Comparative Analysis of Decision Tree Algorithms: ID3, C4.5 and Random Forest

Shiju Sathyadevan and Remya R. Nair

Abstract To analyze the raw data manually and find the correct information from it is a tough process. But Data mining technique automatically detect the relevant patterns or information from the raw data, using the data mining algorithms. In Data mining algorithms, Decision trees are the best and commonly used approach for representing the data. Using these Decision trees, data can be represented as a most visualizing form. Many different decision tree algorithms are used for the data mining technique. Each algorithm gives a unique decision tree from the input data. This paper focus on the comparison of different decision tree algorithms for data analysis.

Keywords Iterative dichotomiser 3 (ID3) · C4.5 · Randomforest

1 Introduction

We live in a world where vast amounts of data are collected daily. Analyzing such data for acquiring the meaningful information is an important need. Human analysts with no special tool might not yield in producing the right result sets. But data mining can meet this need by providing tools to discover knowledge from data, Classification and prediction are two forms of data analysis tools.

Classification is one of the fundamental and Useful technique in data mining. Using classification algorithm we can construct a model and used to predict the class label of the testing instances. It has been successfully applied to many real world application areas, such as medical diagnosis, weather prediction, credit approval etc. In classification, several approaches are adopted to classify the data.

S. Sathyadevan (✉) · R.R. Nair
Amrita Center for Cyber Security Systems and Networks, Amrita University, Kollam, India
e-mail: shiju.s@am.amrita.edu

R.R. Nair
e-mail: remyarnair@am.amrita.edu

L.C. Jain et al. (eds.), *Computational Intelligence in Data Mining - Volume 1*,
Smart Innovation, Systems and Technologies 31, DOI 10.1007/978-81-322-2205-7_51

Decision trees are a very popular and commonly used approach for classification [1]. This Decision tree classifiers start with the Training set with their associated class labels. The root node is the main feature. Each internal node represents the test attributes, each branch represents the outcome of the test and each leaf node represents the class labels. To identify the class label for an unknown sample, the Decision tree classifier will trace path from root to the leaf node, which holds the class label for that sample.

This paper focus on the comparison of three different decision tree algorithms ID3, C4.5 and Random Forest [2]. These three decision tree algorithms are different in their features and hence in the accuracy of their result sets. ID3 and C4.5 build a single tree from the input data. But there are some differences in these two algorithms. ID3 only work with Discrete or nominal data, but C4.5 work with both Discrete and Continuous data. Random Forest is entirely different from ID3 and C4.5, it builds several trees from a single data set, and select the best decision among the forest of trees it generate. The rest of the paper is organized in four sections. In Sect. 2 the related work and literature of decision tree algorithms is presented. In Sect. 3 the three algorithms', ID3, C4.5 and Random Forest is explained with a brief discussion about the algorithm along with the pseudo code. Section 4 compares the three algorithms and explains the differences among them. In Sect. 5, an application is selected where the three algorithms are applied and compared with respect to their prediction accuracy.

2 Related Work

There are so many researches are done for Decision tree learning algorithms. ID3 and C4.5 are two most popular decision tree algorithms. Among these algorithms, ID3 is the basic algorithm. But it has many drawbacks. So To improve ID3, researchers have proposed many methods, such as, use weighting instead of information gain [3], user's interestingness [4] and attribute similarity to information gain as weight. Chun and Zeng [5] also have proposed improved ID3 based on weighted modified information Gain called ωID3.

Quinlan [6] has developed tree based classification algorithm known as C4.5, an extension to the ID3 algorithm. It uses the gain ratio for building the tree. The C4.5 deals with continuous attributes which was not supported by ID3. It divides the values of a continuous attribute in a two subsets. He also proposed method of pruning, which deals with the removal of unwanted branches which are generated by noise or too small size of training data [7].

In the case of Random Forest algorithm, it builds random trees from a single input dataset. This algorithm is efficient for predicting the accurate results. Because, we compare many trees and obtain the best from them. So the result obtained from Random Forest is more accurate than ID3 and C4.5. So our main goal is to compare the prediction accuracy of these 3 decision tree algorithms ID3, C4.5 and Random Forest and achieve high performance and accurate results.

3 Understanding Decision Tree Classifiers

3.1 ID3 Classifier

ID3 (Iterative Dichotomiser 3) is the basic algorithm for inducing decision trees. This algorithm builds a decision tree from the data which are discrete in nature. For each node, select the best attribute. And this best attribute is selected using the selection criteria—Information Gain [8]. It indicates how much informative a node is. And the attribute which has the highest Information Gain is selected as split node.

3.2 C4.5 Classifier

C4.5 Algorithm is developed based on the Decision tree Algorithm ID3 [9]. ID3 is also used to generate decision trees. But it does not guarantee an optimal solution to analyze continuous data. But C4.5 algorithm overcomes these short comings. They are:

```
Algorithm 1 pseudo code
Input: - Dataset S
Output: - Decision Tree
  Begin
1.Create a Root node for the tree.
2.If all Examples are +ve, Return single Root with label = +.
3.If all Examples are -ve, Return single Root with label = -.
4.If Attributes is empty, Return the single-node tree with
  label = most common value of Target attributes in Examples.
5.Otherwise Begin
   (a) A, the attribute that best Classifies the input data.
   (b) The decision attribute for Root <- A.
   (c) For each possible value, vi , of A,
      (i)   Add a new tree branch below Root corresponding to
            the test A = vi
      (ii)  Let Examples be the subset of Examples that have
            value vi for A
      (iii)  If Examples is empty

          (A) Then below this new branch add a leaf node with
              label = most common value of  Target attribute in
              Examples
          (B) Else below this new branch add the sub tree.
6.Building the decision tree nodes and branches recursively
  until a certain subset of the instances belonging to the same
  category.
  End
```

1. Attribute Selection Criteria

ID3 uses Information Gain as the Selection criteria.

$$Gain(S) = Entropy(S) - \sum_{c=1}^{n} \frac{|S_c|}{S} Entropy(S_v) \quad (1)$$

If the input is continuous, this Gain doesn't give an optimum result. That's why ID3 is only applicable for discrete datas. But C4.5 overcomes this by introducing new method known as Gain Ratio: This method contains two concepts, Gain and Split Info.

$$Gain\,Ratio(S) = \frac{Gain(S)}{Split\,Info(S)} \quad (2)$$

So If the attribute is continuous, this selection criteria gives the optimum result.

3.3 Random Forest Classifier

Random forest is another Decision tree technique that operates by constructing multiple decision trees [10]. This algorithm is based on bagging (Bootstrap aggregating) [11], i.e. after building multiple random training subsets, the algorithm construct one tree per random training subsets. This technique is called random split selection method and the trees known as random trees.

```
Algorithm 2 pseudo code
Input: -  A data set S
Output: - Random Number of Trees
  Begin
1.Choose T - number of trees to grow
2.For b = 1 to T
   a)Draw a Bootstrap sample Z  of size N from the training Data
   b)Grow a random-forest tree T b to the bootstrapped data, by
     re -cursively repeating until the minimum node size nmin is
     reached.
      1)Select m variables at random from the p variables.
      2)Pick the best variable/split-point among the m.
      3)Split the node into daughter nodes.
3.  Output the ensemble of trees fTBg
    End
```

4 Comparisons

Among the three algorithms: ID3, C4.5 and Random Forest. We know that, ID3 is the basic decision tree algorithm. The selection criteria used for ID3 is not suit for continuous datasets. So this algorithm is only applicable for discrete cases. To overcome this problem, Quinlan extend the ID3 to C4.5 by introducing a new selection criteria called Gain Ratio [12]. This gives optimum result with both discrete and continuous case data sets. It is not possible to assure that the tree build from ID3 and C4.5 is an accurate tree, because it only generates a single tree for a given set of input data. So if the new data set is applied to the model(tree) so generated, we get only one prediction result. This Prediction may or may not be correct. Hence the correctness and accuracy of these two algorithms cannot be assured. To overcome this accuracy problem, Random Forest was implemented. This algorithm generates several trees (many Random trees are generated) from a single data set. So the method is to apply the new data set to every tree in the model and list the output generated from it. Best prediction is determined by selecting the majority class value. That is, which ever class among the lot is predicted the most number of times that class is selected as the final prediction.

5 Experimental Results

Dataset used for this application is Credit Approval Dataset, obtained from the Machine Learning repository UCI [13].

In this Credit Approval dataset some of the attributes are linear and some are nominal. Table 1 shows the characteristics of selected Credit Dataset.

Now a day, banks play a crucial role in market economies. For markets and society to function, individuals and companies need access to credit [14]. To a bank, a good risk prediction model for loan allocation is necessary so that the bank can provide as much credits as possible without any risk. This report describes an approach for predicting the loan approving application using ID3, C4.5 and Random Forest [15]. Using these algorithms, we can determine whether the loan can be approved or not. But before approving a loan, bank will look at all the relevant financial history of the borrower. These features are represented as the attributes of the dataset are shown in Table 2.

As a first step, we check the class labels of all the instances in the dataset. If all tends to fall under the same class, it is possible to stop the tree with the leaf node as

Table 1 Training dataset matrix

Dataset	Credit approval
No of instances	25
No of attributes	21
No of classes	2

Table 2 Training dataset attribute details

Attribute name	Description
Checking account	Checking the account status
Credit history	No credits taken/all credits paid back duly
Purpose	Car, furniture, business, education etc.
Credit amount	Credit amount
Savings status	Current saving status of the borrower
Years employed	Unemployed/present employed since
Installment rate	Installment rate in % of disposable income
Personal status	Single, married, divorced, separated
Other debtors	Guarantors/co-applicant
Resident since	Present residence since
Property	Real estate, savings, car, no property
Age	Age of borrower in years
Other plans	Banks, stores, none
Noncredit at bank	Number of existing credits at this bank
Job	Check the borrower is employed or not
Dependents	Number of dependents including family
Telephone	None or yes
Foreign	None or yes
Housing	Rent/own/for free
Approve	Good or bad

the common class label. Otherwise, select the best attribute that partition the training set.

Figure 1 shows the decision tree built by the ID3 algorithm where it uses information gain as the splitting measure. This tree does not represent an optimum result because some of the attributes are continuous and some are discrete. In the case of continuous datasets, the highest gain is for the attribute that has more number of partitions. Here, out of 20 attributes, 7 attributes are continuous (Credit Amount, Duration, Installment rate, Resident Since, Age, Num Credits At Bank and Dependents). Information Gain is used as the selection method. Out of this 7 continuous attributes, highest gain is for the attribute “Credit Amount” (because it has 25 different values). So based on the Information Gain, Credit Amount is selected as the Root node. But this tree cannot be treated as an optimum solution for this application, because there are no other checking conditions (sub node) in this tree.

So if it is tested against a new data set, then it will predict only based on the attribute “Credit Amount”. For example, Table 3 represents testing data consisting of 10 instances, where 30 % of the test data is the training data without class label

Fig. 1 Credit approval tree using ID3

and 70 % of the test data is new test instances. Each instance in this test data will be predicted with the help of the ID3 tree in Fig. 2. So when this data is applied on ID3, each instance traverses the ID3 tree and reaches the class values (in this case either good or bad). This class values will be taken as the predicted result. ID3 tree

Table 3 Test data applied against the model

Checking account	Credit history	Purpose	Credit amount	Savings status	Years employed	Installment rate	Personal status	
<0	Ok	Car	7,200	<100	<7	3	Male-married	…
<200	Ok-till-now	Furniture	6,923	<100	<4	4	Female	…
None	Critical	Car	5,423	<500	<4	4	Male-single	…
None	Ok	Television	426	<100	>=7	4	Male-married	…
>=200	Ok-at-this-bank	Television	409	>=1000	<4	3	Female	…
<0	Ok-till-now	Furniture	7,882	<100	<7	2	Male-single	…
<200	Ok-till-now	Television	2,415	<100	<4	3	Male-single	…
<0	Past-delays	Car	4,870	<100	<4	3	Male-single	…
<0	Ok-till-now	Furniture	1,374	<100	<4	1	Male-married	…
<0	Past-delays	Business	6,836	<100	>=7	3	Male-divorced	…

1	Checking	Duration	CreditHistory	Purpose	CreditAmou	Saving	YearsEmplo	Install	PersonalStatu	OtherDe	Resident	Property	Age	Other	Hou	NumC	Job	Depende	Teleph	Foreig	Approve
2	<0	0	ok	furniture	-1000000000	unknow	unemployed	-10000	female_single	none	0	car	-293	none	own	-1000	skilled	-185	yes	no	good
3	<0	6	critical	television	1169	unknow	>=7	4	male_single	none	4	real_estate	67	none	own	2	skilled	1	yes	yes	good
4	<200	48	ok_til_now	television	5951	<100	<4	2	female	none	2	real_estate	22	none	own	1	skilled	1	no	yes	bad
5	none	12	critical	education	2096	<100	<7	2	male_single	none	3	real_estate	49	none	own	1	unskilled	2	no	yes	good
6	<0	42	ok_til_now	furniture	7882	<100	<7	2	male_single	guarantor	4	savings	45	none	free	1	skilled	2	no	yes	good
7	<0	24	past_delays	car_new	4870	<100	<4	3	male_single	none	4	unknown	53	none	free	2	skilled	2	no	yes	bad
8	none	36	ok_til_now	education	9055	unknow	<4	2	male_single	none	4	unknown	35	none	free	1	unskilled	2	yes	yes	good
9	none	24	ok_til_now	furniture	2835	<1000	>=7	3	male_single	none	4	savings	53	none	own	1	skilled	1	no	yes	good
10	<200	36	ok_til_now	car_used	6948	<100	<4	2	male_single	none	2	car	35	none	rent	1	management	1	yes	yes	good

Fig. 2 Training dataset for building the model

contains 1 root node "Credit Amount" and 25 leaf nodes. Prediction result are as listed in Table 4. So each instance in the above testing data will only check their Credit Amount value, based on which the results will be predicted.

So in actual practice, it is not possible to make a decision whether to approved or reject a loan based on just one attribute which in this case is "Credit Amount". So this cannot be considered as an optimum result. To overcome this, the same data set is pushed through C4.5.

If we apply this Credit dataset on C4.5, it builds a decision tree which is totally different from ID3's output.

In C4.5 we use Gain Ratio as the splitting criteria. Figure 3 shows the C4.5 Decision Tree. But the working of Gain Ratio is different from Information Gain as used in ID3. Gain Ratio doesn't depend on the attribute type, instead calculates Information Gain and Split Info for each attribute. It will select the attribute which has the highest Gain Ratio as the best split-point. So by using this Ratio format formula Eq. (2), it is now possible to overcome the problems faced while using ID3.

The same 3 data instances as shown in Table 3, when applied to C4.5, each instance traverse through the model and predict the output. Here the attribute "Personal status" is selected as the root node as it had the highest gain ratio. "Installment rate" is selected as the sub node to "Persona Status" as it has the highest Gain ratio within that tree. Going forward when new data sets are applied to this model, it will only check "Personal Status" and "Installment Rate", because in this tree these are the 2 nodes whose value will predict the result set. Prediction results are as listed in Table 5.

It is not possible to assure that the tree build from C4.5 is an accurate one because here also only a single tree is generated from the input data. So if a new test data is applied to the model (tree), there will only be one prediction result which may or may not be correct.

If it is possible to compare more than one prediction and pick the best prediction from the lot, then it can provide better accuracy. Random Forest Algorithm can build number of trees from the training data set and is represented in the form of rules [16]. For each tree, corresponding rules are generated. Testing data (as shown in Table 3) is applied on each rule generated from the Random Forest, and pick the one that occurs most frequent as the best decision.

Figure 4 shows the rules generated by applying credit dataset on Random Forest. When a new test data set is applied on to these rules, then whichever class value (Good or bad) occurs the most for an instance against each tree, that class value will be set as the best prediction. The same test data set show under Table 3 is applied against Random Forest algorithm.

Table 4 Prediction generated by ID3

Checking account	Credit history	Purpose	Credit amount	Savings status	Years employed	Installment rat		Approve
<0	Ok	Car	7,200	<100	<7	3	…	Good
<200	Ok-till-now	Furniture	6,923	<100	<4	4	…	Good
None	Critical	Car	5,423	<500	<4	4	…	Bad
None	Ok	Television	426	<100	>=7	4	…	Bad
>=200	Ok-at-this-bank	Television	409	>=1,000	<4	3	…	Bad
<0	Ok-till-now	Furniture	7,882	<100	<7	2	…	Good
<200	Ok-till-now	Television	2415	<100	<4	3	…	Bad
<0	Past-delays	Car	4870	<100	<4	3	…	Bad
<0	Ok-till-now	Furniture	1374	<100	<4	1	…	Bad
<0	Past-delays	Business	6836	<100	>=7	3	…	Good

Fig. 3 Credit approval tree using C4.5

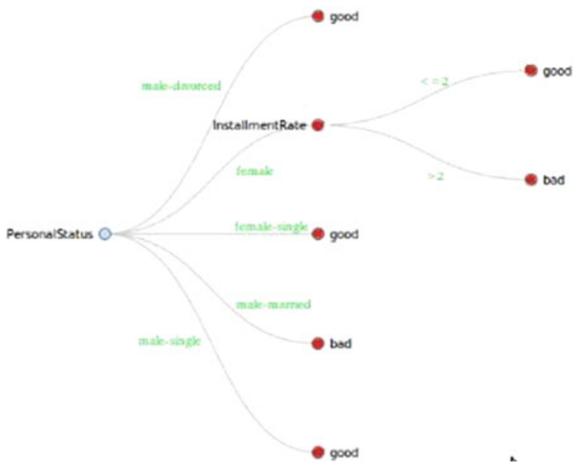

Table 5 Prediction generated by C4.5

Checking account	Credit history	Purpose		Installment rate	Personal status		Approve
<0	Ok	Car	..	3	Male-married	..	Bad
<200	Ok-till-now	Furniture	..	4	Female	..	Good
None	Critical	Car	..	4	Male-single	..	Good
None	Ok	Television	..	2	Male-married	..	Bad
>=200	Ok-at-this-bank	Television	..	3	Female	…	Bad
<0	Ok-till-now	Furniture	..	2	Male-single	…	Good
<200	Ok-till-now	Television	..	3	Male-single	…	Good
<0	Past-delays	Car	..	3	Male-single	…	Good
<0	Ok-till-now	Furniture	..	1	Male-married	…	Bad
<0	Past-delays	Business	..	3	Male-single	…	Good

As shown in Table 6, in the 1st instance, Tree1 and Tree 3 predicted “good” where as Tree 2 predicted as bad. So class value with the maximum occurrence is “good”. So this be selected as the final prediction, where as in the 2nd instances, class value “bad” have the maximum occurrence, and hence its final prediction was set as “bad”. Similarly for the 3rd instance class value “Good” happen to repeat the most.

The output got from these three algorithms is different. In the case of ID3 and C4.5, only one prediction result was generated. Hence it was not easy to confirm that the prediction was correct as there were no way to compare the result sets. But in the case of Random Forest, three different decision trees were built, and hence three predictions were generated. From these three predictions, the class with the most occurrence was treated as the best prediction.

```
Tree3.model
IF {Property=car AND  Duration=<24.0} THEN Approve IS [good]
IF {Property=car AND  Duration=>24.0} THEN Approve IS [good]
IF {Property=unknown} THEN Approve IS [bad]
IF {Property=savings} THEN Approve IS [good]
IF {Property=real_estate AND  CheckingAccount=<200} THEN Approve IS [bad]
IF {Property=real_estate AND  CheckingAccount=none} THEN Approve IS [good]
IF {Property=real_estate AND  CheckingAccount=<0} THEN Approve IS [good]
Tree1.model
IF {SavingsAccount=<100 AND  Purpose=car_new AND  CreditAmount=<4870.0} THEN Approve IS [bad]
IF {SavingsAccount=<100 AND  Purpose=car_new AND  CreditAmount=>4870.0} THEN Approve IS [bad]
IF {Purpose=television AND  CheckingAccount=<200 AND  InstallmentRate=<2.0} THEN Approve IS [bad]
IF {Purpose=television AND  CheckingAccount=<200 AND  InstallmentRate=>2.0} THEN Approve IS [bad]
IF {Purpose=car_used} THEN Approve IS [bad]
IF {Purpose=education} THEN Approve IS [good]
IF {SavingsAccount=<1000} THEN Approve IS [good]
IF {SavingsAccount=<500} THEN Approve IS [bad]
IF {SavingsAccount=unknown} THEN Approve IS [good]
IF {SavingsAccount=>=1000} THEN Approve IS [good]
Tree2.model
IF {NumCreditsAtBank=<3.0} THEN Approve IS [good]
IF {NumCreditsAtBank=>3.0} THEN Approve IS [good]
IF {NumCreditsAtBank=>2.0 AND  InstallmentRate=<4.0 AND  CheckingAccount=none} THEN Approve IS [good]
IF {NumCreditsAtBank=>2.0 AND  InstallmentRate=<4.0 AND  CheckingAccount=<0} THEN Approve IS [bad]
IF {InstallmentRate=>4.0 AND  CreditAmount=<2134.0} THEN Approve IS [good]
IF {InstallmentRate=>4.0 AND  CreditAmount=>2134.0} THEN Approve IS [good]

-----------------------
No of trees generated : 3
-----------------------
```

Fig. 4 Credit approval rules—using random forest

Table 6 Prediction generated by random forest

Instances	Tree1	Tree2	Tree3	Final prediction
1st	Good	Bad	Good	Good
2nd	Bad	Bad	Good	Bad
3rd	Bad	Good	Good	Good
4th	Bad	Good	Good	Good
5th	Bad	Good	Bad	Bad
6th	Good	Good	Good	Good
7th	Bad	Good	Good	Good
8th	Good	Good	Good	Good
9th	Bad	Good	Bad	Bad
10th	Good	Good	Bad	Good

In the case of Prediction accuracy, Random Forest is better when compared to other two algorithms. The Prediction Accuracy of three algorithms is shown in Fig. 5.

This bar chart gives the overall idea about the Prediction Accuracy of the three Algorithms: ID3, C4.5 and Random For-est. For testing, 30 % of the training data without class and 70 % of the new data was selected. So from the above testing, ID3 gives prediction only based on 1 attribute (i.e. Root node-Credit Amount) only 30 % of the data is correctly classified. In the case of C4.5, it gives prediction based on 2 attributes (i.e., Personal-Status and Installment Rate), here 60 % of the testing data is correctly classified where as Random Forest compare multiple prediction

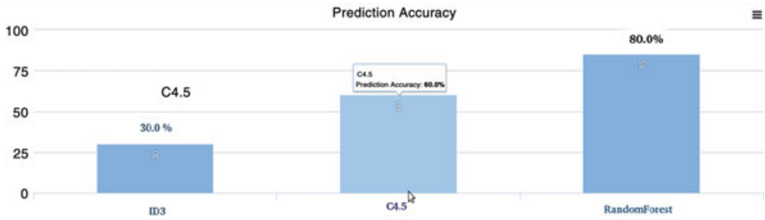

Fig. 5 Prediction accuracy—bar chart

results and select the best prediction from it based on the occurrence of the class values, here out of 10 instances, 8 instances are correctly classified, i.e. 80 % are correctly predicted.. So the Prediction Accuracy of Random Forest is better when compared to ID3 and C4.5. From the test results it is evident that from the three different forms of decision trees, Random Forest could generate better decision.

6 Conclusion

This paper provides a brief explanation of Decision tree learning method and classification techniques. It explains the main three Decision tree algorithms, ID3, C4.5 and Random Forest. Paper highlights the fact that these three decision tree algorithms are different in their prediction accuracy. Experimental Result section, compares these algorithms against a common testing dataset and the result set so generated are also compared. After analyzing and comparing the results, Random Forest gives the better prediction result. So Among these algorithms, Random Forest is best for accurate classification.

Acknowledgments We thank our college Amrita School of Engineering, Amritapuri and Amrita Center of Cyber Security, Amritapuri for giving us an opportunity to be a part of the internship program that leads to the development of this work. Many thanks to Shiju Sathyadevan for countless discussions and feedback that help me to complete the work successfully.

References

1. Rokach, L., Maimon, O.: Decision trees. In Maimon, O.Z., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook vol. 6, pp. 165–192. Springer, Heidelberg (2005)
2. Peng, X., Guo, H., Pang, J.: Performance analysis between different decision trees for uncertain data. In: Proceedings of the 2012 International Conference on Computer Science and Service System, CSSS ’12, pp. 574–577. Washington, DC (2012)
3. Jin, C., De-lin, L., Fen-xiang, M.: An improved ID3 decision tree algorithm. In: Proceedings of 2009 4th International Conference on Computer Science and Education, vol. 1 (2009)
4. Wang, M., Chai, R.: Improved classification attribute selection scheme for decision for decision tree. Comp. Eng. Appl. **3,** 127–129 (2010)

5. Guan, C., Zeng, X.: An improved ID3 based on weighted modified information gain. In: Seventh International Conference on Computational Intelligence and Security, vol. 1, pp. 1283–1285. IEEE Computer Society, Washington, DC (2011)
6. Quinlan, J.R.: C4.5 program for machine learning, vol. 16, pp. 21–30. Morgan Kaufmann Publishers Inc., San Francisco (1993)
7. Quinlan, J.R.: Improved use of continuous attributes inc 4.5. J. Artif. Intell. Res. **4**, 77–90 (1996)
8. de Vries, A.P., Roelleke, T.: Relevance information: a loss of entropy but a gain for idf? In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 282–289 (2005)
9. Ruggieri, S.: Efficient C4.5. IEEE Trans. Knowl. Data Eng. **2,** 438–444 (2002)
10. Kotsiantis, S.: A hybrid decision tree classifier. J. Intell. Fuzzy Syst. **1**, 327–336 (2014)
11. Fern, A., Givan, R.: Online ensemble learning: an empirical study. Mach. Learn **53**, 279–286 (2000) (Morgan Kaufmann Publishers Inc., San Francisco)
12. Quinlan, J.R.: C4.5: Programs for machine learning, vol. 16, pp. 235–240. Morgan Kaufmann Publishers Inc., San Francisco, CA (1993)
13. Bache, K., Lichman, M.: UCI machine learning repository. Digit. Libr. **1** (2013)
14. Abele, J., Mantas, C.J.: Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring. Expert Syst. Appl. Int. J. **41,** 3825–3830. Pergamon Press Inc., USA (2014)
15. Olson, D.L., Delen, D., Meng, Y.: Comparative analysis of data mining methods for bankruptcy prediction. Decis. Support Syst. **2,** 464–473 (2012)
16. Grzymala-Busse, J.W.: Rule induction. In: Maimon, O.Z., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, vol. 6, pp. 277–294. Springer, Heidelberg (2005)
17. Maimon, O., Rokach, L.: The Data Mining and Knowledge Discovery Handbook, vol. 2, pp. 1–17. Springer, Heidelerg (2005)
18. Podgorelec, V., Kokol, P., Stiglic, B., Rozman, I.: Decision trees: an overview and their use in medicine. J. Med. Syst. **5,** 445–463 (2002)
19. Povalej, P., Kokol, P.: End user friendly data mining with decision trees: a reality or a wish? In: Proceedings of the 2007 Annual Conference on International Conference on Computer Engineering and Applications, vol. 3, pp. 35–40. Stevens Point, Wisconsin (2007)