# A New Function for Ensemble Pruning

Souad Taleb Zouggar[1(✉)] and Abdelkader Adla[2]

[1] Department of Economics, University of Oran 2, Oran, Algeria
Souad.taleb@gmail.com
[2] Department of Computer Science, University of Oran 1, Oran, Algeria
adla.abdelkader@univ-oran.dz

**Abstract.** We propose in this work a new function named Diversity and Accuracy for Pruning Ensembles (DAPE) which takes into account both accuracy and diversity to prune an ensemble of homogenous classifiers. A comparative study with a diversity based method and experimental results on several datasets show the effectiveness of the proposed method.

**Keywords:** Data mining · Classification · Decision trees · Ensemble methods
Homogenous ensembles · Bagging · Pruning · Ensemble selection
Hill climbing

## 1 Introduction

Ensemble methods have been proposed to reduce the variance of individual classifiers and consists of two phases: (1) the models generation step in which models can be added without risk of overfitting; these models can be either homogenous or heterogeneous, in the homogenous case models are obtained from the same learning algorithm by varying parameters or using data resampling [6], manipulating input and/or output attributes [1, 7, 12, 17, 29]; (2) the combination step in which the models are aggregated by voting or weighted voting.

However, these methods are known to generate a large number of models which can lead to large storage space and considerable time for classification and prediction. Ensemble selection allows the size reduction of an ensemble consisting of predictive models using different measures based on the diversity and/or the accuracy of the models, this selection allows obtaining smaller ensembles with higher accuracies comparing to the initial ensembles.

Several selection methods have been proposed in the literature; these methods are essentially based on an evaluation function [3, 9, 24] that determines if a model $M$ contributes positively to boost the performances of the whole ensemble. The evaluation is made based on two important properties of ensemble which are diversity and accuracy. Diversity qualifies for a set of classifiers their ability to agree in greater number on good class predictions, and to disagree on classification errors. The diversity and accuracy properties are considered separately [9, 13, 24] or together [4, 11].

In this paper, we propose a novel evaluation function named Diversity and Accuracy for Pruning Ensembles (DAPE) based on both diversity and accuracy, the method is


F. Dargam et al. (Eds.): ICDSST 2018, LNBIP 313, pp. 181–190, 2018.
https://doi.org/10.1007/978-3-319-90315-6_15

applied on homogenous ensembles composed of C4.5 decision trees [26]. This method is based on a DHCEP (Directed Hill Climbing Ensemble Pruning) strategy with a multi-objective function to evaluate the relevance of an ensemble of trees. The function, used in a Hill Climbing process in Forward Selection (FS), allows selection of ensembles with the best compromise between maximum diversity and minimum error rate.

A comparative study of the proposed measure and the UWA measure [25] is carried out on datasets from the UCI Repository [2].

The paper is structured as follows: In Sect. 2 we present the bagging used for homogenous ensemble generation, the combination using weighted and unweighted voting and finally the hill climbing method used for the search of a solution. Section 3 gives a review on recent works in the same domain. Section 4 introduces the proposed method in details. The penultimate section contains a comparative study and experimental results on multiple datasets. Finally we conclude and give some future works.

## 2 Background

In this section, we highlight the basic elements used in this paper, namely, the bagging method used for the generation of the initial ensemble, aggregation by unweighted and weighted vote.

### 2.1 Diversification by Bagging

Bagging **B**ootstrap **Agg**regat**ing**, a resampling method introduced by Breiman in 1996 [6]. Given a learning sample $\Omega_L$ and a learning method which generates a predictor $\hat{h}(.,\Omega_L)$ using $\Omega_L$. The principle of bagging is to draw several bootstrap samples $\left(\Omega_L^{\theta 1}, \ldots, \Omega_L^{\theta q}\right)$ and generate for each one a collection of predictors $(\hat{h}(.,\Omega_L^{\theta 1}), \ldots, \hat{h}(.,\Omega_L^{\theta q}))$ using the base learning method for finally aggregating them.

A bootstrap sample $\Omega_a^l$ is obtained by randomly drawing $n$ observations in the starting sample $\Omega_L$. Each observation have the probability of $1/n$ of being shot; $|\Omega_L| = n$, the random variable $\theta_l$ represents the random drawing.

Initially, Bagging was introduced with a decision tree as basic rule. But the schema is general and can be apply to other basic rules. Bagging transforms an unstable algorithm into a rule with very good properties (consistency and optimal speed of convergence) [5] (Fig. 1).

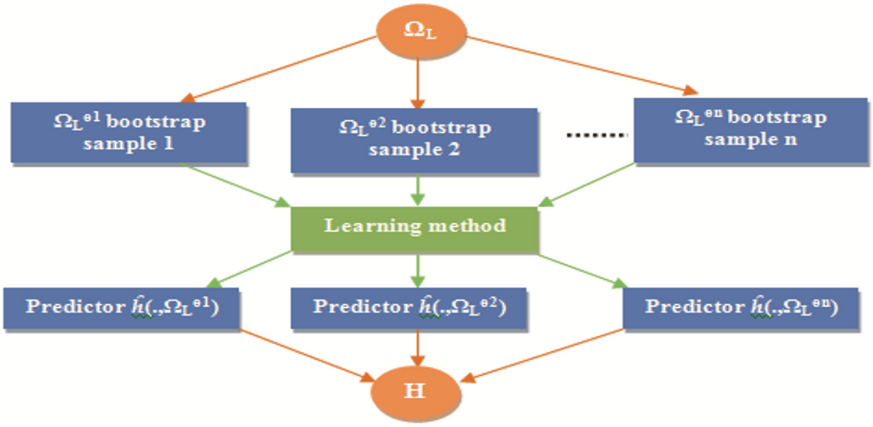


**Fig. 1.** Representative diagram of bagging.

### 2.2 Aggregation (Unweighted Vote, Weighted Vote)

The unweighted and weighted voting are the most used methods for combining (aggregating) whether homogenous or heterogeneous models. In ensemble methods each model, for an instance, gives a class value, a probability, and the class with most votes, highest average probability is assigned to the instance by the ensemble.

In weighted vote, the classification models are associated with weights assigned relatively to their classification accuracy. Formally this can be written: [25] Let $x$ be an instance and $m_{i,i=1..k}$ a set of models that output a probability distribution $m_i(x, c_j)$ for each class $c_{j,j=1...n}$. The output of the (weighted) voting method $y(x)$ for instance $x$ is given by the following mathematical expression:

$$y(x) = argmax_{c_j} \sum_{i=1}^{k} w_i m_i(x, c_j) \tag{1}$$

Where $w_i$ is the weight of model $i$. In the simple case of voting (unweighted), the weights are all equal to one, that is, $w_{i,i=1...k} = 1$.

### 2.3 The Hill Climbing

Hill climbing is an optimization technique belonging to the family of local search. The algorithm starts with any solution to a problem, and then tries iteratively to find a better solution by changing one element of the solution. If the change produces a better solution (maximize or minimize the evaluation function used for the course), an incremental change is made to the new solution. The process is repeated until no improvements can be found (the function reached the maximum or the minimum).

Hill climbing attempts to maximize (or minimize) a target function $f(X)$ where X is a vector of continuous and/or discrete values. Each iteration, hill climbing will adjust a single element in X and determine if the change improves the value of $f(X)$. Any change

improving the function $f(X)$ is accepted, the process continues until no amelioration of the function can be found.

For ensemble selection, DHCEP (Directed Hill Climbing Ensemble Pruning) is used, in this case the vector X is composed of classifiers or predictors. The path can be realized either in backward elimination or in forward selection, in the first case the whole ensemble is considered as a solution and then repeatedly elements not improving the evaluation function are eliminated one by one, in the second case we initialize with an element randomly and we add the elements that improve the evaluation function one by one. The elements to be added or removed are part of the neighborhood of the current solution (Fig. 2).

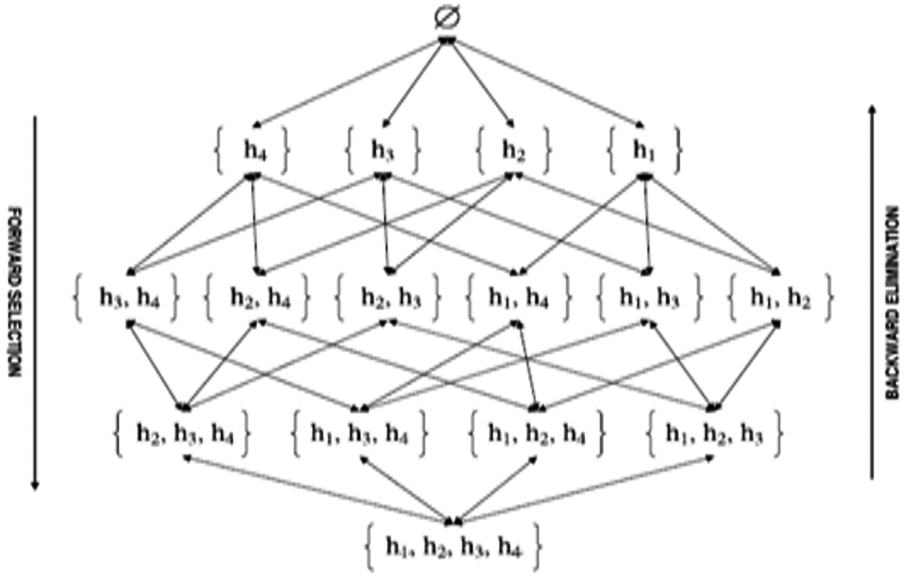


**Fig. 2.** Hill climbing search for selection in an ensemble composed of four classifiers [25].

## 3 Related Work

Several methods have been proposed to reduce the size of a set of classifiers [14, 16, 18, 19, 21, 23, 27]. The methods differ from each other by the adopted research directions, the different evaluation measures or the evaluation ensembles used.

A first type of approaches uses performance measures. Fan et al. [13] propose a profit-based evaluation function and propose dynamic scheduling to accelerate the prediction process. For the reduction of the ensemble size, the total benefit is used as selection criterion in conjunction with a greedy search algorithm with and without back fitting. The path begins with the model with the greatest benefit. A set of instances $x$ is considered, each instance (x) can be positive or negative, $B(x)$ denotes the benefit of predicting $x$ as positive and the total benefit BT = $\sum_x B(x)$, the authors choose the sub ensembles which maximize the total benefit. Caruana et al. [9] use several performance

metrics and a hill climbing strategy for building ensembles of models from libraries of thousands of models. Model libraries are generated using different learning algorithms. The Forward Stepwise selection is used to add to the ensemble the models that maximize its performance.

The second type of approaches uses diversity-based measures [3, 20, 24]. Partalas et al. [25] propose a measure of diversity that considers all the cases that may exist when adding a model $h_t$ to an ensemble. The measure Uncertainty Weighted Accuracy (UWA) considers four cases when adding and using justified weights to distinguish favor cases from others.

$$UWA_{Eval}(h_k, Sub) = \sum_{i=1}^{|Eval|} (\alpha * I(y_i = h_k(x_i) ETy_i \neq Sub(x_i)) - \beta * I(y_i \neq h_k(x_i) ETy_i = Sub(x_i)) + \beta * I(y_i = h_k(x_i) ETy_i = Sub(x_i)) - \alpha * I(y_i \neq h_k(x_i) ETy_i \neq Sub(x_i))) \quad (2)$$

The parameters α, β represent respectively the number of models in the sub-set Sub correctly classifying the instance $(x_i, y_i)$ and the number of models incorrectly classifying the same instance.

Li et al. [22] theoretically deals with the effect of diversity on voting generalization performance using Probably Approximately Correct (PAC) learning. It is revealed that diversity is closely related to the space complexity hypothesis, and strengthening it can be achieved by applying regularization to ensemble methods. Based on this analysis, the authors apply an explicit regularization of the diversity for the selection of ensembles.

Zhou et al. [30] propose a new algorithm based on frequent item learning that links data and the simplified ensemble to a transactional database whose transactions are instances and items are classifiers. A Boolean classification matrix is used for each model of the pruned ensemble. Using this matrix, several candidate ensembles are obtained by iterative and incremental extraction of basic classifiers with the best performances.

Bhatnagar et al. [4] perform ensemble selection using a performance-based and diversity-based function that considers the individual performance of classifiers as well as the diversity between pairs of classifiers. A bottom-up search is performed to generate the sub ensembles by adding various pairs of classifiers with high performance.

Cavalcanti et al. [10] combine in pairs different matrices of diversity using a genetic algorithm. The combined diversity matrix is then used to group similar (not very diverse) models; they must not belong to the same ensemble. To generate candidate ensembles, the combined diversity matrix is transformed into one or more graphs and then a graph coloring technique is applied.

Guo et al. [15] propose a new metric using the margin (instances) and the diversity (of classifiers) to explicitly evaluate the importance of individual classifiers. By adding the models to the ensemble in decreasing order of the metric, the user can choose the first T models to form a sub-ensemble.

Qun et al. [11] emphasize the utility of optimizing predictive performance together with diversity, which are two indispensable and inseparable parameters for ensemble selection. There have been three measures proposed to simplify ensembles using a greedy algorithm: (1) The first measure simultaneously considers the difference

(diversity) between the current subset and the candidate classifier and the performance of each one; (2) The second allows evaluating the diversity within the ensemble and; (3) the last measure reinforces the concern about the accuracy of the resulting sub-ensemble. Experimental results confirm the interest of the three measures which is illustrated by the improvement of performances.

## 4 The Proposed Method

The set of data Ω is divided into two sub samples $\Omega_L$ (generally 80% of Ω) for learning and pruning and $\Omega_T$ (generally 20% of Ω) for testing. A bagging ensemble BE of $t$ C4.5 trees is constructed, BE $= \{T_1, \ldots, T_i, \ldots, T_t\}$, using $\Omega_L$ with $|\Omega_L| = n$. Each tree $T_i$ is represented by a vector $(x_{1i}, x_{2i}, \ldots, x_{ji}, \ldots, x_{ni})^T$. We have the following notations:

- $x_{ij}$: Result of classification of the individual $i$ by the tree $j$, $x_{ij} = 1$ if the individual $i$ is misclassified by the tree $T_j$ and $x_{ij} = 0$ otherwise,
- $x_{i+}$: The total number of errors committed for the individual $I$:

$$x_{i+} = \sum_{j=1}^{t} x_{ij} \tag{3}$$

- $X$: The total number of errors committed by the set,

$$X = \sum_{i=1}^{n} \sum_{j=1}^{t} x_{ij} \tag{4}$$

- $(\Theta_i, x_{i+})$: The relative distribution of the error frequencies associated with the different cases,

$$\theta_i = \frac{x_{i+}}{X},\ i = 1, n \tag{5}$$

- $x_{+j}$: The number of errors committed by the classifier $Tj$ over all the individuals,

$$x_{+j} = \sum_{i=1}^{n} x_{ij} \tag{6}$$

- $e_j$: The error rate associated with the tree $T_j$,

$$e_j = \frac{x_{+j}}{n} \tag{7}$$

The evaluation function to optimize noted $S$ connects diversity $\Theta_i$ and the error rate $e_j$ (α is a parameter for which values are chosen empirically):

$$S = \sum_{i=1,n} \theta_i^2 + \alpha \sum_{j=1,t} e_j^2 \tag{8}$$

The algorithm below presents the proposed method DAPE in a pseudocode:

```
Algorithm DAPE;

Input
B = {T1,...,Tt};
ΩT: selection set;
Neighborhood(Ψj): Function that returns the subsets of
models obtained from Ψj by adding a classifier (tree);
Output
Sub ensemble Ψ0 of B;
Begin
Initialize(Ψ0);
1. Calculate S(Ψ0,ΩT);
   If ∃ Ψj such as S(Ψj,ΩT) < S(Ψ0,ΩT) where Ψj ∈
   Neighborhood(Ψ0)Then Ψ0=argminΨj(S(Ψj,ΩT));
   Goto 1;
End.
```

## 5 Experiments

The experiments consist in building homogeneous sets by sampling the starting sample and using the C4.5 decision tree generation algorithm as a basic rule [26]. The Weka platform [28] is used as a source for the C4.5 learning algorithm and validation. We consider 10 data sets from the UCI Repository [2] which are described in Table 1:

**Table 1.** Description of datasets.

| | Number of instances | Number of descriptors | Class modalities |
|---|---|---|---|
| Breast w | 699 | 9 | 2 |
| Tic tac toe | 958 | 9 | 2 |
| Dermatology | 366 | 34 | 6 |
| ecoli | 336 | 7 | 8 |
| kr-vs-kp | 3196 | 36 | 2 |
| glass | 214 | 9 | 6 |
| heart-h | 294 | 13 | 5 |
| hepatitis | 155 | 19 | 2 |
| ionosphere | 351 | 34 | 2 |
| lymph | 148 | 18 | 4 |

The proposed method, DAPE, is compared to the ensemble pruning method based on diversity UWA [25] detailed in literature and whose source code is available at http://mlkd.csd.auth.gr/ensemblepruning.html. For two methods, the unweighted majority vote is used for the combination of models and the performance calculation.

The methods use a forward selection strategy in a hill climbing scheme. The stop criterion for UWA is the performance on the evaluation sample which generates subsets

of reduced sizes compared with the usual stop criteria defined as a fixed number of models [24]. In our case we use the same function for both the path and the stop.

We use two criteria to compare the pruning methods; the performance and the size of the subsets obtained, ALL designs the ensemble composed of all the models (Tables 2 and 3).

**Table 2.** Comparativestudyof DAPE and UWA based on accuracy of obtained sub ensembles.

| | DAPE | UWA | ALL |
|---|---|---|---|
| Breast w | **0.97459** | 0.95032 | 0.95282 |
| Tic tac toe | 0.94205 | **0.95338** | 0.9333 |
| Dermatology | 0.94861 | **0.96712** | 0.91369 |
| ecoli | **0.84912** | 0.84326 | 0.83429 |
| kr-vs-kp | 0.98622 | 0.99433 | **0.9962** |
| glass | **0.76882** | 0.7619 | 0.76509 |
| heart-h | **0.80951** | 0.79652 | 0.79658 |
| hepatitis | 0.81613 | 0.80965 | **0.81932** |
| ionosphere | **0.9235** | 0.90856 | 0.93425 |
| lymph | **0.8331** | 0.81376 | 0.76997 |
| Average success rates | **0.885165** | 0.87988 | 0.871551 |

**Table 3.** Comparative study of DAPE and UWA based on the size of obtained sub ensembles.

| | DAPE | UWA |
|---|---|---|
| Breast w | **9.8** | 11.7 |
| cmc | **18** | 48.7 |
| Dermatology | 10 | **8.7** |
| ecoli | 12.1 | **11.5** |
| kr-vs-kp | 7.5 | **6.2** |
| glass | **13** | 15.9 |
| heart-h | **13.2** | 15.7 |
| hepatitis | **10.1** | 13 |
| ionosphere | 10.6 | **6.7** |
| lymph | 10.9 | **10.6** |
| Average sizes | **11.31** | 12.31 |

For 6 out of 10 benchmarks, DAPE shows better performances compared to the UWA which scores only 2 better performances, the whole ensemble gives better performances in only 2 cases.

For the average success rate on all datasets, DAPE is ranked first with a rate of 0.885165, exceeding UWA with a rate of 0.5%, and the whole ensemble with 1%.

All the methods DAPE, DEAS and UWA allow obtaining subsets of reduced sizes for 5 cases, 50% of the cases. For the average size on the 10 datasets, DAPE is ranked first with an average size of 11.31.

## 6 Conclusion and Future Work

In this paper we presented a new evaluation function combining performance and diversity for selection in a homogeneous ensemble used in a search process based on climbing hill. The method was evaluated on several benchmarks and compared to the method UWA based only on diversity, the results show the superiority of the proposed method.

We used the DAPE method for the selection in heterogeneous ensembles; where the classifiers are not generated from the same learning algorithm and for the selection in random forest ensembles, knowing that a random forest ensemble improves the performance of a bagging [8].

We also propose to study the possibility of using another search strategy for the selection in order to reduce the search time of the ensembles because the hill climbing strategy requires a non-negligible time when the number of models becomes important.

The last point consists in finding a value for the parameter $\alpha$ for which we have noticed during empirical research that for which an appropriate value could significantly improve the results as observed in our experiments.

## References

1. Amit, Y., Geman, D.: Shape quantization and recognition with randomized trees. Neural Comput. **9**, 1545–1588 (1997)
2. Asuncion, A., Newman, D.J.: UCI machine learning repository (2007). http://www.ics.uci.edu/mlearn/MLRepository.html
3. Banfield, R.E., Hall, L.O., Bowyer, K.W., Kegelmeyer, W.P.: Ensemble diversity measures and their application to thinning. Inf. Fusion **6**(1), 49–62 (2005)
4. Bhatnagar, V., Bhardwaj, M., Sharma, S., Haroon, S.: Accuracy-diversity based pruning of classifier ensembles. Prog. Artif. Intell. **2**(2–3), 97–111 (2014)
5. Biau, G., Cérou, F., Guyader, A.: On the rate of convergence of the bagged nearest neighbor estimate. J. Mach. Learn. Res. **11**, 687–712 (2010)
6. Breiman, L.: Bagging predictors. Mach. Learn. **26**(2), 123–140 (1996)
7. Breiman, L.: Randomizing outputs to increase prediction accuracy. Mach. Learn. **40**, 229–242 (2000)
8. Breiman, L.: Random forests. Mach. Learn. **45**, 5–32 (2001)
9. Caruana, R., Niculescu-Mizil, A., Crew, G., Ksikes, A.: Ensemble selection from libraries of models. In: Proceedings of the 21st International Conference on Machine Learning (2004)
10. Cavalcanti, G.D.C., Oliveira, L.S., Moura, T.J.M., Carvalho, G.V.: Combining diversity measures for ensemble pruning. Pattern Recognit. Lett. **74**, 38–45 (2016). ISSN 0167-8655
11. Qun, D., Ye, R., Liu, Z.: Considering diversity and accuracy simultaneously for ensemble pruning. Appl. Soft Comput. **58**, 75–91 (2017). ISSN 1568-4946
12. Dietterich, T.G., Kong, E.B.: Machine learning bias, statistical bias, and statistical variance of decision tree algorithms. Technical report, Dept of Computer Science, Oregon State University, Covallis, Oregon (1995)
13. Fan, W., Chu, F., Wang, H., Yu, P.S.: Pruning and dynamic scheduling of cost-sensitive ensembles. In: Eighteenth National Conference on Artificial Intelligence, pp. 146–151. American Association for Artificial Intelligence (2002)
14. Fu, Q., Hu, S.X., Zhao, S.Y.: Clustering-based selective neural network ensemble. J. Zhejiang Univ. Sci. A **6**(5), 387–392 (2005)

15. Guo, H., Liu, H., Li, R., Wu, C., Guo, Y., Xu, M.: Margin & diversity based ordering ensemble pruning. Neurocomputing **275**, 237–246 (2017). ISSN 0925-2312
16. Hernández-Lobato, D., Martínez-Munoz, G.: A statistical instance-based pruning in ensembles of independent classifiers. IEEE Trans. Pattern Anal. Mach. Intell. **31**(2), 364–369 (2009)
17. Ho, T.K.: The random subspace method for constructing decision forests. IEEE Trans. Pattern Anal. Mach. Intell. **20**(8), 832–844 (1998)
18. Margineantu, D., Dietterich, T.G.: Pruning adaptive boosting. In: Proceedings of the 14th International Conference on Machine Learning, pp. 211–218. Morgan Kaufmann, San Francisco (1997)
19. Markatopoulou, F., Tsoumakas, G., Vlahavas, I.: Instance-based ensemble pruning via multi-label classification. In: ICTAI 2010 (2010)
20. Martínez-Muñoz, G., Suárez, A.: Aggregation ordering in bagging. In: Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, pp. 258–263. Acta Press (2004)
21. Martínez-Muñoz, G., Hernández-Lobato, D., Suárez, A.: Selection of decision stumps in bagging ensembles. In: de Sá, J.M., Alexandre, L.A., Duch, W., Mandic, D. (eds.) ICANN 2007. LNCS, vol. 4668, pp. 319–328. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74690-4_33
22. Li, N., Yu, Y., Zhou, Z.-H.: Diversity regularized ensemble pruning. In: Flach, P.A., De Bie, T., Cristianini, N. (eds.) ECML PKDD 2012. LNCS (LNAI), vol. 7523, pp. 330–345. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33460-3_27
23. Partalas, I., Tsoumakas, G., Katakis, I., Vlahavas, I.: Ensemble pruning using reinforcement learning. In: Antoniou, G., Potamias, G., Spyropoulos, C., Plexousakis, D. (eds.) SETN 2006. LNCS (LNAI), vol. 3955, pp. 301–310. Springer, Heidelberg (2006). https://doi.org/10.1007/11752912_31
24. Partalas, I., Tsoumakas, G., Vlahavas, I.: Focused ensemble selection: a diversity-based method for greedy ensemble selection. In: Ghallab, M., Spyropoulos, C.D., Fakotakis, N., Avouris, N.M. (eds.) ECAI 2008 - 18th European Conference on Artificial Intelligence. Proceedings of the Frontiers in Artificial Intelligence and Applications, Patras, Greece, 21–25 July 2008, vol. 178, pp. 117–121. IOS Press (2008)
25. Partalas, I., Tsoumakas, G., Vlahavas, I.: An ensemble uncertainty aware measure for directed hill climbing ensemble pruning. Mach. Learn. **81**, 257–282 (2010)
26. Quinlan, J.R.: C4.5 Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
27. Soto, V., Martínez-Muñoz, G., Hernández-Lobato, D., Suárez, A.: A double pruning algorithm for classification ensembles. In: El Gayar, N., Kittler, J., Roli, F. (eds.) MCS 2010. LNCS, vol. 5997, pp. 104–113. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12127-2_11
28. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann, Los Altos (2005)
29. Zheng, Z., Webb, G.I.: Stochastic attribute selection committees. Technical report (TR C98/08), School of Computing and Mathematics, Deakin University, Australia (1998)
30. Zhou, H., Zhao, X., Wang, X.: An effective ensemble pruning algorithm based on frequent patterns. Knowl.-Based Syst. **56**, 79–85 (2014). ISSN 0950-7051