ROBUST'2002, 256 – 267

LESK A BÍDA OPTIMÁLNÍCH STROMŮ

PETR SAVICKÝ, JAN KLASCHKA

Abstract. Optimal classification trees have, by the definition, the smallest error on training data, given the number of leaves. Experiments reported in the Robust 2000 paper suggest that, as regards the generalization properties (i.e. the precision on data unused for training), the optimal trees might be consistently at least as good as the trees grown by classical methods. The results of more thorough experimenting, presented in the current paper, demonstrate, however, that for some classification problems the optimal trees are outperformed by the classical ones.

Резюме. Оптимальные классификационные деревья обладают минимальной ошибкой на используемых в процессе конструкции данных. Первое их на конференциях Робуст представление осуществилось в 2000-ом году. Совершенные до тех пор эксперименты позволяли оптимисм, что касается ошибки на других, независимых данных, поскольку оптимальные деревья показывались лучшими конструируемых традиционными методами деревьев, или, по крайней мере, сравнимыми с ними. Последующие исследования раскрыли, однако, более сложную диалектику сравнения оптимальных деревьев с классическими. Оптимизация, по сравнению с классическими методами, дает лучшие решения одних классификационных проблем, а хучшие решения других. В этой статье приведены примеры обеих родов. Построение критерий позвольяющих различение принадлежимости конкретной проблемы к той, или иной категории, является важной задачей для будущих исследований.

1. Úvod: Od ROBUSTu k ROBUSTu

V pracích Savický *et al.* (2000, 2001) jsme popsali dva algoritmy vytvářející tzv. *optimální klasifikační stromy*, tj. stromy, které mají při daném počtu koncových uzlů nejmenší možnou chybu na trénovacích datech. Současně jsme uvedli výsledky několika numerických experimentů s uměle generovanými daty, v nichž jsme studovali generalizační vlastnosti optimálních klasifikačních stromů, tj. chování na datech nepoužitých ke konstrukci stromu, generovaných z téhož rozdělení jako trénovací data. Ze srovnání se stromy vypěstovanými klasickou metodou CART (viz Breiman *et al.*, 1984) vyšly optimální stromy dobře: *Většinu úloh řešily lépe, žádnou hůře.* Toto pozorování naznačilo, že by generalizační vlastnosti optimálních stromů mohly být *univerzálně* dobré.

V době mezi ROBUSTem 2000 a 2002 jsme však provedli další numerické experimenty daleko většího rozsahu, a výsledek se dostavil: Někdejší optimismus vzal za své. Vedle úloh, kde optimální stromy nad klasickými „vítězí na celé čáře", známe dnes také klasifikační problémy, u kterých vede aplikace našich optimalizačních algoritmů k prokazatelně horším výsledkům než klasické metody. Jádrem tohoto článku

2000 *Mathematics Subject Classification.* Primary 62H30; Secondary 62-07.
Klíčová slova. Klasifikační stromy, optimální stromy, generalizační vlastnosti, Occamova břitva.
Tato práce byla podporována grantem GA ČR 201/00/1482.

je právě to, co jsme v pracích z COMPSTATu 2000 a ROBUSTu 2000 nemohli ukázat, totiž demonstrace „prohry“ optimálních stromů.

2. Základní značení, pojmy a fakta

V této práci studujeme speciální případy klasifikačních úloh s prediktory (nezávisle proměnnými) $X_1, \ldots, X_P$ s hodnotami v oboru reálných čísel a závisle proměnnou Y, jejíž obor hodnot je nějaká konečná množina C (množina tříd). Předpokládáme, že $(X_1, \ldots, X_P, Y)$ je náhodný vektor s distribucí D. Datový soubor $\mathcal{L}$ (ať už trénovací, nebo jiný) v našich úvahách sestává z $(P+1)$-tic $(x_1, \ldots, x_P, y)$ a představuje náhodný výběr z rozdělení D.

Klasifikační úloha obecně spočívá v konstrukci klasifikační funkce f, která každému vektoru $(x_1, \ldots, x_P)$ hodnot prediktorů přiřadí hodnotu $y = f(x_1, \ldots, x_P) \in C$ závisle proměnné Y. V případě klasifikačních stromů (přesněji binárních klasifikačních stromů založených na „jednorozměrných větveních“[1]), kterými se budeme zabývat, je klasifikační funkce f_T reprezentována binárním rozhodovacím stromem T. Každému uzlu stromu T, který není koncový, je přiřazena otázka týkající se hodnoty jednoho z prediktorů, na niž se dá odpovědět „ano“, nebo „ne“, zatímco koncovému uzlu (listu) je přiřazena některá z tříd. Velikostí stromu T rozumíme počet jeho listů $|T|$.

Klasifikační strom T se považuje za tím úspěšnější, čím menší je jeho *skutečná ztráta*

$$R_D(T) = \sum_{i \in C} \sum_{j \in C} c(i,j) p(i,j), \tag{1}$$

kde $c(i,j)$ je „velikost škody“ vzniklé zařazením jednoho případu ze třídy i do třídy j a $p(i,j)$ je pravděpodobnost, že pozorování náhodně „tažené“ z distribuce D bude patřit do třídy i a bude zařazeno do třídy j, tj. $p(i,j) = D\big(Y = i, f_T(X_1, \ldots, X_P) = j\big)$. Empirický odhad skutečné ztráty $R_D(T)$ se hodí odvozovat spíše než z (1) z ekvivalentního vzorce

$$R_D(T) = \sum_{i \in C} \pi_i \sum_{j \in C} c(i,j) p(j|i), \tag{2}$$

kde $\pi_i = D(Y = i)$ je *apriorní pravděpodobnost* třídy i a $p(j|i)$ je pravděpodobnost klasifikace pozorování ze třídy i do j, tedy $p(j|i) = D\big(f_T(X_1, \ldots, X_P) = j | Y = i\big)$.

Nechť $\mathcal{L}$ je datový soubor velikosti n. Označme n_i počet pozorování ze třídy i a n_{ij} počet těch případů ze třídy i, které jsou klasifikovány jako j. Empirickým protějškem skutečné ztráty $R_D(T)$ je $R_{\mathcal{L}}(T)$, *ztráta na souboru* $\mathcal{L}$, kterou dostaneme z pravé strany (2) nahrazením π_i a $p(j|i)$ vhodnými odhady. Podmíněnou pravděpodobnost $p(j|i)$ odhadujeme podílem n_{ij}/n_i. Pokud nemáme (nebo nechceme použít) jiné informace o apriorních pravděpodobnostech π_i, než jaké poskytuje soubor $\mathcal{L}$, můžeme je odhadnout jako n_i/n, takže dostaneme (stejně jako „přímo“ z (1))

$$R_{\mathcal{L}}(T) = \frac{1}{n} \sum_{i \in C} \sum_{j \in C} c(i,j) n_{ij}. \tag{3}$$

Pokud jsou k dispozici externí odhady apriorních pravděpodobností (označíme je s jistou licencí stejně jako samotné apriorní pravděpodobnosti, tj. π_i), a zejména

[1] „Jednorozměrné větvení“ (linear split) dělí případy do dvou podmnožin na základě hodnot jediného prediktoru, na rozdíl např. od „lineárního větvení“ (linear split), které rozděluje případy podle toho, zda hodnota lineární kombinace několika prediktorů je, či není větší než daná mez.

tehdy, liší-li se relativní četnosti tříd v datech výrazně od apriorních pravděpodobností, vyjadřujeme $R_{\mathcal{L}}(T)$ jako

$$R_{\mathcal{L}}(T) = \sum_{i \in C} \pi_i \sum_{j \in C} c(i,j) \frac{n_{ij}}{n_i}.$$

V tomto článku pokládáme vesměs $c(i,j) = 0$ pro $i = j$ a $c(i,j) = 1$ pro $i \neq j$. Místo o *ztrátě* (ať už skutečné, nebo na souboru $\mathcal{L}$) pak můžeme mluvit o *klasifikační chybě* (popř. jen *chybě*) stromu T. Vzorec (1) se v tomto případě redukuje na

$$R_D(T) = 1 - \sum_{i \in C} p(i,i),$$

takže skutečná chyba se rovná pravděpodobnosti, že náhodně zvolené pozorování bude klasifikováno chybně. Obdobně (3) přechází v

$$R_{\mathcal{L}}(T) = 1 - \frac{1}{n} \sum_{i \in C} n_{ii}.$$

Klasifikační chyba na souboru $\mathcal{L}$ je tedy rovna – připomeňme, že jen za předpokladů o apriorních pravděpodobnostech uvedených v souvislosti se vzorcem (3) – relativní četnosti chybně klasifikovaných pozorování.

3. Optimální a klasické stromy

Klasifikační strom T je *optimální* na souboru $\mathcal{L}$, pokud $R_{\mathcal{L}}(T') \geq R_{\mathcal{L}}(T)$ pro každý klasifikační strom T' téže velikosti jako T. Pokud mluvíme o optimálních stromech, aniž bychom uváděli, na jakém souboru jsou optimální, máme na mysli trénovací soubor použitý ke konstrukci. V pracích Savický *et al.* (2000, 2001) byly popsány dva algoritmy, které umožňují konstrukci optimálních klasifikačních stromů, byť za dosti přísných omezujících podmínek: Prediktory musejí nabývat pouze hodnot 0 a 1 a „nesmí jich být mnoho“, protože nároky na paměť rostou s počtem prediktorů exponenciálně. Oba algoritmy konstruují stromy různých velikostí. Tzv. *Algoritmus I* vytváří při každém běhu jeden optimální strom, jehož velikost je nepřímo dána hodnotou jistého vstupního parametru (pro podrobnosti odkazujeme na citované práce). Postupnou volbou různých hodnot daného parametru lze získat různě velké stromy. Velikosti stromů, které takto lze sestrojit, však netvoří souvislou řadu. Věta 1 v článku Savický *et al.* (2001) charakterizuje množinu „dosažitelných“ velikostí. Tzv. *Algoritmus II* vytváří simultánně optimální klasifikační stromy všech velikostí od 1 do zvolené meze M. Algoritmus II má oproti Algoritmu I přibližně M-krát větší nároky na operační paměť. Podle našich dosavadních zkušeností však dává o něco lepší výsledky. Proto jsme v experimentech, o nichž referujeme v této práci, konstruovali optimální klasifikační stromy pomocí Algoritmu II.

Klasické metody jako CART (Breiman *et al.*, 1984), C4.5 (Quinlan, 1986) nebo QUEST (Loh a Shih, 1997) se v zásadě snaží také konstruovat stromy s co nejmenší chybou (resp. ztrátou) na trénovacím souboru. Začnou s jediným uzlem (který je současně kořenem i listem) a postupně přidávají další uzly tak, aby se v každém kroku chování stromu na trénovacích datech v jistém smyslu (přesněji specifikovaném v citované literatuře) co nejvíce zlepšilo. Tyto „lokálně optimální“ kroky však nezaručují, že výsledný strom bude na trénovacích datech optimální.

Jak u optimálních, tak u klasických stromů je třeba nějak stanovit vhodnou velikost. Je známo, že příliš malé stromy mají obvykle jak velkou chybu na trénovacích datech, tak i velkou chybu skutečnou, zatímco stromy příliš velké mají typicky na trénovacích datech velmi malou chybu, ale jejich skutečná chyba je o mnoho větší.

V CARTu (kde se takový přístup objevil poprvé) a v řadě dalších metod se vypěstuje „přehnaně“ velký strom, a jeho prořezáváním (postupným odebíráním uzlů) se pak vytvoří posloupnost různě velkých stromů. Chyba každého z těchto stromů se odhadne buďto pomocí jiných dat nepoužitých v trénovacím souboru, nebo bez použití jiných dat složitější technikou tzv. křížové validace (cross-validation). Jako „správně velký“ strom, který je konečným výstupem metody, se přijme ten strom z posloupnosti, pro nějž je odhad chyby nejpříznivější (případně strom o něco menší, u nějž je odhad skutečné chyby skoro stejně příznivý).

V případě optimálních stromů napodobujeme CART v tom, že z posloupnosti různě velkých stromů (jež však nemusejí být prořezanými podstromy téhož velkého stromu) vybíráme strom s nejmenší chybou na validačním datovém souboru.

4. Optimalita – proč? Vivat Occamova břitva!

Obvykle příliš nepochybujeme o tom, že chceme-li získat co nejlepší generalizaci, je vhodné dávat přednost modelům, které dosahují dobré shody s trénovacími daty[2]. Snaha o co nejlepší shodu modelu s daty je v analýze dat všudypřítomná. Víme ovšem, že ji nesmíme přehánět, a používat příliš složité modely na příliš málo dat. Pokus proložit regresní model o dvaceti parametrech deseti pozorováními nám asi rozmluví program, který si dříve, než havaruje, stačí postěžovat na nějakou singulární matici. Menší disproporce mívají méně nápadné důsledky: Může dojít k tomu, že se model „naučí“ příliš mnoho vlastností dat – nejen ty, které vyplývají z povahy zdroje a měly by se opakovat v jiných výběrech, ale i vlastnosti nahodilé, přítomné právě jen v daném souboru. Nastává pak to, čemu se anglicky říká *overfit*: Model „sedne“ výborně trénovacím datům, ale jeho generalizační vlastnosti jsou o mnoho horší.

Důvod, proč jsme mohli a priori doufat, že optimální stromy budou mít i dobré generalizační vlastnosti, spočívá v tom, že chyba na trénovacích datech je minimalizována *při daném počtu listů*. Nehrozí tedy, že ve srovnání s klasickými metodami bude model zpřesňován na trénovacích datech za cenu zvětšování stromu.

Tzv. princip Occamovy břitvy[3] doporučuje volit ze dvou modelů, které popisují stejně přesně trénovací data, ten menší, protože bude pravděpodobně mít lepší generalizační vlastnosti.

Pokud tomuto principu věříme (není to matematická věta), měli bychom z něj čerpat i značný stupeň důvěry k optimálním stromům. Minimalizace chyby při dané velikosti stromu není sice tatáž úloha jako minimalizace velikosti stromu při dané chybě, ale jsou to úlohy velmi podobné. (Úlohy nelze ztotožnit, když pro nějaké $m \geq 1$ je minimální dosažitelná chyba na trénovacím souboru stejná pro stromy velikosti m i $m + 1$.)

První experimentální výsledky shrnuté v pracích Savický *et al.* (2000, 2001) optimismus ohledně generalizačních vlastností optimálních stromů posílily. V jedné úloze optimální stromy vykazovaly průměrnou skutečnou chybu srovnatelnou s metodou

[2]Nemusí být takový nesmysl, jak by se mohlo na první pohled zdát, usilovat o *co nejhorší* shodu s daty. Vzpomeňme na pana Gypskopfa z Parkinsonových zákonů, který „je ... naprosto neocenitelný, má v sobě schopnost převrácené neomylnosti. Naprosto jistě dokáže pokaždé ukázat nesprávným směrem ... Pro nás je kompasem se střelkou ukazující k jihu.“ (Parkinson, 1984, str. 144.) Lákavá je také vize kompromisní třetí cesty, „dataanalytického světa“, kde by shoda modelu s daty nemusela být ani co nejlepší, ani co nejhorší – prostě by na ní nezáleželo. Modely bychom totiž mohli svobodně vytvářet, aniž bychom se s nějakými daty museli trápit.

[3]Přesněji řečeno, termín „Occamova břitva“ se používá v několika významech, označuje více různých tezí. Přehled včetně souvislostí s filosofií Williama Occama (?1285-?1349) podává např. Domingos (1999). Formulace, kterou uvádíme, se v citované práci nazývá „druhou břitvou“.

CART, v ostatních úlohách pak byly lepší. Poznání, že věci nejsou tak jednoduché, na nás teprve čekalo.

5. Experimenty

5.1. **Klasifikační úlohy.** V experimentech se simulovanými daty jsme studovali několik klasifikačních úloh, které lze verbálně charakterizovat jako rozpoznávání vlastností definovaných booleovskými funkcemi za přítomnosti dvou typů šumu: Hodnoty „užitečných" prediktorů, které určují příslušnost k třídě, jsou zkresleny šumem, a navíc používáme také irelevantní „šumové" prediktory.

Vesměs jsme pracovali s deseti binárními prediktory (včetně irelevantních) a binární závisle proměnnou. Závisle proměnná Y je dána jako $Y = g(\xi_1, \ldots, \xi_r)$, kde $r \leq 10$, sdružené rozdělení proměnných $\xi_1, \ldots, \xi_r$ je rovnoměrné na $\{0,1\}^r$ a g je funkce r binárních proměnných nabývající hodnot 0 a 1. V datech nemáme přímo hodnoty veličin $\xi_1, \ldots, \xi_r$, ale jejich „zašuměných" verzí $X_1, \ldots, X_r$. Šum představují veličiny $\theta_1, \ldots, \theta_r$, které jsou nezávislé na $\xi_1, \ldots, \xi_n$ i navzájem a každá z nich nabývá hodnot 1 a 0 s danou pravděpodobností ν, resp. $1 - \nu$. O ν mluvíme jako o *hladině šumu*. Pokud $\theta_i = 1$, „i-tý prediktor se šumem pokazí", tj. $X_i = 1 - \xi_i$. V opačném případě, tj. pokud $\theta_i = 0$, platí $X_i = \xi_i$. Irelevantní prediktory $X_{r+1}, \ldots, X_{10}$ jsou nezávislé jak navzájem, tak na $\xi_1, \ldots, \xi_r, \theta_1, \ldots, \theta_r$, a každý z nich nabývá hodnot 0 a 1 se stejnou pravděpodobností $1/2$.

Naše úlohy se dělí do dvou skupin podle toho, co na nich chceme ukázat: Zda *lesk* optimálních stromů, nebo jejich *bídu* – tedy zda demonstrujeme lepší, nebo naopak horší generalizační vlastnosti ve srovnání s klasickými stromy. V dalším textu se konkrétně zmíníme o dvou úlohách, po jedné z každé z těchto kategorií. Přehled výsledků týkajících se dalších úloh, jimiž jsme se zabývali, lze nalézt v práci Savický a Klaschka (2001).

5.2. **Obecná struktura experimentů.** Provedli jsme několik experimentů s daty generovanými podle schematu popsaného v předcházejícím paragrafu. V každém z těchto experimentů jsme se zabývali klasifikační úlohou spojenou s nějakým rozdělením D daným konkrétní volbou funkce g (kterou je implicitně určen také počet irelevantních proměnných) a hladiny šumu ν. Pro každé zvolené rozdělení D jsme experimentovali s trénovacími soubory několika velikostí odstupňovaných tak, aby přibližně tvořily geometrickou posloupnost. Generovali jsme náhodně vždy 100 trénovacích souborů dané velikosti a ke každému z nich soubor tzv. validačních dat poloviční velikosti. (To odpovídá doporučení odvozenému ze zkušenosti – viz Breiman *et al.* (1984) – rozdělit data v poměru 2 : 1 na dvě části, z nichž první se použije k vytvoření modelu a druhá k odhadu jeho chyby.)

Na každý trénovací soubor jsme použili Algoritmus II a vytvořili jsme posloupnost optimálních stromů velikosti od 1 do hranice M, která se podle typu problému pohybovala mezi 100 a 160. Pro tyto stromy jsme vypočetli chybu na validačních datech a její minimalizací vybrali z posloupnosti strom T_{opt}. Pro porovnání jsme tatáž trénovací data použili k vypěstování stromu metodou CART (implementovanou v programu CART for Windows, Version 4.0) a validační data k jeho prořezání. Minimalizací chyby na validačních datech (tedy podle pravidla 0 SE v terminologii CARTu) jsme získali strom T_{cart}. Znalost skutečného rozdělení nám dovolila spočítat poté skutečnou chybu stromů T_{opt} a T_{cart}. (V tomto ohledu jsme novátorsky zdokonalili metodiku experimentů, o nichž referujeme v pracích Savický *et al.* (2000, 2001) – tehdy jsme skutečnou chybu odhadovali pomocí velkého testovacího souboru.)

Pro dané rozdělení D a velikost trénovacího souboru jsme výše uvedeným postupem získali 100 dvojic skutečných chyb $R_D(T_{\text{opt}})$ a $R_D(T_{\text{cart}})$. Průměrné skutečné chyby optimálních stromů a stromů vytvořených metodou CART jsme pak porovnávali oboustranným Studentovým t-testem.

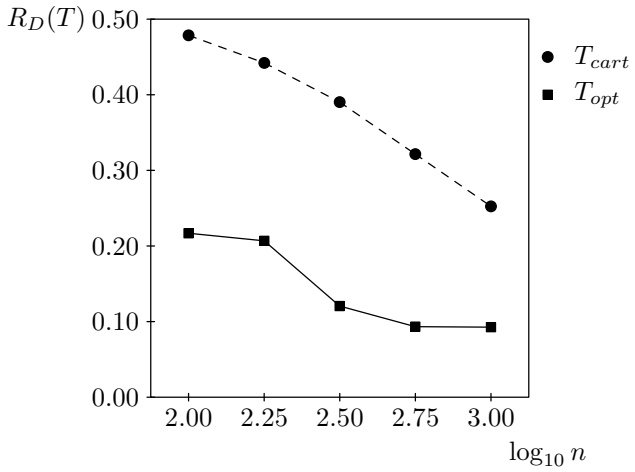
Pro úplnost budiž ještě uvedeno, jak jsme při výpočtech pomocí optimalizačního algoritmu i metodou CART volili parametry ve vzorcích (1) a (3): Každý chybně klasifikovaný případ byl penalizován stejně, $c(i,j) = 1$ pro $i \neq j$ a $c(i,i) = 0$ pro všechna i. Jako apriorní pravděpodobnosti tříd π_i jsme používali skutečné pravděpodobnosti $D(Y = i)$.

5.3. **Lesk: Parita.** *Úloha rozpoznání parity* je založena na funkci parity pěti proměnných

$$g(x_1, \ldots, x_5) = x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5,$$

kde $\oplus$ je součet modulo 2 ($0 \oplus 0 = 1 \oplus 1 = 0$, $0 \oplus 1 = 1 \oplus 0 = 1$). Vektor prediktorů je doplněn pěti irelevantními proměnnými. Hladina šumu byla $\nu = 0.02$. Pracovali jsme s pěti velikostmi souborů trénovacích dat, konkrétně se jednalo o mocniny $10^2, 10^{2.25}, \ldots, 10^3$ zaokrouhlené na celá čísla, tedy 100, 178, 316, 562 a 1000. Maximální velikost optimálních stromů byla $M = 100$. (Nejmenší strom, který přesně vyjadřuje funkci g, má velikost 32.)

Výsledky jsou shrnuty graficky na Obr. 1 a podrobněji numericky v Tab. 1.



Obr. 1. *Problém rozpoznávání parity: Průměrná skutečná chyba 100 optimálních stromů a 100 stromů vypěstovaných metodou CART při různých velikostech souborů trénovacích dat n.*

Počet trén. dat	Skutečná chyba (%) průměr±SE 100 stromů		Rozdíly průměr±SE	Párový t-test (df=99)	
	Opt. stromy	CART		t	p
100	21.68±0.54	47.86±0.36	-26.17±0.66	-39.45	0.000 000
178	20.68±0.39	44.21±0.53	-23.53±0.66	-35.72	0.000 000
316	12.06±0.28	39.03±0.53	-26.97±0.56	-48.60	0.000 000
562	9.33±0.04	32.15±0.41	-22.82±0.41	-54.99	0.000 000
1000	9.26±0.01	25.24±0.33	-15.97±0.33	-48.05	0.000 000

Tab. 1. *Problém rozpoznávání parity: Skutečné chyby 100 optimálních stromů a 100 stromů vypěstovaných metodou CART při různých velikostech souborů trénovacích dat.*

Je evidentní, že optimální stromy jsou při všech studovaných velikostech dat n daleko přesnější než stromy klasické. Vysvětlení je nasnadě. Rozpoznání parity se podobá kombinaci v šachu – vyžaduje schopnost „vidět“ více kroků dopředu (v anglické literatuře *lookahead*), což je právě klíčový rys optimalizačních algoritmů, který u klasických metod chybí. CART (podobně jako to dělají všechny metody, o nichž mluvíme jako o klasických) vybírá větvení v kořeni podle vlastností stromu o dvou listech, a z tohoto pohledu se proměnné $X_1, \ldots, X_5$ mohou jevit jako málo užitečné (žádná z nich neumožňuje dobrou predikci třídy *sama o sobě*). Snadno se tak stane, že se náhodně prosadí některá z irelevantních proměnných. Analogické „omyly“ se pochopitelně dějí i při hledání dalších vhodných větvení, nikoli jen prvního větvení v kořeni. Každým větvením podle irelevantní proměnné se ovšem soubor drobí na menší části, ve kterých je rozpoznání funkce parity stále obtížnější. Naše optimalizační algoritmy naproti tomu vybírají nejvhodnější větvení v kořeni (nebo v kterémkoli jiném uzlu) podle toho, jaký bude mít efekt *v kombinaci s větveními v ostatních uzlech*.

Připomeňme, že o podobné úloze rozpoznání parity pěti proměnných jsme referovali již v pracích Savický *et al.* (2000, 2001). Nynější experiment je však řádově rozsáhlejší (tehdy jsme analyzovali jen deset trénovacích souborů velikosti 500), ale hlavně se změnila povaha proměnných $X_6, \ldots, X_{10}$. Místo „slušnějších“ irelevantních proměnných jsme ve starším experimentu používali tzv. *zavádějící proměnné* (v pracovní hantýrce „misleadery“) – proměnné závislé na třídě Y, které se klasickým metodám důrazněji „vnucují“ k použití v modelu, ale přitom nestačí k vytvoření modelu dostatečně přesného.

5.4. **Bída: Majorita.** *Úloha rozpoznání majority (většiny)* je založena na funkci

$$g(x_1, \ldots, x_7) = \begin{cases} 1 & \sum_{i=1}^{7} x_i \geq 4, \\ 0 & \text{jinak.} \end{cases}$$

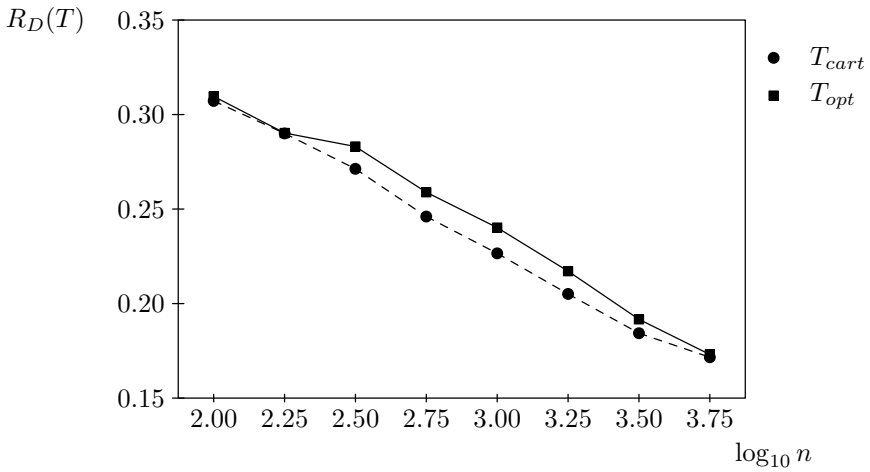
Vektor prediktorů je doplněn třemi irelevantními veličinami. Použili jsme hladinu šumu $\nu = 0.1$. Analyzovali jsme trénovací soubory velikostí $n = 10^2, 10^{2.25}, \ldots, 10^{3.75}$ zaokrouhlených na celá čísla, tj, 100, 178, 316, 562, 1000, 1778, 3162 a 5623. Maximální velikost optimálních stromů byla $M = 160$. (Funkci g lze přesně vyjádřit pomocí stromu velikosti 70.)

Funkce většiny g je monotónní a „hraje do karet“ klasickým metodám konstrukce stromů, které hledají co nejlepší větvení jen „jeden krok dopředu“. Přesto by člověk mohl a priori očekávat, že si optimalizační algoritmus s úlohou poradí ne-li lépe, tedy alespoň stejně dobře.

Výsledky shrnuté v grafu na Obr. 2 a podrobněji v Tab. 2 však jasně ukazují, že optimální stromy mají pro velikosti mezi 316 a 5162 v průměru větší skutečnou chybu než stromy zkonstruované metodou CART.

Podobné výsledky jsme získali pro několik dalších klasifikačních úloh založených (dle schematu popsaného v paragrafu 5.1) na monotónních booleovských funkcích. Pro přehled odkazujeme na technickou zprávu Savický a Klaschka (2001), dostupnou v plném znění na Internetu.

5.5. **Variace experimentu, aneb bída s jistotou.** Výsledky týkající se majority (popř. dalších obdobných úloh) mohou přeci jen budit určitou nedůvěru: Co když „prohra“ s metodou CART není způsobena ani tak tím, že by optimální stromy nebyly dobré, jako spíše tím, že z posloupnosti různě velkých optimálních stromů nevybíráme správně „ten pravý“?



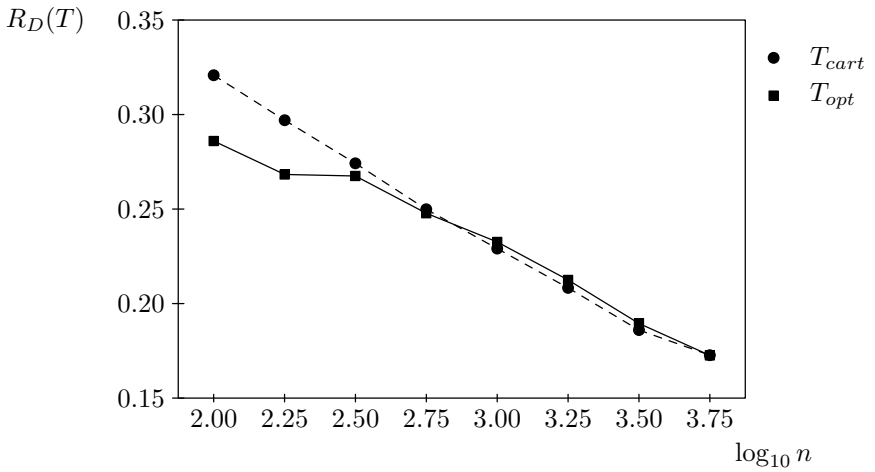
Obr. 2. *Problém rozpoznávání majority: Průměrná skutečná chyba 100 optimálních stromů a 100 stromů vypěstovaných metodou CART při různých velikostech souborů trénovacích dat n.*

Počet trén. dat	Skutečná chyba (%) průměr±SE 100 stromů		Rozdíly průměr±SE	Párový t-test (df=99)	
	Opt. stromy	CART		t	p
100	30.96±0.20	30.72±0.27	0.24±0.30	0.80	0.423 116
178	29.02±0.19	29.00±0.23	0.02±0.25	0.09	0.931 057
316	28.31±0.18	27.13±0.15	1.18±0.20	5.78	0.000 000
562	25.89±0.13	24.60±0.11	1.29±0.13	10.07	0.000 000
1000	24.02±0.10	22.66±0.11	1.36±0.12	11.67	0.000 000
1778	21.71±0.08	20.51±0.08	1.20±0.08	15.10	0.000 000
3162	19.17±0.06	18.43±0.07	0.74±0.06	12.09	0.000 000
5623	17.31±0.04	17.16±0.03	0.15±0.03	5.42	0.000 000

Tab. 2. *Problém rozpoznávání majority: Skutečné chyby 100 optimálních stromů a 100 stromů vypěstovaných metodou CART při různých velikostech souborů trénovacích dat.*

Těmto spekulacím učinila rázně přítrž variace původního experimentu. Dobrá, jeli v posloupnosti optimálních stromů nějaký strom s dobrými generalizačními vlastnostmi, vyberme jej místo stromu T_{opt}. Jak? V reálné analýze dat bychom návod po ruce neměli, ale zde známe skutečné rozdělení, a tudíž i skutečnou chybu každého stromu v posloupnosti. Za výstup metody bereme prostě ten strom T'_{opt}, který má v celé posloupnosti nejmenší skutečnou chybu. Protože při konstrukci a výběru stromu nehrají žádnou roli validační data, vzdáváme se validačních dat i při aplikaci metody CART: tu nyní místo T_{cart} reprezentuje strom T'_{cart} získaný prořezáváním křížovou validací (cross-validation).

Graf na Obr. 3 ukazuje, že rozdíl v neprospěch optimálních stromů dobře viditelný na Obr. 2 se do značné míry setřel. Pro malé velikosti souboru se dokonce poměry výrazně změnily ve prospěch optimálních stromů, ale to lze snadno vysvětlit – znalost skutečné chyby poskytuje při výběru stromu z posloupnosti obrovskou výhodu, protože validační data jsou také malá a dávají nespolehlivý odhad skutečné chyby. V Tab. 3 však vidíme (lépe než v grafu), že pro $n = 1000, 1778$ a 3162 jsou rozdíly ve prospěch metody CART stále vysoce statisticky významné.



Obr. 3. *Problém rozpoznávání parity, modifikovaný experiment: Průměrná skutečná chyba 100 optimálních stromů a 100 stromů vypěstovaných metodou CART při různých velikostech souborů trénovacích dat n. Velikost optimálního stromu je volena na základě skutečné chyby, nikoli chyby na validačních datech.*

Počet trén. dat	Skutečná chyba (%) průměr±SE 100 stromů		Rozdíly průměr±SE	Párový t-test (df=99)	
	Opt. stromy	CART		t	p
100	28.60±0.16	32.08±0.35	-3.48±0.34	-10.28	0.000 000
178	26.83±0.15	29.70±0.27	-2.86±0.29	-9.80	0.000 000
316	26.75±0.11	27.42±0.17	-0.66±0.17	-3.91	0.000 170
562	24.78±0.09	24.99±0.13	-0.21±0.13	-1.69	0.094 232
1000	23.26±0.07	22.91±0.09	0.35±0.08	4.27	0.000 044
1778	21.25±0.07	20.83±0.08	0.41±0.07	6.04	0.000 000
3162	18.95±0.05	18.60±0.06	0.35±0.05	7.59	0.000 000
5623	17.26±0.03	17.27±0.04	-0.01±0.02	-0.48	0.633 092

Tab. 3. *Problém rozpoznávání parity, modifikovaný experiment (volba velikosti optimálního stromu podle skutečné chyby): Skutečné chyby 100 optimálních stromů a 100 stromů vypěstovaných metodou CART při různých velikostech souborů trénovacích dat.*

Ukazuje se tedy, že problém není (jen) ve výběru vhodné velikosti optimálního stromu, ale ve struktuře všech stromů konstruovaných optimalizačními algoritmy.

6. Jak je to možné? Pryč s Occamovou břitvou!

Spolu s optimálními stromy je v úloze rozpoznávání majority (a dalších podobných) bit i princip Occamovy břitvy. Není to ovšem jeho první porážka. Empirické argumenty proti Occamově břitvě, dokonce založené na klasifikačních stromech, se v literatuře objevují. (Pro přehled různých teoretických i empirických argumentů ve prospěch Occamovy břitvy i proti ní viz Domingos, 1999.)

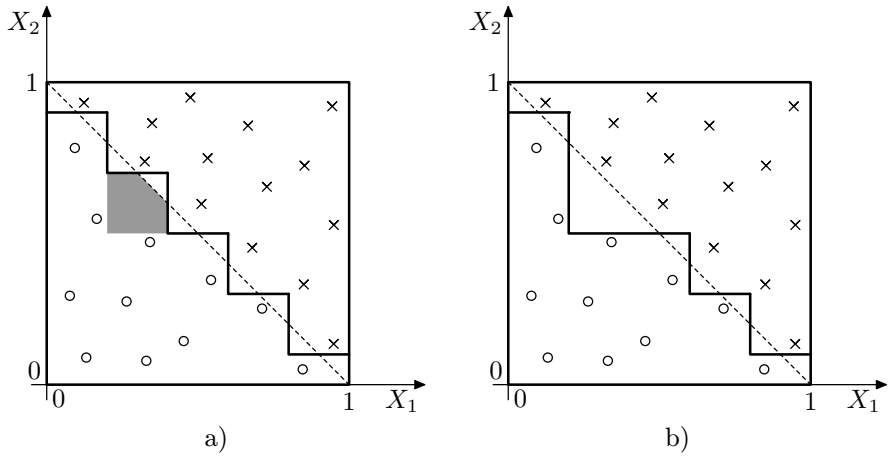
Murphy a Pazzani (1994) studovali klasifikační problém s binární závisle proměnnou Y, která je funkcí pěti binárních prediktorů $X_1, \ldots, X_5$. Množinu všech 32 možných realizací vektoru $(X_1, \ldots, X_5, Y)$ postupně stokrát náhodně rozdělili na trénovací soubor $\mathcal{L}_1$ velikosti 20 a testovací soubor $\mathcal{L}_2$ velikosti 12. Při každém ze 100 opakování vyhledali všechny stromy velikosti nejvýše 10, které klasifikují správně

všechny prvky $\mathcal{L}_1$, a otestovali tyto stromy na $\mathcal{L}_2$. Všechny hodnoty chyby na $\mathcal{L}_2$ pak roztřídili podle velikosti stromu. Podle principu Occamovy břitvy by křivka závislosti průměrné chyby na velikosti měla být neklesající – leč nebyla.

Webb (1996) navrhl algoritmus C4.5X, který je nadstavbou C4.5, jedné z nejpopulárnějších metod konstrukce klasifikačních stromů. Ke stromům vytvořeným programem C4.5 přidává další uzly tak, že se nezmění klasifikace prvků trénovacího souboru. Na reálných i umělých datech ukazuje, že uvedeným zvětšením se generalizační vlastnosti stromů převážně zlepší.

Variaci experimentu s majoritou, o níž referujeme v paragrafu 5.4, lze v souvislosti s Occamovou břitvou chápat jako jistou paralelu Webbova přístupu: Představme si, že výstupem nějaké metody je strom T_1 velikosti m s chybou ε na trénovacích datech. Tatáž trénovací data použijeme ke konstrukci posloupnosti optimálních stromů velikosti $1, \ldots, M$, $M \geq m$. Označme $\mathcal{T}$ množinu všech stromů z uvedené posloupnosti, které mají velikost nejvýše m a chybu na trénovacích datech nejvýše ε. Množina $\mathcal{T}$ je nutně neprázdná. Metodu, jež vyprodukovala strom T_1, bychom mohli „vylepšit“ tím, že T_1 nahradíme stromem T_2, který *nějakým způsobem* vybereme z $\mathcal{T}$. Výsledky v paragrafu 5.5 ukazují, že v případě metody CART bychom tak pro některé klasifikační úlohy v rozporu s principem Occamovy břitvy dostávali převážně stromy s větší skutečnou chybou, ať už má „nějaký způsob“ výběru T_2 z $\mathcal{T}$ jakoukoli konkrétní podobu. (Některé konkrétní verze takového výběru viz Savický a Klaschka, 2001 a 2002 – ve druhém případě se ovšem jedná o stromy regresní, nikoli klasifikační.) Poznamenejme ještě, že se jako argument proti obecné platnosti principu Occamovy břitvy nedají bezprostředně použít výsledky původního nemodifikovaného experimentu z paragrafu 5.4, protože velikosti stromů T_{opt} a T_{cart} nejsou nijak „svázány“, kterýkoli z nich může být větší.

Occamova břitva se možná dá brát jako praktické doporučení, které je často, či dokonce většinou prospěšné. Rozhodně však nepředstavuje princip platný zcela obecně, nezávisle na typu řešené úlohy. Snad nejjasněji o tom svědčí příklad znázorněný na Obr. 4, který ani nepotřebuje numerické experimenty. Představme si, že sdružené rozdělení veličin X_1 a X_2 je rovnoměrné na $[0,1] \times [0,1]$. Nechť $Y = 1$ (křížky na obrázku) pro $X_1 + X_2 > 1$ a $Y = 0$ (kolečka) jinak. Uvažujme jako třídu modelů binární klasifikační stromy, jejichž větvení jsou založena na otázkách „$X_1 < c$?“ a „$X_2 < c$?“, kde c je konstanta mezi 0 a 1. (Naše optimalizační algoritmy takové stromy nekonstruují, ale o to teď nejde.) Konečně velký strom bude diagonální hranici mezi třídami 0 a 1 aproximovat lomenou čarou jako např. na Obr. 4a. Skutečná chyba takového modelu je dána geometricky jako celková plocha trojúhelníkových oblastí, kde lomená čára přesahuje diagonálu (tj. kde nesouhlasí skutečná třída a třída predikovaná modelem). Oblast zvýrazněná na Obr. 4a barvou neobsahuje žádná data, takže lze přejít k modelu na Obr. 4b, který odpovídá menšímu stromu, aniž by se zvětšila chyba na datovém souboru (ta zůstává stejná, totiž nulová). Skutečná chyba tím však evidentně vzroste. Obr. 4 je názornou ukázkou, jak zjednodušení modelu zvýší skutečnou chybu. Kdyby se jednalo o jedinečný jev, k vyvrácení Occamovy břitvy jakožto obecně platného principu by to nestačilo. Podstatné je, že při daném rozdělení vektoru (X_1, X_2, Y) vede takový způsob zmenšování modelu ke zvýšení skutečné chyby *vždy*. Obrátíme-li naopak pozornost ke *zvětšování* stromů, získáme argument ještě přesvědčivější: V modelu s malou skutečnou chybou musí lomená čára, která je hranicí predikovaných tříd, probíhat co nejblíže diagonály, a tedy zákonitě musí mít mnoho zlomů (takže odpovídá velkému stromu). Cestou k co nejmenší skutečné chybě tedy v rozporu s principem Occamovy břitvy není zmenšování modelu, ale naopak jeho zvětšování.



Obr. 4. *Popření principu Occamovy břitvy. Třídy znázorněné kolečky a křížky jsou přesně odděleny diagonálou čtverce. Lomená čára na Obr. 4a reprezentuje klasifikační strom s 9 větveními – 4 podle X_1 a 5 podle X_2. V oblasti zvýrazněné barvou nejsou data, takže chyba na datech se nezvětší přechodem k modelu na Obr. 4b, tj. k lomené čáře, která odpovídá menšímu stromu. Je-li sdružené rozdělení veličin X_1, X_2 na čtverci rovnoměrné, má menší strom větší skutečnou chybu, neboť oblasti vymezené lomenou čarou na Obr. 4b přesahují přes diagonálu větší plochou než na Obr. 4a.*

Domingos (1999) připomíná, že fenomén „overfitu“[4] souvisí jen zprostředkovaně s tím, jak je model velký či složitý: Podstata problému je především v tom, že se model hledá v příliš velké množině kandidátů – čím je jich více, tím pravděpodobnější je, že se vyskytne model, který se náhodně velmi přesně „strefí“ do trénovacích dat (ale ne do skutečného rozdělení). Jedná se vlastně o podobný jev, jako je inflace chyby prvního druhu při testování velkého počtu hypotéz. „Vadou“ složitých modelů z tohoto hlediska není bezprostředně jejich složitost sama o sobě, ale to, že tvoří příliš velkou množinu.

Není divu, že minimalizace chyby na trénovacích datech *při dané velikosti stromu* není dostatečnou prevencí „overfitu“: Optimální strom je výsledkem prohledávání daleko větší množiny kandidátů než stejně velký strom vypěstovaný některou klasickou metodou.

Ještě nás může napadnout námitka, že tendenci optimálních stromů k náhodné shodě s trénovacími daty „hlídají“ validační data. Jistě, odhad skutečné chyby pomocí validačních dat by měl vyloučit extrémní případy a vést k tomu, že se v posloupnosti různě velkých optimálních stromů vytipuje „relativně nejslušnější“ strom. V této fázi se ale vybírá jen z toho, co optimalizační algoritmus nabídne, a to mohou být *pro všechny velikosti* (snad s výjimkou příliš malých) modely s výrazným „overfitem“. Situaci pak pochopitelně nezachrání, ani když v modifikovaném experimentu (viz paragraf 5.5) vyloučíme chyby odhadů založených na validačních datech a vybíráme nejlepší strom na základě přesných hodnot skutečné chyby.

[4]Nemáme po ruce krátký výstižný český ekvivalent, proto nepřekládáme, ale jen opatřujeme uvozovkami.

7. Závěr: Co teď s tím?

Máme ještě v záloze nějaké nápady, jak se pokusit generalizační vlastnosti optimálních stromů zlepšit. Optimálních stromů téže velikosti může být mnoho, takže bychom např. mohli hledat kritéria, podle kterých by optimalizační algoritmy preferovaly „nejnadějnější" exempláře. Nedá se ale očekávat, že by se tak výsledky, jaké jsme viděli v případě úlohy rozpoznávání majority, zcela eliminovaly.

Musíme se tedy smířit s tím, že optimální stromy někde pomáhají, a jinde škodí. Rozeznat úlohy, kde se děje prvé, od těch, kde druhé, je námětem pro budoucí výzkum.

Mimochodem, s „porážkami" optimalizačních algoritmů od klasických metod jsme se dokázali pružně vyrovnat. Hledání pěkných protipříkladů k Occamově břitvě je také výzkumný program. Proběhla řada pracovních telefonátů s klíčovou pasáží: „Je to spočítáno. Jsme opravdu horší." „Tak je to dobrý."

Literatura

[1] Blumer A., Ehrenfeucht A., Haussler D. & Warmuth M. (1987): Occam's razor. *Information Processing Letters* **24**, 377–380.

[2] Breiman L., Friedman J.H., Olshen R.A. & Stone C.J. (1984): *Classification and Regression Trees*. Belmont CA: Wadsworth.

[3] Domingos P. (1999): The role of Occam's razor in knowledge discovery. *Data Mining and Knowledge Discovery* 3 (4), 409-425.

[4] Loh W.-Y. & Shih Y.-S. (1997): Split selection methods for classification trees. *Statistica Sinica* 7, 815-840.

[5] Murphy P.M. & Pazzani M.J. (1994): Exploring the decision forest: An empirical investigation of Occam's razor in decision tree induction. *J. of Art. Int. Res.* 1, 257-275.

[6] Parkinson C.N. (1984): *Nové zákony profesora Parkinsona*. Praha: Mladá fronta.

[7] Quinlan J.R. (1986): Induction of decision trees. *Machine Learning* **1**, 81-106.

[8] Savický P., Klaschka J. & Antoch J. (2000): Optimal classification trees. In: *COMPSTAT 2000, Proceedings in Computational Statistics* (eds. J.G. Bethelem & P.G.M. van der Heiden), 427-432. Heidelberg: Physica-Verlag.

[9] Savický P., Klaschka J. & Antoch J. (2001): Optimální klasifikační stromy. In: *ROBUST'2000, Sborník prací jedenácté letní školy JČMF* (eds. J. Antoch & G. Dohnal), 267-283. Praha: JČMF.

[10] Savický P. & Klaschka J. (2001): Optimally trained classification trees and Occam's razor. Technická zpráva č. 855, Praha: ÚI AV ČR.
`http://www.cs.cas.cz/research/library/reports_800.shtml`

[11] Savický P. & Klaschka J. (2000): Optimally trained regression trees and Occam's razor. In: *COMPSTAT 2002, Proceedings in Computational Statistics*, v tisku.

[12] Webb I.G. (1996): Further experimental evidence against the utility of Occam's razor. *J. of Art. Int. Res.* 4, 397-417.

Ústav informatiky AV ČR, Pod Vodárenskou věží 2, 182 07 Praha
E-mail: `savicky@cs.cas.cz, klaschka@cs.cas.cz`