PRACE NAUKOWE UNIWERSYTETU EKONOMICZNEGO WE WROCŁAWIU
Nr 7 (1207) 2008
TAKSONOMIA 15
Klasyfikacja i analiza danych – teoria i zastosowania

Mariusz Kubus
Politechnika Opolska

ZASTOSOWANIE METODY *BOOSTING* W INDUKCJI REGUŁ

1. Wstęp

Metody indukcji reguł wchodzą w zakres nieparametrycznych metod dyskryminacji o charakterze adaptacyjnym, które można stosować dla zmiennych mierzonych na mocnych i słabych skalach pomiaru. Metody te generują modele w postaci zbioru reguł $\{R_i\}_{i\in\{1,\ldots,r\}}$, gdzie pojedyncza reguła ma postać implikacji:

$$R_i\text{: } koniunkcja\ warunków \rightarrow klasa, \tag{1}$$

co czytamy: *jeśli* obiekt spełnia koniunkcję warunków nakładanych na zmienne, *to* przypisywany jest do odpowiedniej klasy. Reguły są łatwe w interpretacji, gdyż wykorzystują język zbliżony do naturalnego (zob. np. [Gatnar 1998]). Na przykład jeśli składający wniosek kredytowy jest osobą pracującą i celem kredytu jest komputer, to klient jest wiarygodny.

Przez indukcję reguł rozumie się w artykule grupę algorytmów opartych na klasycznym już schemacie separuj-i-zwyciężaj opracowanym przez Michalskiego [1969]. Metody te rozwijały się równolegle z drzewami klasyfikacyjnymi na gruncie komputerowych metod uczenia (*machine learning*). Schemat polega na powtarzaniu dwóch kroków. W kroku „zwyciężaj" generowana jest pojedyncza reguła (1). Dokonuje się tego poprzez heurystyczne przeszukiwanie przestrzeni opisów klas, a więc wszystkich możliwych koniunkcji warunków. Kandydujące opisy oceniane są funkcją kryterium i wybierany jest najlepszy z nich. Następnie w kroku „separuj" usuwane są ze zbioru uczącego obiekty opisane przez wygenerowaną regułę. Na tak zmodyfikowanym zbiorze uczącym generowana jest kolejna reguła itd. Kroki powtarzane są do momentu opisania wszystkich obiektów zbioru uczącego. W algorytmie SLIPPER odejście od tego klasycznego schematu polega na

zastąpieniu kroku „separuj" krokiem polegającym na nadawaniu wag obiektom. Realizuje się to metodą *boosting*.

Metoda *boosting* znana jest głównie z podejścia wielomodelowego (zob. [Breiman 1998; Freund, Schapire 1997; Gatnar 2001]). Została opracowana w odpowiedzi na problem niestabilności drzew klasyfikacyjnych. Aby zmniejszyć zależność jakości predykcji od struktury zbioru uczącego, buduje się wiele modeli składowych na różnych podpróbach zbioru uczącego, a następnie łączy w jeden model zagregowany. Kolejne podpróby powstają w wyniku m-krotnego losowania ze zwracaniem (m – liczba obiektów w zbiorze uczącym), przy czym obiektom nadawane są wagi zależne od klasyfikacji za pomocą modeli składowych. Obiekty błędnie klasyfikowane otrzymują większe wagi, by zwiększyć prawdopodobieństwo ich wylosowania do kolejnej podpróby uczącej. Z kolei Cohen i Singer [1999] w algorytmie SLIPPER stosują *boosting* do budowy pojedynczego modelu w postaci zbioru reguł.

Celem artykułu jest zbadanie wybranych własności algorytmu SLIPPER opracowanego przez Cohena i Singera [1999] oraz porównanie go z dobrze znanym algorytmem drzew klasyfikacyjnych CART.

2. Algorytm SLIPPER

Algorytm SLIPPER (*simple learner with iterative pruning to produce error reduction*) przedstawiony będzie najpierw dla przypadku dyskryminacji dwóch klas. Na początku dane są: zbiór uczący $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_m, y_m) : \mathbf{x}_i \in X,\ y_i \in \{-1, 1\},\ i \in \{1, ..., m\}\}$ oraz jednakowe wagi obiektów $D(i) = 1/m$. Kategorie klas kodowane są liczbami 1 i –1 (1 dla klasy opisywanej). Liczba iteracji może być ustalona arbitralnie lub za pomocą sprawdzania krzyżowego. Dla ustalonej klasy wykonywane są następujące kroki:

1. Generowanie pojedynczej reguły dla aktualnych wag obiektów.
2. Nadanie wagi regule oraz aktualizowanie wag obiektów.

Wagi obiektów aktualizowane są następująco:

$$D_{t+1}(i) = \frac{D_t(i) \cdot e^{-y_i C_{R_t}(i)}}{Z_t}, \tag{2}$$

gdzie: t – numer iteracji,

C_{R_t} – waga reguły wygenerowanej w t-tej iteracji,

Z_t – czynnik normalizujący:

$$Z_t = W_0 + W_+ e^{-C_{R_t}} + W_- e^{C_{R_t}}, \tag{3}$$

gdzie: W_0 – suma wag obiektów nieopisanych przez regułę,

W_+ – suma wag obiektów ustalonej klasy, opisanych przez regułę,

W_- – suma wag pozostałych obiektów opisanych przez regułę.

Schapire i Singer [1998] wykazali, że aby zminimalizować błąd na zbiorze uczącym, należy w każdej iteracji wybierać regułę, dla której wartość Z_t jest zminimalizowana. Warunek ten (pomijając symbol iteracji) można zapisać jako $Z'(C_R) = 0$, co prowadzi do wzoru:

$$C_R = \frac{1}{2} \ln \frac{W_+}{W_-}. \tag{4}$$

Ponieważ może nastąpić sytuacja, że reguła opisuje tylko obiekty ustalonej klasy, wprowadza się modyfikację:

$$\tilde{C}_R = \frac{1}{2} \ln \frac{W_+ + \frac{1}{2m}}{W_- + \frac{1}{2m}}, \tag{5}$$

co ostatecznie ustala wzór na wagę reguły (*confidence in the prediction*). Z kolei wstawiając wartość C_R ze wzoru (4) do (3), można wywnioskować, że minimalizacja Z_t jest równoznaczna z maksymalizacją:

$$\tilde{Z} = \sqrt{W_+} - \sqrt{W_-}. \tag{6}$$

Generowanie pojedynczej reguły może być realizowane jakimkolwiek algorytmem indukcji reguł. Tu wykorzystuje się RIPPER Cohena [1995]. Zmieniona jest jednak funkcja oceny kandydujących reguł. Stosuje się wzór (6), wykorzystując wagi obiektów, a nie jak w klasycznej indukcji reguł – ich liczebności.

Ostatecznie powstaje model dyskryminacyjny w postaci zbioru reguł z wagami, a klasyfikacji dokonuje się na podstawie znaku sumy wag reguł opisujących obiekt:

$$H(x) = sign\left(\sum_{R_t : x \in R_t} \tilde{C}_{R_t} \right), \tag{7}$$

gdzie: $x \in R_t$ oznacza, że obiekt x jest opisany przez regułę R_t.

Istnieją dwa podejścia do konstrukcji opisów klas. Rozwiązują one zarazem zadanie dyskryminacji dowolnej liczby klas k (w szczególności można przyjąć k = 2). Wygenerowane reguły można traktować jako zbiór uporządkowany lub nieuporządkowany. Ma to znaczenie w rozwiązaniu problemu niejednoznaczności rozpoznawania w etapie klasyfikacji. W uporządkowanym zbiorze reguł obiekt jest klasyfikowany przez pierwszą, która go opisze. W nieuporządkowanym zbiorze reguł problem niejednoznaczności rozpoznawania rozstrzyga się metodą głosowania, przy czym w algorytmie SLIPPER wykorzystuje się w tym celu wagi reguł (5). Przejęty z RIPPER sposób konstrukcji uporządkowanego zbioru reguł polega na uporządkowaniu klas według liczebności w zbiorze uczącym (oryginalnie proponuje się uporządkowanie rosnące), a następnie przedstawiony wyżej algorytm wyko-

nuje się $k - 1$ razy, za każdym razem przeciwstawiając i-tej klasie następne w uporządkowaniu, tj. $i + 1$ itd. Oznacza to, że opis ostatniej w uporządkowaniu klasy nie jest budowany. Obiekty są klasyfikowane do niej regułą domyślną, jeśli nie są opisane żadną regułą w modelu. Z kolei konstrukcja nieuporządkowanego zbioru reguł wykorzystuje tradycyjnie stosowany w dyskryminacji schemat przeciwstawiania każdej klasie pozostałych klas. Przedstawiony wcześniej algorytm jest więc powtarzany k razy.

3. Badania empiryczne

Badania przeprowadzono na 5 zbiorach danych udostępnianych przez repozytorium Uniwersytetu Kalifornijskiego [Blake i in. 1998]. Ich krótka charakterystyka przedstawiona jest w tab. 1. Zbiory, które nie miały oryginalnie dołączonego zbioru testowego podzielono losowo na próbę uczącą i testową (1/3 zbioru uczącego). Błąd klasyfikacji szacowano wszędzie na zbiorze testowym.

Tabela 1. Zbiory danych wykorzystane w badaniach

Zbiory	Liczba obiektów	Liczba zmiennych		Liczba klas	Braki danych
		nominalnych	metrycznych		
Adult	48 842	8	6	2	tak
Credit Australian	690	8	6	2	-
Ionosphere	351	-	33	2	-
Lymphography	148	18	-	4	-
Satellite	6 435	-	36	6	-

Źródło: [Blake i in. 1998]].

Najpierw sprawdzono różne podejścia do konstrukcji opisów klas. Stosowano je zarówno w zadaniach dyskryminacji wielu klas, jak i w zadaniach dwóch klas. Liczba iteracji została ustalona za pomocą 5-częściowego sprawdzania krzyżowego (domyślna opcja programu). Uzyskane błędy klasyfikacji oraz złożoność modeli mierzoną liczbą reguł i warunków przedstawiono w tab. 2-3. Jeśli chodzi o uporządkowane zbiory reguł, to mniejsze błędy klasyfikacji otrzymuje się dla rosnącego uporządkowania klas (jest to zgodne z sugestią Cohena i Singera [1999]). Najmniejsze błędy klasyfikacji (na wszystkich zbiorach danych) otrzymano jednak dla nieuporządkowanych zbiorów reguł. Tak konstruowane opisy klas są jednak bardziej złożone, co oznacza, że mają więcej reguł i warunków.

Porównanie algorytmu SLIPPER z popularnym algorytmem drzew klasyfikacyjnych CART zestawiono w tab. 4. Dla uzyskania minimalnego błędu klasyfikacji w CART nie stosowano reguły jednego błędu standardowego (*1SE rule*). Na wszystkich zbadanych zbiorach danych SLIPPER dawał mniejsze błędy klasyfikacji. W zadaniach dyskryminacji wielu klas (*lymphography, satellite*) różnica ta jest szczególnie widoczna. Modele składające się z reguł są jednak o wiele bardziej

złożone, co utrudnia ich interpretację. W przypadku drzew klasyfikacyjnych liczba reguł jest równa liczbie liści.

Tabela 2. Błąd klasyfikacji (w %) szacowany na zbiorze testowym dla różnych wariantów opisu klas, liczba iteracji ustalana za pomocą sprawdzania krzyżowego

Zbiory	Klasy porządkowane:		Nieuporządkowane zbiory reguł
	malejąco	rosnąco	
Adult	15,05	13,4	13,02
Credit Australian	16,52	16,96	12,61
Ionosphere	13,68	9,4	9,4
Lymphography	22,45	16,33	10,2
Satellite	55,85	19,7	11,05

Źródło: obliczenia własne.

Tabela 3. Złożoność modelu (liczba reguł/liczba warunków), liczba iteracji ustalana za pomocą sprawdzania krzyżowego

Zbiory	Klasy porządkowane:		Nieuporządkowane zbiory reguł
	malejąco	rosnąco	
Adult	39/170	55/260	102/449
Credit Australian	9/27	18/53	19/59
Ionosphere	11/35	13/29	23/57
Lymphography	14/35	18/43	34/91
Satellite	131/461	94/369	162/716

Źródło: obliczenia własne.

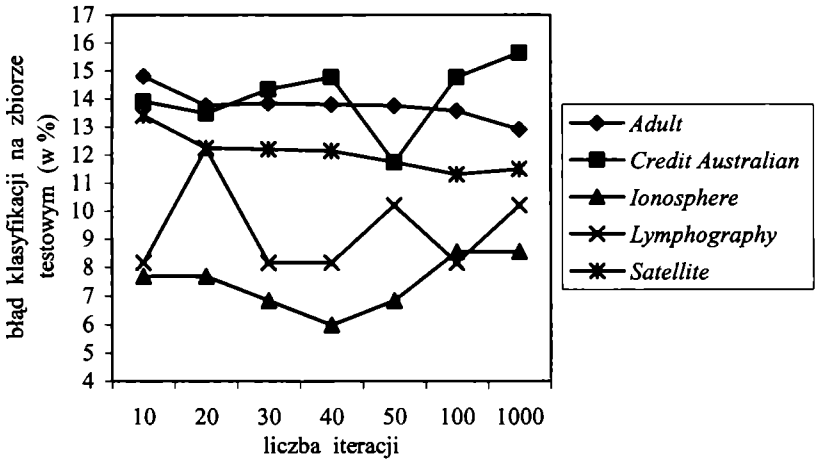
Tabela 4. Błąd klasyfikacji (w %) szacowany na zbiorze testowym i złożoność modelu (mierzona liczbą reguł) w SLIPPER oraz CART

Zbiory	Błąd klasyfikacji		Złożoność modelu	
	CART	SLIPPER	CART	SLIPPER
Adult	15,55	13,02	5	102
Credit Australian	15,2	12,61	2	19
Ionosphere	12	9,4	4	23
Lymphography	24,5	10,2	5	34
Satellite	33,55	11,05	83	162

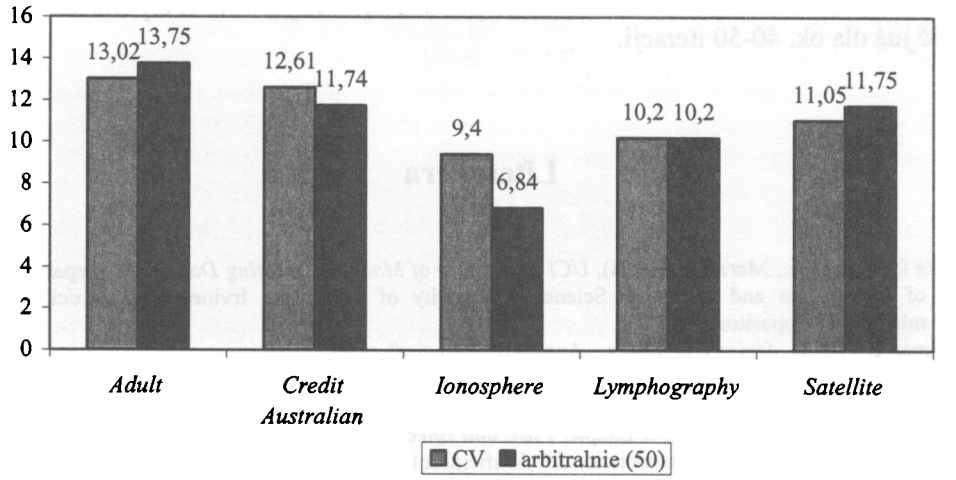
Źródło: obliczenia własne.

Następnie zbadano wpływ (arbitralnie ustalanej) liczby iteracji na błąd klasyfikacji. Zgodnie z intuicją im większa liczba iteracji, tym model jest bardziej złożony (większa liczba reguł) oraz bardziej dopasowany do danych (maleje błąd resubstytucji). Nie oznacza to jednak dokładnej klasyfikacji nowych obiektów spoza zbioru uczącego. Ustalenie niezbyt dużej liczby iteracji może być więc postrzegane jako zatrzymywanie procesu budowy modelu, a więc jako technika upraszczania wstępnego (*pre-pruning*). Podstawowym problemem tego podejścia jest ustalenie dobrego kryterium stopu. Przeprowadzone badania nie dają jednoznacznej odpowiedzi

(zob. rys. 1). Wydaje się, że korzystna liczba iteracji to ok. 40-50, lecz w dużych zbiorach danych (*adult, satellite*) błąd szacowany na zbiorze testowym malał jeszcze przy 100 iteracjach, a nawet przy 1000! Z drugiej strony należy zaznaczyć, że spadek błędu klasyfikacji był na tych zbiorach bardzo wolny. Na rys. 2 pokazano dla porównania błędy klasyfikacji dla arbitralnie ustalonej liczby iteracji równej 50 oraz dla liczby iteracji wybranej za pomocą sprawdzania krzyżowego.



Rys. 1. Wpływ liczby iteracji na błąd klasyfikacji
Źródło: opracowanie własne.



Rys. 2. Błędy klasyfikacji (w %) szacowane na zbiorach testowych dla arbitralnie ustalanej liczby iteracji oraz ustalanej za pomocą sprawdzania krzyżowego (CV)
Źródło: opracowanie własne.

Na mniejszych zbiorach danych (*credit australian, ionosphere, lymphography*) uzyskano w przypadku arbitralnego doboru liczby iteracji błędy klasyfikacji nie większe niż za pomocą sprawdzania krzyżowego. Na nieco większych zbiorach danych (*adult, satellite*) zastosowanie sprawdzania krzyżowego zapewniło nieco mniejszy błąd, ale coraz większe znaczenie ma wówczas czas obliczeń. Przykładowo na zbiorze *adult* (32 561 obiektów w próbie uczącej) ustalenie optymalnej liczby iteracji (sprawdzono liczby od 1 do 100) i zbudowanie modelu trwało 5 min i 6 sek, podczas gdy dla ustalonej arbitralnie liczby iteracji (50): 32 sek. (procesor Pentium-D 945, 3,4 GHz oraz 1GB RAM).

4. Podsumowanie

Algorytm SLIPPER jest zasługującym na uwagę narzędziem dyskryminacji obiektów opisanych zmiennymi mierzonymi na dowolnej skali pomiaru. Daje mniejsze błędy klasyfikacji od powszechnie uznawanego i często stosowanego algorytmu drzew klasyfikacyjnych CART. Modele zbudowane z reguł cechują się wprawdzie dużo większą złożonością od drzew klasyfikacyjnych, co utrudnia interpretację, lecz wobec różnicy w błędach klasyfikacji (zwłaszcza w zadaniach dyskryminacji wielu klas) są konkurencyjne.

Stosowanie algorytmu SLIPPER prowadzące do nieuporządkowanych zbiorów reguł daje modele zapewniające mniejsze błędy klasyfikacji, choć są one bardziej złożone i wymagają dłuższego czasu generowania. Parametrem metody jest liczba iteracji. Arbitralne ustalenie jego wartości, optymalnej z punktu widzenia minimalizacji błędu klasyfikacji, nabiera znaczenia dla dużych zbiorów danych. Na podstawie przeprowadzonych badań wydaje się, że zadowalające rezultaty można uzyskać już dla ok. 40-50 iteracji.

Literatura

Blake C., Keogh E., Merz C.J. (1998), *UCI Repository of Machine Learning Databases*, Department of Information and Computer Science, University of California, Irvine, www.ics.uci.edu/~ mlearn /MLRepository.html.

Breiman L. (1998), *Arcing Classifiers*, „Annals of Statistics" nr 26.

Cohen W.W. (1995), *Fast Effective Rule Induction,* [w:] Proceedings of the 12th International Conference on Machine Learning, red. A. Prieditis, S. Russell.

Cohen W.W., Singer Y. (1999), *A Simple, Fast, and Effective Rule Learner*, Proceedings of Annual Conference of American Association for Artificial Intelligence, s. 335-342.

Freund Y., Schapire R. E. (1997), *A Decision-theoretic Generalization of On-line Learning and an Application to Boosting*, „Journal of Computer and System Sciences" nr 55.

Gatnar E. (1998), *Symboliczne metody klasyfikacji danych*, Wydawnictwo Naukowe PWN, Warszawa.

Gatnar E. (2001), *Nieparametryczna metoda dyskryminacji i regresji*, Wydawnictwo Naukowe PWN, Warszawa.

Michalski R.S. (1969), *On the Quasi-Minimal Solution of the Covering Problem*, Proceedings of the 5th International Symposium on Information Processing (FCIP-69), vol. A3 (Switching Circuits), Bled, Yugoslavia, s. 125-128.

Schapire R.E., Singer Y. (1998), *Improved Boosting Algorithms Using Confidence-rated Predictions*, Proceedings of the Eleventh Annual Conference on Computational Learning Theory, s. 80-91.

BOOSTING APPLICATION IN RULES INDUCTION

Summary

The induction of classification rules in the implication form is an alternative to classification trees in a presence of non-metric variables. Many algorithms of rules induction follow the classical manner *separate-and-conquer* [Michalski 1969]. Cohen and Singer [1999] proposed to use boosting instead of the step *"separate"* and implemented it in SLIPPER.

The goal of this paper is to test the selected properties of SLIPPER algorithm and compare it with popular CART.