Ensemble selection based on classifier prediction confidence.

NGUYEN, T.T., LUONG, A.V., DANG, M.T., LIEW, A.W.-C., MCCALL, J.

2020

Ensemble Selection based on Classifier's Confidence in Prediction

Tien Thanh Nguyen , Anh Vu Luong , Manh Truong Dang ,
Alan Wee-Chung Liew , John McCall

PII: S0031-3203(19)30405-4
DOI: https://doi.org/10.1016/j.patcog.2019.107104
Reference: PR 107104

To appear in: *Pattern Recognition*

Received date: 2 May 2019
Revised date: 27 September 2019
Accepted date: 3 November 2019

Please cite this article as: Tien Thanh Nguyen , Anh Vu Luong , Manh Truong Dang , Alan Wee-Chung Liew , John McCall , Ensemble Selection based on Classifier's Confidence in Prediction, *Pattern Recognition* (2019), doi: https://doi.org/10.1016/j.patcog.2019.107104

Highlights

1. An ensemble selection method that takes into account each base classifier's confidence during classification and its overall credibility on the task is proposed.

2. The overall credibility of a base classifier is obtained by minimizing the empirical 0-1 loss on the entire training set.

3. The classifier's confidence in prediction for a test sample is measured by the entropy of its soft classification outputs for that sample.

4. Extensive comparative experiments with the state-of-the-art algorithms on ensemble selection validated the superior performance of our algorithm.

Ensemble Selection based on Classifier's Confidence in Prediction

Tien Thanh Nguyen[1], Anh Vu Luong[2], Manh Truong Dang[1], Alan Wee-Chung Liew[2], John McCall[1]

[1]School of Computing Science and Digital Media, Robert Gordon University, Aberdeen, Scotland, UK

[2]School of Information and Communication Technology, Griffith University, Australia

[*]Corresponding Author: Alan Wee-Chung Liew

Email: a.liew@griffith.edu.au

Abstract: Ensemble selection is one of the most studied topics in ensemble learning because a selected subset of base classifiers may perform better than the whole ensemble system. In recent years, a great many ensemble selection methods have been introduced. However, many of these lack flexibility: either a fixed subset of classifiers is pre-selected for all test samples (static approach), or the selection of classifiers depends upon the performance of techniques that define the region of competence (dynamic approach). In this paper, we propose an ensemble selection method that takes into account each base classifier's confidence during classification and the overall credibility of the base classifier in the ensemble. In other words, a base classifier is selected to predict for a test sample if the confidence in its prediction is higher than its credibility threshold. The credibility thresholds of the base classifiers are found by minimizing the empirical 0-1 loss on the entire training observations. In this way, our approach integrates both the static and dynamic aspects of ensemble selection. Experiments on 62 datasets demonstrate that the proposed method achieves much better performance in comparison to some ensemble methods.

Keyword: ensemble method; multiple classifier system; ensemble selection; classifier selection; Artificial Bee Colony

1. Introduction

Ensemble learning has been studied extensively and is one of the most active research topics in the machine learning community. This kind of learning naturally emerges based on the fact that no learning algorithm can perform well on all datasets. In machine learning, each classifier uses its own approach to approximate the unknown relationship f between the feature vector and the class labels. As data collected from different sources can vary quite substantially, a learning algorithm may only provide good hypothesis on some datasets. By combining multiple classifiers in a single framework as in ensemble learning, we can diversify the learning and achieve better predictions than using a single classifier [1].

In ensemble methods, we aggregate the outputs of different classifiers to arrive at a collaborated decision. Classifiers can be generated in two different ways: training different algorithms on the same training set (heterogeneous ensemble method) or training a single algorithm on many different training sets (homogeneous ensemble method) [1, 2]. A combining algorithm is then used to combine the outputs of all classifiers to obtain the final decision.

Ensuring diversity in the outputs of the base classifiers is an important factor in a successful ensemble. Existing studies on diversity in an ensemble system mainly focus on its utilization to enhance the ensemble performance, for example in the combining algorithms [3, 4] and in the ensemble selection problem [5-7]. These methods, however, only capture the uncertainty generated by the agreements and disagreements between the different base classifiers. The exploitation of confidence in each base classifier's output to solve the ensemble selection (ES) problem, therefore, needs to be explored.

Our idea is based on the observation in real-life where a decision is sought from the committee of experts but different experts have different background and different level of expertise on a problem. When we know that an expert is very knowledgeable in a particular field, we will trust the recommendation of this expert even though he/she is not entirely confident about the current recommendation. On the other hand, if we know that an expert is less knowledgeable, we will only pay attention to his/her current recommendation if he/she is very sure of it. This idea is applied to select base classifiers for the final ensemble for a prediction. In this work, we encode the level of domain expertise of a base classifier by associating with each base classifier a threshold computed from the entire training set by minimizing the empirical 0-1 loss. Then, based on the soft classification output of a base classifier on a test sample, we quantify the confidence level of the current classification by computing the entropy of each base classifier's output. It is noted that high entropy in the prediction represents low confidence, therefore entropy can be used as a confidence measure. The entropy is then compared to the base classifier's threshold to determine whether the output

of the base classifier should be included in the aggregation. A base classifier's output appears in the final set for subsequent ensemble combination if its confidence level is higher than its threshold.

The contributions of this paper are:

(i) We propose an approach to select a base classifier in an ensemble system based on its overall domain expertise and the confidence value it has on its current prediction. This allows us to integrate both the static and dynamic approach of ensemble selection.

(ii) We search for the individual threshold of each base classifier by minimizing the $0-1$ empirical loss on the training set. The optimal solution is obtained by using the artificial bee colony optimization.

(iii) Experiments on a number of datasets demonstrated the advantage of the proposed method compared to several well-known benchmark algorithms.

We organize the paper as follows. In Section 2, we briefly discuss some related work on ensemble methods and ensemble selection. In Section 3, we describe our approach to measure and select the expert's answer based on its confidence in relation to its domain expertise. In Section 4, we present our experimental studies in which we compare the performance of the proposed method and the benchmark algorithms on some popular datasets. In Section 5, we draw some conclusions.

2. Related work

2.1. Ensemble methods

Research on ensemble methods focuses mainly either on the design of new ensemble systems, improving the ensemble performance, or the study of ensemble properties. There are two approaches to design a new ensemble system: training data generation and combining algorithm formulation. In [8], Younsi and Bagnall introduced two ensemble systems using random sphere cover classifiers (RSC). The first ensemble system is based on the resampling/reweighting mechanism in which the RSCs are generated sequentially and the current RSC focuses more on the hard samples, i.e. at the decision boundary, uncovered cases, or misclassified cases evaluated by the RSC in the previous step. In the second ensemble system, an ensemble of RSCs is constructed on the random subset of attributes obtained from the original attribute set. In [9], Zhang et al. proposed a new ensemble system by training k Nearest Neighbor (kNN) algorithm on the new training sets generated by both random subspace and bootstrap sampling technique. In [10], Pham et al. proposed an incremental ensemble system which is updated with the newly arrived data if the ground truth is available.

The system is constructed by learning the Hoeffding tree on the projected data obtained by using random projections. In [11], Santucci et al. performed sampling on the parameters of three learning algorithms, i.e. Nearest Mean Classifier (NMC), Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA), to construct an ensemble system. In [12], Yijing et al. proposed a new weighted combining rules in which the weight of each base classifier is computed based on its performance on the training data measured by Area under the ROC Curve (AUC). In [3], Kuncheva et al. averaged the meta-data of the training observations based on their class labels to generate the class representations called decision templates. The distance between the prediction on the test sample and the decision templates is used to select the most suitable class label. In [1, 4], Nguyen et al. captured the uncertainty in the outputs of different base classifiers using interval-valued class representations. The best class label for a test sample is then selected based on the transform function from interval to numerical value in [4] or the distance between the prediction on the test sample and the interval class representations [1]. In [2], Nguyen et al. applied the fuzzy IF-THEN rules to the meta-data to generate the fuzzy classification rules, thereby taking into account the uncertainty between the outputs of different base classifiers.

Meanwhile, based on the observation that the inclusion of some base classifiers may adversely affects the performance of the ensemble, several ensemble selection methods have been proposed to choose the optimal subset of base classifiers [6]. In [13], Yu et al. introduced a progressive subspace ensemble in which a selection process was developed to select and weigh the based classifiers generated by the Random Subspace method. Class label is then assigned to the test sample by weighted voting from the selected based classifiers. In [14], Garcia-Pedrajas et al. developed an ensemble focusing on difficult samples in the classification. However, instead of weighting and sampling the samples like in Boosting, the authors focused on the misclassified samples in the projected subspaces obtained by using the supervised linear/nonlinear projections on random subspaces. In [15], Nguyen et al. introduced a method to weigh the base classifiers in a random projection-based ensemble system in which random projections are applied to the original training data to obtain new training sets. In that study, the weighs of base classifiers are obtained by solving a linear regression problem between the predictions for training observations and the ground truth.

Finally, the properties of ensemble systems have been studied in order to boost the performance of ensemble systems. In [16], Tang et al. analyzed the relationship between six diversity measures and the concept of margin. It was concluded that maximizing the diversity among the base classifiers is equivalent to maximizing the minimum margin of the ensemble on the training samples under some specific conditions. In [17], Jackowski introduced two new diversity measures for data stream classification ensemble called pair and pool error trend diversity measure. These diversity measures were developed based on the direction and extent

of changes in classification error rate of the base classifiers when the stream changes. A new definition for ensemble margin from the prediction of base classifiers was introduced in [18]. This concept was then used to construct the classification loss function which is minimized to learn the weight of base classifiers. In [7], Guo et al. proposed an ensemble pruning method based on the margin and diversity of ensemble systems.

2.2. Ensemble selection methods

The purpose of **Ensemble Selection (ES)** (also known as selective ensemble or ensemble pruning) is to search for a suitable subset of base classifiers that is better than using the whole ensemble. In ES, a single base classifier or an **Ensemble of Classifiers (EoC)** can be obtained via static or dynamic approach. The static approach selects only one subset of base classifiers during the training phase and uses it to predict all unseen samples. This, therefore, limits the flexibility of the selection procedure. Meanwhile, the dynamic approach selects only one classifier (dynamic classifier selection (DCS)) or an EoC (dynamic ensemble selection (DES)) with the most competencies in a defined region associated with each unseen sample. An issue of this approach is the dependence on the performance of techniques that define the region of competence (RoC) [19].

We first briefly review some static approaches to ensemble selection. In recent years, many statistical methods have been proposed to search for the optimal subset of ensembles. These can be mainly grouped into two categories: ordering-based methods and optimization-based methods.

Ordering-based methods try to order the base classifiers according to some criteria, and only the top classifiers are selected in the final ensemble. Some examples of the ordering criteria are validation error [20], kappa measure [20], complementary measure [21], and margin [21]. Recently Guo et al. [7] ordered the base classifiers using an evaluation measure considering both the margin and the diversity.

Optimization-based methods formulate ensemble selection as an optimization problem which can be solved by heuristic optimization or mathematical programming. In [22], Nguyen et al. encoded the classifiers and the features in a single chromosome and used a Genetic Algorithm (GA) to simultaneously search for the optimal set of base classifiers and the associated features. In [23], Chen et al. used the Ant Colony Optimization (ACO) to find the optimal set of base classifiers and meta-classifier in the ensemble system based on stacking. For the mathematical programming strategy, Zhang et al. [24] formulated the ES problem as a quadratic integer programming problem and used semi-definite programming to acquire an approximate solution. Although this method outperforms the other heuristics in the author's evaluation, fixing the number of selected base classifiers is a hindrance to efficient

performance. In [5], Li et al. theoretically analyzed the effect of diversity on the performance of voting. They concluded that the complexity of hypothesis space could be reduced by promoting larger diversity, resulting in better generalization performance. Their results were used to construct a greedy forward ensemble pruning method with diversity regularization.

In dynamic approaches, a base classifier or an EoC is selected to classify each test sample based on the competence level of the base classifiers computed according to some criteria on a local region of the feature space. Here the RoC can be defined by kNN methods (META-DES [25], KNORA Union and KNORA Eliminate [26]) and potential functions (in DES-KL and DES-P [27]). The criteria include accuracy [19], meta-learning [25], and probabilistic competency in the classifiers' prediction [28]. Comparison experiments in [6] indicated that a simple dynamic selection method like KNORA Union can be competitive or sometimes outperforms more complex methods. In [19], Cruz et al. showed that the effectiveness of DES methods is very sensitive to the choice of techniques that define the RoC. Moreover, the different distributions between the test set and the validation set in which the RoC of each test sample is obtained may degrade the system performance. A detailed review of the methods for DCS and DES can be found in [6, 19].

3. The model

3.1. Problem formulation

Assume that we have a committee of K experts $\{ K_k \}$ each of whom gives an answer to a problem. Classically, the answers from all experts are received and combined to obtain the final decision. However, some of the answers do not have high enough confidence and should be excluded from the final committee decision. Here we assume that each answer has its own confidence and that we prefer highly confident answers to those with low confidence before making the final decision. Moreover, we also assume that each of the experts has its own level of domain expertise (credibility) as they come with different background. Our approach takes account of each expert's credibility threshold and selects an expert's answer for aggregation if and only if its confidence is higher than the credibility threshold.

The proposed model is shown in Figure 1. First, experts of the committee give their answers to the problem. The confidence in each expert's answer is computed and then compared to the associated credibility threshold α_k. We determine which answers are included in the final aggregation based on the comparison between the expert's confidence and its credibility threshold:

$$\begin{cases} \text{confidence}(\mathcal{K}_k) \geq \alpha_k & K_k \text{ is selected to solve the problem} \\ \text{confidence}(\mathcal{K}_k) < \alpha_k & \text{otherwise} \end{cases} \quad (1)$$

There are two questions concerning the proposed model:

- How to measure the confidence of each expert's answer?
- How to determine the credibility threshold of each expert?

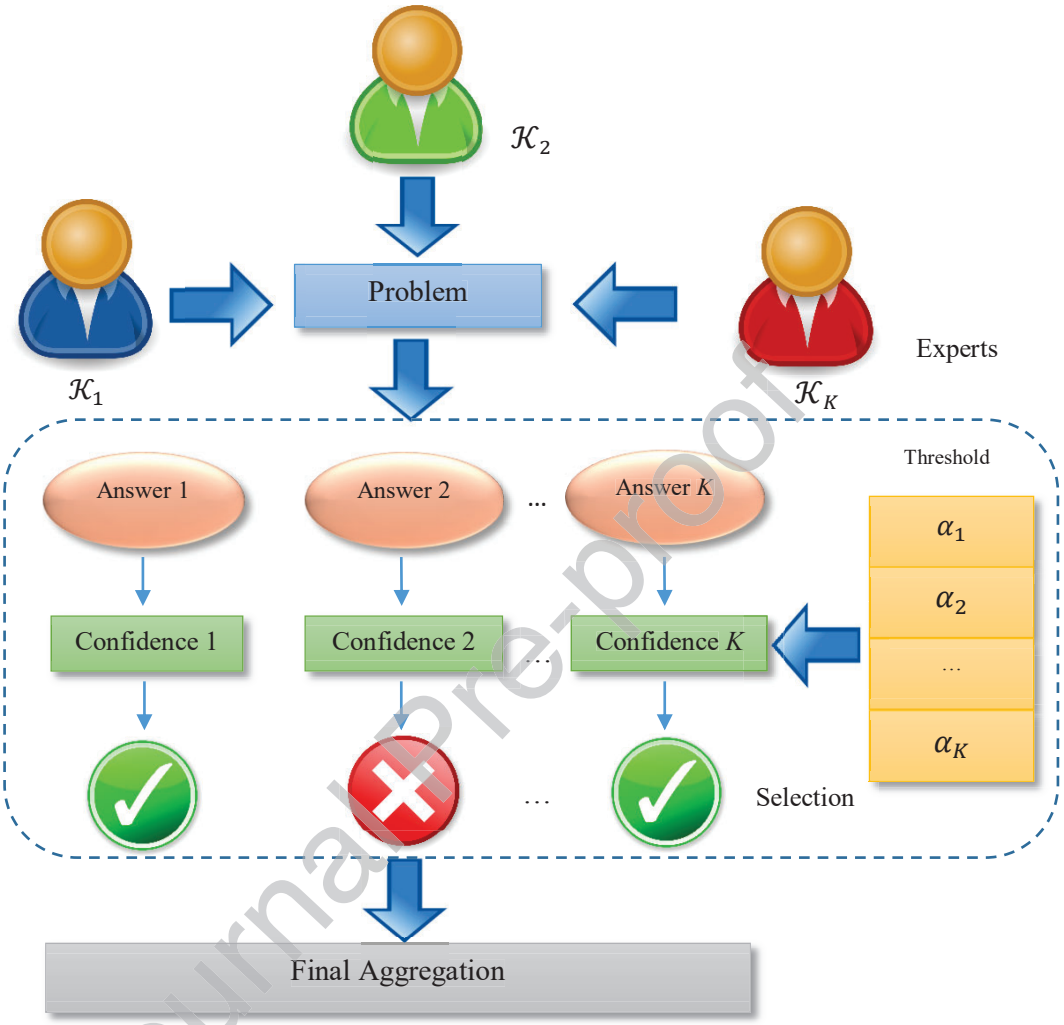


Figure 1. The proposed model to select answers based on expert's confidence

3.2. The proposed model for ensemble learning

The idea of expert selection based on the credibility threshold is applied to our ensemble system in which each base classifier is treated as an expert in the committee. Let $\mathcal{X}$ and $\mathcal{Y}$ denote the feature space and the set of class labels, respectively. Given a training set $\mathcal{D} = \{(\mathbf{x}_i, \hat{y}_i)\}\ i = 1, \ldots, N; \mathbf{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}, |\mathcal{Y}| = M$, from which we learn the set of base classifiers $\boldsymbol{\mathcal{H}} = \{h_k\}\ k = 1, \ldots, K$. For a sample $\mathbf{x}_i$, the classifiers' output is given in *Soft Label* [3, 4] which is the probability that $\mathbf{x}_i$ is assigned to y_m given by h_k: $\mathrm{P}_k(y_m|\mathbf{x}_i) \in [0,1]$ and $\sum_{m=1}^{M} \mathrm{P}_k(y_m|\mathbf{x}_i) = 1$. The outputs of all classifiers in $\boldsymbol{\mathcal{H}}$ on the training set are given by:

$$\mathbf{L} = \begin{bmatrix} \mathrm{P}_1(y_1|\mathbf{x}_1) \ldots \mathrm{P}_1(y_M|\mathbf{x}_1) & \cdots & \mathrm{P}_K(y_1|\mathbf{x}_1) \ldots \mathrm{P}_K(y_M|\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \mathrm{P}_1(y_1|\mathbf{x}_N) \ldots \mathrm{P}_1(y_M|\mathbf{x}_N) & \cdots & \mathrm{P}_K(y_1|\mathbf{x}_N) \ldots \mathrm{P}_K(y_M|\mathbf{x}_N) \end{bmatrix} \quad (2)$$

Meanwhile, the outputs of the K base classifiers on an observation $\mathbf{x}_i$ **(**called the meta-data of $\mathbf{x}_i$**)** is given in (3)

$$\mathbf{L}(\mathbf{x}_i) = [\mathrm{P}_1(y_1|\mathbf{x}_i) \ldots \mathrm{P}_1(y_M|\mathbf{x}_i) \quad \ldots \quad \mathrm{P}_K(y_1|\mathbf{x}_i) \ldots \mathrm{P}_K(y_M|\mathbf{x}_i)] \quad (3)$$

Clearly, the prediction of each classifier shows how confident that classifier support the decision. In fact, the decision is more convincing if the classifier's output for one class is much higher than that for the other classes, for example, $P_k(+1|\mathbf{x}_i) = 0.9$ and $P_k(-1|\mathbf{x}_i) = 0.1$ in a binary classification problem with two labels $\{-1,1\}$. In contrast, it is difficult to make a decision if the difference between the predictions is not significant, for example, when $P_k(+1|\mathbf{x}_i) = 0.51$ and $P_k(-1|\mathbf{x}_i) = 0.49$. In our method, each base classifier is associated with it a credibility threshold such that a classifier is included in the selected EoC if its confidence on a prediction is higher than its credibility threshold. This, therefore, provides a reliable mechanism for the decision making as we only include the prediction from base classifiers with high enough confidence.

To measure the confidence in the output of a base classifier h_k on a sample $\mathbf{x}_i$, we compute the entropy $\mathrm{E}_k(\mathbf{x}_i)$ of the output of h_k. Note that the smaller the entropy is, the lower the uncertainty in the prediction, and therefore the higher the confidence of the classifier's output, i.e. confidence is inversely proportional to entropy. Different entropy measures can be computed from the meta-data of $\mathbf{x}_i$ given by h_k, for example, Shannon entropy $\mathrm{ES}_k(\mathbf{x}_i)$ (4), Min entropy $\mathrm{EM}_k(\mathbf{x}_i)$ (5), and Collision entropy $\mathrm{EC}_k(\mathbf{x}_i)$ (6).

$$\mathrm{ES}_k(\mathbf{x}_i) = -\sum_{m=1}^{M} \mathrm{P}_k(y_m|\mathbf{x}_i) \log\{\mathrm{P}_k(y_m|\mathbf{x}_i)\} \quad (4)$$

$$\mathrm{EM}_k(\mathbf{x}_i) = \min_m\{-\log\{\mathrm{P}_k(y_m|\mathbf{x}_i)\}\} \quad (5)$$

$$\mathrm{EC}_k(\mathbf{x}_i) = -\log\{\sum_{m=1}^{M}\{\mathrm{P}_k(y_m|\mathbf{x}_i)\}^2\} \quad (6)$$

The credibility threshold of a classifier reflects how knowledgeable a classifier should be before it can be included in the final ensemble to predict for $\mathbf{x}_i$. We can define the credibility threshold of a classifier to be the maximum amount of uncertainty we are willing to accept before we lost confidence in its prediction. Let $\alpha_k \in [0, log(M)]$ be the credibility threshold for the classifier h_k, the classifier selection criterion is then given by:

$$\begin{cases} \mathrm{E}_k(\mathbf{x}_i) < \alpha_k \; h_k \text{ is selected to classify } \mathbf{x}_i \\ \mathrm{E}_k(\mathbf{x}_i) \geq \alpha_k \;\; h_k \text{ is not selected to classify } \mathbf{x}_i \end{cases} \quad (7)$$

To compute the credibility threshold of each of the base classifiers, we compute the empirical 0-1 loss of the ensemble over the entire training observations. The ensemble 0-1 loss on a sample $\mathbf{x}_i$ with $\boldsymbol{\alpha} = \{\alpha_k, k = 1, \dots, K\}$ is given by:

$$l(\mathbf{x}_i, \boldsymbol{\alpha}) = \mathbb{I}\left[\hat{y}_i \neq \max_{y_m;\, m=1,\dots M} \sum_{k=1}^{K} \mathbb{I}[\mathrm{E}_k(\mathbf{x}_i) < \alpha_k] \mathrm{P}_k(y_m|\mathbf{x}_i)\right] \tag{8}$$

where $\mathbb{I}[.]$ is the indicator function which returns 1 if the argument is true and 0 otherwise. The empirical 0-1 loss over the entire training observations $\mathcal{L}_{0-1}$ is given by (9) and the value of $\boldsymbol{\alpha} = \{\alpha_k\}$ is obtained by minimizing $\mathcal{L}_{0-1}$ subjected to $\alpha_k \in [0, log(M)]\ k = 1, \dots, K$

$$\begin{cases} \min \mathcal{L}_{0-1}(\boldsymbol{\alpha}) = \frac{1}{N} \sum_{i=1}^{N} l(\mathbf{x}_i, \boldsymbol{\alpha}) \\ \text{subject to } \alpha_k \in [0, log(M)]\ k = 1, \dots, K \end{cases} \tag{9}$$

3.3. The algorithms

The proposed approach is applied to the heterogeneous ensemble system in which the EoC is obtained by training K different learning algorithms on a given training set and the prediction of all base classifiers is aggregated by a combining algorithm to obtain the final prediction [1, 2]. The training phase of the proposed method is presented in Algorithm 1. Given the training set $\mathcal{D}$, we learn the set of classifiers $\mathcal{H}$ by using K learning algorithms $\{\mathcal{K}_k\}$ (step 1). After that, we use T-fold cross validation to generate the meta-data of $\mathcal{D}$ (step 2-9). We then compute the entropy of each classifier on the meta-data of each sample using one of Eqn. (4), (5) or (6). The credibility thresholds of the base classifiers are obtained by minimizing the empirical loss function $\mathcal{L}_{0-1}$ in Eqn (9).

Nowadays, many applications of Evolutionary Algorithm (EA), a family of robust and effective search method, have been found in research for ensemble selection approach [29]. Classical optimization methods may be more efficient than EA when solving linear or strongly convex problems, for non-differentiable, discontinuous, or multi-modal objective functions that appear in many real-life applications, EA can be a better choice. In this study, we use the Artificial Bee Colony (ABC) algorithm, a popular EA, to search for the optimal solution of the optimization problem in $\mathcal{L}_{0-1}$ (9). The ABC algorithm, proposed by Karaboga [30], is a member of the swarm intelligence family, a meta-heuristic search algorithm inspired by the intelligent foraging behavior of honey bee swarm. This algorithm provides a simple but competitive tool in searching for the optimal solution for optimization problems [31]. Moreover, aside from two common parameters (i.e. the number of candidates in each generation and the number of generations) as in all population-based optimization algorithms, ABC only has one control exploration parameter which makes it more practical than many other population-based algorithms.

In ABC, first, we randomly initialize the population of $nPop$ food sources representing the possible solutions of the optimization problem. There are three types of bees in the swarm: worker bee, onlooker bee, and scout. The number of worker bees and onlooker bees is equal to the number of solutions in the swarm. Worker bees exploit the food sources and share the information of nectar amount (the fitness of the solutions) to the onlooker bees. The onlooker bees tend to select good food sources. A food source becomes exhausted if it does not improve in a predetermined number of cycles. The worker bees of exhausted food sources become scouts. Scouts then start searching for new food sources.

In detail, we first randomly initialize the population of $nPop$ possible solutions of the optimization problem. Let $\boldsymbol{\alpha}_i = \{\alpha_{i1}, \dots, \alpha_{iD}\}$ be the i^{th} solution in the swarm where D is the dimension size. Each member of $\boldsymbol{\alpha}_i$ is produced randomly in $[lb_k, ub_k]$ according to Eq. (10):

$$\alpha_{i,k} = lb_k + rand(0,1) \times (ub_k - lb_k) \tag{10}$$

where lb_k and ub_k are the lower bound and upper bound of the k^{th} dimension and $rand(0,1)$ is a random number within $[0,1]$. After initialization, other three phases are conducted as following:

Employed Bee Phase: Each employed bee generates a new candidate solution $\mathbf{v}_i$ in the neighborhood of $\boldsymbol{\alpha}_i$ as:

$$v_{i,k} = \alpha_{i,k} + \phi_{i,k} \times (\alpha_{i,k} - \alpha_{j,k}) \tag{11}$$

where $\alpha_{i,k}$ and $\alpha_{j,k}$ is the k^{th} components of $\boldsymbol{\alpha}_i$ and another candidate $\boldsymbol{\alpha}_j$ which is randomly selected from the swarm, k is a randomly selected index from the set $\{1, \dots, D\}$, and ϕ_{ik} is a random number within $[-1,1]$. The value of $v_{i,k}$ is stochastic and belongs to (see Fig.S1 in the Supplement Material for the illustration):

$$\left[\alpha_{i,k} - \left|\alpha_{i,k} - \alpha_{j,k}\right|, \alpha_{i,k} + \left|\alpha_{i,k} - \alpha_{j,k}\right|\right] = \begin{cases} \left[\alpha_{j,k}, 2\alpha_{i,k} - \alpha_{j,k}\right] & if\ \alpha_{i,k} \geq \alpha_{j,k} \\ \left[2\alpha_{i,k} - \alpha_{j,k}, \alpha_{j,k}\right] & otherwise \end{cases} \tag{12}$$

If the fitness value of the new candidate solution $\mathbf{v}_i$ is better than that of $\boldsymbol{\alpha}_i$, $\boldsymbol{\alpha}_i$ is replaced by $\mathbf{v}_i$. Otherwise, $\boldsymbol{\alpha}_i$ remains unchanged.

Onlooker Bee Phase: After all the employed bees finish the search process, they pass the information of the solutions to the onlooker bees. The solutions are selected via a roulette wheel selection mechanism where a solution with a higher fitness will have a higher chance to be selected. The probabilistic selection for the i^{th} solution is given by:

$$\mathrm{P}(\boldsymbol{\alpha}_i) = \frac{fit_i}{\sum_{i=1}^{SN} fit_i} \tag{13}$$

The fit_i is the fitness function of the i^{th} solution in the swarm given by:

$$fit_i = \exp\left(\frac{-\mathcal{L}_{0-1}(\boldsymbol{\alpha}_i)}{(\sum \mathcal{L}_{0-1}(\boldsymbol{\alpha}_i))/SN}\right) \quad (14)$$

where $\mathcal{L}_{0-1}(\boldsymbol{\alpha}_i)$ is the classification error rate of the method if the configuration associated with $\boldsymbol{\alpha}_i$ is selected (see Eq. (9)).

Scout Bee Phase: If the solution cannot be improved over a predefined number of cycles $maxC$, the solution is abandoned and the employed bee of the abandoned solution becomes a scout. The scout discovers a new solution according to Eq. (10) and the abandonment counter for $maxC$ is reset

The training phase of the proposed method is described in Algorithm 1. We generate the meta-data of the training observations by using T-fold cross validation. If a separate validation set $\mathcal{V}$ is available, steps 2-9 in Algorithm 1 can be replaced by a step that acquires the meta-data of $\mathcal{V}$ using the classifiers in $\boldsymbol{\mathcal{H}}$. After getting the meta-data of the training observations, we compute the entropy of each classifier's output for each observation (steps 10-12). The ABC algorithm is then used (steps 13-19) to find the optimal value of the credibility threshold for each base classifier.

In the classification phase in Algorithm 2, the output of a base classifier h_k on an unlabeled sample $\mathbf{x}^u$ is obtained as *Soft Label* given by $\{\mathrm{P}_k(y_m|\mathbf{x}^u)\}$. We then computed the entropy $\mathrm{E}_k(\mathbf{x}^u)$. Based on the selection criteria (7), we determine whether h_k should be included in the ensemble. All the selected base classifiers are added to EoC $\boldsymbol{\mathcal{S}}$ to predict the label for $\mathbf{x}^u$. The Sum Rule is applied to the outputs of the classifiers in $\boldsymbol{\mathcal{S}}$ to provide the final prediction $\hat{y}$. It is noted that an exception to step 7 in Algorithm 2 is performed where the whole ensemble is used if no classifier satisfies the selection criteria in step 5.

Algorithm 1: Training phase

Input: Training set $\mathcal{D}$, K learning algorithms $\{\mathcal{K}_k\}$, maximum number of iteration:$maxT$, population size: $nPop$, abandonment limit parameter: $maxC$

Output: The optimal credibility threshold $\{\hat{\alpha}_k\}$ and $\boldsymbol{\mathcal{H}}$

1. Learn K classifiers $\boldsymbol{\mathcal{H}} = \{h_k\}$ on $\mathcal{D}$ using $\{\mathcal{K}_k\}$
   (Generate the meta-data)
2. Meta-data $\mathbf{L} = \emptyset$
3. $\mathcal{D} = \mathcal{D}_1 \cup \ldots \cup \mathcal{D}_T, \quad \mathcal{D}_i \cap \mathcal{D}_j = \emptyset \ (i \neq j)$
4. For each $\mathcal{D}_i$
5. $\quad \mathcal{D}^{-i} = \mathcal{D} - \mathcal{D}_i$
6. $\quad$ Learn ensemble of classifiers $\mathcal{C}^i$ on $\mathcal{D}^{-i}$ using $\{\mathcal{K}_k\}$

7. Classify samples of $\mathcal{D}_i$ by $\mathcal{C}^i$
8. Add outputs on samples in $\mathcal{D}_i$ to $\mathbf{L}$ (2)
9. End For

(Compute the entropy on the meta-data of each observation)

10. For each $\mathbf{x}_i$ in $\mathcal{D}$
11. Compute $\mathrm{E}_k(\mathbf{x}_i)$ from $\mathbf{L}$
12. End For

(Use ABC method to find the optimal credibility threshold)

13. For each candidate $\boldsymbol{\alpha}_i$ generated by ABC
14. For each $\mathbf{x}$ in $\mathcal{D}$
15. Compute $l(\mathbf{x}, \boldsymbol{\alpha}_i)$ by (8)
16. End

Compute $\mathcal{L}_{0-1}(\boldsymbol{\alpha}_i)$ by (9)

17. Compute the fitness value of $\boldsymbol{\alpha}_i$ by (14) and $\mathrm{P}(\boldsymbol{\alpha}_i)$ by (13)
18. End
19. Select optimal $\{\hat{\alpha}_k\}$ with the largest fitness value from the last population of ABC algorithm
20. Return $\{\hat{\alpha}_k\}$ and $\boldsymbol{\mathcal{H}}$

Algorithm 2: Classification phase

Input: Unlabeled sample $\mathbf{x}^u$, optimal credibility threshold $\{\hat{\alpha}_k\}$, and the set of classifiers $\boldsymbol{\mathcal{H}}$

Output: Predicted class label for $\mathbf{x}^u$

1. Obtain the meta-data of $\mathbf{x}^u$ by using $\boldsymbol{\mathcal{H}}$
2. Selected classifiers $\boldsymbol{\mathcal{S}} = \emptyset$
3. For each k^{th} classifier
4. Compute $\mathrm{E}_k(\mathbf{x}^u)$
5. If $\mathrm{E}_k(\mathbf{x}^u) < \hat{\alpha}_k$: $\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{S}} \cup h_k$
6. End For
7. If $\boldsymbol{\mathcal{S}} = \emptyset$: $\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{H}}$
8. $\hat{y} = \max_{y_m;\, m=1,\ldots M} \sum_{h_k \in \boldsymbol{\mathcal{S}}} \mathrm{P}_k(y_m|\mathbf{x}^u)$

4. Experimental Studies

4.1. Experimental datasets

We compared the proposed method and benchmark algorithms by conducting experiments on a number of datasets from some data sources such as UCI (http://archive.ics.uci.edu/ml/datasets.html), OpenML (https://www.openml.org), and MOA library (https://moa.cms.waikato.ac.nz) as shown in Table 1. These datasets are popular in experiments with various classification systems. Here the datasets were selected in a diverse way to ensure objectivity in the comparison. The number of observations is from hundreds (e.g., Hepatitis and Wine) to millions (e.g., Poker, BNG-Zoo, and BNG-Bridge). The number of dimensions is from 3 (e.g. Haberman) to 7129 (e.g. Leukemia and CNS) while the number of classes varies from 2 (e.g., Artificial and Heart) to 100 (e.g. Plant-Margin and Plant-Shape).

Table 1. THE UCI DATASETS USED IN THE EXPERIMENT

Dataset name	# of observations	# of classes	# of dimensions
Abalone	4174	3	8
Appendicitis	106	2	7
Artificial	700	2	10
Assetnegotiation-F2	1000000	2	5
Assetnegotiation-F3	1000000	2	5
Assetnegotiation-F4	1000000	2	5
Australian	690	2	14
Banana	5300	2	2
Biodeg	1055	2	41
Blood	748	2	4
BNG-Bridges	1000000	6	12
BNG-Zoo	1000000	7	17
Breast-Tissue	106	6	9
Bupa	345	2	6
Cleveland	297	5	13
CNS	60	2	7129
Colon	62	2	2000
Contraceptive	1473	3	9
Dermatology	358	6	34
Dowjones-1985-2003	138166	30	8
Duke	44	2	7129
Electricity	45312	2	8
Fertility	100	2	9
Glass	214	6	9
Haberman	306	2	3
Hayes-Roth	160	3	4
Heart	270	2	13
Hepatitis	80	2	19
Hyperplane	1000000	2	10
Iris	150	3	4
Isolet	7797	26	617
Led7digit	500	10	7
Letter	20000	26	16
Leukemia	72	2	7129
Magic	19020	2	10
Mammographic	830	2	5
Marketing	6876	9	13
Multiple-Features	2000	10	649
Musk	476	2	166

Page-Blocks	5472	5	10
Phoneme	5404	2	5
Pima	768	2	8
Plant-Margin	1600	100	64
Plant-Shape	1600	100	64
Poker	1025009	10	10
RandomTree	1000000	2	10
Skin-NonSkin	245057	2	3
Sonar	208	2	60
Spambase	4601	2	57
Svmguide	391	3	20
Tae	151	3	5
Texture	5500	10	40
Twonorm	7400	2	20
Vehicle	846	4	18
Vertebral	310	3	6
Waveform-w-Noise	5000	3	40
Waveform-wo-Noise	5000	3	21
Wdbc	569	2	30
Wine	178	3	13
Wine-Red	1599	6	11
Wine-White	4898	7	11
Yeast	1484	10	8

4.2. Experimental Settings and Benchmark Algorithms

To construct the heterogeneous ensemble system, we used three learning algorithms: LDA, Naïve Bayes, and kNN. It is noted that these algorithms use significantly different learning strategies, therefore they produce diverse predictions needed in an ensemble system [1]. In this study, the value of k in kNN classifier was set to 5, denoted as $k\text{NN}_5$ and k was not optimally chosen for the experimental datasets. Here we only aim to demonstrate that an ensemble system built using simple learning algorithms can achieve high performance. For the ABC algorithm, the maximum number of iterations $maxT$ was set to 100, the number of food sources $nPop$ was set to 50, and abandonment limit parameter $maxC$ was set to $round(0.6 \times K \times nPop)$

We performed extensive comparative studies using a number of existing algorithms as benchmarks: two well-known ensemble methods, namely RotBoost [32] (which is a combination between two powerful ensemble methods namely Rotation Forest [33] and AdaBoost [34]) and Random Subspace [35] with 200 decision tree classifiers (See Fig.S2 and discussion in the Supplement Material for the detail. This values also was used in some previous studies such as [36]), and Decision Template [3] which also captures the uncertainty between the classifiers in the combining algorithms. For the ensemble selection methods, we selected the three top-performing DCS/DES methods, namely META-DES [25], KNORA UNION and KNORA ELIMINATE [26] reported in the recent survey [6, 19] as benchmark algorithms. The number of nearest neighbors in these dynamic methods was set to 7 since it is

the optimal value reported to provide the best performance [19, 25]. We also selected the ACO methods [23] as benchmark algorithms since they belong to the static ensemble selection approach. Finally, a Genetic Algorithm-based method which searches for the best subset of meta-data to enhance the ensemble performance was also selected as a benchmark algorithm (called GA Meta-data) [37].

For the datasets with less than one million instances, we run 10-fold cross validation 3 times to obtain 30 test results for each dataset. Otherwise, we only run 10-fold cross validation 1 time on the datasets with more than 1 million instances. We then computed the mean and variance of the classification error rates and reported them in Tables 2 and 3.

4.3. Statistical Test of Significance

We used the Friedman test to test the null hypothesis that "all methods perform equally". The test was conducted for multiple methods on multiple datasets. If the null hypothesis is rejected, we used the Nemenyi post-hoc test to compare all pairwise combinations of the methods on multiple datasets. For these tests, the level of significance was set to 0.05. We used the software package provided at http://sci2s.ugr.es/keel/multipleTest.zip for the Friedman and Nemenyi tests.

We also conducted the Wilcoxon signed-rank test [15] to test the null hypothesis that "two methods perform equally on a dataset". The null hypothesis of this test is rejected if the P-value is smaller than a given confidence level α. Here, α was set to 0.05. We used the software package provided in the Matlab library for the Wilcoxon signed-rank test.

5. Results and Discussion

5.1. Different entropy formulations

In this study, three different entropy measures (4)-(6) were used to quantify the information content in the output of the base classifiers. Here we aimed to assess the influence of entropy measure on the performance of the proposed method. The experimental results are shown in Figure 2, with the detailed results provided in Table S1 the Supplementary Material. Clearly, the entropy measure only affects slightly the classification error rates on the experimental datasets. The most significant difference in classification error rate using Shannon, Min, and Collison entropy is on the Hepatitis dataset, in which the results were 0.1542, 0.1458, and 0.1333 respectively. Meanwhile, on the other datasets, the differences in the classification error rate for the 3 entropy measures were very small. In practice, therefore, any one of the 3 entropy measures can be used with the proposed method. In what follows, we used Shannon entropy to compute the credibility threshold for each base classifier.

5.2. Comparison of the benchmark algorithms

We conducted the '*multiple methods-multiple datasets*' statistical test on the performance score of all methods. We also compared the performance of the proposed method and each benchmark algorithm on each dataset by using the Wilcoxon test. The purpose of this test was to see which method, among the pair, performs better for a dataset. Some observations can be made from Table 2, 3, and Figure 3, 4:

- The P-value of the Friedman test, in this case, is 3.324E-06, therefore we rejected the null hypotheses that the performances of all methods are equal. From the Nemenyi test results shown in Figure 3, the proposed method is better than KNORA UNION, Decision Template, RotBoost, META-DES, GA Meta-data, and ACO, while there is no statistical difference in the pairwise comparison between the proposed method and Random Subspace and KNORA ELIMINATE. Among all methods, the proposed method ranks first (rank value 3.27), followed by KNORA ELIMINATE (rank value 4.68) and Random Subspace (rank value 4.77). The proposed method ranks first in 9 cases (14.52 %), ranks second in 14 cases (22.58%). The proposed method only performs poorly on 4 datasets: Cleveland and Appendicitis (rank 7^{th}), Mammographic and Pima (rank 6^{th}), however the differences between classification error rate of the proposed method and the benchmark algorithms are not very significant. Meanwhile, GA Meta-data and ACO are the two poorest methods in the experiment where GA Meta-data ranks fifth (rank value 5.78) and ACO ranks sixth (rank value 5.89).

- Compare to Decision Template, we rejected 28 null hypotheses of the Wilcoxon test, in which in 26 cases, the proposed method is better than this method. Compare to Random Subspace, 38 null hypotheses in the Wilcoxon test was rejected in which the proposed method wins on 26 datasets and loses on 12 datasets. Our method is significantly better than Random Subspace on some datasets such as 0.2124 vs. 0.2771 on Artificial, 0.1069 vs. 0.3746 on Banana, 0.2673 vs. 0.4460 on Led7digit, 5.2777E-04 vs. 2.6171E-03 on Skin_NonSkin, 0.0048 vs. 0.0238 on Texture, and 0.4050 vs. 0.4975 on Yeast. The proposed method also outperforms RotBoost on 27 datasets and loses on 11 datasets. In comparison to the 5 ensemble selection methods, the proposed method shows outstanding performance. The proposed method is better than GA Meta-data (28 wins and 7 losses), ACO (28 wins and 9 losses), KNORA Eliminate (25 wins and 6 losses), KNORA Union (32 wins and 3 losses), and META-DES (30 wins and 8 losses). The detail results of Wilcoxon test are shown in Table S2 in the Supplement Material.

Both the '*multiple methods-multiple datasets'* test and the Wilcoxon test demonstrated convincingly the better performance of our method compared to the benchmark algorithms on the experimental datasets.

We also noted some observations based on the three properties of experimental datasets in Table 1-3:

- RotBoost is the best method for the 10 large scale datasets in our experiment as it ranks first on average on these datasets with rank value 3. The proposed method ranks third on these datasets with rank value 3.9. In contrast, the proposed method performs well on 21 small scale datasets i.e. datasets with less than 500 observations. Specifically, on 17 small datasets with low dimensions, the proposed method ranks first with rank value 3.74. GA Meta-data is the poorest methods on these small datasets in which it ranks last with rank value 6.29. On 4 small datasets with high dimension (Colon, Duke, CNS, and Leukemia), the proposed method ranks first on one dataset (Leukemia) and ranks second on two datasets (Duke and CNS). META-DES is the poorest method in this case in which its rank value is 7.38. The detail of ranking on these datasets can be found in Table S6-S8 in the Supplement Material.
- On 30 low dimension datasets i.e. datasets that have less than or equal to 10 features, the proposed method is better than all benchmark algorithms in term of average ranking. Specifically, the proposed method ranks first with rank value 3.70 and is better than the second method KNORA ELIMINATE with rank value 4.30. On 7 datasets with more than 100 features, the proposed method ranks first on 2 datasets (Isolet and Leukemia) and does not rank below 5 on any datasets (see Table S9-S10 in the Supplement Material). In term of the number of class labels, the proposed method outperforms the benchmark algorithms on datasets with a large number of class label or binary label. On 6 datasets with more than 26 class labels and on 31 binary datasets, the proposed method has rank value only 2.6 and 3.24. RotBoost and META-DES seem to be poor methods for datasets with high dimension or large number of class label as these methods rank last in these cases (with rank value higher than 7.2) (see Table S11-S12 in the Supplement Material).

5.3. Different numbers of base classifiers

We built a new heterogeneous ensemble system with 7 learning algorithms to assess the difference in performance scores with different number of base classifiers. Four new base classifiers, namely the Nearest Mean Classifier (denoted by NMC), Decision Tree C4.5, L2-Loss Linear SVM (denoted by L2LSVM), and Discriminant Restrict Boztman Machine

(DRBM) were added to form the new ensemble. Here we used the Decision Tree C4.5 from the Statistics and Machine Learning Toolbox of Matlab. The NMC and DRBM were obtained from PRTools (available at http://prtools.org/), and the L2LSVM was obtained from the LibLinear library (available at https://www.csie.ntu.edu.tw/~cjlin/liblinear/). It is noted that the performance score of RotBoost and Random Subspace do not change in this experiment.

We once again conducted the Friedman test to compare multiple methods on multiple datasets. In this case, the P-value computed by the test is 2.2E-16. Hence, we rejected the null hypotheses that all methods perform equally and conducted the Nemenyi post-hoc test for all pairwise comparisons among the 9 methods (see Figure 5). The proposed method ranks first among all methods (rank value 2.35). In this case, Decision Template performs well, ranked second with rank value 4.06. Our method is better than all benchmark algorithms based on the Nemenyi test. This result is more significant than the comparison using 3 classifiers.

Clearly, the proposed method continues to outperform each benchmark algorithm and the result is also more significant than the comparison using 3 classifiers concerning Wilcoxon signed-rank test (see Fig 6 and Table S3 in the Supplement Material). Our method is better than Decision Template (it wins in 30 cases and loses in 6 cases), better than KNORA Union (it wins in 33 cases and loses in only 1 case), better than KNORA Eliminate (it wins in 50 cases and does not lose on any case). Our method also performed considerably better than GA Meta-data and ACO as it wins both methods on 42 and 44 datasets, respectively. Our method continues to be better than RotBoost (it wins on 37 datasets and loses on 2 datasets) and Random Subspace (it wins on 33 datasets and loses on 8 datasets). The '*multiple methods-multiple datasets'* test and the Wilcoxon test in the case of using 7 base classifiers once again show the better performance of our method on the experimental datasets.

Similar observations are found when we compared the proposed method and the benchmark algorithms based on the three properties of experimental datasets. The differences in average ranking in this case is even more significant than the case using 3 classifiers. For example, the proposed method is remarkably better than all benchmark algorithms on 10 large scale datasets on average in which its rank value is 1.45 compared to 3.90 of the second method GA Meta-data. The detail can be found in Table S13 in the Supplement Material.

We also compared the performance of the proposed method constructed using 3 or 7 learning algorithms. In general, using 7 learning algorithms gives better results than using 3 learning algorithms, for example, on Wine-White (classification error of 0.38 vs. 0.4539), Spambase (0.0734 vs. 0.0975), Sonar (0.1681 vs. 0.2161), Skin_NonSkin (4.2847E-04 vs. 5.2777E-04), Musk (0.0672 vs. 0.1337), and Wine-Red (0.3677 vs. 0.4157). Only on some datasets like Haberman and Fertility, the ensemble with 3 learning algorithms is about 2% better than the ensemble with 7 learning algorithms (see Fig.S3 in the Supplement Material).

Therefore, using more learning algorithms to construct the ensemble produces a more diverse ensemble system from which our ensemble selection mechanism can select from.

5.4. Analysis of computational complexity

The complexity of the training process of the three EA-based methods (ACO, GA Meta-data and proposed method) is $\mathcal{O}\left(\max\left(T \times \arg\max_{k=1,\dots,K} \mathcal{O}(\mathcal{K}_k), (\text{searching process})\right)\right)$ in which $\mathcal{O}(\mathcal{K}_k)$ denotes the complexity of learning algorithm $\mathcal{K}_k$, $\mathcal{O}\left(T \times \arg\max_{k=1,\dots,K} \mathcal{O}(\mathcal{K}_k)\right)$ is the time complexity of generating meta-data of training set via running T-fold cross-validation and $\mathcal{O}(\text{searching process})$ is the time complexity of each procedure to search for optimal configuration. It is noted that the searching process includes generating a candidate and evaluating its fitness in a generation. For GA Meta-data and ACO, we used Decision Tree C4.5 (its time complexity is $\mathcal{O}(D \times N)$ in which D is the data dimension) to train on the meta-data of training observations associated with each candidate. Therefore the complexities of the search process of GA Meta-data and ACO method are $\mathcal{O}\left((candidate\_generation\_GA + D_{GA}^* \times N) \times NPop \times maxT\right)$ and $\mathcal{O}\left((candidate\_generation\_ACO + D_{ACO}^* \times N) \times NPop \times maxT\right)$ in which D_{GA}^* and D_{ACO}^* are dimensions of meta-data associated with a candidate in GA Meta-data and ACO, respectively, $(D_{GA}^*, D_{ACO}^* < M \times K)$. In the proposed method, each candidate has K dimensions which are the credibility thresholds of the base classifiers. To evaluate the fitness of a candidate, we conducted K comparisons between the thresholds and the entropy value on each meta-data of N training instances. Therefore, the time complexity of the searching process of the proposed method is $\mathcal{O}\left((candidate\_generation\_ABC + K \times N) \times NPop \times maxT\right)$. It is observed that the computational complexities of these three methods are mainly due to the difference in the process to generate candidates. ABC method is slow in coverage as it focuses more on exploration when generating new candidates [38], resulting in higher running time than GA and ACO on some datasets. Moreover, the dimension of data does not cause the differences in computational complexity of these methods.

We reported the training time of the ACO, GA-Meta data, and proposed method on 10 datasets in Table 5 (We tested these methods in Matlab running on a PC with Intel Core i5 with 2.5 GHz processor and 8G RAM). Some observations can be made in accordance with the analysis of computational complexity mentioned above:

- ACO took the least running time among the three methods. This method searches for the optimal set of classifiers on a narrow search space. Hence, ACO can quickly converge to obtain the optimum in the training process.

- On 4 datasets with a large number of class labels (Plant-Margin, Plant-Shape with 100 class labels, Isolet with 26 class labels and Poker with 10 class labels): GA Meta-data took much more running time than ACO and the proposed method. For example, on Isolet dataset, GA Meta-data took 3518.63 seconds, 10 times longer than the proposed method. It is noted that the dimension of metadata is the product of the number of class labels and the number of base classifiers. For datasets with a large number of class labels, the dimension of meta-data is high, resulting in a large solution space for GA to search for the optimum.

- On AssetNegotiation-F2, a dataset with a large number of observations, the running time of the proposed method was much higher than ACO and GA Meta-data. This is a binary dataset, which means the dimension of the meta-data is small. GA Meta-data, in this case, can obtain the optimum in shorter running time than the proposed method.

- On 2 small datasets i.e. Haberman and Wine with small number of observations, dimensions, and class labels, the proposed method also took longer running time than the two baselines. This is also the case for several high dimension datasets with a small number of observations and class labels such as Leukemia and CNS.

The running time of the proposed method can be reduced by using parallel implementation which is often used to accelerate population-based algorithms [39]. Here, we discuss two models to parallelize the ABC algorithm, which not only achieves significant speedup but also keeps the quality of solutions. The first model is the master-slave model in which there is one population maintained by a master processor and computation is conducted by many slave processors or the coarse-grained (subpopulation) model like in [40] in which each subpopulation runs the algorithm in one processor independently and exchanges the results with other subpopulation. The detail of parallel implementation for the ABC algorithm can be found in [39, 40]

5.5. Discussions

The Decision Template method averages the meta-data associated with each class label to obtain the decision template. For datasets like Fertility (80% of instances belongs to the first class label) or Hayes-Roth (80% of instances belongs to the first and second class labels), the base classifiers tend to predict the dominated class label. The meta-data, therefore, is very similar among all the class labels (see Fig.S4 in the Supplement Material). In this case, Decision Template usually performs poorly compared to the other heterogeneous ensemble methods.

Meanwhile, the ACO and GA Meta-data are the two poorest methods in our experiment. ACO used the Ant Colony Optimization algorithm to search for the optimal subset

of base classifiers for all the test samples. These methods, therefore, are less flexible than the proposed method since our method selects a particular EoC for each test sample. For GA Meta-data, the Genetic Algorithm is used to select the optimal set of meta-data for the combiner. As mentioned earlier, the meta-data is the concatenation of predictions from the base classifiers. In fact, the meta-data can be viewed as scaled data from feature domain to posterior probability domain so that in the new domain, each observation is re-scaled to a different position compared to that in the feature domain. This improves the discriminative characteristic of data on some datasets in case of correct predictions where observations that belong to the same class will have similar posterior probabilities and stay close together in the posterior domain. However, in cases of wrong predictions, the discriminative characteristic of data will be downgraded. Consequently, learning in the posterior probability domain by a traditional learning algorithm (for example Decision Tree in the original papers or the proposed method) can obtain either good or poor performances. On the Hepatitis dataset, for example, some algorithms like DRBM and $k\text{NN}_5$ perform poorly with a classification error rate of 0.22 and 0.1938, respectively, and greatly lead to poorer discriminative characteristic in the meta-data for Decision Tree to learn on. This explains why GA Meta-data obtains high classification error rate on this dataset.

Random Subspace method where subsets of features are randomly selected to form the new training sets to learn the base classifiers generally performs better on very high dimensional datasets. However, for datasets with a small number of features, the new training set generated by Random Subspace is usually not diverse enough or not a representative of the underlying classes, resulting in poor performance. We illustrate an example on binary Hepatitis dataset which has 80 observations (13 observations belong to the first class label and the others belong to the second class label) and 19 features. On this dataset, the classification error rate of the proposed method is 0.1542 when using 7 classifiers which is 2.09% worse than Random Subspace. For Random Subspace, we randomly select $d = round(\sqrt{D}) = round(\sqrt{19}) = 4$ features from 19 features to create the subspace from which we get the original training data to generate the new training sets. (This choice of d is common in some machine learning libraries such as scikit-learn). In this case, the subspace of features can create 200 significantly diverse training sets, which makes Random Subspace method perform well on this data. It is observed that Random Subspace method is also significantly better than the proposed method (0.0805 vs. 0.1337) on the Musk dataset with 166 features by the same reason.

RotBoost performs well on some large scale datasets compared to the proposed method and other heterogeneous ensemble methods using 3 base classifiers. However, RotBoost performs poorly on datasets with high dimensions or a large number of class labels. When we used 7 base classifiers to construct the heterogeneous ensemble methods, RotBoost

underperforms in the comparison to the proposed method on the experimental datasets. RotBoost uses Rotation Forest to transform the data and then use AdaBoost to train the ensemble on transformed data. Like in GA Meta-data, this transformation may improve or downgrade the discriminative characteristic of data on some datasets which significantly affect the performance of AdaBoost in the second stage of RotBoost.

KNORA Union has average performance with mid-ranking whereas KNORA Eliminate is the second best method for 3 base classifiers but is the poorest for 7 base classifiers in our experiment. META-DES ranks 7th and 4th when using 3 and 7 base classifiers, respectively. The three methods are in the DES family where they aim to dynamically select a specific set of base classifiers to predict for each test sample based on the classifiers' performance on the RoC. Cruz et al. [19] showed that DES method's performance is mainly dependent on the choice of techniques that define the RoC. Moreover, different distributions between the test set and the validation set where the RoC of each test sample is obtained can degrade the system's performance.

The proposed method ranks first in the Nemenyi post hoc test for both ensembles with 3 and 7 base classifiers. Our method is more flexible than static ensemble selection as it selects a different EoC for different test sample dynamically based on the confidence in the classifier's prediction on the test sample. The credibility threshold allows us to observe the behavior of a classifier on each dataset. Table 6 shows the credibility threshold for the 3 base classifiers and the maximum entropy value in each dataset. Based on Eqn. (7) we can see that the classifiers with high credibility threshold would have more chance to be included in the aggregation. For example, on Biodeg datasets, the credibility thresholds of LDA, Naïve Bayes, and $k\text{NN}_5$ are 0.9981, 0.0019, and 1.0000 respectively. This means the Naïve Bayes has very little chance and LDA has very high chance to be selected for aggregation. Since the credibility threshold of $k\text{NN}_5$ is equal to the maximum of the entropy value in the dataset, it is always included in the aggregation for all samples. Meanwhile, on the Haberman dataset, the credibility threshold of $k\text{NN}_5$ is equal to 0, which means it would never appear in the aggregation for this dataset.

6. Conclusions

In this study, we proposed a novel ES method by selecting the base classifiers with high confidence in their prediction, taking into account the level of expertise of each base classifier. We quantified the classifier's level of expertise for a problem by computing its credibility threshold based on minimizing the 0-1 loss function on all the labeled observations in a training set. This constitutes the static aspect of our ensemble selection approach. The dynamic aspect of our ensemble selection approach is done by measuring the base classifier's confidence in its prediction of a test sample using an entropy measure. A base classifier is selected for aggregation if its entropy value (reflecting its uncertainty in prediction) is less than

or equal to the credibility threshold. The empirical evaluation on 62 UCI datasets showed that the proposed ensemble selection method is significantly better than the 8 benchmark algorithms we compared with, which include two well-known homogeneous ensemble methods (RotBoost and Random Subspace), Decision Template, three DES methods (META-DES, KNORA Union and KNORA Eliminate), and two static ensemble selection methods based on evolutionary algorithms (ACO and GA Meta-data). Clearly, our method provides a flexible ensemble selection mechanism in which a specific EoC is selected for each test sample based on the credibility of each base classifier. Moreover, based on the credibility threshold, we can observe the behavior of a base classifier on a dataset, where some base classifiers are always considered and some would never be considered in the aggregation. In this study, we used the original ABC algorithm to search for the optimal confidence threshold for each classifier. This algorithm has some benefits in practice such as having a small number of control parameters and providing simple implementation. In term of algorithm characteristic, although the original ABC algorithm shows powerful ability to investigate the various unknown regions in the solution space to find the global optimum (exploration), it is weak in applying knowledge of previous good solutions to search for the better one (exploitation). This makes the original ABC algorithm slow to converge, resulting in high computation cost. Experimental studies showed that although the proposed method is better than the benchmark algorithm in term of classification accuracy, the running time of the proposed method is significantly higher than other ensemble selection methods like ACO and GA Meta-data on some datasets. One of our future works is to apply different EAs or parallelizing the proposed method to resolve the issue of high computation cost.

We proposed using entropy in this work to measure the confidence in the prediction of each classifier with the note that confidence is inversely proportional to entropy. Although entropy is an effective way to measure uncertainty, it has a weakness when applied to predictions in some cases. For instance, entropy works well with two prediction vectors for a 3-class classification problem such as (0.5, 0.4, 0.1) and (0.6, 0.3, 0.1) as the second vector has a smaller entropy value, and therefore, higher confidence than the first one. However, when comparing two prediction vectors (0.5, 0.39501, 0.10499) and (0.6, 0.1944, 0.2056), the second vector clearly gives us more confidence than the first one when making decision that the sample belongs to the first class. The entropy, however, does not reflect this as the entropy of the two vectors are equal (1.37072). This is the motivation for us to look for alternatives in the future to quantify confidence in the base classifiers' predictions.

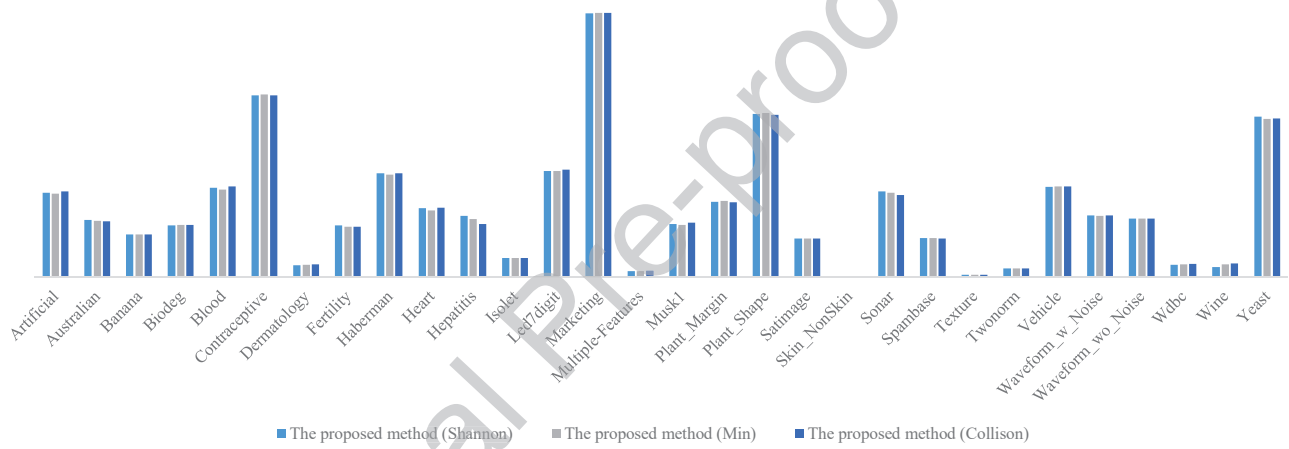


Figure 2. Classification error rates of the proposed method using 3 different entropies

Table 2. Mean and variance of classification error of five ensemble methods (using 3 base classifiers)

	GA Meta-data		ACO		Decision Template		Random Subspace		RotBoost	
	Mean	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*
Abalone	0.4736	5.43E-04	0.4720	7.98E-04	0.4787	1.47E-02	0.4679	6.52E-04	**0.4585**	6.98E-04
Appendicitis	0.1600	1.04E-02	0.1827	1.51E-02	0.1245	6.51E-02	0.1385	7.41E-03	0.1358	8.84E-03
Artificial	0.2295	2.39E-03	0.2257	2.41E-03	0.2486	3.91E-02	0.2771	1.99E-03	0.2976	2.01E-03

AssetNegotiation-F2	0.0850	9.76E-03	0.1558	2.56E-02	0.0996	5.83E-07	0.0701	1.01E-05	**0.0511**	1.87E-07
AssetNegotiation-F3	0.0535	1.23E-05	0.1219	1.07E-02	0.0881	3.58E-07	0.0737	2.99E-05	**0.0519**	1.67E-07
AssetNegotiation-F4	0.1015	9.79E-03	0.2295	3.78E-02	0.0697	2.35E-07	0.0871	1.62E-05	**0.0528**	3.06E-07
Australian	0.1807	1.30E-03	0.1816	2.38E-03	0.1415	3.82E-02	0.1522	1.09E-03	0.1749	2.55E-03
Banana	0.1116	1.23E-04	0.1129	2.32E-04	0.1108	1.19E-02	0.3746	2.20E-03	**0.1050**	1.33E-04
Biodeg	0.1836	1.22E-03	0.1800	1.21E-03	0.1435	3.81E-02	0.1352	6.30E-04	0.1703	1.26E-03
Blood	0.2344	6.75E-04	0.2820	2.86E-03	0.2714	6.00E-02	0.2228	7.17E-04	0.2317	1.25E-03
BNG-Bridges	0.2968	4.50E-06	0.2950	9.08E-06	0.3180	2.02E-06	0.3748	3.10E-05	**0.2352**	5.16E-06
BNG-Zoo	0.0560	3.61E-07	0.0557	1.64E-06	0.0709	5.75E-07	0.1321	9.99E-06	**0.0530**	5.59E-06
Breast-Tissue	0.3400	9.36E-03	0.4033	8.70E-03	0.3882	1.34E-01	**0.3267**	8.48E-03	0.3655	1.49E-02
Bupa	0.3804	8.55E-03	0.3548	5.43E-03	0.3331	5.72E-02	0.3420	5.80E-03	0.3384	7.62E-03
Cleveland	0.4433	4.06E-03	0.4643	6.10E-03	0.4227	7.43E-02	0.4184	1.45E-03	0.4217	3.30E-03
CNS	0.4333	1.96E-02	0.4111	1.80E-02	0.4000	3.44E-02	0.3778	9.14E-03	0.3722	8.67E-03
Colon	0.2087	2.46E-02	0.2079	2.38E-02	**0.1437**	1.52E-02	0.1571	1.71E-02	0.1619	1.76E-02
Contraceptive	0.5237	1.30E-03	0.5028	1.95E-03	**0.4505**	4.09E-02	0.4777	6.12E-04	0.4621	1.17E-03
Dermatology	0.0383	1.03E-03	0.0337	8.15E-04	0.0448	2.30E-02	**0.0261**	4.18E-04	0.0429	6.19E-04
DowJones-1985-2003	**0.0000**	0.00E+00	**0.0000**	0.00E+00	2.3498E-03	1.75E-07	6.9482E-04	3.16E-07	9.6503E-06	6.05E-10
Duke	0.1300	2.69E-02	**0.1083**	2.47E-02	0.1250	2.50E-02	0.1400	2.44E-02	0.1800	3.49E-02
Electricity	0.1911	3.15E-05	0.1911	3.15E-05	0.1959	4.26E-05	**0.1166**	6.79E-05	0.1914	4.96E-05
Fertility	0.1900	1.29E-02	0.1467	3.16E-03	0.3733	1.86E-01	**0.1200**	1.60E-03	**0.1200**	1.60E-03
Haberman	0.2964	2.26E-03	0.2984	1.72E-03	0.3558	6.25E-02	0.2961	3.00E-03	0.2800	2.77E-03
Hayes-Roth	0.2917	9.20E-03	**0.2708**	1.31E-02	0.3146	1.20E-01	0.3458	1.95E-02	0.3938	8.11E-03
Heart	0.2395	6.93E-03	0.2185	8.26E-03	**0.1679**	5.03E-02	0.1877	5.85E-03	0.1753	2.83E-03
Hepatitis	0.1750	1.42E-02	0.2083	9.72E-03	0.1667	1.15E-01	**0.1333**	5.14E-03	0.1500	6.67E-03
Hyperplane	**9.5400E-04**	1.40E-08	9.7500E-04	3.76E-08	0.0276	3.05E-07	0.1321	1.77E-05	0.0486	6.37E-06
Iris	0.0333	1.70E-03	0.0400	1.96E-03	**0.0267**	3.32E-02	0.0511	2.87E-03	0.0378	1.68E-03
Isolet	0.0623	6.53E-05	0.0650	6.63E-05	0.0515	7.71E-03	0.0588	6.12E-05	0.1001	9.06E-05
Led7digit	0.2973	4.50E-03	0.3013	6.05E-03	0.2700	6.58E-02	0.4460	2.99E-03	0.2753	5.70E-03
Letter	0.0610	5.80E-05	**0.0506**	3.20E-05	0.0782	7.02E-03	0.1011	7.08E-05	0.1239	9.84E-05
Leukemia	0.0375	5.24E-03	0.0595	7.06E-03	0.0423	5.54E-03	**0.0286**	4.63E-03	0.0750	7.66E-03
Magic	0.1920	1.37E-04	0.1902	4.75E-05	0.1907	9.80E-03	0.1730	4.67E-05	**0.1562**	5.47E-05
Mammographic	0.2032	1.97E-03	0.2169	1.76E-03	0.1908	2.83E-02	**0.1639**	1.98E-03	0.1839	2.12E-03
Marketing	0.7322	4.03E-04	0.7323	3.85E-04	0.6709	1.46E-02	0.6716	1.56E-04	**0.6622**	2.12E-04
Multiple-Features	0.0150	1.00E-04	**0.0125**	5.29E-05	0.0130	8.26E-03	0.0182	6.91E-05	0.0448	2.46E-04
Musk	0.1344	1.61E-03	0.1245	1.77E-03	0.1359	3.38E-02	**0.0805**	1.26E-03	0.2477	3.53E-03
Page-Blocks	0.0420	4.35E-05	0.0462	7.31E-05	0.0540	8.94E-03	**0.0319**	3.02E-05	0.0546	1.49E+01
Phoneme	**0.1149**	2.11E-04	**0.1149**	2.11E-04	0.1519	1.47E-02	0.1627	3.36E-04	0.1822	1.99E-04
Pima	0.3056	2.34E-03	0.3078	2.35E-03	0.2504	4.11E-02	0.2570	2.28E-03	0.2487	2.04E-03
Plant-Margin	0.2708	6.75E-04	0.2829	8.96E-04	0.1885	1.22E-02	**0.1721**	5.98E-04	0.4910	1.40E-03
Plant-Shape	0.4669	6.80E-04	0.4592	6.11E-04	0.4104	2.45E-02	**0.3804**	9.81E-04	0.5042	1.41E-03
Poker	0.3629	1.51E-05	**0.3621**	4.84E-06	0.7380	8.90E-06	0.4988	9.76E-12	0.3682	3.85E-06
RandomTree	0.1050	7.88E-07	0.1050	7.88E-07	0.1050	7.88E-07	0.2701	1.29E-04	**0.0800**	2.55E-05
Satimage	0.1064	1.26E-04	0.0932	1.19E-04	0.1257	8.97E-05	**0.0886**	9.68E-05	0.1303	1.44E-04
Skin-NonSkin	**4.3119E-04**	1.41E-08	4.3390E-04	1.28E-08	0.0264	9.04E-04	2.6171E-03	7.51E-08	6.0843E-03	4.12E-06
Sonar	0.2583	8.89E-03	0.2368	5.91E-03	0.2002	5.69E-02	**0.1457**	6.05E-03	0.2325	7.80E-03

Spambase	0.1185	1.95E-04	0.1224	2.96E-04	0.0964	1.28E-02	0.0960	1.83E-04	**0.0762**	1.78E-04
Svmguide	0.2393	3.69E-03	0.2487	5.97E-03	**0.1660**	5.03E-03	0.2661	3.53E-03	0.2847	2.53E-03
Tae	0.5453	1.35E-02	0.5129	1.30E-02	**0.4335**	1.45E-01	0.5035	9.64E-03	0.5150	1.72E-02
Texture	0.0050	7.64E-06	0.0051	4.94E-06	0.0078	1.02E-05	0.0238	3.72E-05	0.0404	1.10E-04
Twonorm	0.0330	3.61E-05	0.0310	3.45E-05	**0.0193**	3.94E-03	0.0258	1.65E-05	0.0327	4.11E-05
Vehicle	0.2627	1.90E-03	0.2597	1.44E-03	**0.2128**	2.26E-02	0.2600	1.30E-03	0.3180	2.31E-03
Vertebral	0.1893	3.38E-03	**0.1527**	3.46E-03	0.1978	5.20E-02	0.2893	3.22E-03	0.1785	4.49E-03
Waveform-w-Noise	0.1787	2.04E-04	0.1770	2.22E-04	0.1638	1.58E-02	0.1755	2.94E-04	0.1625	4.66E-04
Waveform-wo-Noise	0.1738	4.45E-04	0.1705	2.75E-04	0.1479	1.71E-02	0.1498	3.66E-04	0.1559	2.80E-04
Wdbc	0.0352	6.19E-04	0.0457	8.53E-04	0.0369	1.68E-02	0.0381	3.51E-04	0.0545	8.48E-04
Wine	0.0264	1.00E-03	0.0265	1.26E-03	0.0377	5.80E-02	**0.0170**	6.75E-04	0.0320	1.43E-03
Wine-Red	0.4653	2.14E-03	0.4690	1.05E-03	0.5107	3.73E-02	**0.3110**	5.71E-04	0.3986	9.58E-04
Wine-White	0.4798	4.58E-04	0.4947	6.21E-04	0.5947	1.40E-02	**0.3252**	3.23E-04	0.4335	2.61E-04
Yeast	0.4917	1.58E-03	0.4861	1.99E-03	0.4154	5.12E-02	0.4975	1.31E-03	0.4290	1.27E-03

The bold value indicates the best result on each dataset

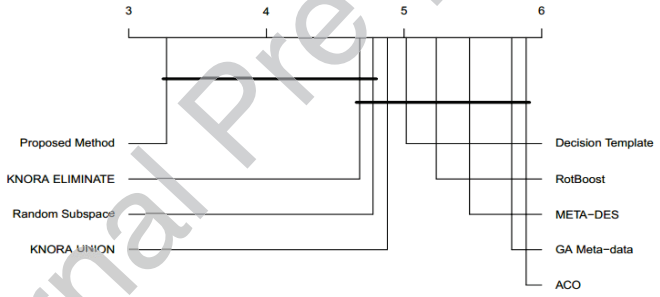


Figure 3. The result of Nemenyi test (using 3 base classifiers)

Table 3. Mean and variance of classification error of dynamic ensemble selection methods and the proposed method (using 3 base classifiers)

	META-DES		KNORA ELIMINATE		KNORA UNION		Proposed Method	
	Mean	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*

Abalone	0.4740	5.92E-04	0.4678	4.90E-04	0.4707	3.88E-04	0.4719	4.64E-04
Appendicitis	0.1230	8.59E-03	0.1291	1.14E-02	**0.1133**	8.38E-03	0.1539	8.96E-03
Artificial	0.2362	2.15E-03	0.2257	1.98E-03	0.2171	1.23E-03	**0.2124**	1.04E-03
AssetNegotiation-F2	0.0553	2.42E-07	0.0558	2.98E-07	0.0715	9.12E-07	0.0594	2.63E-07
AssetNegotiation-F3	0.0560	1.51E-07	0.0560	1.62E-07	0.0655	1.64E-07	0.0584	1.45E-07
AssetNegotiation-F4	0.0553	1.91E-07	0.0603	1.22E-06	0.0739	3.10E-07	0.0625	2.32E-07
Australian	0.1614	1.66E-03	0.1589	1.52E-03	**0.1357**	1.06E-03	0.1444	1.25E-03
Banana	0.1125	9.24E-05	0.1157	1.62E-04	0.1079	1.51E-04	0.1069	1.07E-04
Biodeg	0.1428	8.70E-04	0.1479	6.67E-04	0.1479	7.16E-04	**0.1302**	8.32E-04
Blood	0.2321	1.12E-03	0.2286	1.32E-03	**0.2205**	8.87E-04	0.2259	1.38E-03
BNG-Bridges	0.2791	1.88E-06	0.3019	1.40E-06	0.3115	1.19E-06	0.2904	8.90E-07
BNG-Zoo	0.0538	4.22E-07	0.0586	5.27E-07	0.0695	4.18E-07	0.0571	2.3662E-07
Breast-Tissue	0.3909	1.50E-02	0.3430	1.37E-02	0.3706	1.46E-02	0.3833	1.28E-02
Bupa	0.3545	5.83E-03	0.3469	2.47E-03	0.3373	3.13E-03	**0.2948**	2.32E-03
Cleveland	0.4226	5.63E-03	0.4162	3.06E-03	**0.4038**	3.37E-03	0.4308	5.06E-03
CNS	0.3722	2.53E-02	**0.3500**	3.03E-02	0.3611	3.16E-02	0.3611	2.05E-02
Colon	0.2111	1.85E-02	0.2071	2.83E-02	0.1968	3.32E-02	0.1706	2.18E-02
Contraceptive	0.4719	1.03E-03	0.4639	2.01E-03	0.4574	1.16E-03	0.4587	1.29E-03
Dermatology	0.0475	7.92E-04	0.0382	1.07E-03	0.0307	4.35E-04	0.0289	7.14E-04
DowJones-1985-2003	0.1901	1.66E-05	7.0447E-04	4.12E-08	5.4741E-03	2.60E-07	**0.0000**	**0.00E+00**
Duke	0.2483	5.06E-02	0.1433	2.65E-02	0.1967	3.43E-02	0.1167	2.49E-02
Electricity	0.2095	2.92E-05	0.2024	2.42E-05	0.2192	4.93E-05	0.1896	2.44E-05
Fertility	0.1467	3.16E-03	0.1367	2.99E-03	0.1333	2.89E-03	0.1300	2.77E-03
Haberman	0.2703	2.89E-03	0.2778	1.82E-03	0.2767	1.80E-03	**0.2615**	1.45E-03
Hayes-Roth	0.3292	1.53E-02	0.3375	1.14E-02	0.3417	1.56E-02	0.3146	1.11E-02
Heart	0.1827	2.46E-03	0.1938	5.18E-03	0.1753	3.65E-03	0.1741	3.12E-03
Hepatitis	0.1833	1.43E-02	0.1458	7.38E-03	0.1458	9.46E-03	0.1542	1.01E-02
Hyperplane	2.0260E-03	5.80E-08	0.0129	4.42E-08	0.0282	1.87E-07	3.8620E-03	2.46E-08
Iris	0.0311	1.40E-03	0.0378	1.68E-03	0.0356	1.40E-03	0.0378	1.68E-03
Isolet	0.0786	1.14E-04	0.0618	7.51E-05	0.0700	1.04E-04	**0.0481**	6.69E-05
Led7digit	0.2987	3.97E-03	0.2680	4.82E-03	**0.2653**	3.86E-03	0.2673	4.25E-03
Letter	0.1054	5.34E-05	0.0617	3.36E-05	0.1089	5.00E-05	0.0681	4.40E-05
Leukemia	0.0655	7.32E-03	0.0554	4.62E-03	0.0696	7.60E-03	**0.0286**	3.27E-03
Magic	0.1969	4.96E-05	0.1934	5.56E-05	0.1933	4.80E-05	0.1894	3.94E-05
Mammographic	0.1872	1.87E-03	0.1855	1.90E-03	0.1851	1.58E-03	0.1908	1.61E-03
Marketing	0.6799	1.44E-04	0.6791	2.33E-04	0.6700	2.29E-04	0.6674	1.68E-04
Multiple-Features	0.0215	1.45E-04	0.0128	8.11E-05	0.0150	6.50E-05	0.0145	6.89E-05
Musk	0.1484	1.30E-03	0.1708	2.28E-03	0.1695	3.87E-03	0.1337	1.53E-03
Page-Blocks	0.0423	5.97E-05	0.0424	5.44E-05	0.0503	5.14E-05	0.0447	5.97E-05
Phoneme	0.1464	3.03E-04	0.1337	1.77E-04	0.1796	3.30E-04	0.1400	3.09E-04
Pima	0.2448	2.15E-03	**0.2366**	2.31E-03	0.2427	2.84E-03	0.2531	2.61E-03
Plant-Margin	0.3242	9.37E-04	0.2108	5.18E-04	0.2060	6.42E-04	0.1896	4.25E-04
Plant-Shape	0.4660	1.12E-03	0.4477	6.24E-04	0.4354	7.53E-04	0.4125	1.07E-03
Poker	0.4304	1.89E-06	0.3938	2.53E-05	0.3907	2.64E-06	0.3650	2.88E-06
RandomTree	0.1229	4.86E-07	0.1178	8.53E-07	0.1247	4.61E-07	0.1119	9.47E-07

Satimage	0.1036	1.34E-04	0.0996	1.19E-04	0.1281	1.12E-04	0.0970	1.53E-04
Skin-NonSkin	1.6459E-03	9.43E-08	5.1145E-04	1.64E-08	9.0047E-04	4.54E-08	5.2777E-04	1.72E-08
Sonar	0.2533	6.32E-03	0.2375	8.11E-03	0.2437	5.70E-03	0.2161	7.91E-03
Spambase	0.0961	1.60E-04	0.1072	1.23E-04	0.0977	9.32E-05	0.0975	1.26E-04
Svmguide	0.1976	5.39E-03	0.1788	4.29E-03	0.1729	4.78E-03	0.1763	3.36E-03
Tae	0.5118	1.60E-02	0.4863	1.32E-02	0.4925	1.57E-02	0.4903	1.18E-02
Texture	0.0128	2.03E-05	0.0058	4.28E-06	0.0128	1.97E-05	**0.0048**	5.62E-06
Twonorm	0.0217	2.36E-05	0.0222	2.19E-05	0.0217	2.14E-05	0.0214	1.64E-05
Vehicle	0.2774	1.78E-03	0.2651	2.13E-03	0.2569	1.11E-03	0.2282	1.15E-03
Vertebral	0.1839	4.31E-03	0.1753	4.21E-03	0.1968	4.39E-03	0.1860	3.52E-03
Waveform-w-Noise	**0.1532**	1.46E-04	0.1647	2.81E-04	0.1692	1.79E-04	0.1549	2.14E-04
Waveform-wo-Noise	**0.1463**	2.67E-04	0.1569	2.79E-04	0.1653	2.93E-04	0.1475	3.57E-04
Wdbc	0.0416	4.21E-04	0.0475	9.50E-04	0.0399	3.09E-04	**0.0305**	4.51E-04
Wine	0.0224	1.17E-03	0.0339	2.26E-03	0.0282	1.41E-03	0.0245	1.22E-03
Wine-Red	0.4324	7.56E-04	0.4180	8.78E-04	0.4234	1.19E-03	0.4157	1.10E-03
Wine-White	0.4584	3.09E-04	0.4502	4.67E-04	0.4682	3.06E-04	0.4539	3.21E-04
Yeast	0.4277	1.09E-03	0.4077	1.69E-03	**0.3994**	1.50E-03	0.4050	9.56E-04

The bold value indicates the best result on each dataset

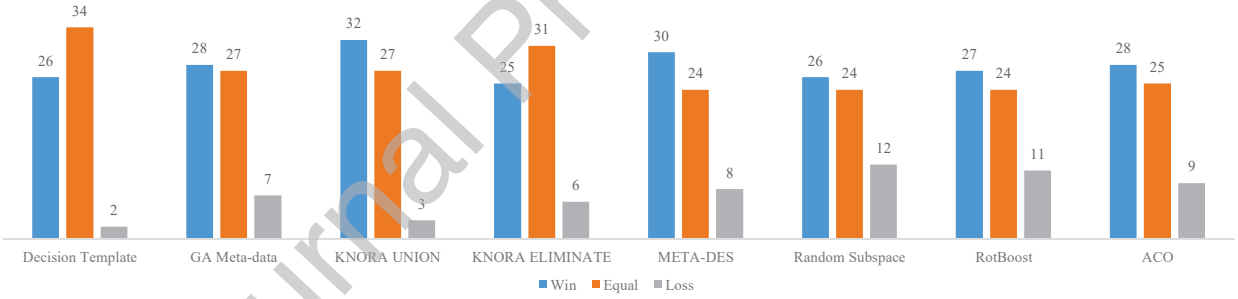


Figure 4. The result of Wilcoxon signed rank test (using 3 base classifiers)

Table 4. Mean and variance of classification error of three ensemble methods (using 7 base classifiers)

	GA Meta-data		ACO		Decision Template		META DES		KNORA ELIMINATE		KNORA UNION		Proposed Method	
	Mean	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*
Abalone*	0.4986	6.84E-04	0.4888	9.25E-04	0.4826	1.54E-02	0.4652	7.22E-04	0.4812	6.90E-04	0.4652	4.42E-04	0.4648	3.45E-04
Appendicitis	0.1767	9.44E-03	0.1630	1.18E-02	**0.1212**	6.89E-02	0.1224	8.28E-03	0.1424	8.98E-03	0.1261	9.77E-03	0.1452	7.40E-03
Artificial	0.2667	3.80E-03	0.2229	2.14E-03	0.2462	4.80E-02	**0.2171**	2.58E-03	0.2538	1.89E-03	0.2210	1.70E-03	0.2233	1.71E-03
AssetNegotiation-F2*	**0.0511**	1.87E-07	0.2283	3.29E-02	0.0779	7.75E-06	0.0519	2.47E-07	0.0657	3.56E-05	0.0632	2.17E-06	**0.0511**	1.87E-07
AssetNegotiation-F3	0.0530	1.36E-05	0.1808	3.59E-02	0.0679	7.12E-06	0.0531	1.48E-07	0.0630	2.09E-05	0.0594	2.40E-06	**0.0518**	1.73E-07
AssetNegotiation-F4*	0.0536	5.46E-06	0.2691	3.60E-02	0.0537	3.42E-07	0.0540	7.98E-07	0.0796	8.83E-05	0.0629	1.27E-06	**0.0528**	3.06E-07
Australian	0.1845	1.97E-03	0.1908	2.62E-03	**0.1324**	3.78E-02	0.1705	1.54E-03	0.1826	2.30E-03	0.1541	1.63E-03	0.1444	1.32E-03
Banana	0.1331	3.65E-04	0.1279	3.51E-04	0.1028	1.34E-02	0.1088	1.04E-04	0.1184	2.10E-04	0.1029	8.77E-05	**0.0997**	8.30E-05
Biodeg	0.1773	1.19E-03	0.1839	1.16E-03	**0.1292**	3.55E-02	0.1419	7.45E-04	0.1823	1.86E-03	0.1422	8.31E-04	0.1311	5.94E-04
Blood	0.2861	3.18E-03	0.2643	1.46E-03	0.2589	5.12E-02	0.2437	1.53E-03	0.2664	2.52E-03	0.2254	1.35E-03	**0.2125**	1.15E-03
BNG-Bridges*	0.3044	1.19E-05	0.2865	3.02E-04	0.2785	2.86E-06	0.2880	1.79E-06	0.2903	3.14E-06	0.2842	3.10E-05	0.2506670	1.57648E-06
BNG-Zoo	0.0572	3.83E-06	0.0554	1.85E-06	0.0566	3.74E-07	0.0517	4.09E-07	0.0577	6.88E-07	0.0571	7.01E-07	**0.0472**	3.86E-07
Breast-Tissue	0.3736	1.81E-02	0.3915	1.65E-02	0.3515	9.92E-02	0.4094	1.71E-02	0.4409	2.05E-02	0.3636	7.66E-03	**0.3249**	8.37E-03
Bupa	0.3585	5.38E-03	0.3606	8.10E-03	0.2994	5.57E-02	0.3544	6.85E-03	0.3446	6.94E-03	**0.2955**	3.57E-03	0.2995	4.36E-03
Cleveland	0.4615	5.21E-03	0.4631	6.00E-03	**0.4150**	7.20E-02	0.4276	4.47E-03	0.4847	7.82E-03	0.4251	3.20E-03	0.4429	3.95E-03
CNS	0.4833	2.84E-02	0.4167	1.81E-02	0.3833	2.62E-02	0.3611	2.79E-02	0.3833	3.55E-02	0.4000	2.89E-02	**0.3556**	2.91E-02
Colon	0.2206	1.66E-02	0.1929	2.30E-02	**0.1238**	1.49E-02	0.1849	2.66E-02	0.2587	3.04E-02	0.1937	3.30E-02	0.1619	1.76E-02
Contraceptive	0.5230	1.59E-03	0.5130	2.00E-03	**0.4337**	3.10E-02	0.4804	1.65E-03	0.4795	1.69E-03	0.4415	1.64E-03	0.4370	1.84E-03
Dermatology	0.0354	7.77E-04	0.0401	9.18E-04	0.0411	2.94E-02	0.0225	6.56E-04	0.0829	3.81E-03	**0.0224**	8.67E-04	0.0262	6.81E-04
DowJones-1985-2003	**0.0000**	0.00E+00	9.6500E-05	1.72E-08	0.0005	3.77E-08	0.0264	6.64E-05	6.9965E-05	3.31E-09	2.0989E-04	2.88E-08	1.4475E-05	8.38E-10
Duke	0.2317	3.42E-02	**0.1350**	2.35E-02	0.1400	2.86E-02	0.1917	3.82E-02	0.2600	3.37E-02	0.2150	3.95E-02	0.1467	3.07E-02
Electricity	0.1160	1.53E-04	0.1150	3.31E-05	0.1324	2.66E-05	0.1401	2.11E-05	0.1543	8.62E-05	0.1558	1.13E-04	**0.1051**	1.75E-05
Fertility*,**	0.1967	1.50E-02	0.1833	1.54E-02	0.3533	1.72E-01	0.1333	3.56E-03	0.1733	9.29E-03	0.1267	3.96E-03	0.1533	3.82E-03
Haberman*	0.3474	4.33E-03	0.3312	4.55E-03	0.3413	8.31E-02	0.2878	2.90E-03	0.2962	2.09E-03	0.2846	1.90E-03	0.2823	2.36E-03
Hayes-Roth	0.1833	9.10E-03	**0.1625**	1.27E-02	0.2188	1.12E-01	0.2521	1.26E-02	0.2229	1.06E-02	0.2063	1.20E-02	0.2083	1.15E-02
Heart	0.2370	6.46E-03	0.2185	3.78E-03	**0.1679**	6.58E-02	0.2037	6.56E-03	0.2543	7.02E-03	0.1889	4.60E-03	0.1852	4.57E-03
Hepatitis**	0.2125	2.62E-02	0.1917	8.06E-03	0.1875	1.17E-01	0.1625	1.16E-02	0.1625	1.58E-02	0.1417	1.33E-02	0.1542	1.32E-02
Hyperplane	6.6000E-05	9.44E-10	2.1900E-04	8.75E-08	5.2910E-03	1.24E-05	9.6100E-04	3.75E-08	1.0540E-03	1.24E-08	2.7740E-03	2.88E-07	**4.4000E-05**	1.06E-09
Iris	**0.0311**	1.70E-03	0.0444	2.47E-03	0.0578	8.88E-02	0.0400	1.96E-03	0.0489	1.46E-03	0.0400	1.96E-03	0.0356	1.11E-03
Isolet	0.0597	2.32E-04	0.0640	1.16E-04	0.0435	8.63E-03	0.0596	5.53E-05	0.1174	2.50E-04	0.0608	7.28E-05	**0.0433**	5.97E-05
Led7digit	0.3293	4.69E-03	0.3013	5.20E-03	0.2700	6.30E-02	0.3120	5.46E-03	0.2947	5.09E-03	**0.2673**	4.65E-03	0.2773	4.37E-03
Letter	0.0763	2.14E-04	0.0540	3.94E-04	0.0783	6.83E-03	0.1219	9.41E-05	0.0718	3.94E-05	0.1022	4.75E-05	**0.0530**	3.45E-05
Leukemia	0.0417	4.07E-03	0.0554	4.62E-03	**0.0190**	2.36E-03	0.0375	3.88E-03	0.0423	4.18E-03	0.0333	5.01E-03	0.0238	2.83E-03
Magic*	0.2042	1.89E-04	0.1735	4.94E-05	0.1676	6.90E-03	0.1838	5.97E-05	0.1805	5.87E-05	0.1726	5.77E-05	0.1570	3.62E-05
Mammographic**	0.2237	2.55E-03	0.2092	2.35E-03	0.1956	2.32E-02	0.2072	1.75E-03	0.2149	1.96E-03	0.1851	1.20E-03	0.1779	1.49E-03
Marketing*	0.7341	2.62E-04	0.7337	2.67E-04	0.6701	7.87E-03	0.6771	1.50E-04	0.6723	2.04E-04	0.6675	2.51E-04	0.6625	1.42E-04
Multiple-Features	0.0137	6.99E-05	0.0142	8.35E-05	**0.0107**	6.53E-03	0.0182	8.91E-05	0.0190	8.40E-05	0.0185	6.86E-05	0.0117	5.22E-05
Musk	0.0889	1.73E-03	0.1002	2.69E-03	**0.0623**	3.58E-02	0.1491	2.42E-03	0.1274	3.13E-03	0.1534	3.23E-03	0.0672	6.31E-04
Page-Blocks**	0.0355	5.84E-05	0.0370	4.09E-05	0.0384	8.82E-03	0.0390	3.91E-05	0.0416	3.68E-05	0.0478	4.81E-05	0.0358	2.78E-05
Phoneme	0.1271	4.04E-04	0.1172	2.39E-04	0.1395	1.42E-02	0.1322	1.81E-04	0.1297	1.85E-04	0.1462	3.54E-04	**0.1140**	1.14E-04
Pima	0.3047	3.03E-03	0.3077	2.45E-03	0.2491	5.01E-02	0.2614	1.62E-03	0.2891	1.88E-03	0.2435	2.57E-03	**0.2422**	2.11E-03
Plant-Margin**	0.3119	1.16E-03	0.2796	1.08E-03	0.1925	1.79E-02	0.2821	9.29E-04	0.2246	7.34E-04	0.2056	8.19E-04	0.1873	6.63E-04
Plant-Shape**	0.5225	1.40E-03	0.4598	1.30E-03	0.4404	1.88E-02	0.4656	1.01E-03	0.4563	9.01E-04	0.4604	8.08E-04	0.4115	1.15E-03
Poker	0.3751	5.58E-03	0.3164	8.69E-05	0.5017	3.99E-03	0.3928	4.18E-05	0.3954	8.90E-05	0.3353	9.44E-05	**0.3136**	3.23E-05
RandomTree	0.0052	3.49E-06	0.0039	1.82E-07	0.0238	2.44E-06	0.0124	8.83E-07	0.0138	5.40E-07	0.0531	3.14E-05	**0.0038**	1.92E-07
Satimage**	0.1125	1.51E-04	0.0998	3.18E-04	0.1212	8.76E-05	0.1059	1.24E-04	0.1072	1.15E-04	0.1271	1.39E-04	0.0890	7.42E-05
Skin-NonSkin	6.4203E-04	2.35E-07	5.7540E-04	6.44E-08	6.0000E-04	1.17E-04	9.3176E-04	5.66E-08	4.8832E-04	1.67E-08	6.4339E-04	1.84E-08	**4.2847E-04**	2.05E-08

Sonar	0.2114	7.56E-03	0.2391	6.54E-03	**0.1375**	7.16E-02	0.1999	7.85E-03	0.1949	1.30E-02	0.2160	7.41E-03	0.1681	5.02E-03
Spambase	0.0805	1.43E-04	0.0826	1.41E-04	**0.0624**	1.06E-02	0.0749	1.15E-04	0.1222	3.22E-03	0.0845	3.01E-04	0.0734	1.00E-04
Svmguide	0.2471	7.04E-03	0.2702	2.20E-03	**0.1712**	4.56E-03	0.2070	5.24E-03	0.2063	4.09E-03	0.1721	3.43E-03	0.1856	5.79E-03
Tae	0.4989	1.42E-02	0.5581	1.43E-02	**0.3540**	1.33E-01	0.4944	1.55E-02	0.4614	1.10E-02	0.4881	1.93E-02	0.4485	1.40E-02
Texture	0.0050	7.20E-06	0.0059	1.29E-05	0.0081	1.29E-05	0.0088	1.65E-05	0.0056	7.92E-06	0.0210	3.39E-05	**0.0035**	4.48E-06
Twonorm	0.0321	4.00E-05	0.0321	2.96E-05	**0.0192**	3.67E-03	0.0223	1.70E-05	0.0287	3.48E-05	0.0228	2.31E-05	0.0218	1.51E-05
Vehicle	0.2514	2.27E-03	0.2692	1.68E-03	0.2309	3.17E-02	0.2553	1.07E-03	0.2972	1.50E-03	0.2872	1.23E-03	**0.2278**	9.34E-04
Vertebral	0.2022	3.88E-03	0.1828	4.81E-03	**0.1516**	5.23E-02	0.1613	3.47E-03	0.1785	4.84E-03	0.1742	3.44E-03	0.1634	3.81E-03
Waveform-w-Noise	0.1755	2.45E-04	0.1725	1.94E-04	0.1535	1.53E-02	0.1534	2.95E-04	0.1979	7.42E-04	0.1641	1.51E-04	**0.1431**	1.64E-04
Waveform-wo-Noise	0.1773	3.82E-04	0.1739	2.44E-04	0.1423	1.87E-02	0.1391	2.76E-04	0.1783	5.67E-04	0.1597	3.92E-04	**0.1313**	2.86E-04
Wdbc	0.0392	5.48E-04	0.0369	6.24E-04	**0.0322**	1.60E-02	0.0369	3.80E-04	0.0545	8.89E-04	0.0393	4.87E-04	0.0393	5.28E-04
Wine**	0.0225	9.69E-04	0.0225	1.20E-03	0.0582	6.03E-02	0.0358	1.30E-03	0.0341	1.91E-03	0.0415	1.91E-03	0.0395	1.73E-03
Wine-Red**	0.4534	1.86E-03	0.4332	2.49E-03	0.4980	3.62E-02	0.4107	1.06E-03	0.4263	1.84E-03	0.3900	8.16E-04	0.3677	1.15E-03
Wine-White**	0.4879	7.54E-04	0.4578	7.22E-04	0.5208	2.39E-02	0.4240	5.86E-04	0.4610	8.94E-04	0.4226	5.52E-04	0.3800	2.90E-04
Yeast	0.4919	1.11E-03	0.4798	1.91E-03	0.4057	3.95E-02	0.4405	1.00E-03	0.4171	8.63E-04	0.3992	1.05E-03	**0.3906**	7.75E-04

*We do not show the results of Random Subspace and RotBoost method since the results are similar to those in Table 2; The bold value indicates the best result on each dataset; * means that the best results belong to RotBoost; ** means that the best results belong to Random Subspace.*

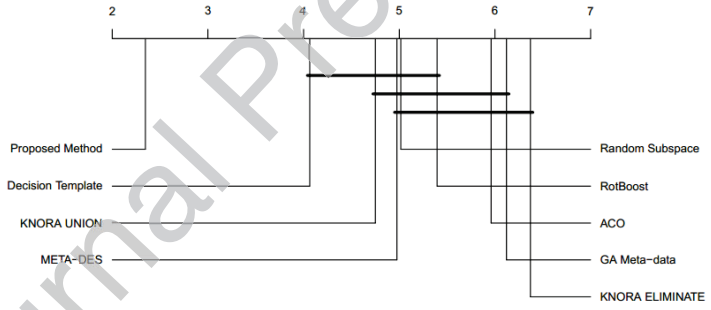


Figure 5. The result of Nemenyi test (using 7 base classifiers)

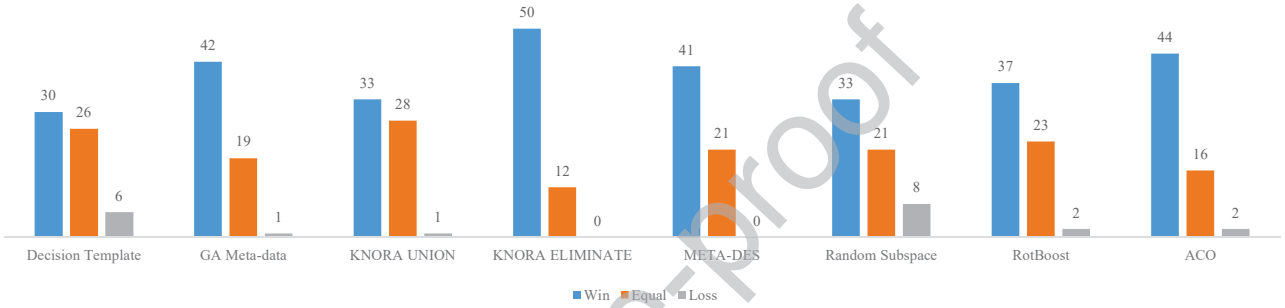


Figure 6. The result of Wilcoxon signed rank test (using 7 base classifiers)

Table 5. The running time (in seconds) of three evolutionary algorithm-based methods

	GA Meta-data	ACO	Proposed Method
AssetNegotiation-F2	547.28	110.15	37118.12
CNS	2.95	3.24	7.86
Haberman	1.81	1.95	12.47
Isolet	3518.63	18.71	369.84
Lekemia	3.02	3.28	8.51
Multiple-Features	44.90	4.27	90.28
Plant-Margin	22258.03	42.75	148.21
Plant-Shape	10503.35	29.87	149.01
Poker	124922.90	706.86	41316.05
Wine	4.80	1.97	8.16

Table 6. The credibility threshold for the three base classifiers and the maximum entropy value in 30 datasets

	LDA	Naïve Bayes	*k*NN5	Max of Entropy
Artificial	0.9892	1.0000	1.0000	1.0000
Australian	0.8873	0.9338	1.0000	1.0000
Banana	1.0000	0.7718	0.7819	1.0000
Biodeg	0.9981	0.0019	1.0000	1.0000
Blood	0.9051	0.9043	1.0000	1.0000
Contraceptive	1.5849	1.5474	1.1945	1.5850
Dermatology	1.9887	0.3226	2.3712	2.5850
Fertility	0.4680	0.1308	0.3906	1.0000
Haberman	0.5200	0.7677	0.0000	1.0000
Heart	0.8517	0.8780	1.0000	1.0000
Hepatitis	0.7485	1.0000	0.4535	1.0000
Isolet	2.0698	0.0000	2.7977	4.7004
Led7digit	2.8893	2.9946	0.0056	3.3219
Marketing	3.1133	2.6677	2.6893	3.1699
Multiple-Features	0.4624	2.2961	1.3451	3.3219
Musk	1.0000	0.0025	1.0000	1.0000
Plant-Margin	4.3688	0.4286	4.3555	6.6439
Plant-Shape	6.6439	1.3479	3.3827	6.6439
Satimage	2.5850	0.0000	2.4138	2.5850
Skin-NonSkin	0.0000	1.0000	1.0000	1.0000
Sonar	1.0000	0.0000	0.7341	1.0000
Spambase	0.9448	1.0000	1.0000	1.0000
Texture	2.2738	0.0000	3.3219	3.3219
Twonorm	0.9999	0.9999	0.4379	1.0000
Vehicle	2.0000	0.0000	1.9328	2.0000
Waveform-w-Noise	1.5422	0.0346	1.5101	1.5850
Waveform-wo-Noise	1.4994	0.0483	1.5850	1.5850
Wdbc	0.6174	0.9805	0.2017	1.0000
Wine	0.8196	1.0091	0.5298	1.5850
Yeast	2.6793	0.2575	3.3219	3.3219

References

[1] T.T. Nguyen, M.P. Nguyen, X.C. Pham, A.W.C. Liew, W. Pedrycz, Combining heterogeneous classifiers via granular prototypes, Applied Soft Computing. 73 (2018), 795-815.

[2] T.T. Nguyen, M.P. Nguyen, X.C. Pham, A.W.C. Liew, Heterogeneous classifier ensemble with fuzzy rule-based meta-learner, Information Sciences. 422 (2018), 144-160.

[3] L.I. Kuncheva, J.C. Bezdek, R.P.W. Duin, Decision templates for multiple classifier fusion: an experimental comparison, Pattern Recognition. 34 (2001), 299-314.

[4] T.T. Nguyen, X.C. Pham, A.W.C. Liew, W. Pedrycz, Aggregation of classifiers: a justifiable information granularity approach, IEEE Transactions on Cybernetics. 49 (6)(2018), 2168 – 2177.

[5] N. Li, Z.-H. Zhou, Selective ensemble under regularization framework, In Proceedings of the 8th International Workshop Multiple Classifier Systems, pages 293–303, Reykjavik, Iceland, 2009.

[6] A.S. Britto, R. Sabourin, L.E.S. Oliveira: Dynamic selection of classifiers-a comprehensive review, Pattern Recognition. 47(11), 3665–3680 (2014).

[7] H. Guo, H. Liu, R. Li, C. Wu, Y. Guo, M. Xu, Margin & diversity based ordering ensemble pruning, Neurocomputing. 275 (2018), 237-246.

[8] R. Younsi, A. Bagnall, Ensembles of random sphere cover classifiers, Pattern Recognition. 49 (2016), 213-225.

[9] Y. Zhang, G. Cao, B. Wang, X. Li, A novel ensemble method for k-nearest neighbor, Pattern Recognition. 85 (2019), 13-25.

[10] X.C. Pham, M.T. Dang, S.V. Dinh, S. Hoang, T.T. Nguyen, A.W.C. Liew, Learning from Data Stream Based on Random Projection and Hoeffding Tree Classifier, in Proceeding of Digital Image Computing: Techniques and Applications (DICTA), 2017.

[11] E. Santucci, L. Didaci, G. Fumera, F. Roli, A parameter randomization approach for constructing classifier ensembles, Pattern Recognition. 69 (2017), 1-13.

[12] L. Yijing, G. Haixiang, L. Xiao, L. Yanan, L. Jinling, Adapted ensemble classification algorithm based on multiple classifier system and feature selection for classifying multi-class imbalanced data, Knowledge-Based Systems. 94 (2016), 88-104.

[13] Z. Yu, D. Wang, J. You, H.-S. Wong, S. Wu, J. Zhang, G. Han, Progressive subspace ensemble learning, Pattern Recognition. 60 (2016), 692-705.

[14] N.G.-Pedrajas, J. M.-Raedo, C. G.-Osorio, J.J. R.-Diez, Supervised subspace projections for constructing ensemble of classifiers, Information Sciences. 193 (2012), 1-21.

[15] T.T. Nguyen, M.T. Dang, A.W.C. Liew, J.C. Bezdek, A weighted multiple classifier framework based on random projection, Information Sciences. 490 (2019), 36-58.

[16] E.K. Tang, P.N. Suganthan, X. Yao, An analysis of diversity measures, Machine Learning. 65(1) (2006), 247-271.

[17] K. Jackowski, New diversity measure for data stream classification ensembles, Engineering Applications of Artificial Intelligence. 74 (2018), 23-34.

[18] L. Li, Q. Hu, X. Wu, D. Yu, Exploration of classification confidence in ensemble learning, Pattern Recognition. 47 (2014), 3120-3131.

[19] R.M.O. Cruz, R. Sabourin, G.D.C. Cavalcanti, Dynamic classifier selection: Recent advances and perspectives, Information Fusion. 41 (2018), 195-216.

[20] D. D. Margineantu, T. G. Dietterich. Pruning adaptive boosting. In Proceedings of the 14th International Conference on Machine Learning, pages 211–218, 1997.

[21] G. Martınez-Munoz, A. Suarez, Aggregation ordering in bagging, In Proceedings of the International Conference on Artifical Intelligence and Applications, 2004, pages 258–263.

[22] T.T. Nguyen, A.W.C. Liew, M.T. Tran, X.C. Pham, M.P. Nguyen, A novel genetic algorithm approach for simultaneous feature and classifier selection in multi classifier system, in: IEEE Congress on Evolutionary Computation (CEC), 2014, pp.1698-1705.

[23] Y. Chen, M-L. Wong, H. Li, Applying Ant Colony Optimization to configuring stacking ensembles for data mining, Expert Systems with Applications. 41 (2014), 2688-2702.

[24] Y. Zhang, S. Burer, W. N. Street, Ensemble Pruning Via Semi-definite Programming, Journal of Machine Learning Research. 7 (2006), 1315-1338.

[25] R.M.O. Cruz, R. Sabourin, G.D.C. Cavalcanti, T.I. Ren: META-DES: a dynamic ensemble selection framework using meta-learning, Pattern Recognition. 48 (5), 1925–1935 (2015).

[26] A.H.R. Ko, R. Sabourin, A.S. Britto: From dynamic classifier selection to dynamic ensemble selection, Pattern Recognition. 41, 1735–1748 (2008).

[27] T. Woloszynski, M. Kurzynski, P. Podsiadlo, G.W. Stachowiak, A measure of competence based on random classification for dynamic ensemble selection, Information Fusion. 13 (3) (2012), 207–213.

[28] T. Woloszynski, M. Kurzynski: A probabilistic model of classifier competence for dynamic ensemble selection, Pattern Recognition. 44 (2011), 2656–2668.

[29] K.-J. Kim, S.-B. Cho, An Evolutionary Algorithm Approach to Optimal Ensemble Classifiers for DNA Microarray Data Analysis, IEEE Trans. of Evolutionary Computation. 12(3) (2008), 377 – 388

[30] D. Karaboga, An Idea Based on Honey Bee Swarm for Numerical Optimization, Technical Report-TR06, Engineering Faculty, Computer Engineering Department, Erciyes University, 2005

[31] D. Karaboga, B. Akay, A comparative study of artificial bee colony algorithm, Applied Mathematics and Computation. 214 (2009), 108-132.

[32] C.-X. Zhang, J.-S. Zhang, RotBoost: A technique for combining Rotation Forest and AdaBoost, Pattern Recognition Letters. 29(10), 2008, 1524-1536.

[33] J.J. Rodriguez, L.I. Kuncheva, C.J. Alonso, Rotation Forest: A New Classifier Ensemble Method, IEEE Transactions On Pattern Analysis And Machine Intelligence. 28(10)(2006), 1619-1630.

[34] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, in: Proceedings of International Conference on Machine Learning (ICML), 1996, pp. 148-156.

[35] T.K. Ho, The random subspace method for constructing decision forests, IEEE Transactions on Pattern Analysis and Machine Intelligence. 20(8) (1998), 832-844.

[36] P. Viola, M. Jones, Robust Real-Time Face Detection, International Journal of Computer Vision. 57 (2002) 137-154.

[37] T.T. Nguyen, A.W.C. Liew, X.C. Pham, M.P. Nguyen, A Novel 2-Stage Combining Classifier Model with Stacking and Genetic Algorithm Based Feature Selection, in: D.-S. Huang, K.-H. Jo, L. Wang (Eds.), Intelligent Computing Methodologies, Springer International Publishing, 2014, pp. 33-43.

[38] G. Zhu, S. Kwong, Gbest-guided artificial bee colony algorithm for numerical function optimization, Applied Mathematics and Computation. 217(7)(1), 2010, 3166-3173.

[39] Rustu Akay, Alper Basturk, Adem Kalinli, Xin Yao, Parallel population-based algorithm portfolios: An empirical study, Neurocomputing. 247 (2017), 115-125.

[40] Leila Asadzadeh, A parallel artificial bee colony algorithm for the job shop scheduling problem with a dynamic migration strategy, Computers & Industrial Engineering.102 (2016), 359-367.

Author Biographies

Tien Thanh Nguyen received his PhD degree in computer science from the School of Information & Communication Technology, Griffith University, Australia in 2017. He is currently a Research Fellow at the School of Computing Science and Digital Media, Robert Gordon University, Aberdeen, Scotland, UK. His research interest is in the field of machine learning, pattern recognition, and evolutionary computation. He is a member of the IEEE since 2014.

Anh Vu Luong is currently a PhD student at the School of Information & Communication Technology, Griffith University, Australia. His research interest is in the field of machine learning and pattern recognition.

Manh Truong Dang is currently a Research Assistant at the School of Computing Science and Digital Media, Robert Gordon University, Aberdeen, Scotland, UK. His research interest is in the field of machine learning and pattern recognition.

Alan Wee-Chung Liew is currently an Associate Professor at the School of Information & Communication Technology, Griffith University, Australia. His research interest is in the field of machine learning, pattern recognition, computer vision, medical imaging, and

bioinformatics. He has served on the technical program committee of many international conferences and is on the editorial board of several journals, including the IEEE Transactions on Fuzzy Systems. He is a senior member of the IEEE since 2005.

John McCall is currently a Professor at the School of Computing Science and Digital Media, Robert Gordon University, Aberdeen, Scotland, UK. His research interest is in the area of naturally-inspired computing and their application to real-world problems arising in complex engineering and medical/biological systems.

Ensemble Selection based on Classifier's Confidence in Prediction

Tien Thanh Nguyen[1], Anh Vu Luong[2], Manh Truong Dang[1], Alan Wee-Chung Liew[2*], John McCall[1]

[1]School of Computing Science and Digital Media, Robert Gordon University, Aberdeen, Scotland, UK

[2]School of Information and Communication Technology, Griffith University, Australia

[*]Corresponding Author: Alan Wee-Chung Liew

Email: a.liew@griffith.edu.au

Supplementary Material

Paper: Ensemble selection based on classifier's confidence in predictions

Table S1. Classification error of the proposed method using 3 different entropy measures

	Proposed Method (Shannon)		**Proposed Method (Min)**		**Proposed Method (Collison)**	
	Mean	*Variance*	*Mean*	*Variance*	*Mean*	*Variance*
Artificial	0.2124	1.04E-03	0.2100	1.13E-03	0.2162	1.21E-03
Australian	0.1444	1.25E-03	0.1420	1.00E-03	0.1406	1.05E-03
Banana	0.1069	1.07E-04	0.1071	1.05E-04	0.1070	9.80E-05
Biodeg	0.1302	8.32E-04	0.1308	7.13E-04	0.1314	8.51E-04
Blood	0.2259	1.38E-03	0.2206	9.89E-04	0.2290	1.35E-03
Contraceptive	0.4587	1.29E-03	0.4619	1.49E-03	0.4589	1.94E-03
Dermatology	0.0289	7.14E-04	0.0299	7.94E-04	0.0317	4.66E-04
Fertility	0.1300	2.77E-03	0.1267	2.62E-03	0.1267	2.62E-03
Haberman	0.2615	1.45E-03	0.2584	1.95E-03	0.2616	1.78E-03
Heart	0.1741	3.12E-03	0.1679	2.81E-03	0.1753	2.92E-03
Hepatitis	0.1542	1.01E-02	0.1458	1.26E-02	0.1333	1.35E-02
Isolet	0.0481	6.69E-05	0.0481	6.69E-05	0.0482	6.90E-05
Led7digit	0.2673	4.25E-03	0.2680	4.04E-03	0.2707	3.89E-03
Marketing	0.6674	1.68E-04	0.6683	1.50E-04	0.6680	1.45E-04
Multiple-Features	0.0145	6.89E-05	0.0142	5.68E-05	0.0157	6.96E-05
Musk	0.1337	1.53E-03	0.1309	1.66E-03	0.1372	1.60E-03
Plant-Margin	0.1896	4.25E-04	0.1919	3.68E-04	0.1892	4.82E-04
Plant-Shape	0.4125	1.07E-03	0.4150	1.16E-03	0.4100	9.68E-04
Satimage	0.0970	1.53E-04	0.0969	1.55E-04	0.0971	1.49E-04
Skin-NonSkin	5.2777E-04	1.72E-08	5.2097E-04	1.83E-08	5.3049E-04	2.64E-08
Sonar	0.2161	7.91E-03	0.2130	8.07E-03	0.2067	5.63E-03
Spambase	0.0975	1.26E-04	0.0980	1.49E-04	0.0966	1.31E-04
Texture	0.0048	5.62E-06	0.0050	5.72E-06	0.0049	5.54E-06
Twonorm	0.0214	1.64E-05	0.0215	1.90E-05	0.0214	1.86E-05
Vehicle	0.2282	1.15E-03	0.2285	1.23E-03	0.2289	1.17E-03
Waveform-w-Noise	0.1549	2.14E-04	0.1547	2.00E-04	0.1551	2.06E-04
Waveform-wo-Noise	0.1475	3.57E-04	0.1469	3.37E-04	0.1471	3.46E-04
Wdbc	0.0305	4.51E-04	0.0316	3.78E-04	0.0328	4.88E-04
Wine	0.0245	1.22E-03	0.0320	1.45E-03	0.0341	2.80E-03
Yeast	0.4050	9.56E-04	0.4000	1.21E-03	0.4010	1.13E-03

Table S2. Wilcoxon test results between the proposed method and each benchmark algorithm (using 3 base classifiers)

	Decision Template		GA Meta-data		ACO		KNORA UNION		KNORA ELIMINATE		META-DES		Random Subspace		RotBoost	
	P-Value	W/L	P-Value	W/L	P-Value	W/L	P-Value		P-Value	W/L	P-Value	W/L	P-Value	W/L	P-Value	W/L
Abalone	1.99E-01		9.46E-01		5.40E-01		7.50E-01		3.08E-01		8.13E-01		2.29E-01		7.20E-03	No
Appendicitis	6.24E-02		6.92E-01		8.43E-02		9.77E-03	No	9.18E-02		2.54E-02	No	2.81E-01		1.88E-01	
Artificial	4.17E-04	W	2.15E-02	W	6.35E-02		5.39E-01		1.68E-02	W	3.70E-03	W	7.15E-06	W	5.77E-06	W
Assetnegotiation-F2	1.95E-03	W	8.40E-02		1.00E+00		1.95E-03	W	1.95E-03	No	1.95E-03	No	1.00E+00		1.95E-03	No
Assetnegotiation-F3	1.95E-03	W	5.86E-03	L	1.00E+00		1.95E-03	W	1.95E-03	No	1.95E-03	No	1.95E-02	W	1.95E-03	No
Assetnegotiation-F4	1.95E-03	W	5.57E-01		1.93E-01		1.95E-03	W	3.91E-03	No	1.95E-03	No	1.95E-03	W	1.95E-03	No
Australian	5.53E-01		1.70E-05	W	2.38E-04	W	2.65E-02	L	9.49E-03	W	2.04E-02	W	3.27E-01		1.53E-03	W
Banana	1.19E-01		7.52E-04	W	3.76E-03	W	5.30E-01		9.62E-06	W	2.28E-03	W	1.73E-06	W	4.02E-01	
BNG-Bridges	1.95E-03	W	1.95E-03	W	5.86E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	L	1.95E-03	W	1.95E-03	L
Biodeg	1.20E-01		1.50E-05	W	6.30E-06	W	1.99E-03	W	5.44E-03	W	4.25E-03	W	1.94E-01		2.36E-04	W
Blood	7.68E-04	W	2.58E-01		1.09E-04	W	3.32E-01		7.12E-01		2.50E-01		5.63E-01		6.24E-01	
BNG-Zoo	1.95E-03	W	1.95E-03	L	9.77E-03	L	1.95E-03	W	1.95E-03	W	1.95E-03	L	1.95E-03	W	1.95E-03	L
Breast-Tissue	9.91E-01		5.58E-02		1.72E-01		3.43E-01		7.50E-02		9.36E-01		2.61E-02	L	2.81E-01	
Bupa	3.06E-02	W	2.74E-05	W	8.41E-04	W	1.05E-04	W	2.13E-04	W	2.29E-03	W	1.09E-02	W	2.87E-02	W
Cleveland	5.80E-01		4.64E-01		8.93E-02		7.97E-03	L	2.50E-01		4.29E-01		5.09E-01		5.01E-01	
CNS	9.96E-02		1.18E-01		1.06E-01		7.73E-01		9.25E-01		4.74E-01		9.37E-01		8.05E-01	
Colon	3.28E-01		1.17E-01		1.44E-01		2.67E-01		1.22E-01		1.88E-02	W	6.35E-01		8.73E-01	
Contraceptive	2.29E-01		3.78E-06	W	9.28E-06	W	9.45E-01		3.11E-01		5.78E-02		3.43E-03	W	5.01E-01	
Dermatology	4.66E-04	W	1.49E-01		9.06E-02		5.03E-01		1.89E-01		6.09E-04	W	5.85E-01		6.01E-02	
Dowjones-1985-2003	1.72E-06	W	1.00E+00		5.00E-01		1.71E-06	W	1.70E-06	W	1.73E-06	W	2.43E-06	W	1.25E-01	
Duke	1.00E+00		7.50E-01		1.00E+00		5.86E-03	W	2.89E-01		1.81E-03	W	2.50E-01		9.77E-03	W
Electricity	3.02E-06	W	6.60E-02		6.60E-02		1.73E-06	W	1.73E-06	W	1.73E-06	W	1.73E-06	L	2.33E-01	
Fertility	7.31E-06	W	7.81E-03	W	1.80E-01		1.00E+00		6.25E-01		2.73E-01		2.50E-01		2.50E-01	
Haberman	3.26E-05	W	8.23E-05	W	1.35E-03	W	1.80E-02	W	3.91E-02	W	2.74E-01		3.15E-02	W	1.34E-01	
Hayes-Roth	8.77E-01		4.21E-01		6.74E-02		2.57E-01		1.58E-01		4.13E-01		1.34E-01		3.11E-03	W
Heart	1.71E-01		1.39E-04	W	9.82E-03	W	8.52E-01		9.39E-02		1.97E-01		3.18E-01		7.67E-01	
Hepatitis	6.26E-01		3.37E-01		3.97E-02	W	6.69E-01		6.86E-01		2.33E-01		2.75E-01		8.70E-01	
Hyperplane	1.95E-03	W	1.95E-03	L	1.95E-03	L	1.95E-03	W	1.95E-03	W	1.95E-03	L	1.95E-03	W	1.95E-03	W
Iris	9.15E-01		5.70E-01		1.10E-01		1.00E+00		1.00E+00		2.50E-01		1.56E-01		1.00E+00	
Isolet	2.37E-02	W	2.54E-06	W	1.92E-06	W	1.72E-06	W	1.71E-06	W	1.72E-06	W	5.18E-06	W	1.73E-06	W
Led7digit	9.25E-01		1.45E-04	W	1.77E-04	W	5.37E-01		9.41E-01		1.57E-04	W	1.68E-06	W	3.80E-01	
Letter	3.68E-06	W	2.35E-05	L	1.73E-06	L	1.73E-06	W	1.76E-05	L	1.72E-06	W	1.73E-06	W	1.73E-06	W

Leukemia	2.50E-01		5.63E-01		3.13E-02	W	2.54E-02	W	1.02E-01		5.47E-02		1.00E+00		1.07E-02	W
Magic	6.22E-01		4.05E-01		5.23E-01		7.46E-05	W	9.21E-04	W	3.87E-06	W	1.72E-06	L	1.72E-06	L
Mammographic	6.81E-01		9.01E-02		5.08E-03	W	1.05E-01		2.64E-01		6.46E-01		3.07E-04	L	3.66E-01	
Marketing	2.62E-01		1.72E-06	W	1.73E-06	W	4.25E-01		2.21E-03	W	1.47E-04	W	1.09E-01		1.09E-01	
Multiple-Features	9.91E-01		8.23E-01		6.72E-01		7.04E-01		2.82E-01		1.82E-03	W	3.71E-02	W	1.66E-06	W
Musk	9.18E-01		8.64E-01		2.16E-01		2.32E-03	W	6.01E-04	W	7.51E-02		1.87E-04	L	2.83E-06	W
Page-Blocks	3.58E-04	W	1.10E-01		3.99E-01		4.25E-05	W	4.18E-02	L	3.23E-03	L	2.09E-06	L	6.00E-05	W
Phoneme	3.38E-03	W	1.73E-06	L	1.73E-06	L	1.73E-06	W	2.34E-02	L	7.44E-03	W	6.62E-06	W	1.73E-06	W
Pima	3.28E-01		5.44E-05	W	1.82E-04	W	1.22E-01		5.97E-02		3.34E-01		5.72E-01		5.88E-01	
Plant-Margin	9.04E-01		1.69E-06	W	1.71E-06	W	3.85E-04	W	1.76E-04	W	1.71E-06	W	7.44E-03	L	1.71E-06	W
Plant-Shape	7.40E-01		5.40E-06	W	9.11E-06	W	5.99E-04	W	2.54E-04	W	4.97E-06	W	6.44E-05	L	1.72E-06	W
Poker	1.95E-03	W	8.40E-02		1.95E-03	L	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	3.91E-03	W
Randomtree	1.95E-03	L	1.95E-03	L	1.95E-03	L	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	L
Satimage	1.73E-06	W	9.96E-04	W	4.43E-02	L	1.73E-06	W	1.48E-01		1.93E-03	W	1.89E-04	L	1.73E-06	W
Skin-Nonskin	1.73E-06	W	7.65E-04	L	8.30E-04	L	1.72E-06	W	4.15E-01		1.73E-06	W	1.73E-06	W	1.73E-06	W
Sonar	2.65E-01		3.79E-02	W	2.68E-01		3.79E-02	W	2.06E-01		4.45E-02	W	1.61E-03	L	2.62E-01	
Spambase	4.78E-01		2.84E-06	W	7.18E-06	W	7.33E-01		4.34E-04	W	6.82E-01		6.34E-01		9.95E-06	L
Svmguide	3.27E-01		5.91E-05	W	3.07E-05	W	5.74E-01		6.13E-01		1.68E-02	W	1.40E-05	W	4.15E-06	W
Tae	3.40E-02	L	2.78E-02	W	4.14E-01		8.27E-01		8.20E-01		4.83E-01		9.89E-01		3.59E-01	
Texture	1.68E-05	W	7.65E-01		8.13E-01		1.70E-06	W	2.57E-02	W	2.76E-06	W	1.71E-06	W	1.73E-06	W
Twonorm	6.87E-02		2.28E-06	W	1.73E-06	W	5.54E-01		2.21E-01		7.52E-01		8.32E-06	W	1.89E-06	W
Vehicle	5.19E-02		1.01E-03	W	1.50E-03	W	7.27E-04	W	3.72E-04	W	2.63E-05	W	1.85E-03	W	2.83E-06	W
Vertebral	6.51E-01		9.14E-01		9.12E-03	L	4.10E-01		3.52E-01		7.33E-01		2.45E-06	W	7.03E-01	
Waveform-w-Noise	1.82E-02	W	7.27E-06	W	1.94E-05	W	1.11E-05	W	1.73E-03	W	5.59E-01		4.27E-06	W	3.98E-02	W
Waveform-wo-Noise	9.65E-01		2.49E-06	W	4.52E-06	W	2.52E-06	W	1.58E-03	W	6.57E-01		2.62E-01		3.99E-03	W
Wdbc	3.31E-02	W	2.57E-01		6.01E-03	W	2.77E-02	W	1.02E-02	W	2.58E-02	W	9.59E-02		2.67E-04	W
Wine	1.55E-01		1.00E+00		6.78E-01		5.55E-01		3.75E-01		6.35E-01		2.66E-01		2.58E-01	
Wine-Red	1.71E-06	W	1.03E-04	W	5.61E-05	W	1.71E-01		7.37E-01		2.24E-03	W	1.72E-06	L	6.82E-03	L
Wine-White	1.73E-06	W	6.98E-05	W	3.89E-06	W	2.46E-04	W	3.76E-01		8.16E-02		1.73E-06	L	3.72E-05	L
Yeast	3.65E-01		2.46E-06	W	2.12E-06	W	1.79E-01		5.67E-01		3.85E-04	W	1.73E-06	W	5.20E-04	W

**The color values indicate that we reject the null hypothesis that 'two methods perform equally on the dataset', 'W' or 'L' mean for the dataset, the proposed method wins (in green color) or loses (in red color) to the benchmark algorithm based on the Wilcoxon signed rank test*

Table S3. Wilcoxon test results between the proposed method and each benchmark algorithm (using 7 base classifiers)

	vs. Decision Template		vs. GA Meta-data		vs. KNORA UNION		vs. KNORA ELIMINATE		vs. META-DES		vs. Random Subspace		vs. RotBoost		vs. ACO	
	P-Value	W/L	P-Value	W/L	P-Value	W/L	P-Value	W/L	P-Value	W/L	P-Value	W/L	P-Value	W/L	P-Value	W/L
Abalone	3.32E-04	W	2.59E-05	Yes	8.20E-01		4.24E-03	W	7.73E-01		5.89E-01		2.61E-01		3.51E-03	W
Appendicitis	1.12E-01		1.22E-01		1.52E-01		8.02E-01		5.76E-02		4.53E-01		5.01E-01		3.45E-01	
Artificial	4.38E-02	W	9.14E-05	W	9.68E-01		2.52E-03	W	4.82E-01		1.28E-05	W	1.51E-05	W	9.15E-01	
Assetnegotiation-F2	1.95E-03	W	1.00E+00		1.95E-03	W	1.95E-03	W	1.95E-03	W	3.91E-03	W	1.00E+00		7.81E-03	W
Assetnegotiation-F3	1.95E-03	W	5.00E-01		1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	3.13E-02	W	3.91E-03	W
Assetnegotiation-F4	1.95E-03	W	5.00E-01		1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.00E+00		5.86E-03	W
Australian	2.10E-01		4.61E-04	W	2.21E-01		3.22E-04	W	3.44E-03	W	2.09E-01		9.75E-04	W	8.30E-05	W
Banana	1.88E-01		2.56E-06	W	3.91E-02	W	6.48E-06	W	7.06E-05	W	1.73E-06	W	1.41E-02	W	2.35E-06	W
BNG-Bridges	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	L	1.95E-03	W
Biodeg	7.81E-01		1.44E-05	W	1.07E-02	W	5.23E-06	W	6.13E-02		3.19E-01		2.58E-05	W	4.19E-06	W
Blood	3.86E-04	W	7.18E-06	W	1.25E-02	W	7.76E-06	W	5.98E-04	W	1.09E-01		8.11E-03	W	1.01E-04	W
BNG-Zoo	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W
Breast-Tissue	3.80E-01		3.39E-02	W	1.32E-01		2.49E-03	W	1.06E-02	W	8.06E-01		5.43E-02		1.67E-02	W
Bupa	9.67E-01		3.31E-03	W	6.66E-01		2.14E-02	W	1.29E-04	W	4.34E-02	W	5.86E-02		7.01E-03	W
Cleveland	1.11E-01		2.53E-01		1.61E-01		1.70E-02	W	5.29E-01		8.24E-02		5.11E-02		1.78E-01	
CNS	2.03E-01		8.08E-03	W	2.65E-01		3.39E-01		7.72E-01		9.85E-01		8.95E-01		2.01E-01	
Colon	1.56E-02	L	8.01E-02		2.93E-01		6.35E-03	W	3.75E-01		6.14E-01		1.00E+00		4.03E-01	
Contraceptive	7.85E-01		8.04E-06	W	4.73E-01		1.49E-05	W	7.94E-06	W	1.13E-04	W	1.37E-03	W	2.12E-06	W
Dermatology	3.25E-03	W	8.35E-02		3.86E-01		6.10E-05	W	4.51E-01		8.70E-01		6.45E-03	W	1.21E-03	W
Dowjones-1985-2003	1.70E-06	W	3.13E-02	L	8.08E-06	W	2.00E-04	W	1.73E-06	W	2.46E-06	W	7.81E-01		1.54E-03	W
Duke	1.00E+00		1.95E-02	W	6.84E-03	W	4.38E-03	W	1.09E-01		1.00E+00		1.60E-01		2.81E-01	
Electricity	1.73E-06	W	1.92E-06	W	1.73E-06	W	1.73E-06	W	1.73E-06	W	2.67E-02	W	1.73E-06	W	2.55E-06	W
Fertility	4.02E-05	W	8.21E-02		9.63E-02		2.83E-01		2.43E-01		1.07E-02	L	1.07E-02	L	1.86E-01	
Haberman	1.07E-02	W	1.07E-04	W	4.30E-01		1.20E-01		6.66E-01		2.08E-01		9.90E-01		9.71E-04	W
Hayes-Roth	6.72E-01		1.41E-01		9.55E-01		3.42E-01		4.31E-02	W	7.86E-05	W	7.12E-06	W	3.03E-02	L
Heart	2.05E-01		1.39E-03	W	7.50E-01		3.34E-04	W	6.91E-02		8.22E-01		4.55E-01		1.19E-02	W
Hepatitis	1.07E-01		8.44E-02		6.94E-01		7.60E-01		8.26E-01		2.90E-01		8.25E-01		1.76E-01	
Hyperplane	1.95E-03	W	1.21E-01		1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	5.86E-03	W
Iris	1.66E-01		2.89E-01		1.00E+00		3.13E-02	W	1.00E+00		1.09E-01		1.00E+00		5.50E-03	W
Isolet	8.94E-01		6.26E-06	W	2.34E-06	W	1.73E-06	W	2.34E-06	W	2.32E-06	W	1.73E-06	W	1.73E-06	W
Led7digit	4.40E-01		5.89E-05	W	2.98E-02	L	9.08E-03	W	2.34E-04	W	1.67E-06	W	5.98E-01		4.65E-03	W
Letter	1.73E-06	W	1.73E-06	W	1.72E-06	W	1.72E-06	W	1.73E-06	W	1.73E-06	W	1.73E-06	W	1.31E-02	L
Leukemia	1.00E+00		3.36E-01		5.00E-01		1.25E-01		3.75E-01		1.00E+00		5.86E-03	W	9.77E-02	

Magic	5.60E-06	W	1.73E-06	W	3.15E-06	W	1.73E-06	W	1.73E-06	W	1.73E-06	W	4.17E-01		1.72E-06	W
Mammographic	1.79E-02	W	2.52E-04	W	9.98E-02		1.56E-05	W	6.74E-04	W	4.80E-03	L	2.77E-01		1.30E-03	W
Marketing	9.84E-03	W	1.73E-06	W	6.09E-02		2.46E-03	W	2.71E-05	W	1.14E-02	W	8.12E-01		1.73E-06	W
Multiple-Features	5.29E-01		2.79E-01		8.36E-05	W	3.44E-05	W	2.50E-04	W	8.16E-04	W	1.68E-06	W	7.79E-02	
Musk	5.23E-01		8.52E-03	W	5.57E-06	W	6.97E-05	W	1.72E-06	W	1.79E-01		1.73E-06	W	4.45E-03	W
Page-Blocks	8.59E-02		2.64E-01		2.53E-06	W	7.00E-05	W	6.56E-03	W	7.61E-04	L	2.35E-06	W	1.50E-01	
Phoneme	1.92E-06	W	3.07E-03	W	1.73E-06	W	1.31E-05	W	2.34E-06	W	1.73E-06	W	1.72E-06	W	2.84E-01	
Pima	7.19E-01		4.46E-05	W	9.43E-01		9.73E-05	W	2.07E-02	W	8.83E-02		5.65E-01		1.37E-05	W
Plant-Margin	2.39E-01		2.94E-05	W	1.52E-04	W	5.30E-06	W	1.69E-06	W	1.40E-02	L	1.72E-06	W	1.71E-06	W
Plant-Shape	1.11E-03	W	1.69E-06	W	1.63E-06	W	5.46E-05	W	3.09E-06	W	4.03E-05	L	1.71E-06	W	3.19E-05	W
Poker	1.95E-03	W	2.32E-01		1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	2.32E-01	
RandomTree	1.95E-03	W	1.95E-02	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	1.95E-03	W	4.88E-02	W
Satimage	1.73E-06	W	2.85E-06	W	1.73E-06	W	1.73E-06	W	3.16E-06	W	8.12E-01		1.73E-06	W	1.59E-03	W
Skin-NonSkin	9.31E-06	W	1.55E-03	W	2.51E-06	W	3.13E-02	W	1.73E-06	W	1.73E-06	W	1.73E-06	W	4.99E-03	W
Sonar	1.84E-02	L	3.22E-02	W	2.37E-03	W	1.01E-01		3.14E-02	W	2.62E-01		1.84E-04	W	5.00E-04	W
Spambase	4.45E-05	L	2.21E-03	W	1.48E-03	W	6.03E-05	W	4.89E-01		2.35E-06	W	8.33E-02		1.77E-03	W
Svmguide	4.87E-02	L	9.75E-04	W	3.35E-01		7.57E-02		1.69E-02	W	9.30E-05	W	1.10E-05	W	2.11E-05	W
Tae	2.36E-03	L	9.37E-02		1.29E-01		8.78E-01		9.07E-02		3.54E-02	W	7.84E-03	W	5.65E-04	W
Texture	3.45E-06	W	9.86E-03	W	1.70E-06	W	1.69E-06	W	8.97E-06	W	1.73E-06	W	1.73E-06	W	4.09E-03	W
Twonorm	1.65E-02	L	2.21E-06	W	6.94E-02		7.23E-06	W	3.35E-01		1.06E-05	W	1.90E-06	W	1.73E-06	W
Vehicle	9.31E-01		1.57E-02	W	2.33E-06	W	3.71E-06	W	1.60E-03	W	6.02E-04	W	2.71E-06	W	1.98E-04	W
Vertebral	1.25E-01		8.17E-04	W	3.52E-01		2.22E-01		5.62E-01		1.58E-06	W	2.90E-01		2.46E-01	
Waveform-w-Noise	7.74E-04	W	2.54E-06	W	2.51E-06	W	1.73E-06	W	8.36E-05	W	3.46E-06	W	7.76E-06	W	1.72E-06	W
Waveform-wo-Noise	2.28E-03	W	1.71E-06	W	2.09E-06	W	1.72E-06	W	3.06E-04	W	3.42E-06	W	2.53E-06	W	1.71E-06	W
Wdbc	3.18E-01		8.71E-01		7.94E-01		2.01E-02	W	5.42E-01		7.87E-01		1.12E-02	W	6.00E-01	
Wine	6.63E-02		1.21E-01		7.78E-01		8.70E-01		8.18E-01		3.42E-03	L	3.57E-01		2.42E-01	
Wine-Red	1.69E-06	W	2.81E-06	W	4.37E-03	W	1.70E-05	W	2.90E-05	W	5.28E-06	L	8.52E-05	W	4.66E-06	W
Wine-White	1.73E-06	W	1.73E-06	W	4.50E-06	W	1.73E-06	W	1.73E-06	W	1.73E-06	L	2.55E-06	W	1.73E-06	W
Yeast	2.85E-02	W	1.73E-06	W	5.25E-02		2.05E-04	W	2.52E-06	W	1.73E-06	W	9.24E-06	W	1.91E-06	W

**The color values indicate that we reject the null hypothesis that 'two methods perform equally on the dataset', 'W' or 'L' mean for the dataset, the proposed method wins (in green color) or loses (in red color) to the benchmark algorithm based on the Wilcoxon signed rank test*

Table S4. Ranking of all methods on experimental datasets (using 3 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
Abalone	7	6	5	3	1	8	2	4	9
Appendicitis	8	9	7	6	5	2	4	1	3
Artificial	5	3.5	1	8	9	6	3.5	2	7
AssetNegotiation-F2	7	9	4	5	1	2	3	6	8
AssetNegotiation-F3	2	9	5	7	1	3.5	3.5	6	8
AssetNegotiation-F4	8	9	4	7	1	2	3	6	5
Australian	8	9	3	4	7	6	5	1	2
Banana	5	7	2	9	1	6	8	3	4
Biodeg	9	8	1	2	7	3	5.5	5.5	4
Blood	7	9	3	2	5	6	4	1	8
BNG-Bridges	5	4	3	9	1	2	6	7	8
BNG-Zoo	4	3	5	9	1	2	6	7	8
Breast-Tissue	2	9	6	1	4	8	3	5	7
Bupa	9	8	1	5	4	7	6	3	2
Cleveland	8	9	7	3	4	5	2	1	6
CNS	9	8	2.5	6	4.5	4.5	1	2.5	7
Colon	8	7	4	2	3	9	6	5	1
Contraceptive	9	8	3	7	4	6	5	2	1
Dermatology	6	4	2	1	7	9	5	3	8
DowJones-1985-2003	2	2	2	5	4	9	6	8	7
Duke	4	1	2	5	7	9	6	8	3
Electricity	3.5	3.5	2	1	5	8	7	9	6
Fertility	8	6.5	3	1.5	1.5	6.5	5	4	9
Haberman	7	8	1	6	5	2	4	3	9
Hayes-Roth	2	1	3.5	8	9	5	6	7	3.5
Heart	9	8	2	6	3.5	5	7	3.5	1
Hepatitis	7	9	5	1	4	8	2.5	2.5	6
Hyperplane	1	2	4	9	8	3	5	7	6
Iris	3	8	6	9	6	2	6	4	1
Isolet	5	6	1	3	9	8	4	7	2
Led7digit	6	8	2	9	5	7	3	1	4
Letter	2	1	4	6	9	7	3	8	5

Leukemia	3	6	1.5	1.5	9	7	5	8	4
Magic	6	4	3	2	1	9	8	7	5
Mammographic	8	9	6.5	1	2	5	4	3	6.5
Marketing	8	9	2	5	1	7	6	3	4
Multiple-Features	5.5	1	4	7	9	8	2	5.5	3
Musk	4	2	3	1	9	6	8	7	5
Page-Blocks	2	6	5	1	9	3	4	7	8
Phoneme	1.5	1.5	4	7	9	5	3	8	6
Pima	8	9	6	7	4	3	1	2	5
Plant-Margin	6	7	3	1	9	8	5	4	2
Plant-Shape	8	6	3	1	9	7	5	4	2
Poker	2	1	3	8	4	7	6	5	9
RandomTree	3	3	5	9	1	7	6	8	3
Satimage	6	2	3	1	9	5	4	8	7
Skin-NonSkin	1	2	4	7	8	6	3	5	9
Sonar	9	5	3	1	4	8	6	7	2
Spambase	8	9	5	2	1	3	7	6	4
Svmguide	6	7	3	8	9	5	4	2	1
Tae	9	7	3	5	8	6	2	4	1
Texture	2	3	1	8	9	6.5	4	6.5	5
Twonorm	9	7	2	6	8	3.5	5	3.5	1
Vehicle	6	4	2	5	9	8	7	3	1
Vertebral	6	1	5	9	3	4	2	7	8
Waveform-w-Noise	9	8	2	7	3	1	5	6	4
Waveform-wo-Noise	9	8	2	4	5	1	6	7	3
Wdbc	2	7	1	4	9	6	8	5	3
Wine	4	5	3	1	7	2	8	6	9
Wine-Red	7	8	3	1	2	6	4	5	9
Wine-White	7	8	4	1	2	5	3	6	9
Yeast	8	7	2	9	6	5	3	1	4
Average	5.78	5.89	3.27	4.77	5.23	5.48	4.68	4.88	5.02

Table S5. Ranking of all methods on experimental datasets (using 7 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
Abalone	9	8	2	5	1	3.5	6	3.5	7
Appendicitis	9	8	7	5	4	2	6	3	1
Artificial	7	3	4	8	9	1	6	2	5
AssetNegotiation-F2	2	9	2	7	2	4	6	5	8
AssetNegotiation-F3	3	9	1	8	2	4	6	5	7
AssetNegotiation-F4	3	9	1.5	8	1.5	5	7	6	4
Australian	8	9	2	3	6	5	7	4	1
Banana	8	7	1	9	4	5	6	3	2
Biodeg	7	9	2	3	6	4	8	5	1
Blood	9	7	1	2	4	5	8	3	6
BNG-Bridges	8	5	2	9	1	6	7	4	3
BNG-Zoo	7	4	1	9	3	2	8	6	5
Breast-Tissue	6	7	1	2	5	8	9	4	3
Bupa	8	9	3	5	4	7	6	1	2
Cleveland	7	8	6	2	3	5	9	4	1
CNS	9	8	1	4	3	2	5.5	7	5.5
Colon	8	6	3.5	2	3.5	5	9	7	1
Contraceptive	9	8	2	5	4	7	6	3	1
Dermatology	5	6	4	3	8	2	9	1	7
DowJones-1985-2003	1	5	3	8	2	9	4	6	7
Duke	8	1	4	2.5	5	6	9	7	2.5
Electricity	3	2	1	4	9	6	7	8	5
Fertility	8	7	5	1.5	1.5	4	6	3	9
Haberman	9	7	2	5	1	4	6	3	8
Hayes-Roth	2	1	4	8	9	7	6	3	5
Heart	8	7	3	4	2	6	9	5	1
Hepatitis	9	8	4	1	3	5.5	5.5	2	7
Hyperplane	2	3	1	9	8	4	5	6	7
Iris	1	6	2	8	3	4.5	7	4.5	9
Isolet	5	7	1	3	8	4	9	6	2
Led7digit	8	6	4	9	3	7	5	1	2
Letter	4	2	1	6	9	8	3	7	5

Leukemia	6	8	2	3	9	5	7	4	1
Magic	9	6	2	5	1	8	7	4	3
Mammographic	9	7	2	1	3	6	8	4	5
Marketing	9	8	2	5	1	7	6	3	4
Multiple-Features	3	4	2	5.5	9	5.5	8	7	1
Musk	4	5	2	3	9	7	6	8	1
Page-Blocks	2	4	3	1	9	6	7	8	5
Phoneme	3	2	1	8	9	5	4	7	6
Pima	8	9	1	5	3	6	7	2	4
Plant-Margin	8	6	2	1	9	7	5	4	3
Plant-Shape	9	5	2	1	8	7	4	6	3
Poker	5	2	1	8	4	6	7	3	9
RandomTree	3	2	1	9	8	4	5	7	6
Satimage	6	3	2	1	9	4	5	8	7
Skin-NonSkin	5	3	1	8	9	7	2	6	4
Sonar	6	9	3	2	8	5	4	7	1
Spambase	5	6	2	8	4	3	9	7	1
Svmguide	6	8	3	7	9	5	4	2	1
Tae	6	9	2	7	8	5	3	4	1
Texture	2	4	1	8	9	6	3	7	5
Twonorm	7.5	7.5	2	5	9	3	6	4	1
Vehicle	3	6	1	5	9	4	8	7	2
Vertebral	8	7	3	9	5.5	2	5.5	4	1
Waveform-w-Noise	7.5	6	1	7.5	4	2	9	5	3
Waveform-wo-Noise	8	7	1	4	5	2	9	6	3
Wdbc	5	2.5	6.5	4	8.5	2.5	8.5	6.5	1
Wine	2.5	2.5	7	1	4	6	5	8	9
Wine-Red	8	7	2	1	4	5	6	3	9
Wine-White	8	6	2	1	5	4	7	3	9
Yeast	8	7	1	9	5	6	4	2	3
Average	6.12	5.96	2.35	5.02	5.40	4.98	6.37	4.75	4.06

Table S6. Ranking of all methods on large scale datasets (using 3 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
AssetNegotiation-F2	7	9	4	5	1	2	3	6	8
AssetNegotiation-F3	2	9	5	7	1	3.5	3.5	6	8
AssetNegotiation-F4	8	9	4	7	1	2	3	6	5
BNG-Bridges	5	4	3	9	1	2	6	7	8
BNG-Zoo	4	3	5	9	1	2	6	7	8
DowJones-1985-2003	2	2	2	5	4	9	6	8	7
Hyperplane	1	2	4	9	8	3	5	7	6
Poker	2	1	3	8	4	7	6	5	9
RandomTree	3	3	5	9	1	7	6	8	3
Skin-NonSkin	1	2	4	7	8	6	3	5	9
Average	3.50	4.40	3.90	7.50	3.00	4.35	4.75	6.50	7.10

*The datasets with more than 100000 observations

Table S7. Ranking of all methods on small scale datasets with high dimension (using 3 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
Colon	8	7	4	2	3	9	6	5	1
Duke	4	1	2	5	7	9	6	8	3
CNS	9	8	2.5	6	4.5	4.5	1	2.5	7
Leukemia	3	6	1.5	1.5	9	7	5	8	4
Average	6.00	5.50	2.50	3.63	5.88	7.38	4.50	5.88	3.75

*The datasets with less than 500 observations

Table S8. Ranking of all methods on small scale datasets with low dimension (using 3 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
Appendicitis	8	9	7	6	5	2	4	1	3
Breast-Tissue	2	9	6	1	4	8	3	5	7
Bupa	9	8	1	5	4	7	6	3	2
Cleveland	8	9	7	3	4	5	2	1	6
Dermatology	6	4	2	1	7	9	5	3	8
Fertility	8	6.5	3	1.5	1.5	6.5	5	4	9
Haberman	7	8	1	6	5	2	4	3	9
Hayes-Roth	2	1	3.5	8	9	5	6	7	3.5
Heart	9	8	2	6	3.5	5	7	3.5	1
Hepatitis	7	9	5	1	4	8	2.5	2.5	6
Iris	3	8	6	9	6	2	6	4	1
Musk	4	2	3	1	9	6	8	7	5
Sonar	9	5	3	1	4	8	6	7	2
Svmguide	6	7	3	8	9	5	4	2	1
Tae	9	7	3	5	8	6	2	4	1
Vertebral	6	1	5	9	3	4	2	7	8
Wine	4	5	3	1	7	2	8	6	9
Average	6.29	6.26	3.74	4.26	5.47	5.32	4.74	4.12	4.79

*The datasets with less than 500 observations

Table S9. Ranking of all methods on low dimension datasets (using 3 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
Abalone	7	6	5	3	1	8	2	4	9
Appendicitis	8	9	7	6	5	2	4	1	3
Artificial	5	3.5	1	8	9	6	3.5	2	7
AssetNegotiation-F2	7	9	4	5	1	2	3	6	8
AssetNegotiation-F3	2	9	5	7	1	3.5	3.5	6	8
AssetNegotiation-F4	8	9	4	7	1	2	3	6	5
Banana	5	7	2	9	1	6	8	3	4
Blood	7	9	3	2	5	6	4	1	8
Breast-Tissue	2	9	6	1	4	8	3	5	7
Bupa	9	8	1	5	4	7	6	3	2
Contraceptive	9	8	3	7	4	6	5	2	1
DowJones-1985-2003	2	2	2	5	4	9	6	8	7
Electricity	3.5	3.5	2	1	5	8	7	9	6
Fertility	8	6.5	3	1.5	1.5	6.5	5	4	9
Haberman	7	8	1	6	5	2	4	3	9
Hayes-Roth	2	1	3.5	8	9	5	6	7	3.5
Hyperplane	1	2	4	9	8	3	5	7	6
Iris	3	8	6	9	6	2	6	4	1
Led7digit	6	8	2	9	5	7	3	1	4
Magic	6	4	3	2	1	9	8	7	5
Mammographic	8	9	6.5	1	2	5	4	3	6.5
Page-Blocks	2	6	5	1	9	3	4	7	8
Phoneme	1.5	1.5	4	7	9	5	3	8	6
Pima	8	9	6	7	4	3	1	2	5
Poker	2	1	3	8	4	7	6	5	9
RandomTree	3	3	5	9	1	7	6	8	3
Skin-NonSkin	1	2	4	7	8	6	3	5	9
Tae	9	7	3	5	8	6	2	4	1
Vertebral	6	1	5	9	3	4	2	7	8
Yeast	8	7	2	9	6	5	3	1	4
Average	5.20	5.87	3.70	5.78	4.48	5.30	4.30	4.63	5.73

*The datasets with less than or equal to 10 features

Table S10. Ranking of all methods on high dimension datasets (using 3 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
Colon	8	7	4	2	3	9	6	5	1
Duke	4	1	2	5	7	9	6	8	3
CNS	9	8	2.5	6	4.5	4.5	1	2.5	7
Isolet	5	6	1	3	9	8	4	7	2
Leukemia	3	6	1.5	1.5	9	7	5	8	4
Multiple-Features	5.5	1	4	7	9	8	2	5.5	3
Musk	4	2	3	1	9	6	8	7	5
Average	5.50	4.43	2.57	3.64	7.21	7.36	4.57	6.14	3.57

*Datasets with more than or equal to 100 features

Table S11. Ranking of all methods on datasets with large number of class labels (using 3 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
DowJones-1985-2003	2	2	2	5	4	9	6	8	7
Isolet	5	6	1	3	9	8	4	7	2
Letter	2	1	4	6	9	7	3	8	5
Plant-Margin	6	7	3	1	9	8	5	4	2
Plant-Shape	8	6	3	1	9	7	5	4	2
Average	4.60	4.40	2.60	3.20	8.00	7.80	4.60	6.20	3.60

*Datasets with more than 26 class labels

Table S12. Ranking of all methods on binary datasets (using 3 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
Appendicitis	8	9	7	6	5	2	4	1	3
Artificial	5	3.5	1	8	9	6	3.5	2	7
AssetNegotiation-F2	7	9	4	5	1	2	3	6	8
AssetNegotiation-F3	2	9	5	7	1	3.5	3.5	6	8
AssetNegotiation-F4	8	9	4	7	1	2	3	6	5
Australian	8	9	3	4	7	6	5	1	2
Banana	5	7	2	9	1	6	8	3	4
Biodeg	9	8	1	2	7	3	5.5	5.5	4
Blood	7	9	3	2	5	6	4	1	8
Bupa	9	8	1	5	4	7	6	3	2
Colon	8	7	4	2	3	9	6	5	1
Duke	4	1	2	5	7	9	6	8	3
Electricity	3.5	3.5	2	1	5	8	7	9	6
CNS	9	8	2.5	6	4.5	4.5	1	2.5	7
Fertility	8	6.5	3	1.5	1.5	6.5	5	4	9
Haberman	7	8	1	6	5	2	4	3	9
Heart	9	8	2	6	3.5	5	7	3.5	1
Hepatitis	7	9	5	1	4	8	2.5	2.5	6
Hyperplane	1	2	4	9	8	3	5	7	6
Leukemia	3	6	1.5	1.5	9	7	5	8	4
Magic	6	4	3	2	1	9	8	7	5
Mammographic	8	9	6.5	1	2	5	4	3	6.5
Musk	4	2	3	1	9	6	8	7	5
Phoneme	1.5	1.5	4	7	9	5	3	8	6
Pima	8	9	6	7	4	3	1	2	5
RandomTree	3	3	5	9	1	7	6	8	3
Skin-NonSkin	1	2	4	7	8	6	3	5	9
Sonar	9	5	3	1	4	8	6	7	2
Spambase	8	9	5	2	1	3	7	6	4
Twonorm	9	7	2	6	8	3.5	5	3.5	1
Wdbc	2	7	1	4	9	6	8	5	3
Average	6.03	6.39	3.24	4.55	4.76	5.39	4.94	4.79	4.92

Table S13. Ranking of all methods on large scale datasets (using 7 base classifiers)

	GA Meta-data	ACO	Proposed Method	Random Subspace	RotBoost	META-DES	KNORA ELIMINATE	KNORA UNION	Decision Template
AssetNegotiation-F2	2	9	2	7	2	4	6	5	8
AssetNegotiation-F3	3	9	1	8	2	4	6	5	7
AssetNegotiation-F4	3	9	1.5	8	1.5	5	7	6	4
BNG-Bridges	8	5	2	9	1	6	7	4	3
BNG-Zoo	7	4	1	9	3	2	8	6	5
DowJones-1985-2003	1	5	3	8	2	9	4	6	7
Hyperplane	2	3	1	9	8	4	5	6	7
Poker	5	2	1	8	4	6	7	3	9
RandomTree	3	2	1	9	8	4	5	7	6
Skin-NonSkin	5	3	1	8	9	7	2	6	4
Average	3.90	5.10	1.45	8.30	4.05	5.10	5.70	5.40	6.00

*The datasets with more than 100000 observations

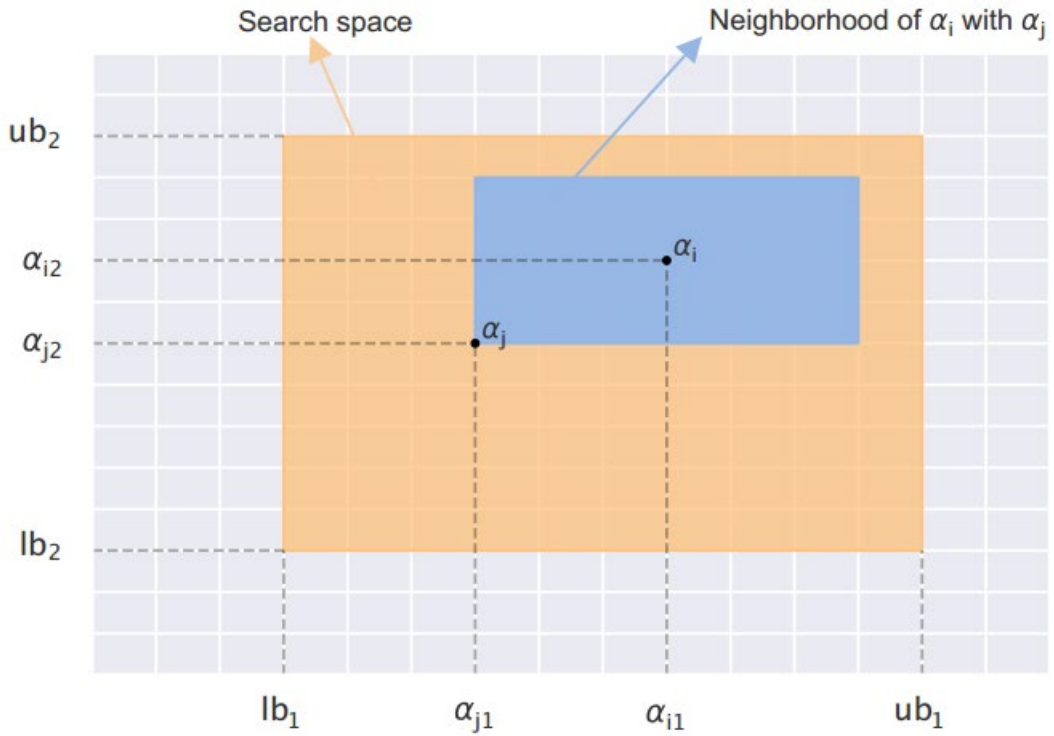


Fig.S1. The search space and neighborhood to generate new candidate

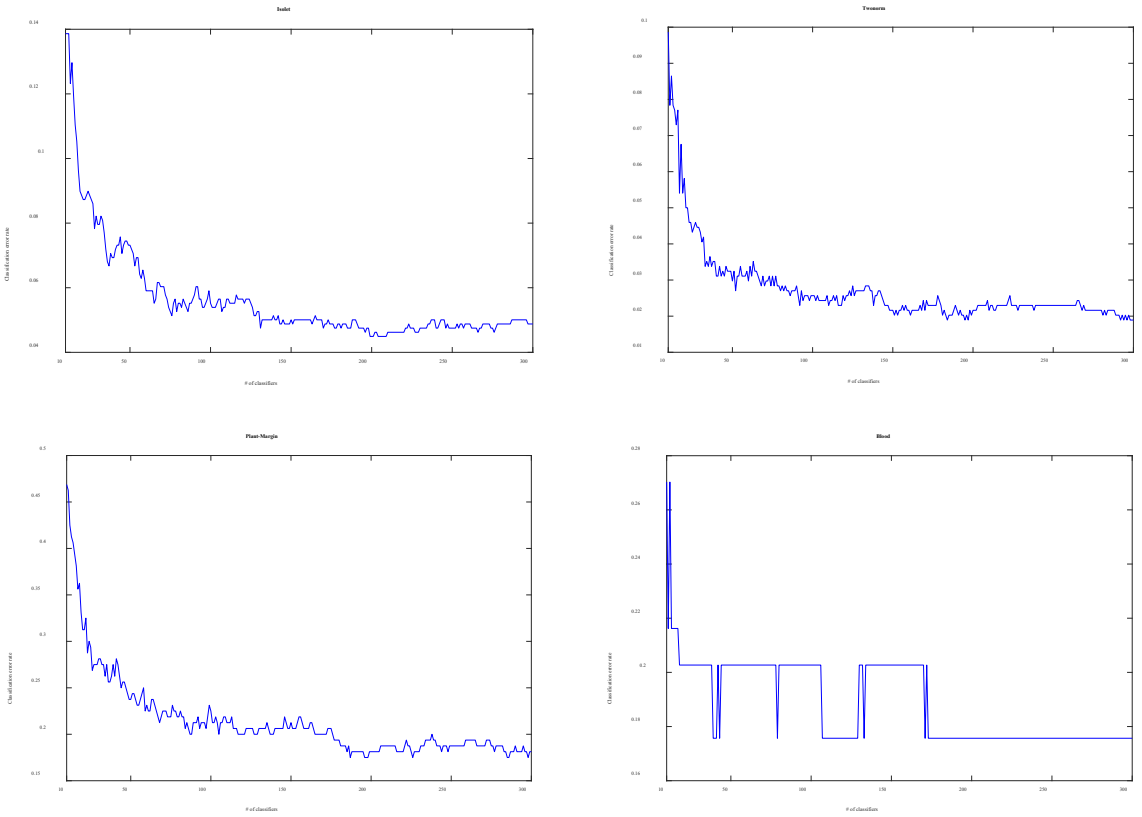


Fig.S2. Classification error with different number of base classifiers in Random Subspace

In our experiment studies, the number of tree classifiers in the random subspace method was set to 200. This value is suggested in several research published before [1-3]. Fig.S2 presents the relationship between the classification error rate and the number of classifiers used in the Random Subspace method on 4 datasets. It is observed that the classification error rate reduces and then changes very slightly beyond 200 classifiers. On the Blood dataset, the convergence occurs even before using 200 classifiers. Therefore, using 200 classifiers is a good choice for Random Subspace method that balances between computational complexity and performance.

[1] C.D. Sutton, Classification and Regression Trees, Bagging, and Boosting, in: C.R. Rao, E.J. Wegman, J.L. Solka (Eds.), Handbook of Statistics, Elsevier, 2005, pp. 303-329.

[2] P. Viola, M. Jones, Robust Real-Time Face Detection, International Journal of Computer Vision. 57 (2002) 137-154.

[3] T. Dietterich, An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization, Machine Learning. 40 (2000) 139-157.

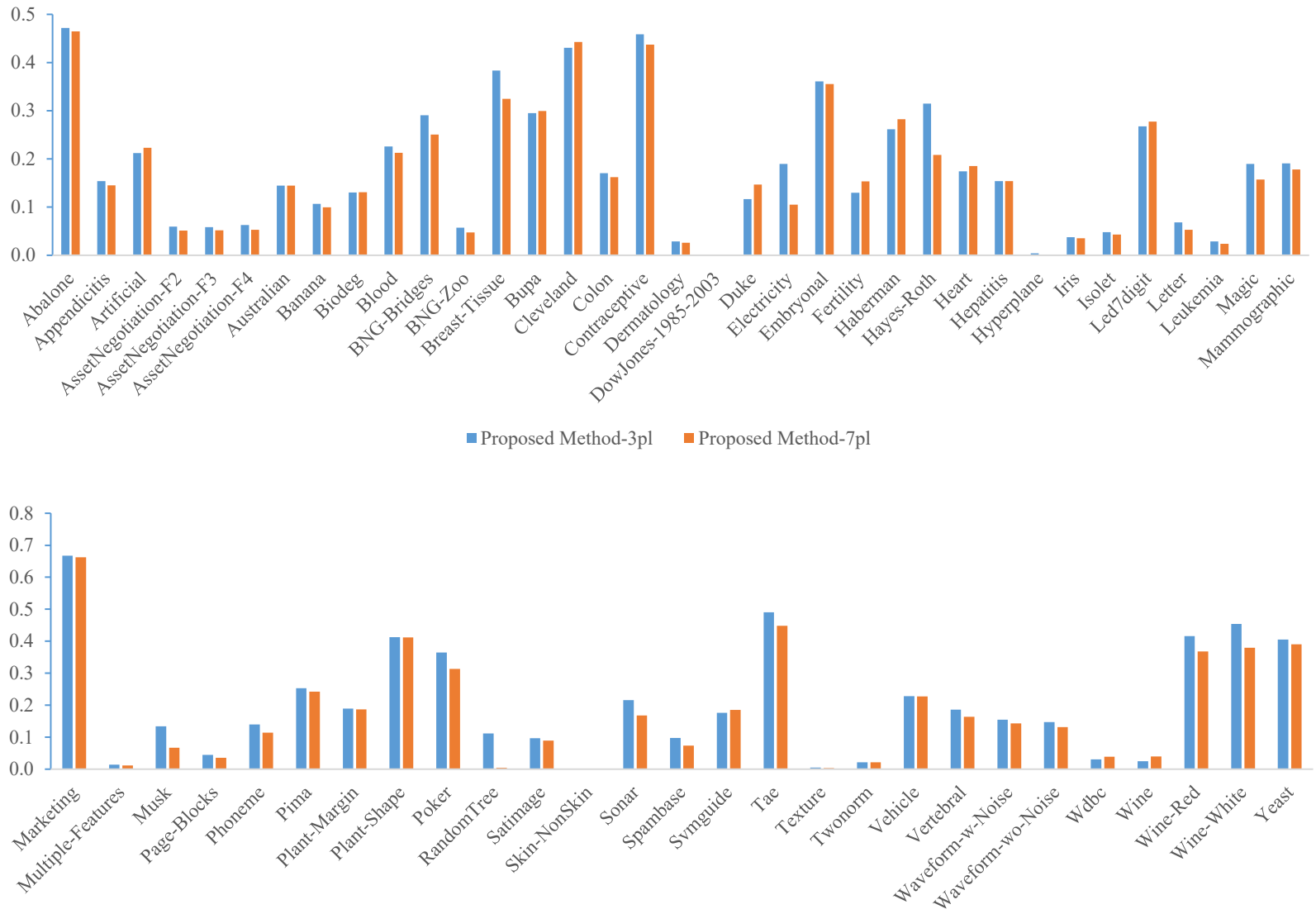
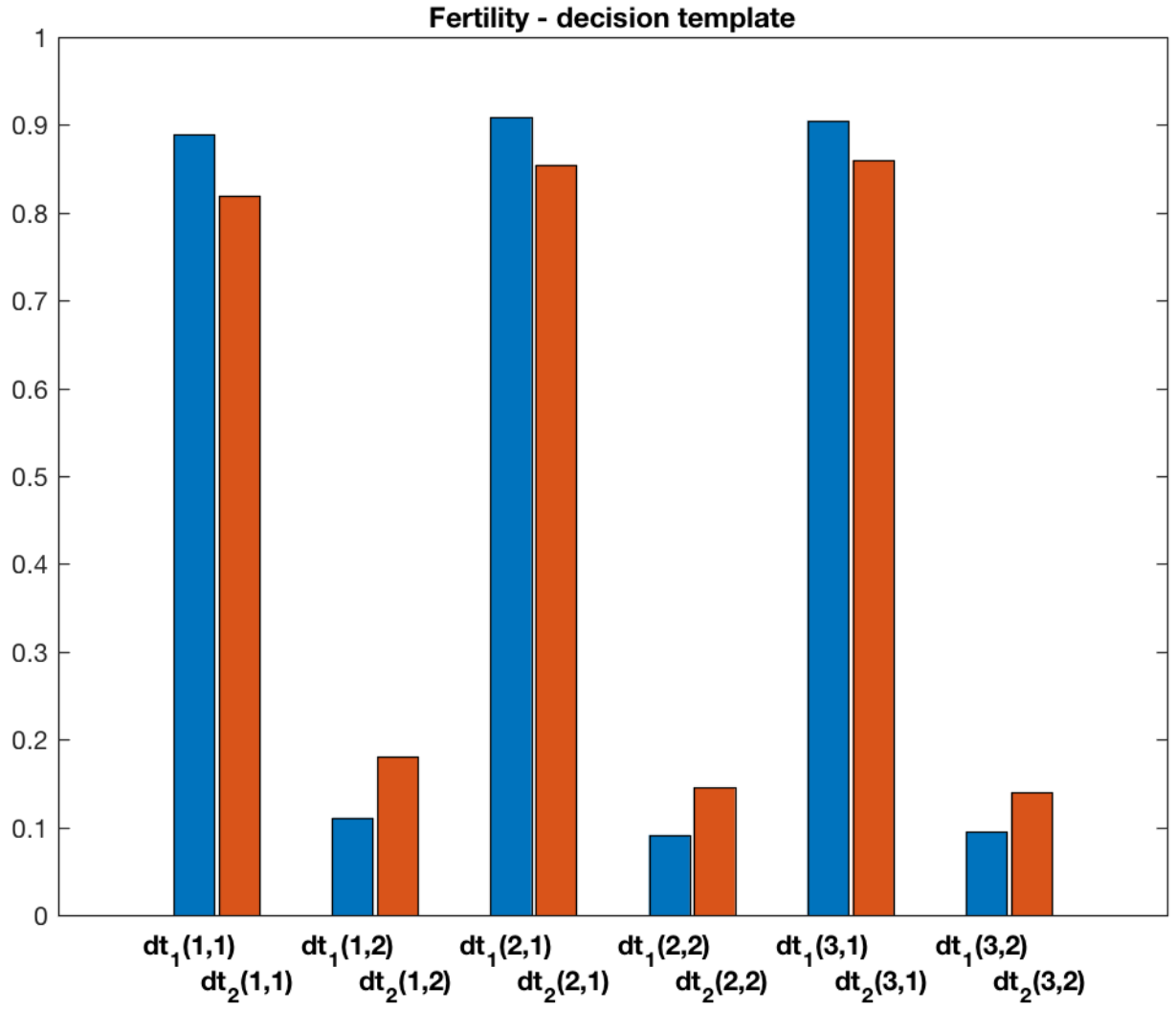


Fig.S3. The comparison between proposed methods using 3 and 7 base classifiers


Fertility - decision template
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
dt1(1,1)
dt2(1,1)
dt1(1,2)
dt2(1,2)
dt1(2,1)
dt2(2,1)
dt1(2,2)
dt2(2,2)
dt1(3,1)
dt2(3,1)
dt1(3,2)
dt2(3,2)

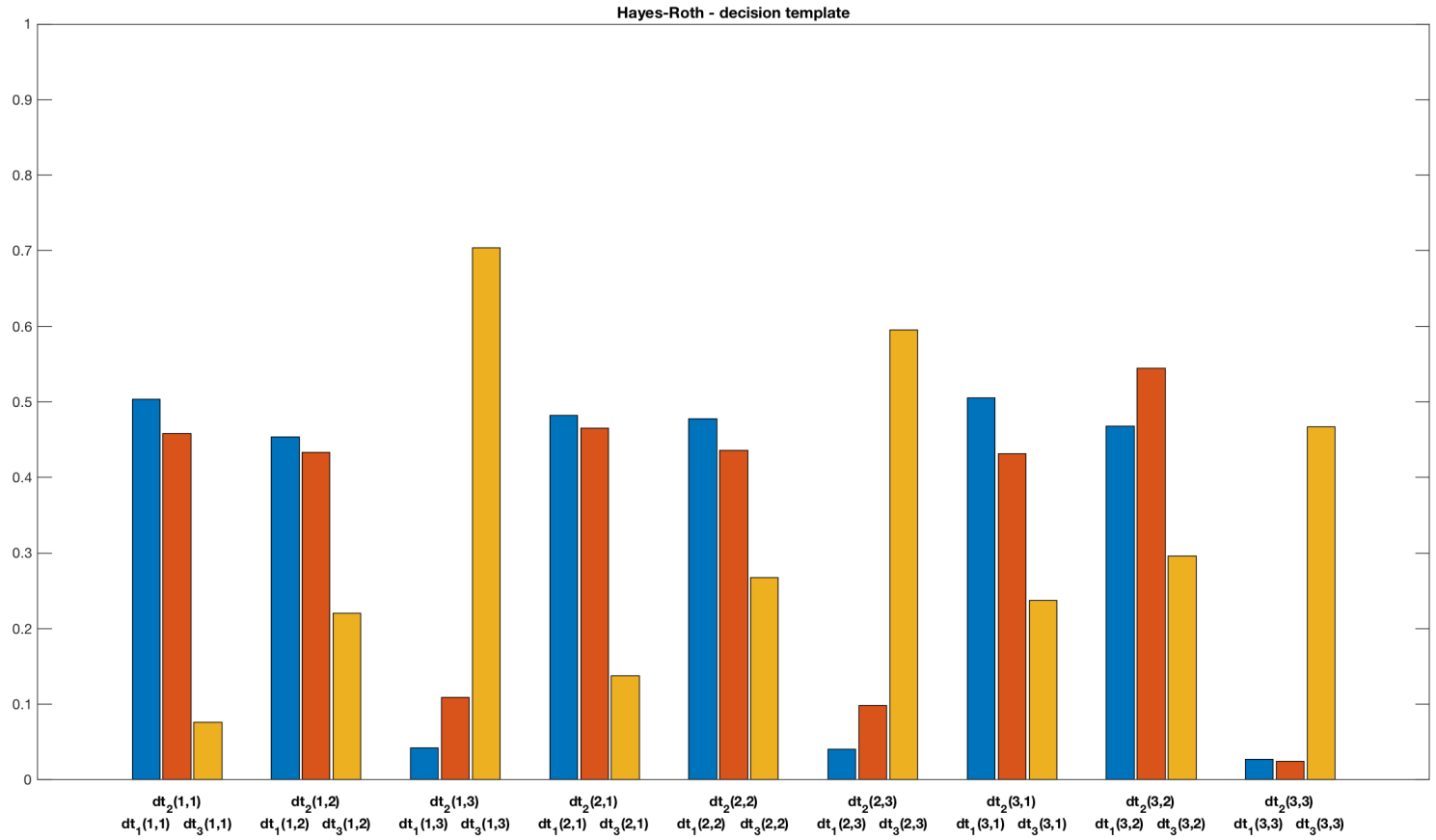


Fig.S4. The decision templates generated on Fertility and Hayes-Roth (using 3 base classifiers)