EKONOMETRIA ECONOMETRICS 1(55) • 2017

ISSN 1507-3866
e-ISSN 2449-9994

Mariusz Łapczyński
Uniwersytet Ekonomiczny w Krakowie
e-mail: lapczynm@uek.krakow.pl

O MOŻLIWOŚCIACH WYKORZYSTANIA ROTACYJNEGO LASU W BADANIACH RYNKOWYCH I MARKETINGOWYCH

THE POSSIBILITY OF USE OF ROTATION FOREST IN MARKETING SURVEYS

DOI: 10.15611/ekt.2017.1.06
JEL Classification: C52, C53, M31

Streszczenie: Rotacyjny las (*rotation forest*) jest narzędziem analitycznym służącym do budowy zagregowanych modeli predykcyjnych. Pojedyncze modele drzew klasyfikacyjnych powstają na podstawie podprób bootstrapowych, a do ich budowy używa się innych zbiorów zmiennych niezależnych. Początkowo dzieli się zbiór tych zmiennych na *k* rozłącznych podzbiorów, a następnie w każdym z nich stosuje się analizę głównych składowych w celu uzyskania liniowej kombinacji zmiennych wejściowych. Celem artykułu jest porównanie skuteczności modeli prognostycznych zbudowanych za pomocą rotacyjnego lasu z innymi modelami zagregowanymi: metodą *bagging*, drzewami wzmacnianymi AdaBoost i losowym lasem. Do analiz wykorzystano 11 zbiorów obserwacji pobranych z popularnego repozytorium *on-line*. Obliczenia zostały wykonane w programie WEKA (*Waikato Environment for Knowledge Analysis*), a ocena modeli została dokonana za pomocą czterech miar: dokładności, czułości, precyzji i miary *F*. Wyniki wskazują na ograniczone możliwości wykorzystania tego modelu zagregowanego w badaniach rynkowych i marketingowych. Najważniejsze przeszkody dotyczą poziomu pomiaru zmiennych niezależnych i zasobów sprzętowych niezbędnych do analizy dużych zbiorów danych.

Słowa kluczowe: badania marketingowe, predykcja, podejście wielomodelowe, rotacyjny las.

Summary: Rotation forest is an example of ensemble model that combines decision trees with principal component analysis. Single decision trees are based on bootstrap subsamples and different sets of independent variables (rotated feature space). The goal of this article is to compare the effectiveness of predictive models built by using rotation forest with other models based on bagging, Adaboost and random forest. Data sets used in this analysis come from popular UCI Machine Learning repository. Predictive models were built by open source WEKA software. The author used two algorithms of decision trees (J48 and SimpleCart) and modified the type of split (binary/not binary) and pruning options. The performance of models was evaluated by using popular measures based on misclassification matrix: accuracy, recall, precision and F-measure. The results indicate a limited possibility of using this algorithm in

marketing. The main obstacles relate to the measurement scale of independent variables and computer power required to analyze large data sets.

Keywords: marketing survey, prediction, ensemble models, rotation forest.

1. Wstęp

Łączenie modeli i narzędzi analitycznych jest obecnie powszechną praktyką podczas budowy modeli prognostycznych w wielu dziedzinach i obszarach badawczych. Większa czasochłonność związana z analizą danych jest bardzo często kompensowana wyższą trafnością predykcji, redukcją problemu niezbilansowanych prób lub dostarczeniem czytelnych wzorców ze zbiorów danych o złożonej strukturze. Celem niniejszego artykułu jest prezentacja algorytmu rotacyjny las, który jest przykładem modelu zagregowanego – łączącego wiele modeli bazowych (składowych).

E. Gatnar wyróżnia trzy sposoby łączenia modeli: równoległe, szeregowe i hybrydowe [Gatnar 2008], i to właśnie architektura równoległa odnosi się do rotacyjnego lasu. Wyższa trafność predykcji modelu zagregowanego jest zwykle wynikiem zróżnicowania (*diversity*) modeli bazowych, którą uzyskuje się poprzez:

- losowanie prób uczących o różnej zawartości obiektów – każdy model bazowy powstaje na podstawie innego zbioru (chociaż w części) przypadków;
- stosowanie różnych zestawów zmiennych niezależnych dla każdego modelu składowego;
- przekształcenie zmiennej zależnej, tak jak w metodzie *twicing* lub ECOC (*Error-Correcting Output Coding*);
- używanie różnych ustawień (parametrów) modeli bazowych;
- budowę modeli składowych przy użyciu różnych narzędzi analitycznych;
- budowę modeli bazowych przy użyciu metod niestabilnych, np. za pomocą drzew klasyfikacyjnych lub sieci neuronowych.

Rotacyjny las był dosyć często wykorzystywany w analizie danych medycznych, m.in. do prognozowania zachorowań na raka piersi [Aličković, Subasi 2015], do przewidywania zwijania się białek [Dehzangi i in. 2010], w badaniach dotyczących zniesienia czucia bólu [Hu i in. 2012], w diagnozowaniu chorób wieńcowych [Karabulut, İbrikçi 2012] czy w rozpoznawaniu choroby Parkinsona [Ozcift 2012]. Inne ciekawe zastosowania dotyczyły rozpoznawania rodzajów i marek pojazdów na podstawie danych z monitoringu [Zhang, Zhou 2012], diagnozowania usterek turbin wiatrowych [Santos i in. 2012] czy identyfikowania niebezpiecznych gazów w systemach kanalizacyjnych [Ojha, Dutta, Chaudhuri 2016]. Na gruncie szeroko rozumianej ekonomii rotacyjny las był stosowany do prognozowania cen nieruchomości [Lasota, Łuczak, Trawiński 2012] i prognozowania migracji klientów [Idris, Khan, Lee 2013; De Bock, Van den Poel 2011].

2. Charakterystyka rotacyjnego lasu

Rotacyjny las (*rotation forest*) to podejście wielomodelowe, stosowane w prognozowaniu zmiennych jakościowych i ilościowych, które w pewnych elementach jest podobne do procedury popularnego losowego lasu (*random forest*). Pojedyncze modele składowe (bazowe) to drzewa klasyfikacyjne albo regresyjne, do budowy których wykorzystuje się wszystkie przypadki z próby uczącej z rotowaną przestrzenią zmiennych objaśniających.

Na wstępie przyjmuje się, że próba[1] ucząca zawiera N obserwacji, n zmiennych objaśniających i zmienną zależną Y (jakościową albo ilościową) [Rodriguez, Kuncheva, Alonso 2006]. Dla każdego modelu składowego (1, 2, 3, …, L):

1) dzieli się próbę uczącą na K podzbiorów, w taki sposób, żeby każdy podzbiór zawierał wszystkie przypadki (N) i podzbiór M ($M = n / K$) zmiennych objaśniających (podzbiory zmiennych objaśniających zazwyczaj są rozłączne; chociaż nie jest to warunek konieczny do przeprowadzenia analizy, trzeba dodać, że rozłączność zapewnia większą różnorodność modeli składowych);

2) w każdym podzbiorze eliminuje się losowo przypadki należące do niektórych kategorii zmiennych zależnych, jest to tożsame z redukcją liczby przypadków w tych podzbiorach;

3) w kolejnym kroku losuje się podpróbę bootstrapową o liczebności 75% z każdego podzbioru próby uczącej zredukowanego w kroku nr 2;

4) w każdej podpróbie bootstrapowej przeprowadza się analizę głównych składowych w taki sposób, żeby liczba głównych składowych była równa liczbie zmiennych objaśniających, pierwotne zmienne zostają zastąpione głównymi składowymi, które są ich linowymi kombinacjami;

5) informację o głównych składowych ze wszystkich K podprób bootstrapowych wykorzystuje się do oryginalnej próby uczącej zawierającej wszystkie n zmiennych niezależnych, buduje się model składowy (np. drzewo klasyfikacyjne), wykorzystując wszystkie N obserwacji z próby uczącej i zbiór głównych składowych (nowych zmiennych objaśniających).

---

[1] Wielu autorów używa określeń „zbiór uczący" (z którego można losować próby bootstrapowe) i „zbiór testowy". Terminy „próba ucząca" i „próba testowa" mogą budzić wątpliwości, ponieważ kojarzone są z próbą pobieraną losowo z populacji, co nie zawsze ma miejsce w przypadku analizy danych wtórnych. Autor zdecydował się używać określeń „próba ucząca", „próba testowa" i „podpróba bootstrapowa" z dwóch powodów. Po pierwsze, w literaturze zachodniej część autorów używa określeń *learning sample*, a część *training set* oraz konsekwentnie *bootstrap subsample* i *bootstrap sample*. Po drugie, dosyć często zbiór obserwacji, który jest dzielony na podzbiory, do budowy i testowania modelu bywa losowany z dużego zbioru danych, więc można by uznać, że już na wstępie może on być traktowany jako próba losowa. Taka sytuacja ma miejsce w niektórych zbiorach obserwacji wykorzystanych w tych badaniach (są próbą pobraną losowo z większego zbioru danych) i, zdaniem autora, będzie występować coraz częściej w kontekście analizy dużych zbiorów danych (*big data*), gdzie problemem jest niewystarczająca moc obliczeniowa komputerów, a w konsekwencji konieczność redukcji liczby obserwacji.

Zaletą rotacyjnego lasu jest – podobnie jak w przypadku losowego lasu – wykorzystywanie podzbioru zmiennych objaśniających podczas budowy modelu składowego. Skraca to czas obliczeń zwłaszcza w zbiorach obserwacji z dużą liczbą zmiennych niezależnych. Liczba wszystkich możliwych podziałów (T) zbioru zmiennych objaśniających n na K podzbiorów o liczebności M zmiennych wynosi:

$$T = \frac{n!}{K!(M!)^K}, \quad [1]$$

co dla 10 zmiennych daje 945 możliwych podziałów, a dla 14 zmiennych już 135 135 podziałów. Jest to bez wątpienia ten parametr modelu, który zwiększa różnorodność modelu zagregowanego. Należy jednak w tym miejscu dodać, że prawdopodobieństwo, iż wszystkie modele składowe będą różne, wynosi:

$$P(\text{różne modele składowe}) = \frac{T!}{(T-L)!T^L}, \quad [2]$$

co przy liczbie modeli składowych L większej niż 30 oraz liczbie predyktorów równej 8 powoduje, że wynik jest bliski zeru. Innymi słowy, jest prawie niemożliwe, aby wszystkie modele składowe były różne. To ograniczenie skłoniło autorów algorytmu do zastosowania podrób bootstrapowych o liczebności 75%, które obok podziału zbioru cech zwiększają różnorodność modeli bazowych.

Różnorodność jest jednym z wymogów, jakie stawia się modelom zagregowanym, ponieważ łączenie modeli o podobnych właściwościach (liczbie analizowanych przypadków, liczbie analizowanych zmiennych, postaci analizowanych zmiennych itp.) jest bezcelowe [Rodriguez, Alonso 2004]. Drugi wymóg, jaki powinien być spełniony w podejściu wielomodelowym, to dokładność (*accuracy*), jaką powinny charakteryzować się modele składowe. W przypadku rotacyjnego lasu wykorzystuje się wszystkie główne składowe (wyjaśnia się całkowitą wariancję), a zatem nie ma utraty informacji, co z kolei pozwala przyjąć, że wynik agregacji nie powinien być gorszy od wyniku uzyskanego za pomocą modelu bazowego.

Włączenie analizy głównych składowych do procedury rotacyjnego lasu powoduje pewne ograniczenie, jakim jest możliwość wykorzystania wyłącznie ilościowych zmiennych niezależnych. Wprawdzie zdaniem twórców metody w przypadku zmiennych dyskretnych należy zamienić je na ciągłe (co oznacza zapewne utworzenie zmiennych sztucznych), jednak należy pamiętać, że może to budzić sporo wątpliwości. Pamiętając o założeniach klasycznej analizy głównych składowych, można przypuszczać, że lepiej użyć wielowymiarowej analizy korespondencji, odmiany PCA dla zmiennych kategorialnych (*Categorical Principal Components Analysis*, CATPCA, skalowanie optymalne) albo wykorzystać korelacje tetrachoryczne w tradycyjnej wersji PCA.

3. Inne modele zagregowane wykorzystane w badaniach

Modele zbudowane za pomocą rotacyjnego lasu zostały porównane z innymi popularnymi modelami zagregowanymi: podejściem *bagging*, algorytmem AdaBoost

i losowym lasem. Skrót *bagging* pochodzi od angielskich terminów *bootstrap aggregating* [Breiman 1994] i oznacza agregację bootstrapową, czyli łączenie wielu modeli zbudowanych na podstawie różnych podprób bootstrapowych. Schemat postępowania jest następujący:

1) z próby uczącej L o liczebności n losuje się w sposób prosty niezależny (ze zwracaniem) próby liczące n przypadków, powstają w ten sposób podpróby bootstrapowe (L_1, L_2, … L_k); na podstawie każdej z nich budowany jest pojedynczy model;

2) wynik predykcji to w przypadku zmiennych zależnych jakościowych wynik głosowania modeli składowych, natomiast w przypadku zmiennych zależnych ilościowych – wynik uśredniania.

Algorytm AdaBoost [Freund, Shapire 1997] służy do budowy modelu zagregowanego w architekturze szeregowej. Jest to skrót od angielskich terminów *adaptive boosting*, co tłumaczy się jako adaptacyjne wzmacnianie lub poprawianie wyniku predykcji. W pierwszym kroku powstaje model, który zwykle dosyć słabo klasyfikuje obiekty (*weak classifier*). Po ocenie trafności predykcji nadaje się przypadkom z próby uczącej wagi. Te przypadki, które zostały błędnie sklasyfikowane, otrzymują wyższe współczynniki wagowe, zaś te, które zostały poprawnie sklasyfikowane, otrzymują niższe współczynniki wagowe. Po wprowadzeniu wag dokonuje się ponownego losowania próby uczącej i na jej podstawie buduje kolejny model. Oznacza to, że postać i skuteczność kolejnych modeli predykcyjnych jest uzależniona od wyników modeli poprzednich, ponieważ to one decydują o współczynnikach wagowych przypisywanych obiektom z próby uczącej. Ostateczny wynik predykcji to wynik ważonego głosowania. Nie jest to głosowanie większościowe (*majority voting*) jak w procedurze *bagging*, ponieważ każdy kolejny model charakteryzuje się coraz lepszą trafnością predykcji.

Idea losowego lasu[2] oparta jest na algorytmie drzew klasyfikacyjnych CART. W trakcie analizy budowane jest wiele pojedynczych drzew [Breiman 2001], które ostatecznie klasyfikują nowy obiekt do jednej z klas – wariantu zmiennej objaśnianej. Klasyfikowanie obiektów przez pojedyncze modele bywa określane terminem „głosowanie" (*voting*) i oznacza, że rozpoznawany obiekt trafia do klasy, która została wskazana przez większość modeli. W przypadku modeli regresyjnych wynikiem prognozy jest uśredniony wynik z modeli składowych. Procedura analityczna składa się z trzech etapów [Breiman, Cutler 2007]:

1) z próby uczącej L o liczebności n losuje się podpróby bootstrapowe tak samo jak w procedurze *bagging*;

---

[2] Pierwotnie autor algorytmu – L. Breiman – użył określenia *random forests*, co w dosłownym tłumaczeniu oznacza losowe lasy. Autor zdecydował się na używanie określenia losowy las w liczbie pojedynczej, ponieważ zgodnie ze *Słownikiem języka polskiego* las to zwarty zespół roślinności z przewagą drzew. Przemawia za tym również fakt, że jeden model składa się z wielu drzew, a zatem o losowych lasach można by mówić wówczas, gdyby utworzyć kilka modeli zagregowanych. Termin „las" w liczbie pojedynczej jest również używany w programie STATISTICA, w pakiecie *R* (randomForest) oraz w programie Random Forest firmy Salford Systems.

2) z całego zbioru *M* zmiennych objaśniających losuje się zbiór *m* zmiennych (gdzie $m < M$); losowanie to przeprowadza się na każdym etapie podziału pojedynczych modeli, przy czym ustalona na początku wartość *m* pozostaje niezmieniona przez cały czas trwania analizy;

3) każdy model budowany jest do możliwie maksymalnie dużych rozmiarów (pomimo że używa się tu algorytmu CART, to jednak nie korzysta się z opcji przycinania).

4. Opis zbiorów obserwacji

W badaniach wykorzystano 11 zbiorów obserwacji dostępnych *on-line* w popularnym repozytorium *UCI Machine Learning Repository*. Pierwszy zbiór wykorzystany w badaniach (*balance scale*) zawiera wyniki eksperymentu psychologicznego. Zmienna zależna dotyczy skali równowagi i przyjmuje trzy warianty: równowaga, odchylenie w lewo i odchylenie w prawo. Dane z drugiego zbioru (*banknote authentication*) zostały wyodrębnione ze zdjęć oryginalnych i fałszywych banknotów. Fotografie wykonano kamerą przemysłową w odcieniach szarości, a ostateczne kopie miały rozdzielczość 400 na 400 pikseli. Zmienne objaśniające to wariancja, skośność, kurtoza i entropia zdjęć uzyskanych na drodze transformacji falkowej (*wavelet*), a zmienna objaśniana to przynależność do jednej z dwóch kategorii: „fałszywy" albo „oryginalny". W kolejnym zbiorze obserwacji (*glass identification*) jakościowa zmienna zależna informowała o rodzaju szkła (okienne, z szyb samochodowych, z reflektorów samochodowych, z mebli itp.), a 9 ilościowych zmiennych niezależnych odnosiło się do zawartości substancji chemicznych w szkle (m.in. żelazo, glin magnez, silikon, wapń) oraz do współczynnika załamania światła.

W kolejnym zbiorze obserwacji (*ionosphere*) wykorzystanym w badaniu zmienna zależna odnosi się do stanu jonosfery (*bad* / *good*), natomiast 34 niezależne zmienne ilościowe do pomiarów uzyskanych z radarów umieszczonych na półwyspie Labrador w Kanadzie. Zbiór liczy 351 przypadków. Piąty zbiór obserwacji (*iris*) dotyczy kwiatów irysa (kosaćca). Zmienna zależna to odmiana kwiatu (kosaciec szczecinkowy – *setosa*, kosaciec wirginijski – *virginica* i kosaciec różnobarwny – *versicolor*), zmienne niezależne zaś to dane o długości i szerokości płatka korony – *petal* oraz o długości i szerokości działki kielicha – *sepal* (wszystkie metryczne). W zbiorze danych *seeds* zmienna zależna odnosi się do ziaren trzech odmian pszenicy (Kama, Rosa, Kanadyjska), a zmienne niezależne do wyników badania struktury jądra tych ziaren za pomocą promieni rentgenowskich, w tym m.in. obwodu, długości, zwartości, szerokości czy współczynnika asymetrii. Wszystkie zmienne objaśniające są zmiennymi ciągłymi.

Następny zbiór obserwacji (*segment*) zawiera informacje o fotografiach, które pogrupowano w segmenty na podstawie 19 ilościowych zmiennych niezależnych, wśród których znajdowały się m.in.: nasycenie kolorów, odcień, intensywność koloru czerwonego (R), zielonego (G) lub niebieskiego (B). W ósmym zbiorze da-

nych – *sonar* – dwuwariantowa zmienna zależna odnosi się do skały (*rock*) i metalowego cylindra, natomiast zbiór sześćdziesięciu ilościowych zmiennych niezależnych dotyczy sygnałów wysyłanych przez sonar pod różnymi kątami i w różnych warunkach. Następny zbiór obserwacji (*spambase*) odnosi się do zjawiska spamu, czyli wysyłania niechcianych wiadomości za pośrednictwem poczty elektronicznej. Binarna zmienna zależna przyjmuje dwie kategorie (spam/nie spam), a 57 ilościowych zmiennych niezależnych dotyczy częstotliwości występowania określonych słów, np. *free, business, credit, money, original* itp. Zbiór danych *vehicle* ma czterowariantową zmienną zależną o dosyć niespójnych kategoriach: opel, saab, bus i van. Zbiór ilościowych zmiennych niezależnych obejmuje wymiary pojazdów (długość, promień skrętu, proporcje osi itp.). Ostatni zbiór danych – *wine* – dotyczy win otrzymywanych z trzech odmian winogron występujących w jednym z regionów Włoch. Zmienne niezależne to wyniki analizy chemicznej (zmienne ilościowe), które informują m.in. o zawartości alkoholu, magnezu, kwasu jabłkowego, związków polifenolowych, pochodnych flawinowych itp. Zmienna zależna to 3-wariantowa zmienna kategorialna odnosząca się do klasy wina.

Tabela 1. Charakterystyka zbiorów danych

Zbiór obserwacji	Liczba zmiennych niezależnych	Liczba kategorii zmiennej zależnej	Najrzadziej występująca kategoria	Liczba przypadków
balance scale	4	3	B	625
banknote	4	2	„1”	1372
glass	9	6	tableware	214
ionosphere	34	2	b	351
iris	4	3	wszystkie po 50	150
seeds	7	3	wszystkie po 70	210
segment	19	7	wszystkie po 330	2310
sonar	60	2	rock	208
spambase	57	2	„1”	4601
vehicle	18	4	van	846
wine	13	3	„3”	178

Źródło: opracowanie własne.

W tabeli 1 zestawiono podstawowe informacje o zbiorach obserwacji wykorzystanych w badaniach. Wskazano w niej liczbę zmiennych niezależnych, liczbę kategorii zmiennej zależnej, liczbę przypadków oraz nazwę kategorii zmiennej zależnej, która występowała najrzadziej (tę, którą potencjalnie najtrudniej opisać lub przewidzieć). W tym ostatnim przypadku intencją autora było ujednolicenie kryteriów oceny modeli. W modelach dyskryminacyjnych jakość rozwiązania oceniana

na podstawie macierzy błędnych klasyfikacji zależy w niektórych miernikach od tego, która kategoria zmiennej objaśnianej jest brana pod uwagę. Jak łatwo zauważyć, wszystkie zbiory obserwacji mają wyłącznie ilościowe zmienne objaśniające.

5. Wyniki analiz

Modele rotacyjnego lasu oraz pozostałe modele zagregowane (*bagging*, *AdaBoostM1* i losowy las) zostały zbudowane za pomocą programu WEKA. Każdy zbiór danych został podzielony na próbę (zbiór) uczącą (66%) i próbę (zbiór) testową (34%). W analizie uwzględniono następujące modyfikacje algorytmu:

- połączono analizę głównych składowych z odpowiednikiem algorytmu C4 (J48) i odpowiednikiem algorytmu CART (SimpleCart);
- w ramach algorytmów drzew klasyfikacyjnych stosowano przycinanie lub nie stosowano przycinania;
- w ramach algorytmów drzew klasyfikacyjnych stosowano podziały binarne lub dowolne (tylko w J48);
- modyfikowano liczbę modeli składowych (10, 20, 30, 40, 50, 100 i 200).

Do oceny jakości rozwiązania wykorzystano popularne miary oparte na macierzy błędnych klasyfikacji, jak dokładność, czułość (określana w literaturze *data mining* terminem *recall*), precyzja i miara F (*F-measure*)[3]. W czterech kolejnych tabelach (2-5) przedstawiono szczegółowe wyniki dla zbioru danych *vehicle*. Dla

Tabela 2. Wartości miary dokładności (*accuracy*) na przykładzie zbioru *vehicle*

Model zagregowany	Liczba modeli składowych						
	10	20	30	40	50	100	200
RL (J48 binarne nieprzycięte)	0,806	0,819	0,788	0,816	0,809	0,823	0,802
RL (J48 dowolne nieprzycięte)	0,806	0,819	0,788	0,816	0,809	0,823	0,802
RL (J48 binarne przycięte)	0,806	0,819	0,795	0,816	0,809	0,814	0,799
RL (J48 dowolne przycięte)	0,806	0,819	0,795	0,816	0,809	0,814	0,799
RL (SimpleCart przycięte)	0,819	0,785	0,813	0,802	0,781	0,809	0,781
RL (SimpleCart nieprzycięte)	0,826	0,809	0,826	0,816	0,802	0,799	0,799
Bagging	0,719	0,712	0,736	0,719	0,733	0,740	0,747
AdaBoostM1	0,771	0,795	0,792	0,788	0,809	0,795	0,792
Losowy las	0,743	0,736	0,754	0,757	0,764	0,760	0,764

Źródło: opracowanie własne przy użyciu programu WEKA.

[3] Posługując się popularnymi skrótami z macierzy błędnych klasyfikacji: TP (prawdziwie pozytywne), FP (fałszywie pozytywne), TN (prawdziwe negatywne) i FN (fałszywie negatywne), można zapisać wzory miar jakości w następujący sposób: dokładność = ((TP+TN)/(TP+FP+TN+FN)), czułość = (TP/(TP+FN)), precyzja = (TP/(TP+FP)) i miara F = (2 x precyzja x czułość) / (precyzja + czułość).

ułatwienia przeglądania wyników najlepsze rozwiązania wyróżniono odcieniami szarości, a najgorsze podkreślono. Skrót RL użyty w pierwszej kolumnie tabel oznacza rotacyjny las.

Z danych zamieszczonych w tab. 2-5 wynika, że trudno jest wskazać najlepszą kombinację parametrów rotacyjnego lasu. W przypadku dokładności modeli najlepsze rozwiązanie uzyskano za pomocą rotacyjnego lasu już po zbudowaniu 10 modeli bazowych. Warto podkreślić, że użyto innego algorytmu drzewa klasyfikacyjnego, niż proponowali to autorzy metody (SimpleCart zamiast J48). Drzewo nie-

Tabela 3. Wartości miary czułości (*recall*) na przykładzie zbioru *vehicle*

Model zagregowany	Liczba modeli składowych						
	10	20	30	40	50	100	200
RL (J48 binarne nieprzycięte)	0,986	1,000	0,986	0,972	0,986	1,000	0,986
RL (J48 dowolne nieprzycięte)	0,986	1,000	0,986	0,972	0,986	1,000	0,986
RL (J48 binarne przycięte)	0,986	1,000	0,986	0,986	0,986	1,000	0,986
RL (J48 dowolne przycięte)	0,986	1,000	0,986	0,986	0,986	1,000	0,986
RL (SimpleCart przycięte)	0,986	1,000	1,000	1,000	1,000	0,986	1,000
RL (SimpleCart nieprzycięte)	0,986	1,000	0,986	0,986	0,986	0,986	0,986
Bagging	0,903	0,930	0,930	0,930	0,930	0,944	0,930
AdaBoostM1	0,930	0,944	0,958	0,944	0,958	0,958	0,915
Losowy las	0,972	0,944	0,918	0,958	0,958	0,972	0,972

Źródło: opracowanie własne przy użyciu programu WEKA.

Tabela 4. Wartości miary precyzji (*precision*) na przykładzie zbioru *vehicle*

Model zagregowany	Liczba modeli składowych						
	10	20	30	40	50	100	200
RL (J48 binarne nieprzycięte)	0,946	0,959	0,933	0,945	0,946	0,973	0,959
RL (J48 dowolne nieprzycięte)	0,946	0,959	0,933	0,945	0,946	0,973	0,959
RL (J48 binarne przycięte)	0,946	0,959	0,933	0,946	0,959	0,959	0,959
RL (J48 dowolne przycięte)	0,946	0,959	0,933	0,946	0,959	0,959	0,959
RL (SimpleCart przycięte)	0,909	0,899	0,947	0,910	0,899	0,946	0,899
RL (SimpleCart nieprzycięte)	0,959	0,922	0,972	0,946	0,946	0,959	0,946
Bagging	0,880	0,868	0,892	0,917	0,930	0,918	0,930
AdaBoostM1	0,880	0,944	0,944	0,957	0,958	0,944	0,942
Losowy las	0,920	0,918	0,944	0,919	0,907	0,932	0,920

Źródło: opracowanie własne przy użyciu programu WEKA

Tabela 5. Wartości miary F (*F-measure*) na przykładzie zbioru *vehicle*

Model zagregowany	Liczba modeli składowych						
	10	20	30	40	50	100	200
RL (J48 binarne nieprzycięte)	0,966	0,979	0,959	0,958	0,966	0,986	0,972
RL (J48 dowolne nieprzycięte)	0,966	0,979	0,959	0,958	0,966	0,986	0,972
RL (J48 binarne przycięte)	0,966	0,979	0,959	0,966	0,972	0,979	0,972
RL (J48 dowolne przycięte)	0,966	0,979	0,959	0,966	0,972	0,979	0,972
RL (SimpleCart przycięte)	0,946	0,947	0,973	0,953	0,947	0,966	0,947
RL (SimpleCart nieprzycięte)	0,972	0,959	0,979	0,966	0,966	0,972	0,966
Bagging	0,904	0,898	0,910	0,923	0,930	0,931	0,930
AdaBoostM1	0,904	0,944	0,951	0,950	0,958	0,951	0,929
Losowy las	0,945	0,931	0,931	0,938	0,932	0,952	0,945

Źródło: opracowanie własne przy użyciu programu WEKA.

było przycinane. Jeżeli brać pod uwagę czułość modeli, to tutaj najlepsze rozwiązanie uzyskano po zagregowaniu 20 modeli bazowych. Nie miał znaczenia algorytm drzewa, sposób podziału węzłów (binarny i dowolny) ani fakt ich przycinania. Z danych zamieszczonych w tab. 4-5 wynika natomiast, że najwyższą precyzję i miarę *F* uzyskano dzięki włączeniu do rotacyjnego lasu drzew J48 i budowie aż 100 modeli bazowych. W przypadku pliku *vehicle* najgorsze wyniki uzyskano dzięki agregacji bootstrapowej (*bagging*), ale – co warto podkreślić – wniosek dotyczy wyłącznie tego zbioru obserwacji. „Klasyczne" modele zagregowane charakteryzowały się wysoką jakością w innych zbiorach danych, przewyższając w przypadku niektórych miar jakości rotacyjny las (zbiory: *glass*, *iris*, *segment*, *seeds*, *spambase*, *wine*).

Podsumowując wszystkie jedenaście zbiorów danych i cztery miary jakości, należy stwierdzić, że zdecydowanie najlepsze rozwiązania uzyskano dzięki stosowaniu rotacyjnego lasu. Algorytm ten był lepszy w 7 z 11 zbiorów obserwacji (w jednym zbiorze uzyskano remis, a w trzech zbiorach lepsze były „klasyczne" modele zagregowane). Jeżeli brać pod uwagę algorytm drzew klasyfikacyjnych w rotacyjnym lesie, to zarówno SimpleCart, jak i J48 charakteryzowały się porównywalną jakością rozwiązań, chociaż należy dodać, że odpowiednik C4 budował modele nawet 4-krotnie szybciej. Bez względu na użyty algorytm drzew nie powinno się przycinać modeli, gdyż drzewa przycięte okazały się lepsze jedynie w zbiorze danych *glass*. Bez względu na algorytm drzew nie warto było budować modeli o dowolnej liczbie węzłów potomnych, gdyż we wszystkich jedenastu zbiorach danych drzewa binarne były bardziej skuteczne.

W tabeli 6 przedstawiono liczbę modeli bazowych, po agregacji których uzyskano najwyższe wartości dokładności, czułości, precyzji i miary *F*. Odcieniem szarości wyróżniono najmniejszą ich liczbę (10). Wydaje się, że najszybciej uzyskuje się wy-

Tabela 6. Liczba modeli bazowych, po której uzyskano najwyższe miary jakości

Zbiór obserwacji	Dokładność	Czułość	Precyzja	Miara F
balance scale	40	10	20	40
banknote	10	10	10	10
glass	50	10	10	10
ionosphere	20	20	40	20
iris	20	10	20	20
seeds	10	10	10	10
segment	30	30	100	30
sonar	20	20	100	20
spambase	100	20	100	100
vehicle	10	20	100	100
wine	10	10	10	10

Źródło: opracowanie własne.

soką czułość modeli – w sześciu zbiorach danych wystarczył rotacyjny las składający się z 10 modeli bazowych. W trzech zbiorach obserwacji (*banknote*, *seeds*, *wine*) w ogóle nie było konieczności agregowania większej liczby modeli składowych, ale z drugiej strony w czterech zbiorach trzeba było połączyć ich znacznie więcej. W żadnym zbiorze danych nie było potrzeby budowy 200 modeli bazowych.

6. Zakończenie

Rotacyjny las to stosunkowo nowy, jeżeli porównać go do metody *bagging* czy *AdaBoost*, algorytm używany w podejściu wielomodelowym. Łączy analizę głównych składowych z dowolnym algorytmem drzew klasyfikacyjnych lub regresyjnych. Z analiz przeprowadzonych na potrzeby niniejszego artykułu wynika, że najlepsze rozwiązania w modelach dyskryminacyjnych i najkrótszy czas budowy modelu zapewnia użycie algorytmu J48 (odpowiednika C4), nieprzycinanego z binarnymi podziałami węzłów macierzystych.

Pomimo że metoda wykazała się wyższą skutecznością niż „klasyczne" modele zagregowane, to jednak należy wspomnieć o pewnych jej ograniczeniach. Po pierwsze, użycie analizy głównych składowych sprawia, że zmienne objaśniające wykorzystane w budowie modelu powinny być wyłącznie ilościowe. Autorzy algorytmu wspominają wprawdzie o konieczności transformacji zmiennych jakościowych na ilościowe, jednak koncepcja ta może budzić wiele wątpliwości. Co ważne, w badaniach rynkowych i marketingowych wiele cech statystycznych używanych w analizie danych ma charakter jakościowy, więc ich wyłączenie z modelu wydaje się niewskazane. Po drugie, program WEKA użyty do analizy zbiorów obserwacji jest niewydajny w przypadku zbiorów obserwacji o dużej liczbie przypadków. W zbio-

rze *spambase* nie zdołano zbudować 100 modeli bazowych, natomiast w zbiorach liczących ponad 10 tysięcy obiektów nie zbudowano żadnego. Być może pomocny byłby pakiet rotationForest w programie R albo implementacja algorytmu w innych programach analitycznych (np. Matlab). To dosyć duże ograniczenie, ponieważ w badaniach rynkowych i marketingowych – szczególnie w analizie danych wtórnych – zbiory danych mają nierzadko kilka tysięcy obserwacji. Po trzecie, rotacyjny las jako model zagregowany jest poniekąd „czarną skrzynką" i w badaniach rynkowych lub marketingowych może służyć wyłącznie do celów prognostycznych, a nie opisowych. Niestety, to także znacznie ogranicza stosowanie tego narzędzia, ponieważ nie pozwala wyjaśnić badanej dziedziny i nie dostarcza wiedzy potrzebnej do formułowania strategii marketingowych.

Literatura

Aličković E., Subasi A., 2015, *Breast cancer diagnosis using GA feature selection and Rotation Forest*, Neural Computing and Applications, Springer Online, s. 1-11.

Breiman L., 1994, *Bagging predictors*, Technical Report, No. 421, Department of Statistics University of California Berkeley, California, September.

Breiman L., 2001, *Random forests*, Machine Learning, nr 45, s. 5-32.

Breiman L., Cutler A., 2007, *Random Forests*, stat-www.berkeley.edu, 15.10.

De Bock K.W., Van den Poel D., 2011, *An empirical evaluation of rotation-based ensemble classifiers for customer churn prediction*, Expert Systems with Applications, vol. 38 s. 12293-12301.

Dehzangi A. i in., 2010, *Using Rotation Forest for Protein Fold Prediction Problem: An Empirical Study*, [w:] *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, Volume 6023 of the series Lecture Notes in Computer Science, Springer Berlin Heidelberg, s. 217-227.

Freund Y., Shapire R.E., 1997, *A decision-theoretic generalization of on-line learning and an application to boosting*, Journal of Computer and System Sciences, no. 55, s. 119-139.

Gatnar E., 2008, *Podejście wielomodelowe w zagadnieniach dyskryminacji i regresji*, PWN, Warszawa.

Hu Y-J. i in., 2012, *Decision tree-based learning to predict patient controlled analgesia consumption and readjustment*, BMC Medical Informatics and Decision Making, nr 12:131, s. 1-15.

Idris A., Khan A., Lee Y.S., 2013, *Intelligent churn prediction in telecom: employing mRMR feature selection and RotBoost based ensemble classification*, Applied Intelligence, October, vol. 39, issue 3, s. 659-672.

Karabulut E.M., İbrikçi T., 2012, *Effective diagnosis of coronary artery disease using the rotation forest ensemble method*, Journal of Medical Systems, vol. 36, issue 5, October 2012, s. 3011-3018.

Lasota T., Łuczak T., Trawiński B., 2012, *Investigation of rotation forest method applied to property price prediction*, Artificial Intelligence and Soft Computing, vol. 7267 of the series Lecture Notes in Computer Science, s. 403-411.

Ojha V.K., Dutta P., Chaudhuri A., 2016, *Identifying hazardousness of sewer pipeline gas mixture using classification methods: a comparative study*, Neural Computing and Applications, Springer Online, s. 1-12.

Ozcift A., 2012, *SVM feature selection based rotation forest ensemble classifiers to improve computer-aided diagnosis of Parkinson disease*, Journal of Medical Systems, vol. 36, issue 4, August 2012, s. 2141-2147.

Rodriguez J.J., Alonso C.J., 2004, *Rotation-Based Ensembles*, [w:] R. Conejo R. i in. (red.), *Current Topics in Artificial Intelligence,* "Lecture Notes in Computer Science", vol. 3040, Springer-Verlag, Berlin Heidelberg, s. 498-506.

Rodriguez J.J., Kuncheva L., Alonso C.J., 2006, *Rotation Forest: A New Classifier Ensemble Method*, IEEE Transactions On Pattern Analysis And Machine Intelligence, vol. 28, no. 10, October, s. 1619-1630.

Santos P. i in., 2012, *Wind turbines fault diagnosis using ensemble classifiers*, Advances in Data Mining. Applications and Theoretical Aspects, vol. 7377 of the series Lecture Notes in Computer Science, s. 67-76.

Zhang B., Zhou Y., 2012, *Vehicle type and make recognition by combined features and rotation forest ensemble*, International Journal of Pattern Recognition and Artificial Intelligence, vol. 26, no. 3, s. 1-25.