Pattern Anal Applic (2014) 17:97–104
DOI 10.1007/s10044-011-0260-8

THEORETICAL ADVANCES

# Consistency of randomized and finite sized decision tree ensembles

**Amir Ahmad · Sami M. Halawani ·**
**Ibrahim A. Albidewi**

Received: 5 May 2011 / Accepted: 29 November 2011 / Published online: 15 December 2011

**Abstract** Regression via classification (RvC) is a method in which a regression problem is converted into a classification problem. A discretization process is used to covert continuous target value to classes. The discretized data can be used with classifiers as a classification problem. In this paper, we use a discretization method, Extreme Randomized Discretization, in which bin boundaries are created randomly to create ensembles. We present an ensemble method for RvC problems. We show theoretically for a set of problems that if the number of bins is three, the proposed ensembles for RvC perform better than RvC with the equal-width discretization method. We use these results to show that infinite-sized ensembles, consisting of finite-sized decision trees, created by a pure randomized method (split points are created randomly), are not consistent. We also theoretically show, using a set of regression problems, that the performance of these ensembles is dependent on the size of member decision trees.

**Keywords** Ensembles · Decision trees · Discretization · Randomization · Consistency

A. Ahmad (✉) · S. M. Halawani
Faculty of Computing and Information Technology,
King Abdulaziz University, Rabigh, Saudi Arabia
e-mail: amirahmad01@gmail.com

I. A. Albidewi
Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia

## 1 Introduction

Ensembles are a combination of multiple base models [10, 18, 27]; the final classification depends on the combined outputs of individual models. Classifier ensembles have shown to produce better results than single models, provided the classifiers are *accurate* and *diverse* [18]. Ensembles perform best when base models are *unstable*–classifiers whose output undergoes significant changes in generalization with small changes in the training data; decision trees and neural networks are in this class.

Several different methods have been proposed to build decision tree ensembles. Randomization is introduced to build diverse decision trees. *Bagging* [6] and *Boosting* [14] introduce randomization by manipulating the training data supplied to each classifier. Breiman [8] proposed *Random Forests* (RF). To build a tree, it uses a bootstrap replica of the training sample; then during the tree-growing phase, at each node the optimal split is selected from a random subset of size $K$ of candidate features. *Extremely Randomized Trees* (ERT), proposed by Geurts et al. [16], combine the feature randomization of Random Subspaces [19] with a totally random selection of the cut-point. Random decision trees proposed by Fan et al. [12] use completely random splits points.

The excellent performance of ensembles based on randomization has led to many theoretical studies to understand their performance. Lin and Jeon [22] studied RF as a weighted layered nearest-neighbor classifier (a classifier that takes a (weighted) majority vote among the layered nearest neighbors of the observation $O$ ($O_i$ is called a layered nearest neighbor of $O$ if the rectangle defined by $O$ and $O_i$ as their opposing vertices does not contain any other data point.). Geurts et al. [16] show that "extremely and totally randomized tree ensembles hence provide an interpolation of any output variable which, for finite $M$ is piecewise constant (and, hence non-smooth), and for $M \longrightarrow \infty$ piecewise multi-linear and continuous", where $M$ is the size of the ensemble.

Consistency, i.e. the fact that the solution minimizing the empirical error does converge to the best possible error (Bayes error) when the number of examples goes to infinity, is an important property of classifiers. A classifier is called universally consistent if it is consistent for any distribution of ($X$, $Y$), where $X$ is input and $Y$ is output [17]. Generally in literature only "consistent" word is used instead of "universal consistent". For example, the title of the paper [2] "Adaboost is consistent". We will follow the same terminology in our paper.

Various theoretical studies have been carried out to understand the behavior of decision tree ensembles [2, 4]. Bartlett and Traskin [2] showed that AdaBoost is consistent under some conditions (if it is stopped after $n^{1-\epsilon}$ iterations for sample size $n$ and ε (0,1)). Biau et al. [4] showed that RF are not consistent. Biau et al. [4] also analyzed a simple random forest (considered by Breiman [7]). In this method, at each iteration of tree-growing phase, a leaf is chosen uniformly at random, and the split attribute and the split point are selected randomly. They showed that this random forest is consistent. In that proof, they assumed that $k \longrightarrow \infty$ and $k/n \longrightarrow 0$ as $k \longrightarrow \infty$, where $k$ is the number of leaves and $n$ is the number of the data points.

In this paper, we use the methodology suggested by Biau [3] to show that a regression function estimate is consistent. We assume that we are given a training sample $D_n = (X_1, Y_1), \ldots, (X_n, Y_n)$ for a regression function $r$, if the estimated regression function is $r_n(X)$ using the data $D_n$; the regression function estimate $r_n(X)$ is consistent if $E[r_n(X) - r(X)]^2 = 0$ as $n$ goes to $\infty$.

We will use the method of contradiction to show that infinite-sized ensembles, consisting of finite-sized decision trees, created by a pure randomized method (split points are created randomly), are not consistent. *We will show that there exist a regression function r(X) for which the regression function estimate $r_n(X)$ (calculated by infinite sized ensembles, consisting of finite-sized decision trees, created by a pure randomized method), does not fulfill the condition $E[r_n(X) - r(X)]^2 = 0$ as n goes to $\infty$.*

We will use a linear regression function, $y = x$, to show that infinite-sized ensembles, consisting of finite-sized decision trees, created by a pure randomized method (split points are created randomly), are not consistent.

In Sect. 2, we present an ensemble method for the Regression by Classification (RvC) method [20, 24–26] and discuss some of its properties. In Sect. 3, we present experiments to support our theoretical results. In Sect. 4, we use the results obtained in Sect. 2 to show the main result (consistency of ensembles). Section 5 contains the conclusion and future works.

## 2 Regression via classification (RvC)

In machine learning and data mining fields, supervised learning plays an important role [5, 23]. In a regression problem, the target values are continuous, whereas in a classification problem we have a discrete set of classes. The discretization process can be used to convert continuous target values into a discrete set of classes and then classification models are used to solve the classification problems [20, 24–26]. In other words, in a RvC problem, a regression problem is solved by converting it into a classification problem. This method employs any classifier on a copy of the data that has the target attribute discretized. The whole process of RvC comprised of two important stages:

1. The discretization of the numeric target variable in order to learn a classification model. There are different discretization methods e.g. equal-width, equal-frequency, etc [11].
2. The reverse process of transforming the class output of the classification model into a numeric prediction. We may use the mean value of the target variable for each interval as the final prediction.

### 2.1 Extreme randomized discretization (ERD)

Ahmad [1] presents a discretization method, Extreme Randomized Discretization (ERD), for creating ensembles of decision trees. In this method bin boundaries are created randomly. We will use the same method to create ensembles for RvC. Though the same method is used, the theoretical explanation and applications are different. In Ahmad [1], ERD was used to discretize attributes, whereas in this paper, ERD is used to discretize the target variable.

We propose that ERD is useful in creating ensembles for RvC. As discussed above, *In ERD, bin boundaries for the discretization are created randomly*. This may be used in stage 1 of RvC. As it creates diverse datasets, different classifiers can be created. Uncorrelated models are the keys to the success of any ensemble method [21]. *We show that the proposed ensembles for RvC perform better than single models with equal-width discretization for RvC, if the number of bins is 3*. In the next subsection, we will show our theoretical results.

### 2.2 Theoretical results

In this section, all the results are proved under following conditions:

1. the target value is uniformly distributed between 0 and $4L$.
2. Each regression function value is predicted once.
3. The classification error is 0.
4. The mean value of the target variable for each interval is the predicted value. As the target value is uniformly distributed, the center of the bin is the predicted value.
5. $y$ is the target variable.
6 $y_\mathrm{p}$ is the target value of the point $p$.
7. The number of models in an ensemble is $\infty$ and each model has different bin boundaries.
8. The final result of an ensemble is the mean of all the predictions (by single models).

As we have assumed that the classification error is 0, all the theoretical results are independent of the choice of the type of classifiers.

### 2.3 RvC with the equal-width discretization method with two bins

In this case, two equal sized bins are created; the bin boundary is at $2L$, all the points at the left side of the bin boundary will be predicted as $L$ (the mid point of the left bin), and all the points at the right side of the bin boundary will be predicted as $3L$ (the mid point of the right bin). Hence, the points with target values around $L$ and $3L$ will be predicted more accurately, whereas points at the 0, $2L$ and $4L$ will have more error. The mean square error (MSE) in this case is

$$(1/4L)\left(\int_0^{2L}(y-L)^2\mathrm{d}y+\int_{2L}^{4L}(y-3L)^2\mathrm{d}y\right)=0.33L^2. \quad (1)$$

For $4L = 100$, the MSE is 208.33.

### 2.4 RvC with ERD with two bins

ERD creates different bin boundaries, in different runs (we have assumed that no two bin boundaries are same in different runs. This can be achieved by selecting a new boundary from the boundaries that were not selected before). Hence, the predictions are different for different runs.

As given in Fig. 1, the bin boundary ($B_1$) can be anywhere between the minimum value (0) and the maximum value ($4L$) of the continuous target variable. If the target value we want to predict is $y_\mathrm{p}$ and if the bin boundary is at the left side of the $y_\mathrm{p}$, the predicted value is $(4L + B_1)/2$. If the bin boundary is at the right side of the $y_\mathrm{p}$, the predicted value is $(0 + B_1)/2$, as the final result is the mean value of all the predicted values. If the number of runs is $\infty$, The predicted value is

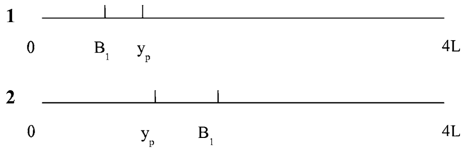

**Fig. 1** In the subfigure *1* (*top figure*) the bin boundary $B_1$ is at the *left side* of the point to be predicted, $y_\mathrm{p}$, whereas in the subfigure *2* (*bottom figure*), the bin boundary $B_1$ is at the *right side* of $y_\mathrm{p}$

$$(1/4L)\left(\int_0^{y_\mathrm{p}}(4L+B_1)/2\mathrm{d}B_1+\int_{y_\mathrm{p}}^{4L}(0+B_1)/2\mathrm{d}B_1\right) \quad (2)$$

The predicted value $= y_\mathrm{p}/2 + L$.
(The general formula;

$$\text{The predicted value} = y_\mathrm{p}/2 + (y_{min} + y_{max})/4. \quad (3)$$

where $y_\mathrm{min}$ is the minimum value of the target and $y_\mathrm{max}$ is the maximum value of the target).

We discuss some of the properties of this result.

For $y_\mathrm{p} = 0$ the predicted value is $L$.
For $y_\mathrm{p} = 2L$ the predicted value is $2L$.
For $y_\mathrm{p} = 4L$ the predicted value is $3L$.

This behavior is different from the *RvC with the equal-width method with two bins* as in this case target points near the mid point of the range are predicted more accurately. *One of the important points about the predicted value function is that it is a continuous function with respect to the target value*. In other words, the predicted values change smoothly with respect to the target value. This is similar to the Geurts's study [16] about the ERT, "extremely and totally randomized tree ensembles hence provide an interpolation of any output variable which for $M \longrightarrow \infty$ is continuous", where $M$ is the size of the ensemble. The MSE in this case is

$$(1/4L)\left(\int_0^{4L}(y-(y/2+L))^2\mathrm{d}y\right)=0.33L^2. \quad (4)$$

For $4L = 100$, the MSE is 208.3.

The MSE in this case is equal to the RvC with the equal width discretization method. Hence, there is no advantage of the proposed ensembles over single models with equal-width discretization, if the number of bins is 2.

### 2.5 RvC with the equal-width discretization method with three bins

In this case the target variable is divided into equal width bins. The size of these bins is $4L/3$, bin boundaries are $4L/3$ and $8L/3$, and mid points of these bins will be $4L/6$, $2L$ and

20*L*/6. Hence, the predicted values will be 4*L*/6, 2*L* and 20*L*/6 depending upon in which bin the point lies. The MSE for this case is

$$(1/4L)\left(\int_{0}^{4L/3}(y-4L/6)^2\mathrm{d}y+\int_{4L/3}^{8L/3}(y-2L)^2\mathrm{d}y\right.$$
$$\left.+\int_{8L/3}^{4L}(y-20L/6)^2\mathrm{d}y\right)=0.14L^2. \tag{5}$$

For 4*L* = 100, the MSE is 87.5.

### 2.6 RvC with ERD with three bins

In this case, there are two bin boundaries; $B_1$ and $B_2$. To calculate the predicted value, we will calculate the mean value of all the predicted values by different models. There are two cases (Fig. 2);

1. The bin boundary $B_1$ is left of the given point $y_p$. The two conditions are possible.

   - *The bin boundary $B_2$ is at the right of $B_1$*. In this case, for different runs $B_2$ is placed at different points between points $B_1$ and 4*L*. This case is similar to the two-bin case with the boundaries; $B_1$ and 4*L*. Hence, for a given $B_1$, the mean value is $y_\mathrm{p}/2 + (4L + B_1)/4$ (by using Eq. 3).
   - *The bin boundary $B_2$ is at the left of $B_1$*. In this case, the predicted values is the center of the rightmost bin. It is ($B_1$ + 4L)/2, this value is independent of $B_2$. Hence, the mean value for a given $B_1$ is ($B_1$ + 4*L*)/2.

   The probability of the first condition = (4*L* − $B_1$)/4*L*.
   The probability of the second condition = $B_1$/4*L*.
   As $B_1$ can take value from 0 to $y_p$. The mean value of this case (the bin boundary $B_1$ is left of the given point $y_p$) is,

$$(1/y_\mathrm{p})\left(\int_{0}^{y_\mathrm{p}}((y_\mathrm{p}/2+((4L+B_1)/4))((4L-B_1)/4L)\right.$$
$$\left.+((B_1+4L)/2)(B_1/4L))\mathrm{d}B_1\right) \tag{6}$$
$$=-y_\mathrm{p}^2/24L+3y_\mathrm{p}/4+L. \tag{7}$$

2. The bin boundary $B_1$ is at right of the given point $y_\mathrm{p}$. The two conditions are possible.

   - *The bin boundary $B_2$ is at the right of $B_1$*. In this condition, the predicted values is the center of the leftmost bin, which is $B_1$/2. Hence, the mean value, for a given $B_1$ is $B_1$/2.

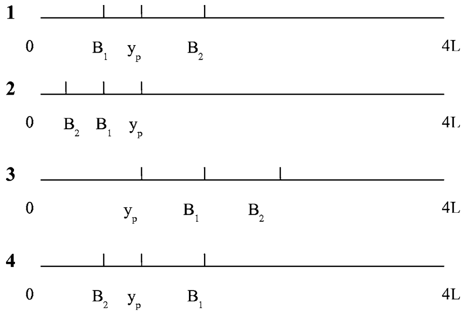

**Fig. 2** *1* The first bin boundary $B_1$ is at the *left side* of the $y_\mathrm{p}$. The second bin boundary $B_2$ is at the *right side* $B_1$. *2* The first bin boundary $B_1$ is at *left side* of the $y_\mathrm{p}$. The second bin boundary $B_2$ is at the *left side* $B_1$. *3* The first bin boundary $B_1$ is at the right side of the $y_\mathrm{p}$. The second bin boundary $B_2$ is at the right side $B_1$. *4* The first bin boundary $B_1$ is at the *right left* side of the $y_\mathrm{p}$. The second bin boundary $B_2$ is at the *left side* $B_1$

   - *The bin boundary $B_2$ is at the left of $B_1$*. In this condition, for different runs $B_2$ is placed at different points between points 0 and $B_1$. This case is similar to two-bin case with the range of the target variable between 0 and $B_1$. Hence, the mean value, for a given $B_1$ is, $y_p/2 + (0 + B_1)/4$

   The probability of the first condition = (4*L* − $B_1$)/4*L*.
   The probability of the second condition = $B_1$/4*L*.
   As $B_1$ can take value from $y_p$ to 4*L*. The mean value of this case (the bin boundary $B_1$ is at right of the given point $y_p$) is

$$1/(4L-y_\mathrm{p})\int_{y_\mathrm{p}}^{4L}(B_1/2)(4L-B_1)/4L+(y_\mathrm{p}/2$$
$$+B_1/4)(B_1/4L)\mathrm{d}B_1 \tag{8}$$
$$=-y_\mathrm{p}^2/24L+5y_\mathrm{p}/12+2L/3 \tag{9}$$

The mean value of all the cases = (The mean value of case 1) (The probability of case 1) + (The mean value of case 2) (The probability of case 2)

$$(-y_\mathrm{p}^2/24L+3y_\mathrm{p}/4+L)y_\mathrm{d}/4L+(y_\mathrm{p}^2/24L+5y_\mathrm{p}/12$$
$$+2L/3)(4L-y_\mathrm{p})/4L. \tag{10}$$
$$=y_\mathrm{p}/2+(2L/3+y_\mathrm{p}^2/8L-y_\mathrm{p}^3/48L^2). \tag{11}$$

For $y_\mathrm{p}$ = 0 the predicted value is 2*L*/3.
For $y_\mathrm{p}$ = 2*L* the predicted value is 2*L*.
For $y_\mathrm{p}$ = 4*L* the predicted value is 14*L*/3.
The MSE for this case is

$$1/4L\left(\int_{0}^{4L}(y-(y/2+2L/3+y^2/8L-y^3/48L^2))\mathrm{d}y\right). \tag{12}$$

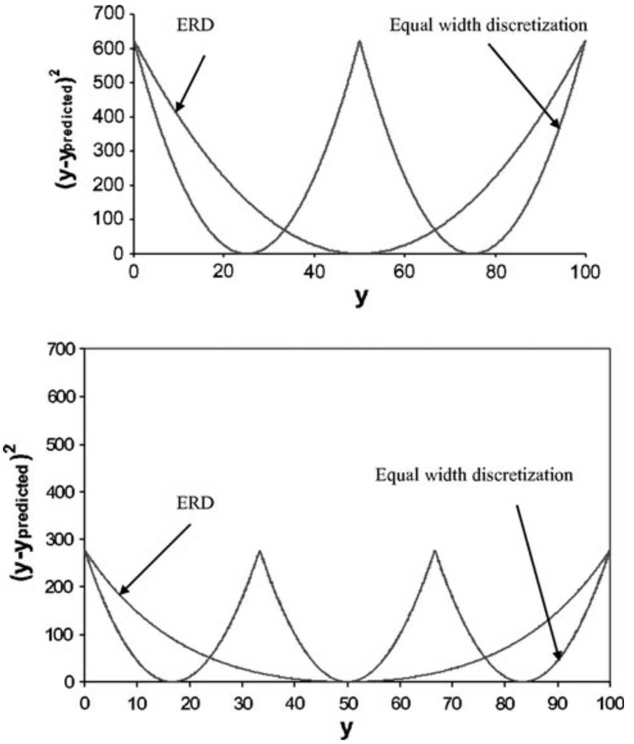

**Fig. 3** MSE (for two bins in *top figure*, and three bins in *bottom figure*) in different cases

MSE = 0.12 $L^2$ (for 4$L$ = 100, the MSE is 75) which is better than RvC with the equal-width method with three bins (MSE = 0.14$L^2$). *This proves that the ensembles with the proposed ensemble method performs better than single model with equal-width discretization for RvC, if the number of bins is 3.*

The MSE graph for two bins and three bins are presented in Fig. 3. One may follow the same kind of calculation to extend these results for bins more than 3. It will be cumbersome but straightforward calculation. As 3 bins improve the performance of ERD ensembles more as compared with single model with equal-width discretization, we may suggest intuitively that the more bins will give more performance advantage to the proposed ensemble method. We will verify this fact experimentally in the next section.

## 3 Experiments

We carried out experiments with $y = x$ function. This is a uniformly distributed function. We generated 10,000 points between $0 \leq x \leq 100$; 5,000 points were used for training and 5,000 points were used for testing. We used unpruned C4.5 decision tree (J48 decision tree of WEKA software [28]) as the classifier. The final result from a classifier was the mean value of the target variable ($y$ in this case) of all the points in the predicted bin. In the results, we found that the classification error was almost 0. As in these experiments all the conditions of our theoretical results were fulfilled, we expected that the experimental results should closely match the theoretical results. We carried out experiments with two bins and three bins. The size of the ensemble was set to 100. The experiments were conducted by using 5 × 2 cross-validation [9]. The average results are presented in the Table 1. Results suggest that there is an excellent match between experimental results and theoretical results for two-bin and three-bin cases. We also carried out experiments with 5, 10, and 20 bins. Results suggest that the ratio of the average MSE of RvC with equal-width discretization to the average MSE of RvC with ERD is increasing with the number of bins. This suggests that there is more performance advantage with ERD when we have large number of bins. This verifies our intuition that as we increase the number of bins the performance advantage increases for ERD ensembles.

### 3.1 Other datasets

In our theoretical studies, we assumed that the classification error is zero; however, this condition is not possible in most of the cases. Hence, we also carried out experiments with different popular regression datasets (these datasets are taken from www.liaad.up.pt/ltorgo/Regression/DataSets.html). In other words, the purpose of these experiments is

**Table 1** MSE in different cases

| The number of bins | MSE for RvC with equal-width bins (theoretical) | MSE for RvC with equal-width bins (experimental) (1) | MSE for RvC with ERD (theoretical) | MSE for RvC with ERD (experimental) (2) | (1)/(2) |
|---|---|---|---|---|---|
| 2 | 208.3 | 209.1 (2.2) | 208.3 | 210.3 (3.1) | 0.99 |
| 3 | 87.5 | 90.3 (1.7) | 75 | 77.3 (1.5) | 1.17 |
| 5 | – | 33.1 (0.8) | – | 18.6 (0.4) | 1.78 |
| 10 | – | 8.3 (0.2) | – | 2.6 (0.1) | 3.19 |
| 20 | – | 2.9 (0.1) | – | 0.4 (0.1) | 7.25 |

For experimental results, the average results are given, s.d. is given in bracket. The final column suggests that the performance advantage of the proposed ensemble method improves with larger number of bins

**Table 2** Details of datasets used in the experiments

| Name | Number of attributes | Number of data points |
|---|---|---|
| Abalone | 8 | 4,177 |
| Bank8FM | 8 | 8,192 |
| Cart | 10 | 40,768 |
| Delta_Ailerons | 6 | 7,129 |
| Delta_Elevator | 6 | 9,517 |
| House (8L) | 8 | 22,784 |
| House (16H) | 16 | 22,784 |
| Housing (Boston) | 13 | 506 |
| Kin8nm | 8 | 8,192 |
| Puma8NH | 8 | 8,192 |
| Puma32H | 32 | 8,192 |

to study the performance of the proposed ensemble method when the classification error is not zero. The information about the datasets is given in Table 2. The size of the ensembles was set to 100 for all the experiments. The number of bins was set to 10 for RvC methods. In RvC methods, we may use any type of classifier for the classification step. However, in this paper, we are concentrating on the theoretical study of decision tree ensembles. Hence, we carried out all experiments with decision trees (unpruned C4.5 decision tree (J48 decision tree of WEKA software [28]). The experiments were conducted by using 5 × 2 cross-validation [9].

Results (Average Root MSE), presented in Table 3 suggest that the proposed ensemble method performs consistently better than a single model (RvC with the equal width discretization method). This shows the effectiveness of our approach.

The number of bins is an important variable, as a small number of bins lead to the better classification; however, the values represented by bins will be less representative of the points in the bins. If the number of bins is large, the number of points in each bin will be small; this leads to the poor classification accuracy. However, the values represented by bins will be more representative of the points in the bins. One may use cross validation to find out the number of bins for the best regression results.

## 4 Consistency of ensembles of randomized trees

In this section, we will use the results obtained in the last section to show that

1. An infinite-sized ensemble, consisting of finite-sized decision trees, created by a pure randomized method (split points are created randomly), is not consistent.

**Table 3** Experimental results for different methods for different datasets

| Name of dataset | MSE for RvC with ERD (1) | MSE for RvC with equal-width bins (2) | Ratio of RMSE (2)/(1) |
|---|---|---|---|
| Abalone | 2.24 (0.05) | 2.89 (0.08) | 1.29 |
| Bank8FM | $3.61\ (0.11) \times 10^{-2}$ | $5.31(0.17) \times 10^{-2}$ | 1.47 |
| Cart | 1.06 (0.02) | 1.46 (0.06) | 1.37 |
| Delta_Ailerons | $1.72\ (0.03) \times 10^{-4}$ | $2.75\ (0.05) \times 10^{-4}$ | 1.59 |
| Delta_Elevator | $1.52\ (0.02) \times 10^{-3}$ | $1.91\ (0.03) \times 10^{-3}$ | 1.25 |
| House (8L) | $3.12\ (0.05) \times 10^{4}$ | $4.12\ (0.08) \times 10^{4}$ | 1.32 |
| House (16H) | $3.51\ (0.07) \times 10^{4}$ | $4.62\ (0.10) \times 10^{4}$ | 1.31 |
| Housing (Boston) | 3.98 (0.09) | 5.23 (0.12) | 1.31 |
| Kin8nm | 0.17 (0.01) | 0.24 (0.02) | 1.41 |
| Puma8NH | 3.28 (0.14) | 4.50 (0.16) | 1.37 |
| Puma32H | $8.21\ (0.43) \times 10^{-3}$ | $1.20\ (0.04) \times 10^{-2}$ | 1.46 |

The average results for RMSE (Root Mean Square Error) are presented. s.d. is given in the bracket. The last column presents the ratio of RMSE of RvC with equal-width bins and RMSE of RvC with ERD. The value greater than 1.0 suggests that RvC with ERD performed better

2. We also show theoretically that the performance of these ensembles are dependent on the size of individual decision trees.

### 4.1 Consistency results

We will show that infinite-sized ensembles of finite-sized decision trees are not consistent for $y = x$ problem. Hence, they are not universal consistent. While growing a decision tree, at each node the available data points are split into two partitions on the basis of a given split criterion, e.g. information gain, information gain ratio, etc. However, there are various ensemble methods that create individual decision trees with random split points; in other words, split points are selected randomly [13, 16]. If we are creating trees with the data for the problem $y = x$, using the method suggested in ERT [16] and Random trees [13], we randomly select the split point from the data points representing $x$ at the node. If the number of data points is $\infty$, any point between 0 and $4L$ can be selected as the split point. Let us assume the split point is $x_s$; then let all points less then $x_s$ be part of one partition and all points greater than or equal to $x_s$ form the second partition. If we want to create a decision tree with only one split point then Fig. 4 will be our tree (with one parent and two child nodes) and the average value of $y$ of all the points in a partition will be the prediction of that partition. We can say this is similar to selecting a split point $y_s$ ($y_s = x_s$ because $y = x$) randomly

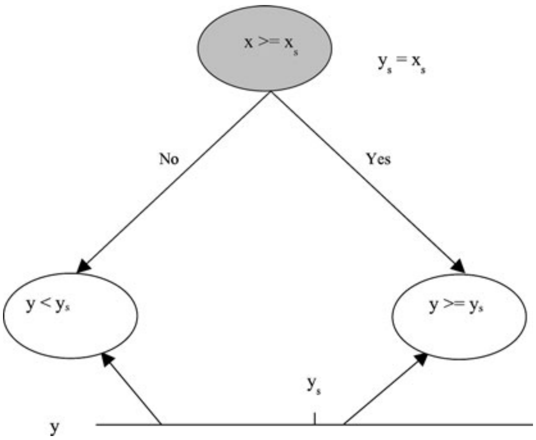

**Fig. 4** A decision tree with two leaves

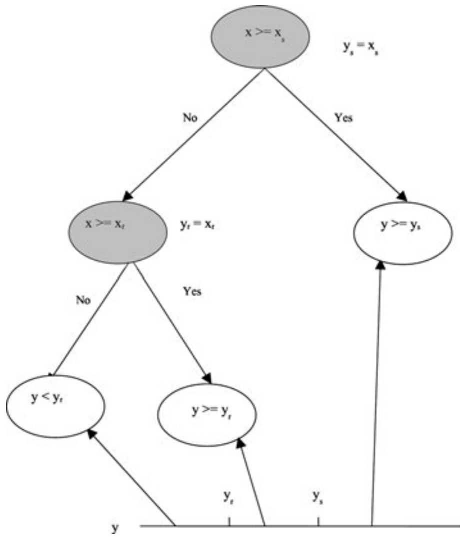

**Fig. 5** A decision tree with three leaves

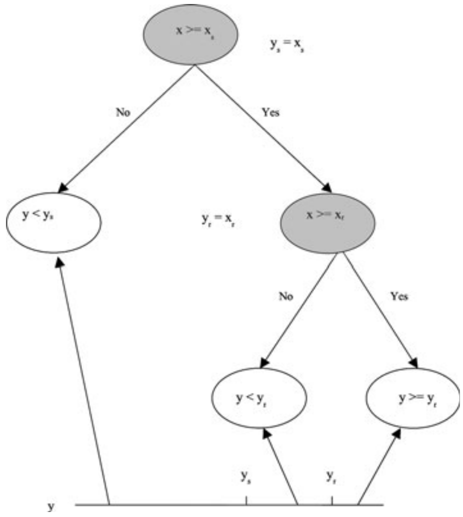

**Fig. 6** A decision tree with three leaves

from the $y$ values and calculating the average of the values of all points less then $y_s$ be part of one partition and all points greater than or equal to $y_s$ form the next partition. This is similar to RvC in which discretization is created by ERD (as discussed in Sect. 2,). Hence, if we have infinite decision trees created by this method with different split points, it is exactly the same as the RvC with ERD having two bins. If we want to predict a $y$ value for a given $x$, for each tree, the result will depend upon the value of the $x$ is less than $x_s$ or, greater than or equal to $x_s$ or we can say the value is predicted depending upon $y_s$. Hence, the final result will be the same as RvC with ERD having two bins. If we all predicting all points with $x$ values between 0 and $4L$ the MSE will be $0.14L^2$. Hence, $E[r_n(X) - r(X)]^2 \neq 0$ as $n$ goes to $\infty$. This shows that ensembles consisting of decision trees with one split point (created randomly) are not consistent.

If we create trees with two split points (created randomly), $x_s$ and $x_r$, we will have three leaves; if $y_s = x_s$ and $y_r = x_r$, three leaves will consist of points with, $y < y_r$, $y_r \leq y < y_s$, $y \geq y_s$ or $y < y_s$, $y_s \leq y < y_r$, $y \geq y_r$ (Figs. 5 and 6). If the values of leaves are taken as the average of the point present in leaves, this is similar to RvC with ERD with three bins. Hence, if we predict all points with $x$ values between 0 and $4L$ the MSE will be $0.12L^2$. Hence, $E[r_n(X) - r(X)]^2 \neq 0$ as $n$ goes to $\infty$. This proves that ensembles consisting of decision trees with two split points (created randomly) are not consistent.

We did not prove the results for RvC, with ERD with more than three bins. However, we note that for the MSE calculation of RvC with ERD with three bins, we used the result of RvC with ERD with two bins. This suggests that for the higher number of bins, the MSE results of lower number of bins are used. As the MSEs with two bins and three bins are not zero, we can say MSE derived with these results for four bins will not be zero. This argument can be extended further to suggest that the MSE for RvC with ERD with finite number of bins ($m$ bins) will not be zero. Experiment results suggest the same. As this is similar to ensembles with trees with finite random splits ($m - 1$ splits), we can say that $E[r_n(X) - r(X)]^2 \neq 0$ as $n$ goes to $\infty$. This proves that ensembles consisting of decision trees with finite number of split points (created randomly) are not consistent.

## 4.2 The effect of the size of the tree

Generally trees with random splits in an ensemble are not pruned [8, 16]. In other words, large-sized trees are preferred. Models have been suggested that show the importance of size of the ensemble [15]. Our model suggests that for the regression function $y = x$, infinite-sized ensembles with decision trees with three leaves are more accurate than infinite-sized ensembles with decision trees with two leaves. This shows the importance of large-sized trees in ensembles. In other words, to calculate the performance of the ensembles with trees with random splits along with the size of the ensembles, the size of the trees should also be considered. This may look obvious as an ensemble of trees behaves as a large tree; hence an ensemble of large trees has better representation power. However, in the present paper, we have considered that the size of ensembles is $\infty$ and all trees in an ensemble are uncorrelated. According to the study by Guerts et al. [16] for infinite-sized ensembles, "extremely and totally randomized tree ensembles hence provide an interpolation of any output variable". This says nothing about the size of the member trees. However, our study suggests that even with infinite-sized ensembles, the sizes of individual trees should be considered to calculate the performance of ensembles.

## 5 Conclusion

In the present paper, we presented an ensemble method for RvC problem. We showed theoretically for a set of a problems that this ensemble method performs better than a single model for RvC with equal-width discretization, when the number of bins in 3. We used the similarity of the proposed ensemble method and ensembles of randomized trees to show that ensembles of infinite size, consisting of finite-sized decision trees, created by a pure randomized method (split points are created randomly), are not consistent. We also showed that the even with the infinite-sized ensembles, the size of member trees is an important factor for the performance of ensembles. In this paper, we proved the results for a set of problems; however, in future, we will try to prove these results for a wide range of problems.

## References

1. Ahmad A (2010) Data transformation for decision tree ensembles, Ph.D. thesis, School of Computer Science, University of Manchester, Manchester
2. Bartlett PL, Traskin M (2007) Adaboost is consistent. J Mach Learn Res 8:2347–2368
3. Biau G (2010) Analysis of a random forests model, Technical report, Universit Paris
4. Gerard Biau, Luc Devroye (2008) Consistency of random forests and other averaging classifiers. J Mach Learning Res 9:2015–2033
5. Bishop CM (2008) Pattern recognition and machine learning. Springer-Verlag, New York
6. Breiman L (1996) Bagging predictors. Mach Learn 24(2):123–140
7. Breiman L (2000) Some infinite theory for predictor ensembles. Technical Report 577, Statistics Department, University of California, Berkeley
8. Breiman L (2001) Random forests. Mach Learn 45(1):5–32
9. Dietterich TG (1998) Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput 10:1895–1923
10. Dietterich TG (2000) Ensemble methods in machine learning. Proc Conf Multiple Classifier Syst 1857:1–15
11. Dougherty J, Kahavi R, Sahami M (1995) Supervised and unsupervised discretization of continuous features. In: Proceedings of the twelth international conference on machine learning, California
12. Fan W, McCloskey J, Yu PS (2006) A general framework for accurate and fast regression by data summarization in random decision trees. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 136–146
13. Fan W, Wang H, Yu PS, Ma S (2003) Is random model better? on its accuracy and efficiency. In: Proceedings of third IEEE international conference on data mining (ICDM2003), pp 51–58
14. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 55(1):119–139
15. Fumera G, Roli F, Serrau A (2008) A theoretical analysis of bagging as a linear combination of classifiers. IEEE Transact Pattern Anal Mach Intell 30(7):1293–1299
16. Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. Mach Learn 63(1):3–42
17. Gyorfi L, Lugosi G, Devroye L (1996) A probabilistic theory of pattern recognition, Springer, Berlin
18. Hansen LK, Salamon P (1990) Neural network ensembles. IEEE Transact Pattern Anal Mach Intell 12(10):993–1001
19. Ho TK (1998) The Random subspace method for constructing decision forests. IEEE Transact Pattern Anal Mach Intell 20(8):832–844
20. Indurkhya N, Weiss SM (2001) Solving regression problems with rule-based ensemble classifiers, ACM international conference knowledge discovery and data mining (KDD01), pp 287–292
21. Kuncheva LI (2004) Combining pattern classifiers: methods and algorithms. Wiley-Interscience, Hoboken
22. Lin Y, Jeon y (2006) Random forests and adaptive Neighbors. J Am Stat Assoc474 (101): 578–590
23. Mitchell TM (1997) Machine learning. McGraw-Hill, New York
24. Torgo L, Gama J (1996) Regression by classification. Advances in Artificial Intelligence, pp 51–60
25. Torgo L, Gama J (1997) Regression using classification algorithms. Intell Data Anal 4(1):275–292
26. Torgo L, Gama J (1997) Search-based Class Discretization, Proceedings of the 9th European Conference on Machine Learning, pp 266–273
27. Tumer K, Ghosh J (1996) Correlation and error reduction in ensemble classifiers. Connect Sci 8(3):385–404
28. Witten IH, Frank E (2005) Data mining: practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco