

TEHNICI DE EXPLORARE A DATELOR -DATA MINING -

CURS 3

Factori de influența a rezultatelor KDD

- Rezultatul procesului de KDD este puternic influențat de :
 - volumul mare de date și problemele legate de scalabilitatea metodelor de data mining
 - natura dinamică a datelor
 - probleme legate de calitatea datelor

Volumul de date de prelucrat

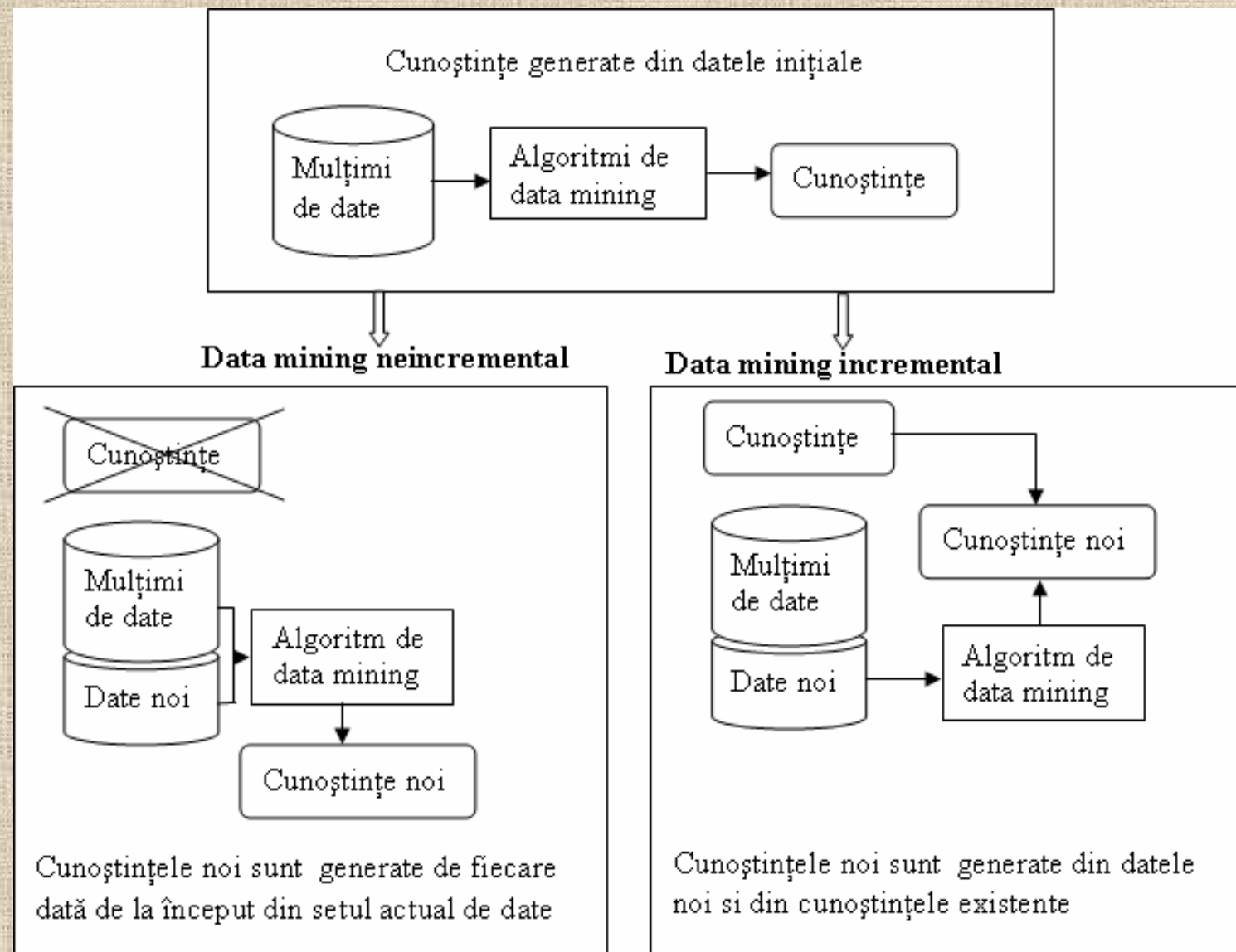
- Principala caracteristică a sistemelor de KDD – abilitatea de a manipula masive de date
 - Solicita utilizarea de metode și algoritmi scalabili
- Pentru fiecare metoda de data mining este necesară evaluarea sensibilității la dimensiunea datelor, tradusă prin timpul necesar pentru procesarea datelor
- Trebuie considerate 3 aspecte:
 - numărul obiectelor, care poate varia de la câteva sute la câteva milioane
 - numărul de caracteristici – de la câteva la câteva mii
 - numărul valorilor unei caracteristici de la 2 la câteva milioane

Tehnici de îmbunătățire a scalabilității

- Tehnici care **cresc viteza algoritmilor**
 - Euristice – simplifică procesarea datelor (ex. La regulile de asociere se vor lua in considerare numai acele reguli care nu depasesc o anumita dimensiune)
 - Optimizarea algoritmilor prin utilizarea structurilor de date eficiente (vectori de biti, tabele hash, arbori de cautare binari) pentru a stoca si manipula date
 - Paralelizarea – distribuie operatiile de prelucrare a datelor mai multor procesoare care lucreaza in paralel
- Tehnici care **partitioneaza multimile de date**
 - Se reduce dimensiunea multimii de intrare prin reducerea numarului de obiecte, caracteristici, valori per caracteristica si procesatea secventiala sau paralela a datelor divizate in submultimi
 - Esantionare – se utilizeaza doar o submultime *reprezentativa* de obiecte sau caracteristici
 - Aplicabil cand se lucreaza cu voluma mari da date in care obiectele sau caracteristicile pot fi redundante, foarte asemanatoare sau irelevante => un esantion poate furniza informatii relevante despre intreaga multime
 - Discretizare – reducerea numarului de valori pentru un atribut
 - NOTA: ac tehnici pot fi folosite numai in cazul in care rezultatele generate pentru fiecare submultime pot fi combinate intr-un rezultat care sa se potriveasca intregului set de date

Date dinamice

- De regula mulțimile de date sunt dinamice – pot fi adăugate noi attribute sau caracteristici sau pot fi înlăturate sau înlocuite cu unele noi
- => algoritmi ar trebui să evolueze în timp
 - cunoștințele derivate din date ar trebui să fie actualizate incremental



Date incomplete

- Datele disponibile nu conțin suficiente informații pentru a descoperi cunoștințe noi
 - Descrierea insuficientă a unui obiect prin caracteristici
 - Număr insuficient de obiecte
 - Valori lipsă ale unei anumite caracteristici
 - Ex: se analizează datele pacienților cu probleme cardiace. Este imposibil să se facă distincție între pacienții sănătoși și cei bolnavi, dacă sunt disponibile numai date demografice
- Este necesară identificarea problemei și găsirea măsurilor pt îndepărtarea sa
 - Trebuie analizată mulțimea datelor existente și determinat dacă caracteristicile și obiectele dau o reprezentare corespunzătoare pentru problema de rezolvat
 - Semn al incompletitudinii datelor – când noile cunoștințe au calitate scăzută iar aceasta nu poate fi îmbunătățită prin aplicarea altor metode de data mining
- Soluția: colectarea și înregistrarea de date adiționale

Date redundante

- Exista doua sau mai multe obiecte identice sau doua sau mai multe caracteristici intre care exista corelații puternice
 - Pot fi eliminate pentru a reduce timpii de prelucrare
- In anumite cazuri, datele redundante pot aduce informații utile, gen frecventa obiectelor identice
- Un caz special de date redundante – datele irelevante – caracteristici sau obiecte care nu sunt semnificative pentru analiza
- Datele redundante pot fi identificate prin algoritmi de selecție si extragere a caracteristicilor

Valori lipsă

- Pot rezulta din:
 - Introducerea de date incomplete
 - Măsurători incorecte
 - Erori ale echipamentelor
- Sunt de regula cunoscute ca valori NULL si sunt simbolizate prin .null., '*' sau '?'
- In domenii precum medicina, cca. 50% sunt valori lipsa
- \exists doua metode de abordare:
 - Eliminarea valorilor lipsa - se renunță la obiectul sau caracteristica unde sunt valori lipsa
 - Posibil daca este un număr mic de valori lipsa si daca acestea nu au importanta cruciala pentru analiza
 - Completarea valorilor lipsa
 - 2 tipuri de algoritmi:
 - Completarea cu o singura valoare
 - Completarea cu mai multe valori selectate după diferite criterii dintr-o mulțime de valori calculate

Pregătirea datelor pentru data mining

- Consuma cea mai importantă parte a eforturilor depuse în procesul de KDD
 - Integrarea datelor din mai multe surse
 - Extragerea caracteristicilor si selecția
 - Discretizarea datelor
 - Curățarea datelor

Integrarea datelor din mai multe surse

- Datele asupra cărora se aplica algoritmi de data mining provin din surse diferite sau din baze de date relaționale
 - necesară aducerea acestora într-o singură mulțime de date
 - prin denormalizare – joncțiuni între datele care se găsesc în diferitele tabele ale unei baze de date relaționale
 - prin integrarea datelor din surse multiple – din interiorul sau din exteriorul organizației
- Diferitele surse de date folosesc:
 - stiluri diferite de păstrare a înregistrărilor
 - convenții de notații diferite
 - perioade de timp diferite
 - grade de agregare a datelor diferite
 - diferite chei primare
 - au diferite tipuri de erori
 - Datele trebuie asamblate, integrate și curățate
 - » o soluție posibilă pentru datele interne organizației – Data Warehouse
 - » Datele din surse externe (ex. datele demografice pentru aplicații de marketing sau vânzări) trebuie de asemenea curățate și integrate cu restul datelor

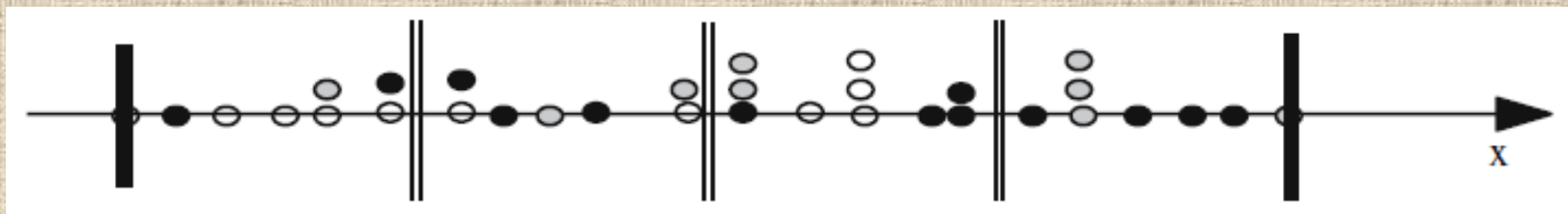
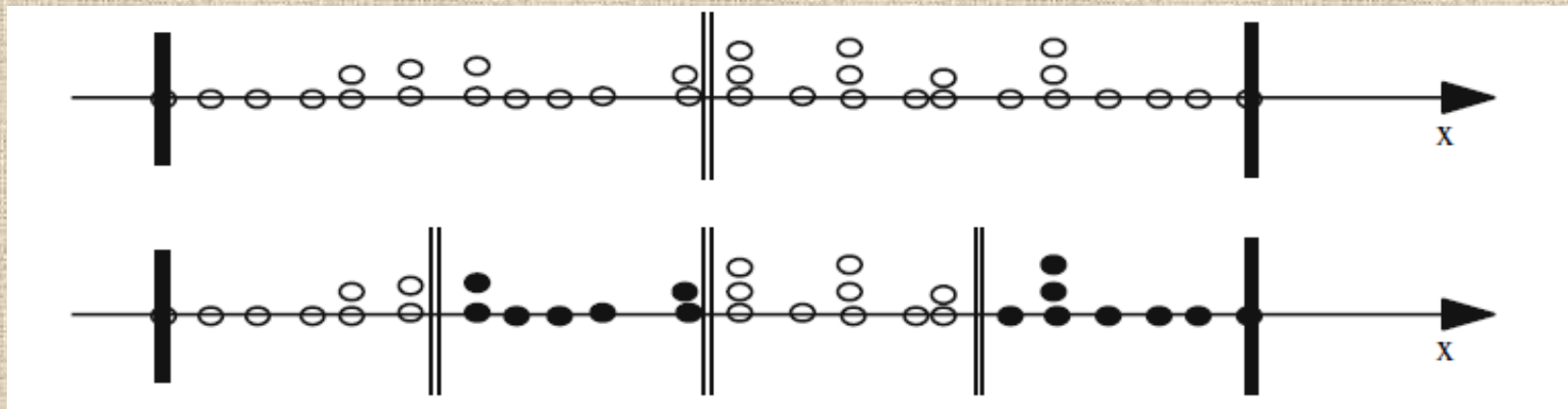
Extragerea caracteristicilor si selecția

- Extragerea caracteristicilor – alege un subset de caracteristici din multimea celor existente in setul original de date
 - Genereaza noi caracteristici pe baza celor din setul original
- Pentru datele numerice, independente de timp, exista 2 tipuri de metode:
 - Supervizate – analiza discriminanta liniara Fisher
 - Nesupervizate – analiza componentelor principale (PCA), analiza componentelor independente (ICA)
- Pentru seriile de timp – transformata Fourier
- Motive (pentru transformarea datelor si reducerea dimensionalitatii modelelor):
 - eliminarea redundantei datelor
 - comprimarea setului de date
 - obtinerea unor tipare reduse care contin doar caracteristicile relevante si care permit proiectarea de clasificatori cu capacitati de generalizare crescute
 - descoperirea variabilelor intrinseci din date care permit proiectarea unui model de date si imbunatatirea intelegerii fenomenului care genereaza tiparele

Discretizarea datelor

- Doua obiective :
 - reducerea numarului de valori distincte ale unei caracteristici – reducerea complexitatii modelului
 - transformarea valorilor continui in valori discrete – deoarece majoritatea algoritmilor de data mining opereaza cu valori discrete
- Scop: reducerea numarului de valori pentru attributele continui prin gruparea acestora intr-un numar de n intervale (bins)
- Probleme asociate:
 - cum se alege numarul de intervale
 - cum se decide latimea acestor intervale
- Discretizarea poate fi facuta cu sau fara a lua in considerare informatii referitoare la clase
 - algoritmi de discretizare supervizata (class aware)
 - algoritmi de discretizare nesupervizata (class-blind)
- Daca exista acces la datele de antrenare si exista informatii referitoare la clase algoritmul de discretizare ar trebui sa o foloseasca, mai ales daca, ulterior, pentru construirea modelului se utilizeaza algoritmi supervizati
 - alg de discretizare ar trebui sa maximizeze interdependentele dintre valorile atributelor si eticheta de clasa
 - alt avantaj –se minimizeaza pierderea din informatia originala

Discretizarea datelor (2)



Discretizarea datelor (3)

- Discretizarea atributelor poate fi:
 - statica – presupune discretizarea fiecarui atribut independent de discretizarea altor attribute
 - dinamica – toate attributele sunt discretizate simultan, tinandu-se cont de interdependentele dintre ele
 - locala- partiile produse se aplica numai unor zone localizate ale spatiului exemplilor
 - globala – toate attributele sunt discretizate si produc $n_1 * n_2 * \dots * n_d$ regiuni
 - n_i – numarul intervalelor pentru atributul cu indicele i
- Prin discretizare se asociaza fiecarui interval obtinut o valoare discreta care poate fi:
 - nominala
 - numerica

Discretizarea datelor (4)

- Procesul de discretizare presupune doi pasi:
 1. Se alege numarul de intervale discrete – de obicei este dat de utilizator, poate fi determinat prin metode euristie sau poate fi calculat cu ajutorul unor algoritmi
 2. Se determina limitele fiecarui interval in parte (uneori acestea sunt date de insusi algoritmul de discretizare)

Discretizarea datelor (5)

- numarul de intervale in care este impartita plaja de valori a unui atribut influenteaza puternic algoritmi de data mining:
 - eficienta procesului de constructie a modelului
 - calitatea modelului – abilitatea sa de a se generaliza pentru date noi
 - => un algoritm de discretizare ar trebui sa genereze un numar cat mai mic posibil de intervale
 - dar
 - prea putine intervale pot ascunde date referitoare la relatiile intre variabilele de clasa sicele interval – in special cand valorile atributelor nu sunt distribuite uniform, cand o mare parte din informatia originala poate fi pierduta
 - in realitate se face un compromis

Algoritm de discretizare nesupervizat

- Cel mai simplu de implementat
 - specifica numarul de intervale si cum ar trebui incluse datele in aceste intervale
 - practic:
 - numarul de intervale pentru fiecare atribut ar trebui sa nu fie mai mic decat numarul claselor (daca acesta este cunoscut)
 - numarul de intervale poate fi calculat dupa formula:

$$n_{Fi} = M / (3 * C)$$

- unde: n_{Fi} – numarul de intervale pentru fiecare caracteristica
 - M- numarul exemplilor de antrenare
 - C- numarul categoriilor cunoscute

Algoritmi de discretizare nesupervizati

Exemple

1. Discretizare in intervale de lungime egala
 - Se gasesc valorile minima si maxima pentru fiecare caracteristica
 - Se divide acest interval intr-un numar n_{Fi} de intervale cu lungime identica, decis de catre utilizator
2. Discretizare in intervale cu frecventa de valori egala
 - Se gasesc valorile minima si maxima pentru fiecare caracteristica
 - Se sorteaza valorile atributelor in ordine descrescatoare
 - Se divide aceasta gama intr-un numar n_{Fi} de intervale, astfel incat fiecare interval sa contina acelasi numar de valori sortate

Curățarea datelor !!!
discutata in materialele anterioare

