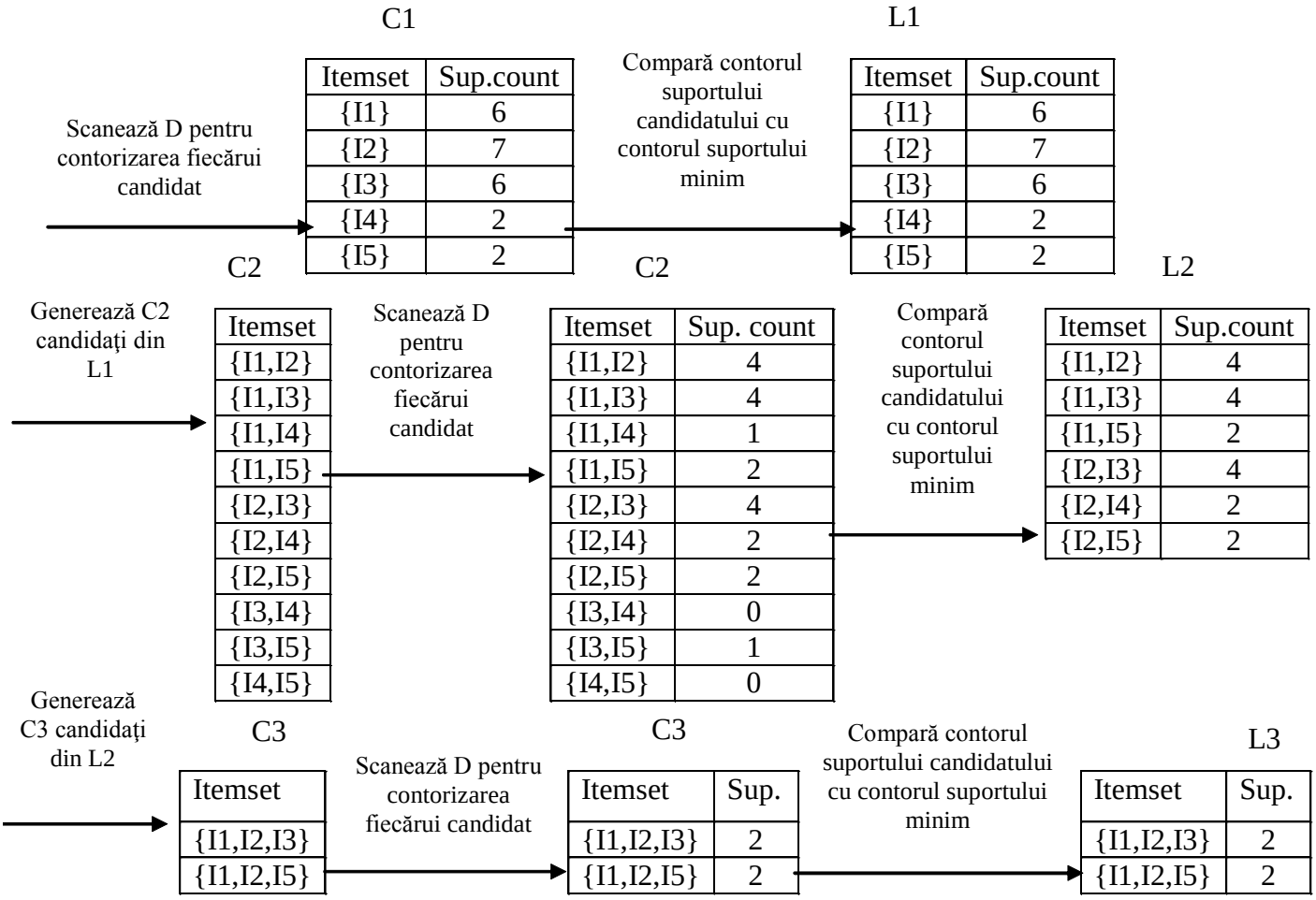
4. Sunt scanate tranzacţiile şi contorizat numărul de apariţii pentru fiecare itemset candidat,

5. Este determinat 2-itemseturile frecvente L2, considerând doar acei candidaţi care au suportul minim.

6. Se generează lista de 3-itemseturi frecvente. C3=L2xL2. Bazându-ne pe proprietatea Apriori în care toate sub-seturile de itemseturi frecvente trebuie să fie de asemenea frecvente, putem determina că candidaţi de 4 nu este posibil să fie frecvenţi. Observăm că pentru k-candidaţi trebuie să verificăm k-1 subseturi frecvente.

7. Baza D este scanată pentru a determina L3

8. Algoritmul generează L3xL3 candidaţi de 4-itemseturi. C4 {I1,I2,I3,I5} este tăiat devreme ce subsetul din el nu este frecvent , C4 este mulţimea vidă şi algoritmul se termină, găsind toate itemseturile frecvente.



Exemplul cu baza de date D prezentat detaliat pe fiecare etapă în parte

2 Măsurarea gradului de interes

2.1 Consideraţii generale

Măsurarea gradului de interes este utilizată pentru a separa modelele (patternurile) neinteresante de cele interesante considerate cunoştinţe. Este utilizată pentru a ghida procesul de data mining sau pentru a evalua modelele descoperite. Aceste măsuri pot reduce substanţial numărul de modele, de obicei doar o mică parte din modelele descoperite vor fi într-adevăr interesante pentru un utilizator. Există câteva măsuri obiective pentru a calcula gradul de interes pentru modele. Unele se bazează pe structura modelelor descoperite iar altele se bazează pe statistică. În general, fiecărei măsuri îi este asociat un prag care poate fi controlat de către utilizator.

Fie $I = \{i1, i2, ..., im\}$ un set de (itemi) articole, iar D taskul de date relevant care este un set de tranzacţii din baza de date unde fiecare tranzacţie *T* este un set de itemuri (articole) astfel încât $T \subseteq I$. Fiecărei tranzacţii îi este asociat un identificator numit TID. Fie *A* un set de articole. Tranzacţia *T* se spune că conţine *A* dacă şi numai dacă $A \subseteq T$. Regula de asociere este o implicaţie de forma $A \Rightarrow B, unde\ A \subset I,\ B \subset I,\ si\ A \cap B = \phi$. Regula $A \Rightarrow B$ păstrează tranzacţiile din setul *D* care **suportă *s***, unde s este procentajul tranzacţiei din D care conţine $A \cup B$. Acesta este notată cu probabilitatea $P(A \cup B)$. Regula $A \Rightarrow B$ reprezintă **încrederea *c*** în tranzacţiile din setul *D* iar *c* este procentajul tranzacţiilor din D care conţin *A* şi *B*. Aceasta este notată ca fiind probabilitatea *P(B/A)*.

2.2 Măsuri ale gradului de interes sunt:

- *Simplitatea* - aceasta poate fi văzută ca funcţie de structura modelului, definită în termeni de dimensiune în biţi a modelului sau ca număr de atribute sau operaţii care se aplică în model.

- *Certitudinea* – unde fiecare model descoperit are asociată o măsură a certitudini care evaluează validitatea sau „credibilitatea" modelului. O măsură a certitudini pentru regula „A=>B", unde A şi B sunt mulţimi de itemuri este încrederea (confidence)

$$\text{confidence}(A \Rightarrow B) = \frac{\#\_tuples\_containing\_both\_A\_and\_B}{\#\_tuples\_containing\_A} \quad \textbf{(1)}$$

- *Utilitatea* - măsoară gradul potenţial de utilizare sau gradul de interes pentru un model. Acesta poate fi estimat de o funcţie de utilitate cum ar fi „suportul" (support). Suportul asociat unui model referă procentul de înregistrări (tupluri) din task-uri relevante pentru care modelul este adevărat.

$$\text{support}(A \Rightarrow B) = \frac{\#\_tuples\_containing\_both\_A\_and\_B}{total\_\#\_of\_tuples} \quad \textbf{(2)}$$

De îndată ce itemurile frecvente din tranzacţii au fost găsite, este simplu de a genera reguli puternice din acestea (reguli care satisfac atât suportul minim cât şi încrederea minimă) şi se pot lua în considerare doar mulţimile de itemuri frecvente care satisfac pragul minim de încredere. Aceasta poate fi făcută utilizând formulele pentru confidenţă şi suport prezentate mai sus. Pentru fiecare mulţime de itemuri sunt generate toate submulţimile nevide şi pentru fiecare subset nevid sunt generate regulile care îndeplinesc pragul de încredere minim. De vreme ce regulile sunt generate din mulţimile de itemuri frecvente fiecare în parte va satisface în mode automat pragul de suport minim. Ca si exemplu presupunem că datele conţin

mulţimea de itemmuri frecvente L={I1, I2, I5}. Toate subseturile nevide ale lui L sunt {I1,I2}, {I1,I5}, {I2,I5}, {I1}, {I2}, şi {I5}. Regulile de asociere rezultate sunt:

$$I1 \wedge I2 \Rightarrow I5, \quad confidence = 2/4 = 50\%$$

$$I1 \wedge I5 \Rightarrow I2, \quad confidence = 2/2 = 100\%$$

$$I2 \wedge I5 \Rightarrow I1, \quad confidence = 2/2 = 100\%$$

$$I1 \Rightarrow I2 \wedge I5, \quad confidence = 2/6 = 33\%$$

$$I2 \Rightarrow I1 \wedge I5, \quad confidence = 2/7 = 29\%$$

$$I5 \Rightarrow I1 \wedge I2, \quad confidence = 2/2 = 100\%$$

Dacă pragul de încredere minim este, să zicem, 70%, atunci doar ce-a de-a doua, a treia şi ultima regulă satisfac pragul şi sunt prezentate în continuare ca fiind reguli puternice.