



Rozhodovací stromy a jejich konstrukce z dat

Příklad „počítačová hra“.


Můžeme počítač naučit rozlišovat přátelské a nepřátelské roboty?



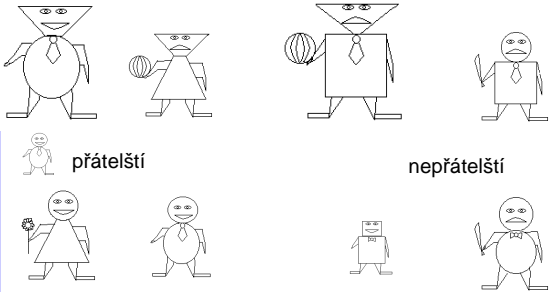
Učení s učitelem:
u některých už víme, jakou mají povahu (**klasifikace**)

Neparametrická úloha:
Nic nevíme o pravděpodobnosti distribuci jednotlivých objektů

2




Příklad „počítačová hra“ 1. Můžeme se naučit roboty rozlišit na základě krátké zkušenosti?



přátelští

nepřátelští

3

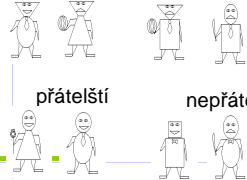


Reprezentace úlohy pomocí atributů


Můžeme navrhnout odpovídající klasifikační algoritmus ?

Klasifikace	Usmiva_se	kravata	tělo	hlava	v_ruce
přítel	ano	ano	kruh	3úhelník	nic
přítel	ne	ne	3úhelník	3úhelník	balon
nepřítel	ne	ano	čtverec	3úhelník	balón
nepřítel	ne	ano	čtverec	kruh	meč
přítel	ano	ne	3úhelník	kruh	květ
přítel	ano	ano	kruh	kruh	nic
nepřítel	ano	ne	čtverec	čtverec	nic
nepřítel	ne	ne	kruh	kruh	meč

Vyhledávání v tabulce, ...



4



Rozhodovací strom 1 pro danou množinu příkladů

Klasifikace	Usmívá_se	kravata	tělo	hlava	v_ruce
přítel	ano	ano	kruh	3úhelník	nic
přítel	ne	ne	3úhelník	3úhelník	balón
nepřítel	ne	ano	čtverec	3úhelník	balón
nepřítel	ne	ano	čtverec	kruh	meč
přítel	ano	ne	3úhelník	kruh	květ
přítel	ano	ano	kruh	kruh	nic
nepřítel	ano	ne	čtverec	čtverec	nic
nepřítel	ne	ne	kruh	kruh	meč

5

Nature Inspired Technologies Group

Rozhodovací strom 2 pro tutéž množinu příkladů

Klasifikace	Usmívá_se	kravata	tělo	hlava	v_ruce
přítel	ano	ano	kruh	3úhelník	nic
přítel	ne	ne	3úhelník	3úhelník	balón
nepřítel	ne	ano	čtverec	3úhelník	balón
nepřítel	ne	ano	čtverec	kruh	meč
přítel	ano	ne	3úhelník	kruh	květ
přítel	ano	ano	kruh	kruh	nic
nepřítel	ano	ne	čtverec	čtverec	nic
nepřítel	ne	ne	kruh	kruh	meč

6

Nature Inspired Technologies Group

Rozhodovací strom 2 pro tutéž množinu příkladů

Který strom je lepší a jak jej najdeme?

Klasifikace	Usmívá_se	kravata	tělo	hlava	v_ruce
přítel	ano	ano	kruh	3úhelník	nic
přítel	ne	ne	3úhelník	3úhelník	balón
nepřítel	ne	ano	čtverec	3úhelník	balón
nepřítel	ne	ano	čtverec	kruh	meč
přítel	ano	ne	3úhelník	kruh	květ
přítel	ano	ano	kruh	kruh	nic
nepřítel	ano	ne	čtverec	čtverec	nic
nepřítel	ne	ne	kruh	kruh	meč

7

Nature Inspired Technologies Group

Klasifikace	úsměv	kravata	tělo	hlava	v_ruce
přítel	ano	ano	kruh	3úhelník	nic
přítel	ne	ne	3úhelník	3úhelník	balón
nepřítel	ne	ano	čtverec	3úhelník	balón
nepřítel	ne	ano	čtverec	kruh	meč
přítel	ano	ne	3úhelník	kruh	květ
přítel	ano	ano	kruh	kruh	nic
nepřítel	ano	ne	čtverec	čtverec	nic
nepřítel	ne	ne	kruh	kruh	meč

Souhrn významu atributů	úsměv	Kravata	tělo=3úh.	hlava=3úh	v_r.=nic
Ano:3P,1N	Ano:2P,2N	Ano:2P,0N	Ano:2P,1N	Ano:2P,1N	Ano:2P,1N
Ne:1P,3N	Ne:2P,2N	Ne:2P,4N	Ne:2P,3N	Ne:2P,3N	Ne:2P,3N

8

Nature Inspired Technologies Group

Indukce rozhodovacího stromu z trénovací množiny

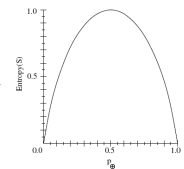
dáno: S ... trénovací množina (množina klasifikovaných příkladů)

1. Nalezni "nejlepší" atribut at_i (t.j. atribut, jehož hodnoty nejlépe diskriminují mezi pozitivní a neg. příklady) pro S a tím odhodnot kořen vytvářeného stromu.
2. Rozděl množinu S na podmnožiny S_1, S_2, \dots, S_n podle hodnot atributu at_i , a pro každou množinu příkladů S_i vytvoř nový uzel jako následníka právě zpracovávaného uzlu (kořenu)
3. Pro každý nově vzniklý uzel s přiřazenou podmnožinou S_i proved':
 - Jestliže všechny příklady v S_i mají tutéž klasifikaci (všechny jsou pozitivní nebo všechny jsou negativní),
 - pak uzel ohodnocený S_i je prohlášen za list vytvářeného rozhodovacího stromu (a tedy se už dále nevětví),
 - jinak jdi na bod 1 s tím, že $S := S_i$.

9

Entropie množiny S vzhledem k dané klasifikaci

- ❖ Posuzuje „různorodost“ klasifikace prvků z množiny S
- ❖ Nechť klasifikaci představuje atribut y , který má jen 2 hodnoty $\{0,1\}$. Pak označme $S^0 = \{z \in S : z_y = 0\}$ a $S^1 = \{z \in S : z_y = 1\}$



$$Entropy(S) = E(S) = - |S^0|/|S| * \log_2 |S^0|/|S| - |S^1|/|S| * \log_2 |S^1|/|S|,$$

kde $|A|$ označuje mohutnost množiny A

- ♦ Je-li $S^0 = \emptyset$, pak $Entropy(S) = 0$...
- ♦ Je-li $|S^0| = |S^1|$, pak $Entropy(S) = 1$
- ♦ Je-li $|S| = 1$, pak $Entropy(S) = 0$

10

Volba nejlepšího atributu pro množinu příkladů S vzhledem k dané klasifikaci

Kriterium minimální entropie rozkladu (KMER)

- ❖ Nechť at je pevně zvolený atribut, který může nabývat hodnot v_1 až v_n .
- ❖ Označme $S_j = \{z \in S : z_{at} = v_j\}$ podmnožinu S , která obsahuje právě ty objekty, které v atributu at mají hodnotu v_j .
- ❖ **Vážená entropie $E(S, at)$ rozkladu S podle hodnot atributu at** charakterizuje „čistotu“ klasifikace v jednotlivých složkách rozkladu S a je definována $E(S, at) = \sum_{i=1}^n |S_i|/|S| * E(S_i)$

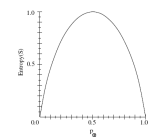
KMER vypočte $E(S, at)$ pro všechny atributy at a jako nejlepší atribut at^0 zvolí ten z nich, pro který je hodnota $E(S, at^0)$ nejmenší

11

Základní algoritmus ID3

- ❖ Realizuje prohledávání prostoru všech stromů, které lze zkonstruovat v jazyku trénovacích dat :

- ♦ shora dolů
- ♦ s použitím hladové strategie
- ❖ Volba atributu pro větvení na základě charakterizace „(ne)homogenity“ nově vzniklého pokrytí (používají se různé míry), např.:



- ♦ **Kriterium minimální entropie rozkladu**
- ♦ **Informační zisk (gain)** odhaduje předpokládané snížení entropie pro pokrytí vzniklé použitím hodnot odpovídajícího atributu

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

12

Nebezpečí použití kriteria KMER

Co se stane, pokud některý atribut má hodně „skoro“ unikátních hodnot, které takřka jednoznačně charakterizují každý trénovací příklad? Například pro rodné číslo je $|S_i| = 1$ a tedy

$$E(S) = 0 \text{ a } E(S, \text{rodne_cislo}) = 0$$

Tento argument je tedy kriteriem KMER vybrán jako nejlepší !

Je takový atribut opravdu užitečný pro testovací data?

13

Nebezpečí použití kriteria KMER

Co se stane, pokud některý atribut má hodně „skoro“ unikátních hodnot, které takřka jednoznačně charakterizují každý trénovací příklad? Například pro rodné číslo je $|S_i| = 1$ a tedy

$$E(S) = 0 \text{ a } E(S, \text{rodne_cislo}) = 0$$

Tento argument je tedy kriteriem KMER vybrán jako nejlepší !

Je takový atribut opravdu užitečný pro testovací data?

Ne, má malou generalizační schopnost!

Nebylo by vhodné takovou situaci nějak „penalizovat“? JAK? Zde (pro $KMER = 0$) nepomůže multiplikační koeficient! Raději využijeme hodnotu doplňkovou, kterým je **kriterium zisku**:

$$Gain E(S,at) = E(S) - E(S,at) = E(S) - \sum_{i=1}^n |S_i|/|S| * E(S_i)$$

14

Jak charakterizovat rozklad množiny S na c disjunktních podmnožin S_i podle všech hodnot uvažovaného atributu A ?

$$SplitInformation(S, A) \equiv - \sum_{i=1}^c \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}$$

SplitInformation odpovídá entropii rozdělení S podle všech hodnot atributu A . Např. je-li $|S_i|/|S| = 1$ pro všechna $i < (c + 1)$, je jeho hodnota rovna $(\log_2 c)$. Používá se pro výpočet *GainRatio*, který penalizuje atributy s příliš mnoha hodnotami:

$$GainRatio(S, A) \equiv \frac{Gain(S, A)}{SplitInformation(S, A)}$$

15

Volba atributů a další „speciální situace“

❖ Reálné hodnoty.

→ používá se **diskretizace**

❖ Různé ceny pro získání hodnoty atributu

16

Volba atributů a „speciální situace“-1

- ❖ Reálné hodnoty → používá se **diskretizace**
- ❖ **JAK SE VOLÍ VHDNÉ MEZNÍ HODNOTY?**

Vhodné řešení (Fayyad 91): Uspořádejte příklady podle velikosti zpracovávaného atributu a zvolte jako kandidátní mezní hodnoty ty, které leží v intervalu, kde se mění klasifikace. Hodnota, která maximalizuje **Gain**, je nutně jednou z nich.

Temperature:	40	48	60	72	80	90
PlayTennis:	No	No	Yes	Yes	Yes	No

17

Volba atributů a „speciální situace“ 2

- ❖ Různé ceny pro získání hodnoty atributu.
- ❖ Určíme-li cenu $Cost(A)$ v intervalu $<0,1>$, pak použijeme změněné kritérium, např.

- Tan and Schlimmer (1990)

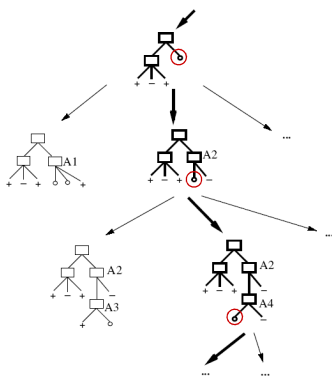
$$\frac{Gain^2(S, A)}{Cost(A)}$$

- Nunez (1988)

$$\frac{2^{Gain(S,A)} - 1}{(Cost(A) + 1)^w}$$

18

Postup prohledávání



19

Vlastnosti ID3: důsledky postupu prohledávání

- ❖ Pro klasifikační úlohu s diskretními atributy je prohledávaný **prostor hypotéz úplný** (tj. je schopný reprezentovat libovolnou možnou cílovou funkci) --> **existuje mnoho hypotéz konzistentních s daty!**
- ❖ Aktuální **množina hypotéz je vždy jednoprvková** (hlavová volba následníka), nelze jej tedy použít pro odpověď na dotaz „kolik je alternativních stromů konzistentních s daty?“
- ❖ Nepoužívá zpětný chod --> **možnost uvíznutí v lokálním optimu**
- ❖ **Rozhoduje se na základě všech příkladů** (nikoliv inkrementálně) --> metoda není příliš ovlivněna šumem

20

Kdy je vhodné použít algoritmy pro konstrukci rozhodovacího stromu?

- ❖ Cílová funkce má diskrétní hodnoty (jedná se o **klasifikační problém**)
- ❖ Instance trénovacích dat mají jednotný formát popisující hodnoty atributů
- ❖ Trénovací data mohou
 - ♦ být zašuměná
 - ♦ obsahovat chybějící hodnoty
- ❖ Je potřeba reprezentovat disjunkci podmínek (pravidla)

21

Proč dáváme přednost jednoduchým hypotézám?

Argument : Jednoduchých hypotéz je výrazně méně než složitých. Proto, pokud některé z jednoduchých h . data odpovídají, pak asi nejde o „náhodný jev“

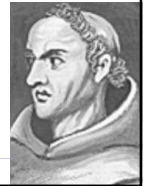
Occamova břitva :

Nejlepší hypotéza je ta nejjednodušší, která odpovídá datům.

Související problémy:

- proč zrovna **tato** malá množina?
- pozor na použitý jazyk!

William of Ockham, born in the village of Ockham in Surrey (England) about 1285, was the most influential philosopher of the 14th century and a controversial theologian.



22 witten & eibe

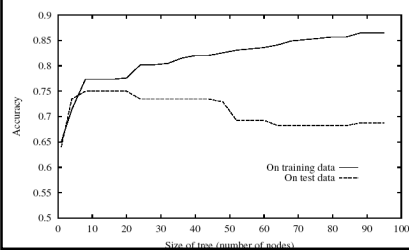
Otázky související s ID3

- ❖ Jak velké stromy konstruovat? Až do pokrytí všech příkladů? Co s přeučněním?
- ❖ Spojitý definiční obor atributů
- ❖ Metody volby nejvhodnějšího atributu
- ❖ Atributy o různých cenách
- ❖ Chybějící hodnoty
- ❖ ... ???

23

Přeučení


- ❖ Necht' H je prostor hypotéz. Hypotéza $h \in H$ je přeučena, pokud existuje jiná hypotéza $h_1 \in H$ taková, že chyba h na trénovacích datech je menší než chyba h_1 , avšak na celém prostoru instancí uvažovaných objektů je chyba h_1 menší než chyba h

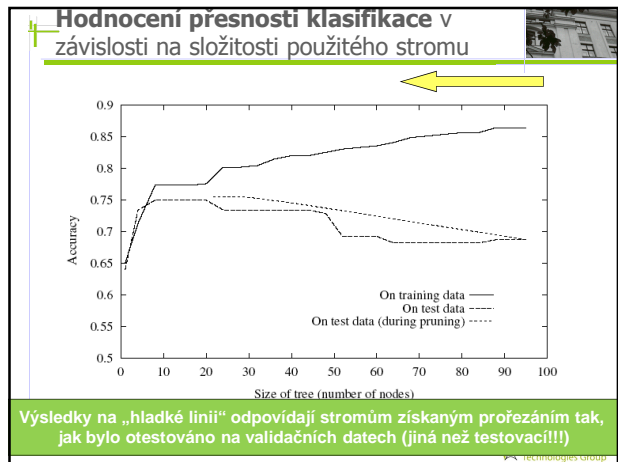


Často pozorovaná vlastnost zkonstruovaných stromů

Jak se vyhnout přeučení?

- ❖ Jak zvolit správnou velikost stromu?
- ❖ **Jak takový strom získat?**
 1. Zastavit růst stromu dřív než jsou vyčerpána všechna trénovací data
 2. **Prořezávání hotového stromu** – ukazuje se jako zvlášť užitečné! Volba vhodného prořezání pomocí **validační množiny dat** (vybraná nezávisle, tedy bez náhodných vlivů případně přítomných v trénovacích datech).
- ❖ **Algoritmus prořezávání „redukce chyby“:**
 - ❖ Vyberte uzel, odstraňte podstrom, v něm začínající a přiřaďte většinou klasifikaci.
 - ❖ Pokud se chyba na validačních datech zmenšila proveďte uvedené profíznutí (ze všech možností vyberte tu s největším zlepšením).

25 




Rozhodovací strom jako logický výraz

Klasifikace	Usmívá_se	kravata	tělo	hlava	v_ruce
přítel	ano	ano	kruh	3úhelník	nic
přítel	ne	ne	3úhelník	3úhelník	balón
nepřítel	ne	ano	čtverec	3úhelník	balón
nepřítel	ne	ano	čtverec	kruh	meč
přítel	ano	ne	3úhelník	kruh	květ
přítel	ano	ano	kruh	kruh	nic
nepřítel	ano	ne	čtverec	čtverec	nic
nepřítel	ne	ne	kruh	kruh	meč

```

graph TD
    A[kravata] -- ano --> B[Usmívá_se]
    A -- ne --> C[tělo]
    B -- ano --> D[přítel]
    B -- ne --> E[nepřítel]
    C -- jiné --> F[nepřítel]
    C -- 3úh. --> G[přítel]
  
```

(Kravata=ano & usmívá_se=ano) V (Kravata=ne & tělo=3úh.) -> přítel


27 

Závěrečné prořezávání pravidel (rule post-pruning) použité v C4.5

1. Vytvořte přeučený rozhodovací strom
2. Zapište výsledný strom ve tvaru disjunkce pravidel (každá větev = jedno pravidlo)
3. Každé jednotlivé pravidlo co nejvíc prořežte (odstraní se postupně ty podmínky, které nezhorší jeho klasifikační přesnost)
4. Uspořádejte výsledná pravidla podle jejich odhadnuté přesnosti a dále je používejte jako rozhodovací seznam

Odhad přesnosti pravidla

- ❖ na validační množině (= relativní počet správných závěrů)
- ❖ na trénovacích datech (= „pesimistický odhad počtu správných závěrů za předpokladu binomického rozdělení“)

28 

Metoda n nejbližších sousedů

○ Prvky 1. třídy
 ☆ Prvky 2. třídy
 △ Neznámý prvek

○ Prvky 1. třídy
 ☆ Prvky 2. třídy
 △ Neznámý prvek

Obrázek 1: Popis klasifikace 1-NN Obrázek 2: Popis klasifikace 3-NN

- ❖ Pro nový objekt je vypočtena vzdálenost od všech objektů v trén. příkl.
- ❖ Je nalezeno všech n trén. příkladů (= množina \mathcal{T}), které jsou k novému objektu nejbliž. Nový objekt získá klasifikaci, která je v \mathcal{T} nejčastější.
- ❖ Možné zobecnění: hledá se nejlepší vážící koeficient pro jednotlivé atributy.

29

Příklad: Létání na simulátoru F16

Úkol: sestavit řídicí systém pro ovládání leteckého simulátoru F16 tak, aby splnil předem definovaný plán letu daný takto:

1. vzlet a výstup do výšky 2000 stop
2. let v dané výšce směrem N do vzdálenosti 32000 stop od místa startu
3. zahnout vpravo v kurzu 330°
4. ve vzdálenosti 42000 stop od místa startu (ve směru S-N) provést obrát vlevo a zamířit zpět do místa startu (obrat je ukončen při kurzu mezi 140° a 180°)
5. vyrovnat směr letu s přistávací dráhou, tolerance 5° pro kurz a 10° pro výchylku křídél oproti horizontu
6. klesat směrem k počátku přistávací dráhy
7. přistát

Trenovací data: 3x30 letů (od 3 pilotů). Každý let popsán pomocí 1000 záznamů (poloha a stav letounu, pilotem provedený řídicí zásah)

30

Záznam: Poloha a stav

on_ground	boolean: je letadlo na zemi?
g_limit	boolean: je překročen g limit letadla?
wing_stall	boolean: je letadlo stabilní?
twist	integer: 0°-360°, výchylka křídél vůči obzoru
elevation	integer: 0°-360°, výchylka trupu vůči obzoru
azimuth	integer: 0°-360°, směr letu
roll_speed	integer: 0°-360°, rychlost změny výchylky křídél [°/s]
elev_speed	integer: 0°-360°, rychlost změny výchylky trupu [°/s]
azimuth_speed	integer: 0°-360°, rychlost změny kurzu [°/s]
airspeed	integer: rychlost letadla v uzlech
climbspeed	integer: rychlost změny výšky [stop/s]

31

Záznam: Poloha a stav + Řízení

E/W distance	real: vzdálenost ve směru východ-západ od místa startu
N/S distance	real: vzdálenost ve směru sever-jih od místa startu
fuel	integer: váha paliva v librách

Řízení:

rollers	real: nastavení ovladače horizontálního vychýlení
elevator	real: nastavení ovladače vertikálního vychýlení
thrust	integer: 0-100%, plyn
flaps	integer: 0°, 10° nebo 20°, nastavení křídlových lopatek

Každá ze 7 fází letu vyžaduje vlastní typ řízení (jiné zásahy pilota):
 trénovací příklady rozděleny do 7 odpovídajících skupin. V každé skupině je zkonstruován zvlášť rozhodovací strom pro každý typ řídicího zásahu (rollers, elevator, thrust, flaps), t.j. 7 x 4 stromů

32

Zkuste navrhnout nejjednodušší klasifikační algoritmus



přátelští

nepřátelští

