- problém velkých a komplexních dat

data mining = extrakce informací z dat
= nikoliv jen SQL query
= něco „navíc"

machine learning = vyvinutí znalosti dat pro zlepšení činnosti

KDD = knowledge discovery in database

- transformaci dat na informace

DM
1. data preprocessing
2. data mining
3. post processing

⊕ opakování

ML
1. data preprocessing
2. feature selection (výběr atributů)
3. výběr metody ML
4. training (learning)
5. evaluation

- ML je zřetel / součást AI

- proč? škálovatelnost
  komplexnost
  dimenze
  více zdrojů dat

- predikování → přiřazení ML
  - hodnota atributu na základě jiných

- deskriptivní → DM
  - popisuje souvislosti v datech

1. učení s učitelem
2. učení bez učitele
3. zpětnovazební učení
4. evoluční učení

1. klasifikace
2. regrese

ML

1. asociační analýza (vzory)
2. shlukování (podobnost skupin)
3. detekce anomálií (outliers)

DM

- nutné znát data
- typy kvalita, vztahy
- nutné předzpracování

- při: musíme vědět že něco je jen ID

dataset = množina objektů
atribut = vlastnost
- obvykle to jsou tabulky

- asymetrické atributy => dělat jsou pouze nenulové hodnoty

1. kategorické (kvalitativní) = symboly
   a. nominální - pouze odlišnost - ID, psč
   b. ordinální - uspořádání $<, \geq, \ldots$ - stupeň vzdělání

2. numerické (kvantitativní) = čísla
   a. intervalové - konstantní větší jednotka
   b. poměrové - počet, délka

- diskrétní -> z jich spočetná množina
  spojité -> třeba desetinná čísla

- metrika zajímá vliv na kvalitu: dimenze
  rychlost
  rozlišení

1. zaznamenané data
   - všechny obj. mají stejnou množinu atributů

2. grafová data
   - vztahy mezi objekty

3. časová data
   - v čase nebo prostoru
   - hledání „posloupností"

- data se shromažďují do budoucna (má vliv na kvalitu)
- ice a chybné hodnoty

převod typů atr.

- metody nelze užít speciální
  požadavky
- diskretizace = spojité na diskrétní
  binarizace = diskrétní na binární
  (one-hot encoding)

make_dummies

- rovnoměrné a cílené diskretizace

stejná šířka intervalů, ale

snaží se o maximální jedinečnost

řešit tříde popsat třídu y

úpravy hodnot atributů

- transformace proměnných
- možná změna rozsahu, pravděpodobnostního rozd.

standardizace

- obecně zajistit vlastnosti číselných hodnot
  (při min = 0 a max = 1, rozložit mezi)

předzpracování dat

agregace

- kombinace více do jednoho
- jak s ní? součet, počet, ...?

vzorkování (sampling)

- nesmí být zkreslen
- náhodně / stratifikace
  proporcionální

redukce dimenzionality

- méně atributů
- zachovat vizualizace i práce
- které?
- feature creation vs feature selection

- kvalita měření pomocí

precision - blízkost opakovaných měření stejné veličiny; směrodatná odchylka

bias - odchylka měření od skutečnosti

accuracy - blízkost precision skutečnosti

outlier - odlišení od většiny (může být legitimní)

- chybějící hodnota a její doplnění / eliminace?

odhad

podle nejbližších

průměr

regrese

- nekonzistentní hodnota -> obvykle překlep (může opravit)

podobnost a blízkost

- jak určit?

- v hodně metodách

$$\left(\sum_{k=1}^{m} |x_k - y_k|^r\right)^{1/r}$$

a) euklidovská vzdálenost $r=2$

b) Minkowského — " — $r=?$

c) Hamingova — " — $r = 1$ a XOR $\oplus$

d) Mahalanobisova — " — (eliminuje vliv vzájemného rozptylu atributů)
- inverzní kovariantní matice

simple matching coefficient (pro binární kontingenční tab.)

Jaccard coefficient (binární; ignoruje f_{00})

kosinová podobnost (ve textu)

Data exploration

- cílené první seznámení při pohledu na data
- statistiky: průměr, percentily, medián, ...
  a
  vizualizace: grafy, matice, ... histogram

- funkce pro podobnost je definována pokud splňuje: nezápornost
  identitu
  symetrii
  trojúhelníkovou nerovnost

korelace objektů

Pearsonův koeficient

$$\mathrm{corr}(x, y) = \frac{s_{xy}}{s_x \, s_y}$$

kovariance

směrodatná odchylka

$[-1, 1]$

$x_i = a y_i + b$

Regrese

- predikce - e spojitých cílových atributů
- z existující hodnoty odhadujeme hodnotu

1) lineární regrese

$$f(x_1, \dots, x_n) = a_0 + a_1 x_1 + \dots + a_n x_n \approx y$$

a ... regresní koef.

- řešíme problém minimalizace chyb

⇓

chybová fce squared error => least squared problem

- vznikne soustava lin. rovnic

2) nelineární regrese

- jakékoliv jiné fce (diferencovatelné)
- např: kvadratické

hodnocení regresu

- accuracy $= \frac{TP + TN}{TP + TN + FP + FN}$ - confusion matrix
- error rate $= 1 - acc$
- precision $= \frac{TP}{TP + FP}$

- recall (sensitivity) $= \frac{TP}{TP + FN}$
- specificity $= \frac{TN}{TN + FP}$
- f1 score

problém a model

- nedostatečné modelování a disciplíny
- rozhodit obj. do definovaných tříd

⇓

hledáme ohraničení a přesnou funkci

- model je „reprezentace" vztahu mezi vstupními atributy a třídou
- modelů je mnoho: strom
  rule-based
  k NN
- u modelů se říká hypotéza

class imbalance problem

- nevyvážený počet objektů více tříd
- online učení ⇒ 99% NEG 1% POS

⇓

model vždy řekne NEG, když to nebude rozhodování

- online to i nefiltry (např: accuracy)

- ŘEŠENÍ: úprava dat (under-sampling)

úprava algo. (cost sensitive learning, váha vzorků)

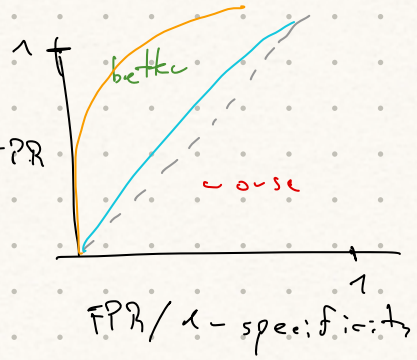
použití speciálních metrik na balancení

- ROC křivka a AUC

↳ TPR vs FPR

true positive rate

recall / TPR



rozhodovací stromy

- kořen => vstup
  - vnitřní uzel => rozhodovací podmínka
  - list => klasifikovaná třída
- hierarchický neparametrický model
- z 1 uzlu více než 2 hrany klidně
- pro 1 problém je exponenciálně mnoho stromů

⇓

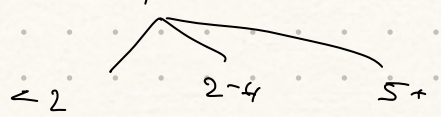
greedy strategie

⇓

Hunter algoritmus (zakład složitějších algoritmů)

1. pokud všechny objekty ∈ D mají stejnou class vrať uzel
2. zkusme test nad atributy (obvykle jednoduchý) rozklad dejme D na množiny se stejnými výsledky

- otázky mohou být: typu „počet částí?"



- po vytvoření může se provést strom (zjednodušení přeučení)

test nad atributy

a) binární : 2 hodnoty výsledku

b) spojité : disjunktní intervaly

c) nominální/ordinální : k hodnot výsledků

- ony testy mohou být větší vyhodnot

(im)purity = (ne)čistota

- chceme minimalizovat

$$I(t) = \frac{m(i|t)}{m(t)}$$

počet obj. v t patřících do i

t ... uzel

počet obj. v t

- chceme co nejvíce čisté listy

information gain

- určuje vyhodnocení testu
- míra by být maximální
- zkonstruovat strom NP-hard => heuristika
- čas. složitost nezávisí na teoretických detech

$$\Delta(t) = I(t) - \sum_{t'} \frac{N(t')}{N(t)} I(t')$$

entropie

1. specifikace entropii stupňů rozsahu $I(A)$

2. výt entropii přechodů stupňů na zařízení $I(A)$

3. výsledkem je ta aktuální dělení

oblique rozhodovací stromy (testy nad více atributy)

generalizace

- naučit se při trénování spíšeni odhadovat budoucí neznámé data
- chyba generalizace = nahlédnutí chyb na testovacích datech

přeučení (overfitting)

- dokonale modelují trénovací data včetně šumu
- ale selže na testovacích
- málo dat a uměle složitý model
- odhad chyby generalizace
- princip KISS pro vyhnutí se
- preferujeme jednodušší model

- v rozhodovacích stromech:

  pre-pruning – zastavení učení (max. hloubka, min. objektů, ...)

  post-pruning – prořezání po učení
  – lepší, ale náročnější

validační metody

hold-out – oddělení validační část (50/50 nebo

1/3 a 2/3)

random subsampling - opakovaný hold-out

cross validation - rozdělení na k stejně velkých disjunktních částí

.632 bootstrap

rule-based

- pravidla IF - THEN
- předpoklad → následek
- „pravidlo popisuje „obj" pokud je platné pro atributy
- vlastnosti pravidla mají vliznou kvalitu

  1) coverage - kolik obj. je pokrytých
  2) accuracy - kolik z pokrytých obj. má „správnou" class label

- pravidla obvykle vzájemně vyloučená (pokud ne, tak priorita nebo hlasování)
- vyčerpávající pokrytí = všechny obj. mají nějaké „své" pravidlo

  (jinak nějaké DEFAULT vítězí)

tvorba pravidel

- přímé metody - postupné vytváření množiny objektů
- nepřímé metody - z stromů, jiných složitých struktur

sekvenční pokrývání (příště)

- jeden přístup
- ekvivalent LEARN-ONE-RULE
- postupně pokrývat jednotlivé skupiny z D, a ony pokryté obj. pak odstranit
- vždy se konstruuje jedno pravidlo, které se tvoří od počátečné podmínky a přidává se (dokud se zlepšuje výsledek)
- druhá možnost je vyhnout pozitivní obj. a jeho atributy využít pro tvorbu, případně z nich odebírat
- pruning = pravidla se zpětně odstraní

RIPPER

- vylepšený algoritmus
- sekvenční pokrývání (podle class label vzestupně)
- od obecného ke specifickému (dokud pokrývá jen pozitivní objekty)

- když se stromu vezme cesta z kořene do listu a spojíme podmínky, dostaneme pravidlo

- tabla frecuencia C4.5 reglas

- clasificar: igual predecir igual - stump
- procedimiento se pueden descriptivamente
- ubicación que class imbalanced data

k-nearest neighbors

- neparametrické
- není fáze učení
- podobnost se určí jako vzdálenost
- podle jejich class label se určí „label"
- důležitá je volba velikosti k
- není žádný model (instance-based classifier)
- každý z k objektů může mít stejnou váhu nebo lze být vážené
- atributy a jejich hodnoty jako reálné
- dotaz jsou časově složité $O(N)$
- hranice oblastí v prostoru bývá velmi členité (dobře kopírující)
- atributy je nutné standardizovat pro stejný rozsah
- nesmí být chybějící hodnoty
- často Euclidovská vzdálenost (ucházející) $\sum_{i=1}^{d} (x_i - y_i)^2$ $r=2$

Manhattanská $\sum_{i=1}^{d} |x_i - y_i|$ $r=1$

Minkowskiho $\left(\sum_{i=1}^{d} |x_i - y_i|^r\right)^{1/r}$ $r=\infty$

Bayesovská klasifikace

- na základě pravděpodobností:

věta: $$P(Y \mid X) = \frac{P(X \mid Y)\, P(Y)}{P(X)}$$

- příklad s nemocí a testováním v populaci

- X ... atributy $\quad$ Y ... class label

  vektor

- $P(Y)$ ... prior pravděpodobnost $\quad$ $\bar{x}$ ... stř. hodnota (průměr)

  $P(Y \mid X)$ ... posterior — " —

  $P(X)$ je konstantní

  σ^2 ... rozptyl

  $$\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n-1}$$

- princip Maximum a Posteriori

  $$\hat{y} = \arg\max P(X \mid Y = y)\, P(Y = y)$$

Naivní Bayes

- předpokládá podmíněnou nezávislost atributů dané class (atributy nejsou navzájem závislé)

- — lze na odhadnout pravděpodobnost pravděpodobnost

  $P(X_i \mid Y)$

  $\Downarrow$

  přímo z trénování

- pokud pro některý atribut X je $P(X|Y)=0$ pak bychom nemohli klasifikovat, proto se používá odhad a) Laplace - odhad

b) m - odhad

vlastnosti

- rychlé trénování i klasifikace
- odolný vůči šumu i prázdným
- vnímatelný a korelované atributy, zhoršují výsledky
- dobře zvládá chybějící atributy

příklad

- vzorec pro naivní

$$P(x_i|Y) = \left(\leftarrow \frac{1}{\sqrt{\sigma^2}\sqrt{2\pi}} e \right)^{-\frac{1}{2}\cdot\frac{(x_i-\bar{x})^2}{\sigma^2}}$$

průměr

rozptyl

- udělám pro všechny atributy a ANO i pro všechny a NE
- u ANO i NE (zvlášť to mezi sebou vynásobím)
- to kde dostanu větší číslo je pravděpodobnější

asociační analýza základ

- odhalení skrytých závislostí v datech
- pokladny v transakční databázi
- frequent item-sets
- typicky v market basket analysis

- items
  transakce
  item-sets
  asociační pravidla
  support count $\sigma(X) = |\{t_i \in T \mid X \subseteq t_i\}|$
  support $s(X \to Y)$ ← podíl transakcí obsahující X i Y zároveň
  confidence c

počet transakcí, kde je daná položka

problém extrahování pravidel

- oddělení exponenciální nároky výpočetní pravidel
- hledání tzv. frequent item-sets
- minimální s_{min} a c_{min}

v kolika transakcích se asi část dané výrazu minimálně

jak moc je pro obsažení Y pokud obsahuje X

Zajímavost vztahů

- potřebujeme jiné metriky než jen zajímavost
- lift : $=1$ -> nezávislé >1 -> silné <1 -> nepodporují se

levarage

IS míra

ϕ - koeficient

- objektivní a subjektivní míry
- kontingenční tabulka
- Simpsonův paradox

$$\text{lift}(X \to Y) = \frac{c(X \to Y)}{s(Y)} \longrightarrow \frac{s(X \cup Y)}{s(X)}$$

$$= \frac{N \cdot \sigma(X \cup Y)}{\sigma(X) \cdot \sigma(Y)}$$

Apriori

- generuje frequent itemsets
- princip: všechny podmnožiny frequent jsou také frequent
  : všechny nadmnožiny infrequent nemohou být frequent
- využívá hashovací stromy

FP-Growth

- strom FP tree
- fce int. sestupně dle support count
- rozdělí a panuje
- pouze 2 průchody datasetu

ECLAT

- vertikální data (tzv. tid-list)
- každá položka má ID transakce, kde se vyskytuje
- průnik tid-listů

- po vytvořování můžeme se generují pravidla

- z {A, B} víme že platí A → B nebo B → A

z {A, B, D} —— " —— AB → D

AD → B

DB → A

- ověření postupu tak, že vezmu levou stranu pravidla a u toho dosadím se vystupuje a pokud to stejný způsob tím uznáme

Cíl shlukování

- seskupení do skupin (shluků)
- tvořily nejsou známy předem
- unsupervised
- podobné v sebe, odlišné co nejdál od sebe
- shluky můžou být určitá podmnožina na jediné – vzorku dat

typy shlukování

rozklad – shluky se nepřekrývají

hierarchické – uvnitř a stromové uspořádané

výlučné – každý bod je striktní v 1 shluku

překrývající se

fuzzy – body mají stupně příslušnosti pro shluky

úplné vs částečné

/ každý bod musí někde být

\ umožňuje některé body vyhodit

typy shluků

dobře oddělené

prototypové

grafové

hustotové

konceptuální - obj. sdílí vlastnost

k-means metoda

- iterativní rozhledový
- do k předem dané shluků
- prototypy centroid a medoid

1) zvolí se prototypy
2) spočítá se vzdálenost bodů k prototypům a přiřadí se
3) přepočítají se prototypy jako těžiště přiřazených bodů
4) opakování dokud se prototypy nezmění

- jak volit počáteční prototypy
- na shluky může být použit postprocessing

pečlivé k-means

- zadání se shluky úroveň objektů a ten se dělí pokud není k shluků

vlastnosti

- jednoduché a rychlé
- obvykle nutné spustit vícekrát

- špatně reaguje na anomálie

- předchozí sklo = jen prototyp
- selhává na cizích tvarech (pistorové, potěšlé, unavené)

hierarchické shlukování

- vizualizace dendrogramem

1) aglomerativní – spojování 2 shluků

2) divizní – rozdělování shluků na singletony

blízkost shluků

grafové

- single link / min
- complete link / max
- průměrné (mezi min a max)

prototypové

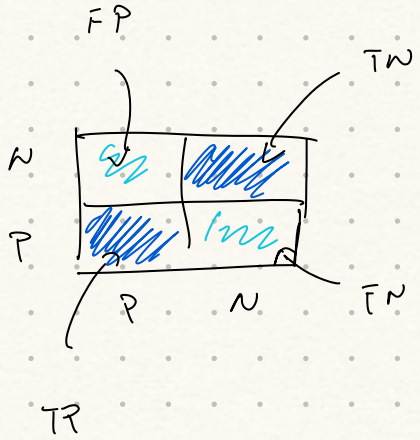
- centroidní
- Wardova metoda

hustotní shlukování

- hledá hustší oblasti objektů
- jak definovat hustotu?

a) center based – např. DBSCAN ($k = 4$)

- každá metoda nejlepší shluky najde
- nemusí být vždy dobrou cestou informace



AND : 6 NE : 5 (11)

$$\frac{6}{11} \log_2 \frac{6}{11} + \frac{5}{11} \log_2 \frac{5}{11} + 0{,}994$$

$$E + \Sigma$$

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

support count : ile razy się X : Y / use

confidence : prawd. że ten bank Y posiad X

MSE $\quad E(c) = \sum_{i=1}^{n} (y_i - f(x_i))^2$

$$f_1\text{-score} = \frac{2TP}{2TP + FP + FN}$$

k-NN $\quad O(N \cdot m)$

N ... počet bodů

m ... počet dimenzí