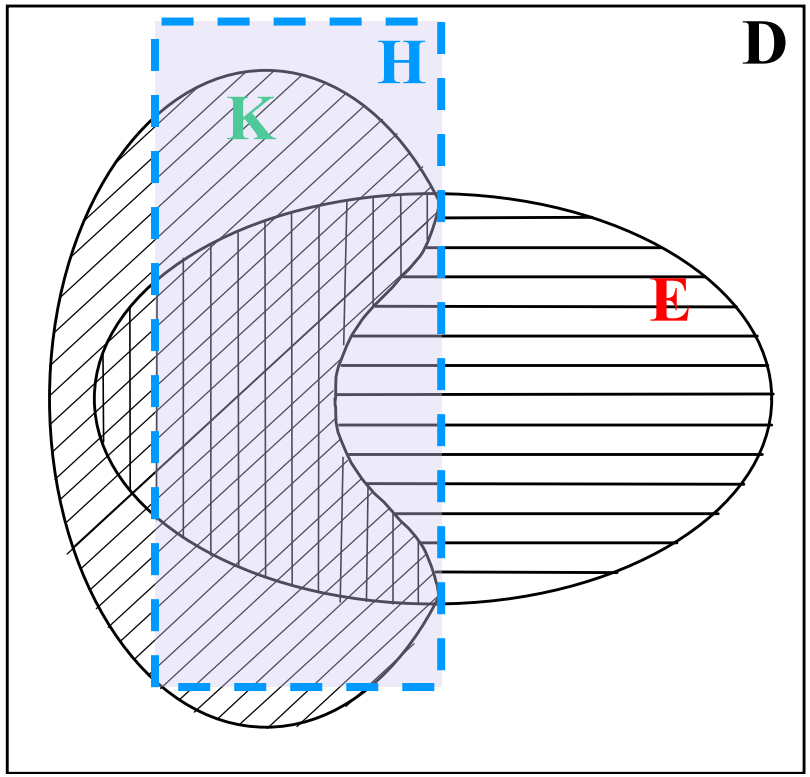
Výpočetní teorie strojového učení
a
pravděpodobně skoro správné
(PAC) učení

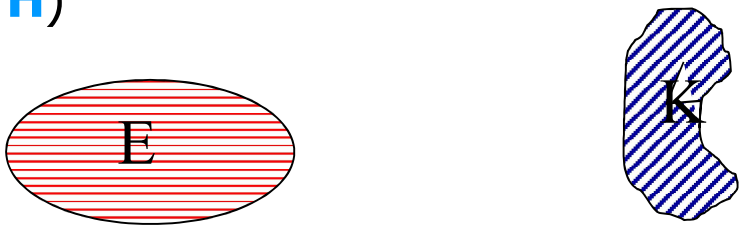
Koncept a hypotéza

- **Koncept** = reálně existující třída objektů vymezená na známém definičním oboru (např. 3D vektory reálných čísel). Objekty klasifikujeme podle toho, zda ke konceptu patří nebo nepatří. Obvykle známe jen *konečnou množinu příkladů* prvků daného konceptu (*pozitivní příklady*), případně i množinu prvků, které ke konceptu nepatří (*negativní příklady*), které dohromady tvoří konečnou *trénovací množinu konceptu*.
- **Trénovací množina** konceptu *nenabízí definici konceptu, ale popisuje koncept implicitně* (t.j. pomocí příkladů).
- **Hypotéza** = pokus o formální charakterizaci (*explicitní popis*) konceptu, např. pomocí formule ve výrokové logice (pravidla) nebo množina aritmetických výrazů, která má být splněna.

Cíl induktivního strojového učení



Na základě omezeného vzorku příkladů **E** $= \mathbf{E}^+ \cup \mathbf{E}^-$, charakterizovat (popsat) zamýšlenou skupinu objektů (koncept **K**) tak, aby **navržený popis** (hypotéza **H**)



- pro prvky z **E** co nejlépe odpovídal právě jen prvkům reprezentujícím koncept, tj. prvkům z $\mathbf{E}^+$
- a byl použitelný pro určení příslušnosti ke konceptu i pro objekty mimo **E**

Které hypotézy hledáme?

Hypotéza je pro trénovací data **E**

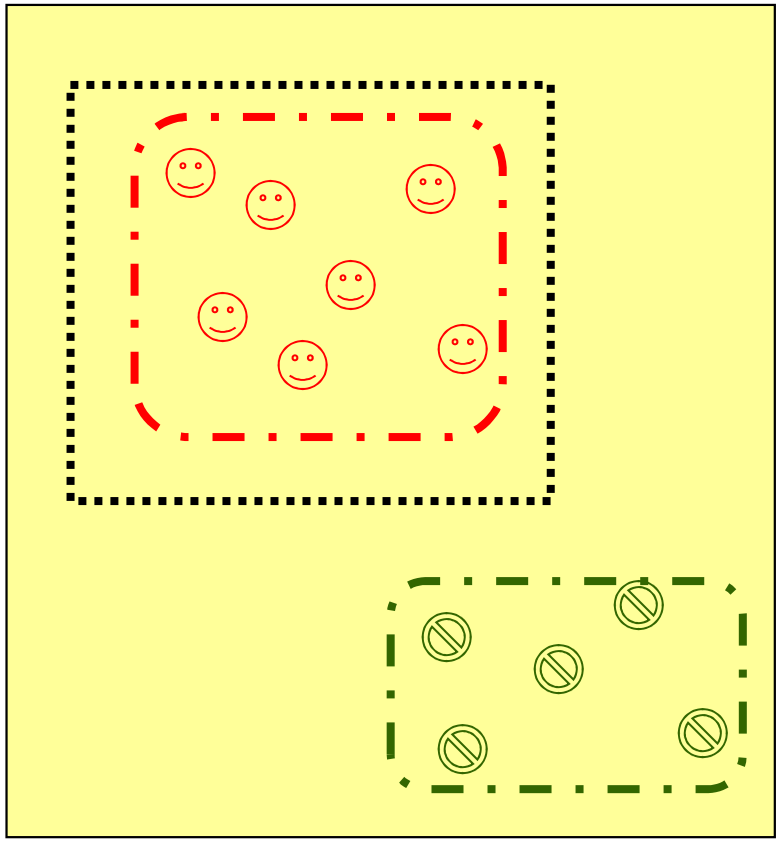
- *korektní*, když nepokrývá žádný negativní příklad ($\mathbf{E}^-$).
- *úplná*, když pokrývá všechny pozitivní příklady ($\mathbf{E}^+$).

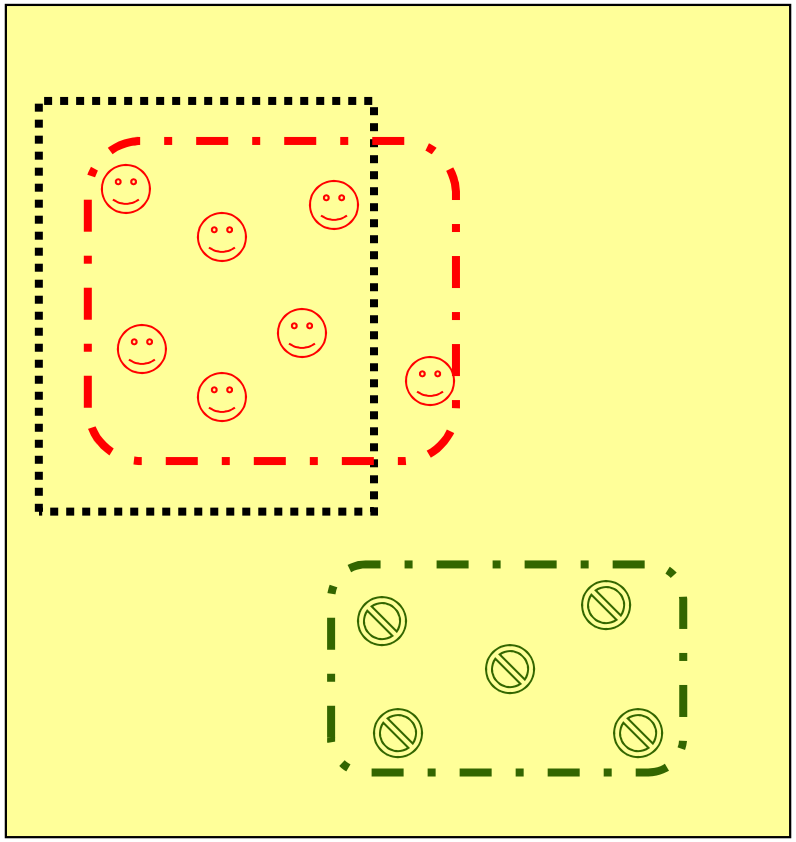
Hledáme **konsistentní (= korektní + úplnou) hypotézu** pro daná trénovací data.

Trénovací data by měla mít stejnou distribuci jako všechna data o uvažované úloze. Tento požadavek má být zárukou toho, že hypotéza se bude chovat nad daty, která nebyla v trénovací množině, podobně jako nad trénovací množinou, tj. bude „většinou správně“ přiřazovat klasifikaci .

Korektní hypotéza pro **Trénovací množinu**

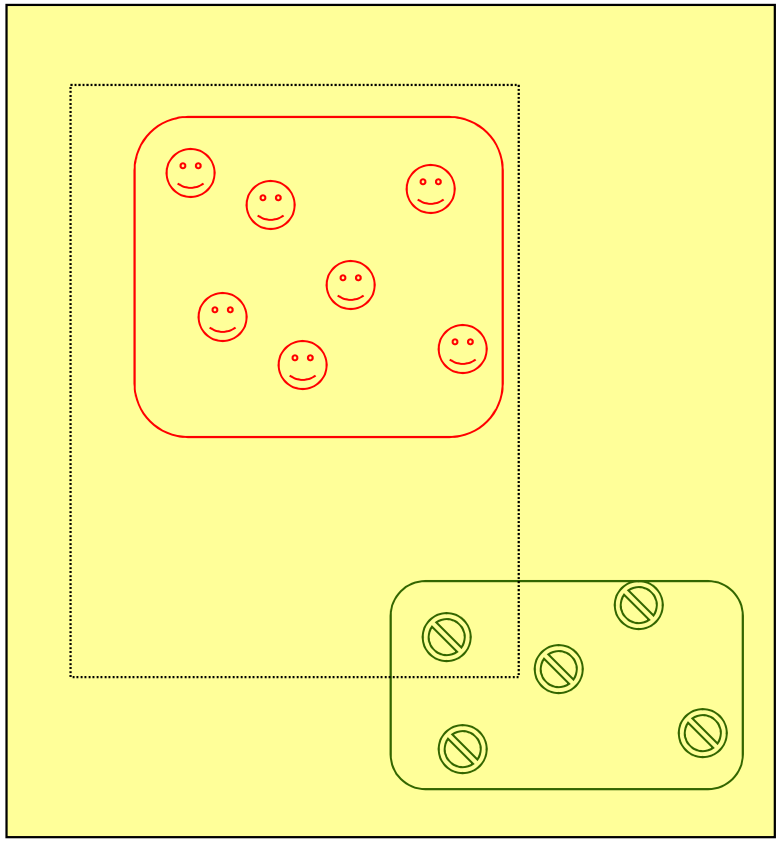
úplná (complete) **neúplná (incomplete)**



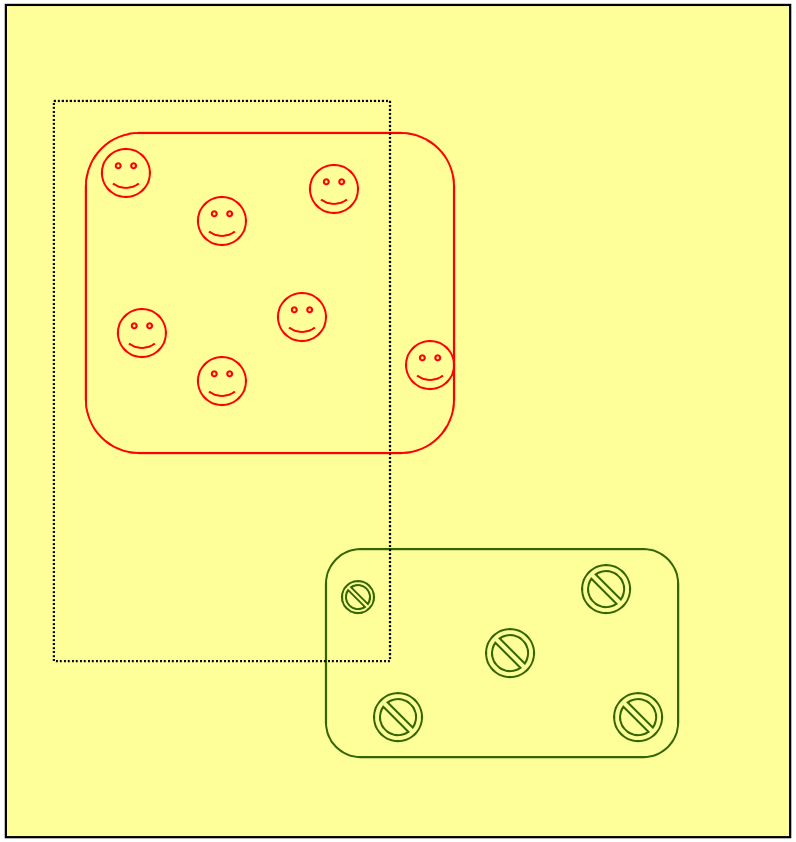


Nekorektní hypotéza

úplná (complete)



neúplná (incomplete)



Výpočetní teorie strojového učení

Věta o ošklivém kačátku. Nechť E je klasifikovaná trénovací množina pro koncept K, který tvoří podmnožinu *konečného* definičního oboru $\mathbf{D}$ všech myslitelných instancí objektů, které lze popsat v uvažovaném jazyce pro popis trénovacích dat.

Pokud $E \subseteq \mathbf{D}$ a $E \neq \mathbf{D}$, pak pro každý prvek $g \in \mathbf{D} \setminus E$ platí, že násl. hodnoty

- počet korektních a úplných hypotéz pro E, pro které „g patří ke konceptu K“
- počet korektních a úplných hypotéz pro E, pro které platí opak, tedy „g nepatří ke konceptu K“

jsou úplně stejné. Tedy pro libovolný objekt z množiny $\mathbf{D} \setminus E$ si nemůžeme být jisti, zda do konceptu K patří (pravděpodobnost, že do konceptu patří je totožná s pravděpodobností, že do něj nepatří).

Situace se změní, jakmile jako hypotéza nemůže sloužit libovolná podmnožina definičního oboru! Nechť například pro objekty v rovině uvažujeme jen hypotézy odpovídající konvexním tvarům. Pak existuje řada objektů z $\mathbf{D} \setminus E$ pro které si **můžeme být zcela jisti jejich klasifikaci**!

Výpočetní teorie strojového učení

Přijetí nějakých omezení na typ hledaných hypotéz tedy zvyšuje šanci na to, že budeme schopni správně rozhodovat o neznámých objektech.

V takovém případě je přirozené, že přestaneme striktně trvat na tom, že hledáme **korektní a úplne hypotézy!**

Cílem strojového učení pak už není hledání *přesně správné hypotézy*, ale hledání **skoro správné hypotézy** (approximately correct), která splňuje rozumně zvolené *doplňkové požadavky,* např. je dostatečně jednoduchá, má předem stanovený tvar (bias) , …

Hledáme vhodný kompromis mezi paměťovými nároky pro reprezentaci konceptu a mezi pravděpodobností chybné klasifikace.

Východiska výpočetní teorie stroj.učení PAC

Výchozí předpoklad - **stacionarita**: Trénovací i testovací množina jsou vybírány z téže populace za použití totožné distribuce pravděpodobnosti.

(Probably Aproximately Correct) **PAC učení**: Kolik trénovacích příkladů je třeba, aby se podařilo eliminovat všechny velmi špatné hypotézy?

D definiční obor s distribucí **δ**

$\mathbf{X} \subseteq \mathbf{D}$ nějaká množina příkladů (s distribucí **δ**)

f skutečný popis konceptu

H množina všech možných hypotéz, $\mathbf{h} \in \mathbf{H}$ je aktuální hypotéza

$\mathbf{er_X(h)}$ = pravděpodobnost jevu „ $\mathbf{x} \in \mathbf{X}$ a platí $\mathbf{f(x) \neq h}(x)$“

= P({**x**: $\mathbf{x} \in \mathbf{X}$ s distribucí **δ** a platí $\mathbf{f(x) \neq h(x)}$ })

tuto hodnotu studují „křivky učení“

er(h) **zkrácený zápis pro** $\mathbf{er_D(h)}$

Hypotéza **h** je **ε-skoro správná**, pokud $\mathbf{er(h)} < \varepsilon$

Základní otázka PAC

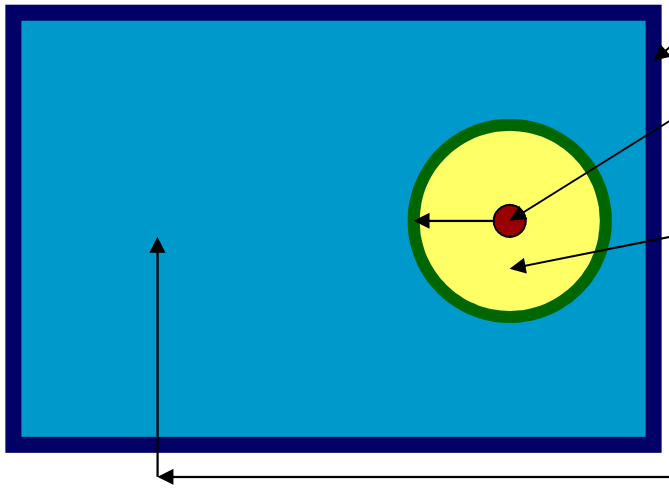
Buď **m** mohutnost trénovací množiny

Můžeme určit **m** *tak, aby pouhá konsistence hypotézy s trénovací množinou byla dostatečnou zárukou toho, že jsme našli skoro správnou hypotézu?*

Takový odhad by mohl sloužit např. jako vodítko při shromažďování či posuzování trénovacích dat

V dalším předpokládáme, že hledaný popis konceptu **f** lze najít v uvažované množině hypotéz **H** (tedy existuje korektní a úplná hypotéza pro daný koncept)

Základní pojmy PAC



H prostor všech hypotéz

f (hledaný) cílový koncept z mn. **H**

$\mathbf{H}_{\varepsilon}$ je **ε** okolí **f**, tj. množina obsahující pouze ty hypotézy, jejichž pravděpodobnost chyby je < **ε**

$\mathbf{H}_{\varepsilon\text{-}\textbf{špatná}} = \mathbf{H} - \mathbf{H}_{\varepsilon}$

obsahuje právě ty hypotézy, jejichž prst chyby je větší než ε

Pokusme se odhadnout, za jakých okolností platí, že
P(**hypotéza** konzistentní se všemi učicími příklady je z $\mathbf{H}_{\varepsilon\text{-}\textbf{špatná}}$) < δ

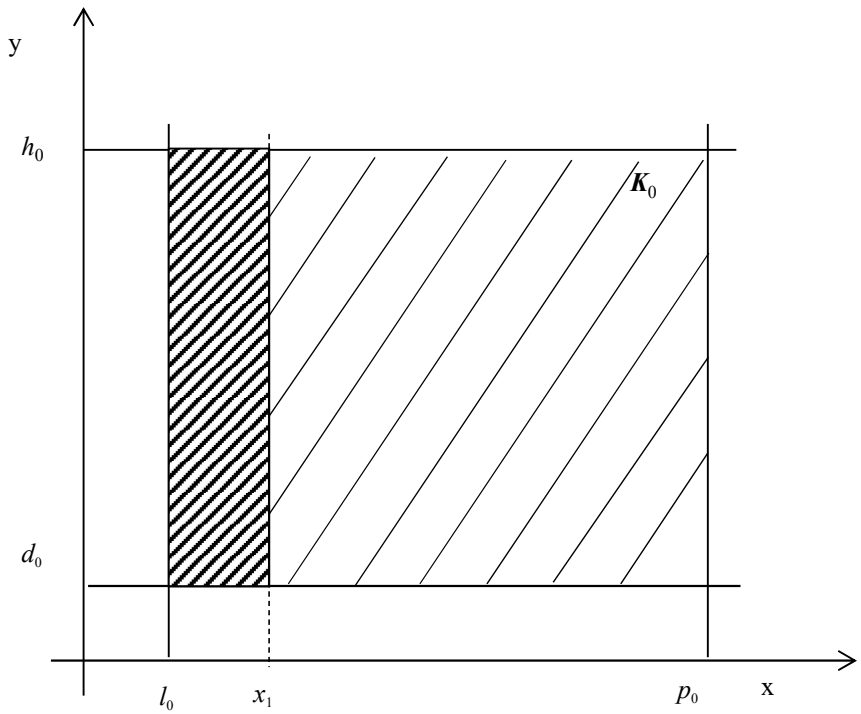
Algoritmus **M** učící se pojem **„středně velký objekt“**, jehož definice má tvar **Interval$_1$** X **Interval$_2$**

Pro všechny vstupující pozitivní příklady sledujte hodnoty **MAX$_i$** a **MIN$_i$** ve všech atributech ***i*** pro dosud přečtené vzorky. Hypotéza nechť je pak interval

$$\langle \mathbf{MIN_1}, \mathbf{MAX_1} \rangle \times \langle \mathbf{MIN_2}, \mathbf{MAX_2} \rangle$$

Tvrzení:

K tomu, aby se tento algoritmus naučil skoro přesně (s chybou menší než δ) hypotézu, která klasifikuje s chybou menší než **ε**, stačí, aby prohlédl **4/ε*ln(4/δ)** příkladů.



Důkaz stojí na pozorování**:**

Pro navržený algoritmus **M** platí „všechny generované hypotézy jsou konzistentní, tj. pro libovolnou navrženou hypotézu **h** platí, **h** ⊆ **k**“.

Tuto vlastnost budeme dále označovat jako ***Inkluze***.

Důsledek ***Inkluze:***

Pro libovolnou hypotézu **h** navrženou algoritmem **M** platí, že pravděpodobnost jevu „nový objekt bude splňovat **h** (tj. spadne do odpovídajícího intervalu)“ je menší než tato pravděpodobnost pro cílový koncept **k** = <***l,p***> **x** < ***d,n***>, ozn. **P(k).**

Jaká je pravděpodobnost toho, že nový objekt **x** bude špatně klasifikován?

Jistě platí, že **P(x: h(x) ≠ k(x)) < P(k).**

Neboť **P(x: h(x) ≠ k(x)) = P(x: ¬ h(x) & k(x))=**

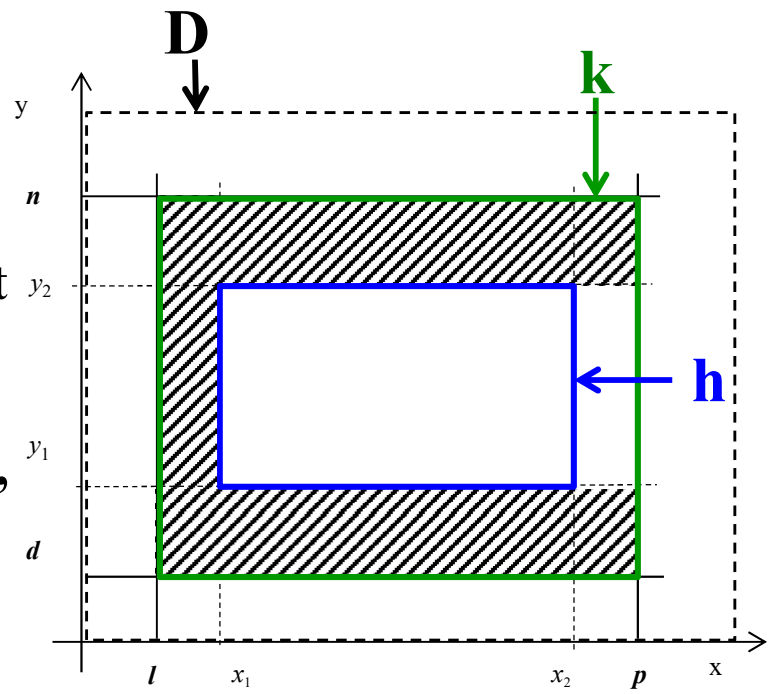
= P(x: k(x) & h(x) ≠ k(x)) < P(k).

- Nechť **P(k) < ε**, pak jistě i pravděpodobnost chyby je menší než ε.
- Nechť **P(k) > ε** . Zvolme x_0 tak, aby se vzorek nedostal do bezprostřední blízkosti ***l***, tj. padne mimo levý sloupec

  $\mathbf{x_0 = \inf\{x: P(< l,x > \times < d,n >) > \varepsilon/4\}}$

Jistě platí P(<*l*, x_0 > x <*d,n* >) > ε/4,

a tedy pravděpodobnost, že vzorek padne mimo levý sloupec je menší než **(1- ε/4)**



Odhad počtu trénovacích příkladů pro koncept vyjádřený intervalem (např. „středně velký objekt“)

Pro **m** různých vzorků platí:

Pravděpodobnost toho, že žádný z **m** vzorků se nedostane do bezprostřední blízkosti levého roku (každý vzorek padne mimo levý sloupec), je menší než

$$(1- \varepsilon /4)^m$$

Tentýž postup lze použít i pro ostatní rohy:

Pravděpodobnost jevu „žádný prvek z trénovací množiny obsahující **m** vzorků nepadnul do rohu “ je menší než

$$\mathbf{4\ (1- \varepsilon /4)^m < \delta}$$

$$4\ (1- \varepsilon /4)^m \cong 4\ e^{-m * \varepsilon /4} < \delta$$

$$\mathbf{m > 4/\varepsilon * \ln(4/ \delta)}$$

Odhad potřebného počtu příkladů – obecný případ *

Tvrzení: Nechť **h** je hypotéza konzistentní se všemi trénovacími příklady.

Pokud platí $P(\mathbf{h} \in \mathbf{H}_{\varepsilon\text{-špatná}}) < \delta$, pak pravděpodobnost toho, že „**h** je **ε-skoro správná**“ je větší než **(1- δ).**

Zdůvodnění:

Předpokládáme, že v **H** *existuje nějaká hypotéza konzistentní se všemi trénovacími příklady*. Jistě platí

$$P(\mathbf{h} \in H - \mathbf{H}_{\varepsilon\text{-špatná}}) + P(\mathbf{h} \in \mathbf{H}_{\varepsilon\text{-špatná}}) = 1$$

Víme, že $\mathbf{H}_{\varepsilon} = \mathbf{H} - \mathbf{H}_{\varepsilon\text{-špatná}}$, a proto

$$P(\mathbf{h} \in \mathbf{H}_{\varepsilon}) \geq (1 - \delta).$$

$\mathbf{h} \in \mathbf{H}_{\varepsilon}$ znamená, že **h** klasifikuje prvky z **X** s chybou menší než ε, tj. **h** je **ε-skoro správná**

Za jakých okolností můžeme zajistit, aby pro **h** konzistentní s trén. daty platilo $P(\mathbf{h} \in \mathbf{H}_{\varepsilon\text{-špatná}}) < \delta$? *

Nechť **b** je libovolná opravdu špatná hypotéza. Jde tedy o h., tž. **er(b)** = pravděpodobnost jevu „$\mathbf{x} \in \mathbf{X}$ a platí $\mathbf{f(x)} \neq \mathbf{b}(x)$“ $> \varepsilon$, tj. $\mathbf{b} \in \mathbf{H}_{\varepsilon\text{-špatná}}$. *Platí:*

- Pravděpodobnost, že **b** *správně* klasifikuje 1 zvolený příklad: P(**b** správně klasifikuje 1 zvolený příklad) $\leq (1-\varepsilon)$
- Pravděpodobnost, že **b** správně klasifikuje **m** zvol. příkladů: P(**b** správně klasifikuje **m** zvolených příkladů) $\leq (1-\varepsilon)^m$

Pravděpodobnost, že *existuje* prvek z $\mathbf{H}_{\varepsilon\text{-špatná}} = \{b_1,\ldots, b_k\}$, který správně klasifikuje **m** zvol. příkladů, je rovna pravděpod., že „$\mathbf{b_1}$ správně klasifikuje **m** zvolených příkladů“ nebo …nebo „$\mathbf{b_k}$ správně klasifikuje **m** zvolených příkladů“, tj.

$$P(\mathbf{h} \in \mathbf{H}_{\varepsilon\text{-špatná}}) \leq \sum_{i \leq k}(1-\varepsilon)^m = |\mathbf{H}_{\varepsilon\text{-špatná}}|*(1-\varepsilon)^m \leq |\mathbf{H}|*(1-\varepsilon)^m$$

Pokud $|\mathbf{H}|*(1-\varepsilon)^m < \delta$, pak jistě $P(\mathbf{h} \in \mathbf{H}_{\varepsilon\text{-špatná}}) < \delta$

Za jakých okolností můžeme zajistit, aby $P(\mathbf{h} \in \mathbf{H}_{\varepsilon\text{-špatná}}) < \delta$? *

Stačí, aby platilo $|\mathbf{H}|*(1-\varepsilon)^m < \delta$.

Pro $|\varepsilon| < 1$, platí, že $(1-\varepsilon)^m \cong e^{-\varepsilon * m}$

Podmínku lze tedy přepsat do tvaru

$$\ln(|\mathbf{H}|* e^{-\varepsilon * m}) < \ln \delta \quad \textit{čili} \quad \ln|\mathbf{H}| - \varepsilon * m < \ln \delta$$

Postačující podmínka pro počet příkladů **m** je tedy

$$m \geq (\ln|\mathbf{H}| - \ln \delta)/\varepsilon = \mathbf{1}/\varepsilon * (\ln|\mathbf{H}| + \ln(1/\delta))$$

Máme-li k dispozici alespoň $\mathbf{1}/\varepsilon * (\ln|\mathbf{H}| + \ln(1/\delta))$ příkladů a učicí algoritmus navrhuje hypotézu **h**, která je se všemi příklady konzistentní, pak pravděpodobnost toho, že „chyba **h** je menší než ε (**h** je **ε-skoro správná)**“ je větší než $(1-\delta)$.

Proč dáváme přednost jednoduchým hypotézám?

Argument : Jednoduchých hypotéz je výrazně méně než složitých. Proto, pokud data odpovídají některé z jednoduchých h. , 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.

Praktické použití odhadu počtu příkladů
$\mathbf{m \geq = 1/\varepsilon *(\ln |H| + \ln (1/\delta))}$

Nechť $\mathbf{H_B(n)}$ je *množina všech bool. funkcí pro* **n** bool. atributů, tj. zobrazení z n-tic skládajících se z 0 a1 do { 0, 1}.

- ***Velikost definičního oboru***: $\mathbf{2^n}$
- Počet funkcí z množiny mohutnosti **a** do { 0, 1}: $\mathbf{2^a}$.

Tedy mohutnost $\mathbf{H_B(n)}$ je $\mathbf{2^{2^n}}$

Postačující podmínka pro počet příkladů $\mathbf{m_B(n)}$, které potřebujeme k tomu, abychom se skoro správně naučili koncept popsaný booleovskou funkcí o **n** atributech, je

$$m_B(n) \geq \mathbf{c}/\varepsilon *(\mathbf{2^n} + \lg (1/\delta)),$$

Tedy: máme-li se skoro správně naučit koncept popsaný **obecnou bool**. funkcí, pak potřebujeme více než $\mathbf{2^n}$ příkladů. Jinými slovy: **musíme znát celý definiční obor**! *Věta o ošklivém kačátku.*

Důsledek odhadu

Je-li H množina možných hypotéz, pak se lze skoro správně (s pravděpodobností větší než (1- δ)) naučit hypotézu, jejíž chyba je menší než $\boldsymbol{\varepsilon}$, pokud máme **m** trénovacích příkladů a platí $\mathbf{m} \geq \mathbf{1}/\ \boldsymbol{\varepsilon} * (\ln |\mathbf{H}| + \ln (1/\delta))$. **(i)**

Pozorování: **m** je funkcí **|H|**

Podaří-li se nám získat nějakou doplňkovou informaci (omezení na tvar přípustných hypotéz), která ***zhora*** omezuje rozsah **H**, pak vystačíme s menším počtem trénovacích příkladů !!! Zde hraje významnou roli **doménová znalost**.

Pokusme se

- provést odhad mohutnosti množiny hypotéz pro některé běžné typy hypotéz (rozhodovací stromy,…)
- a zjistit vliv tohoto odhadu na požadovaný počet trénovacích příkladů

Odhad mohutnosti množiny hypotéz pro **rozhodovací seznam** (lin. reprezentace stromu) *

L_n jazyk obsahující přesně **n** binárních atributů

Ω def. obor uvažovaného konceptu má tedy 2^n různých prvků

Rozhodovací seznam (decision list) v jazyce L_n je uspořádaný seznam $\mathfrak{R}=[\ t_1:c_1, \ldots ,t_m:c_m]$, kde

- t_i je test vyjádřený ve tvaru konjunkce literálů z L_n
- $c_i \in \{0,1\}$ je přiřazená klasifikace.

Nechť $o \in \Omega$, pak $\mathfrak{R}(o) = c_i$, kde t_i je první test, který objekt o splňuje (tj. $t_1(o)=0, \ldots, t_{i-1}(o)=0, t_i(o)=1$). Pokud $t_k(o)=0$ pro všechna $k \leq m$, pak $\mathfrak{R}(o) = 0$

Každý strom hloubky **n** *(délka nejdelší větve) nebo formuli v disjunktivní normální formě lze napsat jako rozhodovací seznam v jazyce* L_n*: např.*

(s_1 & s_3 & not s_3) v (not s_1 & s_3 & not s_6) *odpovídá*

[(s_1 & s_3 & not s_3):1, (not s_1 & s_3 & not s_6):1]

Odhad mohutnosti ***k*-DL(*n*)** *

Nechť ***k*-DL(*n*)** je množina všech rozhodovacích seznamů, jejichž testy mají přípustnou délku omezenou pevně zvoleným číslem $k < n$. *Jak ovlivní volba **k** mohutnost množiny hypotéz**?***

Odhad mohutnosti H, *je-li prostor hypotéz* 1-DL(*n*)

$|\mathbf{1\text{-}DL}(n)| <$ počet permutací z n (tedy $n!$) krát 5^n.

Z pevně zvolené permutace lze totiž vytvořit 5^n různých rozhodovacích seznamů (test je buď „*nezařazen*“ nebo je použit s nebo negace a s výsledkem **1**, **0**).

$$|1\text{-}\mathbf{DL}(n)| < n! * 5^n$$

Protože $\ln(\mathbf{n!}) < n * \ln n$, *platí*

$$\ln|\mathbf{1\text{-}DL}(n)| < \ln(n! * 5^n) < \mathbf{O}(n * \ln n) + n \ln 5$$

Skoro správného učení lze dosáhnout při počtu trénovacích příkladů

$\mathbf{m \geq = 1/\varepsilon} * (\ln |\mathbf{H}| + \ln(1/\delta))$, což je pro 1-**DL**(*n*) číslo srovnatelné s $(n * \lg n)$.
Tato je výrazně méně něž mohutnost 2^n celého uvažovaného definičního oboru

Odhad mohutnosti *k*-DL(*n*) *

Conj(n,k): počet různých konjunkcí *nejvýše* ***k*** literálů sestrojených z ***n*** atributů.

Conj0(n,j): počet všech konjunkcí *přesně* ***j*** literálů sestrojených z ***n*** atributů (pomocná veličina pro odhad ***Conj(n,k)***).

Postupujeme takto

$Conj0(n,j) < 2^{j} * n^{j} = (2n)^{j}$ člen vyjadřující „znaménko“ atomu

$Conj(n,k) < \sum_{i<k+1} Conj0(n, i)$

$$< \sum_{i<k+1} (2n)^{i} = 2n(2^{k-1} n^{k-1} - 1)/(2n - 1) \approx O(n^{k}) \qquad \text{(ii)}$$

Horní odhad pro počet prvků ***k*-DL(*n*):** Rozhodovací seznam je vlastně uspořádaná posloupnost neopakujících se prvků z ***Conj(n,k),*** z nichž každý je klasifikován jednou z hodnot {0,1, × }, kde „×“ chápeme tak, že daná konjunkce v rozhodovacím seznamu není. Zřejmě tedy

$$k\text{-DL}(n) | < 3^{|Conj(j,k)|} |Conj(j,k)|!$$

Odhady mohutnosti k-DL(n) a $|k$-DL(n)$|$ *

Víme, že $|k\text{-DL}(n)| < 3^{|\text{Conj}(j,k)|}\,|\text{Conj}(j,k)|!$ *Z toho plyne, že*

$$\ln |k\text{-DL}(n)| < |\text{Conj}(j,k)| \ln 3 + \ln (\text{Conj}(j,k))!$$

Použitím vztahu $\lg n! < n * \lg n$ *dostáváme*

$$\ln |k\text{-DL(n)}| < |\text{Conj}(j,k)| * (\ln 3 + \ln|\text{Conj}(j,k)|)$$
$$< O(n^k) * (\ln 3 + \ln O(n^k)) \approx O(n^k \ln (n^k))$$

Po dosazení do vzorce $m \geq 1/\varepsilon * (\ln |H| + \ln (1/\delta))$ *dostáváme odhad pro hypotézy ve tvaru rozhodovacího listu*

$$m_{k\text{-DL}}(n) \approxeq c/\varepsilon\ (O(n^k \ln n^k) + (1/\delta))$$

Pro rozhodovací stromy s omezenou hloubkou je odhad ještě poněkud nižší, protože pro mohutnost prostoru hypotéz platí

$$|k\text{-DT}(n)| \approxeq 3^{|\text{Conj}(j,k)|}$$

Odpovídající počet trén. příkladů je $m_{k\text{-DT}}(n) \approxeq c/\varepsilon\ (n^k + (1/\delta))$

Věta o PAC učení rozhodovacího stromu

Nechť objekty jsou charakterizovány pomocí ***n*** binárních atributů a nechť připouštíme jen hypotézy ve tvaru rozhodovacího stromu s maximální délkou větve ***k***. Dále nechť δ, ε jsou malá pevně zvolená kladná čísla blízká 0. Pokud algoritmus strojového učení vygeneruje hypotézu φ, která je konzistentní se všemi **m** příklady trénovací množiny a platí

$$m \geq m_{k\text{-DT}}(\boldsymbol{n}) \geq c\,(\,n^k + \ln(1/\delta))\,/\,\varepsilon$$

pak φ je ε-skoro správná hypotéza s pravděpodobností větší než $(1-\delta)$, t.j. chyba hypotézy φ na celém definičním oboru konceptu je menší než ε s pravděpodobností větší než $(1-\delta)$.

Některá zajímavá pozorování vyplývající z **PAC učení**

- **Je možné** naučit se koncept z omezené množiny trénovacích příkladů skoro správně i pokud možných hypotéz je nekonečně !
- Při učení se hypotéz z konečné množiny možných h. bývá počet potřebných příkladů funkcí $\mathbf{n^k}$, kde **n** je počet atributů použitých v popisu příkladů a **k** charakterizuje složitost hypotézy (např. pro rozhodovací strom je **k** jeho maximální hloubka) - ***lze dosáhnout snížení tohoto čísla?***
- ***Změna reprezentace trénovacích příkladů. Používané metody:***
  - Snížení počtu atributů prostřednictvím nových **odvozených atributů,** které jsou funkcí nějaké skupiny původních atributů (NN, statistické řešení – Support Vector Machine)
  - Eliminace zbytečného zavádění nových atributů (např. při převodu multirelační úlohy na atributovou)
  - zjišťování **relevance**