

FIO
HIEZNY KL
RAC
EILTCN HAY
ZFY
EHCNZRIAFY

Igor T. Podolak

KLASYFIKATOR
HIERARCHICZNY

z nakładającymi się grupami klas



KLASYFIKATOR
HIERARCHICZNY



Igor T. Podolak

KLASYFIKATOR HIERARCHICZNY

z nakładającymi się grupami klas

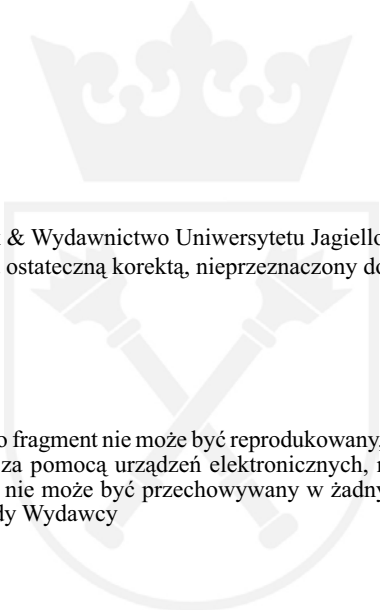
Praca naukowa finansowana przez Ministerstwo Nauki i Szkolnictwa Wyższego; projekt badawczy nr 6548/B/T02/2011/40

RECENZENCI

prof. dr hab. inż. Leszek Rutkowski, czł. koresp. PAN
prof. dr hab. Mariusz Flasiński

PROJEKT OKŁADKI

Agnieszka Winciorek



© Copyright by Igor Podolak & Wydawnictwo Uniwersytetu Jagiellońskiego
Preprint, egzemplarz przed ostateczną korektą, nieprzeznaczony do sprzedaży
Kraków 2012
All rights reserved

Niniejszy utwór ani żaden jego fragment nie może być reprodukowany, przetwarzany i rozpowszechniany w jakikolwiek sposób za pomocą urządzeń elektronicznych, mechanicznych, kopiujących, nagrywających i innych oraz nie może być przechowywany w żadnym systemie informatycznym bez uprzedniej pisemnej zgody Wydawcy

ISBN 978-83-233-3444-6



www.wuj.pl

Wydawnictwo Uniwersytetu Jagiellońskiego
Redakcja: ul. Michałowskiego 9/2, 31-126 Kraków
tel. 12-631-18-80, tel./fax 12-631-18-83
Dystrybucja: tel. 12-631-01-97, tel./fax 12-631-01-98
tel. kom. 506-006-674, e-mail: sprzedaz@wuj.pl
Konto: PEKAO SA, nr 80 1240 4722 1111 0000 4856 3325

Spis treści

Spis rysunków	vii
Spis tabel	ix
Wprowadzenie	xi
Wstęp	1
1. Podział zadania klasyfikacji	9
1.1. Klasyfikator	10
1.2. Podział przestrzeni klas	10
1.2.1. Losowa symulacja klasyfikatora dla podproblemu	15
1.2.2. Symulacja z wykorzystaniem prawa Bayesa	17
1.2.3. Wykorzystanie ewaluacji ryzyka w rzeczywistym problemie	19
1.3. Wykorzystanie słabych klasyfikatorów	24
2. Klasyfikator Hierarchiczny HCOC	33
2.1. Podział przestrzeni wyjściowej klas	33
2.1.1. Klastry i ich nakładanie się	37
2.2. Definicja HCOC	38
2.2.1. Macierz klastrowania F	40
2.3. Ewaluacja klasyfikatora HCOC	46
2.3.1. Agregacja wyników klasyfikacji w modelach złożonych	46
2.3.2. Wagi klastrów HCOC	47
2.3.3. Ewaluacja poddrzew	49
2.3.4. Zależność wag klastrów od składowych klas	55
2.4. HCOC jako rozwiązanie zadania przez podział	57
2.4.1. Klasyfikacja przykładów	60
2.5. Nauczanie pojedynczych węzłów	63
2.6. Zadanie klastrowania	65
2.6.1. Rozszerzenie algorytmu klastrowania aglomeratywnego	66
2.6.2. Bayesowskie podejście do klastrowania	67

2.6.3.	Urównoleglenie klastrowania przy wykorzystaniu algorytmu Rosnącego Gazu Neuronowego GNG	70
2.6.4.	Wykorzystanie metod genetycznych dla klastrowania	73
2.6.5.	Inne funkcje dopasowania	74
2.6.6.	Alternatywne klastrowanie dla lasu drzew decyzyjnych	76
2.6.7.	Problem zapewnienia różnorodności klasyfikatorów	77
2.7.	Zbieżność nauczania HCOC	79
2.7.1.	HCOC jako złożony klasyfikator	79
2.7.2.	Błąd HCOC a słabość klasyfikatorów bazowych	84
2.7.3.	Zależność błędu HCOC od błędu generalizacji	84
3.	Eksperymenty i doświadczenia	89
3.1.	Eksperyment Mixture of Gaussians dla wielu klas wyjściowych	89
3.2.	Rozpoznawanie przedmiotów z bazy COIL i porównanie z innym modelem hierarchicznym	93
3.3.	Zbiory porównawcze z repozytoriów	94
3.4.	Klasyfikacja tekstur	94
3.5.	Zastosowania w teorii automatów	96
3.6.	Rozpoznawanie twarzy	99
3.7.	Ekstrakcja reguł	102
	Zakończenie	107
	Dodatek A. Problemy nauczania maszynowego	109
A.1.	Nadzorowane nauczanie maszynowe	109
A.2.	Atrybuty przykładów uczących	109
A.3.	Funkcje kosztu i ryzyka	110
A.4.	Klasyfikatory „monolityczne”	111
A.5.	Klasyfikatory złożone	113
A.6.	Agregacja wyników	115
A.7.	Sposoby selekcji końcowej klasy	117
A.8.	Podejścia przy aglomeratywnym klastrowaniu	117
A.9.	Algorytm GNG	118
A.10.	Niektóre miary różnorodności	118
A.11.	Rozkład Beta	120
A.12.	Rozkład Dirichleta	121
A.13.	Walidacja krzyżowa i błąd $Err^{(0.632)}$	122
	Bibliografia	125
	Skorowidz pojęć	135

Spis rysunków

1.1. Mapy entropii dla podproblemów	13
1.2. Przepływ danych w prostym HCOC	14
1.3. Budowa hipotezy dla podproblemu przez losowy wybór odpowiedzi	15
1.4. Przypisanie do klas w podproblemie \mathcal{I}	16
1.5. Ewaluacja klasyfikatora dla podproblemu	17
1.6. Ewaluacja wartości funkcji ryzyka podproblemów problemu vowel	20
1.7. Ewaluacja wartości funkcji ryzyka podproblemów problemu vowel	21
1.8. Wartość $\alpha(K)$ jako problem geometryczny dla $K = 4$	29
1.9. Wartości $\alpha(K)$ dla różnych K	30
2.1. Statystyka różnych słabych klasyfikatorów dla problemu vowel . .	34
2.2. Statystyka różnych słabych klasyfikatorów dla problemu primary tumor	35
2.3. Średnie aktywacje sieci nauczonych dla problemu vowel	36
2.4. Średnie aktywacje sieci nauczonych dla problemu vowel	36
2.5. Przykładowa struktura HCOC	39
2.6. Działanie HCOC na syntetycznym problemie złożonym z 10 gau- sowskich chmur	41
2.7. Struktura HCOC dla zadania Mixture of Gaussians	42
2.8. Mixture of Gaussians – podział i ewaluacja RESTRICTED	43
2.9. Problem Mixture of Gaussians – podział i ewaluacja α -RESTRICTED .	44
2.10. Przypisanie przykładów do klas	47
2.11. Ewaluacja metodą ALL-SUBTREES	52
2.12. Wykresy empirycznych gęstości aktywacji $Cl_j(x)$	54
2.13. Stosunek sumy najwyższych aktywacji do sumy aktywacji losowych	56
2.14. Stosunek sumy najwyższych aktywacji wraz z aktywacją dla klasy prawdziwej do sumy dla aktywacji losowych	57
2.15. Stosunek wartości wag klastrów z klasą prawdziwą do wag kla- strów baz klasy prawdziwej	58
2.16. Względne wartości wag klastrów z klasą prawdziwą do wag innych klastrów	59

2.17. Względne wartości wag klastrów z klasą prawdziwą do wag innych klastrów dla zbiorów audiology i arrhythmia	59
2.18. Pojedyncze drzewa decyzyjne dla breast tumor	65
2.19. Algorytm klastrowania przez rozszerzenie SAHN	68
2.20. Algorytm klastrowania Bayesowskiego	70
2.21. Klastrowanie algorytmem GNG	72
2.22. Porównanie metod klastrowania Bayesowskiego i GNG	73
2.23. Schemat procedury genetycznej	75
2.24. Pojedyncze drzewo decyzyjne dla problemu vowel	77
3.1. Problem Mixture of Gaussians	90
3.2. Rozrzuty błędów dla HCOC w Mixture of Gaussians	91
3.3. Różne ewaluacje problemu Mixture of Gaussians	92
3.4. Model hierarchiczny Wanga i Casasenta i problem COIL	94
3.5. Podział na klastry w problemie rozpoznawania tekstur	97
3.6. Automat synchronizujący	101
3.7. Doświadczenie z rozpoznawaniem twarzy	102
3.8. Doświadczenie z rozpoznawaniem twarzy	103
A.1. Architektura dwupoziomowego modelu typu HME	114
A.2. Różne rodzaje zależności w komitetach	116
A.3. Algorytm Growing Neural Gas	119

Spis tabel

1.1. Korelacja pomiędzy wartościami ryzyka problemu a jego ewaluacją	19
1.2. Korelacje dla błędów podproblemów problemu tumor i ich ewaluacji	20
1.3. Korelacje Spearmana dla różnych typów ewaluacji $R[HCOC]$	22
1.4. Korelacje podejść A i C	24
1.5. Wartości $\alpha(K)$	29
2.1. Liczba poprawnych klastrowań	46
2.2. Porównania ewaluacji standardowej i stochastycznej Γ	50
2.3. Frakcja powierzchni pod krzywą empirycznych gęstości aktywacji	54
2.4. Porównanie różnych funkcji <i>fitness</i>	79
3.1. Porównanie wyników HCOC z tymi dla modelu Wang i Casasenta	94
3.2. Wyniki HCOC dla zbiorów porównawczych	95
3.3. Porównania działania HCOC dla zbiorów testowych	96
3.4. Wyniki klasyfikacji tekstur	98
3.5. Wyniki predykcji długości minimalnego słowa synchronizującego	100
3.6. Korelacje między długością MSW a wartościami cech	101
3.7. Reguły z HCOC	104
A.1. Różne typy i możliwości reprezentacji atrybutów	110
A.2. Różne możliwe funkcje kosztu	111
A.3. Miary różnorodności klasyfikatorów	121

Wprowadzenie

Praca przedstawia koncepcję hierarchicznego klasyfikatora HCOC (ang. *Hierarchical Classifier with Overlapping Clusters*) zbudowanego przy użyciu nowego paradygmatu automatycznego podziału na podproblemy. Wykorzystuje on przybliżoną klasyfikację przez już utworzony klasyfikator bazowy w węźle, implementowany przez prosty klasyfikator, dla podziału zadania na podproblemy. Praca przedstawia założenia teoretyczne proponowanego modelu, jego definicję, opis podstawowych zagadnień związanych z budową. W kolejnych rozdziałach zawarte są propozycje zastosowanych algorytmów wraz z uzasadnieniem teoretycznym i doświadczeniami wykazującymi ich przydatność i poprawność. Przedstawione są teoretyczne uzasadnienia poprawności modelu pokazujące jego zbieżność i przewagę nad modelami monolitycznymi.

Część z zawartego materiału ukazała się wcześniej w czasopismach informatycznych o zasięgu międzynarodowym, część prezentowana była na konferencjach i opublikowana w związanych z nimi wydawnictwach.

Praca jest częściowo finansowana przez grant Narodowego Centrum Nauki, nr NCN 6458/T02/2011/40.

Wstęp

Jednym z podstawowych problemów w nauczaniu maszynowym (ang. *machine learning*, ML) jest problem *klasyfikacji*. Zadanie polega na przypisaniu podanego za pomocą wektora *atrybutów* obiektu do jednej ze zdefiniowanych wcześniej *klas*. Wektor atrybutów ma zwykle skończoną, ustaloną z góry, stałą długość; liczba klas jest także skończona i ustalona.

W ramach ML zaproponowanych zostało wiele algorytmów, które rozwiązują ten problem przez *nauczanie nadzorowane*, polegające na przedstawieniu statystycznemu *algorytmowi uczącemu* szeregu przykładów z przypisaną prawidłową klasą. Nauczanie tworzy *model*, który, z pewną dokładnością, potrafi rozpoznawać przykłady przedstawione wcześniej w trakcie nauczania oraz rozpoznawać część z tych, które nie zostały wcześniej pokazane. Oczywiście możliwe jest skonstruowanie takiego algorytmu, który (choćby przez zapamiętanie wszystkich przykładów w tabeli) będzie poprawnie rozpoznawać wszystkie przykłady przedstawione w procesie nauczania. Pociąga to jednak za sobą słabsze rozpoznawanie przykładów nieprzedstawionych, czyli słabszą generalizację.

Celem ML jest znalezienie algorytmu (albo rodziny algorytmów), który będzie osiągał wysoki stopień generalizacji (tzn. będą prawidłowo rozpoznawać przykłady, które nie były uwzględnione w zbiorze uczącym). Istotne jest, aby znaleziony algorytm był jednocześnie stosunkowo prosty w użyciu, nie wymagał dodatkowej wiedzy eksperckiej, długiego czasu prób przy wyszukiwaniu suboptymalnego modelu, był maksymalnie prosty itd. Istnieje szereg wiele podejść, takich jak systemy ekspertowe, klasyfikatory Bayesowskie, różne zastosowania logiki rozmytej, drzewa decyzyjne, algorytmy typu EM, sieci neuronowe i inne. Z ich pomocą możliwe jest budowanie podstawowych monolitycznych modeli, tzn. opartych na jednym podejściu, w odróżnieniu od modeli *hybrydowych* łączących kilka różnych podejść do utworzenia jednego modelu.

Jednym z podstawowych problemów jest to, że znalezienie optymalnego, a raczej najlepszego możliwego do znalezienia, modelu jest zwykle bardzo czasochłonne, wymaga wiedzy specjalistycznej, zależy od reprezentatywności dostępnego zbioru uczącego. Często też, aby osiągnąć wystarczająco dużą skuteczność, szukany model wymaga bardzo wielu parametrów, co wiąże się ze słabym poziomem generalizacji.

Alternatywnym podejściem jest tworzenie rozwiązań złożonych z wielu prostych modeli o niskiej skuteczności, jednak łatwych do zbudowania, których końcowe klasyfikacje są łączone. Może to polegać na podziale zbioru uczącego na szereg podprzestrzeni tworzących podproblemy. Takimi podejściami są, między innymi, *bagging*, *mixture of experts*, *uśrednianie*, *stacking*, *bumping* i inne (Hastie *et al.*, 2001; Bishop, 2006). Ważną ich cechą jest próba podejścia do częściowego rozwiązania problemu *bias-variance*.

Indywidualne modele mogą być budowane niezależnie (poprzez podział przestrzeni wejściowej) albo sekwencyjnie. Sekwencyjne podejście jest szczególnie wydajne, jeśli $k + 1$ model uwzględnia wyniki działania i skuteczność poprzedniego, k -tego modelu, a stąd, pośrednio, wszystkich innych wcześniej zbudowanych. Budowany model skupia się na rozwiązaniu dla tych elementów przestrzeni problemu, które wcześniej okazały się trudne do rozwiązania. Takie podejście leży u podstaw *boostingu*, w którym budowanych jest sekwencja *słabych*, bardzo prostych, modeli. Ich odpowiednie złożenie okazuje się już modelem silnym. Schapire wykazał równoważność takiego słabego nauczania z nauczaniem silnym (Schapire, 1990). Z modelu Schapire'a pochodzi, między innymi, model AdaBoost.

Modele typu AdaBoost (od *ang.* Adaptive Boosting) dzielą wejściową przestrzeń atrybutów. Robią to w sposób *miękki* przez ustanowienie rozkładu prawdopodobieństwa losowania przykładów w trakcie nauczania. Dzięki temu niektóre z modeli składowych uwzględniają w większym stopniu pewne części przestrzeni wejściowej (przestrzeni problemu), inne skupiają się na częściach, dla których te pierwsze dają słabsze wyniki. Oznacza to *podział przestrzeni wejściowej* problemu. Takie podejście pozwala na sukcesywne poprawianie rozwiązania (końcowej *hipotezy*) poprzez dodawanie nowych, prostych, klasyfikatorów i odpowiednie uśrednianie wyniku.

W tej pracy proponowany jest model, w którym dzielona jest nie przestrzeń wejściowa, lecz przestrzeń wyjściowa klas. Podział polega na wyróżnieniu grup klas, które w naturalny sposób tworzą podproblemy, a następnie rozwiązywanie tych podproblemów z osobna. Podejście to bierze swój początek z obserwacji, że dla zbioru uczącego, w którym przykłady różnych klas nie są zrównoważone, proste (słabe) klasyfikatory wykazują naturalną skłonność do grupowania wyników wokół klas przeważających w zbiorze (Podolak, 2008; Podolak, Roman, 2012). Wynika to z faktu, że niektóre klasy są bardziej do siebie podobne, a stąd modele generują podobne wzorce aktywacji. Stąd rozróżnienie, a więc i prawidłowa klasyfikacja, przykładów z klas podobnych do siebie będzie trudniejsza niż klasyfikacja przykładów z klas różniących się.

Poza klasyfikacją, możliwe jest także zastosowanie podejścia HCOC w problemach aproksymacji, co zostało pokazane w pracy (Brodowski, Podolak, 2011). W tej pracy skupimy się jednak nad samym aspektem klasyfikacji.

Bibliografia

- Alpaydin, E. (2004). *Introduction to Machine Learning*. The MIT Press.
- Andrews, R., Diederich, J., Tickle, A.B. (1995). Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-Based Systems*, 8(6), 373–389.
- Bartlett, P.L., Jordan, M.I., McAuliffe, J.D. (2006). Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473), 138–156.
- Bekkerman, R., Bilenko, M., Langford, J., editors (2012). *Scaling up Machine Learning: Parallel and Distributed Approaches*. Cambridge University Press.
- Bengio, Y., LeCun, Y. (2007). Scaling learning algorithms towards AI. W: L. Bottou, O. Chapelle, D. DeCoste, J. Weston, editors, *Large-Scale Kernel Machines*. MIT Press.
- Besson, F.M. (2010). Hadoop MapReduce. *Sort*, s. 3–4.
- Bishop, C.M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press.
- Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- Bishop, C.M., Svensén, M. (2003). Bayesian hierarchical mixtures of experts. W: U. Kjærulff, C. Meek, editors, *Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, s. 57–64.
- Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2), 123–140.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45, 5–32.
- Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J. (1984). CART: Classification and Regression Trees. W: X. Wu, V. Kumar, editors, *The top ten algorithms in data mining*. Chapman & Hall/CRC Press, s. 179–201.
- Brodatz, P. (1966). *Textures: A Photographic Album for Artists and Designers*. Dover Publications.
- Brodowski, S., Podolak, I.T. (2011). Hierarchical estimator. *Expert Systems with Applications*, 38(10), 12237–12248.

- Brown, G., Wyatt, J., Harris, R., Yao, X. (2005). Diversity creation methods: a survey and categorisation. *Information Fusion*, **6**, 5–20.
- Brugger, D., Bogdan, M., Rosenstiel, W. (2008). Automatic cluster detection in Kohonen's SOM. *IEEE Transactions on Neural Networks*, **19**(3), 442–459.
- Bühlmann, P., Yu, B. (2006). Sparse boosting. *Journal of Machine Learning Research*, **7**, 1001–1024.
- Casasent, D., Wang, Y.-C.F. (2005). A hierarchical classifier using new support vector machines for automatic target recognition. *Neural Networks*, **18**, 541–548.
- Černý, J., Pirická, A., Rosenauerová, B. (1971). On directable automata. *Kybernetika*, **7**(4), 289–298.
- Cesa-Bianchi, N., Gentile, C., Zaniboni, L. (2006). Incremental algorithms for hierarchical classification. *Journal of Machine Learning Research*, **7**, 31–54.
- Chou, Y.-Y., Shapiro, L.G. (2003). A hierarchical multiple classifier learning algorithm. *Pattern Analysis & Applications*, **6**(2), 150–168.
- Christiani, N., Shawe-Taylor, J. (2000). *Support Vector Machines and other Kernel Based Learning Methods*. Cambridge University Press.
- Ciampi, A., Lechevallier, Y., Limas, M.C., Marcos, A.G. (2007). Hierarchical clustering of subpopulations with a dissimilarity based on the likelihood ratio statistic: application to clustering massive data sets. *Pattern Analysis and Applications*, **11**(2), 199–220.
- Dara, R.A., Kamel, M.S., Wanas, N. (2009). Data dependency in multiple classifier systems. *Pattern Recognition*, **42**(7), 1260–1273.
- Day, W., Edelsbrunner, H. (1984). Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, **1**(1), 7–24.
- Dean, J., Ghemawat, S. (2008). MapReduce : simplified data processing on large clusters. *Communications of the ACM*, **51**(1), 1–13.
- Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, **7**, 1–30.
- Dietterich, T.G., Bakiri, G. (1995). Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, **2**, 263–286.
- Duch, W., Itert, Ł. (2003). Committees of undemocratic competent models. W: *International Conference of Artificial Neural Networks*. Springer, s. 33–36.
- Duch, W., Wiczczonek, T., Biesiada, J., Blachnik, M. (2004a). Comparison of feature ranking methods based on information entropy. W: *Proceedings of the IEEE International Joint Conference on Neural Networks*. IEEE, s. 1415–1419.

- Duch, W., Setiono, R., Żurada, J.M. (2004b). Computational intelligence methods for rule-based data understanding. *Proceedings of the IEEE*, **92**(5), 771–805.
- Duda, R.O., Hart, P.E., Stork, D.G. (2000). *Pattern Classification*. Wiley Interscience.
- Efron, B. (1983). Estimating the error rate of a prediction rule: improvement on cross-validation. *Journal of the American Statistical Society*, **78**(382), 316–331.
- Efron, B., Tibshirani, R. (1997). Improvements on cross-validation: The .632+ bootstrap method. *Journal of the American Statistical Association*, **92**(438), 548–560.
- Eibl, G., Pfeiffer, K.-P. (2005). Multiclass boosting for weak classifiers. *Journal of Machine Learning*, **6**, 189–210.
- Ferguson, T.S. (1973). A bayesian analysis of some nonparametric problems. *Annals of Statistics*, **1**(2), 209–230.
- Fisher, R.A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, **7**(2), 179–188.
- Frank, E., Hall, M., Pfahringer, B. (2003). Locally weighted Naive Bayes. W: *Proceedings of the Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, s. 249–256.
- Freund, Y., Schapire, R.E. (1997). A decision theoretic generalization of online learning and an application to boosting. *Journal of Computer and System Sciences*, **55**, 119–139.
- Friedman, J., Hastie, T., Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, **28**(2), 337–407.
- Friedman, J., Hastie, T., Rosset, S., Tibshirani, R., Zhu, J. (2004). Discussion of three boosting papers. *Annals of Statistics*, **32**(1), 102–107.
- Friedman-Hill, E. (2003). Jess, the rule engine for Java platform. Sandia National Laboratory.
- Fritzke, B. (1993). Kohonen Feature Maps and Growing Cell Structures: a performance comparison. W: S.J. Hanson, J.D. Cowan, C.L. Giles, editors, *Advances in Neural Information Processing Systems*, vol. 5. Morgan Kaufmann, s. 115–122.
- Fritzke, B. (1995). A Growing Neural Gas network learns topologies. W: G. Tesauero, D.S. Touretzky, T.K. Leen, editors, *Advances in Neural Information Processing Systems*, vol. 7. MIT Press, s. 625–632.
- Fritzke, B. (1997). A self-organizing network that can follow non-stationary distributions. W: *International Conference on Artificial Neural Networks*. Springer, s. 613–618.
- Giacinto, G., Roli, F. (2001). An approach to the automatic design of multiple classifier systems. *Pattern Recognition Letters*, **22**, 25–33.
- Giarratano, J., Riley, G. (1994). *Expert Systems: Principles and Programming*. PWS Publishing.

- Hand, D., Mannila, H., Smyth, P. (2001). *Principles of Data Mining*. MIT Press.
- Hastie, T., Tibshirani, R. (1998). Classification of pairwise couplings. W: *Advances in Neural Information Processing Systems*, vol. 10. MIT Press, s. 507–513.
- Hastie, T., Tibshirani, R., Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Verlag.
- Haykin, S. (2009). *Neural Networks: a Comprehensive Foundation*. Prentice Hall.
- Hinton, G.E., Osindero, S. (2006). A fast learning algorithm for Deep Belief Networks. *Neural Computation*, **18**, 1527–1554.
- Holland, J.H. (1992). *Adaptation in Natural and Artificial Systems*. MIT Press.
- Ibrahim, S., Jin, H., Lu, L., Qi, L., Wu, S., Shi, X. (2009). Evaluating MapReduce on virtual machines : the Hadoop case. W: M.G. Jaatun, G. Zhao, C. Rong, editors, *Cloud Computing*, vol. 5931 of LNCS. Springer, s. 519–528.
- Itti, L., Baldi, P. (2006). Bayesian surprise attracts human attention. W: Y. Weiss, B. Schölkopf, J. Platt, editors, *Advances in Neural Information Processing Systems*, vol. 18. MIT Press, s. 547–554.
- Jordan, M.I., Jacobs, R.A. (1993). Hierarchical Mixtures of Experts and the EM algorithm. A.I. Memo 1440, Massachusetts Institute of Technology, Artificial Intelligence Laboratory.
- Kari, J. (2003). Synchronizing finite automata on eulerian digraphs. *Theoretical Computer Science*, **295**(1–3), 432–438.
- Kearns, M.J., Valiant, L.G. (1989). Cryptographic limitations on learning boolean formulae and finite automata. W: *Proceedings of 21st Annual ACM Symp. on Theory of Computing*. ACM Press, s. 434–444.
- Kearns, M.J., Valiant, L.G. (1994). Cryptographic limitations on learning boolean formulae and finite automata. *Journal of the ACM*, **41**(1), 67–95.
- Kittler, J., Hojjatoleslami, A., Winder, T. (1997). Strategies for combining classifiers employing shared and distinct pattern representations. *Pattern Recognition Letters*, **18**, 1373–1377.
- Kohavi, R., Quinlan, J.R. (2002). *Data mining tasks and methods: Classification: decision-tree discovery*. Oxford University Press, s. 267–276.
- Kohonen, T., Honkela, T. (2007). Kohonen network. *Scholarpedia*, **2**(1), 1568.
- Kulkarni, A.V. (1978). On the mean accuracy of hierarchical classifiers. *IEEE Transactions on Computers*, **27**, 771–776.
- Kumar, S., Ghosh, J. (1999). GAMLS: A Generalized framework for Associative Modular Learning Systems. W: *Proceedings of the Applications and Science of Computational Intelligence*. s. 24–34.

- Kumar, S., Ghosh, J., Crawford, M.M. (1999). A versatile framework for labeling imagery with a large number of classes. W: *International Joint Conference on Neural Networks*. s. 2829–2833.
- Kumar, S., Ghosh, J., Crawford, M.M. (2002). Hierarchical fusion of multiple classifiers for hyperspectral data analysis. *Pattern Analysis & Applications*, 5(2), 210–220.
- Kuncheva, L.I. (2000). Clustering-and-selection model for classifier combination. W: *Knowledge-Based Intelligent Engineering Systems and Allied Technologies, 2000. Proceedings. Fourth International Conference on*, vol. 1. IEEE, s. 185–188.
- Kuncheva, L.I. (2005). *Combining Pattern Classifiers*. Wiley Press.
- Kuncheva, L.I. (2009). That elusive diversity in classifier ensembles. *Pattern Recognition and Image Analysis*, 2652, 1126–1138.
- Kurzyński, M. (1983). Decision rules for a hierarchical classifier. *Pattern Recognition Letters*, 1, 305–310.
- Manning, C.D., Raghavan, P., Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.
- Meynet, J., Thiran, J.-P. (2010). Information theoretic combination of pattern classifiers. *Pattern Recognition*, 43(10), 3412–3421.
- Michalewicz, Z. (1996). *Genetic Algorithms + Data Structures = Evolution Programs*. Springer.
- Minsky, M., Papert, S. (1969). *Perceptrons: An Introduction to Computational Geometry*. MIT Press.
- Nene, S.A., Nayar, S.K., Murase, H. (1996). Columbia Object Image Library (COIL-100). Technical Report CUCS-006-96, Columbia University.
- Newman, D., Hettich, S., Blake, C., Aha, D., Murphy, P. (1998). UCI repository of machine learning databases.
- Nguyen, A.V., Hudson, D.L., Cohen, M.E. (2010). Hadoop and MapReduce for the storage and processing of physiological data. W: *International Conference on Software Engineering Data Engineering International Society for Computers and their Applications ISCA*. s. 366–369.
- Olschewski, J., Ummels, M. (2010). The complexity of finding reset words in finite automata. W: *Proceedings of the 35th International Conference on Mathematical Foundations of Computer Science*. Springer, s. 568–579.
- Oza, N.C., Tumer, K. (2008). Classifier ensembles: select real world applications. *Information Fusion*, 9, 4–20.

- Panda, B., Herbach, J.S., Basu, S., Bayardo, R.J. (2012). MapReduce and its application to massively parallel learning of decision tree ensembles. W: R. Bekkerman, M. Bilenko, J. Langford, editors, *Scaling up machine learning: parallel and distributed approaches*. Cambridge University Press, s. 23–48.
- Parker, J.R. (2001). Rank and response combination from confusion matrix data. *Information Fusion*, **2**(2), 113–120.
- Podolak, I.T. (2007). Hierarchical rules for a hierarchical classifier. W: *Proceedings of the 8th International Conference on Adaptive and Natural Computing Algorithms*, vol. 4431 of LNCS. Springer, s. 749–757.
- Podolak, I.T. (2008). Hierarchical classifier with overlapping class groups. *Expert Systems with Applications*, **34**(1), 673–682.
- Podolak, I.T., Bartocha, K. (2009). A hierarchical classifier with Growing Neural Gas clustering. W: *Adaptive and Natural Computing Algorithms*, vol. 5495 of LNCS. Springer, s. 283–292.
- Podolak, I.T., Roman, A. (2009). A new notion of weakness in classification theory. W: *Advances in Intelligent and Soft Computing*, no. 57 in AISC. Springer, s. 239–245.
- Podolak, I.T., Roman, A. (2011a). Fusion of supervised and unsupervised training methods for a multi-class classification problem. *Pattern Analysis and Applications*, **14**(4), 395–413.
- Podolak, I.T., Roman, A. (2011b). Risk estimation for hierarchical classifier. W: *Hybrid Artificial Intelligent Systems*, no. 6678 in LNCS. Springer, s. 156–163.
- Podolak, I.T., Roman, A. (2011c). Risk function estimation for subproblems in a hierarchical classifier. *Pattern Recognition Letters*, **32**, 2136–2142.
- Podolak, I.T., Roman, A. (2012). Theoretical foundations and practical results for the hierarchical classifier. *Computational Intelligence*, **28**(4). w druku.
- Podolak, I.T., Xi, D., Lee, S.-W. (2002). Facial component extraction and face recognition with support vector machines. W: *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*. IEEE, IEEE Computer Society, s. 76–81.
- Podolak, I.T., Biel, S., Bobrowski, M. (2006). Hierarchical classifier. W: R. Wyrzykowski, editor, *Parallel Processing and Applied Mathematics*, vol. 3911 of LNCS. Springer, s. 591–598.
- Podolak, I.T., Roman, A., Jędrzejczyk, D. (2012a). Application of hierarchical classifier to minimal synchronizing word problem. W: L. Rutkowski, editor, *Lecture Notes in Computer Science*, vol. 7267 of LNCS. Springer, s. 421–429.
- Podolak, I.T., Roman, A., Deszyńska, A. (2012b). On the number of clusterings in a hierarchical classification model with overlapping clusters. *Schedae Informaticae*, **20**, 137–158.

- Quinlan, J.R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc.
- Quinlan, J.R. (1996). Learning decision tree classifiers. *ACM Computing Surveys*, **28**, 71–72.
- R Development Core Team (2005). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Rokach, L. (2009). Taxonomy for characterizing ensemble methods in classification tasks: a review and annotated bibliography. *Computational Statistics and Data Analysis*, **53**, 4046–4072.
- Rosenblatt, F. (1962). *Principles of Neurodynamics: Perceptron and the Theory of Brain Mechanisms*. Spartan Books.
- Rumelhart, D.E., Hinton, G.E., Williams, R.J. (1986). Learning representations by back-propagating errors. *Nature*, **323**(6088), 533–536.
- Samaria, F., Harter, A. (1994). Parameterisation of a stochastic model for human face identification. W: *Proceedings of the Second IEEE Workshop on Applications of Computer Vision*. IEEE, s. 138–142.
- Schaefer, R., Telega, H. (2007). *Foundations of Global Genetic Optimization*, vol. 74 of *Studies in Computational Intelligence*. Springer Verlag.
- Schapire, R.E. (1990). The strength of weak learnability. *Machine Learning*, **5**, 197–227.
- Schapire, R.E., Singer, Y. (1999). Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, **37**(3), 297–336.
- Scholkopf, B., Smola, A.J. (2002). *Learning with Kernels*. MIT Press.
- Schraudolph, N.N., Yu, J., Gunter, S. (2007). A stochastic quasi-newton method for on-line convex optimization. W: *Proceedings of the 11th Int. Conf. on Artificial Intelligence and Statistics*. s. 443–440.
- Schuerman, J., Doster, W. (1984). A decision theoretic approach to hierarchical classifier design. *Pattern Recognition*, **17**(3), 359–369.
- Sethi, I.K., Sarvarayudu, G. P.R. (1982). Hierarchical classifier design using mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **4**(4), 441–445.
- Setiono, R. (2001). Feedforward neural network construction using cross validation. *Neural Computation*, **13**, 2865–2877.
- Setiono, R., Leow, W.K. (2000). FERNN: an algorithm for fast extraction of rules from neural networks. *Applied Intelligence*, **12**(1/2), 15–25.
- Shipp, C.A., Kuncheva, L.I. (2002). Relationships between combination methods and measures of diversity in combining classifiers. *Information Fusion*, **3**(2), 135–148.

- Sneath, P.H., Sokal, R.R. (1973). *Principles of Numerical Taxonomy*. Freeman, San Francisco.
- Tapia, E., Bulacio, P., Angelone, L. (2010). Recursive ECOC classification. *Pattern Recognition Letters*, **31**(3), 210–215.
- Ting, K.M., Wells, J.R., Tan, S.C., Teng, S.W., Webb, G.I. (2010). Feature-subspace aggregating: ensembles for stable and unstable learners. *Machine Learning*, **82**(3), 375–397.
- Titsias, M.K., Likas, A. (2002). Mixture of Experts classification using a hierarchical mixture model. *Neural Computation*, **2244**, 2221–2244.
- Tou, J.Y., Tay, Y.H., Lau, P.Y. (2007). Gabor filters and grey-level co-occurrence matrices in texture classification. W: *Proceedings for Multimedia University International Symposium on Information and Communications Technologies M2USIC 2007*. s. 1–5.
- Trahtman, A.N. (2006). An efficient algorithm finds noticeable trends and examples concerning the Černý conjecture. W: *Lecture Notes in Computer Science*, vol. 4162 of LNCS. Springer, s. 789–800.
- Tsapanos, N., Tefas, A., Nikolaidis, N., Pitas, I. (2011). Shape matching using a binary search tree structure of weak classifiers. *Pattern Recognition*, **45**(6), 2363–2376.
- Tumer, K., Ghosh, J. (1996). Error correlation and error reduction in ensemble classifiers. *Connection Science*, **8**(3–4), 385–404.
- Valentini, G., Masulli, F. (2002). Ensembles of learning machines. W: R. Tagliaferri, M. Marinaro, editors, *Neural Nets, WIRN*, vol. 2486 of LNCS. Springer, s. 3–19.
- Valiant, L.G. (1984). A theory of the learnable. *Communications of the ACM*, s. 1134–1142.
- Vapnik, V.N. (1996). *The Nature of Statistical Learning Theory*. Springer.
- Vapnik, V.N. (1998). *Statistical Learning Theory*, vol. 2 of *Wiley Series on Adaptive and Learning Systems for Signal Processing, Communications and Control*. Wiley-Interscience.
- Wanas, N., Dara, R.A., Kamel, M.S. (2006). Adaptive fusion and co-operative training for classifier ensembles. *Pattern Recognition*, **39**(9), 1781–1794.
- Wang, Y.-C.F., Casasent, D. (2008). New support vector-based design method for binary hierarchical classifiers for multi-class classification problems. *Neural Networks*, **21**(2–3), 502–510.
- Wang, Y.-C.F., Casasent, D. (2009). A support vector hierarchical method for multi-class classification and rejection. W: *Proceedings of International Joint Conference on Neural Networks*. IEEE, s. 3281–3288.
- Witten, I.H., Frank, E. (2000). *Data Mining: Practical Machine Learning Tools with Java Implementations*. Morgan Kaufmann, San Francisco.

- Wołoszynski, T., Kurzyński, M. (2011). A probabilistic model of classifier competence for dynamic ensemble selection. *Pattern Recognition*, **44**(10–11), 2656–2668.
- Yule, G.U. (1900). On the association of attributes in statistics: with illustrations from the material of the childhood society, &c. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **194**(252–261), 257–319.
- Zhang, H. (2004). The optimality of Naive Bayes. W: V.Barr, M.Zdravko, editors, *Proceedings of the Seventeenth International Florida Intelligence Research Society*.
- Zwitter, M., Sokolic, M. (1988). Primary tumor data set.



Skorowidz pojęć

- AdaBoost, 2, 5, 25
- agregacja wyników, 115
 - bepośrednio zależna od danych, 116
 - klasyfikatorów, 46
 - niezależna od danych, 116
 - pośrednio zależna, 116
- algorytm
 - Growing Neural Gas, 71
- arg-max, 10
- atrybuty
 - ilościowe, 109
 - jakościowe, 109
- automat synchronizujący, 96
- błąd
 - $Err^{(632)}$, 123
 - $Err^{(1)}$, 123
- błędy
 - przypadkowe, 48
 - systematyczne, 48
- bagging, 113
- boosting, 24, 115
 - przez filtrowanie, 25
- bootstrap, 122
- drzewa decyzyjne, 37, 112
- entropia problemu, 12
- ewaluacja poddrzew, 49
 - α -RESTRICTED, 53
 - ALL-SUBTREES, 51
 - RESTRICTED, 52
 - SINGLE-PATH, 51
 - relacje między, 54
- Feating, 6
- funkcja kosztu, 110
- funkcja ryzyka, 110
 - empiryczna, 111
- funkcje fitness, 79
- generalizacja, 1
- Growing Neural Gas, GNG, 71
- HCOC, 38
 - definicja, 38
 - górną granicą ryzyka, 74
 - zbieżność nauczania, 79
- Hierarchical Mixture of Experts, 114
- indeks Giniego, 12
- klastrowanie, 37
 - α -RESTRICTED, 40
 - RESTRICTED, 40
 - aglomeratywne, 66
 - Bayesowskie, 67, 69
 - complete linkage, 117
 - genetyczne, 73
 - GNG, 70
 - urównoleglenie, 71
 - group average, 118
 - lasu drzew decyzyjnych, 76
 - liczba poprawnych, 46

- nakładanie się, 37
- SAHN, 66, 67
- single linkage, 117
- klastry, 67
 - nakładanie się, 37
- klasyfikacja, 109
 - grupa-klas-przeciwko-reszcie, 45, 57
 - jeden-przeciwko-jednemu, 57
 - jeden-przeciwko-wszystkim, 57
- klasyfikator, 10
 - HCOC
 - definicja, 38
 - ewaluacja, 46
 - klasyfikacja przykładów, 60
 - hierarchiczny, 115
 - słaby, 25, 26, 30, 40
 - uogólniony, 61
 - funkcja kosztu, 61
 - macierz generalizacji, 62
- komitet maszyn, 113
- las drzew decyzyjnych, 64
- liniowa analiza dyskryminacyjna, 112
- macierz
 - ECOC, 45
 - generalizacji, 62
 - klastrowania, 40
 - liczba klastrów, 46
 - poprawność, 42
 - misclassification, 67
- MapReduce, 7
- measure of belief, 52
- miara
 - pseudo-loss, 25
 - różnorodności, 118
 - κ , 120
 - Kohavi–Wolpert, 120
 - korelacja, 120
 - Q-statystyka, 120
 - złożoności problemu, 11
 - miara ryzyka
 - podproblemu, 14, 18
 - miary różnorodności, 78
 - modele agregacji wyników, 115
 - naiwny klasyfikator Bayesa, 113
 - najbliższych sąsiadów, metody, 112
 - nauczanie, 1
 - PAC, 24
 - perceptron, 111
 - wielowarstwowy, 111
 - podział
 - przestrzeni atrybutów, 9, 33
 - przestrzeni klas, 10, 33
 - zbioru klas, 9
 - problem
 - klasyfikacji, 109
 - różnorodność klasyfikatorów, 77
 - redukcja atrybutów, 60
 - Region of Interest, ROI, 40
 - rozkład
 - Beta, 120
 - Dirichleta, 27, 121
 - rozpoznawanie twarzy, 99
 - SAHN, 66
 - samorganizujące się mapy, 112
 - SOM, 112
 - Support Vector Machine, SVM, 113
 - wagi klastrów, 47, 61
 - standardowo, 48
 - stochastycznie Γ , 48
 - zależność od klas, 55
 - walidacja krzyżowa, 122
 - złożoność podproblemu, 12
 - zbiór uczący, 109

W czasach, gdy ilość i rozmiar danych stają się ogromne, istnieje duże zapotrzebowanie na podejścia do klasyfikacji, które będą skuteczne w rozwiązywaniu takich problemów. W niniejszej pracy przedstawiono algorytmy i związane z nimi zagadnienia wielopoziomowego, hierarchicznego klasyfikatora. Nowością w zaproponowanym ujęciu jest schemat podziału na podzadania przez podział przestrzeni rozpoznawanych klas obiektów. Algorytm wykorzystuje w tym celu wiedzę zdobytą przez mało dokładnie nauczone, „słabe” klasyfikatory. Ich słabość pozwala na szybkie nauczanie niewymagające wielu doświadczeń. Taki mechanizm umożliwia budowę silnego klasyfikatora na kolejnych dodawanych poziomach. Autor prezentuje podstawowe zasady hierarchicznej klasyfikacji, nowe definicje słabego klasyfikatora, szczegółowy opis modelu oraz doświadczenia pokazujące skuteczność podejścia.

„Książka przedstawia oryginalne rezultaty, które częściowo już znalazły swoje miejsce w literaturze światowej we wcześniejszych publikacjach autora”.

Z recenzji prof. Leszka Rutkowskiego (Politechnika Częstochowska)

„W monografii został przedstawiony oryginalny model klasyfikatora, który może być wykorzystany jako skuteczne narzędzie klasyfikacji obrazów”.

Z recenzji prof. Mariusza Flasińskiego (Uniwersytet Jagielloński)

Dr Igor Podolak jest informatykiem pracującym na Wydziale Matematyki i Informatyki Uniwersytetu Jagiellońskiego w Krakowie. Specjalizuje się w problematyce sztucznej inteligencji, a szczególnie w zagadnieniach klasyfikacji z wykorzystaniem złożonych systemów.

WYDAWNICTWO
UNIwersytetu
JAGIELLOŃSKIEGO

www.wuj.pl

