A New Rough Set Based Classification Rule Generation Algorithm(RGA)

Honghai Feng[1,2], Yanyan Chen[3], Kaiwei Zou[1], Lijuan Liu[1], Qiannan Zhu[1], Zhuo Ran[1], Li Yao[1], Lijin Ji[1], and Sai Liu[1]

[1] School of Computer and Information Engineering
[2] Institute of Data and Knowledge Engineering
[3] Library
Henan University, Kaifeng, Henan, 475001, China
honghf@126.com

Abstract. Rough sets theory has taken an important role in data mining. This paper introduces a new rough set based classification rule generation algorithm. It has three features: the first is that the new algorithm can be used in inconsistent systems. The second is its ability to calculate the core value without attributes reduction before. The third is that every example gives a rule and the core values are added first in rule generation process. Experimental results indicate that the classification performanceismuch better than the standard rough set, its variants andJRIPPER, a little better thanCBA and KNN,andcompetive to C4.5in terms of 8 measures. The higher performance of the new algorithm may get benefit from its enough higher accuracy rules and having some properties like KNN.

Keywords: Rough sets, classification rule, C4.5, JRIPPER, CBA.

1 Introduction

In artificial intelligenceone of the main predicting tasks is classification.Accuracy is always a concern. Even though many machine learning theories can, the rule based methods are indispensable and have their own merits. Firstly, the classification accuracy of the rule based methods is comparable to the top classification algorithms. As a well-known fact, C4.5 [1] and JRIPPER [2] are good classifiers in classification accuracy. Secondly, a rule based method is a white box, which can induce understandable knowledge that may be essential in special area such as medical fields. Thirdly, rule based classification methods are validated andeffective in practical applications.

The main rule based techniques are decision trees, sequential covering, associative rules, and rough set [3] based methods, representative algorithms or softwares are C4.5, RIPPER, CBA [4], and ROSETTA [5].

By running the free software WEKA, we can find that C4.5 and RIPPER are two excellent classifiers in classification performance; meanwhile there are some other rule based algorithms comparable to them in classification accuracy [6-10]including the rough sets based classifiers.

A. Moonis et al. (Eds.): IEA/AIE 2014, Part I, LNAI 8481, pp. 369–378, 2014.

In the standard rough set theory, attribute reduction only preserves the dependencies of decision attributes on condition attributes in consistent examples, i.e., new inconsistencies will not be allowed to be brought about after attribute reduction only in the consistent examples. This paper introduces a new rough set based classification rule generation algorithm that is an extension of the standard rough set method to inconsistent systems. The new algorithm can preserve the dependencies of decision attributes on condition attributes in all examples including inconsistent examples, i.e., new inconsistencies will not be allowed to be brought about after attribute reduction in all examples. In addition, in standard rough set method an attribute reduction step should be carried out before rule generation, which may remove some important attributes and some attribute values may become indispensable that may not in original attribute set. In order to get better predictive performance the new algorithm generates the core value for every example first and then generates rule on basis of it directly. In the course of rule generation ofthe new algorithm the core values are added first and every example generates one rule.Meanwhile, the new algorithm does not reduce attributes before rule generation and the sub-optimum attribute reducts can being gotten by the way after rule generation.

2 New Rough Set Based Classification Rule Induction Algorithm

Table 1. Algorithm RGA

Algorithm RGA
Input: Data set U, condition attribute set C, decision attribute set D
Output: Labels of every test examples
begin
getIND(A, I){
$IND(A,I)=\phi$
for each attribute $a \in A$
for each example $x \in I$
$IND(A,I)=\{x\}$
for each example y ($y \neq x$);
if $a(x)=a(y)$
add $y$to $IND(A,I)$
return $IND(A,I)$
}
getInconsistentExamples(){
$For\ every\ x$

Table 1. (*continued*)

Calculate $\underset{i}{\cap}[x]_{C_i}$ and $[x]_D$

If $\underset{i}{\cap}[x]_{C_i} \not\subset [x]_D$

x is an inconsistent example

}

getCoreValue1(){

Get all the consistent examples

For every consistent example

For every condition attribute C_i

Calculate $\bigcap_{i=1,\ i\neq j}^{n}[x]_{C_i}$ and $[x]_D$

If $\bigcap_{i=1,\ i\neq j}^{n}[x]_{C_i} \not\subset [x]_D$

$C_j(x)$ is a core value

}

getCoreValue2(){

Get all the inconsistent examples;

For every inconsistent example

For every condition attribute C_i,

Calculate $\bigcap_{i=1,\ i\neq j}^{n}[x]_{C_i}, [x]_D$ and get $\{x', \cdots, x^{(n)}\}$, $\{y, y', \cdots, y^{(n)}\}$

If $\bigcap_{i=1,\ i\neq j}^{n}[x]_{C_i} \not\subset [x]_D \cup \{x', \cdots, x^{(n)}\} \cup \{y, y', \cdots, y^{(n)}\}$

$C_j(x)$ is a core value

}

getRules(){

rule set $RULES = \varphi$

rule $RULE = \varphi$

condition attribute set A

for every example x

$RULE = \wedge(C_i = C_i(x))$ where $C_i(x) \in CORE(x)$

}

classifyInstance(){

matched rules $MRS = \phi$

Table 1. (*continued*)

for every test example x

for every rule r in $RULES$

if r matches x

add r to MRS

if $T=|MRS|\neq\phi$

$H(x)=\text{sign}\sum_{t=1}^{T}\alpha_t r_t(x)$, where α =coverage of $r_t(x)$

else

$H(x)$ assigns the class label that the majority examples hold originally

}

end

For every example x, rule $r_t(x)$ outputs only one class label which the example x belongs to with a biggest confidence value. The total time complexity of the new algorithm is $O(mn^2)$ where m is the number of the condition attributes, and n is the number of the examples.

3 Data Sets

In order to get faithful results 77 data sets are used in this experiment, and all the data sets are obtained from the repository of Machine Learning databases at UCI [11], see their characteristics in Table 2. Some data sets are discretized by supervised discretization methods with WEKA and denoted as like australian_dis, and some data sets are discretized by unsupervised discretization methods with WEKA and denoted as like autos_undis. The java class weka.filters.supervised.attribute. AttributeSelection in WEKA is used for supervised discretization and discretization is by Fayyad &Irani's MDL method (the default) [12]. weka.filters.unsupervised.attribute.Discretize is used for unsupervised discretization, and discretization is by simple binning. The default value of bins is 10.

Table 2. Data sets

Data sets	features	classes	cases
adult-stretch	4	2	20
audiology	69	224	24
australian_dis	14	2	690
autos_undis	24	7	205
balance-scale_sup	4	3	625
blood_tranfusion	3	2	748

Table 2. (*continued*)

breastCancer	9	2	286
b-c-w(Prognostic)	8	2	699
b-c-w-image	32	2	196
bridges_dis	9	4	105
bridges_version2	11	7	107
car	6	4	1728
cleve_dis	11	2	303
cmc_dis	9	3	1473
colic_sup_missing	16	2	368
cpu	8	8	209
crx_dis	15	2	690
cylinder-bands	31	2	540
Dermatology_dis	34	6	366
diabetes_sup	6	2	768
echocardiogram	11	3	132
ecoli_sup	6	8	336
flag_dis	26	8	194
flare_data1	12	6	323
flare_data2	12	6	1066
german_dis	19	2	1000
glass_undis	7	6	214
haberman_unsup	3	2	306
hayes-roth_dis	4	3	132
heart-c_sup	11	2	303
heart-h_unsup	12	2	294
heart-statlog_sup	9	2	270
hepatitis_unsup	19	2	155
ionosphere_sup	33	2	351
iris_dis	4	3	150
labor_dis	16	2	57
led7	7	10	3200
led_24	24	10	1000
lenses_dis	4	3	24
liverdisorders_unsup	6	2	345
lung-cancer	56	3	32
Lymphography	18	4	148
mammo _dis	5	2	961
molecular-biology	57	2	106
monks1	6	2	432
monks2	6	2	432
monks3	6	2	432
new_thyroid_dis	5	3	215
post-operative	8	3	90
primary-tumor	17	21	339
promoter_gene	57	2	106

Table 2. (*continued*)

Robot_FailureLP4_dis	90	3	118
Robot_FailureLP5_dis	90	5	165
shuttle-landing	6	2	15
solar-flare_1	11	6	323
solar-flare_2	11	6	1066
sonar_unsup	60	2	208
soybean_unsupmissing	35	19	683
space_shuttle_disun	2	3	23
spect_train	22	2	80
sponge	44	3	76
tae	2	3	151
Teaching Assistant	4	3	151
tic tac toe	9	2	958
trains	32	2	10
urinary	6	4	120
Vehicle_dis	18	4	846
vote_unsup_missing	16	2	435
vowel	11	11	900
wine	13	3	178
yeast_dis	8	10	1484
yellow-small	4	2	20
zoo	16	7	101
Arrhythmia_supdis	133	13	452
b-c-w-cell	272	2	569
libras_movement_dis	74	15	360
Mammals_unsup	464	4	1000
Spectrometer	93	48	531

4 Experimental Results

The new algorithm (RGA), CBA, Explore, LEM2, the standard rough set methods with genetic selecting attribute before rule generation(SRGeS) and the variable precision rough set(VPR) are programmed with JAVA and embedded into WEKA 3.6.5. The C4.5 and Jrip are transformed from J48 and Jrip in WEKA 3.6.5. The KNN(IB1, K=1) is from WEKA and with no modification and transformation.

The experiment uses a ten-fold cross validation procedure that performs 10 randomized train and test runs on the dataset.

The experimental results in term of mean absolute error are listed in Table 3. VPR and SRGeS represent variable precision rough set algorithm and standard rough set method with genetic selecting attribute respectively. The detailed experimental results about every algorithm on every data set in terms of other measures like percent correct, weighted average area under ROC, weighted average F-measure, weighted average IR precision, weighted average IR recall, weighted average true negative rate, and weighted average true positive rate have not been offered, but the comprehensive results are provided in Table 4. The first line of Table 4 lists the 8 algorithms except for RGA, the first column represents the 8 performance measures and other 6 metrics for analysis and the others stand for comparison of performances in terms of 8 measures across 9 algorithms. The count (xx/ yy/ zz) of the number of times represents that the other listed schemes are bigger than (xx), the same as (yy), or smaller than (zz) the baseline scheme (the new algorithm, RGA).

Table 3. Mean_absolute_error results

Data sets	RGA	CBA	Explore	C4.5	Jrip	LEM2	VPR	SRGe	KNN
adult-stretch	0.01 |	0.00	0.02	0.00	0.24 v	0.00	0.04	0.53 v	0.10
audiology	0.02 |	0.03 v	0.06 v	0.02	0.05 v	0.02	0.02	0.05 v	0.02
australian	0.16 |	0.14	0.19	0.20 v	0.36 v	0.23 v	0.25 v	0.24 v	0.20 v
autos	0.06 |	0.05	0.07	0.07	0.14 v	0.07	0.08	0.18 v	0.04 *
balance	0.19 |	0.21 v	0.34 v	0.27 v	0.28 v	0.19	0.31 v	0.19	0.20
transfusion	0.31 |	0.27 *	0.34 v	0.36 v	0.37 v	0.31	0.33	0.31	0.34
b-c-w-w	0.07 |	0.05 *	0.05	0.08	0.27 v	0.08	0.07	0.09	0.04
b-c-w-d	0.38 |	0.37	0.24 *	0.36	0.36	0.49 v	0.32	0.34	0.36
breast-cancer	0.32 |	0.34	0.32	0.36	0.41 v	0.30	0.37	0.36	0.34
bridges	0.11 |	0.10	0.15	0.13	0.20 v	0.14	0.13	0.21 v	0.12
bridges2	0.14 |	0.10 *	0.12	0.14	0.16	0.14	0.14	0.19 v	0.13
car	0.05 |	0.05	0.05	0.04	0.19 v	0.05	0.08 v	0.05	0.11 v
cleve	0.19 |	0.20	0.21	0.31 v	0.37 v	0.24	0.35 v	0.21	0.21
cmc	0.34 |	0.34	0.36 v	0.36 v	0.41 v	0.34	0.37 v	0.34	0.35
horse-colic	0.17 |	0.19	0.21	0.24 v	0.36 v	0.24	0.27 v	0.27 v	0.26 v
cpu	0.08 |	0.07	0.09	0.09	0.13 v	0.08	0.09	0.10	0.08
crx	0.17 |	0.14 *	0.18	0.19	0.36 v	0.23 v	0.31 v	0.35 v	0.20
cylinder	0.19 |	0.23	0.29 v	0.41 v	0.41 v	0.25	0.21	0.37 v	0.21
Dermatology	0.04 |	0.03	0.08 v	0.03	0.16 v	0.03	0.04	0.22 v	0.02 *
pima_diabetes	0.28 |	0.25 *	0.34 v	0.31 v	0.40 v	0.29	0.38 v	0.28	0.29
echocard	0.34 |	0.28	0.31	0.27	0.36	0.32	0.35	0.33	0.29
ecoli	0.05 |	0.05	0.08 v	0.06	0.13 v	0.05	0.10 v	0.05	0.06
flags	0.11 |	0.11	0.13	0.12	0.14 v	0.12	0.11	0.17 v	0.11
flare_data1	0.10 |	0.10	0.11	0.11	0.14 v	0.12 v	0.12	0.14 v	0.11
flare_data2	0.10 |	0.10	0.11 v	0.10 v	0.19 v	0.11 v	0.11 v	0.15 v	0.11
german	0.29 |	0.26	0.35 v	0.34 v	0.42 v	0.30	0.31	0.35 v	0.31
Glass	0.10 |	0.08 *	0.12 v	0.10	0.15 v	0.10	0.14 v	0.10	0.09
haberman	0.33 |	0.31	0.39	0.38 v	0.39 v	0.32	0.35	0.33	0.32
hayes-roth	0.10 |	0.09	0.18	0.13	0.18 v	0.15 v	0.20 v	0.10	0.15
heart-c	0.08 |	0.08	0.08	0.11 v	0.15 v	0.10	0.15 v	0.08	0.09
heart-h	0.09 |	0.08	0.09	0.12	0.15 v	0.10	0.10	0.16 v	0.08
heart-s	0.19 |	0.17	0.20	0.24	0.36 v	0.19	0.38 v	0.19	0.19
hepatitis	0.15 |	0.19	0.20	0.27 v	0.34 v	0.26 v	0.16	0.30 v	0.23 v

Table 3. (*continued*)

ionosphere	0.10	0.09	0.08	0.13	0.29 v	0.11	0.09	0.20 v	0.07
iris	0.05	0.05	0.06	0.06	0.20 v	0.05	0.06	0.05	0.04
labor	0.13	0.14	0.24	0.17	0.34 v	0.23	0.11	0.39 v	0.08
led7	0.08	0.07 *	0.13 v	0.08	0.10 v	0.07 *	0.08	0.08	0.08
LED_24	0.07	0.12 v	0.18 v	0.07	0.12 v	0.10 v	0.07	0.11 v	0.11 v
lenses	0.25	0.20	0.21	0.15 *	0.31	0.31	0.20	0.25	0.24
liver-dis	0.39	0.31 *	0.37	0.45	0.48 v	0.40	0.47 v	0.39	0.37
lung-cancer	0.41	0.43	0.39	0.34	0.39	0.48	0.44	0.39	0.41
lymphography	0.11	0.10	0.14	0.13	0.21 v	0.12	0.09	0.19 v	0.10
mammographic	0.23	0.21	0.24	0.25 v	0.37 v	0.24	0.30 v	0.23	0.24
promoters	0.33	0.14 *	0.38	0.20	0.35	0.39	0.41	0.49	0.16
monks1	0.00	0.00	0.21 v	0.04	0.29 v	0.00	0.15 v	0.37 v	0.28 v
monks2	0.47	0.32 *	0.43 *	0.44	0.44	0.44	0.49	0.47	0.45
monks3-weka.	0.00	0.00	0.02	0.00	0.26 v	0.00	0.14 v	0.48 v	0.20 v
new_thyroid	0.03	0.04	0.04	0.06	0.18 v	0.03	0.07	0.04	0.02
postoperative	0.28	0.27	0.24 *	0.28	0.28	0.25	0.27	0.27	0.28
primary-tumor	0.06	0.06	0.07 v	0.06	0.07 v	0.06	0.06 v	0.06	0.06
promoter_gene	0.37	0.13 *	0.42	0.24	0.36	0.26	0.41	0.51	0.16 *
Robot_F_LP4	0.11	0.10	0.15	0.08	0.21 v	0.12	0.11	0.24 v	0.05 *
Robot_F_LP5	0.15	0.18	0.15	0.12	0.19 v	0.17	0.15	0.25 v	0.11 *
Shuttle	0.30	0.55	0.25	0.41	0.42	0.45	0.37	0.46	0.30
solar-flare	0.10	0.10	0.11	0.11	0.17 v	0.11 v	0.11 v	0.15 v	0.11
sonar	0.41	0.37	0.47	0.36	0.44	0.39	0.44	0.45	0.21 *
soybean	0.01	0.02	0.07 v	0.01 *	0.03 v	0.01 *	0.01	0.08 v	0.01 *
space_shuttle	0.18	0.17	0.19	0.27	0.27	0.17	0.26	0.18	0.17
spect	0.38	0.34	0.32	0.37	0.44	0.43	0.45	0.42	0.46
sponge	0.07	0.07	0.05	0.08	0.10	0.09	0.04	0.06	0.04
tae	0.39	0.38	0.44 v	0.41 v	0.44 v	0.41 v	0.42 v	0.39	0.39
Teaching	0.33	0.35	0.36	0.38	0.42 v	0.33	0.37	0.33	0.34
tic-tac-toe	0.08	0.00 *	0.19 v	0.17 v	0.24 v	0.02 *	0.08	0.14 v	0.18 v
trains	0.51	0.30	0.66	0.20	0.25	0.20	0.36	0.71	0.40
urinary	0.00	0.00	0.15 v	0.00	0.19 v	0.00	0.00	0.12 v	0.00
vehicle	0.14	0.15	0.31 v	0.16	0.26 v	0.16	0.21 v	0.23 v	0.15
vote	0.06	0.05	0.08	0.06	0.28 v	0.06	0.13	0.12 v	0.08
vowel	0.04	0.06 v	0.16 v	0.05 v	0.09 v	0.03	0.05 v	0.04	0.03 *
wine	0.01	0.02	0.04	0.05 v	0.22 v	0.03	0.01	0.22 v	0.01
yeast	0.10	0.10	0.13 v	0.11	0.13 v	0.10	0.12 v	0.15 v	0.10
yellow-small	0.00	0.00	0.02	0.00	0.24 v	0.00	0.09	0.00	0.15
zoo	0.01	0.02	0.06 v	0.02	0.12 v	0.03	0.02	0.12 v	0.01
Arrhythmia	0.05	0.07 v	0.05	0.04	0.07 v	0.04	0.07 v	0.08 v	0.04
breast-c-w-c	0.04	0.04	0.04	0.06	0.26 v	0.07	0.04	0.11 v	0.04
libras_m	0.04	0.12 v	0.13 v	0.05	0.08 v	0.04	0.04	0.11 v	0.03 *
Mammals	0.00	0.00	0.00	0.00	0.10 v	0.00	0.00	0.02 v	0.00
spectrometer	0.02	0.03 v	0.03 v	0.02	0.03 v	0.02	0.02	0.04 v	0.02 *
(v/ /*)	(7/58/12	(25/49/3	(18/57/2	(62/15/0)	(10/64/3	(25/52/0	(38/39/0)	(8/59/10)	

Table 4. Comparison of performances in term of 8 measuresacross 9 algorithms

	CBA	Explore	C4.5	Jrip	LEM2	VPRS	SRGeS	KNN
1.	(7/58/12)	(25/49/3)	(18/57/2)	(62/15/0)	(10/64/3)	(25/52/0)	(38/39/0)	(8/59/10)
2.	(4/67/6)	(1/55/21)	(8/64/5)	(3/65/9)	(3/62/12)	(0/63/14)	(1/49/27)	(6/55/16)
3.	(8/61/8)	(7/47/23)	(16/56/5)	(8/63/6)	(5/56/16)	(5/65/7)	(2/58/17)	(3/49/25)
4.	(5/64/8)	(1/49/27)	(7/65/5)	(3/64/10)	(3/65/9)	(0/63/14)	(0/47/30)	(7/56/14)
5.	(5/63/9)	(1/52/24)	(3/67/7)	(1/66/10)	(1/67/9)	(0/63/14)	(0/48/29)	(5/59/13)
6.	(4/67/6)	(1/55/21)	(8/64/5)	(3/65/9)	(3/62/12)	(0/63/14)	(1/49/27)	(6/55/16)
7.	(5/64/8)	(3/51/23)	(11/60/6)	(4/57/16)	(4/63/10)	(0/53/24)	(0/44/33)	(7/61/9)
8.	(4/67/6)	(1/55/21)	(8/64/5)	(3/65/9)	(3/62/12)	(0/63/14)	(1/49/27)	(6/55/16)
9.	(0/1/76)	(46/6/25)	(5/9/63)	(0/0/77)	(23/8/46)	(10/30/37)	(12/48/17)	
10.	(3/48/26	(16/44/17)	(0/15/62)	(1/12/64)	(7/59/11)	(0/76/1)	(0/27/50)	
11.	(2/9/66)	(41/8/2)	(23/17/37)	(0/1/76)	(54/9/14)	(1/28/48)	(15/42/20)	
12.	(73/4/0)	(50/9/18)	(50/14/13)	(77/0/0)	(42/23/12)	(27/30/10)	(4/46/27)	
13.	(17/30/30)	(10/45/22)	(0/6/71)	(0/1/76)	(0/37/40)	(1/30/46)	(2/73/2)	
14.	(0/2/75)	(47/6/24)	(4/8/65)	(0/0/77)	(16/6/55)	(14/38/25)	(11/49/17)	

1.Mean_absolute_error. 2. Percent_correct.
3. Weighted_avg_area_under_ROC. 4. Weighted_avg_F_measure.
5. Weighted_avg_IR_precision6. Weighted_avg_IR_recall
7. Weighted_avg_true_negative_rate 8. Weighted_avg_true_positive_rate
9.Total_Length_of_All_Rules in the rule set 10.amount of attributes in the rule set
11.Mean length of the rules in the rule set 12.Mean coverage of the rules in the rule set
13.Mean accuracy of the rules in the rule set 14.Amount of rules in the rule set.

5 Conclusions, Discussions and Future Works

5.1 Conclusions

It can be seen from Table 4 that in term of Weighted_avg_IR_precision RGA ranks first. In term of the Mean_absolute_error CBA ranks first, RGA second and C4.5 fifth. C4.5 worsen classification performances on lots of data sets slightly, but improve the classification performances on some data sets significantly. In terms of Percent_correct, weighted average F_measure, weighted average IR recall, weighted average true negative rate, and weighted average true positive rate C4.5 ranks first and RGA second. In term of Weighted_avg_area_under_ROC C4.5 ranks first, Jrip second and RGA third.

5.2 Discussions

(1) CBA and RGA have the lowest measure of mean absolute error. It can be found in Table 4 that the only consistent factor between CBA and RGA is that in term of rules' mean accuracy CBA ranks first and RGA the second. So we can guess that the measure of mean absolute error relates most to the metric of mean accuracy of rules in rule set.

(2)The rules in LEM2 have bigger mean coverage, but longer mean length and lower mean accuracy than RGA. The bigger mean coverage is dueto that LEM2 select the attribute values with biggest coverage to construct a rule. The longer mean length and lower mean accuracy is due to that in LEM2 the equivalence class of the condition attributes should be included in the equivalence class of the decision class, this does not be satisfied for inconsistent examples and as a result very long rules will be generated for inconsistent examples, and the final classification performance is impacted.

(3) The two differences of schema between RGA and SRGeS are that SRGeS has the attribute reduction step before rule generation and does not handle the inconsistent examples. So the metric of amount of attribute in rule set in SRGeS is very small (see Table 4) and may remove some significant attributes.

(4)RGA generates a rule for one example, whereas KNN treat every original example as a rule. Obviously, the length of a generated rule is shorter and more abstract than an original example in KNN. So RGA has higher performance than KNN.

References

1. Quinlan, R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo (1993)
2. Cohen, W.W.: Fast Effective Rule Induction. In: Twelfth International Conference on Machine Learning, pp. 115–123 (1995)
3. Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences 11, 341–356 (1982)
4. Liu, B., Hsu, W., Ma, Y.: Integrating Classification and Association Rule Mining. In: Fourth International Conference on Knowledge Discovery and Data Mining, pp. 80–86 (1998)
5. `http://www.lcb.uu.se/tools/rosetta/`
6. Thabtah, F.A., Cowling, P.I.: A greedy classification algorithm based on association rule. Applied Soft Computing 7, 1102–1111 (2007)
7. Yin, X., Han, J.: CPAR: classification based on predictive association rule. In: Proceedings of the SDM, San Francisco, CA, pp. 369–376 (2003)
8. Lim, T.-S., Loh, W.-Y.: A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-Three Old and New Classification Algorithms. Machine Learning 40, 203–228 (2000)
9. Thabtah, F., Cowling, P., Hammoud, S.: Improving rule sorting, predictive accuracy and training time in associative classification. Expert Systems with Applications 31, 414–426 (2006)
10. Li, R., Wang, Z.-O.: Mining classification rules using rough sets and neural networks. European Journal of Operational Research 157, 439–448 (2004)
11. Murphy, P.M., Aha, D.W.: UCI repository of machine learning databases, machine-readable data repository, Irvine, CA, University of California, Department of Information and Computer Science (1992)
12. Fayyad, U.M., Irani, K.B.: Multi-interval discretization of continuousvalued attributes for classification learning. In: Thirteenth International Joint Conference on Articial Intelligence, pp. 1022–1027 (1993)