Otázka: Existuje podpokrytie pokrytia $\mathbf{S}$ obsahujúce najviac $k$ množín?

Je treba poznamenať, že veľkosť inštancie závisí od $|U| = u$ aj od $|\mathbf{S}| = n$. V skutočnosti môžeme popísať $(U, \mathbf{S})$ pomocou matice veľkosti $u \times n$, v ktorej ku každému prvku (riadok) je vyjadrená píslušnosť k množine v systéme $\mathbf{S}$. Hodnota $u.n$ môže byť považovaná za veľkosť inštancie $(U, \mathbf{S})$ a toto je parameter, ktorého sa týka otázka polynomiálneho algoritmu.

Predchádzajúci problém je zapísaný v existenčnom tvare. Je možné sformulovať optimalizačný problém v nasledujúcom tvare:
PODPOKRYTIE
Inštancia: Dvojica $(U, \mathbf{S})$ podľa vyššie uvedenej definície a kladné celé číslo $k \leq |\mathbf{S}|$
Otázka: Aká je veľkosť minimálneho podpokrytia pre $(U, \mathbf{S})$?

Je zrejmé, keď máme odpoveď na druhý problém čas behu je polynomiálny v $u.n$, tak je zodpovedaný aj prvý problém v polynomiálnom čase.

Poznámka: urobené na cvičeniach.

## 1.5 OCCAM algoritmy

Nech $\Omega \to H$ je reprezentácia booleovských funkcií. Nech $\|\omega\|$ je miera veľkosti reprezentácie definovanej pre každé $\omega \in \Omega$. Pre každé $r \geq 1$ definujeme

$$\Omega_r = \{\omega \in \Omega | \quad \|\omega\| = r\}$$

a nech $H_r$ označuje podmnožinu $H$ obsahujúcu tie hypotézy $h_\omega$, ktorých minimálna reprezentácia má veľkosť $r$.

Hovoríme, že také hypotézy majú veľkosť reprezentácie $r$.

Potom $H$ môže byť odstupňované pomocou veľkosti reprezentácie $H = \bigcup H_r$. Učiaci algoritmus $L$ pre $H$ má vstup - tréningovú vzorku pre nejakú cieľovú funkciu $t \in H$. Predpokladajme, že $t \in H_r$; inak povedané najmenšia reprezentácia $t$ má veľkosť $r$. Výstup z $L$ bude špecifikovaný reprezentáciou $\omega \in \Omega_q$. Potrebujeme uvažovať vzťah medzi $q$ a $r$. Na základe výsledkov predchádzajúceho odseku môže byť ťažké nájsť najmenšiu možnú hodnotu $q$, ale môžeme požadovať nájdenie nie najkratšej možnej reprezentáciu, ale dostatočne krátku. Táto idea je presnejšie vyjadrená v nasledujúcej definícii Blumera (1987).

**Definícia 1.5.1** *Hovoríme, že učiaci algoritmus $L$ pre $H$ je* ***Occam*** *vzhľadom na reprezentáciu $\Omega \to H$, ak*

- *$L$ je konzistentný*
- *k danej tréningovej vzorke $\overline{s}$ dĺžky $m$ pre cieľovú hypotézu $t \in H_r$ výstupná hypotéza $L(s) = h_\omega$ je taká, že $\|\omega\| \leq m^\alpha \cdot r^\beta$, kde $0 < \alpha < 1$ a $\beta \geq 1$ sú konštanty.*

Hranica pre $\|\omega\|$ hovorí, že výstup je komprimovaný vzhľadom na dĺžku tréningovej vzorky a rastie len polynomilne s veľkosťou minimálnej reprezentácie dĺžky cieľovej hypotézy. Podmienka $\alpha < 1$ znamená, že výstup je vlastne komprimovaný tvar vstupu; ak povolíme $\alpha = 1$, potom výstup by bol porovnateľný s veľkosťou tréningovej vzorky, ktorej bitová dĺžka je lineárna v $m$ a žiadna signifikantná komprimácia by nebola dosiahnutá. Nasledujúca veta ukazuje, že výstup krátkej reprezentácie v tomto zmysle, postačí pre PAC učenie.

Aby sme sformulovali túto vetu, vráťme sa k našej originálnej definícii učiaceho algoritmu s konceptovým a hypotézovým priestorom ako rôznymi. Tento rozdiel je tu použiteľný, pretože sme sa zaujímali o hypotézy v $H_r$ použitím plného zdroja $H$.

**Veta 2** *Nech $H$ je priestor booleovských funkcií s reprezentáciou $\Omega \to H$, nech $H = \bigcup H_r$ je odstupňovaný veľkosťou reprezentácie. Ak $L$ je Occam učiaci algoritmus vzhľadom na danú reprezentáciu, potom pre každé $r, L$ existuje PAC učiaci algoritmus pre $(H_r, H)$ so zložitosťou vzorky $m_L(H_r, \delta, \epsilon)$ polynomiálnou v $r$, $\delta^*$ a $\epsilon^{-1}$.*

**Dôkaz:** Predpokladajme, že sú dané $\delta, \epsilon, \mu$ a $t \in H_r$. Pre každé dané $m$ nech $L(m,t)$ označuje množinu hypotéz $h \in H$ takú, že $h$ je výstup $L(\overline{s})$ algoritmu $L$ pre nejakú tréningovú vzorku $\overline{s}$ dĺžky $m$ a cieľový koncept $t$. Inak povedané, $L(m,t)$ je efektívny hypotézový priestor pre $t$. Podľa 2. podmienky Occam algoritmu členmi $L(m,t)$ sú hypotézy $h_\omega$, pre ktoré $\omega$ má najviac $M = \lfloor m^\alpha \cdot r^\beta \rfloor$ bitov, a celkový počet takých $\omega$ je najviac $2^{M+1}$. Odtiaľ

$$|L(m,t)| \leq 2^{m^\alpha \cdot r^\beta + 1}.$$

Poznamenajme, že hranica závisí len od $r$ nie od $t$ samotnej; inak povedané, platí uniformne pre všetky $t \in H_r$. Teraz zopakujeme argument daný v predchádzajúcom odseku. Pravdepodobnosť, že ľubovoľná daná $\epsilon$-zlá hypotéza z $H$ súhlasí s $t$ na tréningovej vzorke dĺžky $m$ je $(1-\epsilon)^m$. Pretože $L$ je konzistentný, jeho výstupné hypotézy súhlasia s tréningovouou vzorkou a teda pravdepodobnosť, že výstupná hypotéza je $\epsilon$-zlá je najviac

$$|L(m,t)| \cdot (1-\epsilon)^m \leq 2^{m^\alpha \cdot r^\beta + 1}(1-\epsilon)^m.$$

Zostáva dokázať, že toto môže byť $< \delta$, ak vezmeme dostatočne veľké $m$ a že $m$ je polynóm v $r$, $\delta$ a $\epsilon^{-1}$. Použitím nerovnosti $(1-\epsilon)^m < e^{-\epsilon \cdot m}$ a preusporiadaním dostávame, že $\epsilon \cdot m \geq A \cdot m^\alpha + B$, kde $A = r^\beta \cdot \ln 2$ a $B = \ln(\frac{2}{\delta})$.
Pretože $\alpha > 1$, podmienka platí ak

$$m^{1-\alpha} \geq (A+B)/\epsilon, t.j. \quad m \geq m_0 = \left\lceil \left(\frac{A+B}{\epsilon}\right)^{1/(1-\alpha)} \right\rceil.$$

Inak povedané, výraz pre $m_0$ je horná hranica pre zložitosť vzorky. Zrejme $m_0$ je polynóm v $r$, pretože A je $O(r^3)$ a tak $m_0$ je $O(r^{\beta/(1-\alpha)})$; ďalej $m_0$ je tiež polynóm v $\delta^*$ a $\epsilon^{-1}$.
□

Zdôrazňujeme znovu podmienku $\alpha < 1$; je zrejmé, že podmienka $\epsilon \cdot m > A \cdot m^\alpha + B$ môže byť splnená, ak $\alpha = 1$.

Existuje bezprostredný dôsledok tejto vety, že ak čas behu Occam algoritmu je polynomiálny v $m$, potom čas behu PAC učiaceho algoritmu je polynomiálny v $r$, $\delta^*$ a $\epsilon^{-1}$. Inak povedané, Occam algoritmus $L$ pre $H$ PAC učí každý $H_r$ podľa $H$ a urobí to efektívne vzhľadom na veľkosť reprezentácie a $\delta$ a $\epsilon$. Poznamenajme, že nemusí nutne platiť, že $H$ samotný je PAC naučiteľný aj keď by to mohlo tak byť, ak existuje horná hranica na veľkosť reprezentácie hypotézy v $H$.

## 1.6 Príklady Occam algoritmov

Predpokladajme, že je daný systém $\mathbf{S} = \{S_1, S_2, \ldots, S_n\}$ konečných množín $U = \bigcup_{i=1}^{n} S_i$. Chceme nájsť najmenšie podpokrytie $(U, \mathbf{S})$; t. j. najmenší podsystém $\overline{\mathbf{S}}$, ktorého zjednotením je $U$. Videli sme, že tento problém je NP-ťažký. Neznamená to, že neexistujú efektívne prostriedky na dosiahnutie aproximatívneho riešenia pre tento problém. Existuje jednoduchá intuitívna metóda na nájdenie aproximatívneho riešenia, založená na "greedy" metóde, ktorá sa zdá byť veľmi účinnou. Najprv vyberieme množinu $S_{j_1}$, ktorá obsahuje najväčší počet prvkov z $U$ a odstránime ju z $\overline{s}$. Potom vyberieme $S_{j_2}$, ktorá obsahuje najväčší počet zvyšných prvkov, atď. Pokračujeme týmto spôsobom, v každom kroku vyberieme množinu, ktorá obsahuje najväčší počet zvyšných prvkov.

**Greedy algoritmus pre minimálne pokrytie**

`set X=U;`
`while X`$\neq \emptyset$` do`
`begin choose `$S_j$` such that `$|S_j \cap X|$` is maximal;`
`set X=X-`$S_j$`;`
`end;`

Pretože $\overline{\mathbf{S}}$ pokrýva $U$, proces musí skončiť s podpokrytím $S' = \{S_{j_1}, S_{j_2}, \ldots, S_{j_k}\}$. Samozrejme, veľkosť $k$ výsledného podpokrytia nebude vo všeobecnosti najmenším možným podpokrytím, ale bolo ukázané (Nigmatullin (1969), Yohnsson (1974)), že platí nasledujúci vzťah: $k \leq l \cdot (\ln|U| + 1)$, kde $l$ je veľkosť minimálneho podpokrytia.

S



$S_1\ S_2 \quad S_1\ S_2 \quad S_4\ S_2$

$\cup$

$S_1\ S_2\ S_3\ S_4\ S_3$

Toto poskytuje dobrú hornú hranicu pre **pomer výkonnosti** $\frac{k}{l}$ a v tomto zmysle greedy algoritmus je dobrá aproximácia pre problém.

Čas behu závisí od $u = |U|$ a $n = |S|$, $\quad u \leq n, \quad k \leq \min(u, n) \quad \Rightarrow k \leq n \cdot (\ln u + 1)$.

Každý výberový krok obsahuje nájdenie maxima z najviac $n$ celých čísel a vymazanie najviac $u$ prvkov z každej z najviac $n$ množín. Počet operácií $O(u \cdot n)$. Celkový čas behu je $O(u \cdot n \cdot \min(u, n))$.

**Greedy** metóda môže byť použitá na odvodenie algoritmov pre určité triedy booleovských formúl. Podľa Hausslera (1988) ukážeme technicky ako greedy algoritmus pre pokrytie môže byť použitý na priestor $M_n$-monočlenov, ukážeme, že výsledný učiaci algoritmus je Occam.

Počiatočná hypotéza je jednočlen bez literálov, identicky 1-ková funkcia. V každom kroku je pridaný 1 literál do priebežnej konjunkcie literálov podľa pravidla založeného na greedy algoritme pre pokrytie.Budeme hovoriť, že literál $\lambda$ *eliminuje* negatívny príklad $x$, ak $\langle\lambda\rangle(x) = 0$. Vezmeme prvky, ktoré budú pokryté ako množinu záporných príkladov v danej tréningovej vzorke a pokrývajúce množiny ako množiny záporných príkladov eliminovaných literálmi určitého druhu. V každom stave vyberieme literál, ktorý eliminuje najväčší počet záporných príkladov zo vzorky, pridáme tento literál do formuly a vymažeme príklady, ktoré eliminuje.

Prečo toto pracuje?

Nech $\overline{s}$ je tréningová vzorka pre monočlen a $E$ nech je množina príkladov v $\overline{s}$ takých, že $E = E^+ \cup E^-$,
Pre ľubovoľný literál $\lambda$ položíme $S_\lambda = \{x \in E^- | \quad \langle\lambda\rangle(x) = 0\}$
Nakoniec, nech

$$\Lambda = \{\lambda | \quad \langle\lambda\rangle(x) = 1 \quad \forall x \in E^+\}$$

**Lema 1** *Systém množín* $\mathbf{S} = \{S_\lambda | \quad \lambda \in \Lambda\}$ *pokrýva* $E^-$.

**Dôkaz:** Pretože $\overline{s}$ je tréningová vzorka pre monočleny, vieme, že existuje monočlen $t = \langle\lambda_1 \wedge \ldots \wedge \lambda_l\rangle$ taký, že pre $x \in E, t(x)$ je 1 alebo 0 podľa toho či $x$ je v $E^+$ alebo v $E^-$. Toto implikuje, že $\lambda_1, \ldots, \lambda_l$ všetky patria do $\Lambda$. A tiež, že pre ľubovoľné $x \in E^-$ aspoň jeden z literálov $\lambda_j$ vyskytujúcich sa v $t$ je taký, že $\langle\lambda_j\rangle(x) = 0$. Inak povedané $x \in S_{\lambda_j} \in \mathbf{S}$.
□

**Lema 2** *Ak* $\mathbf{S}' = \{S_{\lambda_1}, \ldots, S_{\lambda_k}\}$ *je ľubovoľné podpokrytie* $(E^-, S)$, *potom monočlen* $h = \langle\lambda_1 \wedge \ldots \wedge \lambda_k\rangle$ *je konzistentný s* $\overline{s}$.

**Dôkaz:** Predpokladajme, že $x \in E^+$. Pretože $\lambda_1, \ldots, \lambda_k$ sú členmi $\Lambda$, na $x$ dávajú hodnotu 1, teda $h(x) = 1$. Ak $x \in E^-$, potom pretože $S'$ je podpokrytie, existuje $j$, $1 \leq j \leq k$ také, že $x \in S_{\lambda_j}$. Teda $\langle\lambda_j\rangle(x) = 0$, a teda $h(x) = 0$.
□

Tieto lemy ukazujú, že greedy algoritmus pre problém pokrytia môže byť transformovaný na algoritmus pre hľadanie monočlenov konzistentných s danou tréningovou vzorkou.

Aby sme zreteľne videli, že je to Occam algoritmus, uvažujme jeho chovanie sa na tréningovej vzorke pre monočlen $t$, ktorého najmenšia reprezentácia je pomocou formuly, ktorá obsahuje $l$ literálov. Minimálna reprezentácia $t$ je veľkosti $r = \lceil l \cdot \log n \rceil$. Výsledok práce greedy algoritmu pre problém pokrytia implikuje, že počet $k$ literálov vo výstupnej formule je taký, že $k \leq l \cdot (\ln |E^-| + 1)$.Preto veľkosť výstupnej formuly splňuje

$$||\omega|| = \lceil k \cdot \log n \rceil \leq \lceil l \cdot (\ln |E^-| + 1) \cdot \ln n \rceil \leq r \cdot (\ln |E|^- + 1)$$

Platí, že $|E|^- \leq m$.
$||\omega|| \leq r \cdot (\ln m + 1)$, čo triviálne implikuje Occam kompresnú podmienku.
$||\omega|| \leq m^\alpha \cdot r^\beta \quad \alpha = \frac{1}{2}, \quad \beta = 1$.
□

Greedy algoritmus sa líši evidentne od štandardného algoritmu pre monočleny. Namiesto začiatku s identicky nulovou funkciou (konjunkcia $2n$ literálov) a nasledujúcim vymazaním literalov použitím kladných príkladov greedy algoritmus štartuje s funkciou identicky rovnou 1 (prázdna konjunkcia literálov) a potom pridáva literály použiím záporných príkladov. Teda, pokým štandardný algoritmus je bezpamäťový on-line algoritmus, greedy algoritmus určite taký nie je. Avšak greedy algoritmus ako Occam algoritmus má dôležitú výhodu v tom, že jeho výstupom sú konzistentné hypotézy, ktoré sú relatívne jednoduché.

## 1.7 Epac učenie

Predpokladajme, že $H = \bigcup H_n$ je hypotézový veľkosťou príkladov odstupňovaný priestor a $\Omega \to H$ je reprezentácia pre $H$. Potom by sme mohli odstupňovať každé $H_n$ pomocou veľkosti reprezentácie takto $H_n = \bigcup H_{n,r}$, kde $H_{n,r}$ pozostáva z tých hypotéz $H_n$, ktoré majú minimálnu veľkosť reprezentácie $r$. Teda

$$H = \bigcup_n \bigcup_r H_{n,r}$$

je *dvojite odstupňovaný*.

Nech $L$ je učiaci algoritmus pre $H$ v obvyklom zmysle, že $L(\bar{s})$ je v $H_n$ vždy, keď $\bar{s}$ je tréningová vzorka pre hypotézy v $H_n$. Hovoríme, že $L$ je *efektívny PAC* alebo *ePAC* ak (Valiant, 1991)

- čas behu $R_L(m,n)$ je polynomiálny v $m$ aj v $n$;
- zložitosť vzorky $m_L(H_{n,r}, \delta, \epsilon)$ je polynomiálna v $n, r, \delta^*$ a $\epsilon^{-1}$.

Teda $ePAC$ učiaci algoritmus zaručuje, že vydá pravdepodobnostne aproximovaný správny výstup s časom behu polynomiálnym v $n, r, \delta^*$ a $\epsilon^{-1}$.

Jeden spôsob na zaručenie druhej podmienky je použitie nejakej verzie Occam podmienok. V tomto kontexte hovoríme, že $L$ je Occam ak podmienky stanovené v definícii pre Occam algoritmus platia pre každé $H_n$ s konštantami $\alpha$ a $\beta$ nezávislými od $n$. Potom máme nasledujúci výsledok.

**Veta 3** *Predpokladajme, že hypotézový priestor je $H = \bigcup H_{n,r}$ ako bolo uvedené vyššie a $L$ je Occam algoritmus pre učenie $H_{n,r}$ pomocou $H_n$ s polynomiálnym časom behu $R_L(m,n)$. Potom $L$ je $ePAC$.*

**Dôkaz:** Z dôkazu vety vyššie uvedenej máme hornú hranicu

$$m_0(H_r, \delta, \epsilon) = \lceil (\frac{A+B}{\epsilon})^{1/(1-\alpha)} \rceil$$

pre zložitosť vzorky algoritmu $L$ na $H_{r,n}$, kde $A = r^\beta \ln 2$ a $B = \ln(2/\delta)$. Ako sme už poznamenali, toto je polynóm v $r, \delta^*$ a $\epsilon^{-1}$. Pretože $\alpha$ a $\beta$ sú nezávislé od $n$, aj $m_0(H_r, \delta, \epsilon)$ je nezávislé od $n$. Výsledok vyplýva z toho, že horná hranica na čas behu algoritmu $L$ v $PAC$ učení $H_{n,r}$ je

$$R_L(m_0(H_r, \delta, \epsilon), n)$$

čo je polynóm v $n, r, \delta^*$ a $\epsilon^{-1}$.

□

**Príklad:** Odstupňovaný hypotézový priestor $M = \bigcup M_n$ monočlenov môže byť odstupňovaný dvojite ako $M = M_{n,r}$, kde $M_{n,r}$ pozostáva z tých monočlenov $n$ premenných, ktoré majú veľkosť reprezentácie $r$. V predchádzajúcom odseku bol popísaný algoritmus pre učenie $M_{n,r}$ pomocou $M_n$ založený na greedy metóde a ukázali sme, ze má Occam vlastnosť pri $\alpha = 1/2$ a $\beta = 1$. Čas behu $R_L(m,n)$ je $O(m.n.min(m,n))$, čo je určite polynóm v $m$ a $n$. Z toho nám vyplýva, že greedy algoritmus pre $M$ je $ePAC$.

□

## 1.8 Ďalšie poznámky

Ako sme už uviedli, fakt, že $C^k$ nie je efektívne naučiteľný vzhľadom na veľkosť príkladov, je výsledok, ktorý závisí od reprezentácie. Keby výstupné hypotézy mohli byť reprezentované iným spôsobom než konjunkcie najviac $k$ klauzúl, tak generovanie pravdepodobnostne aproximatívne korektných hypotéz by mohlo byť jednoduchšie. Kearns a Valiant (1989) v tomto smere dosiahli veľmi silný výsledok založený na kryptografických predpokladoch.

## 1.9 Úlohy

1. Ukážte, že $C_n^k \subseteq D_{n,k}$ pre všetky $k$ a $n$ a že inklúzia je striktná pre niektoré hodnoty $n$ a $k$.