Machine learning a data mining 1

Jan Outrata

KATEDRA INFORMATIKY
UNIVERZITA PALACKÉHO V OLOMOUCI

přednášky

Obsah

Anotace

Předmět je první částí dvousemestrálního kurzu věnovaného principům a hlavním metodám získávání znalostí z dat (data mining) a strojového učení (machine learning). Po úvodu do problematiky a rozboru dat jsou, z algoritmického hlediska, probírány základní data mining metody klasifikace, asociační analýzy a shlukování využívané (nejen) pro machine learning.

Literatura

- Tan P.-N., Steinbach M., Kumar V.: Introduction to Data Mining. Pearson Education, 2005.
- Marsland S.: Machine Learning: An Algorithmic Perspective, 2nd ed. Chapman and Hall/CRC, 2014.
- Zaki M. J., Meira W. Jr: Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press, 2014.
- Poole D. L., Mackworth A. K.: Artificial Intelligence: Foundations of Computational Agents, 2nd ed. Cambridge University Press, 2017.

Úvod

Úvod

- narůstající objem generovaných a sbíraných dat (výzkum, senzory, obchod, osobní; Zettabyte Era) – jednoduchost sběru a uložení $\rightsquigarrow$ jakákoliv, kdekoliv, kdykoliv, často jen s vírou užitečnosti
- omezení lidí a tradičních (statistických) metod analýzy a využívání dat: **velikost a komplexnost dat („surovost", nový typ)**, potřeba **jiné zpracování (výstup)**

$\rightarrow$ **data mining** – extrakce „informace, znalosti" z dat (nových typů, novým způsobem), tradiční základem (motivace stejné), potřeba **porozumění** datům a oblasti

$\rightarrow$ **machine learning** – využití „informace, znalosti" z dat pro zlepšení nějaké činnosti, potřeba schopnosti **adaptace a generalizace** na data a z dat

- metody často těžko zařaditelné – jen extrakce „informace, znalosti" nebo už i její využití k něčemu?? $\rightsquigarrow$ splývání

Úvod

- aplikace (a také motivace/důvody vzniku):
  - **výzkum (a průmysl)**: sběr dat o lidech a pro nové objevy a potřeby, např. o pacientech (testy), genech a proteinech (sekvenování genomu, exprese), klimatu (senzory satelitů), vesmíru (teleskopy), materiálech (výsledky pokusů), systémech a strojích (monitoring, asistence) – velikost dat, nový typ (časoprostornost), šumovost apod.
  - **obchod**: info o nákupech a zákaznících (z prodejen, e-shopů, supportu aj., profilace), typicky pro vylepšování a cílení nabídky (personalizace, doporučování, predikce), cenové politiky (konkurence), optimalizaci zásob a distribuce zboží, detekce podvodů (služby)
  - **IT**: komunikační data (síťový provoz, management, obchod), multimedia – velikost, stream, zpracování, rozpoznávání (obraz, zvuk)
  - ! otázky narušení soukromí (obchod, zdravotnictví) → privacy-preserving data mining
- konference na data mining: IEEE ICDM, ACM SIGKDD, SDM, PKDD, PAKDD
- konference na machine learning: ICML, AAAI

Data mining (DM, „dolování dat“)

= proces extrakce nových a užitečných (jinak skrytých) informací z (rozsáhlých) dat

- ne získávání (partikulárních) informací, např. databázový nebo vyhledávací dotaz = **information retrieval** – datové struktury (indexy), vyhledávací algoritmy, DM vylepšují
- součást celého procesu získávání (objevování) znalosti z dat, „transformace“ dat na informace = **knowledge discovery in databases (KDD)** (více např. Fayyad):
  1. **data preprocessing** – sběr a předzpracování vstupních dat, např. získávání z více zdrojů, formátování, čištění (od šumu), výběr a úpravy atributů (feature selection, dimensionality reduction, normalization) a záznamů aj., časově náročné
  2. data mining – vybraná metoda se zvolenými parametry
  3. **postprocessing** – úprava výstupní informace pro využití (např. v systémech pro rozhodování, v metodách **machine learning**), např. filtrace validní a užitečné (statisticky, testování hypotéz), vizualizace, interpretace
  4. **opakování** – s jinou metodou nebo parametry

Machine learning (ML, strojové učení)

= proces adaptace na a generalizace z dat o (obecně) „činnosti“ pro její správnější „vykonávání“, na základě informací z dat, např. „naučení se“ kroků k vítězství ve hře na základě průběhů her, ale třeba i jen „vytvoření“ predikce hodnoty nějaké veličiny na základě hodnot jiných veličin

- člověk se učí ze zkušenosti: zapamatování, adaptace na a generalizace z ní pro správnější rozhodování → modelování informacemi z dat a procesu učení metodou ML
- části/fáze:
  1. data preprocessing – stejné jako u KDD
  2. **feature selection** – výběr jen některých „proměnných“ (atributů dat), někdy sloučené se sběrem a předzpracováním dat (např. data rozsáhlá, těžko k získání všech, chybovost při získávání apod.)
  3. výběr metody ML a volba parametrů – experimentování
  4. **training** = proces učení – aplikace metody ML na případná tzv. trénovací data (training set) pro naučení se **modelu dat**, může být výpočetně náročnější (probíhá málo často) než následné použití modelu na další data
  5. **evaluation** – testování a vyhodnocení „správnosti“ naučeného modelu na tzv. testovacích datech (testing/validation set), před jeho použitím na další data, často porovnáním s informacemi od expertů v oblasti

Machine learning (ML, strojové učení)

- část oblasti/základ **umělé inteligence (artificial intelligence, AI)** – „inteligentní chování“ na základě zjištěných informací a (strojového) učení, ale také uvažování a usuzování (reasoning, logical deduction) aj. – symbolické zpracování dat a informací (DM a ML = subsymbolic)

DM a ML

Důvody pro DM a ML:

- **škálovatelnost metod** – pro rozsáhlá data (mnoho záznamů) $\rightarrow$ vyhledávací problémy a strategie, efektivní datové struktury, paralelní a distribuované algoritmy
- **dimenze dat** = mnoho atributů (člověk $\sim$ tři dimenze?), navíc časoprostorových – výpočetní složitost algoritmů narůstá $\rightarrow$ metody redukce dimenzionality (projekce, může skrýt informaci)
- **komplexnost dat** – heterogenní (symbolické i číselné) atributy, semi-strukturované záznamy (text, časové aj.) – vztahy (např. časoprostorové korelace, souvislosti, hierarchické)
- **více zdrojů dat** – problémy komunikace při výpočtu, konsolidace výsledků, bezpečnosti aj. $\rightarrow$ distribuované algoritmy
- **netradiční analýza** – automatizace procesu generování a testování hypotéz, jiný typ výstupu (interpretace)

DM a ML

Vznik z a využití metod:

- **statistiky/matematiky** (DM): vzory (sampling, **pattern recognition**), odhady (estimation), testování hypotéz (více k DM vs. statistika např. Hand)
- **biologie** (ML): modelování/simulace procesu učení (adaptace, generalizace) a biologických procesů (neuronové sítě, evoluční algoritmy)
- **informatiky**: algoritmizace, optimalizace, složitosti výpočtů, evolučních výpočtů, teorie informace, vizualizace, information retrieval, ...
- **IT**: databázových a informačních systémů, paralelních a distribuovaných systémů

DM a ML

Úlohy (typicky):

- **prediktivní** (ML) = predikce hodnoty určitého atributu (**cílový (target), závislá proměnná**) na základě hodnot jiných atributů (**explanatory, nezávislá proměnná**)
- **deskriptivní** (DM) = explorativní extrakce vzorů/informace (korelace, trendy, shluky, anomálie) popisujících vztahy v datech

Typy ML:

- **supervised learning** (učení s učitelem, z příkladů) = naučení se správných akcí (v rámci činnosti) na základě známých správných akcí pro nějaká, trénovací data (o akcích/činnosti), nejběžnější
- **unsupervised learning** (učení bez učitele) = naučení se vzorů (správných) akcí vyhledáním v datech a kategorizací vzorů, pro data neznámé správné akce
- **reinforcement learning** (zpětnovazební učení, s kritikou) = naučení se správných akcí prohledáváním a zkoušením možností na základě dat a sdělení o (míře) ne/správnosti akce, ne přímo správné akce
- **evolutionary learning** (evoluční učení) = naučení se správnější akce modelováním biologické evoluce (adaptace pro přežití) v prostředí daném daty

Úlohy DM a ML

Prediktivní modelování (ML)

= tvorba/naučení **modelu dat**/cílového atributu (závislé proměné) jako funkce jiných atributů (nezávislých proměnných) – často ve formě pravidel „jestliže hodnoty, pak hodnota“

- **klasifikace** – pro diskrétní cílový atribut, např. bude-nebude pršet, zákazník koupí-nekoupí, pacient má-nemá nemoc, ...
- **regrese** – pro spojitý cílový atribut, např. předpověď počasí (teploty), pravděpodobnost zemětřesení, ...

→ minimalizace chyby mezi modelem predikovanou a skutečnou hodnotou cílového atributu

Úlohy DM a ML

Prediktivní modelování / klasifikace (ML)

název	hymenofor	prsten	pochva	jedovatá
hřib borový	rourky	ne	ne	ne
kozák	rourky	ne	ne	ne
klouzek	rourky	ano	ne	ne
čirůvka	lupeny	ne	ne	ne
bedla	lupeny	ano	ne	ne
žampión	lupeny	ano	ne	ne
muchomůrka zelená	lupeny	ano	ano	ano
muchomůrka červená	lupeny	ano	ne	ano
závojenka	lupeny	ne	ne	ano
hřib satan	rourky	ne	ne	ano

část modelu dat?:

jestliže hymenofor = rourky a prsten = ano, pak jednovatá = ne

jestliže hymenofor = lupeny a pochva = ano, pak jedovatá = ano

Úlohy DM a ML

Asociační analýza (DM)

= extrakce vzorů („zajímavých“ vztahů, asociací, skrytých v datech) popisujících související (asociované) atributy – typicky ve formě implikací mezi atributy (items) nebo jejich podmnožin (itemsets)

- např. geny se související funkcí, webové stránky přistupované společně, spolu se vyskytující lidé, ...

→ efektivně jen nejzajímavější (často se v datech vyskytující) vzory, kvůli (exponenciální) velikosti prohledávaného prostoru vzorů

Úlohy DM a ML

Asociační analýza (DM)

tID	obsah košíku
1	chleba, máslo, mléko
2	mléko, káva, kuře, ovoce, sušenky
3	máslo, vejce, ovoce, zelenina
4	chleba, ryba, ovoce, zelenina
5	chleba, vejce, kuře, ovoce
6	máslo, mléko, káva, sušenky

vzory?:

$\{\text{káva}\} \rightarrow \{\text{mléko, sušenky}\}$

$\{\text{zelenina}\} \rightarrow \{\text{ovoce}\}$

Úlohy DM a ML

Shlukování (shluková analýza) (DM)

- nalezení skupin (shluků) záznamů navzájem podobnějších v rámci shluku než mimo něj (v jiných shlucích)
- např. podobné objekty, příbuzné oblasti, ...

→ na základě (definované) podobnosti nad záznamy

Úlohy DM a ML

Shlukování (shluková analýza) (DM)

- nalezení skupin (shluků) záznamů navzájem podobnějších v rámci shluku než mimo něj (v jiných shlucích)
- např. podobné objekty, příbuzné oblasti, ...

→ na základě (definované) podobnosti nad záznamy

článek	klíčová slova
1	škola, rodiče, ekonomika, kvalita, prestiž, hodnocení, technologie, vzdělání
2	škola, rodiče, kvalita, individualita, hra, hodnocení, vzdělání
3	metodika, škola, hodnocení, inovace, program, technologie, vzdělání
4	emoce, strach, věk, terapie, rodina, porucha
5	věk, vývoj, emoce, porucha, vztahy
6	produktivita, bezpečnost, vzdělání, emoce, výsledky, vztahy

shluky?:

$\{1,2,3\}$, $\{4,5,6\}$

Úlohy DM a ML

Detekce anomálií (DM)

- identifikace záznamů, tzv. **anomálií (outliers)**, výrazně odlišných od ostatních – problém s falešným označením běžných záznamů
- např. detekce podvodů, (síťových) incidentů, neobvyklých vzorů, ...

→ tvorba „profilů“ (vzorů) běžných záznamů a porovnání záznamů s nimi

Úlohy DM a ML

Detekce anomálií (DM)

- identifikace záznamů, tzv. **anomálií (outliers)**, výrazně odlišných od ostatních – problém s falešným označením běžných záznamů
- např. detekce podvodů, (síťových) incidentů, neobvyklých vzorů, ...

→ tvorba „profilů“ (vzorů) běžných záznamů a porovnání záznamů s nimi

odchozí platby z účtu (po měsících):
650, 730, 50, 580, 6800, 880
490, 920, 660, 6800, 390
570, 410, 50000, 770, 6800, 840

anomálie (outliers)?:
50, 50000

Data

Data pro DM a ML

Významné „znát data“:

- **typ dat** – např. typy atributů popisujících objekty/záznamy (kvalitativní, kvantitativní), obecné a speciální vlastnosti (časoprostornost, vztahy mezi objekty); určuje použité metody
- **kvalita dat** – např. šum, anomálie (outliers), chybějící hodnoty, nekonzistence, duplicity, vychýlení (bias), nereprezentativnost; zvýšení zvyšuje kvalitu výsledků analýzy
- **předzpracování** – např. převedení spojitých atributů na diskrétní, snížení počtu atributů; pro zvýšení kvality nebo přizpůsobení zvolené metodě analýzy
- **vztahy** (předem) – např. podobnosti nebo vzdálenosti mezi objekty; analýza nad vztahy mezi objekty místo přímo nad objekty (např. shlukování)

Data pro DM a ML

Významné „znát data“:

- **typ dat** – např. typy atributů popisujících objekty/záznamy (kvalitativní, kvantitativní), obecné a speciální vlastnosti (časoprostornost, vztahy mezi objekty); určuje použité metody
- **kvalita dat** – např. šum, anomálie (outliers), chybějící hodnoty, nekonzistence, duplicity, vychýlení (bias), nereprezentativnost; zvýšení zvyšuje kvalitu výsledků analýzy
- **předzpracování** – např. převedení spojitých atributů na diskrétní, snížení počtu atributů; pro zvýšení kvality nebo přizpůsobení zvolené metodě analýzy
- **vztahy** (předem) – např. podobnosti nebo vzdálenosti mezi objekty; analýza nad vztahy mezi objekty místo přímo nad objekty (např. shlukování)

Př. vědět, že nějaký atribut je jen identifikátor objektů

Data pro DM a ML

dataset (data set) = kolekce (množina) **objektů**/záznamů (bodů, vektorů, vzorů, událostí, případů, pozorování apod.)

objekty popsány **atributy** (proměnnými, charakteristikami, vlastnostmi apod.) = vlastnost charakterizující objekt, s různou hodnotou pro různé objekty, např. tvar, nebo v čase, např. velikost

$\Rightarrow$ **objekt-atributová data** – strukturovaná, charakteristická pro DM a ML

- reprezentace datasetu typicky tabulkou – objekty = řádky, atributy = sloupce, ale i jinak (např. grafem)

Data pro DM a ML

dataset (data set) = kolekce (množina) **objektů**/záznamů (bodů, vektorů, vzorů, událostí, případů, pozorování apod.)

objekty popsány **atributy** (proměnnými, charakteristikami, vlastnostmi apod.) = vlastnost charakterizující objekt, s různou hodnotou pro různé objekty, např. tvar, nebo v čase, např. velikost

⇒ **objekt-atributová data** – strukturovaná, charakteristická pro DM a ML

- reprezentace datasetu typicky tabulkou – objekty = řádky, atributy = sloupce, ale i jinak (např. grafem)

atribut → Teorie měření (measurement theory):

míra (měření) = předpis (funkce) přiřazující atributu symbolickou nebo číselnou hodnotu, např. malý/střední/velký vs. číslo

(proces) **měření** = aplikace míry (funkční hodnota), přiřazení konkrétní hodnoty atributu pro daný objekt, např. změření velikosti

Typ atributu

- vlastnosti atributu nemusí být stejné jako vlastnosti hodnot míry

→ jaké vlastnosti atributu jsou reflektovány hodnotami míry (a obráceně, jaké vlastnosti hodnot, např. různost nebo uspořádání, jsou konzistentní s vlastnostmi atributu), např. ID vs. velikost a číslo

∼ **typ míry**

⇒ identifikace vlastností (operací) hodnot korespondujících s vlastnostmi atributu: **různost**, **uspořádání**, **sčítání** (a odčítání), **násobení** (a dělení)

- významný, pro „zacházení s atributem“, např. nepočítat průměr ID, nebo výběr metody analýzy

Asymetrické atributy

= důležité jen nenulové hodnoty = prezence

- často u datasetů většina hodnot pro objekty nulových
- významné pro např. asociační analýzu

Typy atributů

(S. S. Stevens, psycholog)

- **kategorické (kvalitativní)** $\sim$ symboly
  - **nominální**: hodnoty jen různá jména, jen pro rozlišení objektů (různost), např. ID, tvar, validní operace např. výběr, kontingence, korelace, transformace neměnící význam bijektivní
  - **ordinální**: pro uspořádání objektů, např. pořadí, známky, navíc operace např. medián, korelace s uspořádáním (rank), transformace zachovávající pořadí
- **numerické (kvantitativní)** $\sim$ čísla
  - **intervalové**: smysluplné rozdíly mezi hodnotami (existuje jednotka), např. datum, navíc operace např. (aritmetický) průměr, směrodatná odchylka, transformace lineární
  - **poměrové (ratio)**: smysluplné i podíly hodnot, např. počet, délka, navíc operace např. (geometrický) průměr, procenta, transformace i nelineární

- **diskrétní** = konečně nebo spočetně mnoho hodnot, typicky symbolických nebo celočíselných, **binární** = dvě hodnoty, typicky kategorické (ale i např. počet)
- **spojité (continuous)** = nespočetně mnoho hodnot, reálně-hodnotové (reprezentované s plovoucí řádovou čárkou s omezenou přesností), typicky numerické

Typy datasetů

Charakteristiky mající vliv na DM a ML:

- **dimenze** = počet atributů – mnoho → curse of dimensionality → předzpracování redukce
- **řídkost (sparsity)** = relativní množství nulových hodnot (asymetrických) atributů – často vysoká
- **rozlišení** – vlastnosti dat různé pro různé „úrovně pohledu“, např. rozdíly hodnot dle jejich přesnosti, „viditelnost“ vzorů nebo šumu, typicky u časoprostorových dat

Záznamová data

= všechny objekty popsány stejnou pevnou množinou atributů

- žádné (explicitní) vztahy mezi objekty nebo atributy
- reprezentace tabulkou, uložení **flat** file nebo relační databáze
- typická pro DM a ML

Typy datasetů

Záznamová data

- **transakční** = kolekce transakcí (objekty) jako množin položek (items, atributy) ∼ „nákupní košíky“ produktů... market basket data, položky → asymetrické atributy, i nebinární (počet, cena)
- **maticová** – číselné atributy, objekty ∼ body (vektory) vícerozměrného prostoru, rozměr ∼ atribut, → $n \times m$ matice pro n objektů (řádky) a m atributů (sloupce) – maticové operace
- **řídká (sparse)** – asymetrické atributy stejného typu, např. také transakční, document-term (dokumenty = objekty, termy/slova v nich = atributy, počet výskytů = hodnota, na pořadí slov nezáleží)

Grafová data

- **graf pro vztahy mezi objekty** – objekty = uzly, vztahy = hrany a vlastnosti hran (orientace, váha apod.), např. hypertextové dokumenty s odkazy
- **objekt = graf** = strukturované objekty – mají „podobjekty“ se vztahy → **subgraph mining**, např. chemické sloučeniny

Typy datasetů

Uspořádaná data

= uspořádané atributy, např. v čase nebo prostoru

- **sekvenční** = sekvence (hodnot) atributů, např. genomická (geny jako sekvence nukleotidů) – typicky predikce podobností genů z podobností sekvencí
- **temporální** (časová, sequential) – čas pro objekt nebo hodnotu atributu, rozšíření záznamových dat, např. transakční s časem celých transakcí nebo zařazení položek do transakce, **časová autokorelace** = podobné hodnoty atributů blízké v čase
  - **časových řad** – objekt = časová řada, např. měření v čase
- **prostorová** – prostorové informace k hodnotě atributu, např. měření/modelování pomocí mřížky, meteorologická, **prostorová autokorelace** = podobné hodnoty atributů blízké v prostoru

Ne objekt-atributová data:

→ extrakce atributů a vytvoření objekt-atributových

- např. společné části prvků dat = objektů jako (asymetrické) binární atributy
- problém postihnout všechny informace v datech, např. vztahy mezi objekty nebo atributy navzájem: např. časové řady pro body (= objekty) v prostoru

Kvalita dat

- analyzovaná data často sbíraná za jiným účelem (nebo „do budoucna“) ⇒ nemožnost „řešení kvality u zdroje“, prevence problémů

→ dosažení požadované úrovně u dat i výsledků analýzy:

- detekce a oprava problémů = **data cleaning**
- metody s tolerancí problémů

Problémy při sběru/měření dat

- způsobené lidskými chybami, omezeními sběrného/měřícího zařízení, vlivy okolí, systematickými chybami apod.

→ chyby způsobu měření: šum, artefakty, bias, nedostatečná přesnost hodnot aj.

→ chyby sběru/procesu měření: anomálie (outliers), chybějící a nekonzistentní hodnoty, duplicitní objekty aj.

- systematické i náhodné
- dále obecné, pro (běžné) specifické, např. časté překlepy, specifické metody detekce a opravy

Kvalita dat

Šum a artefakty

= náhodné (šum) nebo systematické (artefakty) chybné hodnoty nebo přidané objekty, např. kvůli vadě měřidla

- často pro maticová, časoprostorová data → metody redukce šumu ve zpracování signálu, obrazu apod.
- obecně obtížné odstranit (co je „šum“ a co není?) → metody s **odolností vůči šumu**

Přesnost (precision) a bias

- pro informování o nebo stanovení kvality měření (dat) a výsledků
- **přesnost (precision)** = blízkost opakovaných měření (stejné veličiny), měřena obvykle směrodatnou odchylkou, použití pouze **platných číslic (significant digits)** – pro vyjádření hodnot jen tolik číslic, kolik odpovídá přesnosti, např. měření pravítkem s milimetry jen na milimetry, s přesností $\pm 0,5$ mm
- **bias** = systematická odchylka měření od skutečné hodnoty, obvykle rozdíl mezi průměrem a hodnotou
- **přesnost (accuracy)** = blízkost opakovaných měření skutečné hodnotě, obecnější ∼ stupeň chyby měření, závisí na precision a bias

Kvalita dat

Anomálie (Outliers)

= vlastnostmi odlišné objekty od většiny ostatních nebo netypické hodnoty atributu, mnoho definic, typicky „extrémní“

- mohou být legitimní (a hledané) – rozdíl oproti šumu nebo artefaktům

Chybějící hodnoty (nebo i celé objekty)

- nezískané, atribut pro nějaký objekt (podmíněně) neplatný apod.

→ vyřazení objektů nebo atributů – ne mnoho, i neúplně popsané objekty mohou být užitečné nebo atributy významné

→ odhad – např. nejčastější (kategorická) hodnota, průměr, interpolace z ostatních („sousedních/nejbližších“ numerických) hodnot/objektů, hodnota z „nejbližšího/nejpodobnějšího“ objektu aj.

→ ignorování – adaptace metod, vynechání při používání atributu (např. výpočet podobnosti objektů), může vést k jen přibližným nebo i jiným výsledkům

Kvalita dat

Nekonzistentní hodnoty

- vzhledem k významu atributu nebo explicitním vztahům mezi atributy
- často překlepy a zjevné chyby, snadno zjistitelné a opravitelné
- pro opravu potřeba dodatečná nebo externí informace

Duplicitní objekty

= reprezentující stejný „skutečný“ – potřeba znát, i s příp. různými hodnotami některých atributů (= nekonzistence) → **deduplikace**

! ne „podobné“ objekty reprezentující různé „skutečné“

Kvalita dat

Problémy při použití dat

= vhodná pro zamýšlené použití?

- aktuálnost – užitečnost dat i výsledků analýzy jen po omezenou dobu
- relevance – potřebné informace?, jinak nízká vypovídající hodnota výsledků, problém **biasu vzorkování (sampling bias)** = data (vzorek) neobsahují dostatek různých objektů podle jejich zastoupení ve skutečnosti $\Rightarrow$ chybné (zavádějící) výsledky analýzy
- znalost (dokumentace) – např. o typech atributů, přesnosti hodnot, chybějících hodnotách (jejich specifikace), provázaných atributech $\rightarrow$ redundantní, výběr jen některých, původu dat atd.

Předzpracování dat (data preprocessing)

→ výběr objektů a/nebo atributů nebo vytvoření/změna atributů pro zlepšení analýzy (čas, kvalita)

Agregace

= kombinace více objektů do jednoho

- jak zkombinovat hodnoty atributů? – např. součet, průměr (kvantitativní), výběr nejčastější, sjednocení (kvalitativní)
- efektivně zrušení atributů nebo redukce hodnot atributu, např. dnů na měsíce
- menší data ⇒ možné náročnější metody, „pohled z vyšší úrovně“, agregované „stabilnější“ – menší variabilita (rozptyl), ale možná ztráta detailů

Vzorkování (Sampling)

= výběr objektů; pro prvotní průzkum (statistika) i finální analýzu (schůdnější, náročnějším algoritmem)

∼ celá data, jestliže je **reprezentativní** = maximálně stejné vlastnosti (zájmu) jako celá data, např. průměr hodnot atributů

→ **náhodné** – stejná pravděpodobnost výběru každého objektu

- bez ponechání – v populaci k výběru, pravděpodobnost výběru roste
- s ponecháním – objekt může být vybrán vícekrát
- nezohlednění (různých) četností výskytu objektů různých typů – méně četných méně vybraných

→ **stratifikované** – vybraný stejný počet objektů z každé dané skupiny (typu) objektů nebo relativní k velikosti skupiny

! velikost vzorku – větší (pravděpodobně) reprezentativnější, menší možná ztráta informace → dostatečná pravděpodobnost výběru

→ **adaptivní/progresivní** – zvyšující se velikost vzorku dokud ne dostatečná, např. dokud se zvyšuje kvalita výsledků, např. přesnost modelu

Redukce dimenzionality

- pro metody lepší menší počet atributů (dimenze dat): eliminace redundantních nebo irelevantních, redukce šumu, srozumitelnější model, snadnější vizualizace (dat i výsledků, často po dvojicích nebo trojicích atributů), výpočetní nároky, ...
- **curse of dimensionality** = (značně) náročnější analýza při zvyšující se dimenzi dat $\Rightarrow$ data řidší, relativně málo objektů pro tvorbu modelu (např. klasifikace), méně významné hustota a vzdálenost mezi objekty (např. shlukování)

$\rightarrow$ výběr atributů z původních = **feature selection**

$\rightarrow$ vytvoření nových atributů z původních = **feature creation**

$\rightarrow$ lineární algebra: projekce dat z vysokodimenzionálního prostoru do ménědimenzionálního, typicky pro spojité atributy, **analýza hlavních komponent (principal component analysis, PCA)** (nové atributy = hlavní komponenty = navzájem ortogonální lineární kombinace původních zachycující maximum variace dat) a **singular value decomposition (SVD)**, **faktorová analýza** (původní atributy = lineární kombinace nových „skrytých“), locally linear embedding (LLE), multidimensional scaling (MDS) aj.

Převod typů atributů

- některé metody vyžadují kategorické atributy, např. klasifikace, nebo (asymetrické) binární atributy, např. asociační analýza

→ **diskretizace** = převod spojitého atributu na diskrétní (kategorický)

→ **binarizace** = převod spojitého nebo diskrétního atributu na binární atributy

- také snížení počtu hodnot kategorického atributu – diskretizace (ordinální), sloučení více hodnot do jedné (nominální) – na základě např. vztahů mezi hodnotami
- vliv na výsledky metody, ideálně dle metody

Binarizace

1. jednoznačné přiřazení čísla z intervalu $[0, m-1]$ každé z m hodnot kategorického atributu, se zachováním pořadí u ordinálního atributu
2. reprezentace čísel ve dvojkové soustavě s $\lceil \log_2(m) \rceil$ ciframi a nový binární atribut pro každou cifru

! nechtěné vztahy mezi binárními atributy – původní hodnoty reprezentované více (korelovanými) atributy, ne asymetrické

→ nový (asymetrický) binární atribut pro každou hodnotu kategorického atributu (včetně dvouhodnotového)

Diskretizace atributů

1. rozdělení (rozklad) hodnot spojitého atributu, po jejich setřízení, do n (disjunktních) intervalů specifikováním $n-1$ dělicích bodů
2. zobrazení (hodnot) intervalu na (stejnou) hodnotu nového kategorického atributu

? kolik intervalů/dělicích bodů a jakých hodnot $x_1, \ldots, x_{n-1}$

- $\{(x_0, x_1], (x_1, x_2], \ldots, (x_{n-1}, x_n)\}$, x_0, x_n příp. $\pm\infty$, nebo $x_0 < x \leq x_1 < x \leq x_2 \ldots x_{n-1} < x < x_n$

$\rightarrow$ zadaný počet stejně velkých intervalů – ovlivnění anomáliemi (outliers)

$\rightarrow$ zadaný počet intervalů s maximálně stejným počtem hodnot pro (dané) objekty

$\rightarrow$ intervaly blízkých hodnot $\rightsquigarrow$ shlukovací metoda, např. K-means; vizuálně, doménově specificky aj.

$\sim$ unsupervised diskretizace

Supervised diskretizace atributů

= využití hodnot cílového atributu = **tříd klasifikace (class labels)**

- cíl: maximální jedinečnost hodnoty cílového atributu (class label) pro objekty s hodnotou (diskretizovaného atributu) v intervalu (= „class label v intervalu“) = „čistota“ (purity) intervalu $\rightsquigarrow$ jak měřit?, minimálně velké intervaly

$\rightarrow$ interval = každá hodnota a opakované slučování „podobných“ sousedních – jak podobných (na základě class labels)?

- impurity i-tého intervalu = **entropie**: $e_i = \sum_{j=1}^{k} p_{ij} \log_2 p_{ij}$, k počet různých class labels, $p_{ij} = m_{ij}/m_i$ podíl class label j v intervalu i, m_{ij} počet výskytů class label j v intervalu i, m_i počet všech class labels v intervalu i
- entropie rozdělení (rozkladu): $e = \sum_{i=1}^{n} w_i e_i$, n počet intervalů, $w_i = m_i/m$ podíl class labels v intervalu i, m počet všech class labels

$\rightarrow$ interval = všechny setřízené hodnoty a opakovaně bisekce intervalu s nejvyšší entropií tak, aby rozdělení mělo nejmenší entropii, dokud ne zadaný počet intervalů nebo dostatečně nízká nebo již neklesající entropie – potřeba testovat jako dělicí bod (libovolný) bod mezi každými dvěma sousedními hodnotami intervalu pro objekty s různými class labels

Úpravy hodnot atributů

$\sim$ transformace proměnných (variable transformation)

$=$ stejná úprava všech hodnot atributu x pro všechny objekty

Jednoduché funkce

- pro číselné atributy např. $\log x$, $\sqrt{x}$, $1/x$ aj.

! změna rozsahu hodnot, jejich pravděpodobnostního rozložení, např. na normální, uspořádání (!), nedefinování funkce pro některé hodnoty aj.

Normalizace/standardizace

$=$ zajištění nějaké vlastnosti (číselných) hodnot

- např. běžná „normalizace“ $(x - \min(x))/(\max(x) - \min(x))$ pro $\min(x) = 0$, $\max(x) = 1$, (statistická) standardizace $(x - \overline{x})/s_x$, $\overline{x}$ průměr hodnot x, s_x směrodatná odchylka hodnot x pro $\overline{x} = 0$ a $s_x = 1$
- potřebné pro „férové“ porovnávání nebo kombinaci atributů, např. s různými rozsahy hodnot („škálami“), nebo „korektní“ porovnání objektů na základě atributů, např. podobnost/vzdálenost objektů

Podobnost a nepodobnost objektů

- používané v mnoha metodách (shlukování, klasifikace metodou nejbližšího souseda, detekce anomálií aj.) – převod dat do prostoru (ne)podobností a analýza v něm
- **podobnost s** objektů $\sim$ (numerická) míra stupně stejnosti objektů, typicky mezi 0 (žádná) a 1 (plná)
- **nepodobnost d** objektů $\sim$ (numerická) míra stupně různosti objektů, často $\sim$ **vzdálenost** – speciální případ, běžně od 0 (žádná) do ∞

$\rightsquigarrow$ **blízkost (proximity) p**

Transformace

- blízkost (podobnost) $\rightarrow [0, 1]$: pro podíl, typicky $p' = (p - \min(p))/(\max(p) - \min(p))$ pro konečné $\max(p) - \min(p)$, pro $p \in [0, \infty]$ např. $p' = p/(p + 1)$ – jiné (nelineární) vztahy mezi hodnotami, možná ztráta informace nebo jiný význam
- podobnost $\longleftrightarrow$ nepodobnost: typicky $d = 1 - s$ pro $s, d \in [0, 1]$, jinak např. $s = 1/(d + 1), s = e^{-d}, s = 1 - (d - \min(d))/(\max(d) - \min(d))$, obecně jakákoliv monotónní klesající funkce

Blízkost objektů

... kombinace blízkostí hodnot atributů objektů

- měla by respektovat typy atributů, může být potřeba normalizace/standardizace

Atributy:

- nominální: jen různost hodnot $\Rightarrow s = 1$ pro stejné hodnoty, jinak $s = 0$, d opačně
- ordinální: pořadí hodnot, bližší $\sim$ podobnější $\Rightarrow$ jednoznačné zobrazení hodnot na čísla se zachováním pořadí a $d =$ rozdíl odpovídajících čísel / rozsah čísel – jaká čísla (a rozdíly mezi nimi odpovídající rozdílům mezi hodnotami)?
- intervalové a poměrové: $d =$ (absolutní) rozdíl hodnot

- atributy stejného typu $\rightarrow$ nepodobnosti (vzdálenosti) a podobnosti objektů dále
- atributy různého typu $\rightarrow$ blízkosti hodnot atributů jednotlivě (nebo po skupinách stejného typu) a kombinace, typicky průměr – s výjimkou asymerických atributů s nulovými hodnotami pro objekty a atributů s chybějící hodnotou pro objekt
- různé váhy atributů

Nepodobnosti (vzdálenosti) objektů

- pro objekty ... vektory $\mathbf{x} = (x_1, \ldots, x_m)$ hodnot m (číselných) atributů
- **euklidovská vzdálenost**:

$$\sqrt{\sum_{k=1}^{m} (x_k - y_k)^2}$$

- **Minkowského vzdálenost**:

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

  - $r = 1$: city block (Manhattan, taxicab, L_1 norm), maximální, pro binární atributy **Hammingova vzdálenost** = počet atributů (bitů) s rozdílnými hodnotami pro objekty
  - $r = 2$: euklidovská (L_2 norm)
  - $r = \infty$: supremum (L_{max}, L_∞ norm), $= \lim_{r \to \infty} \ldots$, minimální

- atributy s různými rozsahy hodnot („škálami“) $\rightarrow$ (statistická) standardizace

Nepodobnosti (vzdálenosti) objektů

- **Mahalanobisova vzdálenost**: pro atributy s různými rozsahy (rozptyly) hodnot a korelacemi mezi sebou

$$\sqrt{(\mathbf{x}-\mathbf{y})^T\mathbf{S}^{-1}(\mathbf{x}-\mathbf{y})}$$

  $\mathbf{S}^{-1}$ inverzní kovariantní matice dat (matice kovariancí mezi každými dvěma atributy)
- pro spojité atributy, např. (hustá) číselná data

Míry (ne)podobnosti

Metrika

1. **pozitivita**: $d(\mathbf{x}, \mathbf{y}) \geq 0 \ \forall \mathbf{x}, \mathbf{y}$, $d(\mathbf{x}, \mathbf{y}) = 0 \leftrightarrow \mathbf{x} = \mathbf{y}$
2. **symetrie**: $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x}) \ \forall \mathbf{x}, \mathbf{y}$
3. **trojúhelníková nerovnost**: $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z}) \ \forall \mathbf{x}, \mathbf{y}, \mathbf{z}$

$\sim$ vzdálenost, např. Minkowského, Mahalanobisova

- ne metrika např. velikost (počet prvků) rozdílu množin: $d(A, B) = |A \setminus B| \rightarrow$ metrika?

Podobnost (typicky)

1. $s(\mathbf{x}, \mathbf{y}) \in [0, 1] \ \forall \mathbf{x}, \mathbf{y}$, $s(\mathbf{x}, \mathbf{y}) = 1 \leftrightarrow \mathbf{x} = \mathbf{y}$
2. **symetrie**: $s(\mathbf{x}, \mathbf{y}) = s(\mathbf{y}, \mathbf{x}) \ \forall \mathbf{x}, \mathbf{y}$
3. některé lze převést na metriky, např. Jaccard koeficient, kosinovou
4. ne symetrická např. podíl počtu binárních atributů s hodnotou 1 pro jednotlivé objekty $\rightarrow$ symetrická?

Podobnosti objektů

Podobnostní koeficienty

= podobnosti pro binární atributy, $\in [0, 1]$

- f_{xy} = počet atributů s hodnotou x pro $\mathbf{x}$ a y pro $\mathbf{y}$, $xy \in \{00, 01, 10, 11\}$
- **simple matching koeficient**:

$$\frac{f_{00} + f_{11}}{f_{00} + f_{01} + f_{10} + f_{11}}$$

- **Jaccard koeficient**: pro asymetrické binární atributy, např. (řídká) transakční data

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

Podobnosti objektů

Pro (asymetrické) číselné atributy, např. (řídká) document-term data, $\in [0, 1]$:

- **kosinová podobnost**: kosinus úhlu mezi (vektory) $\mathbf{x}$ a $\mathbf{y}$

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{||\mathbf{x}|| \, ||\mathbf{y}||} = \frac{\sum_{k=1}^{m} x_k y_k}{\sqrt{\sum_{k=1}^{m} x_k^2} \sqrt{\sum_{k=1}^{m} y_k^2}} = \frac{\mathbf{x}}{||\mathbf{x}||} \cdot \frac{\mathbf{y}}{||\mathbf{y}||}$$

- **rozšířený Jaccard (Tanimoto) koeficient**:

$$\frac{\mathbf{x} \cdot \mathbf{y}}{||\mathbf{x}||^2 + ||\mathbf{y}||^2 - \mathbf{x} \cdot \mathbf{y}}$$

Korelace objektů

= míra lineární závislosti mezi hodnotami atributů objektů ($x_k = ay_k + b$), $\in [-1, 1]$

- podobně i korelace mezi hodnotami atributů (napříč objekty)
- **Pearsonův korelační koeficient**:

$$corr(\mathbf{x}, \mathbf{y}) = \frac{s_{\mathbf{xy}}}{s_{\mathbf{x}} s_{\mathbf{y}}}$$

$s_{\mathbf{xy}}$ kovariance mezi $\mathbf{x}$ a $\mathbf{y}$, $s_{\mathbf{x}}$ směrodatná odchylka hodnot $\mathbf{x}$

= kosinová podobnost při $\overline{\mathbf{x}} = \overline{\mathbf{y}} = 0$

Zkoumání dat (data exploration) [SAMOSTUDIUM]

= předběžný průzkum dat pro prvotní charakteristiky $\Rightarrow$ pro výběr metod předzpracování a analýzy

- může „již něco odhalit“, např. vzory po vizualizaci
- použití pro porozumění a interpretaci výsledků analýzy, zejména vizualizace

$\rightarrow$ **souhrnné statistiky**: četnosti, percentily, průměr a medián, rozptyl a směrodatná odchylka, kovariance a korelace aj.

$\rightarrow$ **vizualizace**: histogram, grafy, diagramy, matice atd.

$\rightarrow$ **On-Line Analytical Processing (OLAP)** – data jako vícedimenzionální pole hodnot, souhrnné tabulky, agregace dat (přes dimenze nebo hodnoty)

$\sim$ část **explorativní analýzy dat** (Tukey, statistik, 1970s)

Souhrnné statistiky

Klasifikace

Regrese

= **prediktivní modelování spojitého** cílového atributu (**závislé proměnné**) y jako **(cílové) funkce**, **regresního modelu**, f jiných atributů (**nezávislých proměnných**) $x_1, \ldots, x_m$

$\rightarrow$ predikce (neznámé) hodnoty atributu y objektu z jeho (známých) hodnot atributů $x_1, \ldots, x_m$

- **univariate** = jeden atribut x
- **multivariate** = více atributů $x_1, \ldots, x_m$
- mnoho aplikací: modelování a předpovědi počasí, cen, prodeje, vývoje atd.

Lineární regrese

= **lineární** cílová funkce (model): $f(x_1, \ldots, x_m) = a_0 + a_1 x_1 + \ldots + a_m x_m \approx y$, $a_0, \ldots, a_m$ **regresní koeficienty**

- n objektů $\Rightarrow$ maticová reprezentace: objekty $\sim$ řádky, atributy $\mathbf{x_j} = (x_{1j}, \ldots, x_{nj})^T$, $j = 1, \ldots, m \sim$ sloupce (maticová data), cílový atribut $\mathbf{y} = (y_1, \ldots, y_n)^T \sim$ sloupec

$\rightarrow$ soustava lineárních rovnic $f(\mathbf{x_1}, \ldots, \mathbf{x_m}) = \mathbf{Xa} \approx \mathbf{y}$, $\mathbf{X} = (\mathbf{1}\ \mathbf{x_1}\ \ldots\ \mathbf{x_m})$ **design matrix**, $\mathbf{1} = (1, \ldots, 1)^T, \mathbf{a} = (a_0, \ldots, a_m)^T$:

$$\begin{aligned} f(x_{11}, \ldots, x_{1m}) = a_0 + a_1 x_{11} + \ldots + a_m x_{1m} &\approx y_1 \\ &\vdots \\ f(x_{n1}, \ldots, x_{nm}) = a_0 + a_1 x_{n1} + \ldots + a_m x_{nm} &\approx y_n \end{aligned}$$

- právě jedno (přesné) řešení $\mathbf{a} = \mathbf{X}^{-1}\mathbf{y}$, pokud existuje $\mathbf{X}^{-1}$ (při $n = m + 1$), jinak žádné nebo nekonečně mnoho – nelze použít, obvykle $n > m + 1$

Lineární regrese

→ nejbližší řešení $\mathbf{a}$ . . . minimalizace chyby mezi predikovanou $f(\mathbf{x_1}, \ldots, \mathbf{x_m})$ a skutečnou hodnotou cílového atributu $\mathbf{y}$ = **regresní problém**: **chybová funkce** – typicky squared error $E = \sum_i^n (y_i - f(x_{i1}, \ldots, x_{im}))^2 = \sum_i^n (y_i - (a_0 + a_1 x_{i1} + \ldots + a_m x_{im}))^2 \rightarrow$ **least squares problem + method**

- univariate: $E = \sum_i^n (y_i - (a_0 + a_1 x_i))^2 \rightarrow \frac{\partial E}{\partial a_0} = -2 \sum_i^n (y_i - (a_0 + a_1 x_i)) = 0$, $\frac{\partial E}{\partial a_1} = -2 \sum_i^n (y_i - (a_0 + a_1 x_i)) x_i = 0 \rightarrow$

$$\begin{pmatrix} n & \sum_i^n x_i \\ \sum_i^n x_i & \sum_i^n x_i^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} \sum_i^n y_i \\ \sum_i^n x_i y_i \end{pmatrix}$$

$a_0 = \overline{y} - a_1 \overline{x}, \quad a_1 = \sigma_{xy}/\sigma_{xx}, \qquad \sigma_{xy} = \sum_i^n (x_i - \overline{x})(y_i - \overline{y}), \sigma_{xx} = \sum_i^n (x_i - \overline{x})^2$
funkce (model): $f(x) = a_0 + a_1 x = \overline{y} + \sigma_{xy}/\sigma_{xx}(x - \overline{x}) \approx y$

- multivariate:

$$\begin{pmatrix} n & \sum_i^n x_i \\ \sum_i^n x_i & \sum_i^n x_i^2 \end{pmatrix} = \begin{pmatrix} \mathbf{1}^T\mathbf{1} & \mathbf{1}^T\mathbf{x} \\ \mathbf{x}^T\mathbf{1} & \mathbf{x}^T\mathbf{x} \end{pmatrix} = \mathbf{X}^T\mathbf{X}, \qquad \begin{pmatrix} \sum_i^n y_i \\ \sum_i^n x_i y_i \end{pmatrix} = \begin{pmatrix} \mathbf{1}^T\mathbf{y} \\ \mathbf{x}^T\mathbf{y} \end{pmatrix} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{X}^T\mathbf{X}\mathbf{a} = \mathbf{X}^T\mathbf{y} \rightarrow \mathbf{a} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Nelineární regrese

= nelineární cílová funkce (model): $f(x_1, \ldots, x_m) = a_0 + \sum_k a_k g_k(x_1, \ldots, x_m) \approx y$, g_k jakékoliv (diferencovatelné) funkce, typicky polynomické

- např. kvadratická: $f(x_1, x_2) = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_1 x_2 + a_4 x_1^2 + a_5 x_2^2$
  $\mathbf{X} = (\mathbf{1}\ \mathbf{x_1}\ \mathbf{x_2}\ \mathbf{x_1 x_2}\ \mathbf{x_1^2}\ \mathbf{x_2^2})$
- lze použít stejnou metodu (least squares) pro určení regresních koeficientů a_0, a_k:
  $\mathbf{X}^T\mathbf{X}\mathbf{a} = \mathbf{X}^T\mathbf{y} \rightarrow \mathbf{a} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

Vyhodnocení přesnosti (goodness of fit) regresního modelu

1. **chybová funkce** – squared error $E = \sum_i^n (y_i - f(x_{i1}, \ldots, x_{im}))^2 \in [0, \infty] \rightarrow$ minimální (0)
2. **koeficient determinace**: $\rightarrow$ maximální (1)

$$R^2 = \frac{S_r}{S_t} = \frac{\sum_i^n (f(x_{i1}, \ldots, x_{im}) - \overline{y})^2}{\sum_i^n (y_i - \overline{y})^2} \in [0, 1]$$

S_r regression (explained) sum of squares, S_t total sum of squares
$E = S_t - S_r$
univariate: $R^2 = \sigma_{xy}^2 / \sigma_{xx}\sigma_{yy}$, $corr(x, y) = \sigma_{xy} / \sqrt{\sigma_{xx}\sigma_{yy}}$

Klasifikace

= modelování **diskrétního** cílového atributu, **tříd klasifikace (class labels)**, jako **(cílové) funkce**, **klasifikačního modelu**, jiných atributů

$\sim$ rozřazení objektů (**instancí**) záznamových dat do jedné z definovaných **kategorií** (class labels), na základě (ostatních) atributů, např.

$\rightarrow$ **prediktivní modelování**: predikce class label pro objekt (s neznámou class label) na základě jeho (známých) hodnot atributů = **prediktorů**, např.

$\rightarrow$ **deskriptivní modelování**: rozlišení/kategorizace objektů (s neznámými class labels) do různých tříd na základě (známých) hodnot atributů objektů $\sim$ zjištění, jaké atributy (jejich hodnoty) určují třídy, např.

- typicky **nominální (binární)** class labels, pro ordinální **rank classification** – zohledňování pořadí class labels, také jiné vztahy, např. inkluzivní, mezi třídami/kategoriemi objektů
- mnoho aplikací: diagnóza nemocí, rozhodování o klientech (banky apod.), klasifikační atlasy, call centrum aj.

Klasifikace: příklad dat

ID	jméno	věk	partner	dětí	zaměstnání	příjem Kč	nemovitost	dávka
1	Adriana Dobrovičová	32	ano	5	ne	6 335	ne	ANO
2	František Smutný	58	ne	0	ano	9 055	ne	NE
3	Zuzana Nesmělá	25	ne	2	ne	4 625	ano	ANO
4	Diego Horváth	46	ano	6	ano	15 500	ne	NE
5	Eva Vopršálková	28	ne	1	ano	9 290	ne	ANO
6	Denisa Šťastná	34	ne	1	ne	14 836	ano	NE
7	Zdeněk Pokorný	57	ne	2	ne	6 060	ano	ANO
8	Esmeralda Balogová	19	ano	3	ne	1 420	ne	ANO
9	Radovan Bobek	43	ano	2	ne	14 600	ano	NE
10	Vladimíra Komínková	62	ne	0	ne	6 214	ne	ANO
11	Mojmír Zatuchlý	49	ano	3	ano	16 150	ne	NE
12	Nikola Gáborová	31	ano	7	ne	10 375	ne	?

atributy: věk, partner ve společné domácnosti, počet nezletilých dětí, zaměstnání, průměrný měsíční příjem za poslední kvartál v Kč, vlastní nemovitost

cílový atribut (třídy klasifikace, class labels, kategorie): přidělení sociální dávky ANO/NE

Klasifikační model

- vytvářený ze vstupních dat klasifikační metodou/technikou (**klasifikátor, classifier**) = **učení** (indukce)
- např. rozhodovací stromy, pravidlový (rule-based), naivní bayesovský (naıve Bayes), umělé neuronové sítě, support vector machines aj.

! **klasifikační problém**: co nejlepší modelování vstupních dat, ale zároveň i správná predikce class label pro nové objekty = **aplikování** (dedukce) – **schopnost generalizace**

- data (objekty se známou class label) pro učení = **training set**, data (objekty s neznámou class label) pro aplikování = **test set**
- vyhodnocení výkonnosti (performance) na základě počtů správných a nesprávných predikcí → **confusion matrix** (matice záměn, chybová), např. pro binární class labels:

		predikovaná třída	
		0	1
skutečná třída	0	f_{00}	f_{01}
	1	f_{10}	f_{11}

f_{ij} počet objektů s class label i predikovanou jako j

Klasifikační model

Základní ukazatele výkonnosti (performance)

- **accuracy (přesnost)**: $\rightarrow$ nejvyšší

$$\frac{\text{počet správných predikcí}}{\text{počet všech predikcí}} = \frac{\sum_i f_{ii}}{\sum_{i,j} f_{ij}}$$

- **error rate (chybovost)**: $\rightarrow$ nejnižší

$$\frac{\text{počet nesprávných predikcí}}{\text{počet všech predikcí}} = \frac{\sum_{i \neq j} f_{ij}}{\sum_{i,j} f_{ij}}$$

Rozhodovací stromy

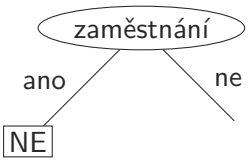
? predikce class label/zařazení do třídy pro (nový) objekt = klasifikace: série vhodných otázek, v závislosti na odpovědích na předchozí otázky, na hodnoty atributů objektu, např. zaměstnání?: ano , ne

→ organizace otázek a odpovědí ve formě (rozhodovacího) stromu, např.

Rozhodovací stromy

? predikce class label/zařazení do třídy pro (nový) objekt = klasifikace: série vhodných otázek, v závislosti na odpovědích na předchozí otázky, na hodnoty atributů objektu, např. zaměstnání?: ano $\Rightarrow$ dávka NE, ne

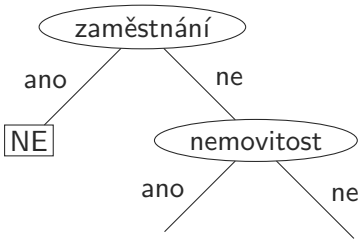
$\rightarrow$ organizace otázek a odpovědí ve formě (rozhodovacího) stromu, např.



Rozhodovací stromy

? predikce class label/zařazení do třídy pro (nový) objekt = klasifikace: série vhodných otázek, v závislosti na odpovědích na předchozí otázky, na hodnoty atributů objektu, např. zaměstnání?: ano $\Rightarrow$ dávka NE, ne $\rightarrow$ nemovitost?: ne , ano

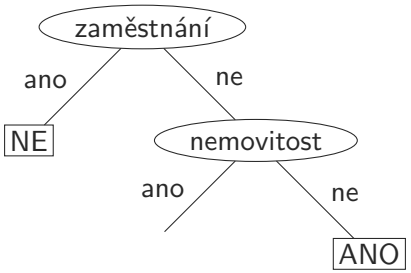
$\rightarrow$ organizace otázek a odpovědí ve formě (rozhodovacího) stromu, např.



Rozhodovací stromy

? predikce class label/zařazení do třídy pro (nový) objekt = klasifikace: série vhodných otázek, v závislosti na odpovědích na předchozí otázky, na hodnoty atributů objektu, např. zaměstnání?: ano $\Rightarrow$ dávka NE, ne $\rightarrow$ nemovitost?: ne $\Rightarrow$ ANO, ano

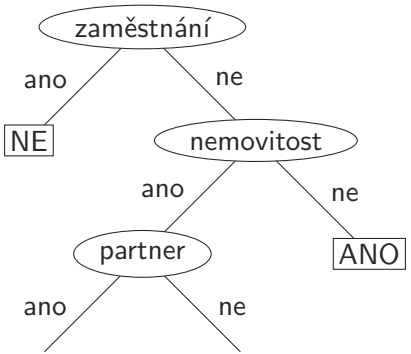
$\rightarrow$ organizace otázek a odpovědí ve formě (rozhodovacího) stromu, např.



Rozhodovací stromy

? predikce class label/zařazení do třídy pro (nový) objekt = klasifikace: série vhodných otázek, v závislosti na odpovědích na předchozí otázky, na hodnoty atributů objektu, např. zaměstnání?: ano ⇒ dávka NE, ne → nemovitost?: ne ⇒ ANO, ano → partner?: ano , ne

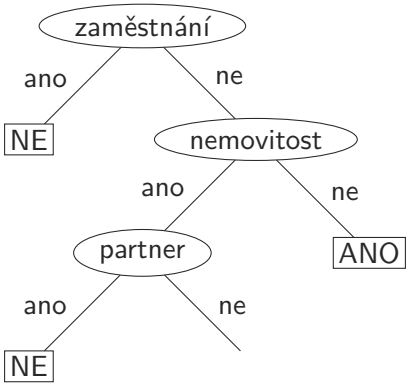
→ organizace otázek a odpovědí ve formě (rozhodovacího) stromu, např.



Rozhodovací stromy

? predikce class label/zařazení do třídy pro (nový) objekt = klasifikace: série vhodných otázek, v závislosti na odpovědích na předchozí otázky, na hodnoty atributů objektu, např. zaměstnání?: ano ⇒ dávka NE, ne → nemovitost?: ne ⇒ ANO, ano → partner?: ano ⇒ NE, ne

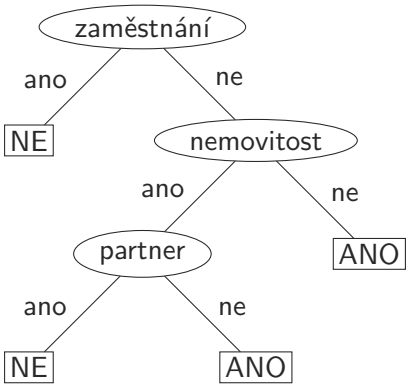
→ organizace otázek a odpovědí ve formě (rozhodovacího) stromu, např.



Rozhodovací stromy

? predikce class label/zařazení do třídy pro (nový) objekt = klasifikace: série vhodných otázek, v závislosti na odpovědích na předchozí otázky, na hodnoty atributů objektu, např. zaměstnání?: ano $\Rightarrow$ dávka NE, ne $\rightarrow$ nemovitost?: ne $\Rightarrow$ ANO, ano $\rightarrow$ partner?: ano $\Rightarrow$ NE, ne $\Rightarrow$ ANO

$\rightarrow$ organizace otázek a odpovědí ve formě (rozhodovacího) stromu, např.



Rozhodovací stromy

= hierarchická (stromová) struktura:

- **vnitřní uzly** – jedna příchozí hrana, dvě a více odchozích hran, přiřazená **podmínka/test nad atributy**, typicky jedním
- **listové uzly** – jedna příchozí hrana, žádná odchozí hrana, přiřazená class label
- **kořenový uzel** = vnitřní nebo listový, žádná příchozí hrana
- **orientované hrany** mezi uzly – od rodičovského k potomkům, přiřazený **výsledek testu** u rodičovského uzlu, typicky hodnota(y) atributu

→ klasifikace objektu: počínaje kořenovým uzlem opakovaně aplikace na objekt testu u aktuálního uzlu a přesun po hraně k uzlu potomka vybrané/ho dle výsledku testu, až přesun do listového uzlu s class label, např.

Vytvoření (indukce) rozhodovacího stromu

- exponenciálně mnoho různě přesných (accurate) stromů pro danou množinu atributů

→ obvyklá **greedy strategie** pro suboptimálně přesný strom – série lokálně optimálních stanovení testu u uzlu, typicky výběru atributu

Huntův algoritmus

- základ mnoha používaných, např. ID3, C4.5, CART

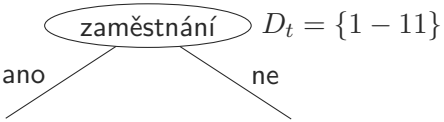
= rekurzivní rozklad (**split**) množiny vstupních (trénovacích) objektů na (disjunktní) podmnožiny s cílem stejné class label u objektů

- D_t množina vstupních (trénovacích) objektů asociovaných s uzlem t stromu, na počátku všechny ($D_t \neq \emptyset$)

1. jestliže všechny objekty z D_t mají stejnou class label y, vrať listový uzel t s y,
2. jinak stanov **test nad atributy** (typicky jedním), rozkládající (splitting) D_t na podmnožiny objektů se stejným výsledkem testu (typicky hodnota(y) atributu) a
   - vytvoř vnitřní uzel t s tímto testem a, pro každý různý výsledek testu, s odchozí hranou s výsledkem testu,
   - na každou hranu napoj výsledek tohoto algoritmu pro $D_{t'}$ = podmnožina objektů s výsledkem testu u hrany a
   - vrať t

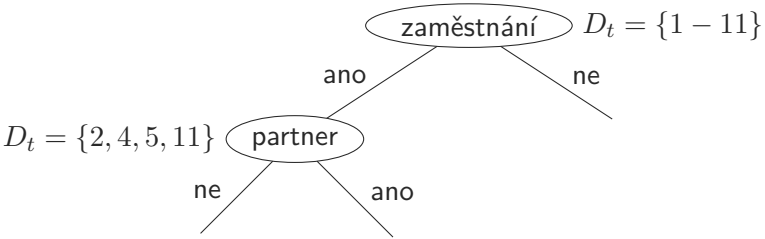
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



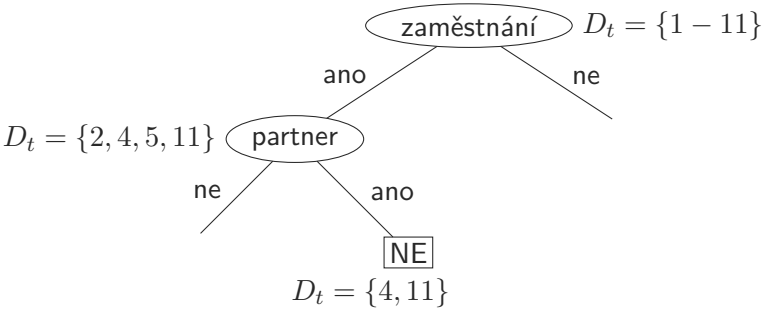
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



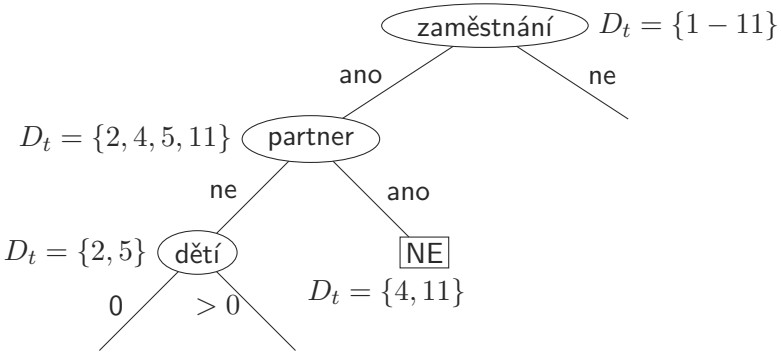
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



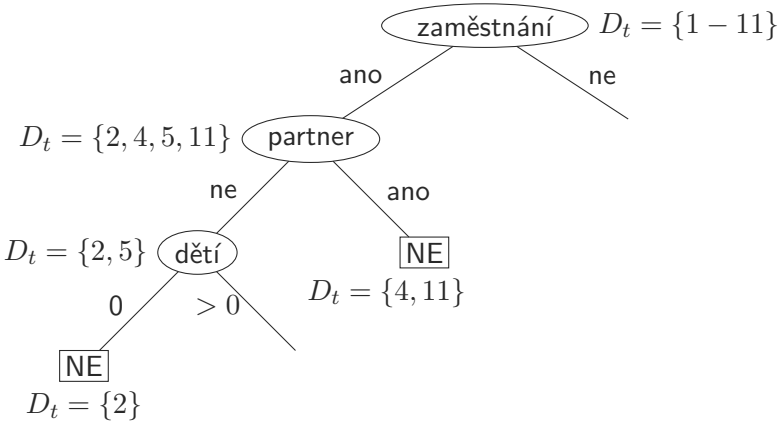
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



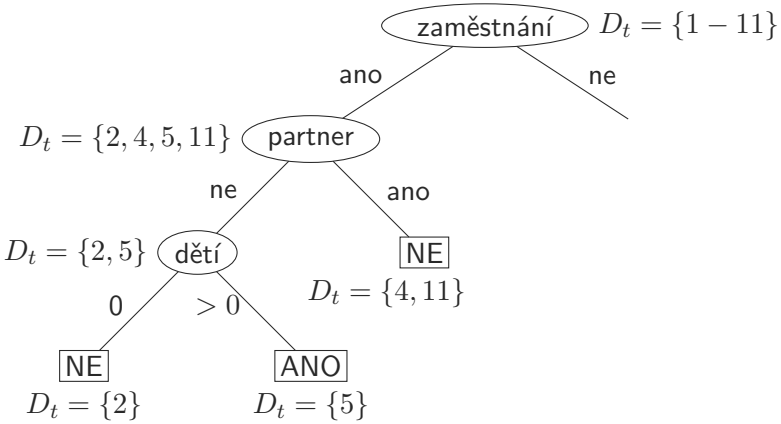
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



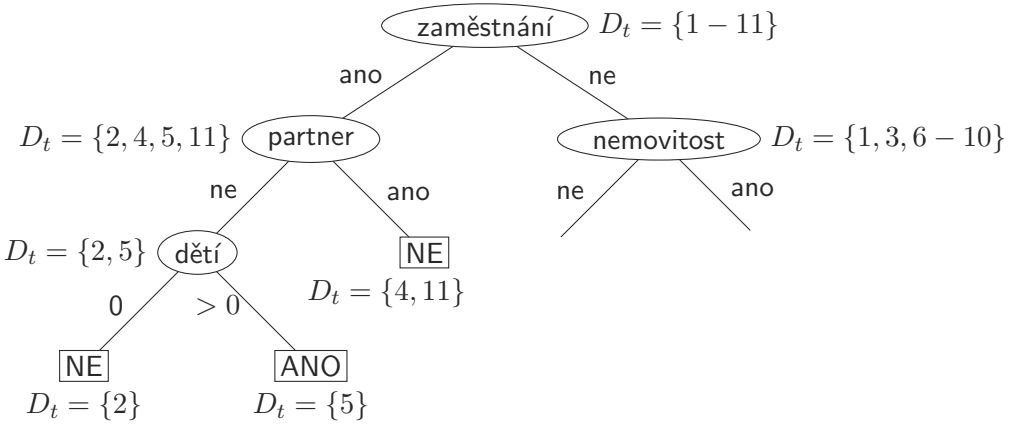
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



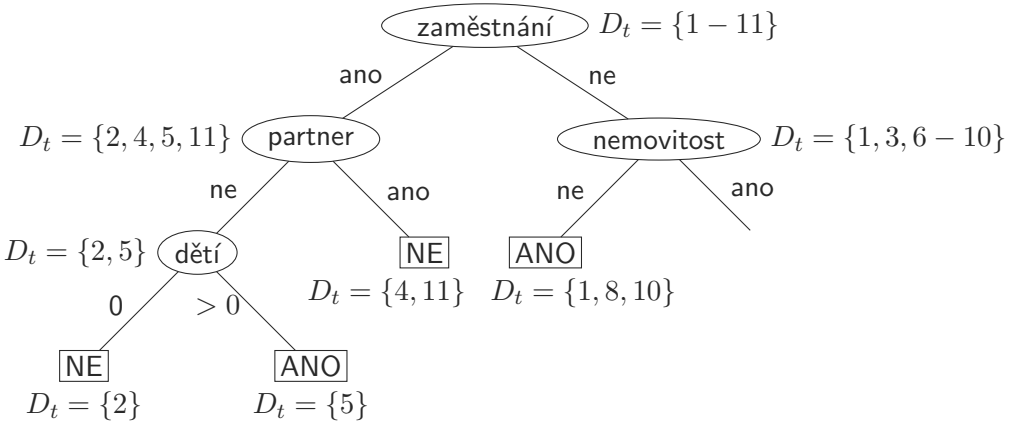
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



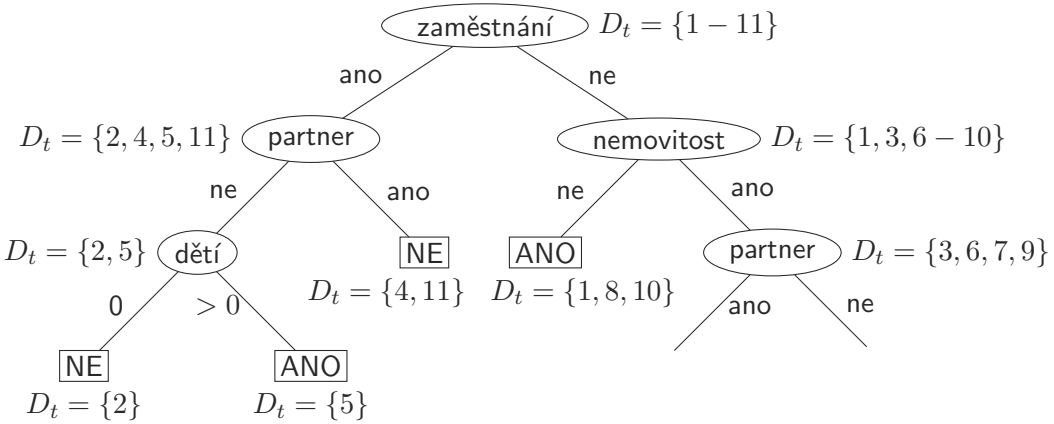
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



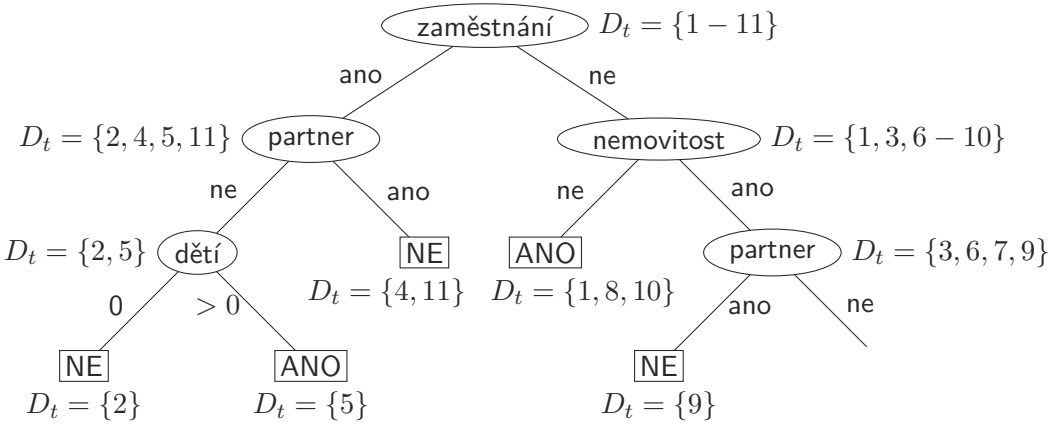
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



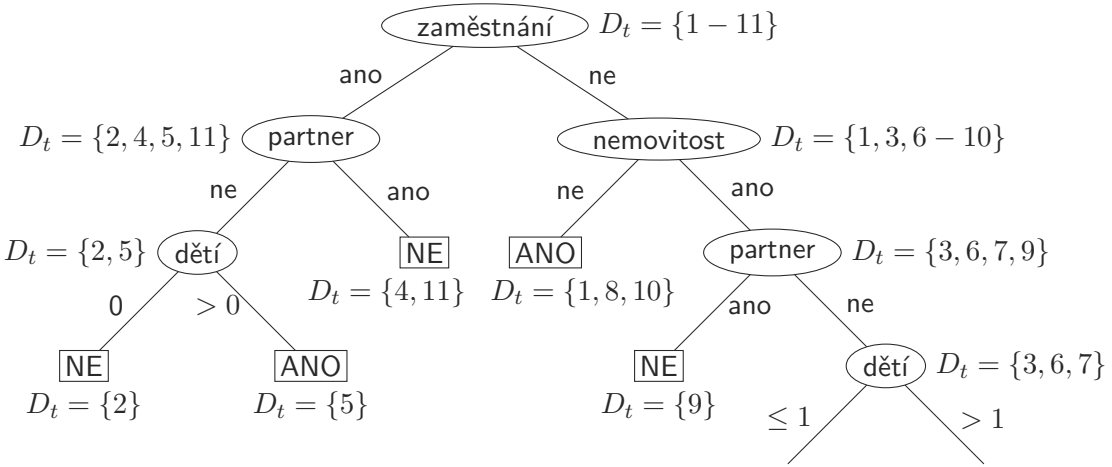
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



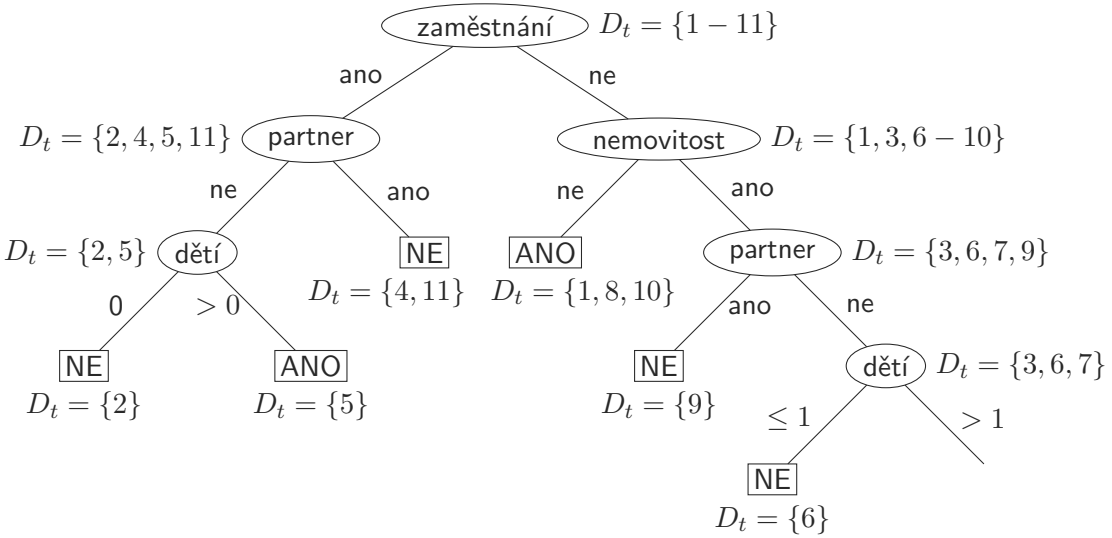
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



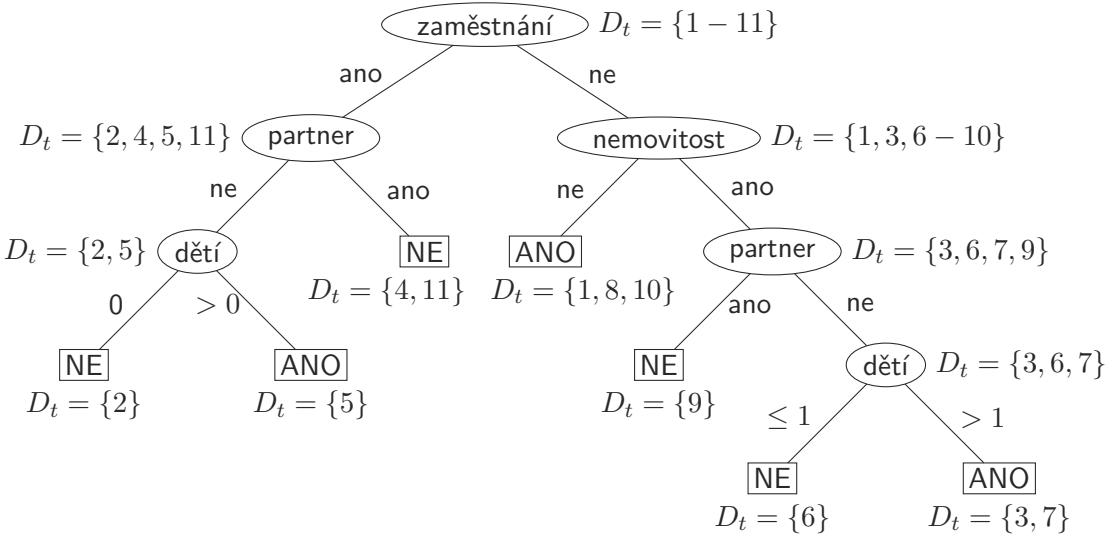
Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus: příklad



Vytvoření (indukce) rozhodovacího stromu

Huntův algoritmus

!! test nad atributy pokrývající všechny hodnoty atributů v testu, nejen ty v (aktuální) D_t – generalizace

? D_t z kroku 1 prázdná $\rightarrow$ class label y přiřazená $t =$ (obvykle) majoritní class label objektů asociovaných s rodičovským uzlem

? jediný výsledek testu nad atributy (stejný pro všechny objekty z D_t), tj. ne rozklad D_t $\rightarrow$ uzel t listový s (obvykle) majoritní class label objektů z D_t

! stanovení testu nad atributy – pro různé typy atributů, vyhodnocení „výhodnosti“ testu

- jiné podmínky pro listový uzel, např. minimální velikost D_t – listové uzly (dříve) místo některých vnitřních $\rightarrow$ zlepšení schopnosti generalizace
- po vytvoření „prořezání“ (**pruning**) pro zamezení problému **přeučení (overfitting)** $\rightarrow$ zlepšení schopnosti generalizace, viz dále

Test nad atributy

- binární: která hodnota? $\rightarrow$ dva výsledky = dvě hodnoty
- nominální:
  1. která hodnota? $\rightarrow k$ výsledků $= k$ hodnot (multiway split)
  2. které hodnoty (výběr)? $\rightarrow$ méně než k výsledků = disjunktních podmnožin k hodnot, typicky dvou (binary split), např. CART – $2^{k-1} - 1$ možností
- ordinální: jako nominální, podmnožiny „zachovávající pořadí hodnot“ (není $x < y$ a současně $x' > y'$ pro $x, x' \in X$ a $y, y' \in Y$)
- spojité: jaké hodnoty (porovnání s konkrétními)? $\rightarrow$ výsledky = disjunktní intervaly setřízených hodnot vymezené dělicími body, typicky dva (binary split) – kolik?, jaké? $\rightsquigarrow$ supervised diskretizace atributu a ordinální (s příp. jinou měrou impurity intervalu, viz dále)

? které atributy, binary nebo multiway split, kolik intervalů (spojitý atribut), kombinace (testů)

$\rightarrow$ typicky jeden atribut (dělicí, **splitting attribute**) – který?

Test nad atributy $\sim$ rozřazení objektů

* rozklad (split) množiny D_t objektů asociovaných s uzlem t stromu

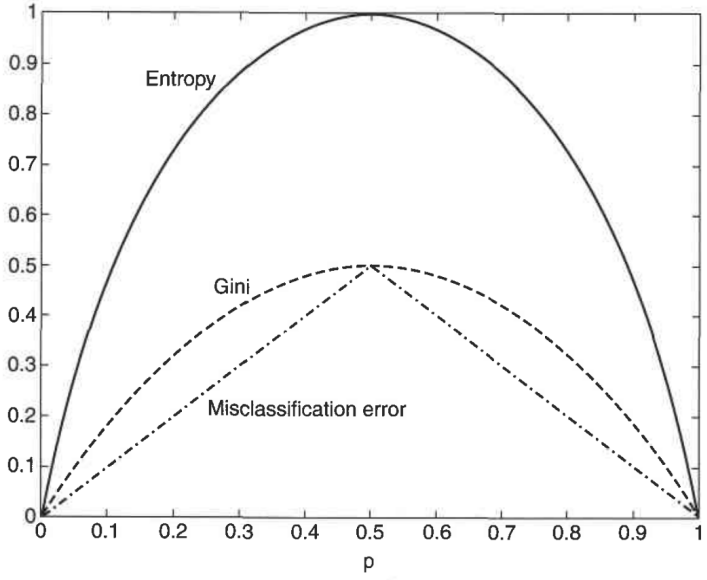
$\rightarrow$ „výhodnost“ testu $\sim$ „výhodnost“ rozkladu (dělicí kritérium, splitting criterion)

$+$ cíl: stejná class label u všech objektů v podmnožinách $D_{t'}$ rozkladu (asociovaných s novými uzly t' stromu)

$\rightarrow$ měření jedinečnosti class label u objektů asociovaných s uzlem stromu $=$ „čistota“ (purity) uzlu $\rightarrow$ maximalizace

- $p(i|t) = m(i|t)/m(t)$ podíl ($\approx$ pravděpodobnost) objektů asociovaných s uzlem t s class label i, $m(i|t)$ počet objektů asociovaných s uzlem t s class label i, $m(t)$ počet objektů asociovaných s uzlem t – purity $t \approx$ rozdílnost (neuniformnost) $p(i|t)$
- **míry impurity** $I(t)$ uzlu $t \rightarrow$ minimalizace, nejčastější: např. pro $D_t = \{1 - 11\}$
  - **entropie**$(t) = -\sum_i p(i|t) \log_2 p(i|t)$ $\quad -(\frac{6}{11} \log_2 \frac{6}{11} + \frac{5}{11} \log_2 \frac{5}{11}) \doteq 0.994$
  - **Gini index**$(t) = 1 - \sum_i [p(i|t)]^2$ $\quad 1 - ((\frac{6}{11})^2 + (\frac{5}{11})^2) \doteq 0.496$
  - **klasifikační chyba**$(t) = 1 - \max_i p(i|t)$ $\quad 1 - \max(\frac{6}{11}, \frac{5}{11}) \doteq 0.364$

Test nad atributy $\sim$ rozřazení objektů



Test nad atributy $\sim$ rozřazení objektů

- „výhodnost“ testu $\sim$ „výhodnost“ rozkladu D_t – dělicí kritérium **gain** $\rightarrow$ maximalizace:

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

  $N(t)$ počet objektů asociovaných s uzlem t stromu
  ... rozdíl impurity (rodičovského) uzlu t, před rozkladem, s váženým součtem impurity nových uzlů t' (potomků), po rozkladu
  information gain Δ_{info} = gain pro entropii jako míru $I(t)$ impurity uzlu

$\Rightarrow$ splitting attribute = atribut s maximální gain

Např. pro $D_t = \{1 - 11\}$, dělicí atribut = zaměstnání: hodnoty ano, ne $\rightarrow$
$D_{t'_{ano}} = \{2, 4, 5, 11\}, D_{t'_{ne}} = \{1, 3, 6 - 10\}$:
$I(t'_{ano}) = -(\frac{1}{4}\log_2\frac{1}{4} + \frac{3}{4}\log_2\frac{3}{4}) \doteq 0.811$, $I(t'_{ne}) = -(\frac{5}{7}\log_2\frac{5}{7} + \frac{2}{7}\log_2\frac{2}{7}) \doteq 0.863$
$\Delta_{info}(t) = 0.994 - (\frac{4}{11} \cdot 0.811 + \frac{7}{11} \cdot 0.863) = 0.15$

Vytvoření (indukce) rozhodovacího stromu: příklad

$D_t = \{1 - 11\}$

věk:	19	25	28	32	34	43	46	49	57	58	62	$\rightarrow \leq 33, > 33$
	ANO	ANO	ANO	ANO	NE	NE	NE	NE	ANO	NE	ANO	

$D_{t'_{\leq 33}} = \{1, 3, 5, 8\}, D_{t'_{>33}} = \{2, 4, 6, 7, 9 - 11\}$

$I(t'_{\leq 33}) = -(\frac{4}{4}\log_2\frac{4}{4} + \frac{0}{4}\log_2\frac{0}{4}) = 0,\ I(t'_{>33}) = -(\frac{2}{7}\log_2\frac{2}{7} + \frac{5}{7}\log_2\frac{5}{7}) \doteq 0.863$

$\Delta_{info}(t) = 0.994 - (\frac{4}{11}\cdot 0 + \frac{7}{11}\cdot 0.863) = 0.445$

Vytvoření (indukce) rozhodovacího stromu: příklad

$D_t = \{1 - 11\}$
věk: $\Delta_{info}(t) = 0.445$
partner: ano, ne
$D_{t'_{ano}} = \{1, 4, 8, 9, 11\}, D_{t'_{ne}} = \{2, 3, 5 - 7, 10\}$
$I(t'_{ano}) = -(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}) \doteq 0.971$, $I(t'_{ne}) = -(\frac{4}{6}\log_2\frac{4}{6} + \frac{2}{6}\log_2\frac{2}{6}) \doteq 0.918$
$\Delta_{info}(t) = 0.994 - (\frac{5}{11} \cdot 0.971 + \frac{6}{11} \cdot 0.918) = 0.052$

Vytvoření (indukce) rozhodovacího stromu: příklad

$D_t = \{1 - 11\}$
věk: $\Delta_{info}(t) = 0.445$
partner: $\Delta_{info}(t) = 0.052$
děti: $\{0\}$, $\{1, 2\}$, > 2
$D_{t'_{\{0\}}} = \{2, 10\}, D_{t'_{\{1,2\}}} = \{3, 5 - 7, 9\}, D_{t'_{>2}} = \{1, 4, 8, 11\}$
$I(t'_{\{0\}}) = -(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}) = 1$, $I(t'_{\{1,2\}}) = -(\frac{3}{5}\log_2\frac{3}{5} + \frac{2}{5}\log_2\frac{2}{5}) \doteq 0.971$,
$I(t'_{>2}) = -(\frac{2}{4}\log_2\frac{2}{4} + \frac{2}{4}\log_2\frac{2}{4}) = 1$
$\Delta_{info}(t) = 0.994 - (\frac{2}{11}\cdot 1 + \frac{5}{11}\cdot 0.971 + \frac{4}{11}\cdot 1) = 0.007$

Vytvoření (indukce) rozhodovacího stromu: příklad

$D_t = \{1 - 11\}$
věk: $\Delta_{info}(t) = 0.445$
partner: $\Delta_{info}(t) = 0.052$
děti: $\Delta_{info}(t) = 0.007$
zaměstnání: ano, ne
$D_{t'_{ano}} = \{2, 4, 5, 11\}, D_{t'_{ne}} = \{1, 3, 6 - 10\}$:
$I(t'_{ano}) = -(\frac{1}{4}\log_2\frac{1}{4} + \frac{3}{4}\log_2\frac{3}{4}) \doteq 0.811$, $I(t'_{ne}) = -(\frac{5}{7}\log_2\frac{5}{7} + \frac{2}{7}\log_2\frac{2}{7}) \doteq 0.863$
$\Delta_{info}(t) = 0.994 - (\frac{4}{11} \cdot 0.811 + \frac{7}{11} \cdot 0.863) = 0.15$

Vytvoření (indukce) rozhodovacího stromu: příklad

$D_t = \{1 - 11\}$
věk: $\Delta_{info}(t) = 0.445$
partner: $\Delta_{info}(t) = 0.052$
děti: $\Delta_{info}(t) = 0.007$
zaměstnání: $\Delta_{info}(t) = 0.15$
nemovitost: ano, ne
$D_{t'_{ano}} = \{3, 6, 7, 9\}, D_{t'_{ne}} = \{1, 2, 4, 5, 8, 10, 11\}$:
$I(t'_{ano}) = -(\frac{2}{4}\log_2\frac{2}{4} + \frac{2}{4}\log_2\frac{2}{4}) = 1,\ I(t'_{ne}) = -(\frac{4}{7}\log_2\frac{4}{7} + \frac{3}{7}\log_2\frac{3}{7}) \doteq 0.985$
$\Delta_{info}(t) = 0.994 - (\frac{4}{11}\cdot 1 + \frac{7}{11}\cdot 0.985) = 0.004$

Vytvoření (indukce) rozhodovacího stromu: příklad

$D_t = \{1 - 11\}$

věk: $\Delta_{info}(t) = 0.445$

partner: $\Delta_{info}(t) = 0.052$

děti: $\Delta_{info}(t) = 0.007$

zaměstnání: $\Delta_{info}(t) = 0.15$

nemovitost: $\Delta_{info}(t) = 0.004$

příjem:

1 420	4 625	6 060	6 214	6 335	9 055	9 290	14 600	14 836	15 500	16 150
ANO	ANO	ANO	ANO	ANO	NE	ANO	NE	NE	NE	NE

$\rightarrow \leq 9,000, > 9,000$

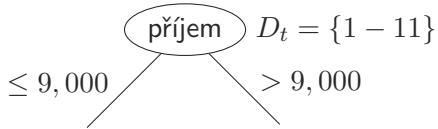
$D_{t'_{\leq 9,000}} = \{1, 3, 7, 8, 10\}, D_{t'_{>9,000}} = \{2, 4 - 6, 9, 11\}$

$I(t'_{\leq 9,000}) = -(\frac{5}{5}\log_2\frac{5}{5} + \frac{0}{5}\log_2\frac{0}{5}) = 0,\ I(t'_{>9,000}) = -(\frac{1}{6}\log_2\frac{1}{6} + \frac{5}{6}\log_2\frac{5}{6}) \doteq 0.65$

$\Delta_{info}(t) = 0.994 - (\frac{5}{11} \cdot 0 + \frac{6}{11} \cdot 0.65) = 0.639$

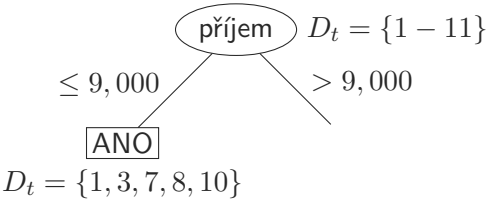
Vytvoření (indukce) rozhodovacího stromu: příklad

$D_t = \{1 - 11\}$
věk: $\Delta_{info}(t) = 0.445$
partner: $\Delta_{info}(t) = 0.052$
děti: $\Delta_{info}(t) = 0.007$
zaměstnání: $\Delta_{info}(t) = 0.15$
nemovitost: $\Delta_{info}(t) = 0.004$
příjem: $\Delta_{info}(t) = 0.639$



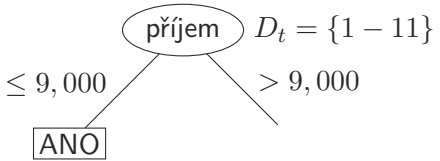
Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $\leq 9,000$
$D_t = \{1, 3, 7, 8, 10\}$
dávka: ANO



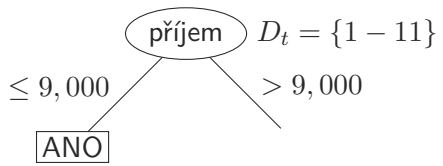
Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $> 9,000$
$D_t = \{2, 4-6, 9, 11\}$
věk: $\leq 33, > 33$
$D_{t'_{\leq 33}} = \{5\}, D_{t'_{>33}} = \{2, 4, 6, 9, 11\}$
$I(t'_{\leq 33}) = -(\frac{1}{1}\log_2\frac{1}{1} + \frac{0}{1}\log_2\frac{0}{1}) = 0$, $I(t'_{>33}) = -(\frac{0}{5}\log_2\frac{0}{5} + \frac{5}{5}\log_2\frac{5}{5}) = 0$
$\Delta_{info}(t) = 0.639 - (\frac{1}{6}\cdot 0 + \frac{5}{6}\cdot 0) = 0.639$



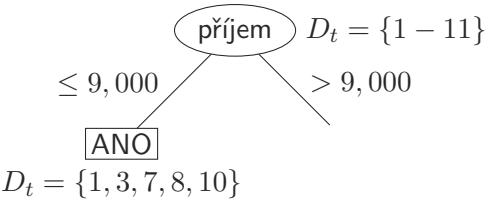
Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $> 9,000$
$D_t = \{2, 4-6, 9, 11\}$
věk: $\Delta_{info}(t) = 0.639$
partner: ano, ne
$D_{t'_{ano}} = \{4, 9, 11\}, D_{t'_{ne}} = \{2, 5, 6\}$
$I(t'_{ano}) = -(\frac{0}{3}\log_2\frac{0}{3} + \frac{3}{3}\log_2\frac{3}{3}) =$ 0, $I(t'_{ne}) = -(\frac{1}{3}\log_2\frac{1}{3} + \frac{2}{3}\log_2\frac{2}{3}) \doteq 0.918$
$\Delta_{info}(t) = 0.639 - (\frac{3}{6}\cdot 0 + \frac{3}{6}\cdot 0.918) = 0.18$



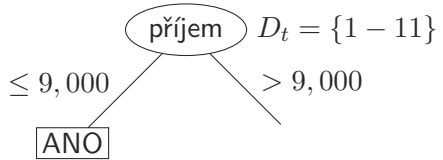
Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $> 9,000$
$D_t = \{2, 4-6, 9, 11\}$
věk: $\Delta_{info}(t) = 0.639$
partner: $\Delta_{info}(t) = 0.18$
děti: $\{0\}$, $\{1,2\}$, > 2
$D_{t'_{\{0\}}} = \{2\}, D_{t'_{\{1,2\}}} = \{5,6,9\}, D_{t'_{>2}} = \{4,11\}$
$I(t'_{\{0\}}) = -(\frac{0}{1}\log_2\frac{0}{1} + \frac{1}{1}\log_2\frac{1}{1}) = 0$, $I(t'_{\{1,2\}}) = -(\frac{1}{3}\log_2\frac{1}{3} + \frac{2}{3}\log_2\frac{2}{3}) \doteq 0.918$, $I(t'_{>2}) = -(\frac{0}{2}\log_2\frac{0}{2} + \frac{2}{2}\log_2\frac{2}{2}) = 0$
$\Delta_{info}(t) = 0.639 - (\frac{1}{6}\cdot 0 + \frac{3}{6}\cdot 0.918 + \frac{2}{6}\cdot 0) = 0.18$



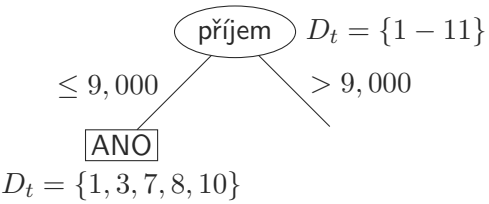
Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $> 9,000$
$D_t = \{2, 4-6, 9, 11\}$
věk: $\Delta_{info}(t) = 0.639$
partner: $\Delta_{info}(t) = 0.18$
děti: $\Delta_{info}(t) = 0.18$
zaměstnání: ano, ne
$D_{t'_{ano}} = \{2, 4, 5, 11\}, D_{t'_{ne}} = \{6, 9\}$: $D_t = \{1, 3, 7, 8, 10\}$
$I(t'_{ano}) = -(\frac{1}{4}\log_2\frac{1}{4} + \frac{3}{4}\log_2\frac{3}{4}) \doteq 0.811$, $I(t'_{ne}) = -(\frac{0}{2}\log_2\frac{0}{2} + \frac{2}{2}\log_2\frac{2}{2}) = 0$
$\Delta_{info}(t) = 0.639 - (\frac{4}{6}\cdot 0.811 + \frac{2}{6}\cdot 0) = 0.098$



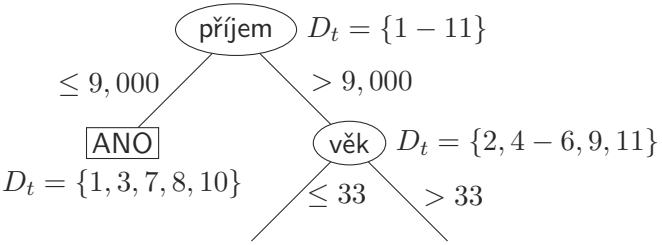
Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $> 9,000$
$D_t = \{2, 4-6, 9, 11\}$
věk: $\Delta_{info}(t) = 0.639$
partner: $\Delta_{info}(t) = 0.18$
děti: $\Delta_{info}(t) = 0.18$
zaměstnání: $\Delta_{info}(t) = 0.098$
nemovitost: ano, ne
$D_{t'_{ano}} = \{6, 9\}, D_{t'_{ne}} = \{2, 4, 5, 11\}$:
$I(t'_{ano}) = -(\frac{0}{2}\log_2\frac{0}{2} + \frac{2}{2}\log_2\frac{2}{2}) = 0$, $I(t'_{ne}) = -(\frac{1}{4}\log_2\frac{1}{4} + \frac{3}{4}\log_2\frac{3}{4}) \doteq 0.811$
$\Delta_{info}(t) = 0.639 - (\frac{2}{6}\cdot 0 + \frac{4}{6}\cdot 0.811) = 0.098$



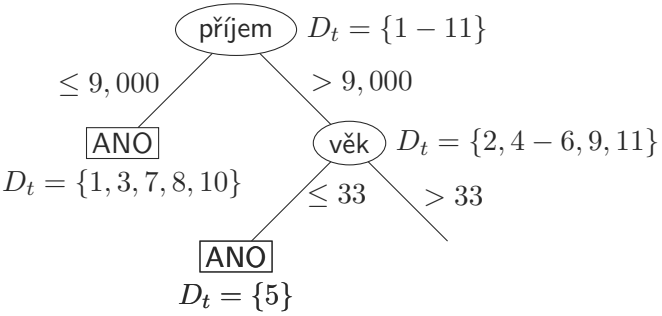
Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $> 9,000$
$D_t = \{2, 4-6, 9, 11\}$
věk: $\Delta_{info}(t) = 0.639$
partner: $\Delta_{info}(t) = 0.18$
děti: $\Delta_{info}(t) = 0.18$
zaměstnání: $\Delta_{info}(t) = 0.098$
nemovitost: $\Delta_{info}(t) = 0.098$



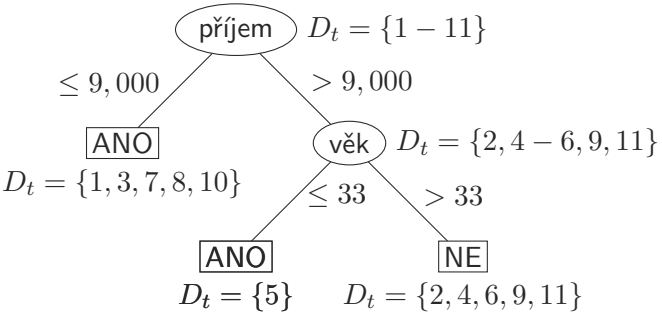
Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $> 9,000$, věk: ≤ 33
$D_t = \{5\}$
dávka: ANO



Vytvoření (indukce) rozhodovacího stromu: příklad

příjem: $> 9,000$, věk: > 33
$D_t = \{2, 4, 6, 9, 11\}$
dávka: NE



Test nad atributy $\sim$ rozřazení objektů

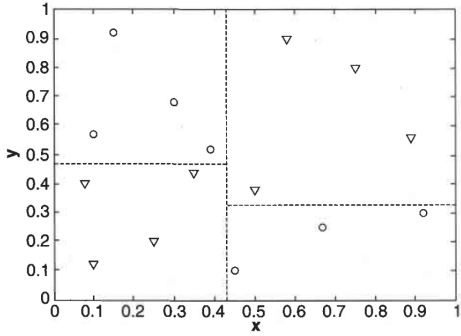
- testy nad jedním atributem = rozklady množiny / separace objektů na (disjunktní) podmnožiny s hranicemi (**decision boundaries**) nezávislými na jiných atributech (tzv. rektilineárními), např. $x < x_1$ a $y < y_1, y_2$

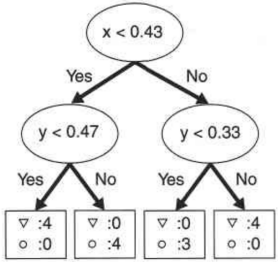
$\Rightarrow$ problém s klasifikací (separací) objektů s různými class labels s hranicemi mezi nimi závislými současně na více atributech, např. $x + y < 1$ nebo některé booleovské funkce, např. parita $\rightarrow$ špatná generalizace (potřeba „plný“ strom)

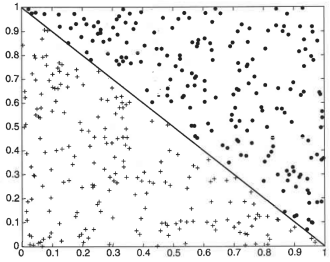
$\rightarrow$ **oblique rozhodovací stromy** = test nad více atributy (aritmetické, logické funkce atributů), ale výpočetně náročnější

$\rightarrow$ **constructive induction** = před indukcí tvorba nových, pro klasifikace lepších, atributů jako (aritmetických, logických) kombinací původních atributů (= feature creation/construction), ale potenciálně redundantních

Test nad atributy $\sim$ rozřazení objektů







Test nad atributy $\sim$ rozřazení objektů

? binary nebo multiway split, kolik intervalů (spojitý atribut) – jaká gain?

Test nad atributy $\sim$ rozřazení objektů

? binary nebo multiway split, kolik intervalů (spojitý atribut) – jaká gain?

- více výsledků testu $\Rightarrow$ menší podmnožiny rozkladu množiny objektů asociovaných s uzlem stromu $\rightsquigarrow$ častěji nižší (nulové) impurity uzlů (potomků) s asociovanými objekty v podmnožinách (objekty mají častěji stejnou class label) $\Rightarrow$ vyšší gain, např.
  pro $D_t = \{1 - 11\}$, dělicí atribut = dětí: hodnoty $\{0\}$, $> 0 \rightarrow D_{t'_{\{0\}}} = \{2, 10\}$,
  $D_{t'_{>0}} = \{1, 3 - 9, 11\}$
  $I(t'_{\{0\}}) = -(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}) = 1$, $I(t'_{>0}) = -(\frac{5}{9}\log_2\frac{5}{9} + \frac{4}{9}\log_2\frac{4}{9}) \doteq 0.991$
  $\Delta_{info}(t) = 0.994 - (\frac{2}{11} \cdot 1 + \frac{9}{11} \cdot 0.991) = 0.001$

Test nad atributy $\sim$ rozřazení objektů

? binary nebo multiway split, kolik intervalů (spojitý atribut) – jaká gain?

- více výsledků testu $\Rightarrow$ menší podmnožiny rozkladu množiny objektů asociovaných s uzlem stromu $\rightsquigarrow$ častěji nižší (nulové) impurity uzlů (potomků) s asociovanými objekty v podmnožinách (objekty mají častěji stejnou class label) $\Rightarrow$ vyšší gain, např.
  pro $D_t = \{1 - 11\}$, dělicí atribut = dětí: hodnoty $\{0\}, > 0$ $\Delta_{info}(t) = 0.001$
  hodnoty $\{0\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, $\{5\}$, $\geq 6 \rightarrow D_{t'_{\{0\}}} = \{2, 10\}, D_{t'_{\{1\}}} = \{5, 6\}, D_{t'_{\{2\}}} = \{3, 7, 9\}, D_{t'_{\{3\}}} = \{8, 11\}, D_{t'_{\{4\}}} = \emptyset, D_{t'_{\{5\}}} = \{1\}, D_{t'_{\geq 6}} = \{4\}$:
  $I(t'_{\{0\}}) = -(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}) = 1$, $I(t'_{\{1\}}) = -(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}) = 1$,
  $I(t'_{\{2\}}) = -(\frac{2}{3}\log_2\frac{2}{3} + \frac{1}{3}\log_2\frac{1}{3}) \doteq 0.918$, $I(t'_{\{3\}}) = -(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}) = 1$,
  $I(t'_{\{4\}}) = 0$, $I(t'_{\{5\}}) = -(\frac{1}{1}\log_2\frac{1}{1} + \frac{0}{1}\log_2\frac{0}{1}) = 0$, $I(t'_{\geq 6}) = -(\frac{0}{1}\log_2\frac{0}{1} + \frac{1}{1}\log_2\frac{1}{1}) = 0$
  $\Delta_{info}(t) = 0.994 - (\frac{2}{11}\cdot 1 + \frac{2}{11}\cdot 1 + \frac{3}{11}\cdot 0.918 + \frac{2}{11}\cdot 1 + \frac{0}{11}\cdot 0 + \frac{1}{11}\cdot 0 + \frac{1}{11}\cdot 0) = 0.198$

Test nad atributy $\sim$ rozřazení objektů

? binary nebo multiway split, kolik intervalů (spojitý atribut) – jaká gain?

- více výsledků testu $\Rightarrow$ menší podmnožiny rozkladu množiny objektů asociovaných s uzlem stromu $\rightsquigarrow$ častěji nižší (nulové) impurity uzlů (potomků) s asociovanými objekty v podmnožinách (objekty mají častěji stejnou class label) $\Rightarrow$ vyšší gain, např.
  pro $D_t = \{1 - 11\}$, dělicí atribut = dětí: hodnoty $\{0\}$, > 0 $\Delta_{info}(t) = 0.001$
  hodnoty $\{0\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, $\{5\}$, ≥ 6 $\Delta_{info}(t) = 0.198$
- ovšem také $\rightsquigarrow$ klasifikace na základě menšího (statisticky nevýznamného) počtu objektů asociovaných s listovými uzly (data fragmentation problem) = nižší schopnost generalizace – extrém jeden objekt (atribut „typu ID“)

$\rightarrow$ pouze binary split, např. CART

$\rightarrow$ zahrnutí počtu výsledků testu do dělicího kritéria, např. v C4.5 **gain ratio**:

$$\frac{\Delta_{info}}{split\ info}, \quad split\ info(t) = -\sum_{t'} \frac{N(t')}{N(t)} \log_2 \frac{N(t')}{N(t)}$$

Test nad atributy $\sim$ rozřazení objektů

např. pro $D_t = \{1 - 11\}$, dělicí atribut $=$ dětí:
hodnoty $\{0\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, $\{5\}$, ≥ 6

Test nad atributy $\sim$ rozřazení objektů

např. pro $D_t = \{1 - 11\}$, dělicí atribut = dětí:
hodnoty $\{0\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, $\{5\}$, $\geq 6 \rightarrow D_{t'_{\{0\}}} = \{2, 10\}, D_{t'_{\{1\}}} = \{5, 6\}, D_{t'_{\{2\}}} = \{3, 7, 9\}, D_{t'_{\{3\}}} = \{8, 11\}, D_{t'_{\{4\}}} = \emptyset, D_{t'_{\{5\}}} = \{1\}, D_{t'_{\geq 6}} = \{4\}$:

$split\ info(t) = -(\frac{2}{11}\log_2\frac{2}{11} + \frac{2}{11}\log_2\frac{2}{11} + \frac{3}{11}\log_2\frac{3}{11} + \frac{2}{11}\log_2\frac{2}{11} + \frac{0}{11}\log_2\frac{0}{11} + \frac{1}{11}\log_2\frac{1}{11} + \frac{1}{11}\log_2\frac{1}{11}) \doteq 2.482$

$\Delta_{info}(t)/split\ info = 0.198/2.482 \doteq 0.08$

Test nad atributy $\sim$ rozřazení objektů

např. pro $D_t = \{1 - 11\}$, dělicí atribut = dětí:
hodnoty $\{0\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, $\{5\}$, ≥ 6 $\Delta_{info}(t)/split\ info \doteq 0.08$

hodnoty $\{0\}$, $\{1,2\}$, $> 2 \rightarrow D_{t'_{\{0\}}} = \{2,10\}$, $D_{t'_{\{1,2\}}} = \{3,5-7,9\}$, $D_{t'_{>2}} = \{1,4,8,11\}$
$split\ info(t) = -(\frac{2}{11}\log_2\frac{2}{11} + \frac{5}{11}\log_2\frac{5}{11} + \frac{4}{11}\log_2\frac{4}{11}) \doteq 1.495$
$\Delta_{info}(t)/split\ info = 0.007/1.495 \doteq 0.005$

Test nad atributy $\sim$ rozřazení objektů

např. pro $D_t = \{1 - 11\}$, dělicí atribut = dětí:
hodnoty $\{0\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, $\{5\}$, ≥ 6 $\Delta_{info}(t)/split\ info \doteq 0.08$

hodnoty $\{0\}$, $\{1, 2\}$, > 2 $\Delta_{info}(t)/split\ info \doteq 0.005$

hodnoty $\{0\}$, $> 0 \rightarrow D_{t'_{\{0\}}} = \{2, 10\}$, $D_{t'_{>0}} = \{1, 3 - 9, 11\}$
$split\ info(t) = -(\frac{2}{11} \log_2 \frac{2}{11} + \frac{9}{11} \log_2 \frac{9}{11}) \doteq 0.684$
$\Delta_{info}(t)/split\ info = 0.001/0.684 \doteq 0.001$

Vlastnosti rozhodovacích stromů

- vytvoření optimálního stromu = **NP-těžký problém** $\rightarrow$ heuristický přístup, např. greedy, top-down rekurzivní rozklad – výpočetně nenáročné algoritmy, klasifikace objektu rychlá (daná maximální hloubkou stromu)
- relativně **snadná interpretace**, **přesnost (accuracy) srovnatelná** s jinými klasifikátory (pro běžná data, pro hůře klasifikovatelná nižší)
- **robustní vůči šumu** v datech (zejména při řešení přeučení), redundantní (korelované) atributy nemají vliv na přesnost, příliš mnoho irelevantních atributů strom zbytečně zvětšuje $\rightarrow$ feature selection
- výběr míry impurity má malý vliv na přesnost (chovají se podobně), větší vliv má strategie prořezání stromu, viz dále
- neparametrická klasifikační metoda – nepředpokládá určitá pravděpodobností rozložení class labels a atributů

Přeučení (overfitting) modelu

- **chyba učení (training, resubstitution error)** = počet chyb klasifikace trénovacích dat
- **chyba generalizace (generalization error)** = množství chyb klasifikace (neznámých) testovacích dat

* klasifikační problém $\sim$ dobrý klasifikační model: co nejlepší modelování trénovacích dat, ale zároveň i správná klasifikace neznámých dat (generalizace) $\Rightarrow$ nízké obě chyby

= vyšší chyba generalizace v důsledku příliš přesného modelování trénovacích dat a špatné klasifikace neznámých dat

- zpočátku učení s malým modelem chyby na trénovacích i testovacích datech vysoké = **„podučení“ (underfitting)** modelu, s narůstajícím modelem (učení) klesají, avšak od určité velikosti modelu chyba na testovacích datech začne růst (na trénovacích datech dál klesá) = přeučení

Přeučení (overfitting) modelu

- příčina: šum a chyby v trénovacích datech – např. chybná class label, výjimečné testovací objekty s jinou class label než stejné trénovací objekty, např.
- příčina: nedostatek reprezentativních objektů v trénovacích datech (pro určité kombinace hodnot atributů), např.
- důvod (?): opakovaně výběr nejlepší (maximalizující nějaké kritérium) možnosti z několika a její přidání do modelu, pokud je dostatečná (tzv. multiple comparison procedure) – čím více možností, tím větší šance a zvětšování modelu, zvláště při malém počtu trénovacích objektů (velký rozptyl hodnot kritéria výběru), např. splitting atribute a gain „hlouběji“ v rozhodovacích stromech $\rightarrow$ zahrnout počet možností do kritéria výběru, výběr možnosti jen při dostatečném počtu trénovacích objektů

Odhad chyby generalizace

- cíl = **model selection**: velikost/složitost modelu odpovídající nejnižší chybě generalizace – jaká?
- při učení dostupná jen trénovací data, ne neznámá (testovací) $\Rightarrow$ odhad chyby

Přeučení (overfitting) modelu

Odhad chyby generalizace

= chyba učení – předpoklad: trénovací data dobře reprezentují všechna, ovšem minimální $\Rightarrow$ složitý model $\Rightarrow$ přeučení

$\rightarrow$ preference jednoduššího modelu – dle principu **Ockhamovy břitvy** (úspornosti) nebo **KISS**: „z více modelů se stejnou chybou má být preferován ten nejjednodušší“ $\rightarrow$ zahrnutí složitosti modelu do odhadu chyby:

- \+ penalizace za nárůst modelu, např. za každý listový uzel stromu, ve vztahu k chybě klasifikace, neboli jak moc se chyba nárůstem modelu musí snížit
- \+ velikost modelu, např. stromu – minimum description length (MDL) princip

$\rightarrow$ statistická korekce – horní mez na základě předpokládané distribuce chyb klasifikace a počtu trénovacích objektů klasifikovaných např. v listových uzlech stromu

= chyba na **validation set** = část trénovacích dat nepoužitých k učení, obvykle třetina, viz dále vyhodnocení výkonnosti, použití u metod s parametrem pro složitost modelu, např. úroveň prořezání stromu – výběr toho s nejnižší touto chybou

Přeučení (overfitting) modelu

Řešení v rozhodovacích stromech

- **prepruning** = zastavení větvení stromu před dosažením plného (přeučeného) bezchybně modelujícího celá vstupní data → přísnější podmínka pro listový uzel (místo vnitřního) – jaká?, např.
  - minimální počet objektů asocionavých s listovým uzlem – jaký?
  - minimální hodnota dělícího kritéria (gain) – jaká?, další větvení může vést k lepšímu modelu (menší chyba generalizace)
  - minimální zlepšení odhadu chyby generalizace – jaké?, aj.
- **post-pruning** = „prořezání“ stromu po vytvoření plného, dokud zlepšení od listů ke kořenu nahrazování podstromů:
  - listovým uzlem s majoritní class label objektů asociovaných s uzly podstromu (subtree replacement)
  - nejčastěji používanou větví (podstromem) podstromu = s nejvíce objekty asociovanými s uzly (subtree raising)
- post-pruning obvykle lepší (z plného stromu), za cenu vytvoření plného stromu

Vyhodnocení výkonnosti (performace)

- vytvořeného modelu na testovacích/validation datech – nezkreslený odhad chyby generalizace (oproti odhadu z učení), také pro porovnání výkonnosti různých metod

* základní ukazatele: accuracy (přesnost) a error rate (chybovost)

! potřeba znát class labels testovacích objektů

Holdout

- původní data (s class labels) náhodně disjunktně rozdělena na trénovací a testovací množiny – typicky 50-50 nebo 2/3 trénovací, a
- učení na trénovací, vyhodnocení modelu na testovací
- nevýhody: méně dat pro učení $\Rightarrow$ horší model, závislost modelu na velikosti množin – kvalita modelu vs. důvěryhodnost odhadu výkonnosti (malý rozptyl)

Random subsampling

= opakované holdout pro zlepšení odhadu výkonnosti

- přesnost modelu = průměr přesností z opakování
- nevýhody: stále méně dat pro učení než maximum, různé objekty různě často pro učení a testování

Vyhodnocení výkonnosti (performace)

Cross-validation

- **k-fold**: původní data (s class labels) náhodně disjunktně rozdělena na k stejně velkých množin, typicky 10
- vyhodnocení modelu na jedné (testovací), učení na sjednocení zbývajících $k-1$ (trénovací)
- k opakování pro každou množinu jako testovací
- chybovost modelu $=$ součet chybovostí z opakování
- **leave-one-out**: $k =$ celkový počet objektů – maximum dat pro učení
- výhody: každý objekt stejně často $(k-1)$ pro učení a jednou pro testování, testovací množiny disjunktní a obsahující všechny objekty
- nevýhody: výpočetní náročnost, maximum dat pro učení vs. důvěryhodnost odhadu výkonnosti (malý rozptyl)

Vyhodnocení výkonnosti (performace)

Bootstrap

= část původních dat (s class labels) pro učení (trénovací) vybraná náhodně (sampling) s ponecháním, tj. objekt může být vybrán vícekrát

- v průměru konverguje k 63.2 % objektů původních dat (N, pravděpodobnost výběru objektu $1-(1-1/N)^N$, $\lim_{N\to\infty} 1-(1-1/N)^N = 1-e^{-1} \doteq 0.632$)
- zbývající část pro vyhodnocení modelu (testovací)
- opakování, více variant pro přesnost modelu z přesností z opakování, např. pro .632 bootstrap $= \frac{1}{b}\sum_{i=1}^{b} 0.632 \cdot a_i + 0.368 \cdot a$, b počet opakování, a_i přesnost z opakování, a přesnost modelu naučeného z celých původních dat

Class imbalance problem

= významně různý počet objektů vstupních dat s různými class labels (třídami klasifikace)

- v aplikacích běžné, zejména u binární klasifikace (dvě třídy), např. významně méně vadných výrobků, podvodných finančních transakcí, pozitivních případů nemoci, ... různé poměry
- správná klasifikace méně zastoupené třídy důležitější – problém, pro metody všechny třídy stejně významné → zaměření se, specializace, problém šumu

→ **cost-sensitive learning** = zahrnutí ceny za (mis)klasifikaci do kritérií tvorby modelu s cílem nejnižší celkové ceny, např. gain u rozhodovacích stromů

→ **vzorkování (sampling)** = podvzorkování více zastoupené nebo nadvzorkování méně zastoupené třídy (objektů s ní), nebo obojí, u dat pro učení pro stejné zastoupení

- přesnost (accuracy) pro měření výkonnosti (performance) nevhodná – všechny třídy stejně významné: např. při 1 % pozitivních má model klasifikující vše jako negativní přesnost 99 %

Class imbalance problem

Alternativní ukazatele výkonnosti (performace) – pro binární klasifikaci

- méně zastoupená třída (objekty s ní) $\sim$ pozitivní $(+)$, více (majoritní) $\sim$ negativní $(-)$
- confusion matrix:

		predikovaná třída	
		$+$	$-$
skutečná třída	$+$	f_{++} (TP)	f_{+-} (FN)
	$-$	f_{-+} (FP)	f_{--} (TN)

TP = true positive, FN = false negative, FP = false positive, TN = true negative

- **true positive rate / sensitivity** = podíl správně predikovaných pozitivních objektů: $TPR = TP/(TP + FN)$
- **true negative rate / specificity** = podíl správně predikovaných negativních objektů: $TNR = TN/(TN + FP)$
- **false positive rate** = podíl negativních objektů nesprávně predikovaných jako pozitivní: $FPR = FP/(TN + FP)$
- **false negative rate** = podíl pozitivních objektů nesprávně predikovaných jako negativní: $FNR = FN/(TP + FN)$

Class imbalance problem

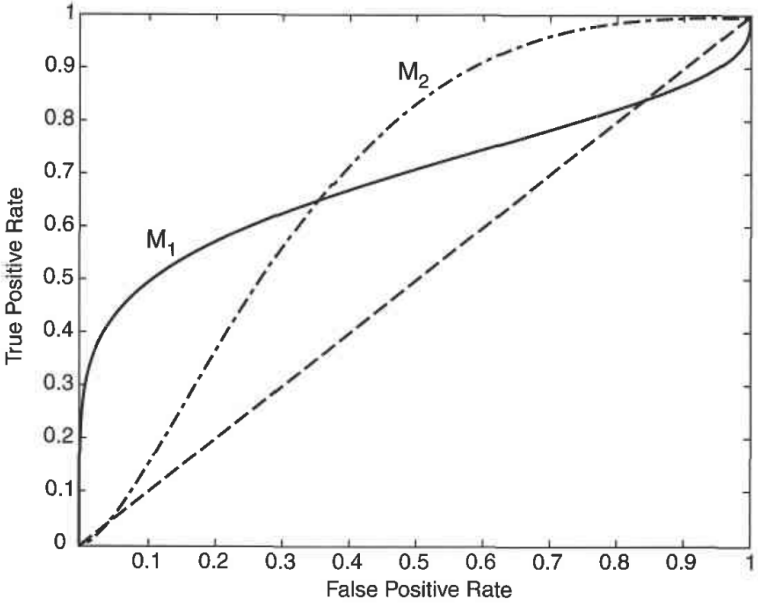
Alternativní ukazatele výkonnosti (performace) – pro binární klasifikaci

- **precision** = podíl správně predikovaných pozitivních objektů z predikovaných jako pozitivní: $p = TP/(TP + FP)$
- **recall** = podíl správně predikovaných pozitivních objektů (ze všech pozitivních): $r = TP/(TP + FN) = TPR$

→ (oba) nejvyšší – pouze jeden snadné, např. vše predikované jako pozitivní

- $\mathbf{F_1} = 2pr/(p + r) = 2TP/(2TP + FP + FN) \sim$ harmonický průměr p a $r = 2/(1/p + 1/r) \rightsquigarrow$ menší z nich $\Rightarrow$ vysoký $F_1 \sim$ vysoké oba
- $\mathbf{F}_\beta = (\beta^2 + 1)pr/(\beta^2 p + r) = (\beta^2 + 1)TP/((\beta^2 + 1)TP + \beta^2 FP + FN)$, = precision pro $\beta = 0$, recall pro $\beta \to \infty$
- **weighted accuracy (vážená přesnost)** = $(w_1 TP + w_4 TN)/(w_1 TP + w_2 FP + w_3 FN + w_4 TN)$
- **ROC** (receiver operating characteristic) **křivka**: vztah mezi TPR ($[0, 1]$ osa y) a FPR ($[0, 1]$ osa x) pro různé modely (s různými parametry), ..., např., plocha pod ní, ...

Class imbalance problem



Pravidlová (rule-based) klasifikace

= pomocí množiny **klasifikačních pravidel „jestliže-pak“** = model:

$R = r_1 \vee r_2 \vee \ldots r_n$ **rule set**

$\mathbf{r_i} : ((\mathbf{x_{i1}\ op\ v_{i1}}) \wedge (\mathbf{x_{i2}\ op\ v_{i2}}) \wedge \ldots (\mathbf{x_{ik}\ op\ v_{ik}})) \longrightarrow \mathbf{y_i}$ (antecedent/prekondice $\longrightarrow$ konsekvent)

x_{ij} atribut, v_{ij} hodnota atributu x_{ij}, $op \in \{=, \neq, <, >, \leq, \geq\}$ relační operátor, y_i predikovaná class label, např.

$r_1 : ((zaměstnání = ne) \wedge (nemovitost = ne)) \longrightarrow ANO$

$r_2 : ((zaměstnání = ne) \wedge (nemovitost = ano) \wedge (partner = ne)) \longrightarrow ANO$

$r_3 : ((zaměstnání = ano) \wedge (partner = ano)) \longrightarrow NE$

$r_4 : ((dětí > 1)) \longrightarrow ANO$

- pravidlo r **pokrývá** objekt x: prekondice r je pravdivá pro hodnoty atributů objektu x, např. r_1 pokrývá objekt 1 z příkladu ke klasifikaci, nepokrývá objekt 3
- základní ukazatele kvality pravidla r, pro množinu D objektů (dataset):
  - **coverage (pokrytí)** = podíl objektů z D pokrytých r
  - **accuracy (přesnost)** = podíl objektů majících class label rovnu konsekventu r z objektů pokrývaných r

např. r_2 má pokrytí 3/11 a přesnost 2/3 na příkladu ke klasifikaci

Klasifikace objektu

= predikovaná class label pravidla pokrývajícího objekt, např. objekt 12 z příkladu ke klasifikaci pokrývají pravidla r_1 a r_4 $\Rightarrow$ class label ANO – pravidel může být víc nebo žádné

- vyčerpávající (exhaustive) pravidla v R: v R existuje pravidlo pro každou kombinaci hodnot atributů $\Rightarrow$ každý objekt je pokrývaný alespoň jedním pravidlem z R, např.
  $r'_1 : ((zaměstnání = ne) \wedge (nemovitost = ne)) \longrightarrow ANO$
  $r'_2 : ((zaměstnání = ne) \wedge (nemovitost = ano)) \longrightarrow NE$
  $r'_3 : ((zaměstnání = ano)) \longrightarrow NE$
  - jinak navíc v R **výchozí pravidlo** $() \longrightarrow \mathbf{y}$ – pokrývající objekt když ne žádné jiné, y typicky majoritní class label trénovacích objektů nepokrývaných jinými pravidly
- vzájemně výlučná pravidla v R: žádná dvě nepokrývají stejný objekt $\Rightarrow$ každý objekt je pokrývaný nejvýše jedním pravidlem z R, např. výše, jinak při více s různými class labels:
- neuspořádaná pravidla: klasifikace objektu (obvykle) většinovou class label pokrývajících pravidel, příp. vážené pomocí accuracy

Klasifikace objektu

- uspořádaná pravidla (R **decision list**): sestupně dle priority, např. accuracy, coverage, počet atributů, pořadí vytvoření, klasifikace objektu prvním pokrývajícím
  - dle pravidel: dle ukazatele kvality pravidla – méně prioritní pravidla hůře interpretovatelná (předpokládaná negace antecedentu všech předchozích pravidel), např. interpretace r_4 (v $r_1 - r_4$): jestliže (má zaměstnání a nemá partnera) nebo (nemá zaměstnání, má partnera a má nemovitost) a má více než jedno dítě, pak přidělit dávku
  - dle class labels: se stejnou class label neuspořádaně za sebou – pořadí class labels?, většina používaných, např. C4.5rules, RIPPER

Vytvoření (indukce) pravidel

- přímé metody: přímo z (trénovacích) dat, rozkládání množiny objektů (prostoru atributů) podle hodnot atributů na podmnožiny klasifikovatelné (pokrytelné) jedním pravidlem, např. sekvenční pokrývání, RIPPER
- nepřímé metody: z jiných (složitějších) klasifikačních modelů, např. rozhodovacího stromu, např. C4.5rules, neuronové sítě

Sekvenční pokrývání

- greedy strategie, pravidla postupně pro uspořádané class labels vyjma poslední – pořadí např. dle podílu objektů s class label (vzestupně), ceny za misklasifikaci (sestupně)

Input : množina D trénovacích objektů, uspořádané class labels $Y = \{y_1, y_2, \ldots, y_k\}$
Output: rule set R
$R = \emptyset$;
foreach $y \in Y \setminus \{y_k\}$ **do**
  while neplatí ukončovací podmínka **do**
    $r \leftarrow$ Learn-One-Rule(D, y);
    $D \leftarrow D \setminus$ objekty pokrývané r;
    $R \leftarrow R \vee r$;
$R \leftarrow R \vee () \longrightarrow y_k$;

Sekvenční pokrývání

LEARN-ONE-RULE(D, y)

→ nejlepší pravidlo r : antecedent $\longrightarrow y$ pokrývající nejvíce objektů z D s class label y (= **pozitivní objekty**) a nejméně ostatních (= **negativní**)

- greedy postup tvorby (antecedentu) pravidla, dokud se např. zlepšuje jeho kvalita nebo pokrývá jen pozitivní objekty, strategie:
  - od obecného ke specifickému: počáteční pravidlo $() \longrightarrow y$, přidávání porovnání atributů do antecedentu, např. $() \longrightarrow NE$, $((zaměstnání = ano)) \longrightarrow NE$, $((zaměstnání = ano) \wedge (partner = ano)) \longrightarrow NE$
  - od specifického k obecnému: náhodně vybraný pozitivní objekt (atributy s hodnotami pro objekt) jako počáteční antecedent, odebírání porovnání atributů z něj pro větší pokrytí, např. pro objekt 3 z příkladu na klasifikaci $((zaměstnání = ne) \wedge (nemovitost = ano) \wedge (partner = ne)) \longrightarrow ANO$, $((zaměstnání = ne) \wedge (partner = ne)) \longrightarrow ANO$, $((zaměstnání = ne)) \longrightarrow ANO$
  - beam search: více nezávisle vytvářených nejlepších pravidel současně
- kvalita pravidla: nejen přesnost (accuracy), ale i pokrytí (coverage), např. pro 60/100 pozitivních/negativních objektů lepší r_x pokrývající 50/5 než r_y pokrývající 2/0
- poté možné odstranění (pruning) pravidla pro snížení chyby generalizace – metody odhadu viz dříve (část přeučení (overfitting) modelu)

Kvalita pravidla

- statistický test likelihood ratio – vyšší při více správných predikcích než náhodných:

$$2 \sum_c n_c \log_2(n_c/e_c)$$

  c class label, n_c počet pokrývaných objektů s class label c, e_c očekávaný n_c při náhodných predikcích pravidlem (tj. počet pokrývaných · počet s class label c / počet všech objektů), např. pro r_x je $2 \cdot (50 \cdot \log_2(50/(55 \cdot 60/160)) \ldots) \approx 99.9$, pro r_y je 5.7
- míra Laplace $\rightsquigarrow$ accuracy při vyšší coverage:

$$\frac{n_+ + 1}{n + k}$$

  n_+ počet pozitivních pokrývaných objektů (= support pravidla), n počet pokrývaných objektů, k počet class labels, např. pro r_x je $51/57 \approx 89.5\,\%$, pro r_y je $3/4 = 75\,\%$
- **FOIL's information gain** – vyšší pro vyšší n_+^r a accuracy, pro nové pravidlo r a předchozí r':

$$n_+^r \cdot \left(\log_2 \frac{n_+^r}{n_+^r + n_-^r} - \log_2 \frac{n_+^{r'}}{n_+^{r'} + n_-^{r'}} \right)$$

  např. při r' pokrývajícím 60/100 pro r_x je $50 \cdot (\log_2(50/55) - \ldots) \approx 63.9$, pro r_y je 2.8

Algoritmus RIPPER

- široce používaný
- pravidla vytvářena sekvenčním pokrývaním – pořadí class labels vzestupně dle podílu objektů s class label, ukončovací podmínka (s nepřidáním pravidla) maximální navýšení velikosti rule set (description length) nebo chybovost pravidla na validation set přes 50 %
- tvorba pravidla od obecného ke specifickému – dokud pokrývá jen pozitivní objekty, ukazatel kvality pravidla FOIL's information gain
- pruning pravidla postupným odebíráním posledně přidaného porovnání atributu – při vyšším $(n_+ - n_-)/(n_+ + n_-)$ ($\sim$ accuracy) na validation set, poté může pokrývat i negativní objekty
- dále nahrazování pravidel jinými lepšími

Pravidla z rozhodovacího stromu

- pravidlo $\sim$ cesta z kořenového do listového uzlu: konjunkce testů nad atributy u vnitřních uzlů na cestě = antecedent, class label u listového uzlu = konsekvent, např. pro strom z příkladu dříve:
  $((příjem \leq 9,000)) \longrightarrow ANO$
  $((příjem > 9,000) \wedge (věk \leq 33)) \longrightarrow ANO$
  $((příjem > 9,000) \wedge (věk > 33)) \longrightarrow NE$
- pravidla pro všechny cesty vyčerpávající a vzájemně výlučná $\Rightarrow$ každý objekt je pokrývaný právě jedním pravidlem
- některá pravidla lze zjednodušit (nemusí pak být vzájemně výlučná)

Algoritmus C4.5rules

- pravidla pro všechny cesty z kořenového do listového uzlu
- pruning pravidel odebíráním porovnání atributu – pro co nejnižší odhad chyby generalizace (dle preference jednoduššího modelu, viz dříve)
- pravidla uspořádaná dle class labels – pořadí class labels vzestupně dle velikosti pravidel se stejnou class label plus počet misklasifikovaných objektů (description length)

Vlastnosti pravidlové klasifikace

- vytvoření, interpretace, kvalita (přesnost), robustnost pravidel **podobné jako u rozhodovacích stromů**
- rozklad množiny / separace objektů (prostoru atributů) na podmnožiny s hranicemi (decision boundaries) nezávislými na jiných atributech (rektilineárními) – podobně jako rozhodovací stromy, při pokrývání objektu více pravidly i jiné hranice ($\sim$ oblique stromy)
- typicky používaná pro **deskriptivní modelování**
- s uspořádanými pravidly dle class labels vhodná pro data s významně různými počty objektů s různými class labels (**class imbalanced data**)

Klasifikace nejbližšími sousedy (nearest-neighbor)

* klasifikace = (1) vytvoření (učení, indukce) modelu ze vstupních dat, (2) aplikování modelu na nová data (dedukce)
- **eager klasifikace/learner** = nejdříve (1) pro celý model, pak (2), např. rozhodovací stromy, pravidlová
- **lazy klasifikace/learner** = odložení (1) pro část modelu až do její potřeby v (2) nebo i vynechání (1), např. **Rote klasifikace**: model = vstupní data a klasifikace dle vstupního objektu = nový objekt – nemusí existovat

→ klasifikace dle vstupních objektů podobných novému = **nejbližší sousedi** – „když to kváká jako kachna, . . . “

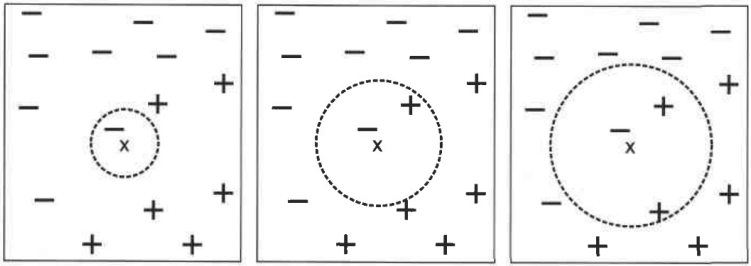
- **k-nejbližší sousedi** = k vstupních objektů nejbližších novému, dle míry (ne)podobnosti (vzdálenosti) = **model dal** – objekt $\sim$ bod v n-rozměrném prostoru, kde n je počet atributů, např.

= majoritní class label nejbližších sousedů, nebo kterákoliv

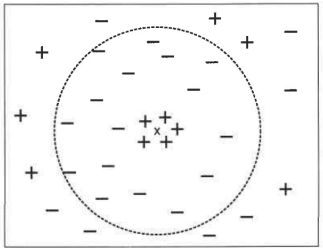
! volba k: malé $\rightsquigarrow$ přeučení kvůli šumu ve vstupních datech, velké $\rightsquigarrow$ chybná klasifikace dle většiny vzdálenějších objektů, např.

→ váhy class label sousedů $\mathbf{x_i}$ dle jejich vzdálenosti: $w_i = 1/d(\mathbf{x}, \mathbf{x_i})^2$

Klasifikace nejbližšími sousedy (nearest-neighbor)



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor



Vlastnosti klasifikace nejbližšími sousedy

- jedna z **instance-based klasifikací/learning** = klasifikace **dle** určitých **vstupních objektů bez** udržování **modelu** (abstrakce) dat – potřeba míry (ne)podobnosti mezi objekty a určení (klasifikační funkce) class label nového objektu dle (ne)podobností
- lazy klasifikátory nevytváří model dat, ale klasifikace objektu může být výpočetně náročná – výpočet (ne)podobností s objekty vstupních dat, eager klasifikátory opačně
- klasifikace na základě „lokální informace“ – vliv šumu (s malým k)
- libovolné proměnlivé hranice (decision boundaries) separace objektů (prostoru atributů) – závisí na konkrétním umístění objektů v prostoru atributů
- zavisí na volbě míry (ne)podobnosti a předzpracování dat, např. normalizaci

Bayesovská klasifikace

- vztah mezi atributy a class label může být neurčitý (nedeterministický), class label nového objektu dle atributů nemusí být možné určit jednoznačně – kvůli šumu v datech, nejednoznačné interpretaci atributů nebo dalším neznámým faktorům mimo atributy, např. predikce zdravotních problémů

$\rightarrow$ pravděpodobnostní modelování vztahu

Bayesova věta

$=$ statistický princip kombinace (v klasifikaci) znalostí class labels a hodnot atributů

- X, Y náhodné proměnné, sdružená (joint) pravděpodonost $P(X = x, Y = y)$, $P(X) = \sum_i P(X, Y = y_i)$ (zákon o úplné pravděpodobnosti), podmíněná pravděpodobnost $P(Y = y|X = x)$: $P(X, Y) = P(Y|X)P(X) = P(X|Y)P(Y)$ (Bayesovo pravidlo)

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

Bayesova věta: příklad

Nakažených 10 %, z nich pozitivně testovaných 90 %, ale z nenakažených pozitivně testovaných 20 %. Nakažený, když pozitivně testovaný?

Bayesova věta: příklad

Nakažených 10 %, z nich pozitivně testovaných 90 %, ale z nenakažených pozitivně testovaných 20 %. Nakažený, když pozitivně testovaný?

$X \in 0, 1 \ldots$ test negativní (0) / pozitivní (1),
$Y \in 0, 1 \ldots$ nákaza ne (0) / ano (1),
$P(Y = 1) = 0.1$, $P(Y = 0) = 1 - P(Y = 1) = 0.9$, $P(X = 1|Y = 1) = 0.9$,
$P(X = 1|Y = 0) = 0.2$

$$
\begin{aligned}
P(Y = 1|X = 1) &= P(X = 1|Y = 1)P(Y = 1)/P(X = 1) \\
&= P(X = 1|Y = 1)P(Y = 1)/(P(X = 1, Y = 1) + P(X = 1, Y = 0)) \\
&= P(X = 1|Y = 1)P(Y = 1)/(P(X = 1|Y = 1)P(Y = 1) + P(X = 1|Y = 0)P(Y = 0)) \\
&= 0.9 \cdot 0.1/(0.9 \cdot 0.1 + 0.2 \cdot 0.9) \\
&\doteq 0.333
\end{aligned}
$$

Bayesovská klasifikace

- $\mathbf{X} = (X_1, \ldots, X_m)$ vektor m atributů, Y class labels, náhodné proměnné
- $P(Y)$ **prior pravděpodobnost** class labels, $P(Y|\mathbf{X})$ **posterior pravděpodobnost** class labels v závislosti na atributech = **model dat** (pro class labels a hodnoty atributů pro objekty)

= class label nového objektu $\mathbf{x} = (x_1, \ldots, x_m)$: y s maximální $P(Y = y|\mathbf{X} = \mathbf{x})$ – její (přesnější) odhad vyžaduje mnoho (ideálně všechny) kombinací class labels y a hodnot x_i atributů $\mathbf{X}$

→ vyjádření $P(Y|\mathbf{X})$ pomocí $P(Y)$, pravděpodobnosti $P(\mathbf{X}|Y)$ v závislosti na class labels a $P(\mathbf{X})$ – s využitím Bayesovy věty

- $P(\mathbf{X})$ pro různé y konstantní ⇒ netřeba, odhad $P(Y)$ z četností y pro objekty vstupních dat

? odhad $P(\mathbf{X}|Y)$ – stále vyžaduje mnoho (ideálně všechny) kombinací y a x_i, ale možné vyjádřit snadněji

- např. X_i výskyt určitého slova ano/ne (nebo i počet), Y spam ano/ne, objekty ∼ maily, $P(Y)$, $P(Y|\mathbf{X})$, $P(\mathbf{X}|Y)$

Naïvní bayesovská klasifikace

= atributy X_i jsou, při dané class label y, jako náhodné proměnné **nezávislé**:

$$P(\mathbf{X}|Y = y) = \prod_{i=1}^{m} P(X_i|Y = y)$$

- nezávislost X_1 a X_2 při Y: $P(X_1|X_2, Y) = P(X_1|Y)$, např. X_1 velikost hlavy, X_2 inteligence, Y ?, $P(X_1, X_2|Y) = \ldots = P(X_1|Y)P(X_2|Y)$

→ místo odhadu $P(\mathbf{X}|Y)$ odhady $P(X_i|Y)$ – stačí méně kombinací y a hodnot x_i (jednoho) atributu X_i

= class label y s maximální

$$P(Y = y|\mathbf{X} = \mathbf{x}) = \frac{P(Y = y)\prod_{i=1}^{m} P(X_i = x_i|Y = y)}{\cancel{P(\mathbf{X} = \mathbf{x})}}$$

? odhad $P(X_i|Y)$

Naïvní bayesovská klasifikace

- diskrétní atribut X: odhad $P(X|Y)$ z relativních četností hodnot X vzhledem k y pro objekty vstupních dat
- spojitý atribut X:
  - diskretizace na ordinální (nahrazení hodnot intervaly) – chyba odhadu závisí na metodě a počtu intervalů (malé unikátní vs. velké více class labels)
  - předpokládáná určitá distribuce $P(X|Y)$ a odhad parametrů funkce hustoty pravděpodobnosti f z hodnot X vzhledem k y pro objekty vstupních dat, např. normální (gaussovská) s parametry střední hodnota μ (průměr $\overline{x}$) a rozptyl σ^2 (s^2):

$$P(x \leq X \leq x + \epsilon | Y = y) \approx \epsilon f(X, \mu_{xy}, \sigma_{xy}) = \epsilon \frac{1}{\sigma_{xy}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(x-\mu_{xy})^2}{\sigma_{xy}^2}}$$

Naïvní bayesovská klasifikace: příklad

$P(partner = ano|dávka = ANO) = 2/6$, $P(partner = ne|dávka = ANO) = 4/6$,
$P(partner = ano|dávka = NE) = 3/5$, $P(partner = ne|dávka = NE) = 2/5$
$P(zaměstnání = ano|dávka = ANO) = 1/6$, $P(zaměstnání = ne|dávka = ANO) = 5/6$,
$P(zaměstnání = ano|dávka = NE) = 3/5$, $P(zaměstnání = ne|dávka = NE) = 2/5$
$P(nemovitost = ano|dávka = ANO) = 2/6$, $P(nemovitost = ne|dávka = ANO) = 4/6$,
$P(nemovitost = ano|dávka = NE) = 2/5$, $P(nemovitost = ne|dávka = NE) = 3/5$

$X = věk$, $dávka = ANO$: $\overline{x} = \frac{32+...+62}{6} \doteq 31.9$, $s^2 = \frac{(32-31.9)^2+...+(62-31.9)^2}{5} \doteq 353.1$
$X = věk$, $dávka = NE$: $\overline{x} = \frac{58+...+49}{5} = 46$, $s^2 = \frac{(58-46)^2+...+(49-46)^2}{4} \doteq 76.5$

$X = dětí$, $dávka = ANO$: $\overline{x} = \frac{5+...+0}{6} \doteq 2.2$, $s^2 = \frac{(5-2.2)^2+...+(0-2.2)^2}{5} \doteq 3$
$X = dětí$, $dávka = NE$: $\overline{x} = \frac{0+...+3}{5} = 2.4$, $s^2 = \frac{(0-2.4)^2+...+(3-2.4)^2}{4} = 5.3$

$X = příjem$, $dávka = ANO$: $\overline{x} = \frac{6335+...+6214}{6} \doteq 5597.3$,
$s^2 = \frac{(6335-5597.3)^2+...+(6214-5597.3)^2}{5} \doteq 6799900.7$
$X = příjem$, $dávka = NE$: $\overline{x} = \frac{9055+...+16150}{5} = 14028.2$,
$s^2 = \frac{(9055-14028.2)^2+...+(16150-14028.2)^2}{4} = 8095111.2$

Naïvní bayesovská klasifikace: příklad

class label ($dávka = ANO/NE$) objektu $\mathbf{X} = (věk = 31, partner = ano, dětí = 7, zaměstnání = ne, příjem = 10375, nemovitost = ne)$?
$\rightarrow P(dávka = ANO|\mathbf{X})$? $P(dávka = NE|\mathbf{X})$?

$P(dávka = ANO) = 6/11$, $P(dávka = NE) = 5/11$
$P(\mathbf{X}|dávka = ANO) = P(věk = 31|ANO)P(partner = ano|ANO)P(dětí = 7|ANO)P(zaměstnání = ne|ANO)P(příjem = 10375|ANO)P(nemovitost = ne|ANO) \doteq$

$P(věk = 31|ANO) = \epsilon \frac{1}{\sqrt{353.1}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(31-31.9)^2}{353.1}} \doteq 0.0212\epsilon$

$P(dětí = 7|ANO) = \epsilon \frac{1}{\sqrt{3}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(7-2.2)^2}{3}} \doteq 0.00495\epsilon$

$P(příjem = 10375|ANO) = \epsilon \frac{1}{\sqrt{6799900.7}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(10375-5597.3)^2}{6799900.7}} \doteq 0.0000286\epsilon$

Naïvní bayesovská klasifikace: příklad

class label ($dávka = ANO/NE$) objektu $\mathbf{X} = (věk = 31, partner = ano, dětí = 7, zaměstnání = ne, příjem = 10375, nemovitost = ne)$?
$\rightarrow P(dávka = ANO|\mathbf{X})$? $P(dávka = NE|\mathbf{X})$?

$P(dávka = ANO) = 6/11,\ P(dávka = NE) = 5/11$
$P(\mathbf{X}|dávka = ANO) = P(věk = 31|ANO)P(partner = ano|ANO)P(dětí = 7|ANO)P(zaměstnání = ne|ANO)P(příjem = 10375|ANO)P(nemovitost = ne|ANO) \doteq 0.0212 \cdot 2/6 \cdot 0.00495 \cdot 5/6 \cdot 0.0000286 \cdot 4/6 \doteq 10^{-9}\epsilon^3$

$P(věk = 31|ANO) = \epsilon \frac{1}{\sqrt{353.1}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(31-31.9)^2}{353.1}} \doteq 0.0212\epsilon$

$P(dětí = 7|ANO) = \epsilon \frac{1}{\sqrt{3}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(7-2.2)^2}{3}} \doteq 0.00495\epsilon$

$P(příjem = 10375|ANO) = \epsilon \frac{1}{\sqrt{6799900.7}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(10375-5597.3)^2}{6799900.7}} \doteq 0.0000286\epsilon$

Naïvní bayesovská klasifikace: příklad

class label ($dávka = ANO/NE$) objektu $\mathbf{X} = (věk = 31, partner = ano, dětí = 7, zaměstnání = ne, příjem = 10375, nemovitost = ne)$?
$\rightarrow P(dávka = ANO|\mathbf{X})$? $P(dávka = NE|\mathbf{X})$?

$P(dávka = ANO) = 6/11,\ P(dávka = NE) = 5/11$
$P(\mathbf{X}|dávka = ANO) = P(věk = 31|ANO)P(partner = ano|ANO)P(dětí = 7|ANO)P(zaměstnání = ne|ANO)P(příjem = 10375|ANO)P(nemovitost = ne|ANO) \doteq 0.0212 \cdot 2/6 \cdot 0.00495 \cdot 5/6 \cdot 0.0000286 \cdot 4/6 \doteq 10^{-9}\epsilon^3$
$P(\mathbf{X}|dávka = NE) = P(věk = 31|NE)P(partner = ano|NE)P(dětí = 7|NE)P(zaměstnání = ne|NE)P(příjem = 10375|NE)P(nemovitost = ne|NE) \doteq$
$P(věk = 31|NE) = \epsilon\frac{1}{\sqrt{76.5}\sqrt{2\pi}}e^{-\frac{1}{2}\frac{(31-46)^2}{76.5}} \doteq 0.0105\epsilon$
$P(dětí = 7|NE) = \epsilon\frac{1}{\sqrt{5.3}\sqrt{2\pi}}e^{-\frac{1}{2}\frac{(7-2.4)^2}{5.3}} \doteq 0.0235\epsilon$
$P(příjem = 10375|NE) = \epsilon\frac{1}{\sqrt{8095111.2}\sqrt{2\pi}}e^{-\frac{1}{2}\frac{(10375-14028.2)^2}{8095111.2}} \doteq 0.0000615\epsilon$

Naïvní bayesovská klasifikace: příklad

class label ($dávka = ANO/NE$) objektu $\mathbf{X} = (věk = 31, partner = ano, dětí = 7, zaměstnání = ne, příjem = 10375, nemovitost = ne)$?
$\rightarrow P(dávka = ANO|\mathbf{X})$? $P(dávka = NE|\mathbf{X})$?

$P(dávka = ANO) = 6/11,\ P(dávka = NE) = 5/11$
$P(\mathbf{X}|dávka = ANO) = P(věk = 31|ANO)P(partner = ano|ANO)P(dětí = 7|ANO)P(zaměstnání = ne|ANO)P(příjem = 10375|ANO)P(nemovitost = ne|ANO) \doteq 0.0212 \cdot 2/6 \cdot 0.00495 \cdot 5/6 \cdot 0.0000286 \cdot 4/6 \doteq 10^{-9}\epsilon^3$
$P(\mathbf{X}|dávka = NE) = P(věk = 31|NE)P(partner = ano|NE)P(dětí = 7|NE)P(zaměstnání = ne|NE)P(příjem = 10375|NE)P(nemovitost = ne|NE) \doteq 0.0105 \cdot 3/5 \cdot 0.0235 \cdot 2/5 \cdot 0.0000615 \cdot 3/5 \doteq 2 \cdot 10^{-9}\epsilon^3$

$$P(věk = 31|NE) = \epsilon \frac{1}{\sqrt{76.5}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(31-46)^2}{76.5}} \doteq 0.0105\epsilon$$

$$P(dětí = 7|NE) = \epsilon \frac{1}{\sqrt{5.3}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(7-2.4)^2}{5.3}} \doteq 0.0235\epsilon$$

$$P(příjem = 10375|NE) = \epsilon \frac{1}{\sqrt{8095111.2}\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(10375-14028.2)^2}{8095111.2}} \doteq 0.0000615\epsilon$$

Naïvní bayesovská klasifikace: příklad

class label ($dávka = ANO/NE$) objektu $\mathbf{X} = (věk = 31, partner = ano, dětí = 7, zaměstnání = ne, příjem = 10375, nemovitost = ne)$?
$\rightarrow P(dávka = ANO|\mathbf{X})$? $P(dávka = NE|\mathbf{X})$?

$P(dávka = ANO) = 6/11$, $P(dávka = NE) = 5/11$
$P(\mathbf{X}|dávka = ANO) = P(věk = 31|ANO)P(partner = ano|ANO)P(dětí = 7|ANO)P(zaměstnání = ne|ANO)P(příjem = 10375|ANO)P(nemovitost = ne|ANO) \doteq 0.0212 \cdot 2/6 \cdot 0.00495 \cdot 5/6 \cdot 0.0000286 \cdot 4/6 \doteq 10^{-9}\epsilon^3$
$P(\mathbf{X}|dávka = NE) = P(věk = 31|NE)P(partner = ano|NE)P(dětí = 7|NE)P(zaměstnání = ne|NE)P(příjem = 10375|NE)P(nemovitost = ne|NE) \doteq 0.0105 \cdot 3/5 \cdot 0.0235 \cdot 2/5 \cdot 0.0000615 \cdot 3/5 \doteq 2 \cdot 10^{-9}\epsilon^3$

$P(dávka = ANO|\mathbf{X}) \doteq 6/11 \cdot 10^{-9}\epsilon^3/P(\mathbf{X}) < P(dávka = NE|\mathbf{X}) \doteq 5/11 \cdot 2 \cdot 10^{-9}\epsilon^3/P(\mathbf{X}) \Rightarrow$ class label NE

Naïvní bayesovská klasifikace

! PROBLÉM: $P(X = x|Y = y) = 0$ pro některý atribut X a jeho hodnotu x a class label $y \Rightarrow P(Y = y|\mathbf{X} = \mathbf{x}) = 0$

- při $P(Y = y|\mathbf{X}) = 0$ pro všechny y nemožnost klasifikace některých nových objektů!

→ **Laplace-odhad** $P(X = x|Y = y)$ (místo relativní četnosti x vzhledem k y pro objekty vstupních dat):

$$P(X = x|Y = y) = \frac{n_{x,y} + 1}{n_y + v}$$

$n_{x,y}$ počet vstupních objektů s hodnotou x atributu X a class label y, n_y počet vstupních objektů s class label y, v počet všech možných hodnot atributu X

→ m-**odhad** $P(X = x|Y = y)$:

$$P(X = x|Y = y) = \frac{n_{x,y} + mp}{n_y + m}$$

p prior pravděpodobnost hodnoty x atributu X pro objekty s class label y (při $n_y = 0$), m tzv. ekvivalentní velikost vzorku pro p („důvěra“ v p při malém n_y) $\approx$ kompromis mezi p a $n_{x,y}/n_y$

- robustnější odhad $P(X|Y)$ při malém počtu vstupních objektů

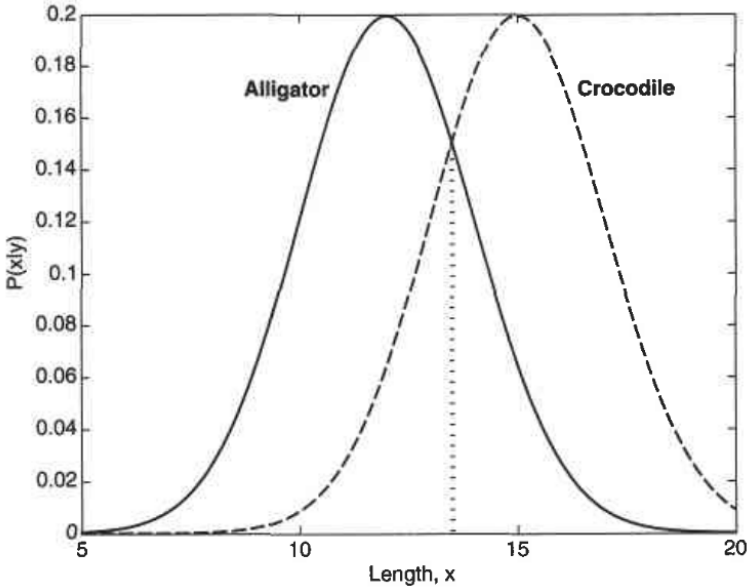
Vlastnosti naïvní bayesovské klasifikace

- **robustní vůči šumu** (izolované body jsou „zprůměrovány“) i **přeučení**, při chybějících hodnotách objekt vynechán nebo hodnoty odhadnuty z pravděpodobnostního rozložení atributu
- robustní vůči irelevantním atributům ($P(X|Y)$ rozložena uniformně)
- **korelované atributy snižují výkonnost** – přestává platit předpoklad nezávislosti atributů: $P(\mathbf{X}|Y=y) = \prod_{i=1}^{m} P(X_i|Y=y)$ se mění na $P(\mathbf{X}|Y=y) = P(X_i|Y=y), i = 1, \ldots, m$

→ **Bayesovské sítě (Bayesian (belief) networks)** – uvažování závislostí mezi atributy v modelu a při klasifikaci

- z pravděpodobnostních rozložení $P(X|Y)$ a $P(Y)$ je možné určit ideální decision boundaries (= hodnota $\mathbf{x}$ pro kterou $P(Y=y_i)P(\mathbf{X}=\mathbf{x}|Y=y_i) = P(Y=y_j)P(\mathbf{X}=\mathbf{x}|Y=y_j)$), např., a minimální chybovost (**Bayes error rate**, = suma ploch pod částmi grafů $P(Y=y|\mathbf{X})$ odpovídajícími misklasifikaci) jakéhokoliv klasifikačního modelu

Vlastnosti naïvní bayesovské klasifikace



Asociační analýza

Asociační analýza

= nalezení „zajímavých“ vztahů (asociací) skrytých v rozsáhlých datech, extrakcí častých vzorů typicky ve formě implikací mezi množinami položek (atributy, items) dat = **asociační pravidla** nebo častých množin položek = **frequent itemsets**

- např. (typicky) analýza nákupů zákazníků v obchodech, nákupních košíků (transakcí) = **market basket analysis**, např.
  - ke zjištění nákupního chování zákazníků pro podporu prodeje (doporučování zboží), naskladnění (optimalizace skladů) atd.
  - pravidla zachycující vztahy mezi prodejemi zboží („hodně zákazníků kupujících toto kupuje i tamto“), např. $\{ketamin\} \longrightarrow \{hašiš\}$
  - ale také např. bioinformatika (geny se související funkcí), medicína (související symptomy, nemoci), web (stránky přistupované společně), vědy o Zemi (vazby mezi oceány, souší, atmosférou) aj.
- binární reprezentace dat: položky (items) $\sim$ **asymetrické binární atributy** – podstatnější prezence položky v transakci, rozšíření na nebinární např. počet, cena

! data obvykle rozsáhlá $\Rightarrow$ výpočetní náročnost, některé vzory nahodilé nebo triviální $\Rightarrow$ výběr „zajímavých“

Asociační analýza: příklad dat

tID	položky košíku (items)
1	extáze, hašiš, LSD, pervitin
2	hašiš, ketamin, LSD
3	heroin, kokain, LSD
4	heroin, kokain
5	extáze, hašiš, heroin, kokain, LSD
6	extáze, hašiš, ketamin, pervitin
7	extáze, hašiš, heroin, ketamin, LSD, pervitin
8	extáze, hašiš, heroin, pervitin

tID	extáze	hašiš	heroin	ketamin	kokain	LSD	pervitin
1	1	1	0	0	0	1	1
2	0	1	0	1	0	1	0
3	0	0	1	0	1	1	0
4	0	0	1	0	1	0	0
5	1	1	1	0	1	1	0
6	1	1	0	1	0	0	1
7	1	1	1	1	0	1	1
8	1	1	1	0	0	0	1

Základní pojmy

- položky (**items**) $I = \{i_1, \ldots, i_m\}$, **transakce** $T = \{t_1, \ldots, t_n\}, t_i \subseteq I$, šířka transakce $|t_i|$, **itemset** $X \subseteq I$, k-itemset: $|X| = k$, transakce t_i obsahuje itemset X: $X \subseteq t_i$
  - **itemset support count**: $\sigma(X) = |\{t_i \in T \mid X \subseteq t_i|$
  - např. $\sigma(\{heroin, kokain\}) = 3$
- **asociační pravidlo** $X \longrightarrow Y, X, Y \subseteq I, X \cap Y = \emptyset$, míry „síly“:
  - **support** $\sim$ „podpora“ pravidla daty = relativní četnost $X \cup Y$ ve (všech) transakcích:

$$s(X \longrightarrow Y) = \frac{\sigma(X \cup Y)}{n}$$

  - **confidence** $\sim$ „důvěryhodnost/spolehlivost“ pravidla v datech = relativní četnost Y v transakcích obsahujících X ... odhad $P(Y|X)$:

$$c(X \longrightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$$

  - např. $s(\{extáze, hašiš\} \longrightarrow \{pervitin\}) = 4/8$, $c(\{extáze, hašiš\} \longrightarrow \{pervitin\}) = 4/5$
  - NE nutně kauzalita! – vyžaduje expertní znalost atributů a typicky zahrnuje vztahy v čase, asociační pravidlo jen vztah mezi výskyty položek v datech

Problém extrakce/dolování pravidel

Association rule mining problem

$=$ v dané množině transakcí nalezení všech asociačních pravidel s danou minimální support s_{min} a minimální confidence c_{min}

- exponenciálně mnoho možných pravidel pro ověření $s \geq s_{min}$ a $c \geq c_{min}$: $3^m - 2^{m+1} + 1$, m počet items, v příkladu 1932 ($m = 7$)

$\rightarrow$ $s(X \longrightarrow Y)$ závisí jen na $\sigma(X \cup Y) \Rightarrow$ (obvykle)

1. nalezení všech častých množin položek, **frequent itemsets** $=$ s $\sigma/n \geq s_{min}$ – výpočetně náročnější
2. vytvoření všech pravidel z frequent itemsets s $c \geq c_{min}$

Generování frequent itemsets

- exponenciálně mnoho možných (candidate) itemsets pro ověření $\sigma \geq ns_{min}$: $2^m - 1$ (bez $\emptyset$), v příkladu 127 $\approx$ (množinové) porovnání každé se všemi transakcemi

$\rightarrow$ zmenšení počtu candidate itemsets a porovnání

Princip Apriori

Všechny podmnožiny frequent itemset jsou také frequent itemsets.

$\Rightarrow$ všechny nadmnožiny infrequent itemset jsou také infrequent itemsets $\Leftarrow \sigma(X) \leq \sigma(Y)$ pro $Y \subseteq X$ (anti-monotonie) $\rightarrow$ **support-based pruning**

Algoritmus Apriori

- první pro dolování asociačních pravidel využívající support-based pruning

1. vygenerování **candidate** k-itemsets C_k z F_{k-1} (princip Apriori), počínaje $k = 1$ ($C_1 = \{\{i\}, i \in I\}$)
2. frequent k-itemsets $F_k = \{X \in C_k \,|\, \sigma(X) \geq ns_{min}\}$ a opakování pro $k = k + 1$ pokud $F_k \neq \emptyset$

- level-wise, generate-and-test

Generování candidate k-itemsets

- úplnost: musí zahrnovat všechny frequent k-itemsets
- „neredundance“ / následný **pruning**: stačí jen takové, jejichž všechny $(k-1)$-prvkové podmnožiny jsou frequent itemsets (princip Apriori) – POZOR! ne všechny pak nutně frequent!
- unikátnost: žádný by neměl být vygenerován vícekrát
- $|C_1| = m = \binom{m}{1}$, $|C_2| = \binom{|F_1|}{2}$, $|C_{k\geq 3}|$ závislé na F_{k-1}
- brute-force: všech $\binom{m}{k}$ k-itemsets, např. $\sum_{k=1}^{3} \binom{7}{k} = 7 + 21 + 35 = 63$
- $F_{k-1} \times F_1$: rozšíření všech frequent $(k-1)$-itemsets o každý frequent 1-itemset (item), maximálně $|F_{k-1}| \times |F_1|$, úplnost, ale ne unikátnost $\rightarrow$ lexikografické uspořádání items a rozšíření jen o větší než všechny v $(k-1)$-itemset, stále redundantní (ale heuristiky pro pruning, např. každý obsažený item musí být v alespoň $k-1$ frequent $(k-1)$-itemsets, jinak infrequent)
- $F_{k-1} \times F_{k-1}$: lexikografické uspořádání items a sjednocení každých dvou frequent $(k-1)$-itemsets se stejnými prvními $k-2$ items, úplnost, unikátnost, pořád redundantní, v algoritmu Apriori (s ověřením častosti u zbývajících $k-2$ $(k-1)$-prvkových podmnožin)

Algoritmus Apriori: příklad

$s_{min} = 0.5 \Rightarrow \sigma \geq 8 \cdot 0.5 = 4$

$C_1 = \{\{i\}, i \in I\} =$
$\{\{extáze\}, \{hašiš\}, \{heroin\}, \{ketamin\}, \{kokain\}, \{LSD\}, \{pervitin\}\}$
$F_1 = \{X \in C_1 \,|\, \sigma(X) \geq 4\} = \{\{extáze\}, \{hašiš\}, \{heroin\}, \{LSD\}, \{pervitin\}\}$

Algoritmus Apriori: příklad

$s_{min} = 0.5 \Rightarrow \sigma \geq 8 \cdot 0.5 = 4$

$C_1 = \{\{i\}, i \in I\} =$
$\{\{extáze\}, \{hašiš\}, \{heroin\}, \{ketamin\}, \{kokain\}, \{LSD\}, \{pervitin\}\}$
$F_1 = \{X \in C_1 \,|\, \sigma(X) \geq 4\} = \{\{extáze\}, \{hašiš\}, \{heroin\}, \{LSD\}, \{pervitin\}\}$
$C_2 =$
$\{\{extáze, hašiš\}, \{extáze, heroin\}, \{extáze, LSD\}, \{extáze, pervitin\}, \{hašiš, heroin\},$
$\{hašiš, LSD\}, \{hašiš, pervitin\}, \{heroin, LSD\}, \{heroin, pervitin\},$
$\{LSD, pervitin\}\}$
$F_2 = \{\{extáze, hašiš\}, \{extáze, pervitin\}, \{hašiš, LSD\}, \{hašiš, pervitin\}\}$

Generování frequent itemsets

Algoritmus Apriori: příklad

$s_{min} = 0.5 \Rightarrow \sigma \geq 8 \cdot 0.5 = 4$

$C_1 = \{\{i\}, i \in I\} =$
$\{\{extáze\}, \{hašiš\}, \{heroin\}, \{ketamin\}, \{kokain\}, \{LSD\}, \{pervitin\}\}$
$F_1 = \{X \in C_1 \,|\, \sigma(X) \geq 4\} = \{\{extáze\}, \{hašiš\}, \{heroin\}, \{LSD\}, \{pervitin\}\}$
$C_2 =$
$\{\{extáze, hašiš\}, \{extáze, heroin\}, \{extáze, LSD\}, \{extáze, pervitin\}, \{hašiš, heroin\},$
$\{hašiš, LSD\}, \{hašiš, pervitin\}, \{heroin, LSD\}, \{heroin, pervitin\},$
$\{LSD, pervitin\}\}$
$F_2 = \{\{extáze, hašiš\}, \{extáze, pervitin\}, \{hašiš, LSD\}, \{hašiš, pervitin\}\}$
$C_3 = \{\{extáze, hašiš, pervitin\}, \{\sout{hašiš, LSD, pervitin}\}\}$
$F_3 = \{\{extáze, hašiš, pervitin\}\}$

Generování frequent itemsets

Algoritmus Apriori: příklad

$s_{min} = 0.5 \Rightarrow \sigma \geq 8 \cdot 0.5 = 4$

$C_1 = \{\{i\}, i \in I\} =$
$\{\{extáze\}, \{hašiš\}, \{heroin\}, \{ketamin\}, \{kokain\}, \{LSD\}, \{pervitin\}\}$
$F_1 = \{X \in C_1 \,|\, \sigma(X) \geq 4\} = \{\{extáze\}, \{hašiš\}, \{heroin\}, \{LSD\}, \{pervitin\}\}$
$C_2 =$
$\{\{extáze, hašiš\}, \{extáze, heroin\}, \{extáze, LSD\}, \{extáze, pervitin\}, \{hašiš, heroin\},$
$\{hašiš, LSD\}, \{hašiš, pervitin\}, \{heroin, LSD\}, \{heroin, pervitin\},$
$\{LSD, pervitin\}\}$
$F_2 = \{\{extáze, hašiš\}, \{extáze, pervitin\}, \{hašiš, LSD\}, \{hašiš, pervitin\}\}$
$C_3 = \{\{extáze, hašiš, pervitin\}, \{hašiš, LSD, pervitin\}\}$
$F_3 = \{\{extáze, hašiš, pervitin\}\}$
$C_4 = F_4 = \emptyset$

Generování frequent itemsets

Zjištění itemset support count

* pro candidate itemsets $X \in C_k$: $\sigma(X) = |\{t_i \in T \,|\, X \subseteq t_i| \geq ns_{min}$
* (množinové) porovnání všech X se všemi transakcemi t_i

→ (lexikografické) procházení všech k-itemsets $Y \subseteq t_i$ pro všechny t_i a počítání případů $Y = X$ pro X – využití např. hešovacích stromů s podmnožinami candidate itemsets v uzlech

Vytvoření pravidel

- z frequent k-itemset X až $2^k - 2$ možných pravidel (bez $\emptyset \longrightarrow X$ a $X \longrightarrow \emptyset$)

$=$ $\mathbf{X \setminus Y \longrightarrow Y}$ pro frequent itemset X a všechny $Y \subset X, Y \neq \emptyset$, s $c(X \setminus Y \longrightarrow Y) \geq c_{min}$ – $s(X \setminus Y \longrightarrow Y) \geq s_{min}$ platí $(\sigma(X) \geq n s_{min})$

- $\sigma(X)$ i $\sigma(X \setminus Y)$ pro ověření $c = \frac{\sigma(X)}{\sigma(X \setminus Y)} \geq c_{min}$ již byly zjištěny při generování (frequent itemsets) $X \setminus Y$ a X
- confidence nemá vlastnost (anti-)monotonie, např. $c(X_1 \longrightarrow Y_1) <=> c(X_2 \longrightarrow Y_2)$ pro $X_1 \subseteq X_2$ a $Y_1 \subseteq Y_2$, ale

$\rightarrow$ jestliže $c(X \setminus Y \longrightarrow Y) < c_{min}$, pak $c(X \setminus Y' \longrightarrow Y') < c_{min}, Y \subseteq Y'$ $(\Leftarrow$ $\sigma(X \setminus Y') \geq \sigma(X \setminus Y))$

Algoritmus Apriori

- pro každý frequent k-itemset X, $k \geq 2$:
  1. vygenerování (frequent) m-itemsets H_m z H_{m-1}, počínaje $m = 1$ $(H_1 = \{\{i\}, i \in X\})$
  2. pro všechny $Y \in H_m$ pravidlo $X \setminus Y \longrightarrow Y$, pokud $c(X \setminus Y \longrightarrow Y) \geq c_{min}$, jinak $H_m = H_m \setminus \{Y\}$, a opakování pro $m = m + 1$ pokud $H_m \neq \emptyset$ a $m + 1 < k$
- vygenerování m-itemsets $=$ vygenerování candidate m-itemsets

Vytvoření pravidel

Algoritmus Apriori: příklad

$s_{min} = 0.5, c_{min} = 0.8$

pro $X = \{extáze, hašiš\}$: $H_1 = \{\{i\}, i \in X\} = \{\{extáze\}, \{hašiš\}\}$
pravidla $\{hašiš\} \longrightarrow \{extáze\}$ $(c(X \setminus Y \longrightarrow Y) = \frac{\sigma(X)}{\sigma(X \setminus Y)} \geq 0.8)$, $\{extáze\} \longrightarrow \{hašiš\}$

Vytvoření pravidel

Algoritmus Apriori: příklad

$s_{min} = 0.5, c_{min} = 0.8$

pro $X = \{extáze, hašiš\}$: $H_1 = \{\{i\}, i \in X\} = \{\{extáze\}, \{hašiš\}\}$
pravidla $\{hašiš\} \longrightarrow \{extáze\}$ ($c(X \setminus Y \longrightarrow Y) = \frac{\sigma(X)}{\sigma(X \setminus Y)} \geq 0.8$), $\{extáze\} \longrightarrow \{hašiš\}$
pro $X = \{extáze, pervitin\}$: $H_1 = \{\{extáze\}, \{pervitin\}\}$
pravidla $\{pervitin\} \longrightarrow \{extáze\}$, $\{extáze\} \longrightarrow \{pervitin\}$

Vytvoření pravidel

Algoritmus Apriori: příklad

$s_{min} = 0.5, c_{min} = 0.8$

pro $X = \{extáze, hašiš\}$: $H_1 = \{\{i\}, i \in X\} = \{\{extáze\}, \{hašiš\}\}$
pravidla $\{hašiš\} \longrightarrow \{extáze\}$ ($c(X \setminus Y \longrightarrow Y) = \frac{\sigma(X)}{\sigma(X \setminus Y)} \geq 0.8$), $\{extáze\} \longrightarrow \{hašiš\}$
pro $X = \{extáze, pervitin\}$: $H_1 = \{\{extáze\}, \{pervitin\}\}$
pravidla $\{pervitin\} \longrightarrow \{extáze\}$, $\{extáze\} \longrightarrow \{pervitin\}$
pro $X = \{hašiš, LSD\}$: $H_1 = \{\{hašiš\}, \{LSD\}\}$
pravidlo $\{LSD\} \longrightarrow \{hašiš\}$, $H_1 = \{\{hašiš\}\}$

Vytvoření pravidel

Algoritmus Apriori: příklad

$s_{min} = 0.5, c_{min} = 0.8$

pro $X = \{extáze, hašiš\}$: $H_1 = \{\{i\}, i \in X\} = \{\{extáze\}, \{hašiš\}\}$
pravidla $\{hašiš\} \longrightarrow \{extáze\}$ $(c(X \setminus Y \longrightarrow Y) = \frac{\sigma(X)}{\sigma(X \setminus Y)} \geq 0.8)$, $\{extáze\} \longrightarrow \{hašiš\}$
pro $X = \{extáze, pervitin\}$: $H_1 = \{\{extáze\}, \{pervitin\}\}$
pravidla $\{pervitin\} \longrightarrow \{extáze\}$, $\{extáze\} \longrightarrow \{pervitin\}$
pro $X = \{hašiš, LSD\}$: $H_1 = \{\{hašiš\}, \{LSD\}\}$
pravidlo $\{LSD\} \longrightarrow \{hašiš\}$, $H_1 = \{\{hašiš\}\}$
pro $X = \{hašiš, pervitin\}$: $H_1 = \{\{hašiš\}, \{pervitin\}\}$
pravidlo $\{pervitin\} \longrightarrow \{hašiš\}$, $H_1 = \{\{hašiš\}\}$

Vytvoření pravidel

Algoritmus Apriori: příklad

$s_{min} = 0.5, c_{min} = 0.8$

pro $X = \{extáze, hašiš\}$: $H_1 = \{\{i\}, i \in X\} = \{\{extáze\}, \{hašiš\}\}$
pravidla $\{hašiš\} \longrightarrow \{extáze\}$ $(c(X \setminus Y \longrightarrow Y) = \frac{\sigma(X)}{\sigma(X \setminus Y)} \geq 0.8)$, $\{extáze\} \longrightarrow \{hašiš\}$
pro $X = \{extáze, pervitin\}$: $H_1 = \{\{extáze\}, \{pervitin\}\}$
pravidla $\{pervitin\} \longrightarrow \{extáze\}$, $\{extáze\} \longrightarrow \{pervitin\}$
pro $X = \{hašiš, LSD\}$: $H_1 = \{\{hašiš\}, \{LSD\}\}$
pravidlo $\{LSD\} \longrightarrow \{hašiš\}$, $H_1 = \{\{hašiš\}\}$
pro $X = \{hašiš, pervitin\}$: $H_1 = \{\{hašiš\}, \{pervitin\}\}$
pravidlo $\{pervitin\} \longrightarrow \{hašiš\}$, $H_1 = \{\{hašiš\}\}$
pro $X = \{extáze, hašiš, pervitin\}$: $H_1 = \{\{extáze\}, \{hašiš\}, \{pervitin\}\}$
pravidla $\{hašiš, pervitin\} \longrightarrow \{extáze\}$, $\{extáze, pervitin\} \longrightarrow \{hašiš\}$,
$\{extáze, hašiš\} \longrightarrow \{pervitin\}$

Vytvoření pravidel

Algoritmus Apriori: příklad

$s_{min} = 0.5, c_{min} = 0.8$

pro $X = \{extáze, hašiš\}$: $H_1 = \{\{i\}, i \in X\} = \{\{extáze\}, \{hašiš\}\}$

pravidla $\{hašiš\} \longrightarrow \{extáze\}$ $(c(X \setminus Y \longrightarrow Y) = \frac{\sigma(X)}{\sigma(X \setminus Y)} \geq 0.8)$, $\{extáze\} \longrightarrow \{hašiš\}$

pro $X = \{extáze, pervitin\}$: $H_1 = \{\{extáze\}, \{pervitin\}\}$

pravidla $\{pervitin\} \longrightarrow \{extáze\}$, $\{extáze\} \longrightarrow \{pervitin\}$

pro $X = \{hašiš, LSD\}$: $H_1 = \{\{hašiš\}, \{LSD\}\}$

pravidlo $\{LSD\} \longrightarrow \{hašiš\}$, $H_1 = \{\{hašiš\}\}$

pro $X = \{hašiš, pervitin\}$: $H_1 = \{\{hašiš\}, \{pervitin\}\}$

pravidlo $\{pervitin\} \longrightarrow \{hašiš\}$, $H_1 = \{\{hašiš\}\}$

pro $X = \{extáze, hašiš, pervitin\}$: $H_1 = \{\{extáze\}, \{hašiš\}, \{pervitin\}\}$

pravidla $\{hašiš, pervitin\} \longrightarrow \{extáze\}$, $\{extáze, pervitin\} \longrightarrow \{hašiš\}$,
$\{extáze, hašiš\} \longrightarrow \{pervitin\}$

$H_2 = \{\{extáze, hašiš\}, \{extáze, pervitin\}, \{hašiš, pervitin\}\}$

pravidla $\{pervitin\} \longrightarrow \{extáze, hašiš\}$,
$\{extáze\} \longrightarrow \{hašiš, pervitin\}$, $H_2 = \{\{extáze, hašiš\}, \{hašiš, pervitin\}\}$

Maximální frequent itemsets

- frequent itemsets bývá v (rozsáhlejších) datech mnoho $\rightarrow$ menší podmnožina frequent itemsets, ze kterých se dají všechny ostatní odvodit

$=$ frequent itemsets X, u kterých není žádná jejich nadmnožina $X' \supset X$ frequent, $\sigma(X') < ns_{min}$

- např. $\{heroin\}, \{hašiš, LSD\}, \{extáze, hašiš, pervitin\}$ (při $s_{min} = 0.5$)
- nejmenší podmnožina frequent itemsets, kde všechny ostatní $=$ podmnožiny – (stále) exponenciálně mnoho

$\rightarrow$ algoritmy pro generování pouze maximálních, např. hypergraph transversal (vertex cover)

- „neposkytují info“ o support count podmnožin – potřeba zjistit

Uzavřené (closed) frequent itemsets

- minimální podmnožina frequent itemsets „s info“ o support count všech ostatních

$=$ frequent itemsets, které jsou **uzavřené (closed) itemsets** $=$ itemsets (i infrequent) X, u kterých nemá žádná jejich nadmnožina $X' \supset X$ stejný support count, $\sigma(X') < \sigma(X)$ (anti-monotonie σ)

- např. $\{hašiš\}, \{heroin\}, \{LSD\}, \{extáze, hašiš\}, \{hašiš, LSD\}$, $\{extáze, hašiš, pervitin\}$ (při $s_{min} = 0.5$), navíc dalších 13 uzavřených (infrequent)
- všechny ostatní frequent itemsets Y = podmnožiny X – **support count** = maximální support count větších uzavřených frequent itemsets, $\sigma(Y) = \max_{X \supset Y} \sigma(X)$ (každá transakce obsahující uzavřené obsahuje i podmnožinu), např. $\sigma(\{extáze\}) = \max_{X \in \{\{extáze, hašiš\}, \{extáze, hašiš, pervitin\}\}} \sigma(X) = \max\{5, 4\} = 5$

$\rightarrow$ algoritmy pro generování pouze uzavřených (frequent) itemsets, např. výpočet formálních konceptů ve FCA

- často postačující (místo všech frequent), např. pro nevytváření **redundantních pravidel** $X \longrightarrow Y$ = existuje $X' \longrightarrow Y'$, $X \subseteq X'$, $Y \subseteq Y'$, se stejnými support a confidence, např. $\{extáze\} \longrightarrow \{pervitin\}$, $\{pervitin\} \longrightarrow \{extáze\}$, $\{pervitin\} \longrightarrow \{hašiš\}$

Uzavřené (closed) frequent itemsets

- minimální podmnožina frequent itemsets „s info“ o support count všech ostatních

= frequent itemsets, které jsou **uzavřené (closed) itemsets** = itemsets (i infrequent) X, u kterých nemá žádná jejich nadmnožina $X' \supset X$ stejný support count, $\sigma(X') < \sigma(X)$ (anti-monotonie σ)

- např. $\{hašiš\}, \{heroin\}, \{LSD\}, \{extáze, hašiš\}, \{hašiš, LSD\}$, $\{extáze, hašiš, pervitin\}$ (při $s_{min} = 0.5$), navíc dalších 13 uzavřených (infrequent)
- všechny ostatní frequent itemsets Y = podmnožiny X – **support count** = maximální support count větších uzavřených frequent itemsets, $\sigma(Y) = \max_{X \supset Y} \sigma(X)$ (každá transakce obsahující uzavřené obsahuje i podmnožinu), např. $\sigma(\{extáze\}) = \max_{X \in \{\{extáze, hašiš\}, \{extáze, hašiš, pervitin\}\}} \sigma(X) = \max\{5, 4\} = 5$

→ algoritmy pro generování pouze uzavřených (frequent) itemsets, např. výpočet formálních konceptů ve FCA

- často postačující (místo všech frequent), např. pro nevytváření **redundantních pravidel** $X \longrightarrow Y$ = existuje $X' \longrightarrow Y'$, $X \subseteq X'$, $Y \subseteq Y'$, se stejnými support a confidence, např. $\{extáze\} \longrightarrow \{pervitin\}$, $\{pervitin\} \longrightarrow \{extáze\}$, $\{pervitin\} \longrightarrow \{hašiš\}$
- množina frequent $\supseteq$ uzavřených frequent $\supseteq$ maximálních frequent itemsets

Generování frequent itemsets – přístupy

- výkonnost algoritmu Apriori klesá s hustotou dat (průměrnou šířkou transakcí) – více itemsets v transakcích $\Rightarrow$ náročnější zjišťování itemset support count, roste maximální velikost frequent itemsets $\Rightarrow$ více candidate itemsets $\rightarrow$ jiné metody

Průchod uspořádanou množinou (svazem) itemsets

- od obecnějších (menších) ke specifičtějším (větším) nebo opačně: první např. algoritmus Apriori, výhodnější pro menší frequent itemsets, druhý pro maximální v hustších datech – využití principu Apriori (ne podmnožiny), také kombinace
- po třídách ekvivalentních: např. u level-wise algoritmů (Apriori) podle velikosti itemset, další podle společného prefixu (prvních uspořádaných items) nebo suffixu dané délky
- do šířky (breadth-first) nebo do hloubky (depth-first): první např. algoritmus Apriori, u nadmnožin/podmnožin využití principu Apriori, druhý často pro maximální (různý pruning podmnožin)

Generování frequent itemsets – přístupy

Reprezentace transakčních dat

- vliv u zjišťování itemset support count
- horizontální = množina transakcí jako podmnožin items – množinové porovnání (candidate) itemset s transakcemi → výkonné datové struktury, např. stromy, častější (včetně algoritmu Apriori)
- vertikální = množina items jako seznamů ID transakcí – průnik seznamů items v itemsets = seznam pro itemset – délka klesá s velikostí itemset, počáteční velké → úsporné datové struktury, např. komprimované

Algoritmus FP-Growth

- reprezentace vstupních dat pomocí stromu **FP-tree** = sestupně dle support count uspořádané frequent items transakcí a počty transakcí v uzlech na cestě stomem od (umělého) kořene do listu – sdílené uzly pro společné prefixy transakcí
- generování frequent itemsets přímo z FP-tree: pro items jako suffixy hledání zvětšujících se frequent itemsets na cestách (prefix path) z nejhloubějších uzlů s item ke kořeni rekurzivně po úrovních stromu „odspodu odřezáváním“ listů/suffixu (s aktualizací počtů transakcí v uzlech a „prořezáním“ stromu → conditional FP-tree)

Vyhodnocení asociativních vzorů

- potenciálně mnoho (u reálných rozsáhlých dat), mnoho nezajímavých $\rightarrow$ kritéria zajímavosti/kvality
- objektivní – statistické míry z dat, např. vzory s vzájemně nezávislými items nebo obsažené v málo transakcích nezajímavé (šum), např. support, confidence, korelace
- subjektivní – např. neobvyklé, nečekané, užitečné info, obtížné použít, typicky potřeba expertní doménové znalosti o datech, např. konceptuální hierarchie, profit items, interpretace/verifikace (při vizualizaci), omezení

Objektivní míry zajímavosti

- založené na četnostech v **kontingenční tabulce** – např. pro dva items x, y

	y	$\overline{y}$	
x	f_{11}	f_{10}	f_{1-}
$\overline{x}$	f_{01}	f_{00}	f_{0-}
	f_{-1}	f_{-0}	n

$\overline{x}$ není v transakci, f_{xy} četnost x a y současně v transakci, f_{1-} (f_{-1}) support count x (y)

Míry zajímavosti

- omezení support a confidence: items s velkými rozdíly „support“ ($\frac{\sigma}{n}$, např. většina nízký, některé vysoký) – jaký minimální s_{min}?, „support“ Y vyšší než (dostatečná) confidence $X \longrightarrow Y$ – ignorován, $X \longrightarrow Y$ „zřejmá“

→ **lift** = confidence $X \longrightarrow Y$ v poměru k „support“ Y:

$$lift(X \longrightarrow Y) = \frac{nc(X \longrightarrow Y)}{\sigma(Y)}$$

pro $X = \{x\}, Y = \{y\}$ rovna **interest factor**:

$$I(x,y) = \frac{n\sigma(\{x\} \cup \{y\})}{\sigma(\{x\})\sigma(\{y\})} = \frac{nf_{11}}{f_{1-}f_{-1}}$$

$\sim$ porovnání f_{11} s tzv. baseline četností při statistické nezávislosti x a $y = \frac{f_{1-}f_{-1}}{n}$
($\frac{f_{11}}{n} = \frac{f_{1-}}{n}\frac{f_{-1}}{n}$ odhady pravděpodobností $P(x,y) = P(x)P(y)$) $\Rightarrow$
při $I(x,y) = 1$ x, y nezávislé, při > 1 kladně korelované, při < 1 záporně korelované
omezení: vysoká $f_{11} \rightsquigarrow I(x,y) = 1$, nízké f_{11}, f_{1-}, f_{-1} (vysoká f_{00}) $\rightsquigarrow I(x,y) > 1$

Míry zajímavosti

- **ϕ-koeficient** $\sim$ korelační koeficient pro binární atributy ($\in [-1, 1]$):

$$\phi(x, y) = \frac{f_{11}f_{00} - f_{01}f_{10}}{\sqrt{f_{1-}f_{-1}f_{0-}f_{-0}}} = \frac{nf_{11} - f_{1-}f_{-1}}{\sqrt{f_{1-}f_{-1}f_{0-}f_{-0}}}$$

  omezení: stejný význam současného výskytu i nevýskytu x, y v transakcích $\Rightarrow$ vhodný pro symetrické binární atributy, ne invariantní ke změně velikosti dat

- **IS míra**:

$$IS(x, y) = \sqrt{I(x, y)\frac{\sigma(\{x\} \cup \{y\})}{n}} = \frac{\sigma(\{x\} \cup \{y\})}{\sqrt{\sigma(\{x\})\sigma(\{y\})}} = \frac{f_{11}}{\sqrt{f_{1-}f_{-1}}}$$

  pro x, y bitové vektory (přes transakce) rovna jejich kosinové podobnosti
  $(\mathbf{x} \cdot \mathbf{y} = \sigma(\{x\} \cup \{y\}),\ ||\mathbf{x}|| = \sqrt{\sigma(\{x\})})$
  $= \sqrt{c(\{x\} \longrightarrow \{y\})c(\{y\} \longrightarrow \{x\})}$ geometrický průměr
  omezení: pro nezávislé x, y $(\sigma(\{x\} \cup \{y\}) = \sigma(\{x\})\sigma(\{y\})) = \sqrt{\sigma(\{x\})\sigma(\{y\})}$ – může být vysoká

Míry zajímavosti

- další pro x, y např. odds ratio $\frac{f_{11}f_{00}}{f_{10}f_{01}}$, Piatetsky-Shapiro $\frac{f_{11}}{n} - \frac{f_{1-}f_{-1}}{n^2}$, Jaccard $\frac{f_{11}}{f_{1-}+f_{-1}-f_{11}}$, Laplace $\frac{f_{11}+1}{f_{1-}+2}$, conviction $\frac{f_{1-}f_{-0}}{nf_{10}}$, certainty factor $(\frac{f_{11}}{f_{1-}} - \frac{f_{-1}}{n})/(1 - \frac{f_{-1}}{n})$
- symetrická m: $m(X \longrightarrow Y) = m(Y \longrightarrow X)$, např. lift/interest factor, obecně pro itemsets, asymetrická např. confidence, pro asociační pravidla
- pro některé různá pořadí hodnot $\Rightarrow$ konfliktní info, např. lift/interest factor a ϕ-koeficient, vlastnosti:
  - inverze (invariance při inverzi) = stejná hodnota při inverzi items x, y jako bitových vektorů (přes transakce), tj. výměně f_{11} za f_{00} a f_{10} za f_{01}, např. ϕ-koeficient, nevhodné pro asymetrické binární atributy, vhodnější ne invariantní, např. lift/interest factor, IS míra
  - null addition = stejná hodnota při přidání transakcí bez items x, y, tj. zvýšení (pouze) f_{00}, např. IS míra, ne např. lift/interest factor, ϕ-koeficient
  - škálování = stejná hodnota při škálování support count items x, y, tj. násobení f_{xy} kladnými k_i: $k_1k_3f_{11}, k_2k_3f_{10}, k_1k_4f_{01}, k_2k_4f_{00}$, např. odds ratio, ostatní ne

Míry zajímavosti

Pro více než dva items

- některé pro dva možné použít i pro víc, např. lift (a support a confidence)
- některé, např. interest factor, IS míra, možné rozšířit na základě vícedimenzionální kontingenční tabulky – např. pro tři items x, y, z

z	y	$\overline{y}$	
x	f_{111}	f_{101}	f_{1-1}
$\overline{x}$	f_{011}	f_{001}	f_{0-1}
	f_{-11}	f_{-01}	f_{--1}

$\overline{z}$	y	$\overline{y}$	
x	f_{110}	f_{100}	f_{1-0}
$\overline{x}$	f_{010}	f_{000}	f_{0-0}
	f_{-10}	f_{-00}	f_{--0}

f_{--1} support count z, f_{1-1} support count $\{x, z\}$, $f_{--1} + f_{--0} = n$

- **interest factor** pro items $i_1, \ldots, i_k$ (k-itemset):

$$I(x_1, \ldots, i_k) = \frac{\sigma(\{x_1\} \cup \ldots \cup \{x_k\})}{\sigma(\{x_1\}) \ldots \sigma(\{x_k\})} = \frac{n^{k-1} f_{1\ldots1}}{f_{1-\ldots-} \cdots f_{-\ldots-1}}$$

$\sim$ porovnání $f_{1\ldots1}$ s baseline četností při statistické nezávislosti $x_1, \ldots, x_k = \frac{f_{1-\ldots-} f_{-\ldots-1}}{n^{k-1}}$

- jinak max, min nebo průměr hodnot pro všechny dvojice items, např. pro ϕ-koeficient

Simpsnův paradox

- pomocí vícedimenzionálních kontingenčních tabulek jsou problémem částečné asociace, např. jen pro dva items $\Rightarrow$ při podmínění hodnotami některých items se asociace mohou měnit $\rightarrow$ pokročilejší statistické techniky, např. loglineární modely

$=$ ovlivnění (včetně např. zrušení nebo **„převrácení“**) asociace mezi items jinými (neuvažovanými) faktory

- např. při stratifikaci dat: pro dvě skupiny transakcí A, B
  $c_A(X \longrightarrow Y) = \frac{\sigma_A(X \cup Y)}{\sigma_A(X)} < c_A(Z \longrightarrow Y)$ a $c_B(X \longrightarrow Y) < c_B(Z \longrightarrow Y)$, přitom
  $c(X \longrightarrow Y) = \frac{\sigma_A(X \cup Y) + \sigma_B(X \cup Y)}{\sigma_A(X) + \sigma_B(X)} > c(Z \longrightarrow Y) = \frac{\sigma_A(Z \cup Y) + \sigma_B(Z \cup Y)}{\sigma_A(Z) + \sigma_B(Z)} \Rightarrow$ potřeba správná stratifikace

skupina	tID	X	Y	Z
A	1	1	1	1
	2	1	0	0
B	3	1	0	1
	4	0	1	1
	5	0	0	1
	6	0	0	1
	7	0	0	1
	8	0	0	1

Shlukování

Shlukování

- $\sim$ seskupení/rozdělení dat do smysluplných, užitečných skupin (**shluků**) – zachycujících strukturu dat, usnadňujících jejich další zpracování (např. souhrny)
- použití a nesčetné aplikace v mnoha oborech a oblastech (DM/ML, informatiky):
- pro **porozumění datům** – lidé přirozeně seskupují/rozdělují (shlukování) a řadí (klasifikace) objekty do skupin/tříd na základě společných charakteristik $\rightarrow$ shluky = možné skupiny, shlukování = hledání skupin
  - biologie (taxonomie a jejich automatické vytváření, analýza genů), psychologie, medicína, sociální vědy (kategorizace a rozšíření nemocí, jevů), obchod (segmentace zákazníků), information retrieval (hierarchické seskupování výsledků vyhledávání) aj.
- pro **další zpracování dat** – abstrakce jednotlivých objektů celky (shluky), charakterizace reprezentativními objekty (**prototypy**) $\rightarrow$ shlukování = jejich hledání
  - např. sumarizace, vzorkování (místo celých velkých dat, srovnatelné výsledky), zvýšení výkonu, úspora dat (místo všech jednotlivých objektů, např. výpočet vzdáleností, přijatelná ztráta informace)

Shlukování (shluková analýza)

= nalezení/vytvoření skupin objektů záznamových dat, na základě popisu objektů (atributů), navzájem si co nejvíce **podobných (je mezi nimi vztah)** v rámci skupiny a různých (bez vztahu) od objektů mimo skupinu (v jiných skupinách)

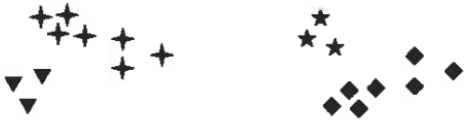
- skupina = **shluk (klastr, cluster)**: nepřesně/nejednoznačně vymezený pojem – co tvoří shluk (a co už ne)??, např., viz dále typy shluků
- forma klasifikace: rozřazení objektů do kategorií/tříd (class labels) $\sim$ shluk
  - ale u *klasifikace* (nové) objekty rozřazeny na základě modelu (předem) vytvořeného z objektů se známými kategoriemi/třídami = **supervised klasifikace** (z příkladů)
  - u shlukování vytvořené pouze z dat $\sim$ **unsupervised klasifikace**

$\sim$! **segmentace/rozklad (partitioning)** – i mimo tradiční pojetí shlukování, např. u grafů, jednoduché dělení dat na části, typicky dle hodnot atributů objektů, např. u obrazu podle barev pixelů, u obchodních dat apod.

Shlukování (shluková analýza)

(a) Original points.

(b) Two clusters.



(c) Four clusters.

(d) Six clusters.

Typy shlukování

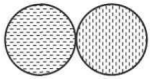
- způsoby seskupování/rozdělování objektů do shluků, **shlukování** = kolekce shluků
- **rozkladové** (partitional) = rozklad množiny objektů na nepřekrývající se podmnožiny (shluky), každý objekt v právě jedné, např.
- **hiearchické** (vnořené, nested) = množina vnořených stromově uspořádaných podmnožin objektů (shluků), každá kromě listů stromu sjednocením potomků (podshluků), kořen zahrnuje všechny objekty, listy často jeden, např., $\sim$ posloupnost rozkladových jako úrovní stromu
- **výlučné** = každý objekt v jednom shluku, např.
- **překrývající se** = objekt může být současně ve více shlucích, také „mezi“ shluky a v kterémkoliv z nich (než svévolně v jednom, lepší ale fuzzy), např.
- **fuzzy** = každý objekt v každém shluku ve stupni příslušnosti od 0 (vůbec není) do 1 (plně je), tj. shluky = fuzzy množiny, často navíc součet stupňů pro objekt roven 1 ($\sim$ **pravděpodobnostní**) a převod na výlučné (jen nejvyšší stupně)
- **úplné** = každý objekt ve shluku, např.
- **částečné** = objekt nemusí být ve shluku („nepatří“, šum, anomálie, „pozadí“), např.

Typy shluků

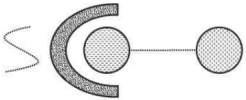
- smysluplnost, užitečnost shluků daná cíly analýzy $\Rightarrow$ různá pojetí/vymezení shluku
- **dobře oddělený (well-separated)** = všechny objekty ve shluku si navzájem podobnější (bližší) než jakémukoliv objektu mimo shluk – obvykle práh dostatečné podobnosti (blízkosti), např., nemusí být kulovitý, „ideál“ splňující jen přirozené oddělené shluky v datech
- **prototypový (prototype-based)** = každý objekt ve shluku podobnější (bližší) prototypovému/reprezentativnímu objektu definujícímu shluk než prototypu jiného shluku – pro numerické atributy typicky centroid (průměr objektů shluku), pro kategorické medoid (medián) $\sim$ **centrový** objekt a shluk, např., tendence kulovitého
- **grafový** = všechny objekty ve shluku „propojené“ (ne nutně navzájem) a nepropojené s objekty mimo shluk = komponenta grafu (uzly objekty, hrany propoje), propojení = do dané vzdálenosti = **styčný (contiguity)**, např., pro nepravidelné nebo provázané shluky, ale problém propojující šum
- **hustotový** = „hustá“ oblast objektů obklopená málo hustou oblastí (šum), hustota = např. počet objektů do dané vzdálenosti od objektu (centrová), např., pro nepravidelné nebo provázané shluky při šumu nebo anomáliích
- **konceptuální** = všechny objekty ve shluku sdílí nějakou vlastnost, např. společně tvoří nějaký specifický tvar shluku, např., $\rightsquigarrow$ pattern recognition

Typy shluků

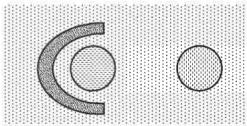
(a) Well-separated clusters. Each point is closer to all of the points in its cluster than to any point in another cluster.



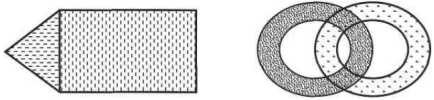
(b) Center-based clusters. Each point is closer to the center of its cluster than to the center of any other cluster.



(c) Contiguity-based clusters. Each point is closer to at least one point in its cluster than to any point in another cluster.



(d) Density-based clusters. Clusters are regions of high density separated by regions of low density.



(e) Conceptual clusters. Points in a cluster share some general property that derives from the entire set of points. (Points in the intersection of the circles belong to both.)

K-means

$=$ rozkladové shlukování s prototypovými shluky zadaného počtu K

- prototyp **centroid** $=$ (typicky) průměr objektů shluku s numerickými atributy, pro jiné (kategorické) atributy medián objektů shluku $=$ prototyp **medoid** (**K-medoids**) – pro objekty stačí jen míra blízkosti

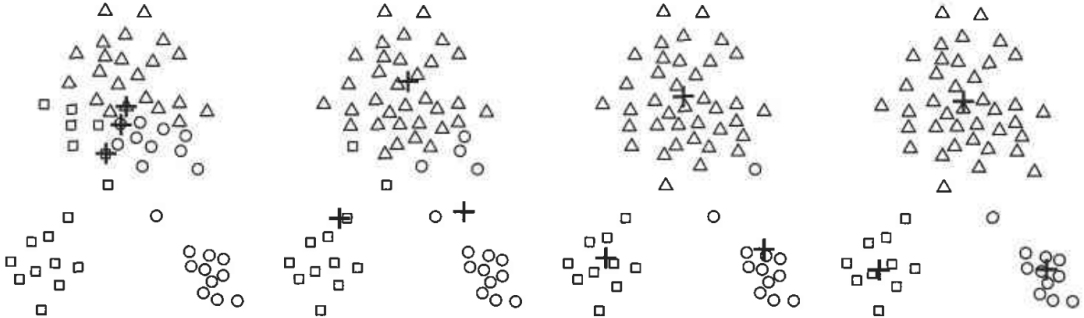
Základní algoritmus

- **zvolení** K počátečních prototypů – objektů z dat nebo i jiných

1. opakovaně přiřazení každého objektu k **nejbližšímu** prototypu $=$ vytvoření K shluků a
2. **aktualizace** prototypů dle objektů shluků, dokud se nezmění (prototypy a tedy i shluky)

- např. (centroid průměr)
- končí (konverguje, „neosciluje“) pro některé míry blízkostí a typy prototypů, např. euklidovská vzdálenost nebo kosinová podobnost a centroid jako průměr objektů
- ze začátku velké změny, ke konci minimální $\rightarrow$ konec při dosažení prahu minimálního počtu objektů přesunutých mezi shluky

K-means



(a) Iteration 1. (b) Iteration 2. (c) Iteration 3. (d) Iteration 4.

Přiřazení objektů k prototypu

- nejbližšímu na základě **míry blízkosti** ((ne)podobnosti) – např. (často) euklidovská (L_2 norm) vzdálenost, kosinová podobnost, také city block (Manhattan, taxicab, L_1 norm) vzdálenost, Jaccard koeficient
- opakované počítání blízkosti každého objektu každému prototypu $\rightarrow$ ušetření, např. půlící (bisecting) K-means

$=$ optimalizace hodnoty **objektivní funkce** (pro dané prototypy) – měří kvalitu shlukování (reprezentativnost prototypů pro objekty shluků), závisí na míře blízkosti, určuje typ prototypu $=$ aktualizaci $\Rightarrow$ optimalizační problém

Aktualizace prototypů

- dle objektů shluků pro optimalizaci hodnoty objektivní funkce (HOF) = **gradientní metoda** – nejlepší prototyp shluku = řešení nulové parciální derivace (pro daný prototyp/shluk) funkce, např.:
  - minimalizace **scatter** = **sumy squared error** shluků – error = míra blízkosti = vzdálenost d objektu x od (nejbližšího) prototypu p_i shluku C_i:

$$\sum_{i=1}^{K} \sum_{x \in C_i} d(x, p_i)^2$$

    $\Rightarrow$ pro d euklidovskou (L_2) vzdálenost nejlepší prototyp = centroid jako průměr objektů shluku
    pro **sumu absolute error** $\sum_{i=1}^{K} \sum_{x \in C_i} d(x, p_i)$ a city block (L_1) vzdálenost nejlepší prototyp = medoid jako medián objektů shluku (K-medoids)
  - maximalizace **koheze** shluků = sumy blízkostí = podobností s objektu od prototypu:

$$\sum_{i=1}^{K} \sum_{x \in C_i} s(x, p_i)$$

    $\Rightarrow$ pro s kosinovou podobnost nejlepší prototyp = centroid jako průměr objektů shluku

Aktualizace prototypů

! (pouze) lokální optimalizace – pro konkrétní prototypy $\Rightarrow$ klíčové počáteční!

Inkrementální aktualizace prototypů

= po přiřazení každého jednoho objektu k prototypu místo všech – aktualizace dvou nebo žádného prototypu

- možná rychlejší konvergence, ale i závislost shluků na pořadí objektů $\rightarrow$ náhodné

Zvolení počátečních prototypů

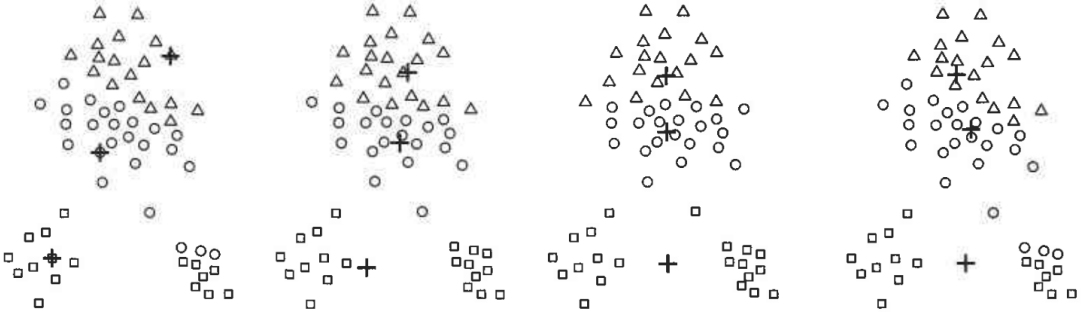
- běžně náhodné objekty, ale možné horší shluky (lokální optima) – i při „rovnoměrněji“ rozložených (v různých shlucích), např.

→ vícekrát náhodné a výběr nejlepších shluků (nejlepší optimum) – nemusí fungovat (ideálně počáteční prototyp v každém shluhu. . .)

→ prototypy shluků hierarchického shlukování z (náhodného) vzorku objektů – relativně malého (hierarchické shlukování je náročnější), ale většího než K

→ první náhodný objekt nebo prototyp všech objektů a každý další nejvzdálenější objekt od všech aktuálních – dobře oddělené, ale mohou být anomálie a výpočet → na (náhodném) vzorku objektů

Zvolení počátečních prototypů



(a) Iteration 1. (b) Iteration 2. (c) Iteration 3. (d) Iteration 4.

Postprocessing shluků

- pro další zlepšení HOF (často lokální optimalizace) – bez zvýšení K

→ střídavé rozdělování a spojování shluků – možný „únik“ z lokálního optima při zachování počtu shluků:

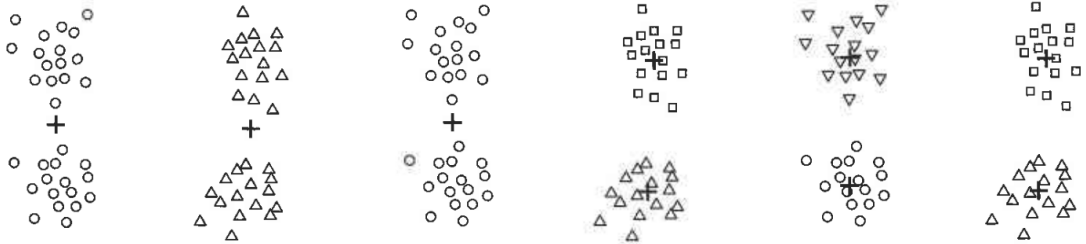
  - rozdělení shluku: s nejhorší HOF, nejvyšší směrodatnou odchylkou nějakého atributu aj., nový prototyp – nejvzdálenější objekt od prototypů, objekt nejvíc zhoršující HOF, náhodný aj.
  - spojení shluků: s nejbližšími prototypy, s nejmenším zhoršením HOF ($\sim$ prototypová a Wardova metoda v hierarchickém shlukování) aj., zrušení prototypu a přiřazení objektů k jiným – shluku s nejmenší HOF

Půlící (bisecting) K-means

= počínaje shlukem všech objektů, opakované **půlení zvoleného shluku** na dva pomocí základního K-means, dokud není K shluků

- např. (centroid průměr)
- vícekrát půlení a výběr toho s nejlepší HOF
- volba shluku: s nejvíce objekty, nejhorší HOF aj.
- postprocessing základním K-means s prototypy (z půlícího) jako počátečními – shluky (z půlícího) nemusí být lokální optimum objektivní funkce (základní K-means používáno „lokálně“ – půlení konkrétních shluků)
- méně závislé na počátečních prototypech (výběr z více půlení a jen dva prototypy v každém)
- posloupnost shlukování z iterací půlení = hierarchické shlukování

Půlící (bisecting) K-means



(a) Iteration 1. (b) Iteration 2. (c) Iteration 3.

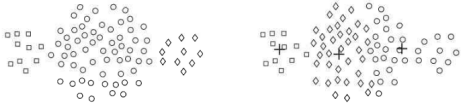
K-means

- jednoduché, rychlé (pokud K malé) – časová složitost lineární ve velikosti vstupních dat (počet iterací je omezený), obvykle ale potřeba více spuštění

Problémy

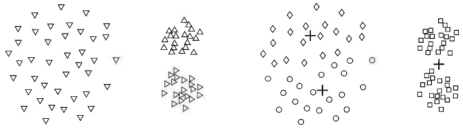
- prázdný shluk = jen prototyp $\rightarrow$ jiný prototyp: nejvzdálenější objekt od ostatních prototypů (nejvíce zhoršuje HOF) nebo objekt ze shluku s nejhorší HOF (= rozdělení shluku)
- **anomálie** („nepatří“ do žádného shluku) – prototypy mohou být horší reprezentanti shluku $\rightarrow$ nalezení a odstranění před nebo po shlukování (pokud nejsou zajímaví nebo potřební jako všechny objekty) – např. objekty nejvíc zhoršující HOF, malé shluky (anomálií), prototyp = medián objektů shluku (K-medoids)
- ne-dobře oddělené nebo ne-kulovité „přirozené“ shluky v datech nebo s (výrazně) rozdílnými velikostmi nebo hustotami objektů $\rightsquigarrow$ „smíchání“ částí shluků nebo spojení menších s částmi větších nebo dohromady, např. $\rightarrow$ vyšší K (části jako (pod)shluky), např.

K-means



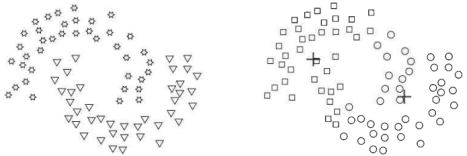
(a) Original points. (b) Three K-means clusters.

K-means with clusters of different size.



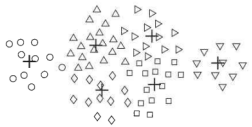
(a) Original points. (b) Three K-means clusters.

K-means with clusters of different density.

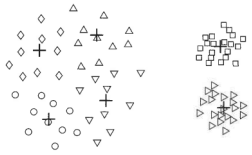


(a) Original points. (b) Two K-means clusters.

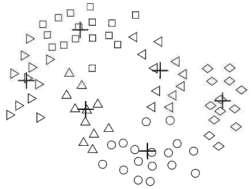
K-means with non-globular clusters.



(a) Unequal sizes.



(b) Unequal densities.



(c) Non-spherical shapes.

Hierarchické shlukování

1. **aglomerativní** = opakované **spojování dvou shluků** až do jednoho počínaje jednotlivými objekty jako jednoprvkovými shluky (singleton)
2. **divizivní** = opakované **rozdělování nějakého shluku** až na singletony počínaje jedním shlukem se všemi objekty

- grafické zobrazení pomocí **dendrogramu** = stromový diagram zobrazující (inkluzivní) vztahy mezi shluky (hrany) i pořadí jejich spojování/rozdělování (výška), např.
- pro aplikace vyžadující hierarchii shluků, např. taxonomie tříd objektů

Aglomerativní – základní algoritmus

1. každý objekt jako jednoprvkový shluk (singleton) a **blízkost shluků** = blízkost objektů
2. opakovaně spojení (sjednocení) dvou nejbližších shluků, dokud nezůstane jeden

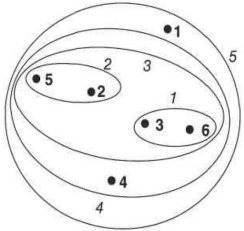
- $2n - 1$ shluků, n počet objektů

Blízkost shluků

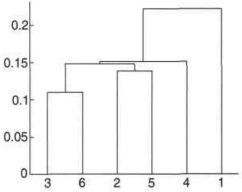
- grafové shluky:
  - **single link/min** = blízkost nejbližších dvou objektů z různých shluků (délka nejkratší hrany mezi dvěma uzly z různých komponent), např. $\rightsquigarrow$ styčné shluky, problém šum a anomálie; $\alpha_A = \alpha_B = 1/2, \beta = 0, \gamma = -1/2$
  - **complete link/max** = blízkost nejvzdálenějších dvou objektů z různých shluků, např., $\rightsquigarrow$ tendence kulovitých shluků; $\alpha_A = \alpha_B = \gamma = 1/2, \beta = 0$
  - **průměrná (group average)** = průměr blízkostí každých dvou objektů z různých shluků, např.; $\alpha_A = n_A/(n_A + n_B), \alpha_B = n_B/(n_A + n_B), \beta = \gamma = 0$
- prototypové shluky:
  - **prototypová (centroidní)** = blízkost prototypů shluků – možné **inverze** = bližší shluky spojené později (blízkosti spojovaných shluků nemusí tvořit neklesající posloupnost); $\alpha_A = n_A/(n_A + n_B), \alpha_B = n_B/(n_A + n_B), \beta = -n_A n_B/(n_A + n_B)^2, \gamma = 0$
  - **Wardova metoda** = zhoršení HOF po spojení shluků (= optimalizace objektivní funkce K-means), např. $\sim$ průměrná při druhé mocnině blízkosti objektů, použití pro zvolení počátečních prototypů K-means; $\alpha_A = (n_A + n_Y)/(n_A + n_B + n_Y), \alpha_B = (n_B + n_Y)/(n_A + n_B + n_Y), \beta = -n_Y/(n_A + n_B + n_Y), \gamma = 0$
- Lance-Williams formula: n_A počet objektů shluku A

$$p(A \cup B, Y) = \alpha_A p(A, Y) + \alpha_B p(B, Y) + \beta p(A, B) + \gamma |p(A, Y) - p(B, Y)|$$

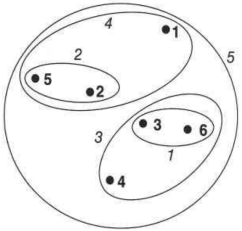
Blízkost shluků



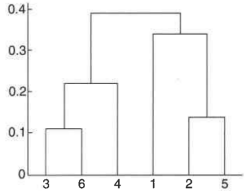
(a) Single link clustering.



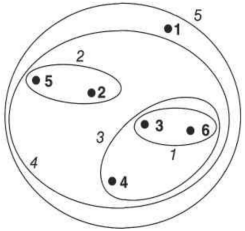
(b) Single link dendrogram.



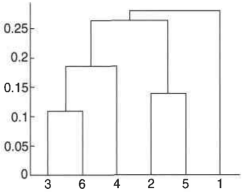
(a) Complete link clustering.



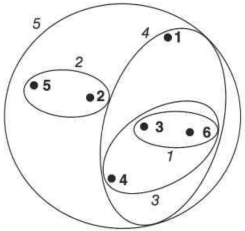
(b) Complete link dendrogram.



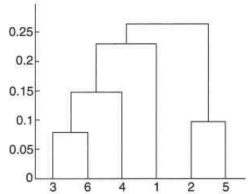
(a) Group average clustering.



(b) Group average dendrogram.



(a) Ward's clustering.



(b) Ward's dendrogram.

Blízkost shluků

- **nevážená**/**vážená** = uvažovaný/neuvažovaný počet objektů shluků $\Rightarrow$ stejné/různé váhy objektů různých shluků (např. různé třídy objektů), např. průměrná nevážená výše, vážená $\alpha_A = \alpha_B = 1/2, \beta = \gamma = 0$

Aglomerativní hierarchické shlukování

- jednoduché, ale časová složitost $O(N^2 \log N)$ ve velikosti N vstupních dat (potřeba blízkosti mezi každými dvěma objekty a zjištění/vyhledání pro shluky)

Problémy

- absence globální objektivní funkce – **lokální optimalizace** spojení shluků, ovšem s využitím blízkostí každých dvou objektů z různých shluků; shluky ale nejsou lokální optima např. (globální) optimalizační funkce K-means, ani při Wardově metodě (objekty shluku ani nemusí být nejblíže prototypu shluku)
- spojení shluků finální (nemění se): problém u vysokorozměrných dat se šumem, ale postprocessing např. přesun větví stromu hierarchie shluků pro optimalizaci nějaké globální objektivní funkce nebo preprocessing např. rozkladové shlukování (K-means) pro malé počáteční shluky místo jednotlivých objektů

Divizivní hierarchické shlukování

- více metod, např. i půlící (bisecting) K-means

Minimum spanning tree (Minimální kostra)

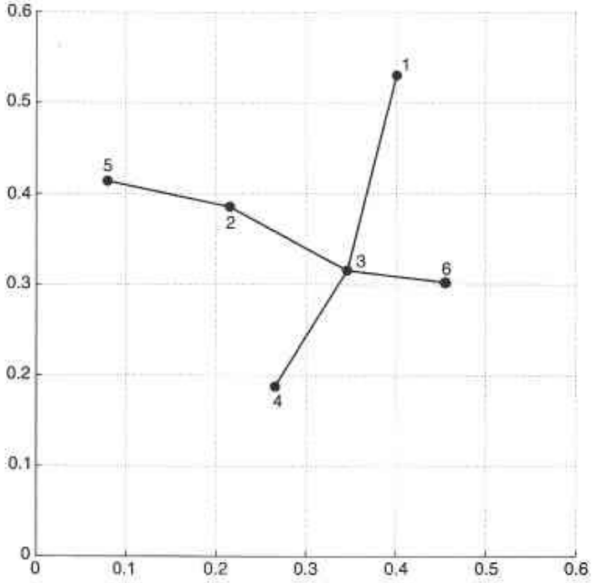
$=$ neprázdný souvislý podgraf ohodnoceného grafu se všemi uzly, bez cyklů (strom) a s minimálním celkovým ohodnocením hran

1. nalezení minimum spanning tree **grafu blízkostí (proximity graph)** $=$ hrany mezi objekty jako uzly ohodnocené blízkostí (nepodobností) objektů, např., $\sim$ shluk se všemi objekty
2. opakovaně zrušení hrany ohodnocené nejmenší blízkostí $\sim$ rozdělení shluku na dva, dokud nezůstanou jen singletony

$\sim$ opakované ponechávání pouze hran grafu blízkostí mezi nejbližšími objekty (rozklad grafu)

$\rightarrow$ stejné shlukování (kolekce shluků) jako single link/min aglomerativní hierarchické shlukování

Divizivní hierarchické shlukování



Hustotové (density-based) shlukování

= vyhledávání „hustších“ oblastí objektů (= shluků) oddělených málo hustými oblastmi ($\sim$ anomálie/šum)

$\rightarrow$ částečné rozkladové shlukování s hustotovými shluky – libovolných tvarů a velikostí, odolné vůči šumu

- více definic **hustoty**

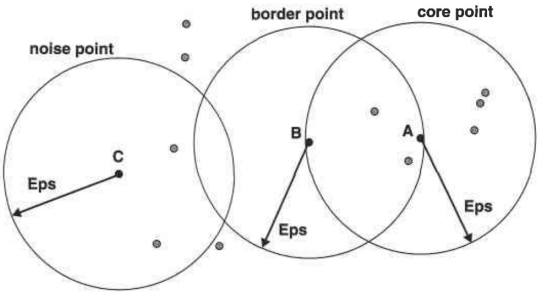
Centrová (center-based) hustota

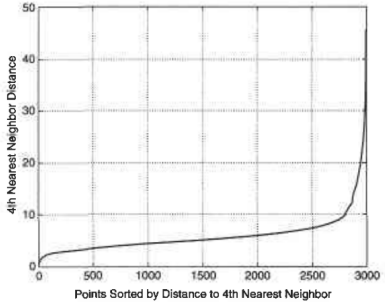
= pro daný objekt počet objektů do dané **vzdálenosti** (včetně daného objektu), např. – v extrémech rovna počtu všech objektů nebo 1

- core objekt (bod) = hustota nad **prahem** $\sim$ „uvnitř“ husté oblasti, např.
- border objekt = ne core, ale do dané vzdálenosti od (asociovaného) core objektu – možno více $\sim$ „na hranici“ husté oblasti, např.
- šumový objekt = ostatní, „mimo“ hustou oblast, např.

? jaká daná vzdálenost d a práh k hustoty: zlomová vzdálenost objektů k jejich k-tému nejbližšímu sousedu (pro objekty ve shluku je malá, pokud k není větší než počet objektů ve shluku, zlom pro ne příliš rozdílně husté shluky), např. – pro různá k se d příliš nemění, pro malá k i málo blízkých anomálií shluky, pro velká k malé shluky šum

Hustotové (density-based) shlukování





DBSCAN

- centrová hustota, pro původní algoritmus $k = 4$

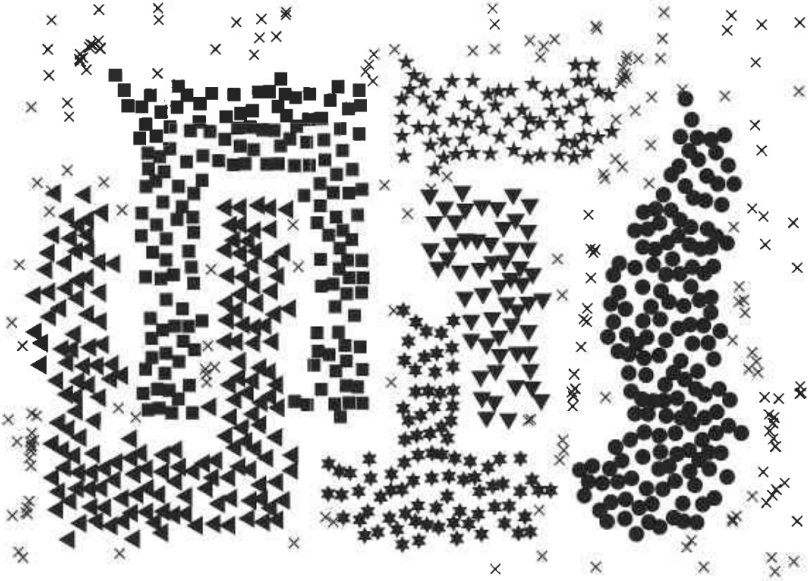
1. shluky = množiny core objektů do dané vzdálenosti d od sebe
2. zařazení border objektů do shluků s asociovanými core objekty – příp. vybraného

- např. $(k = 4, d = 10)$
- jednoduché, časová složitost kvadratická ve velikosti N vstupních dat (s méně atributy při výkonných datových strukturách pro vyhledávání objektů do dané vzdálenosti d od daného objektu, např. kd-stromech, $O(N \log N)$)

Problémy

- mnoho **podobných hustot shluků** – např. hustota méně hustých shluků podobná hustotě šumu kolem hustších shluků $\rightsquigarrow$ (pro dostatečně malou d pro oddělené méně husté shluky) hustší shluky spojeny se šumem kolem nich nebo (pro dostatečně velkou d pro oddělené hustší shluky) méně husté shluky šum
- vysokorozměrná data – náročnější definice hustoty, výpočet vzdáleností objektů

DBSCAN



Vyhodnocení shluků [SAMOSTUDIUM]

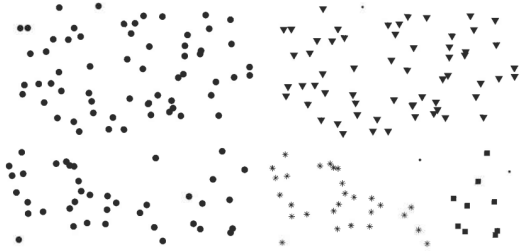
- v (supervised) klasifikaci vyhodnocení vytvořeného modelu dat součást jeho tvorby – ukazatele a vyhodnocení výkonnosti, řešení problému přeučení atd.
- ve shlukování ne – často jako část explorativní analýzy dat ($\Rightarrow$ vyhodnocení nadbytečné), různé typy shluků ($\Rightarrow$ různá vyhodnocení)

$\sim$ **validace shluků** – každá shlukovací metoda *nějaké* shluky v datech najde, i bez přirozených, např. (3 shluky z DBSCAN)

- *existence (tendence) shluků*? = nenáhodná struktura dat + počet shluků? + nenutnost externí informace = **unsupervised**: (interní) míry **koheze** (kompaktnosti, = blízkosti objektů ve shluku) a **separace** (izolace, = dobré oddělenosti) shluků, silhouette koeficient, kofenetický korelační koeficient – využití měr blízkosti objektů, např. objektivní funkce v K-means
- porovnání shluků s externí znalostí/strukturou dat, např. class labels objektů = **supervised**: (externí) míry např. entropie, purity, precision, recall, F – využití podílů class labels ve shlucích, podobnostní koeficienty – využití počtů párů objektů se stejnými/různými shluky a class labels
- porovnání shluků/shlukování mezi sebou = **relativní**: unsupervised i supervised

! problémy měr: použitelnost (např. jen pro dvou/třírozměrná prostorová data), interpretace (jaké hodnoty dobré? $\rightarrow$ statistické rozložení), složitost

Vyhodnocení shluků [SAMOSTUDIUM]



(a) Original points.

(b) Three clusters found by DBSCAN.

(c) Three clusters found by K-means.

(d) Three clusters found by complete link.

Koheze a separace shluků