Expert Systems with Applications 39 (2012) 6396–6401

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Novel ensemble methods for regression via classification problems

Amir Ahmad [a,*], Sami M. Halawani [a], Ibrahim A. Albidewi [b]

[a] *Faculty of Computing and Information Technology, King Abdulaziz University, Rabigh 21911, Saudi Arabia*
[b] *Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia*

ARTICLE INFO

Keywords:
Ensembles
Regression
Classification
Decision trees

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 (ERD), in which bin boundaries are created randomly to create ensembles. We present two ensemble methods for RvC problems. We show theoretically that the proposed ensembles for RvC perform better than RvC with the equal-width discretization method. We also show the superiority of the proposed ensemble methods experimentally. Experimental results suggest that the proposed ensembles perform competitively to the method developed specifically for regression problems.

1. Introduction

In machine learning and data mining fields, supervised learning plays an important role (Bishop, 2008; Mitchell, 1997). In a regression problem, the target values are continuous, whereas in the classification problem we have discrete set of classes. The other difference is that regression values have a natural ordering, whereas for the classification the class values are unordered (Bishop, 2008; Mitchell, 1997). Regression models are not easily understood by domain experts, and thus provide little help in understanding the problem, whereas classification models are more comprehensible, but not very useful, when the target values are continuous. There are some learning schemes, like naive Bayes, which are very successful as classification techniques, however, they are difficult to use as regression schemes. Decision trees (Breiman, Friedman, Olshen, & Stone, 1984; Quinlan, 1993), neural networks (Bishop, 2008; Mitchell, 1997), naive Bayes (Bishop, 2008; Mitchell, 1997), support vector machines (Burges, 1998; Vapnik, 1998), etc. are quite popular for classification problems, whereas regression trees (Breiman et al., 1984), neural networks (Bishop, 2008; Mitchell, 1997), support vector machines (Burges, 1998; Vapnik, 1998), etc. are used for regression problems.

Discretization (Dougherty, Kahavi, & Sahami, 1995) is a process that divides continuous numeric values into a set of intervals (bins) that can be considered as categorical values. Dougherty et al. (1995) define three axes upon which discretization methods can be classified; global vs. local, supervised vs. unsupervised and static vs. dynamic. Supervised methods use the information of class labels, whereas, unsupervised methods do not. Local methods as the one used in C4.5 decision trees, produce partitions that are applied to localized regions of the instance space. Global methods are applied to the entire dataset. In static methods, attributes are discretized independently of each other, whereas, dynamic methods take into account the interdependencies between them. Equal-width intervals, equal-frequency intervals and unsupervised Monothetic Contrast Criteria (MCC) (Van de Merckt, 1993) are unsupervised methods. Discretization methods based on entropy (supervised MCC Van de Merckt, 1993, entropy minimization discretization Fayyad & Irani, 1993, D-2 Catlett, 1991), 1DR (Holte, 1993), adaptive quantizers (Chan, Batur, & Srinivasan, 1991), Vector Quantization (Kohonen, 1989), etc. are supervised methods. Equal-width intervals and equal-frequency intervals are global methods. The discretization used in the C4.5 decision tree growing phase and Vector Quantization are local methods. All these methods are static methods. Dynamic methods are a promising area of research. As these methods are able to capture interdependencies between attributes, they may improve the accuracy of decision rules (Kwedlo & Kretowski, 1999). Kwedlo and Kretowski (1999) show that static methods (that do not capture interdependencies) run the risk of missing information necessary for correct classification.

Researchers (Bibi, Tsoumakas, Stamelos, & Vlahavas, 2008; Indurkhya & Weiss, 2001; Torgo & Gama, 1997, 1997, 1999) suggest that 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. 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

* Corresponding author. Tel.: +966 551346386; fax: +966 22434030.
E-mail addresses: amirahmad01@gmail.com (A. Ahmad), dr.halawani@gmail.com (S.M. Halawani), Ialbidewi@kau.edu.sa (I.A. Albidewi).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.12.029

discretized. The whole process of RvC comprises 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. (Dougherty et al., 1995).
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.

Ensembles are a combination of multiple base models (Dietterich, 2000; Hansen & Salamon, 1990; Tumer & Ghosh, 1996); the final classification or regression results depends on the combined outputs of individual models. Ensembles have shown to produce better results than single models, provided the models are *accurate* and *diverse* (Hansen & Salamon, 1990).

Neural networks and decision tree ensembles are quite popular. Bagging Breiman (1996) and Boosting methods (Freund & Schapire, 1997) are general and can be used with any classifier. Several different methods have been proposed to build decision tree ensembles. Breiman (2001) proposed *Random Forests*. 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. Geurts, Ernst, and Wehenkel (2006) proposd *Extremely Randomized Trees*, which combines the feature randomization of Random Subspaces with a totally random selection of the cut-point. Random decision trees (Fan, McCloskey, & Yu, 2006; Fan, Wang, & Yu, 2003) proposed by Fan et al. use completely random splits points. These decision tree ensemble methods have shown excellent performance for the regression problems.

In spite of the excellent performance of pure randomization-based ensemble methods, there is little theoretical explanation about their performance (Rodriguez, Kuncheva, & Alonso, 2006). The success of an ensemble method depends on its ability to create uncorrelated individual models (Kuncheva, 2004). However, it is very difficult to predict exactly the performance of these ensembles. Our main contributions in this paper are;

1. We propose two novel ensemble methods for RvC problems.
2. We show theoretically that for a set of problem, it is possible to predict the performance of the proposed ensembles. Our theoretical predictions match experimental results.

The paper is organized as follows. In Section 2, we present the proposed ensemble methods for RvC and discuss some of its properties. In Section 3, we present our experimental results. Section 4 contains the conclusion.

2. The proposed method

In this section, we discuss our proposed ensemble methods for RvC. We also show that the one of proposed ensembles for RvC performs better than single model with equal-width discretization for RvC, if the number of bins is 3. Whereas, the second proposed ensemble method performs better than the single model with equal-width discretization for RvC, if the number of bins is 2.

2.1. Extreme Randomized Discretization (ERD)

Ahmad (2010) presented a discretization method, Extreme Randomized Discretization (ERD), for creating ensembles of decision trees. In this method bin boundaries are created randomly. This method was used to discretize attributes. 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 (2010), 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 (Kuncheva, 2004). 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_p is the target value of the point p.
(7) The number of models in an ensemble is ∞ 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

$$(1/4L)\left(\int_0^{2L}(y-L)^2\,dy+\int_{2L}^{4L}(y-3L)^2\,dy\right)=0.33L^2. \tag{1}$$

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_p and if the bin boundary is at the left side of the y_p, the predicted value is $(4L + B_1)/2$. If the bin boundary is at the right side of the y_p, the predicted value is $(0 + B_1)/2$. As the final result is the mean value of all the predictions. If the number of runs is ∞,

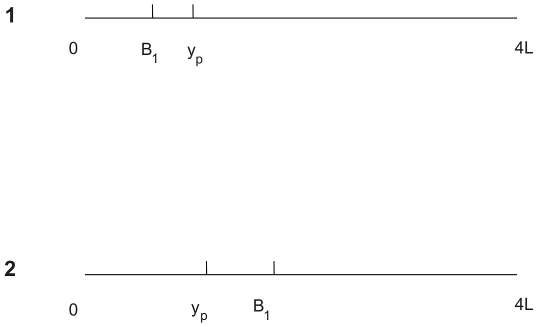

Fig. 1. In the subfigure 1(top figure) the bin boundary B_1 is at the left side of the point to be predicted, y_p, whereas in the subfigure 2 (bottom figure), the bin boundary B_1 is at the right side of y_p.

The predicted value is

$$1/4L\left(\int_0^{y_p}(4L+B_1)/2\ dB_1+\int_{y_p}^{4L}(0+B_1)/2\ dB_1\right) \tag{2}$$

The predicted value = $y_p/2 + L$.

The general formula;

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

where y_{min} is the minimum value of the target and y_{max} is the maximum value of the target.

We discuss some of the properties of this result.

For $y_p = 0$ the predicted value is L.
For $y_p = 2L$ the predicted value is $2L$.
For $y_p = 4L$ 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 (Geurts et al., 2006) about the ERT, "extremely and totally randomized tree ensembles hence provide an interpolation of any output variable which for $M \to \infty$ is continuous", where M is the size of the ensemble.

The MSE in this case is

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

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

$$(1/4L)\left(\int_0^{4L/3}(y-4L/6)^2\ dy+\int_{4L/3}^{8L/3}(y-2L)^2\ dy\right.$$
$$\left.+\int_{8L/3}^{4L}(y-20L/6)^2\ dy\right)=0.14L^2 \tag{5}$$

For $4L = 100$, 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 $4L$. This case is similar to two bins case with the boundaries; B_1 and $4L$. Hence, for a given B_1, the mean value is $y_p/2 + (4L + B_1)/4$ (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 $(B_1 + 4L)/2$.

     The probability of the first condition = $(4L - B_1)/4L$.
     The probability of the second condition = $B_1/4L$.

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

$$1/y_p\left(\int_0^{y_p}((y_p/2+((4L+B_1)/4))((4L-B_1)/4L)+((B_1+4L)/2)(B_1/4L))\ dB_1\right) \tag{6}$$

$$=-y_p^2/24L+3y_p/4+L. \tag{7}$$

2. The bin boundary B_1 is at right 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 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 $B_1/2$.
   - 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 bins case with the range of the target variable between 0 and B_1. Hence, the mean value, for a given B_1 is, $y_p/2 +\ (0 + B_1)/4$

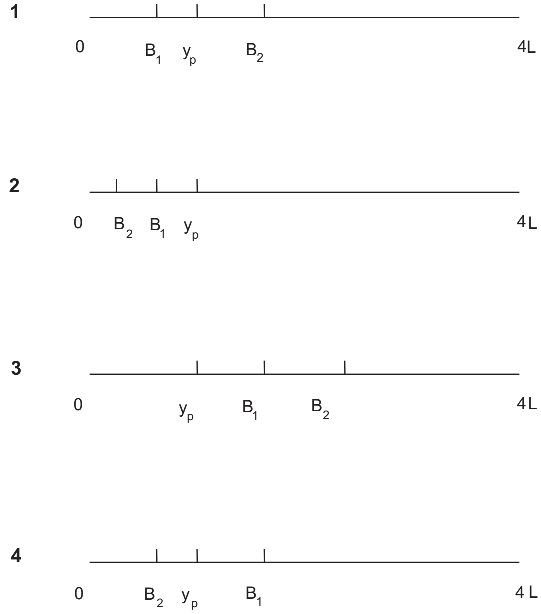

Fig. 2. (1) The first bin boundary B_1 is at the left side of the y_p. The second bin boundary B_2 is at the right side B_1. (2) The first bin boundary B_1 is at left side of the y_p. The second bin boundary B_2 is at the left side B_1. (3) The first bin boundary B_1 is at the right side of the y_p. 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_p. The second bin boundary B_2 is at the left side B_1.

The probability of the first condition = $(4L - B_1)/4L$.
The probability of the second condition = $B_1/4L$.

As B_1 can take value from y_p to $4L$. The mean value of this case (the bin boundary B_1 is at right of the given point y_p) is

$$1/(4L - y_p)\int_{y_p}^{4L}(B_1/2)(4L - B_1)/4L + (y_p/2 + B_1/4) \times (B_1/4L)\ dB_1 \tag{8}$$

$$= -y_p^2/24L + 5y_p/12 + 2L/3 \tag{9}$$

The predicted value = 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)

$$\left(-y_p^2/24L + 3y_p/4 + L\right)y_d/4L + \left(y_p^2/24L + 5y_p/12 + 2L/3\right)(4L - y_p)/4L. \tag{10}$$

$$\text{The predicted value} = y_p/2 + \left(2L/3 + y_p^2/8L - y_p^3/48L^2\right). \tag{11}$$

For $y_p = 0$ the predicted value is $2L/3$.
For $y_p = 2L$ the predicted value is $2L$.
For $y_p = 4L$ the predicted value is $14L/3$.

The MSE for this case is

$$1/4L\left(\int_0^{4L}(y - (y/2 + 2L/3 + y^2/8L - y^3/48L^2))\ dy\right). \tag{12}$$

The MSE is $0.12L^2$ (by using simpson's rule (Burden, 2010), for $4L = 100$, the MSE is 75) which is better than RvC with the equal-width method with three bins (MSE = $0.14L^2$). This proves that the ensembles with the proposed ensemble method perform better than single model with equal-width discretization for RvC, if the number of bins is 3.

The same calculation can be followed 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 to single model with equal-width discretization, we may suggest intuitively that the more bins will give more performance advantage to the proposed ensemble method.

3. Weighted ERD method for two bins

In the RvC method under study, a classifier predicts a bin and the mid point of the bin is the final prediction. In the ERD ensemble method, the final result is the mean of all the values given by different classifiers. In other words, all the classifiers have same weight in the final result. In RvC, we want a small bin size (however, very small size may lead to poor classification results) beacuse in this case the mid point of the bin will be better representative of the points in the bin. Hence, we have higher probability of accurate result if the bin size is small. Hence, we may say that the size of the bin is related with the probability of getting the accurate result. As each classifier predicts a bin, in this method, we assign 1/The size of the predicted bin, as the weight of the classifier output. Hence, we multiply each predicted value with its weight. All the results are then added. The sum is divided with sum of all the weights to get the final result. If the bin boundary B_1 is at the left of the point y_p (Fig. 1), the predicted value is $(B_1 + 4L)/2$ and the weight is inverse of the width of the bin which is $(4L - B_1)$. If the bin boundary B_1 is at the right of the point y_p (Fig. 1), the predicted value is $B_1/2$ and the weight is inverse of the width of the bin which is B_1. The predicted value for a given value y_p

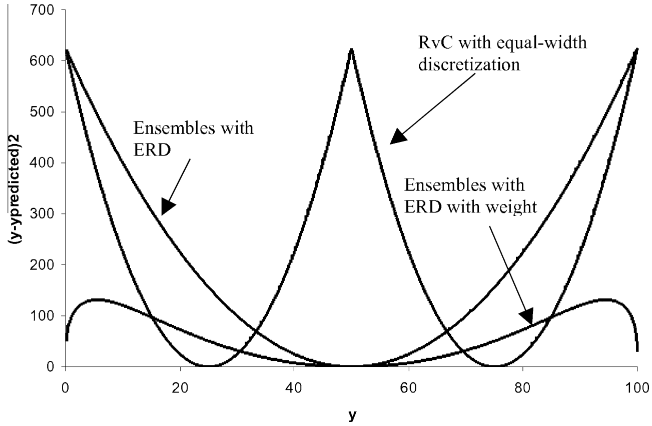

Fig. 3. MSE, for the $y = x$ problem, for different RvC methods with two bins.

$$= \frac{\int_0^{y_p}(((B_1 + 4L)/2)/(4L - B_1))\ dB_1 + \int_{y_p}^{4L}((B_1/2)/B_1)\ dB_1}{\int_0^{y_p}(1/(4L - B_1))\ dB_1 + \int_{y_p}^{4L}(1/B_1)\ dB_1} \tag{13}$$

$$= \frac{2L - y_p + 4L\log(4L/(4L - y_p))}{\log(4L/(4L - y_p)) + \log(4L/y_p)} \tag{14}$$

At $y_p = 2L$ the predicted value is $2L$, which is similar to ERD ensembles.

At $y_p = 0$ and $y_p = 4L$, this is undefined, however at points, other than $2L$, this gives better prediction as compared to ensembles without weight. MSE for different methods are given Fig. 3.

To calculate MSE, we used simpson's rule of numerical integration (Burden, 2010) to caculate the integral and for $4L = 100$,

$$1/4L\left(\int_0^{4L}\left(y - \frac{2L - y + 4L\log(4L/(4L - y))}{\log(4L/(4L - y)) + \log(4L/y)}\right)^2 dy\right). \tag{15}$$

The MSE is around 58, which is less than the MSE by using ensembles without weight (the MSE is 208.3). We did not do the calculation for 3 bins as the calculation became complicated. We verified our theoretical results experimentally. We also did the experiments with diferent number of bins to understand the behavior of the proposed ensemble methods. In the next section, we present our experimental results.

4. Experiments

We carried out experiments with $y = x$ function. This is a uniformly distributed function. We generated 10,000 points between $0 \leqslant x \leqslant 100$. 5000 points were used for training and 5000 points were used for testing. We used unpruned C4.5 decision tree (J48 decision tree of WEKA software Hall et al., 2009) as the classifier. The final result from the 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 experimental results should follow 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 following 5 × 2 cross-validation (Dietterich, 1998). The average results are presented in the Table 1. Results suggest that there is an excellent match in experimental results with theoretical results for two bins and three bins 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

Table 1
MSE in different cases for the $y = x$ problem. For experimental results, the average results are given, s.d. is given in bracket.

The number of bins	RvC with equal-width bins (Theo.)	RvC with equal-width bins (Exp.) (1)	RvC with ERD (Theo.)	RvC with ERD (Exp.) (2)	RvC with ERD with weight (Theo.)	RvC with ERD with weight (Exp.)	(1)/(2)	(1)/(3)
2	208.3	209.1(2.2)	208.3	210.3(3.1)	58.1	60.3(1.9)	0.99	3.48
3	87.5	90.3(1.7)	75	77.3(1.5)	–	15.9(0.4)	1.17	5.66
5	–	33.1(0.8)	–	18.6(0.4)	–	2.9(0.2)	1.78	11.38
10	–	8.3(0.2)	–	2.6(0.1)	–	0.3(0.0)	3.19	27.67
20	–	2.9(0.1)	–	0.4(0.1)	–	.04(0.00)	7.25	72.50

Table 2
MSE in different cases for the nonlinear problem. For experimental results, the average results are given, s.d. is given in bracket.

The number of bins	RvC equal-width bins (1)	RvC with ERD (2)	RvC eith ERD with weight (3)	(1)/(2)	(1)/(3)
2	4.41(0.23) × 10^6	3.68(0.18) × 10^6	1.63(0.06) × 10^6	1.19	2.26
3	2.14(0.09) × 10^6	1.45(0.06) × 10^6	4.22(0.21) × 10^5	1.48	3.43
5	7.26(0.53) × 10^5	4.17(0.31) × 10^5	8.14(0.44) × 10^4	1.74	8.84
10	1.72(0.11) × 10^5	7.29(0.64) × 10^4	1.44(0.03) × 10^4	2.35	11.94
20	4.41(0.36) × 10^4	1.44(0.15) × 10^4	3.05(0.19) × 10^3	3.06	14.22

Table 3
Details of datasets used in the experiments.

Name	Number of attributes	Size
Abalone	8	4177
Bank8FM	8	8192
Cart	10	40768
Delta_Ailerons	6	7129
Delta_Elevator	6	9517
House (8L)	8	22,784
House (16H)	16	22,784
Housing (Boston)	13	506
Kin8nm	8	8192
Puma8NH	8	8192
Puma32H	32	8192

with the number of bins. This suggests that there is more performance advantage with ERD when we have higher number of bins. This verifies our intuition that as we increase the number of bins the performance advantage increases for ERD ensembles.

To study these ensembles, we also tested these ensembles on a highly nonlinear sinusoidal univariate function, $f(x) = 1 + x^2 - 50x\sin(x/2)$ with x is a real number (Fan et al., 2006). We generated 10,000 points between $0 \leqslant x \leqslant 100$, 5000 points were used for training and 5000 points were used for testing. All other experiment setups were the same as the first experiment. Results are presented in Table 2. Results suggest that the proposed ensemble methods performed better than a single model with equal-width discretization. Hence, the proposed ensembles can be useful even for a nonlinear problem.

4.1. Comparative studies with benchmark datasets

We also carried out experiments with other popular datasets used for regression studies. The information about the datasets is given in Table 3. We also did experiments with REP regression trees (available in WEKA software) with the Bagging procedure. The size of the ensembles was set to 100 for all the experiments. The number of bins was set to 10 for regression by classification methods. Results (Root MSE) presented in Table 4 suggest the proposed ensemble methods perform consistently better than a single model (RvC with equal-width discretization method). This shows the effectiveness of the our approach. The comparative study, with REP tree regression trees with Bagging, suggests that our methods perform similar to this method. This shows that the proposed methods are comparable to the method that is developed specifically for the regression problems. Even though the theoretical study suggests that the weights should be useful for the proposed ensembles, we did not see much difference in our two methods. We investigated the reasons for this behavior. We found that in our theoretical calculations, we assumed that the classification error was 0, whereas in these experiments the average classification errors for different datasets were varying between 20% and 40%. This means we were giving weights to the results which were wrong. Hence, weights are not providing the advantage as expected.

In the proposed ensemble method, we used decision trees as the classifier, however, we may use any other classifier. The number of bins is an important variable, as a small number of bins lead to the better classification. However, the value represented by the bins

Table 4
Experimental results for different methods for different datasets. The average results for Root Mean Square Error (RMSE) are presented. s.d. is given in the bracket.

Name of dataset	RvC with ERD	RvC with ERD with weight	RvC with equal-width bins	Bagging with REP regression trees
Abalone	2.24(.05)	2.28(0.06)	2.89(0.08)	2.17(.05)
Bank8FM	3.61(.11) × 10^{-2}	3.64(.15) × 10^{-2}	5.31(0.17) × 10^{-2}	3.52(.12) × 10^{-2}
Cart	1.06(.02)	1.04(.02)	1.46(0.06)	1.05(0.03)
Delta_Ailerons	1.72(.03) × 10^{-4}	1.74(.04) × 10^{-4}	2.75(0.05) × 10^{-4}	2.03(0.03) × 10^{-4}
Delta_Elevator	1.52(.02) × 10^{-3}	1.50(.02) × 10^{-3}	1.91(0.03) × 10^{-3}	1.55(0.02) × 10^{-3}
House (8L)	3.12(.05) × 10^4	3.14(.04) × 10^4	4.12(.08) × 10^4	3.06(.03) × 10^4
House (16H)	3.51(.07) × 10^4	3.51(.07) × 10^4	4.62(.10) × 10^4	3.55(.05) × 10^4
Housing (Boston)	3.98(.09)	4.21(.08)	5.23(0.12)	4.01(0.10)
Kin8nm	0.17(0.01)	0.17(0.01)	0.24(0.02)	0.17(0.01)
Puma8NH	3.28(0.14)	3.25(0.17)	4.50(0.16)	3.25(0.11)
Puma32H	8.21(0.43) × 10^{-3}	8.23(0.51) × 10^{-3}	1.2(.04) × 10^{-2}	7.94(0.39) × 10^{-3}

will be less representative of the values. 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 value represented by the bins will be more representative of the points in the bins. One may use cross validation to find out the best number of bins for the best regression results. However, in the present setup, we used the default value of bins as 10, and the results suggest that even with the default number of bins, we got the results similar to the models specifically designed for regression. This shows the effectiveness of our proposed ensemble method. Hence, we may use classifier models with the proposed ensemble method to solve regression problems.

5. Conclusion

In supervised learning, the target values may be continuous or a discrete set of class. The continuous target values (the regression problem) can be transferred to a discrete set of classes (the classification problem). The discretization process is a popular method to achieve this task. In this paper, we proposed two ensemble methods for RvC problems. We showed theoretically that the proposed ensemble methods performed better than a single model with equal-width discretization method. This is also verified with experiments. Experiments results also suggest that our methods performed similar to the method developed for the regression purpose. This suggests that the proposed ensemble method is useful for regression problems. As the proposed method is independent of the choice of the classifier, various classifiers can be used with the proposed method to solve the regression method. In this paper, we carried out experiments with the decision trees, however in future we will do the experiments with other classifiers like naive Bayes and support vector machines to study its effectiveness with other classifiers. In this paper, in the weighted version of the algorithm, we took weight as 1/(The size of the bin), however, other weight schemes like 1/(The size of the bin)2 can be used. In future, the effect of different weight schemes will be studied.

References

Ahmad, A. (2010). *Data transformation for decision tree ensembles*. PhD thesis, School of Computer Science, University of Manchester.

Bibi, S., Tsoumakas, G., Stamelos, I., & Vlahavas, I. (2008). Regression via classification applied on software defect estimation. *Expert Systems with Applications, 32*, 20912101.

Bishop, C. M. (2008). *Pattern recognition and machine learning*. New York, Inc.: Springer-Verlag.

Breiman, L. (1996). Bagging predictors. *Machine Learning, 24*(2), 123–140.

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and regression trees*. CA: Wadsworth International Group.

Burden, Richard L. (2010). *Numerical analysis* (9th ed.). BookNumerical.

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery, 2*, 121167.

Catlett, J. (1991). *Megainduction: Machine learning on very large databases*. PhD thesis, Basser Department of Computer Science, University of Sydney.

Chan, C. C., Batur, C., & Srinivasan, A. (1991). Determination of quantization intervals in rule based model for dynamic systems. In *Proceedings of the IEEE conference on systems, man, and cybernetics* (pp. 1719–1723).

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation, 10*, 1895–1923.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *Proceedings of conference multiple classifier systems* (Vol. 1857, p. 1–15).

Dougherty, J., Kahavi, R., & Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. In *Machine learning: Proceedings of the twelth international conference*.

Fan, W., Wang, H., Yu, P. S., & Ma. S. (2003). Is random model better? On its accuracy and efficiency. In *Proceedings of third IEEE international conference on data mining (ICDM2003)* (p. 51–58).

Fan, W., McCloskey, J., & Yu, P. S. (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* (p. 136–146).

Fayyad, U. M., & Irani, K. B. (1993). Multi-interval discretization of continuous valued attributes for classification learning. In *Proceedings of the thirteenth international joint conference on artificial intelligence* (p. 1022–1027).

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55*(1), 119–139.

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning, 63*(1), 3–42.

Hall, M., Frank, E., Holmes, Geoffrey, Pfahringer, B., Reutemann, P., & Witten, Ian H. (2009). The WEKA data mining software: An update. *SIGKDD Explorations, 11*(1), 10–18.

Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 12*(10), 993–1001.

Holte, R. C. (1993). Very simple classification rules perform well on most commonly used datasets. *Machine Learning, 11*, 63–90.

Indurkhya, N., & Weiss, S. M. (2001). Solving regression problems with rule-based ensemble classifiers. In *ACM international conference knowledge discovery and data mining (KDD01)* (pp. 287–292).

Kohonen, T. (1989). *Self organization and associative memory*. Berlin, Germany: Springer-Verlag.

Kuncheva, L. I. (2004). *Combining pattern classifiers: Methods and algorithms*. Wiley-Interscience.

Kwedlo, W., & Kretowski, M. (1999). An evolutionary algorithm using mulivariate discretization for decision rule induction. In *Principles of data mining and knowledge discovery* (pp. 392–397).

Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. San Francisco, CA, USA: Morgan Kaufman Publishers Inc.

Rodriguez, J. J., Kuncheva, L. I., & Alonso, C. J. (2006). Rotation forest: A new classifier ensemble method. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 28*(10), 1619–1630.

Torgo, L., & Gama, J. (1997). Search-based class discretization. In *Proceedings of the 9th European conference on machine learning* (pp. 266–273).

Torgo, L., & Gama, J. (1999). Regression by classification. In: *Advances in artificial intelligence* (pp. 51–60).

Torgo, L., & Gama, J. (1997). Regression using classification algorithms. *Intelligent Data Analysis, 4*(1), 275–292.

Tumer, K., & Ghosh, J. (1996). Error correlation and error reduction in ensemble classifiers. *Connection Science, 8*(3), 385–404.

Van de Merckt, T. (1993). Decision trees in numerical attribute spaces. In *Proceedings of the 13th international joint conference on artificial intelligence* (pp. 1016–1021).

Vapnik, V. (1998). *Statistical learning theory*. New York: Wiley-Interscience.