An efficient ensemble pruning approach based on simple coalitional games

Hadjer Ykhlef[a,*], Djamel Bouchaffra[b]

[a]*Department of Computer Science, University of Blida, Algeria.*
[b]*Design of Intelligent Machines Group, CDTA.*

Abstract

We propose a novel ensemble pruning methodology using non-monotone Simple Coalitional Games, termed SCG-Pruning. Our main contribution is two-fold: (1) Evaluate the diversity contribution of a classifier based on Banzhaf power index. (2) Define the pruned ensemble as the minimal winning coalition made of the members that together exhibit moderate diversity. We also provide a new formulation of Banzhaf power index for the proposed game using weighted voting games. To demonstrate the validity and the effectiveness of the proposed methodology, we performed extensive statistical comparisons with several ensemble pruning techniques based on 58 UCI benchmark datasets. The results indicate that SCG-Pruning outperforms both the original ensemble and some major state-of-the-art selection approaches.

Keywords: Ensemble pruning, Simple coalitional game, Banzhaf power index, Weighted voting game, Diversity.

1. Introduction

Ensemble learning remains a challenging task within the pattern recognition and machine learning community [1–4]. A large body of literature has shown that a combination of multiple classifiers is a powerful decision making tool, and usually generalizes better than a single classifier [5–7]. Ensemble learning builds a classification

*Corresponding author. Tel.: +213 792246942
Email addresses: ykhlef.hadjer@gmail.com (Hadjer Ykhlef), djamel.bouchaffra@gmail.com (Djamel Bouchaffra)

model in two steps. The first step concerns the generation of the ensemble members (also called team, committee, and pool). To this end, several methods such as: boosting [5], bagging [6], random subspace [8], and random forest [9] have been introduced in the literature. In the second step, the predictions of the individual members are merged together to give the final decision of the ensemble using a combiner function. Major combining strategies include: majority voting [6], performance weighting [5], stacking [6], and local within-class accuracies [10]. Ensemble learning has demonstrated a great potential for improvement in many real-world applications such as: remote sensing [1], face recognition [2], intrusion detection [3], and information retrieval [4].

It is well-accepted that no significant gain can be obtained by combining multiple identical learning models. On the other hand, an ensemble whose members make errors on different samples reaches higher prediction performance [5, 6]. This concept refers to the notion of *diversity* among the individual classifiers. Unfortunately, the relationship between diversity and the ensemble generalization power remains an open problem. As suggested by many authors [5, 11, 12], an ensemble composed of highly diversified members may result in a better or worse performance. In other words, diversity can be either harmful or beneficial and therefore requires an adequate quantification. As a matter of fact, it has been demonstrated that maximizing diversity measures does not necessarily have a positive impact on the prediction performance of the committee [13].

Despite their remarkable success, ensemble methods can negatively affect both the *predictive performance* and the *efficiency* of the committee. Specifically, most techniques for growing ensembles tend to generate an unnecessarily large number of classifiers in order to guarantee that the training error rate reaches its minimal value. This necessity may result in overfitting the training set, which in turn causes a reduction in the generalization performance of the ensemble. Furthermore, an ensemble made of many members incurs an increase in memory requirement and computational cost. For instance, an ensemble made of C4.5 classifiers can require large memory storage [14]; a set of lazy learning methods, such as k-nearest neighbors and K*, may increase the prediction time. The memory and computational costs appear to be negligible for toy datasets, nevertheless they can become a serious problem when applied to real-world

applications such as learning from data stream.

All the above reasons motivate the appearance of ensemble pruning approaches (also called ensemble shrinking, ensemble thinning, and ensemble selection). Ensemble pruning aims at extracting a subset of classifiers that optimizes a criterion indicative of a committee generalization performance. Given an ensemble composed of n classifiers, finding a subset that yields the best prediction performance requires searching the space of $2^n - 2$ non-empty subsets, which is intractable for large ensembles. This problem has been proven to be NP-complete [7]. To alleviate this computational burden, many ensemble pruning approaches have been introduced in the literature. Most of these techniques fall into three main categories: ranking-based, optimization-based, and clustering-based approaches. Please, refer to the related work subsection for additional details.

Based on these insights, this paper considers the problem of ensemble pruning as a Simple Coalitional Game (SCG). The proposed methodology aims at extracting sub-ensembles with moderate diversities while ignoring extreme scenarios: strongly correlated and highly diversified members. This mission is achieved in three steps: (1) We formulate ensemble pruning as a non-monotone SCG played among the ensemble members. (2) We evaluate the *power* or the *diversity contribution* of each ensemble member using Banzhaf power index. (3) We define the pruned ensemble as the *minimal winning coalition* constituted of the best ranked members. It is worth underscoring that the original definition of Banzhaf power index for non-monotone SCGs is intractable. Specifically, given a n-player game, the calculation of Banzhaf power index involves summing over 2^{n-1} coalitions, which is unfeasible for large values of n. To overcome this computational difficulty, we introduce a *new formulation of Banzhaf power index* for the proposed game, and show that its time complexity is pseudo-polynomial.

1.1. Related work

Tsoumakas et al. classified the ensemble pruning approaches into 4 categories [15]:

1.1.1. Ranking-based approaches

Methods of this category first assign a rank to every classifier according to an evaluation measure (or criterion); then, the selection is conducted by aggregating the ensemble members whose ranks are above a predefined threshold. The main challenge a ranking-based method faces, consists of adequately setting the criterion used for measuring every member's contribution to the ensemble performance. For instance, Margineantu and Dietterich introduced *Kappa pruning*, which selects a subset made of the most diverse members of the ensemble [14]. Specifically, it first measures the agreement between all pairs of classifiers using kappa statistic; it then selects the pairs of classifiers starting with the one which has the lowest kappa statistic (high diversity), and it considers them in ascending order of their agreement until the desired number of classifiers is reached.

Zheng Lu et al. proposed to estimate each classifier's contribution based on the diversity/accuracy tradeoff [16]. Then, they ordered the ensemble members according to their contributions in descending order. In the same regard, Ykhlef and Bouchaffra formulated ensemble pruning problem as an induced subgraph game [17]. Their approach first ranks every classifier by considering the ensemble diversity and the individual accuracies based on Shapley value; then, it constitutes the pruned ensemble by aggregating the top N members.

Galar et al. introduced several criterions for ordering ensemble members in the context of imbalanced classification [18]. They investigated and adapted five well-known approaches: Reduce error [14], Kappa pruning [14], Boosting-based [19], Margin distance minimization [20], and Complementarity measure [20].

1.1.2. Optimization-based approaches

This category formulates ensemble pruning as an optimization problem. A well-known method of this category is GENETIC ALGORITHM BASED SELECTIVE ENSEMBLE (GASEN) [21]. This technique assigns a weight to each classifier; a low value indicates that the associated individual member should be excluded. These weights are initialized randomly, and then evolved toward an optimal solution using *genetic algorithm*. The fitness function is computed based on the corresponding ensemble performance on

a separate sample set. Finally, pruning is conducted by discarding members whose weights are below a predefined threshold.

Zhang et al. formulated ensemble pruning as a *quadratic integer programming* problem that considers the diversity/accuracy tradeoff [22]. Since this optimization problem is NP-hard, they used *semi definite programming* on a relaxation of the original problem to efficiently approximate the optimal solution.

Rokach introduced Collective Agreement-based ensemble Pruning (CAP), a criterion for measuring the goodness of a candidate ensemble [23]. CAP is defined based on two terms: member-class and member-member agreement. The first term indicates how much a classifier's predictions agree with the true class label, whereas the second term measures the agreement level between two ensemble members. This metric promotes sub-ensembles whose members highly agree with the class and have low inter-agreement among each other. Note that CAP provides only a criterion for measuring the goodness of a candidate ensemble in the solution space, and hence requires defining a search strategy like best-first or directed hill climbing [6, 15].

1.1.3. Clustering-based approaches

The key idea behind this category consists of invoking a clustering technique, which allows identifying a set of representative *prototype* classifiers that compose the pruned ensemble. A clustering-based method involves two main steps. In the first step, the ensemble is partitioned into clusters, where individual members in the same cluster make similar predictions (strong correlation), while classifiers from different clusters have large diversity. For this purpose, several clustering techniques such as k-means [24], hierarchical agglomerative clustering [25], and deterministic annealing [26] have been proposed. In the second step, each cluster is separately pruned in order to increase the diversity of the ensemble. For example, Bakker and Heskes selected the individual members at the *centroid* of each cluster to compose the pruned ensemble [26].

1.1.4. Other approaches

This category comprises the pruning approaches that do not belong to any of the above categories. For example, Partlas et al. [27] considered the ensemble pruning

problem from a *reinforcement learning* perspective; Martínez-Muñoz et al. used AdaBoost to prune an ensemble trained by Bagging [19].

1.2. Contributions and outline

The contribution of the proposed research is described by the following tasks:

(1) We propose a novel methodology for pruning an ensemble of learning models based on the minimal winning coalition and Banzhaf power index.

(2) We present a new representation for non-monotone SCGs and provide, under some restrictions, a pseudo-polynomial time algorithm for computing Banzhaf power index.

(3) We show the efficiency of the proposed methodology through extensive experiments and statistical tests using a large set of 58 UCI benchmark datasets.

The rest of this paper is organized as follows. Some diversity measures are defined in Section 2. Necessary concepts from coalitional game theory are described in Section 3. The proposed methodology is presented in Section 4. The experiments are conducted on benchmark datasets, and the results are discussed in Section 5. Finally, conclusions and future work are laid out in Section 6.

2. Diversity measures

Disagreement measure : Given two classifiers h_i and h_j, the disagreement measure [5] is given by:

$$dis_{i,j} = \frac{N^{01} + N^{10}}{N^{11} + N^{00} + N^{01} + N^{10}}, \tag{1}$$

where N^{11}, N^{00}, N^{01}, and N^{10} denote the number of correct/incorrect predictions made by h_i and h_j on the training set (Table 1). Note that a high value of $dis_{i,j}$ corresponds to large diversity between h_i and h_j. Consequently, the diversity function f is defined as:

$$f(h_i, h_j) = dis_{i,j}. \tag{2}$$

Cohen's kappa : Given two classifiers h_i and h_j, Cohen's kappa [5] is defined as:

$$\kappa_{i,j} = \frac{\theta_{i,j} - \vartheta_{i,j}}{1 - \vartheta_{i,j}}, \tag{3}$$

where $\theta_{i,j}$ is the proportion of samples on which both h_i and h_j make the same predictions on the training set, and $\vartheta_{i,j}$ corresponds to the probability that the two classifiers agree by chance. The diversity function f is given by:

$$f(h_i, h_j) = \frac{1}{\kappa_{i,j} + \varepsilon}. \tag{4}$$

A small positive constant ε is introduced to avoid numerical difficulties when the kappa statistic approaches zero.

Mutual information : Brown et al. [28] used mutual information to assess the diversity between two classifiers. They proposed the following expansion: First, let X_i, X_j and Y be three discrete random variables designating the predictions of two classifiers h_i and h_j on the training set and the true class label, respectively. Then, the diversity function f is given by:

$$f(h_i, h_j) = I(X_i; X_j|Y) - I(X_i; X_j), \tag{5}$$

where $I(X_i; X_j|Y)$ and $I(X_i; X_j)$ denote the conditional mutual information and the mutual information, respectively.

Table 1. The number of correct/incorrect predictions made by a pair of classifiers.

	h_j **correct**	h_j **wrong**
h_i **correct**	N^{11}	N^{10}
h_i **wrong**	N^{01}	N^{00}

3. Coalitional game theory: some definitions

Coalitional Game Theory (CGT) [29] models situations that involve interactions among decision-makers, called *players*. The focus is on the outcomes achieved by

groups rather than by individuals. We call each group of players a *coalition*, where $\emptyset$ corresponds to the *empty coalition*, and the set of all players is the *grand coalition*.

Definition 3.1. A simple coalitional game G is a pair (N, v) consisting of a finite set of players $N = \{1, 2, ..., n\}$, and a characteristic function (a.k.a payoff function) $v : 2^N \mapsto \{0, 1\}$, where 2^N denotes the set of all possible coalitions that can be formed. We say a coalition $S \subseteq N$ wins if $v(S) = 1$ and loses if $v(S) = 0$. If in a simple game $v(T) = 1 \Rightarrow v(S) = 1$ for all $T \subseteq S \subseteq N$, then the characteristic function v is said to be *monotone*.

A straightforward representation of a simple coalitional game consists of enumerating the payoffs for all coalitions $S \subseteq N$. However, this naïve representation requires space exponential in the number of players $|N| = n$, which is impractical in most cases. To alleviate this tractability issue, several representations for coalitional games such as marginal contribution nets, network flow games, and weighted voting games [30] have been proposed in the literature. In this work, we consider only weighted voting games.

Definition 3.2. A weighted voting game G is defined by a set of players $N = \{1, ..., n\}$, a list of weights $\mathbf{w} = (w_1, w_2, ..., w_n) \in \mathbb{R}^n_+$, and a threshold $q \in \mathbb{R}_+$ also known as *quota*; we write $G = (N, [\mathbf{w}, q])$. The payoff function is given by: $v(S) = 1$ if $\sum_{i \in S} w_i \geq q$, and $v(S) = 0$ otherwise.

3.1. Banzhaf power index

Definition 3.3. Given a simple coalitional game $G = (N, v)$, Banzhaf index [31], denoted $Bz_i(G)$, measures the *power* controlled by a player i. Formally, it is defined as:

$$Bz_i(G) = \frac{1}{2^{n-1}} \sum_{S \subseteq N \setminus \{i\}} (v(S \cup \{i\}) - v(S)) . \tag{6}$$

Banzhaf index of non-monotone simple games has an interesting interpretation, but before analyzing it, we need to introduce two concepts: *positive* and *negative swings*.

Definition 3.4. A coalition $S \subseteq N$ is a *positive swing* for player i if $S \cup \{i\}$ wins ($v(S \cup \{i\}) = 1$) and S loses ($v(S) = 0$). Conversely, the coalition S is considered as

a *negative swing* for player i if $v(S \cup \{i\}) = 0$ and $v(S) = 1$. Let $swing_i^+$ and $swing_i^-$ denote, respectively, the set of positive and negative swing coalitions for player i. They are defined as:

$$swing_i^+ = \{S \subseteq N \setminus \{i\} | v(S \cup \{i\}) = 1 \wedge v(S) = 0\}. \tag{7}$$

$$swing_i^- = \{S \subseteq N \setminus \{i\} | v(S \cup \{i\}) = 0 \wedge v(S) = 1\}. \tag{8}$$

Since the characteristic function of a simple game is Boolean, the computation of Banzhaf power index is reduced to a counting problem. It suffices to identify all possible values of the formula $v(S \cup \{i\}) - v(S)$, count and sum them. Due to the non-monotonicity property, $v(S \cup \{i\}) - v(S)$ has three possible values: -1,$+1$, and 0. We are only interested in counting the number of ones θ_1 and negative ones θ_{-1}. Notice that θ_1 and θ_{-1} correspond to the number of positive and negative swing coalitions, respectively. Therefore, Banzhaf power index is proportional to the difference between the number of positive and negative swing coalitions. Formally, Banzhaf index of player i can be given by:

$$Bz_i(G) = \frac{1}{2^{n-1}} \times (|swing_i^+| - |swing_i^-|). \tag{9}$$

4. Ensemble pruning approach based on simple coalitional games

4.1. Notations

Let $\Omega = \{h_1, h_2, ..., h_n\}$ be an ensemble made of n classifiers. Every learner is provided with a set of m labeled samples $\Gamma = \{(x_1, y_1), ..., (x_m, y_m)\}$, where $x_i \in \mathcal{X}$ denotes a vector of feature values characterizing the instance i, and $y_i \in \mathcal{Y}$ is the true class label. The learning algorithm induces from Γ a hypothesis h that predicts the class label of a sample x. Given a feature vector x, the ensemble Ω combines the predictions of its members $h_1(x), ..., h_n(x)$ using a combiner function Θ. The combination method is responsible for turning the classifiers' private judgments into a collective decision. We assume that every ensemble member is trained separately using the same training

set Γ. *The problem of ensemble pruning consists of selecting from the ensemble* Ω *a subset* $\omega \subseteq \Omega$ *that yields the best predictive model i.e. with low generalization error, using the combiner method* Θ.

4.2. Ensemble pruning game

The concept of "diversity" is considered as the key success in constructing a committee of classifiers [5, 6]. According to Rokach [5], creating an ensemble of diversified learners lead to uncorrelated errors that boost the group performance globally. Unfortunately, efficiently measuring diversity and understanding its relationship with the classification generalization power of the committee remains an open problem [13, 28, 32]. Several experimental studies have shown that *large diversity* within an ensemble causes a sharp drop in its performance [11]. Furthermore, it is well-known that the action of building an ensemble of *identical classifiers* is ineffective. To seek a tradeoff between these two extreme effects, we propose a methodology that focuses on extracting a set of classifiers with average diversity. More specifically, we cast the problem of ensemble pruning as a simple game that captures several levels of classifiers' disagreement, and promotes average diversity over the other two extreme scenarios (correlation and high diversity). The various steps of SCG-Pruning are depicted by Fig. 1.

We begin this process by setting up a simple game G, built on the initial ensemble of classifiers Ω. A classifier h_i is considered as a player and is associated with a weight $w_i, i \in \{1, ..., n\}$. These weights are computed as follows. We define the *diversity contribution* of a classifier h_i, with respect to the entire ensemble Ω, as the average diversity between h_i and the rest of classifiers, which we denote by $Div_\Omega(h_i)$. In order to approximate high-order-diversity induced by a candidate classifier, we consider that the ensemble members exhibit only pairwise interactions.

Definition 4.1. The diversity contribution of a classifier $h_i \in \Omega$ is defined as:

$$Div_\Omega(h_i) = \frac{1}{n-1} \sum_{h_j \in \Omega \setminus \{h_i\}} f(h_i, h_j), \tag{10}$$

where $f : \Omega \times \Omega \mapsto \mathbb{R}$ assigns to a pair of classifiers (h_i, h_j) a real number that corresponds to the diversity between the decisions of h_i and h_j, with $f(h_i, h_i) = 0$ and

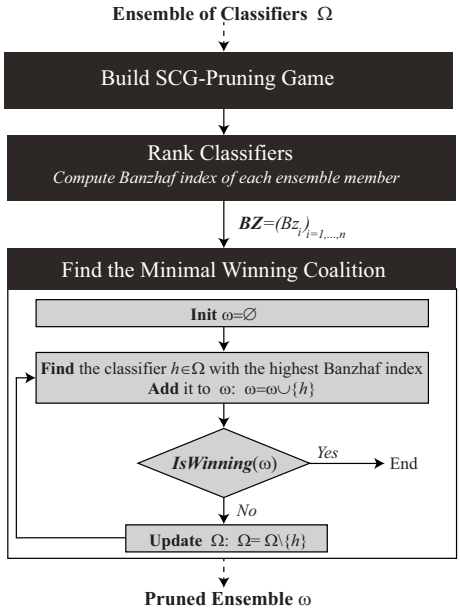


Fig. 1. The SCG-Pruning process.

$f(h_i, h_j) = f(h_j, h_i)$.

Definition 4.2. The weight w_i assigned to a classifier $h_i \in \Omega$ is given by:

$$w_i = \sum_{h_j \in \Omega \setminus \{h_i\}} \mathbb{I}(Div_\Omega(h_i) \geq Div_\Omega(h_j)), \tag{11}$$

where $\mathbb{I}(condition)$ denotes the indicator function, which equals 1 when *condition* is satisfied ($condition = true$), and 0 otherwise. It is noteworthy that each voting weight w_i can be thought as a *level of diversity induced by* h_i, in which highly diversified members receive higher weights.

In addition to the list of weights, we use two thresholds q_1 and q_2 to define the payoff function of the pruning game, such that $q_2 - q_1 > max_{h_i} w_i$ and $q_1 > max_{h_i} w_i$.

Definition 4.3. Given q_1 and q_2, the payoff function of the proposed game $G = (\Omega, [\mathbf{w}, q_1, q_2])$

is defined as:

$$v(S) = \begin{cases} 1 & if \quad q_1 \leq \sum_{h_i \in S} w_i \leq q_2 \\ 0 & Otherwise \end{cases} . \tag{12}$$

Under this payoff function, a coalition S of classifiers wins if the sum of its members' weights falls between q_1 and q_2. The term $\sum_{h_i \in S} w_i$ measures the *amount of diversity* present in S; a low value of this term corresponds to strong correlations between the ensemble members, whereas a large value indicates that the coalition is composed mainly of diversified classifiers. Furthermore, the interval $[q_1, q_2]$ corresponds to the width of *permitted diversity*, in which the lower bound q_1 controls the degree of correlation present in S, and the upper bound q_2 serves as barrier for highly diverse ensembles. Both extreme cases can decrease the generalization performance of the group [13]. When q_1 and q_2 are set properly, this payoff function ignores coalitions made of correlated classifiers (lower bound) and those highly diverse (upper bound). As a result, the focus will only be on groups with moderate diversities that can lead to better generalization performance [11].

Correctly setting the values of q_1 and q_2 is of vital importance for the success of the proposed methodology. We can distinguish two extreme cases: (i) *low values for* q_1 *and* q_2: in this case, the proposed technique focuses mainly on correlated ensembles; and (ii) *high values for* q_1 *and* q_2: this choice considers only ensembles composed of the most diverse members. One should avoid the configurations indicated by (i) and (ii), and set the values of q_1 and q_2 between these two extreme cases. The choice of q_1 and q_2 will be further discussed in the experiments section (subsection 5.1.4).

The next step consists of ranking each classifier based on Banzhaf power index. Under the SCG-Pruning game, the formulation of this solution concept (provided by equation 9) has an interesting interpretation that is summarized as follows. Let consider a coalition of correlated classifiers S, where $v(S) = 0$. If a classifier h_i induces the proper amount of diversity into a losing coalition S and turns it into a winning coalition ($v(S \cup \{h_i\}) = 1$), then h_i is *pivotal* for S and the coalition S is a positive swing for h_i. Conversely, the set of negative swing for a classifier h_i is defined as the ones in which h_i introduces large diversity into winning coalitions and changes their status

into losing coalitions. Therefore, Banzhaf power index assigns high ranks to members that induce diversity into correlated ensembles while penalizing members that exhibit strong disagreement with the group.

The exact and direct computation of Banzhaf index under this representation requires summing over all possible coalitions, which is exponential in the size of the initial committee , and is therefore intractable for large ensembles. To cope with the computational burden, we have investigated the relationship between the proposed game and other representations of simple games. As a result, we have expressed Banzhaf power index within the proposed framework in terms of Banzhaf indices of two weighted voting games (Theorem 4.2).

Theorem 4.1. *Consider the weighted voting game $G_1 = (\Omega, [\mathbf{w}, q_1])$, $Bz_i(G_1)$ player h_i's Banzhaf power index of G_1, and $|swing_i^+|$ the number of positive swing coalitions for h_i under the SCG-Pruning game G, then:*

$$|swing_i^+| = 2^{n-1} \times Bz_i(G_1).$$

Proof. Banzhaf power index of weighted voting games can be written as [33]:

$$\begin{aligned} Bz_i(G_1) &= \frac{1}{2^{n-1}} \times |\{S \subseteq \Omega \setminus \{h_i\} | v_1(S \cup \{h_i\}) = 1 \wedge v_1(S) = 0\}|. \\ &= \frac{1}{2^{n-1}} \times |\{S \subseteq \Omega \setminus \{h_i\} | \mathcal{W}(S) + w_i \geq q_1 \wedge \mathcal{W}(S) < q_1\}|. \end{aligned}$$

where $\mathcal{W}(S) = \sum_{h_j \in S} w_j$.

Since all weights are positive integers, we can write:

$$Bz_i(G_1) = \frac{1}{2^{n-1}} \times |\{S \subseteq \Omega \setminus \{h_i\} | q_1 - w_i \leq \mathcal{W}(S) < q_1\}|. \tag{13}$$

On the other hand, the set of positive swing coalitions for player h_i under G is given

by:

$$
\begin{aligned}
swing_i^+ &= \{S \subseteq \Omega \setminus \{h_i\} | v(S \cup \{h_i\}) = 1 \wedge v(S) = 0\}. \\
&= \{S \subseteq \Omega \setminus \{h_i\} | q_1 \leq \mathcal{W}(S) + w_i \leq q_2 \wedge \mathcal{W}(S) < q_1\}. \\
&= \{S \subseteq \Omega \setminus \{h_i\} | q_1 - w_i \leq \mathcal{W}(S) \leq q_2 - w_i \wedge \mathcal{W}(S) < q_1\}.
\end{aligned}
$$

Since $q_2 - q_1 > max_{h_i} w_i$, which implies $q_1 < q_2 - w_i$ for all $i \in \{1, ..., n\}$. Given this new consideration, $swing_i^+$ can be further simplified as:

$$swing_i^+ = \{S \subseteq \Omega \setminus \{h_i\} | q_1 - w_i \leq \mathcal{W}(S) < q_1\}.$$

Using Banzhaf power index formulation given by equation 13, one can write:

$$|swing_i^+| = 2^{n-1} \times Bz_i(G_1)\square.$$

Corollary 4.1.1. *Given the weighted voting game $G_2 = (\Omega, [\mathbf{w}, q_2 + 1])$, and player h_i's Banzhaf index $Bz_i(G_2)$, then the number of negative swing coalitions for h_i under the SCG-Pruning game G can be expressed as:*

$$|swing_i^-| = 2^{n-1} \times Bz_i(G_2).$$

Theorem 4.2. *Consider the two weighted voting games $G_1 = (\Omega, [\mathbf{w}, q_1])$ and $G_2 = (\Omega, [\mathbf{w}, q_2 + 1])$, then $Bz_i(G)$, player h_i's Banzhaf power index of the SCG-Pruning game G, can be simplified as:*

$$Bz_i(G) = Bz_i(G_1) - Bz_i(G_2).$$

Proof. From equation 9, we have:

$$Bz_i(G) = \frac{1}{2^{n-1}} \times (|swing_i^+| - |swing_i^-|).$$

Using Theorem 4.1 and Corollary 4.1.1, one obtains:

$Bz_i(G) = Bz_i(G_1) - Bz_i(G_2)\square$.

The last step of the SCG-Pruning methodology is to determine the pruned ensemble size L. For this purpose, we propose to map the pruned ensemble to the *minimal winning coalition* composed only of highly ranked classifiers. In CGT, the definition of the minimal winning coalition is outlined by Riker [34]:

"If a coalition is large enough to win, then it should avoid taking in any superfluous members, because the new members will demand a share in the payoffs. Therefore, one of the minimal winning coalitions should form. The ejection of the superfluous members allows the payoff to be divided among fewer players, and this is bound to be advantage of the remaining coalition members" [35].

Notice that this concept does not predict the coalition structure of the game, but it provides strong evidence that one of the minimal winning coalitions will form. Moreover, in political science, this concept refers to group that contains the smallest number of players which can *secure* a parliamentary majority. Putting these notions into the context of SCG-Pruning, the minimal winning coalition corresponds to the smallest sub-ensemble of classifiers that *together exhibit* moderate diversity.

4.3. The SCG-Pruning algorithm

The pseudo code of the proposed approach is depicted by Fig. 2. The SCG-Pruning method takes as input an initial ensemble of classifiers, two thresholds, and a training set. In addition, SCG-Pruning requires defining a pairwise function for estimating the classifiers' voting weights. For instance, the diversity between a pair of classifiers can be estimated using statistical measures [5, 14] like: Cohen's kappa, disagreement measure, Q-statistic, etc., or even information theoretic concepts [28, 32, 36]. The algorithm first computes the classifiers' predictions of every training sample (line [3-7]), and uses them to estimate the voting weights of the ensemble members (line [8-10]). Then, it ranks every individual learner based on Banzhaf power index (line [11-13]). Finally, it sets the pruned ensemble as the minimal winning coalition made of the

best ranked learners (line [14-18]). More specifically, the algorithm iteratively chooses, from among the classifiers not yet selected, the classifier with the highest rank, and adds it to the selected set ω until ω wins.

1: **Input:** Γ: Training set.
Ω: Ensemble of classifiers.
q_1, q_2: Two thresholds.
2: **Initialize:** $\omega = \emptyset$;
*/*Getting classifiers' predictions*/*
3: **For each** $h_i \in \Omega$
4: **For each** $(x_j, y_j) \in \Gamma$
5: $Preds^i_j = h_i(x_j)$;
6: **End for each** (x_j, y_j)
7: **End for each** h_i
*/*Estimating classifiers' weights based on Preds*/*
8: **For each** $h_i \in \Omega$
9: Compute w_i using formula 11;
10: **End for each** h_i
*/*Computing classifiers' Banzhaf indices*/*
11: **For each** $h_i \in \Omega$
12: $Bz_i(G) = Bz_i(G_1) - Bz_i(G_2)$;
13: **End for each** h_i
*/*Searching for the minimal winning coalition*/*
14: **Repeat**
15: $h = argmax_{h_i} Bz_i(G)$;
16: $\omega = \omega \cup \{h\}$;
17: $\Omega = \Omega \setminus \{h\}$;
18: **Until** $v(\omega) = 1$

19: **Output:** ω: Pruned ensemble.

Fig. 2. The SCG-Pruning algorithm.

4.4. Computational complexity

Note that the computational complexity of SCG-Pruning depends mainly on ranking the ensemble members using Banzhaf power index (line 12 of the SCG-Pruning algorithm). It is well-known that the exact computation of Banzhaf index for non-monotone simple games is exponential in the number of players n, which is intractable for large n [30]. Fortunately, under our representation, we were able to reduce that

problem into the estimation of Banzhaf power indices for weighted voting games (Theorem 4.2). In the literature, several techniques for computing Banzhaf power index of weighted voting games have been proposed. The main three methods are: generating functions [37], binary decision diagrams [38], and dynamic programming [33]. In this paper, we have invoked dynamic programming since it has the lowest computational complexity among the others. T. Uno proposed a slight improvement of the original dynamic programming approach, and showed that computing Banzhaf indices of all players can be done in $O(n \times q)$ instead of $O(n^2 \times q)$, where q denotes the quota and n is the number of players [33]. In SCG-Pruning algorithm, computing Banzhaf indices of $G_1 = (\Omega, [\mathbf{w}, q_1])$ and $G_2 = (\Omega, [\mathbf{w}, q_2 + 1])$ can be executed in parallel; hence, the calculation of the classifiers' ranks requires $O(n \times q_2)$ time complexity.

5. Experiments

5.1. Experimental setup

5.1.1. Datasets

To demonstrate the validity and the effectiveness of the proposed methodology, we carried out extensive experiments on 58 datasets selected from the UCI Machine Learning Repository [39]. Some datasets contain missing values due to several factors such as: inaccurate measurements, defective equipment, and human errors. An overview of the datasets properties is shown in Table 2.

We resampled each dataset following Dietterich's 5×2 cross validation (cv). More specifically, we first split (with stratification) the set of samples into two equal-sized folds *train* and *test*. We trained the ensemble members and estimated their weights using *train*; the other fold was dedicated to evaluate the generalization performance of each pruning technique. Then, we reversed the roles of *train* and *test* to obtain another estimate of the generalization accuracy. Repeating these steps five times, we finally obtained 10 trained ensembles and accuracy estimates of each pruning technique. It is noteworthy that we reported only the mean of these 10 accuracy measurements.

Table 2. Properties of the datasets used in the experiments.

Datasets	Abbreviations	Samples	Features	Missing values	Classes
Anneal	*Anneal*	898	38	Yes	6
Audiology	*Audiology*	226	69	Yes	24
Australian credit approval	*Australian*	690	14	No	2
Balance	*Balance*	526	4	No	3
Balloons adult+stretch	*Balloons1*	20	4	No	3
Balloons adult-stretch	*Balloons2*	20	4	No	3
Balloons small-yellow	*Balloons3*	20	4	No	3
Balloons small-yellow+adult-stretch	*Balloons4*	16	4	No	3
Breast cancer wisconsin	*BCW*	699	9	Yes	3
Breast cancer	*BC*	286	9	Yes	2
Car evaluation	*Car*	1728	6	No	4
Chess King-Rook vs King-Pawn	*Chess*	3196	36	No	2
Congressional voting records	*CVR*	435	16	Yes	2
Credit approval	*Credit*	690	15	Yes	2
Cylinder bands	*Cylinder*	540	39	Yes	2
Dermatology	*Dermatology*	366	34	Yes	6
Ecoli	*Ecoli*	336	8	No	8
Glass identification	*Glass*	214	10	No	6
Hayes-Roth	*Hayes-Roth*	160	5	No	4
Hepatitis	*Hepatitis*	155	19	Yes	2
Ionosphere	*Ionosphere*	351	34	No	2
Iris	*Iris*	150	4	No	3
Labor	*Labor*	57	16	Yes	2
Lenses	*Lenses*	24	4	No	3
Letter recognition	*Letter*	20000	16	No	26
Low resolution spectrometer	*LRS*	531	102	No	48
Lymphography	*Lymph*	148	18	No	4
Monks1	*Monks1*	556	6	No	2
Monks2	*Monks2*	601	6	No	2
Monks3	*Monks3*	554	6	No	2
Multi-feature fourier	*MFF*	2000	76	No	10
Multi-feature karhunen-love	*MFKL*	2000	64	No	10
Multi-feature profile correlations	*MFPC*	2000	216	No	10
Multi-feature zernike	*MFZ*	2000	47	No	10
Mushroom	*Mushroom*	8124	22	Yes	2
Musk1	*Musk1*	476	166	No	2
Musk2	*Musk2*	6598	166	No	2
Nursery	*Nursery*	12960	8	No	5
Optical recognition of handwritten digits	*Optical*	5620	64	No	10
Page blocks	*Page blocks*	5473	10	No	5
Pen-based recognition of handwritten digits	*Pen*	10992	16	No	10
Pima indians diabetes	*Pima*	768	8	No	2
Post-operative patient	*POP*	90	8	Yes	3
Soybean large	*Soybean L*	683	35	Yes	19
Soybean small	*Soybean S*	47	35	No	4
Spambase	*Spambase*	4601	57	No	2
SPECT heart	*SPECT*	267	22	No	2
SPECTF heart	*SPECTF*	267	44	No	2
Teaching assistant evaluation	*TAE*	151	5	No	3
Thyroid domain	*Thyroid D*	7200	21	No	3
Thyroid gland	*Thyroid G*	215	5	No	3
Tic-Tac-Toe endgame	*Tic-Tac-Toe*	958	9	No	2
Waveform (version 1)	*Waveform*	5000	21	No	3
Wine	*Wine*	178	13	No	3
Wisconsin diagnostic breast cancer	*WDBC*	569	30	No	2
Wisconsin prognostic breast cancer	*WPBC*	198	32	Yes	2
Yeast	*Yeast*	1484	8	No	10
Zoo	*Zoo*	101	16	No	7

5.1.2. Base classifiers

In order to generate the initial ensemble, we used 20 classifiers chosen from WEKA 3.6 [40], PRTOOLS 5.0.2 [41], and LIBSVM 3.18 [42]. A summary of these learning algorithms and their settings is given in Table 3. We set the rest of the parameters to their default values. It is worth noting that some classifiers do not support missing values. To overcome this problem, we replaced every missing entry with the mean and the mode for numeric and nominal features, respectively.

Table 3. List of classifiers used in the experiments.

No.	Algorithm	Platform	Description
1	J48	WEKA	C4.5 decision tree with the confidence factor set to 0.25. 2/3 of the training data were used for growing the tree, and 1/3 for pruning it.
2	SimpleCart	WEKA	Decision tree learner using CART's minimal cost complexity pruning.
3	Logistic	WEKA	Multinomial logistic regression.
4-6	IBk	WEKA	K-nearest neighbors classifier using linear search with the Euclidean distance, and 3 values for $k = 1, 3, 5$.
7	OneR	WEKA	1R rule-based learning algorithm.
8	NaïveBayes	WEKA	Standard probabilistic naïve Bayes classifier using supervised discretization.
9	Multilayer Perceptron	WEKA	Multilayer perceptron classifier using backpropagation algorithm run for 500 epochs with $(f + 1 + k)/2$ layers, where, f designates the number of features and k is the number of classes of a dataset. The learning rate was set to 0.3, and the momentum coefficient to 0.2.
10-11	Decision Table	WEKA	Simple decision table majority classifier using (10) BestFirst and (11) Genetic search methods with accuracy as the evaluation measure.
12	JRip	WEKA	RIPPER (Repeated Incremental Pruning to Produce Error Reduction) algorithm for rule induction. 2/3 of the training data were used for growing rules, and 1/3 for pruning them.
13	PART	WEKA	PART decision list built using J48 with the confidence factor set to 0.25. 2/3 of the training data were used for growing rules, and 1/3 for pruning them.
14	Fisherc	PRTOOLS	Fisher's least square linear classifier.
15	Ldc	PRTOOLS	Linear Bayes normal classifier. No regularization was performed.
16	Qdc	PRTOOLS	Quadratic Bayes normal classifier. No regularization was performed.
17	Parzendc	PRTOOLS	Parzen density based classifier. The smoothing parameters were estimated from the training data for each class.
18-20	SVM	LIBSVM	Support vector machines using (18) a radial (Gaussian) kernel with $\gamma = 1/f$ where f is the number of features; (19) a polynomial kernel of degree 3; and (20) a linear kernel. The cost parameter C was set to 1.0.

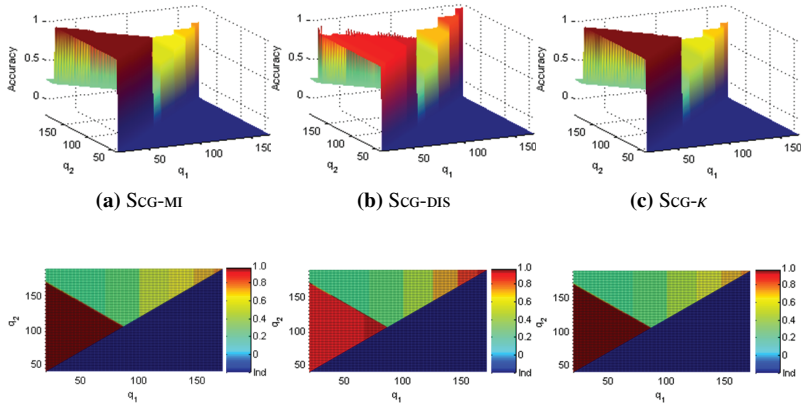
5.1.3. SCG-Pruning configurations

As stated in the previous section, the weights assigned to the ensemble members are computed based on a pairwise diversity measure. In our experiments, we used the three metrics given by equations 2, 4, and 5: disagreement measure (SCG-DIS), Cohen's kappa (SCG-κ), and mutual information (SCG-MI). We invoked MITOOLBOX [43] in order

to compute the information theoretic concepts.

5.1.4. Influence of the thresholds q_1 and q_2

In order to understand how the thresholds q_1 and q_2 affect the performance of the proposed approach, we present a 3D plot which displays the relationship between these thresholds and the accuracy of the produced ensemble by each of the SCG-Pruning variants. Fig. 3 shows the 3D plots for the three SCG-Pruning variants on the "Audiology" dataset. Given a point (x, y, z), x and y coordinates correspond to the values of q_1 and q_2, respectively. The z-coordinate indicates the performance of SCG-Pruning on the training set.



(a) SCG-MI **(b)** SCG-DIS **(c)** SCG-κ

(d) SCG-MI **(e)** SCG-DIS **(f)** SCG-κ

Fig. 3. (a),(b),(c) The impact of (q_1, q_2) on the performance of SCG-MI, SCG-DIS, and SCG-κ, respectively, for the "Audiology" dataset. The x and y axis correspond to the values of q_1 and q_2, respectively. z-axis represents the performance of the pruned ensemble. The figures (d), (e), and (f) show 2D plots from the top view of (a),(b), and (c), respectively

Examining Fig. 3.d, we can identify four main regions: The lower right half of the plot "blue surface" represents the set of impossible configurations of SCG-Pruning game. In this case, the values of q_1 and q_2 violate our initial condition, which states that $q_2 - max_{h_i} w_i > q_1$, and therefore the game can't be defined. The points laying close to the right upper corner of the plot "yellow triangle" (large q_1 and q_2) correspond to the configurations where the pruned ensemble exhibits very large diversity. On the left upper region "green triangle", we observe a very low performance by the three SCG-

Pruning variants. A possible explanation of this behavior is that the proposed game is not well-defined and fails to deliver an appropriate ranking of the ensemble members. More specifically, let consider the two extreme values of the thresholds $q_1 = 20$ and $q_2 = 190$. In this case, the interval that defines if a coalition wins (width of permitted diversity) is extremely large, and hence almost any coalition wins. In addition, the number of negative swings for every player is 0 since no coalition has a weight that exceeds 190. Finally, the last region "red triangle" yields the best performance and corresponds to the set of preferable game settings. We refer to it as $\mathcal{R}$. Under these settings, the proposed approaches produce ensembles with moderate diversities.

Based on these observations, we set the values for these thresholds as follows. For small-sized ensembles, we picked the pair (q_1, q_2) from $\mathcal{R}$ that yields the best performance on the training set; whereas for larger ensembles, we selected their values randomly from the search region $\mathcal{R}$.

5.2. First set of experiments

In the first series of experiments, we considered the size of the pruned ensemble L as an input parameter provided by the user. In this case, the proposed pruning approach selects the top L classifiers based on their Banzhaf indices. We referred to this variant as SCG-Ranking. We compared the proposed variants with: Kappa pruning, greedy, and exhaustive search strategies. For the greedy search [6], we implemented two variants: Forward Selection (Fs) and Backward Elimination (Be). Forward selection starts with an empty set; then, it chooses from among the classifiers not yet selected the classifier which best improves a specific *evaluation criterion* until the pre-set size of the pruned ensemble is met. Conversely, in backward elimination, the pruned ensemble is initialized as the entire ensemble; next, the algorithm proceeds by iteratively eliminating a classifier based on an *evaluation criterion* until the desired number of classifiers is reached. Exhaustive search tests all possible subsets of size L classifiers (there are $\binom{20}{L}$ subsets), and select the ensemble with highest *pre-defined criterion*. Both exhaustive and greedy search approaches require defining a criterion indicative of the ensemble generalization performance. To this end, we implemented the two criteria proposed by Meynet et al. [36]: Mutual Information Diversity (Mid), and Information Theoretic

Score (Its). Table 4 gives a summary of the compared ensemble selection techniques. Note that for all pruning techniques, we set the size of the pruned ensemble to $L = 3$, 5, 7, and 9.

Table 4. Legend for Tables and Figures presented in the first set of experiments.

Pruning technique	Description
Scg-l-κ	SCG-Ranking with Cohen's kappa (equation 4) as the diversity measure.
Scg-l-dis	SCG-Ranking with disagreement measure (equation 2) as the diversity metric.
Scg-l-mi	SCG-Ranking with mutual information (equation 5) as the diversity measure.
Fs-mid	Forward selection using the Mid evaluation criterion.
Fs-its	Forward selection with Its as the search criterion.
Be-mid	Backward elimination that uses Mid evaluation criterion.
Be-its	Backward elimination with Its as the search criterion.
Kappa	Kappa pruning.
Exh-l-mid	Exhaustive search that considers only ensembles of L classifiers using the Mid criterion.
Exh-l-its	Exhaustive search that considers only ensembles of L classifiers using the Its criterion.

Following Demšar's recommendations [44], we carried out a Friedman test to compare these 10 ensemble pruning techniques. This test is useful for comparing several algorithms over multiple datasets. Under the null hypothesis, we assumed that all techniques perform similarly. The mean ranks computed for Friedman tests are given in Table 5. The four Friedman tests reject the null hypothesis that all pruning schemes are equivalent and confirm the existence of at least one pair of techniques with significant differences. A summary of these tests' statistics is given in Table 6.

Table 5. Mean ranks of the 10 compared pruning techniques.

	Scg-l-κ	Scg-l-dis	Scg-l-mi	Fs-mid	Fs-its	Be-mid	Be-its	Kappa	Exh-l-mid	Exh-l-its
$L = 3$	**2.50**	2.66	2.92	7.67	5.99	7.94	7.20	5.90	7.40	4.82
$L = 5$	2.78	3.11	**2.51**	7.34	6.47	7.77	7.14	5.83	7.12	4.94
$L = 7$	2.97	3.44	**2.45**	7.00	6.51	7.44	7.08	6.97	6.89	4.26
$L = 9$	3.33	3.28	**2.32**	7.29	6.47	7.59	7.04	6.67	7.03	3.97

Table 6. Summary of the Friedman tests' statistics.

	L=3	**L=5**	**L=7**	**L=9**
F_F	58.26	46.99	42.15	45.66
α	1×10^{-16}	1×10^{-16}	1×10^{-16}	1×10^{-16}
Degrees of freedom (df)	9 ; 513	9 ; 513	9 ; 513	9 ; 513
F	11.62	11.62	11.62	11.62

Then, we proceeded with a post hoc Nemenyi test at a 5% significance level with the critical value $q_{0.05} = 3.16$ and the critical difference $CD = 1.78$. This test aims at identifying pairs of algorithms that are significantly different. We got the results shown by Figs. 4-7. On the horizontal axis, we represent the average rank of every pruning method, and link using thick lines the group of techniques with no significant differences. On the top left, we show the critical difference used in the test. Figs. 4-7 show that the proposed methodology performs significantly better than the other alternatives. More specifically, we can identify two families of pruning techniques. The first family is mainly composed of the proposed variants. The results indicate that SCG-L-MI performance is in the lead, but the experimental data does not provide any evidence regarding the significance differences among all SCG-Ranking configurations. In addition, as the size of the pruned ensemble increases ($L = 7, 9$), we observe an improvement in the performance of EXH-L-ITS (lower ranks). A possible explanation of this behavior might be related to the criterion ITS. For larger ensembles ($L > 5$), this criterion finds an appropriate subset of classifiers that balances accuracy and diversity, but fails to provide a reliable evaluation for small-sized ensembles. The second family i.e. diversity-based approaches, that is, pruning techniques which construct ensembles made of the most diverse classifiers, exhibit the worst performance. This latter observation confirms our initial claim which states that maximizing diversity deteriorates the generalization performance of the ensemble.

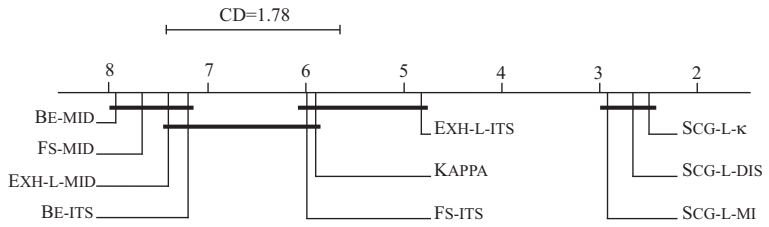


Fig. 4. Pairwise comparisons among all techniques for $L = 3$ using Nemenyi test. The numbers plotted on the horizontal axis correspond to the average ranks given in Table 5. The thick lines connect techniques that are not significantly different, and CD stands for the critical difference.

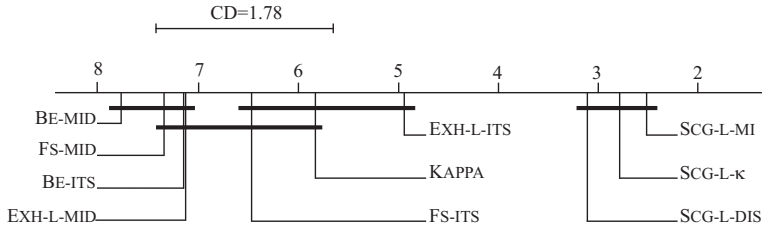


Fig. 5. Pairwise comparisons among all techniques for $L = 5$ using Nemenyi test. The numbers plotted on the horizontal axis correspond to the average ranks given in Table 5. The thick lines connect techniques that are not significantly different, and CD stands for the critical difference.

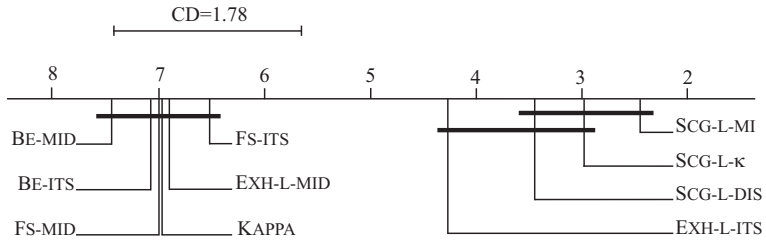


Fig. 6. Pairwise comparisons among all techniques for $L = 7$ using Nemenyi test. The numbers plotted on the horizontal axis correspond to the average ranks given in Table 5. The thick lines connect techniques that are not significantly different, and CD stands for the critical difference.

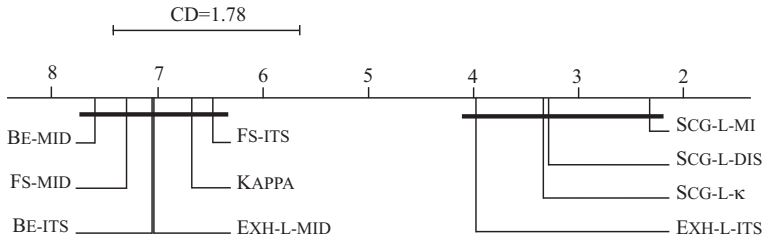
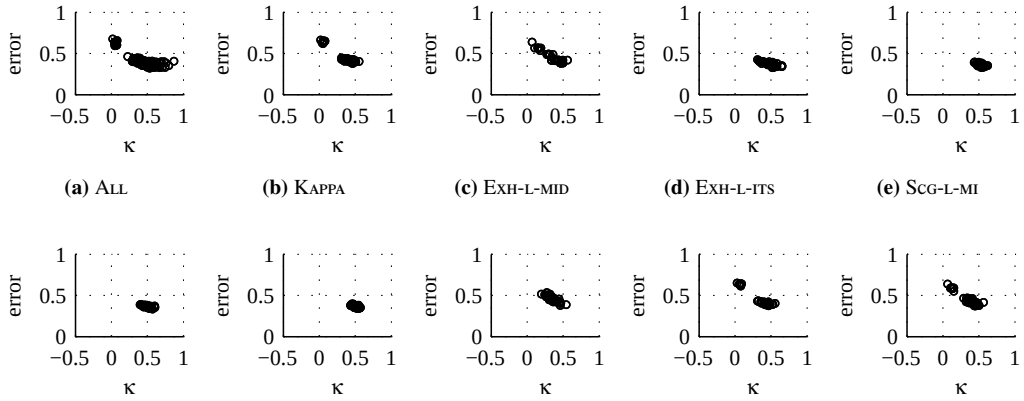


Fig. 7. Pairwise comparisons among all techniques for $L = 9$ using Nemenyi test. The numbers plotted on the horizontal axis correspond to the average ranks given in Table 5. The thick lines connect techniques that are not significantly different, and CD stands for the critical difference.

5.2.1. Kappa error diagrams

This section presents kappa error diagrams in order to gain some insight into the diversity/accuracy tradeoff. These diagrams depict an ensemble of classifiers as a scatterplot. Every pair of classifiers is represented as a point on the plot, where the x-coordinate corresponds to the value of Cohen's kappa κ between the pair, and the y-coordinate is the averaged individual error rate of the two classifiers. Following Garcìa-Pedrajas et al. [11], we estimated the error rate of every classifier on the test set. The aim of this experiment is to investigate whether the proposed idea, that is, constructing an ensemble with moderate diversity is responsible for the excellent results reported by the previous statistical tests. Figs. 8-9 show kappa error diagrams for several pruning techniques with $L = 9$ on two datasets: "Glass identification" and "Lymphography". Note that we also reported kappa error diagrams for the entire ensemble, denoted ALL.

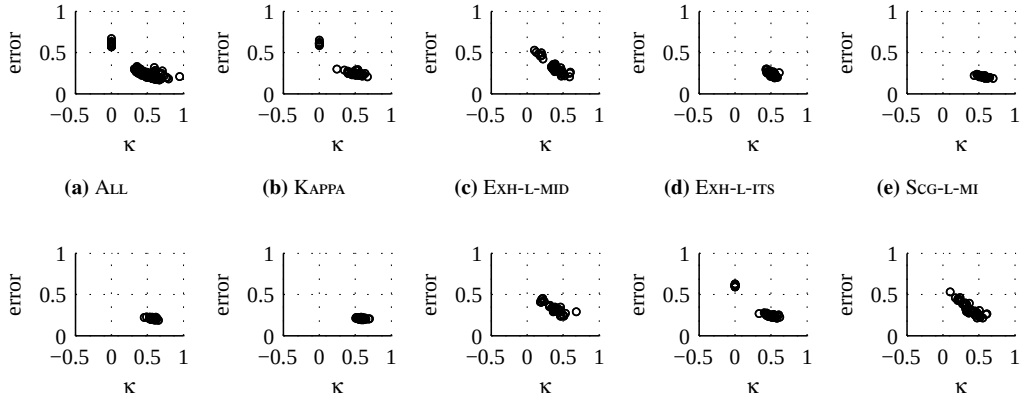


(a) ALL **(b)** KAPPA **(c)** EXH-L-MID **(d)** EXH-L-ITS **(e)** SCG-L-MI

(f) SCG-L-DIS **(g)** SCG-L-κ **(h)** BE-MID **(i)** FS-MID **(j)** BE-ITS

Fig. 8. Kappa error diagrams for the dataset "Glass identification".

The analysis of these diagrams is summarized as follows. First, the diagrams associated with the diversity-based pruning techniques are skewed to the left side of the plot, which indicates large diversity. This behavior is expected since these techniques construct ensembles that are made of the most diverse members. On the other hand, SCG-Ranking variants provide less diversity than the aforementioned approaches. Additionally, when compared to ALL, the proposed approach does not select strongly cor-



(a) ALL **(b)** KAPPA **(c)** EXH-L-MID **(d)** EXH-L-ITS **(e)** SCG-L-MI

(f) SCG-L-DIS **(g)** SCG-L-κ **(h)** BE-MID **(i)** FS-MID **(j)** BE-ITS

Fig. 9. Kappa error diagrams for the dataset "Lymphography".

related classifiers. This behavior is consistent with our initial idea, that is, the proposed methodology extracts sub-ensembles with moderate diversities.

5.2.2. Comparison of the proposed variants

In order to understand how the diversity measure affects the ranking process, we compared, in a pairwise manner, the similarity among the ensembles obtained by the three variants of the proposed methodology. Ulaş et al. [12] define the similarity between two ensembles S_1 and S_2 as:

$$Sim(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}. \tag{14}$$

The similarity varies between 0 and 1, where the value 1 indicates that the two ensembles are identical, and 0 means that they do not share any common members. Table 7 gives the averaged pairwise similarities among the ensembles obtained by the proposed approach variants for $L = 3$, 5, 7, and 9. The analysis of the results reported in Table 7 can be summarized by two important observations. First, the ensembles found by the proposed variants share, in average, at least half of their members. In addition, as the number of classifiers grows, all configurations tend to find very similar ensembles. We believe this behavior arises because the very first classifiers are indistinguishable,

and obtaining an identical ordering by all variants is uncommon. Second, the average similarity between SCG-L-DIS and SCG-L-κ is 0.78 ((0.67 + 0.76 + 0.82 + 0.85)/4), indicating that these two pruning techniques obtain very similar ensembles. This result is expected since both SCG-L-DIS and SCG-L-κ use statistical measures to estimate the diversity between two classifiers. Moreover, the similarity between SCG-L-MI and the statistical-based diversity variants (SCG-L-DIS and SCG-L-κ) is less than the one between SCG-L-DIS and SCG-L-κ, which justifies the different performances reported in the previous section.

Table 7. Averaged pairwise similarity measurements.

L=3	SCG-L-MI	SCG-L-DIS	SCG-L-κ
SCG-L-MI	1.00	0.45	0.56
SCG-L-DIS	0.45	1.00	0.67
SCG-L-κ	0.56	0.67	1.00

L=5	SCG-L-MI	SCG-L-DIS	SCG-L-κ
SCG-L-MI	1.00	0.59	0.69
SCG-L-DIS	0.59	1.00	0.76
SCG-L-κ	0.69	0.76	1.00

L=7	SCG-L-MI	SCG-L-DIS	SCG-L-κ
SCG-L-MI	1.00	0.69	0.75
SCG-L-DIS	0.69	1.00	0.82
SCG-L-κ	0.75	0.82	1.00

L=9	SCG-L-MI	SCG-L-DIS	SCG-L-κ
SCG-L-MI	1.00	0.73	0.78
SCG-L-DIS	0.73	1.00	0.85
SCG-L-κ	0.78	0.85	1.00

5.3. Second set of experiments

In the second experiment, the size of the pruned ensemble is no longer specified. The proposed approach selects the minimal winning coalition composed only of the best classifiers based on their Banzhaf indices. We compared the three variants of SCG-Pruning: SCG-κ, SCG-DIS, and SCG-MI with the following techniques:

EXH searches the space of all possible subsets ($2^{20} - 2$). Then, it chooses the ensemble that maximizes an evaluation criterion. For this search strategy, we implemented the following criteria: Mutual Information Diversity (EXH-MID), and Information Theoretic Score (EXH-ITS) [36].

ALL combines the predictions of all available classifiers without selection using majority vote.

GASEN**.** We evolved a population made of 20 individuals over 100 generations. The mutation and the crossover probabilities were set to 0.05 and 0.6, respectively.

Table 8 shows the results of the second experiment.

Table 8. Summary of mean accuracy results of the second experiment.

Datasets	SCG-κ	SCG-DIS	SCG-MI	GASEN	EXH-MID	EXH-ITS	ALL
Anneal	97.57	96.93	**98.49**	96.24	82.87	76.08	95.23
Audiology	80.09	78.23	**81.15**	77.43	76.46	63.01	76.19
Australian	84.43	84.00	84.81	77.91	76.72	77.01	**85.91**
Balance	90.05	88.06	**93.09**	90.18	67.14	71.61	89.50
Balloons1	**95.00**	**95.00**	**95.00**	92.00	81.00	71.00	93.00
Balloons2	**94.00**	92.00	92.00	87.00	82.00	91.00	86.00
Balloons3	**92.00**	91.00	**92.00**	88.00	75.00	88.00	80.00
Balloons4	71.25	**75.00**	72.50	68.75	62.50	60.50	68.75
BCW	96.02	96.14	96.57	95.88	91.10	95.39	**96.80**
BC	74.27	72.03	**74.97**	71.75	70.00	69.44	74.20
Car	93.65	93.23	**94.72**	87.18	88.30	70.95	92.67
Chess	99.13	99.10	**99.19**	94.14	68.06	68.06	98.03
CVR	93.84	93.84	**95.95**	94.07	91.44	92.92	95.59
Credit	83.74	85.39	85.59	77.97	73.28	72.52	**86.12**
Cylinder	76.52	76.52	76.52	75.78	70.26	66.70	**77.41**
Dermatology	97.49	97.49	**97.54**	97.43	61.80	68.63	97.38
Ecoli	**86.37**	84.94	85.83	83.15	66.73	74.11	**86.37**
Glass	70.56	68.97	**71.31**	70.75	60.70	62.10	69.07
Hayes-Roth	79.38	79.00	**82.50**	78.38	60.13	62.25	74.88
Hepatitis	**82.57**	80.63	82.18	80.65	80.25	80.90	82.32
Ionosphere	90.31	90.49	90.77	88.72	85.24	84.16	**91.85**
Iris	**95.07**	**95.07**	**95.07**	94.40	92.53	93.60	94.53
Labor	89.88	89.88	**90.23**	90.20	77.09	67.62	89.83
Lenses	76.67	77.50	77.50	74.17	**83.33**	65.00	76.67
Letter	95.93	96.05	**95.96**	94.76	65.36	68.33	95.33
LRS	55.48	52.96	**57.82**	50.55	49.96	47.04	53.45
Lymph	85.27	84.19	**86.08**	83.92	76.22	78.51	84.32
Monks1	99.46	99.46	**99.57**	95.68	90.72	90.40	95.14
Monks2	84.70	87.85	86.83	**88.62**	64.30	65.72	67.02
Monks3	97.15	97.15	**98.81**	95.78	86.79	80.90	97.18
MFF	82.20	**82.53**	82.05	80.67	60.89	78.89	82.27
MFKL	97.37	**97.43**	97.37	95.41	61.74	97.38	97.19
MFPC	**97.76**	97.74	97.72	97.07	66.56	88.83	97.65
MFZ	82.72	82.86	**82.84**	69.46	61.32	79.88	82.66
Mushroom	**100.0**	**100.0**	**100.0**	**100.0**	95.07	**100.0**	**100.0**
Musk1	**88.11**	**88.11**	**88.11**	84.50	78.15	77.73	**88.11**
Musk2	**98.54**	**98.54**	**98.54**	96.91	79.72	84.64	97.05
Nursery	98.35	98.29	**98.69**	89.69	70.97	70.97	97.22
Optical	98.67	98.67	**98.69**	98.21	69.39	98.64	98.57
Page blocks	**97.17**	97.00	97.15	95.97	92.98	92.70	96.84
Pen	99.32	99.29	**99.33**	98.46	64.43	66.72	99.10
Pima	73.46	72.58	74.45	71.35	70.29	68.96	**76.69**
POP	70.67	69.11	68.67	**71.11**	68.00	70.22	67.33
Soybean L	92.33	92.24	**92.47**	92.09	69.59	82.94	92.23
Soybean S	**100.0**	**100.0**	**100.0**	98.71	82.63	98.32	**100.0**
Spambase	**94.55**	94.51	94.54	91.46	80.38	79.64	94.39
SPECT	81.87	80.60	81.20	79.84	78.87	79.40	**82.39**
SPECTF	76.70	76.40	76.70	76.55	74.68	**78.80**	78.65
TAE	49.82	48.09	49.41	**50.21**	46.61	46.81	47.30
Thyroid D	**99.51**	99.18	99.48	93.41	99.57	92.58	96.74
Thyroid G	95.44	95.35	**95.72**	94.79	89.02	**95.72**	94.89
Tic-Tac-Toe	97.10	97.04	**97.33**	85.82	80.75	68.31	88.35
Waveform	85.80	**85.85**	84.24	80.15	62.24	61.30	**85.85**
Wine	**98.65**	98.54	98.54	95.96	77.08	98.20	98.43
WDBC	**96.45**	96.42	96.41	91.28	90.83	94.52	96.38
WPBC	77.88	78.59	**78.79**	76.87	75.15	76.26	78.69
Yeast	58.01	56.13	58.50	54.47	54.03	50.96	**60.05**
Zoo	94.67	**95.06**	**95.06**	94.67	92.90	92.11	**95.06**

We made pairwise comparisons between the performance of the entire ensemble "ALL" with each of the above presented ensemble pruning techniques using the Wilcoxon signed-ranks and the sign tests. Due to its robustness, we considered Wilcoxon test as the main comparison statistic. A summary of the Wilcoxon signed-ranks and the sign tests' statistics is shown in Table 9. The first row specifies the number of

win/tie/loss of the technique in the column over the technique in the row. The second and the third rows show the p-values for the sign and the Wilcoxon tests, respectively.

Table 9. Summary of Wilcoxon signed-ranks and sign tests' statistics.

		Scg-κ	Scg-dis	Scg-mi	Gasen	Exh-mid	Exh-its
All	$W/T/L$	**38**/5/15	**34**/5/19	**41**/4/13	13/2/**43**	4/0/**54**	7/1/**50**
	pv_s	2.23×10^{-3} +	4.79×10^{-2} *	3.07×10^{-4} +	1.00×10^{-4} +	3.17×10^{-12} +	2.40×10^{-9} +
	pv_w	2.47×10^{-3} +	9.45×10^{-2}	2.05×10^{-4} +	1.79×10^{-4} +	2.34×10^{-10} +	2.62×10^{-9} +

Differences at 5% significance level are marked with $*$, and at 1% with +.

The results shown in Tables 8 and 9 indicate that the proposed methodology performs better than the other alternatives in most cases. Most importantly, Scg-mi and Scg-κ significantly improve the performance of the initial ensemble with $p - value \leq 2.47 \times 10^{-3}$. Moreover, according to the sign test, the performance of Scg-dis is significantly better than All. However, Wilcoxon test fails to detect this difference. On the other hand, both tests indicate that the rest of the pruning techniques are significantly worse than All. Note that this experiment performs only pairwise comparisons to test whether each pruning technique improves the initial ensemble. In addition, it does not provide any evidence regarding the differences that might exist among the selection approaches. To this end, we carried on with a Friedman test to statistically compare the six pruning techniques. The averaged ranks assigned to these approaches are given in Table 10.

Friedman test rejects the null hypothesis which states that these methods are equivalent with $F_F = 109.70 > F(5, 285) = 19.47$ for $\alpha = 1.0 \times 10^{-16}$ (F_F is distributed according to the F distribution with $6 - 1 = 5$ and $(6 - 1) \times (58 - 1) = 285$ degrees of freedom). Then, in order to identify pairs of pruning techniques with significant performance differences, we followed up this finding with a post hoc Nemenyi test at a 5% significance level with the critical value $q_{0.05} = 2.85$ and the critical difference $CD = 0.99$.

Table 10. Averaged ranks of the 6 compared pruning techniques.

Scg-κ	Scg-dis	Scg-mi	Gasen	Exh-mid	Exh-its
2.24	2.67	1.76	3.86	5.43	5.03

The pairwise comparisons given by Nemenyi test (Fig. 10) reveal the existence of three groups of techniques: SCG-Pruning, GASEN, and EXH variants from the best-performing pruning approach to the worst one. As shown by the first experiment, no significance difference can be observed among the proposed variants. In particular, SCG-MI shows better performance than the other alternatives. We also reported an important drop in the performance of EXH-ITS in contrast to the first experiment. In addition to the observations discussed earlier, we believe this drop occurs because EXH-ITS fails to find the right number of classifiers to include in the final ensemble.

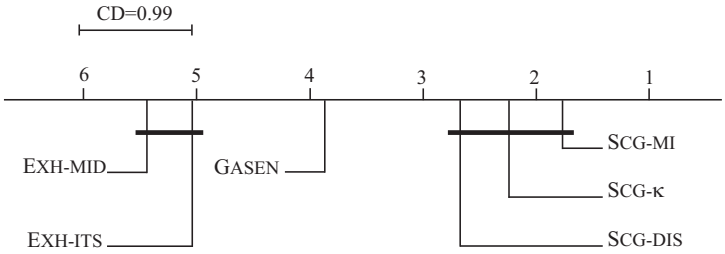


Fig. 10. Pairwise comparisons among the 6 pruning techniques using Nemenyi test. The numbers plotted on the horizontal axis correspond to the average ranks given in Table 10. The thick lines connect techniques that are not significantly different, and CD stands for the critical difference.

5.4. Third set of experiments: influence of the ensemble size

In this experiment, we investigate the influence of the initial ensemble size on the performance of the proposed approach [1]. To this end, we trained an ensemble made of 100 Decision Stump trees using BAGGING. For both learning models, we imported the implementation provided by WEKA, and set all their parameters to the default values. We compared SCG-MI and SCG-κ with Reduce Error (RE) [14], Complementarity Measure (CC) [20], Margin Distance Minimization (MDSQ) [20] with a moving reference point p set to $2\sqrt{2 \times i}/n$ at the i^{th} iteration, Orientation Ordering (OO) [45], Boosting-Based (BB) [19], Genetic algorithm (GASEN), and Kappa pruning (KAPPA). For genetic algorithm, we used the following configurations: crossover probability=0.6, mutation rate=0.05, number of generations=100, and population size=100. It is noteworthy that

[1] We would like to thank the anonymous reviewer for suggesting us to carry out this experiment.

the pruning approaches RE, CC, MDSQ, OO, BB, and KAPPA require setting the size of the pruned ensemble L. In order to make a fair comparison, we set L to the same size found by SCG-MI. Table 11 gives the results of this experiment. The last row specifies the mean rank of each method over all datasets.

Table 11. Summary of mean accuracy results of the third experiment.

Datasets	SCG-κ	SCG-MI	GASEN	MDSQ	RE	OO	KAPPA	CC	BB	BAGGING
Anneal	**83.54**	**83.54**	82.78	82.78	82.78	79.11	78.33	82.34	78.35	82.78
Audiology	**47.17**	47.08	46.46	46.46	46.46	46.46	46.46	46.46	46.46	46.46
Australian	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**
Balance	**80.16**	78.82	80.13	78.72	79.17	79.23	74.49	74.46	77.47	72.38
Balloons1	87.00	87.00	84.00	87.00	81.00	81.00	75.00	72.00	**94.00**	74.00
Balloons2	81.00	76.00	75.00	76.00	72.00	71.00	**82.00**	**82.00**	80.00	72.00
Balloons3	**75.00**	**75.00**	67.00	69.00	68.00	60.00	69.00	64.00	69.00	68.00
Balloons4	67.50	67.50	68.75	65.00	65.00	**70.00**	66.25	66.25	65.00	62.50
BCW	**95.57**	95.11	94.59	94.91	94.39	94.56	95.39	93.45	92.70	93.39
BC	73.71	73.92	72.73	**74.34**	73.71	73.92	70.49	71.61	72.59	71.89
Car	**70.02**	**70.02**	**70.02**	**70.02**	**70.02**	**70.02**	**70.02**	**70.02**	**70.02**	**70.02**
Chess	**66.05**	**66.05**	**66.05**	**66.05**	**66.05**	**66.05**	**66.05**	**66.05**	**66.05**	**66.05**
CVR	**95.63**	**95.63**	**95.63**	**95.63**	**95.63**	94.94	94.94	94.94	95.03	**95.63**
Credit	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**	**85.51**
Cylinder	70.04	69.04	70.52	69.44	70.33	68.19	67.11	64.52	69.04	**70.56**
Dermatology	**59.13**	56.01	53.11	51.69	53.06	50.08	52.08	50.11	50.11	51.37
Ecoli	**67.44**	**67.44**	64.64	64.64	64.64	64.70	63.81	64.58	64.58	64.64
Glass	53.83	**57.38**	52.52	55.05	56.54	51.04	50.16	50.64	50.55	51.25
Hayes-Roth	**60.75**	59.50	**60.75**	56.00	59.38	54.38	54.38	50.08	50.13	56.25
Hepatitis	81.80	81.80	81.03	**83.22**	81.67	82.83	79.48	79.75	80.50	81.03
Ionosphere	83.31	82.79	82.96	83.13	82.79	82.16	83.02	81.48	83.25	**83.37**
Iris	**95.33**	**95.33**	95.07	94.27	95.20	87.60	82.47	80.00	94.67	94.53
Labor	85.97	85.20	83.17	**88.40**	81.77	84.19	88.39	78.95	88.39	82.41
Lenses	**76.67**	70.00	75.83	72.50	**76.67**	71.67	64.17	61.67	67.50	64.17
Letter	70.78	71.29	68.03	68.97	69.91	67.98	67.63	67.08	67.58	**71.94**
LRS	**51.49**	50.06	49.72	49.68	50.10	47.38	48.97	49.72	49.72	49.68
Lymph	76.22	76.08	76.35	**77.30**	75.41	75.81	72.97	72.03	70.81	74.46
Monks1	**74.64**	**74.64**	**74.64**	**74.64**	**74.64**	**74.64**	**74.64**	**74.64**	**74.64**	**74.64**
Monks2	65.19	65.19	65.16	65.39	65.52	65.03	65.39	64.43	65.39	**65.72**
Monks3	78.81	78.81	78.81	78.81	78.81	77.65	77.83	78.48	78.81	**89.89**
MFF	**68.41**	67.70	65.90	61.63	68.26	62.12	63.67	60.68	60.53	62.64
MFKL	65.04	**65.09**	62.17	61.12	63.43	60.63	63.20	60.50	60.58	64.30
MFPC	74.99	73.29	72.04	67.70	**77.89**	65.84	60.77	61.77	62.85	77.88
MFZ	66.62	**67.26**	64.40	64.38	66.60	63.71	63.29	63.39	63.43	66.02
Mushroom	**88.68**	**88.68**	**88.68**	**88.68**	**88.68**	**88.68**	**88.68**	**88.68**	**88.68**	**88.68**
Musk1	**72.27**	71.72	72.18	71.26	72.18	70.76	69.79	70.55	71.89	71.47
Musk2	**84.59**	**84.59**	**84.59**	**84.59**	**84.59**	**84.59**	**84.59**	**84.59**	**84.59**	**84.59**
Nursery	**66.25**	**66.25**	**66.25**	**66.25**	**66.25**	66.08	66.08	66.08	**66.25**	**66.25**
Optical	**65.40**	64.35	63.49	62.96	63.38	62.67	62.62	61.79	61.79	64.12
Page blocks	93.17	**93.18**	93.13	93.13	93.13	93.06	93.06	93.13	93.06	93.13
Pen	**60.66**	60.56	60.59	60.51	60.63	60.05	60.01	60.46	60.49	60.57
Pima	**74.97**	74.66	74.77	74.61	74.58	73.85	71.85	71.59	72.76	74.11
POP	64.22	62.44	68.00	65.33	70.67	62.89	65.78	61.11	64.22	**70.89**
Soybean L	68.26	**68.49**	68.38	68.41	68.43	66.38	66.21	67.44	67.47	67.50
Soybean S	**97.83**	95.80	97.39	90.62	81.49	76.54	71.45	72.84	74.09	96.21
Spambase	**83.31**	83.15	81.73	81.26	81.53	81.04	79.97	79.06	79.95	79.07
SPECT	**79.40**	**79.40**	**79.40**	**79.40**	**79.40**	**79.40**	**79.40**	**79.40**	**79.40**	**79.40**
SPECTF	78.05	77.75	77.83	78.13	78.35	78.20	79.25	76.47	77.30	**79.40**
TAE	47.39	46.71	47.39	46.46	**49.91**	49.27	45.08	44.55	44.96	46.72
Thyroid D	**95.24**	**95.24**	**95.24**	**95.24**	**95.24**	**95.24**	**95.24**	**95.24**	**95.24**	**95.24**
Thyroid G	82.69	**82.78**	81.58	80.93	82.60	81.12	79.54	80.47	80.37	79.72
Tic-Tac-Toe	**70.02**	69.79	69.48	69.94	69.94	69.06	68.85	67.16	68.81	69.94
Waveform	60.90	60.18	60.22	60.28	59.93	60.21	60.08	57.47	58.11	**61.46**
Wine	92.70	92.02	91.35	92.13	90.79	91.46	83.71	80.85	**94.94**	89.44
WDBC	92.83	**92.94**	91.81	92.72	92.44	92.72	92.65	91.21	92.83	90.97
WPBC	72.32	74.24	74.44	73.84	75.56	72.73	76.06	70.71	73.54	**76.36**
Yeast	50.58	50.67	50.61	50.50	50.61	47.78	49.02	50.61	**50.70**	50.54
Zoo	**73.62**	64.37	61.95	62.55	60.58	59.20	65.90	59.40	56.07	61.57
Average ranks	**3.14**	**3.86**	**4.69**	**4.94**	**4.58**	**6.66**	**7.03**	**8.15**	**6.62**	**5.34**

First, we statistically compared the performances of these pruning schemes us-

ing the average ranks over 58 datasets. Friedman test rejects the null hypothesis that all methods have similar performances with $F_F = 20.77 > F(9, 513) = 11.62$ for $\alpha = 1 \times 10^{-16}$ (F_F is distributed according to the F distribution with $10 - 1 = 9$ and $(10 - 1) \times (58 - 1) = 513$ degrees of freedom). Since we are only interested in testing whether the pruning approaches significantly improve the initial ensemble "Bagging", we conducted a Bonferroni-Dunn test at a 10% significance level with the critical value $q_{0.10} = 2.54$ and the critical difference $CD = 1.43$. The results of this test are depicted by Fig. 11. On the horizontal axis, we represent the averaged rank of every pruning technique, and mark using a thick line an interval of $2 \times CD$ one on the right and the other to the left of Bagging's mean rank.

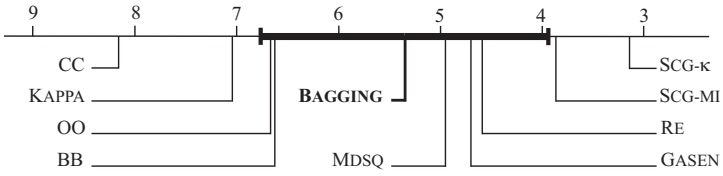


Fig. 11. Comparison of Bagging with 9 pruning techniques using Bonferroni-Dunn test. The numbers plotted on the horizontal axis correspond to the average ranks given in Table 11. All techniques with ranks outside the marked interval are significantly different than Bagging.

The analysis of Bonferroni-Dunn test (Fig. 11) reveals that the performances of Scg-κ and Scg-mi are in the lead followed by Re, Gasen, and Mdsq. Most importantly, we notice that both Scg-κ and Scg-mi fall outside the marked interval. Therefore, we can conclude that the proposed variants perform significantly better than Bagging, while the experimental data cannot detect any improvement of Bagging using Re, Gasen, BB, OO, or Mdsq.

Next, we compared in Table 12 the averaged running time (in seconds) required by every pruning technique over all datasets. Experimentation was conducted on a 3.6 Ghz Intel Core $i7 - 4790$ processor with 8 Gb of system memory.

Table 12. Average pruning times (in seconds) of several pruning approaches.

Scg-κ	Scg-mi	Gasen	Mdsq	Re	OO	Kappa	CC	BB	Fs-its	Fs-mid
0.320	0.401	36.86	0.015	0.793	0.003	0.174	0.032	0.016	5.770	3.075

Orientation ordering is the fastest technique followed by Mdsq, BB, and CC. Both

Scg-κ and Scg-mi converge to similar pruning times. The results also indicate that Gasen and greedy search approaches are slower than the other alternatives. The reported behavior is expected since search-based pruning methods generally tend to have high computational costs.

6. Conclusion and future work

This paper introduced a game theory-based methodology for ensemble pruning. We have developed a simple coalitional game for estimating the power of each member based on its contribution to the overall ensemble diversity. Additionally, we have provided a powerful criterion based on the notion of minimal winning coalition (made of the most powerful members) that allows pruning an ensemble of classifiers. Experimental results show that SCG-Pruning significantly improves the performance of the entire ensemble and outperforms some major state-of-the-art selection approaches. Furthermore, our approach provides a reliable ranking, and succeeds in finding the appropriate number of classifiers to include in the final ensemble. We have noticed that the thresholds q_1 and q_2 are of great importance for determining the right size of the pruned ensemble.

Our future work consists of evaluating SCG-Pruning with other methods for weighing the ensemble members, and for computing the pairwise diversity. Furthermore, we will investigate deeply the relationship between the thresholds (q_1, q_2) and the generalization performance of the pruned ensemble so that they can be set properly for real world applications.

References

[1] M. Han, B. Liu, Ensemble of extreme learning machine for remote sensing image classification, Neurocomputing 149 (2015) 65–70.

[2] A. Mashhoori, Block-wise two-directional 2DPCA with ensemble learning for face recognition, Neurocomputing 108 (2013) 111–117.

[3] B. Kavitha, S. Karthikeyan, P. S. Maybell, An ensemble design of intrusion detection system for handling uncertainty using Neutrosophic Logic Classifier, Knowledge-Based Systems 28 (2012) 88–96.

[4] L. Rokach, R. Romano, O. Maimon, Negation recognition in medical narrative reports, Information Retrieval 11 (6) (2008) 499–538.

[5] L. Rokach, Pattern classification using ensemble methods, 1st Edition, World Scientific Publishing Company, Singapore, 2010.

[6] Z.-H. Zhou, Ensemble methods: Foundations and algorithms, 1st Edition, Taylor & Francis, Boca Raton, FL, 2012.

[7] G. Martínez-Muñoz, D. Hernández-Lobato, A. Suárez, An Analysis of Ensemble Pruning Techniques Based on Ordered Aggregation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2) (2009) 245–259.

[8] S. Sun, An improved random subspace method and its application to EEG signal classification, in: Multiple Classifier Systems, 2007, pp. 103–112.

[9] S. González, F. Herrera, S. García, Monotonic random forest with an ensemble pruning mechanism based on the degree of monotonicity, New Generation Computing 33 (4) (2015) 367–388.

[10] S. Sun, Local within-class accuracies for weighting individual outputs in multiple classifier systems, Pattern Recognition Letters 31 (2) (2010) 119–124.

[11] N. García-Pedrajas, C. García-Osorio, C. Fyfe, Nonlinear boosting projections for ensemble construction, Journal of Machine Learning Research 8 (2007) 1–33.

[12] A. Ulaş, M. Semerci, O. T. Yıldız, E. Alpaydın, Incremental construction of classifier and discriminant ensembles, Information Sciences 179 (9) (2009) 1298–1318.

[13] Y. Bi, The impact of diversity on the accuracy of evidential classifier ensembles, International Journal of Approximate Reasoning 53 (4) (2012) 584–607.

[14] D. D. Margineantu, T. G. Dietterich, Pruning adaptive boosting, in: International Conference on Machine Learning, 1997, pp. 211–218.

[15] G. Tsoumakas, I. Partalas, I. Vlahavas, An ensemble pruning primer, in: Applications of Supervised and Unsupervised Ensemble Methods, 1st Edition, Springer, Berlin, Heidelberg, 2009, Ch. 1, pp. 1–13.

[16] Z. Lu, X. Wu, X. Zhu, J. Bongard, Ensemble pruning via individual contribution ordering, in: International Conference on Knowledge Discovery and Data Mining, 2010, pp. 871–880.

[17] H. Ykhlef, D. Bouchaffra, Induced subgraph game for ensemble selection, in: IEEE International Conference on Tools with Artificial Intelligence, 2015, pp. 636–643.

[18] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, Ordering-based pruning for improving the performance of ensembles of classifiers in the framework of imbalanced datasets, Information Sciences 354 (2016) 178–196.

[19] G. Martínez-Muñoz, A. Suárez, Using boosting to prune bagging ensembles, Pattern Recognition Letters 28 (1) (2007) 156–165.

[20] G. Martínez-Muñoz, A. Suárez, Aggregation ordering in bagging, in: International Conference on Artificial Intelligence and Applications, 2004, pp. 258–263.

[21] Z.-H. Zhou, J.-X. Wu, Y. Jiang, S.-F. Chen, Genetic algorithm based selective neural network ensemble, in: International Joint Conference on Artificial Intelligence, 2001, pp. 797–802.

[22] Y. Zhang, S. Burer, W. N. Street, Ensemble pruning via semi-definite programming, Journal of Machine Learning Research 7 (2006) 1315–1338.

[23] L. Rokach, Collective-agreement-based pruning of ensembles, Computational Statistics and Data Analysis 53 (4) (2009) 1015–1026.

[24] A. Lazarevic, Z. Obradovic, Effective pruning of neural network classifier ensembles, in: International Joint Conference on Neural Networks, 2001, pp. 796–801.

[25] G. Giacinto, F. Roli, G. Fumera, Design of effective multiple classifier systems by clustering of classifiers, in: International Conference on Pattern Recognition, 2000, pp. 160–163.

[26] B. Bakke, T. Heskes, Clustering ensembles of neural network models, Neural Networks 16 (2) (2003) 261–269.

[27] I. Partalas, G. Tsoumakas, I. Vlahavas, Pruning an ensemble of classifiers via reinforcement learning, Neurocomputing 72 (7-9) (2008) 1900–1909.

[28] G. Brown, An information theoretic perspective on multiple classifier systems, in: Multiple Classifier Systems, 2009, pp. 344–353.

[29] M. J. Osborne, A. Rubinstein, A Course in Game Theory, MIT Press, Cambridge, 1994.

[30] G. Chalkiadakis, E. Elkind, M. Wooldridge, Computational aspects of cooperative game theory, Morgan & Claypool Publishers, California, 2011.

[31] J. F. Banzhaf, Weighted voting doesn't work: A mathematical analysis, Rutgers Law Review 19 (2) (1965) 317–343.

[32] Z.-H. Zhou, N. Li, Multi-information ensemble diversity, in: Multiple Classifier Systems, 2010, pp. 134–144.

[33] T. Uno, Efficient computation of power indices for weighted majority games, Tech. rep., National Institute of Informatics, Tokyo (2003).

[34] W. H. Riker, The theory of political coalitions, Midwest Journal of Political Science 7 (3) (1962) 295–297.

[35] A. M. Colman, Game theory and its applications in the social and biological sciences, Butterworth-Heinemann, Oxford, 1992.

[36] J. Meynet, J.-P. Thiran, Information theoretic combination of pattern classifiers, Pattern Recognition 43 (10) (2010) 3412–3421.

[37] E. Algaba, J. Bilbao, J. F. Garca, J. López, Computing power indices in weighted multiple majority games, Mathematical Social Sciences 46 (1) (2003) 63–80.

[38] S. Bolus, Power indices of simple games and vector-weighted majority games by means of binary decision diagrams, European Journal of Operational Research 210 (2) (2011) 258–272.

[39] K. Bache, M. Lichman, UCI Machine Learning Repository (2015).
URL `http://archive.ics.uci.edu/ml`

[40] I. H. Witten, E. Frank, Data mining: Practical machine learning tools and techniques, 3rd Edition, Morgan Kaufmann Publishers, California, 2011.

[41] R. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, D. Tax, S. Verzakov, PRTools 4.1: A matlab toolbox for pattern recognition, Tech. rep., Delft University of Technology, Delft (2007).

[42] C.-C. Chang, C.-J. Lin, LIBSVM : a library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (3) (2011) 27.

[43] G. Brown, A. Pocock, M.-J. Zhao, M. Luján, Conditional likelihood maximisation: A unifying framework for information theoretic feature selection, Journal of Machine Learning Research 13 (2012) 27–66.

[44] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.

[45] G. Martínez-Muñoz, A. Suárez, Pruning in ordered bagging ensembles, in: International Conference in Machine Learning, 2006, pp. 609–616.