An Improved CART Decision Tree for Datasets with Irrelevant Feature

Conference Paper · December 2011

DOI: 10.1007/978-3-642-27172-4_64

5 authors, including:

Ali Mirza Mahmood

DMS SVH College of Engineering

30 PUBLICATIONS **158** CITATIONS

SEE PROFILE

Mohammad Imran

Neil Gogte Institute of Technology

11 PUBLICATIONS **11** CITATIONS

SEE PROFILE

Rajesh Vemulakonda

Mother Theresa Institute of Science & Technology

3 PUBLICATIONS **4** CITATIONS

SEE PROFILE

An Improved CART Decision Tree for Datasets with Irrelevant Feature

Ali Mirza Mahmood[1], Mohammad Imran[2], Naganjaneyulu Satuluri[1],
Mrithyumjaya Rao Kuppa[3], and Vemulakonda Rajesh[4]

[1] Acharya Nagarjuna University, Guntur, Andhra Pradesh, India
[2] Rayalaseema University, Kurnool, Andhra Pradesh, India
[3] Vaagdevi College of Engineering, Warangal, Andhra Pradesh, India
[4] Pursing M.Tech, MIST, Sathupalli, Khamaman District, Andhra Pradesh, India
alimirza.md@gmail.com

Abstract. Data mining tasks results are usually improved by reducing the dimensionality of data. This improvement however is achieved harder in the case that data size is moderate or huge. Although numerous algorithms for accuracy improvement have been proposed, all assume that inducing a compact and highly generalized model is difficult. In order to address above said issue, we introduce Randomized Gini Index (RGI), a novel heuristic function for dimensionality reduction, particularly applicable in large scale databases. Apart from removing irrelevant attributes, our algorithm is capable of minimizing the level of noise in the data to a greater extend which is a very attractive feature for data mining problems. We extensively evaluate its performance through experiments on both artificial and real world datasets. The outcome of the study shows the suitability and viability of our approach for knowledge discovery in moderate and large datasets.

Keywords: Classification, Decision trees, Filter, Randomized gini index.

1 Introduction

In Machine Learning community, and in Data Mining works, Classification has its own importance. Classification is an important part and the research application field in the data mining [1]. With ever-growing volumes of operational data, many organizations have started to apply data-mining techniques to mine their data for novel, valuable information that can be used to support their decision making [2]. Organizations make extensive use of data mining techniques in order to define meaningful and predictable relationships between objects [3]. Decision tree learning is one of the most widely used and practical methods for inductive inference [4]. Decision trees are one of the most effective machine learning approach for extracting practical knowledge from real world datasets [5].

The main contributions of this work can be summarized as follows.

(i)We show that a fast random sampling framework can be used to enhance the generalization accuracy of the tree. (ii) It is worth to note here that the main peculiarity of this composite splitting criterion is that the resulting decision tree is

B.K. Panigrahi et al. (Eds.): SEMCCO 2011, Part I, LNCS 7076, pp. 539–549, 2011.

better in accuracy. (iii) We connect the theoretical results from state-of-the-art decision tree algorithm (CART) showing the viability of our method and also show empirical results supporting our claim.

2 Related Work

In this Section, we present some recent work on decision trees in different areas, Aviad. B [6] have proposes and evaluates a new technique to define decision tree based on cluster analysis. The results of the model were compared to results obtained by conventional decision trees. It was found that the decision rules obtained by the model are at least as good as those obtained by conventional decision trees. In some cases the model yields better results than decision trees. In addition, a new measure is developed to help fine-tune the clustering model to achieve better and more accurate results. Pei-Chann Chang [7] have applied fuzzy logic as a data mining process to generate decision trees from a stock database containing historical information. They have establishes a novel case based fuzzy decision tree model to identify the most important predicting attributes, and extract a set of fuzzy decision rules that can be used to predict the time series behavior in the future. The fuzzy decision tree generated from the stock database is then converted to fuzzy rules that can be further applied in decision-making of stock price's movement based on its current condition. Leyli Mohammad Khanli [8] have applied active rule learning is regarded for resource management in grid computing. Rule learning is very important for updating rules in active database system. But, it is also very difficult because of lacking methodology and support. Decision tree can use into rule learning to cope with the problems arisen in active semantic extraction, termination analysis of rules set and rules update. Also their aim from rule learning is learning new attributes in rules such as time, load balancing regarded to instances of real Grid environment that decision tree can provide it. Ali Mirza Mahmood [9] have proposed the use of expert knowledge in pruning decision trees for applicability in medical analysis. There has been significant research interest in decision trees in recent years. In [10] author have proposed an improved decision tree classification algorithm MAdaBoost which constructs cascade structures of more decision tree classifiers based on AdaBoost for tackling the problem of imbalanced datasets. The improved algorithm eliminates the short coming of imbalance datasets and improves the overall accuracy of cascade classifiers. In [11] author proposed improved decision tree which uses series of pruning techniques that can greatly improve construction efficiency of decision trees when using for uncertain data.

3 Components of Randomized Gini Index

In this Section, we investigate to propose a new Randomized Gini Index framework (RGI). Our randomized sampling method depends on small random subset of attributes. We assume that the subset of the training data is small, i.e. it is computationally cheap to act on such a set in a reasonable time. Also, such randomized sampling is done multiple times. We focus on a set of commonly used random sampling procedure and Filter. Next, we try to adapt and deploy them as RGI components. The next stage of RGI tries to consider both gini index and weights for splitting of attributes. The quality of solution fine-tuning, mainly, depends on the nature of the filter involved and the parameters of random sampling. The following four sub sections, detail different design alternatives for both random sampling and filter procedure search for RGI components.

3.1 Random Sampling Method

Due to the large dimensionality of the feature space, it may be difficult for a search method to search appropriate features. In order to increase the computational speed we used random sampling method [12]. Randomized Sampling (RS) is the process of generating random subset datasets from the original dataset where every feature has equal chance. In random sampling we choose a subset of *m* features out of the presented *n* features such that *m* << *n.* In order to cover a large portion of the features in the dataset, we repeat the selection *t* times. The algorithm for random sampling is given in Algorithm 1.

Algorithm 1. Random sampling RS method

Input: n examples each with p features, K randomized experiments.

Output: Count vector W (1xD vector) representing number of times features were selected in K randomized experiment.

Procedure:

```
Select K randomized sets each of size B and denote then as
$N\,exam_i ; i = 1,2,..K$ and let $V \leftarrow 0$
for $i = 1,2,..K$ do
    Get $N\,exam_i$ set;
    Train $Model_i$ = Filter
    Si = selected features in $Model_i$ via $N\,exam_i$
    $V \leftarrow V + \{x, x \in R^D \ |x_j = 1 ifj \in S_i elsex_j = 0\}$
end
```

3.2 Filter for Attribute Selection

Considered as the earliest approaches to feature selection, filter methods discard irrelevant features, without any reference to a data mining technique, by applying independent search which is mainly based on the assessment of intrinsic attribute properties and their relationship with the data set class (i.e. Relief, Symmetrical uncertainty, Pearson correlation, etc)[13].

3.3 Creating Weighted Vectors for Attributes

After many such randomized experiments, the counts of the number of times a feature was found in those randomized experiments is summed up and normalized and denoted by *V.*

This count vector, denoted by *V*, is then inverted and used as weights for the weighted version of the heuristic algorithm; i.e. weights used in the weighted formulations are *W* = *1*=*V*. Intuitively, if a feature is important and is found multiple times via the RS method, then the corresponding weight for the feature is less and thus it is penalized lesser, encouraging higher magnitude for the feature.

The algorithm for inducing new decision tree by using RGI is shown in Algorithm 2,

Algorithm 2. New Decision Tree (D, A, RGI)

```
Input:      D    – Data Partition
            A    – Attribute List
            RGR – Randomized Gini Index (Gain)
Output:     A Decision Tree.
Procedure:
            Create a node N
            If samples in N are of same class, C then
                return N as a leaf node and mark class C
                If A is empty then
                else
                apply Randomized-Gini Gain ( $a_i, S_w$ )
                label root node N as f(A)
                for each outcome j of f(A) do
                subtree j =New Decision Tree(Dj,A,RGI)
                connect the root node N to subtree j
                endfor
            endif
        endif
        Return N
```

3.4 Inducing Decision Trees

The RGI paradigm needs a heuristic function as its base algorithm. In this paper, we combine it with one of the most popular algorithms, CART. CART has proven to be a benchmark against which the performances of machine learning algorithms are measured. As an algorithm it is robust, accurate, fast, and, as an added bonus, it produces a comprehensible structure summarizing the knowledge it induces. We propose to integrate our weighted features in CART heuristic function. In CART, gini index is used as the heuristic function to perform splitting of the nodes at the growing phase. Gini index is an impurity-based criterion that measures the divergence between the probability distributions of the target attribute's values. The Gini index has been used in various works . The Gini is defined as in equation 1 and gini gain can be calculated by using equation 2

$$Gini(y,S) = 1 - \sum_{c_j \in dom(y)} \left(\frac{\sigma_{y=C_j} S}{|S|} \right)^2 \tag{1}$$

Consequently, the evaluation criteria for selecting the attribute a_i is defined as

$$Gini\,Gain(a_i,S)=Gini(y,S)-\sum_{v_{i,j}\in dom(a_i)}\frac{\sigma_{ai=v_{i,j}}S}{|S|}\times Gini\left(y,\sigma_{ai=v_{i,j}}S\right) \qquad (2)$$

The new composite heuristic function Randomized -Gini index can be obtained by using equation (3),

$$Randomized\,Gini\,Gain(a_i,S_W)=Gini(y,S_W)-\sum_{v_{i,j}\in dom(a_i)}\frac{\sigma_{ai=v_{i,j}}S_W}{|S_W|}\times Gini\left(y,\sigma_{ai=v_{i,j}}S_W\right) \qquad (3)$$

4 Experiments on Synthetic and Real World Datasets

We performed the implementation of our new algorithm within the Weka [14] environment on windows XP with 2Duo CPU running on 2.53 GHz PC with 2.0 GB of RAM. As we mentioned, the RGI paradigm need a base learner classification algorithm. Here, we combine it with one of the most popular algorithms, CART. In this paper, for the process of new heuristic function, we have generated random samples of 5 trails for each original dataset. In second stage, we used a filter approach, to find a set of relevant features and in last stage, we exploited CART's gini index to induce decision trees. In order to test the feasibility of the proposed heuristic function, we have carried out a number of experiments on artificially generated data set, as well as real-world data sets. We choose 40 UCI [15] datasets, which are commonly used in the supervised learning research area. The details of each data set are available in Table 2. We conducted experiments by using 10 fold cross validation for 10 runs to test the performance of various methods. The summary of instantiation of the search instances for RGI is given in Table 1.

Table 1. The Summary of instantiation of the search instances for RGR

State	RGI
Initial State	The empty set of features(0,0,0 . . . 0)
Evaluator	Filter
Learning Scheme	CART
Search algorithm	Hill-climbing or best-first search
Search termination	5

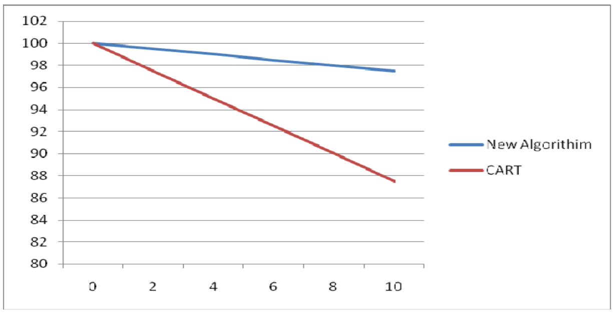

Fig. 1. Experimental results for artificial data on both algorithms

4.1 Synthetic Dataset

First, we tested our method on these artificial data generated by using XOR problem and we added artificial noise in the class labels to study its robustness. The noisy version of each training data set is generated by choosing 2% instances and changing their class labels to other incorrect labels randomly up to 10 %. The experimental results show that our new algorithm can significantly outperform CART. The problem

Table 2. The properties of the 40 UCI datasets

S.No	Dataset	Inst.	Missing values	Numeric. attributes	Nominal attributes	Classes
1.	Anneal	898	no	6	32	5
2.	Anneal.ORIG	898	yes	6	32	5
3.	Arrhythmia	452	yes	206	73	13
4.	Audiology	226	yes	0	69	24
5.	Autos	205	yes	15	10	6
6.	Balance-scale	625	no	4	0	3
7.	Breast-cancer	286	yes	0	9	2
8.	Breast-w	699	yes	9	0	2
9.	Colic-h	368	yes	7	15	2
10.	Colic-h.ORIG	368	yes	7	15	2
11.	Credit-a	690	yes	6	9	2
12.	Credit-g	1000	no	7	13	2
13.	Pima diabetes	768	no	8	0	2
14.	Ecoli	336	no	7	0	8
15.	Glass	214	no	9	0	6
16.	Heart-c	303	yes	6	7	2
17.	Heart-h	294	yes	6	7	2
18.	Heart-statlog	270	no	13	0	2
19.	Hepatitis	155	yes	6	13	12
20.	Hypothyroid	3772	yes	7	22	4
21.	Ionosphere	351	no	34	0	2
22.	Iris	150	no	4	0	3
23.	Kr-vs-kp	3196	no	0	36	2
24.	Labor	57	yes	8	8	2
25.	Letter	20000	no	16	0	26
26.	Lympho	148	no	3	15	4
27.	Mushroom	8124	yes	0	22	2
28.	Optdigits	5620	no	64	0	10
29.	Pendigits	10992	no	16	0	10
30.	Primary-tumor	339	yes	0	17	21
31.	Segment	2310	no	19	0	7
32.	Sick	3772	yes	7	22	2
33.	Sonar	208	no	60	0	2
34.	Soybean	683	yes	0	35	19
35.	Splice	3190	no	0	61	3
36.	Vehicle	846	no	18	0	4
37.	Vote	435	yes	0	16	2
38.	Vowel	990	no	10	3	11
39.	Waveform	5000	no	41	0	3
40.	Zoo	101	no	1	16	7

which we chose in artificial data is the XOR problem. We have created training and testing data contains 64 sample set of the patterns below. {0,0,0; 0,1,1; 1,0,1; 1,1,1}.After that we added ten random binary features to both training and testing data, and observed the performance of both C4.5 and our new algorithm. The experimental results conducted with artificial domains for both C4.5 and new algorithm, are summarized in Fig 1.

One can observe from the results that the accuracy of CART decreases drastically by the addition of irrelevant features. The accuracy of CART has reduced from 100 % to 87 %, where as the accuracy of new algorithm have simply changed from 100 % to 97 %, indicating that new algorithm have removed almost all the irrelevant attributes.

4.2 Real World UCI Datasets

The second type of experiments examined the ability of the algorithms in natural domains. In experiments with natural domain, we could neither vary nor measure the number of irrelevant features in these domains, we could make educated guesses about the prevalence of irrelevant features by comparing the patterns of results to those found with artificial data.

5 Experimental Results

5.1 Results on UCI Datasets

In this section, the result of the comparative analysis based on accuracy between RGR and other traditional bench mark splitting criteria's is shown. The experimental results are summarized in Table 3 to Table 6 and in Figure 2 to Figure 5. In the below tables, GR represents the gain ratio, IG represents information gain, GI represents gini index used in CART, and RGI represents Randomized gini index. The reason that these three traditional splitting criteria's are selected is that they are among the most popular splitting criteria's and many researchers used them in comparative analysis for new splitting criteria in decision trees [16], [17],[18].

Table 3 shows the detailed experimental results of the mean classification accuracy and of Gain Ratio, Information Gain, Gini Index and Randomized Gini Index method using CART as base classifier on each data set. And the mean values, overall ranks and the pairwise *t*-test results are summarized in the Table 5. From Table 3 we can see that RGI can achieve substantial improvement over GR on most data set (11 wins and 3 losses) which suggests that RGI is potentially a good heuristic function for decision trees. RGI also gain significantly improvement over IG (7 wins and 2 losses) and is comparable to GI(6 wins and 4 losses) .The overall rank of RGI on these 40 data sets is 2.22 which the smallest among all these ensemble methods. Thus , compared to RG, IG and GI which generates a decision tree with noise branch due to the presence of noise data in the datasets in the form of irrelevant features.

Table 3. Accuracy of Gain Ratio, Information Gain, Gini Index and Randomized Gini Index on the 40 UCI data sets. Comparative analysis based on accuracy between Gain Ratio, Information Gain, Gini Index and Randomized Gini index.

Dataset	**G**	**IG**	**GI**	**RG**
	R			**I**
Anneal	98.38	96.89	98.25	98.50
Anneal.ORIG	90.30	90.61	91.82	93.23
Arrhythmia	65.12	65.54	71.31	71.33
Audiology	77.90	71.86	74.25	75.30
Autos	73.02	52.31	74.65	77.24
Balance-scale	78.19	77.63	78.73	72.29
Breast-cancer	71.23	66.96	70.22	94.74
Breast-w	94.30	94.01	94.74	94.74
Colic	84.91	84.43	85.37	85.64
Colic.ORIG	66.40	66.84	66.92	70.32
Credit-a	85.93	85.10	85.95	85.04
Credit-g	71.97	71.62	72.63	74.38
Pima_diabetes	73.46	74.82	74.88	74.61
Ecoli	80.62	79.42	81.03	83.17
Glass	67.32	63.57	68.91	71.91
Heart-c	77.72	74.12	78.02	78.85
Heart-h	78.29	78.44	78.59	79.16
Heart-statlog	76.89	76.57	76.73	74.89
Hepatitis	78.14	79.76	78.99	78.11
Hypothyroid	99.49	99.33	99.54	98.83
Ionosphere	88.80	89.22	88.63	88.72
Iris	94.12	94.90	94.61	93.80
Kr-vs-kp	99.18	98.70	99.14	94.02
Labor	79.38	77.81	81.24	80.13
Letter	86.23	82.16	85.12	87.24
Lympho	76.68	72.82	77.19	72.87
Mushroom	100.0	99.96	99.93	100.0
Optdigits	89.42	87.72	89.09	90.41
Pendigits	95.97	94.39	95.45	96.16
Primary-tumor	39.10	37.80	41.13	38.76
Segment	96.07	94.57	95.02	96.03
Sick	98.72	98.46	98.60	98.13
Sonar	71.12	68.77	71.26	71.80
Soybean	88.48	79.30	90.27	90.89
Splice	93.55	52.20	84.34	90.38
Vehicle	70.31	68.91	69.60	67.77
Vote	95.54	95.10	95.06	95.77
Vowel	75.59	50.52	74.39	78.23
Waveform	75.24	76.05	76.67	77.23
Zoo	93.26	40.54	40.54	40.61

Furthermore, RGI is comparable to C4.5 (GR) which is the state-of-the-art decision tree technique. Note that RGI have performed well on C4.5. Table 4 shows the detailed experimental results of the mean tree size and of Gain Ratio, Information Gain, Gini Index and Randomized Gini Index method using CART as base classifier on each data set. And the mean values, overall ranks and the pairwise *t*-test results are summarized in the Table 5.

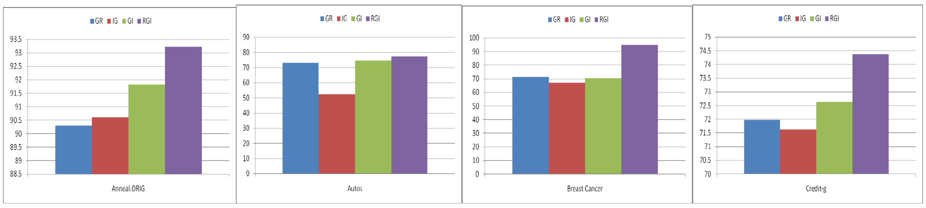

Fig. 2. Test results on accuracy between the Gain Ratio, Information Gain, Gini Index, and Randomized Gini index on anneal.orig, autos, breast-cancer, and credit-g datasets

Table 4. Tree Size of Gain Ratio, Information Gain, Gini Index and Randomized Gini Index on the 40 UCI data sets. Comparative analysis based on accuracy between Gain Ratio, Information Gain, Gini Index and Randomized Gini index.

Dataset	**GR**	**IG**	**GI**	**RGI**
Anneal	43.20	32.15	21.02	21.70
Anneal.ORIG	55.45	53.45	76.10	96.01
Arrhythmia	63.00	18.10	16.60	67.28
Audiology	41.95	28.50	35.82	35.14
Autos	51.55	36.25	43.10	48.86
Balance-scale	62.20	34.80	44.60	32.92
Breast-cancer	16.45	23.65	7.16	7.72
Breast-w	17.50	9.80	15.88	15.88
Colic	9.05	13.55	6.42	5.00
Colic.ORIG	1.00	220.7	54.57	70.32
Credit-a	32.65	19.25	4.60	9.50
Credit-g	96.80	52.95	27.40	26.74
Pima_diabetes	35.70	22.50	16.40	16.02
Ecoli	27.00	13.30	18.00	22.72
Glass	35.20	15.10	21.60	23.92
Heart-c	29.50	14.90	12.50	17.82
Heart-h	17.80	11.40	9.90	16.56
Heart-statlog	28.40	10.70	14.20	15.70
Hepatitis	11.90	5.40	7.60	83.04
Hypothyroid	21.85	16.95	16.80	13.30
Ionosphere	19.30	89.22	8.30	11.98
Iris	6.70	5.50	5.80	8.42
Kr-vs-kp	48.60	44.70	51.20	12.26
Labor	5.30	4.90	7.20	8.22
Letter	1922	984	1702	2203.7
Lympho	19.65	10.40	10.70	17.02
Mushroom	29.85	37.60	13.20	16.28
Optdigits	89.42	87.72	89.09	278.4
Pendigits	95.97	94.39	95.45	351.78
Primary-tumor	39.10	37.80	41.13	30.96
Segment	96.07	94.57	95.02	78.30
Sick	98.72	98.46	98.60	16.96
Sonar	71.12	68.77	71.26	14.66
Soybean	88.48	79.30	90.27	100.60
Splice	93.55	52.20	84.34	90.38
Vehicle	70.31	68.91	69.60	53.36
Vote	95.54	95.10	95.06	9.40
Vowel	75.59	50.52	74.39	168.70
Waveform	75.24	76.05	76.67	168.74
Zoo	93.26	40.54	40.54	1.00

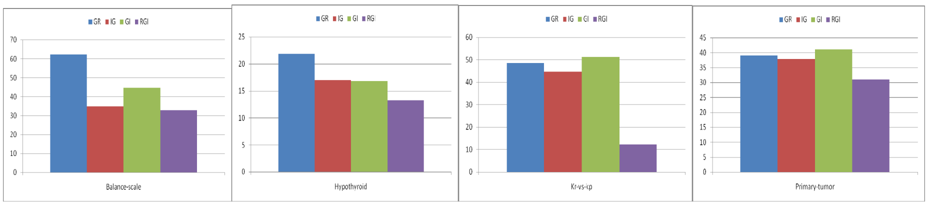

Fig. 3. Test results on tree size between the Gain Ratio, Information Gain, Gini Index, and Randomized Gini index on balance-scale, hypothyroid, kr-vs-kp and primary-tumor datasets

From Table 4 we can see that RGI can achieve substantial improvement over GR on most data set (11 wins and 3 losses) which suggests that RGR is potentially a good heuristic function for decision trees. RGI also gain significantly improvement over IG (7 wins and 2 losses) and is comparable to GI(6 wins and 4 losses). From Table 3, we can also see that RGI is significantly better than remaining benchmark algorithms.

Table 5. Win-Tie-Loss (*w/t/l*) comparisons between Randomized Gini Index against other algorithms using pairwise *t*-tests at 95% significance level, respectively

Results	**Systems**	**Wins**	**Ties**	**Losses**
Accuracy	RGR vs. GR	9	14	17
	RGR vs. IG	19	12	9
	RGR vs. GI	13	13	14
Tree Size	RGR vs. GR	33	2	5
	RGR vs. IG	19	3	17
	RGR vs. GI	19	2	18

6 Conclusion and Future Work

In this article, we propose a frame work to generate decision trees with better accuracy. Our algorithm is also effective to solve the problems discussed in gini index. We performed a series of primary and comparative experiments on 40 real-world data sets, and our method obtained encouraging results. The applications of this RGI in real-world learning tasks, especially medical data analysis tasks, will be fruitful.

There are many problems for future research. First, Analysis must be performed to ensure that the heuristic function still have near-optimal sample complexity. Secondly, we also hope to forge a stronger link between our studies of natural and artificial domain for our pruning technique.

References

[1] Zhao, H., Sinha, A.P.: An Efficient Algorithm for Generating Generalized Decision Forests. IEEE Transactions on Systems, Man, and Cybernetics —Part A: Systems and Humans 35(5), 287–299 (2005)

[2] Hu, J., Deng, J., Sui, M.: A New Approach for Decision Tree Based on Principal Component Analysis. In: Proceedings of Conference on Computational Intelligence and Software Engineering, pp. 1–4 (2009)

[3] Liu, D., Lai, C., Lee, W.: A Hybrid of Sequential Rules and Collaborative Filtering for Product Recommendation. Information Sciences 179(20), 3505–3519 (2009)

[4] Mitchell, T.M.: Machine Learning. McGraw Hill, New York (1997)

[5] Mahmood, A.M., Kuppa, M.R., Reddi, K.K.: A new decision tree induction using composite splitting criterion. Journal of Applied Computer Science & Mathematics 9(4), 69–74 (2010)

[6] Aviad, B., Roy, G.: Classification by Clustering Decision Tree-like Classifier based on Adjusted Clusters. Expert Systems with Applications (2011), doi:10.1016/j.eswa.2011.01.001

[7] Chang, P.-C., Fan, C.-Y., Lin, J.-L.: Trend discovery in financial time series data using a case based fuzzy decision tree. Expert Systems with Applications 38, 6070–6080 (2011)

[8] Khanli, L.M., Mahan, F., Isazadeh, A.: Active rule learning using decision tree for resource management in grid computing. In: Future Generation Computer Systems (2011), doi:10.1016/j.future.2010.12.016

[9] Mahmood, A.M., Kuppa, M.R.: A novel pruning approach using expert knowledge for data-specific pruning. Engineering with Computers (2011), doi:10.1007/s00366-011-0214-1

[10] Wang, Y.: The Cascade Decision-tree Improvement Algorithm Based on Unbalanced Data Set. In: Proceedings of 2010 International Conference on Communications and Mobile Computing, pp. 284–288 (2010)

[11] Tsang, S., Kao, B., Yip, K.Y., Ho, W.-S., Lee, S.D.: Decision Trees for Uncertain Data. IEEE Transactions on Knowledge and Data Engineering 23(1), 64–78 (2011)

[12] Politos, D.N., Romano, J.P., Wolf, M.: Subsampling. Springer, Heidelberg (1999)

[13] Liu, H., Motoda, H.: Computational methods of feature selection. Chapman and Hall/CRC Editions (2008)

[14] Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)

[15] Asuncion, A., Newman, D.: UCI machine learning repository (2007), `http://www.ics.uci.edu/~mlearn/MLRepository.html`

[16] Li, N., Zhao, L., Chen, A.-X., Meng, Q.-W., Zhang, G.-F.: A New Heuristic of the Decision Tree Induction. In: Proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, pp. 1659–1664 (2009)

[17] Qi, C.: A New Partition Criterion for fuzzy Decision Tree Algorithm. In: Proceedings of Workshop on IntelligentInformation Technology Application, pp. 43–46 (2007)

[18] Mahmood, A.M., Kuppa, M.R.: Early Detection of Clinical Parameters in Heart Disease Using Improved Decision Tree Algorithm. In: Proceedings of IEEE 2nd Vaagdevi International Conference on Information Technology for Real World Problems (VCON 2010), Warangal, India, pp. 24–29 (2010)