UDC: 004 Computer science

Optimizing Performances in Knowledge Discovery Using Decision Trees Algorithms

Laviniu Aurelian Bădulescu
University of Craiova, Faculty of Automation, Computers and Electronics, Software Engineering Department, Craiova, Romania (laviniu_aurelian_badulescu@yahoo.com)

Summary

Decision trees classifiers are quite fast compared to other classification algorithms and is effortlessly represented and understand by people. In this paper, we study the behavior of the decision trees induced with 29 attribute selection measures over Census Income database taken from UCI Knowledge Discovery in Database Archive. Through this data mining experiment, we build decision trees in order to get some classification rules. The experiments presume the growing of the decision trees on a training data set, the pruning of the decision trees using two pruning methods: confidence level and pessimistic and finally, the decision trees execution on the test data set. The most important performance for the classification of the different decision trees, the classification accuracy on the test data, data completely unknown at the training of decision tree, has been noticed; this performance is expressed by classification error rate. We conclude the work with a discussion on related work. We can say that the value we encounter for the classification error rate equals with the best value obtained by the other algorithms from literature.

Keywords: *data mining, decision trees, classification error rate*

1. INTRODUCTION

In this paper, we study the behavior of the decision trees (DT) induced with 29 attribute selection measures over *Census Income database* taken from UCI Knowledge Discovery in Database Archive.

Through this mining experiment, we build decision trees in order to get some classification rules. We chose an attribute named "class <=50K, >50K" as class target attribute since we want to learn, based on census data, the proper classification of income (income<=$50K/yr or income>$50K/yr) for every person.

For the growing of the DT, 29 attribute selection measures have been tried. The experiments presume the growing of the DT on a training data set (in fact, there were induced 29 different DT using 29 attribute selection measures at the splitting of a DT node), the pruning of a DT (the 29 DT from the previous step are pruned, using two pruning methods: confidence level pruning and pessimistic pruning method) and finally, the DT execution on the test data set – different data of the ones used at the training of the DT - to calculate the classification error rate of each DT. Along with the performance of the file size for every DT induced with an attribute selection measure, we have also studied the behavior of the height and the number of nodes of every DT. The most important performance for the classification of the different DT, the classification accuracy on the test data, data completely unknown at the training of DT, has been noticed; this performance is expressed by classification error rate on the test data.

The rest of the paper is organized as follows. First we present the *Census Income database* features and the attribute selection measures used in this paper. Then we will discuss the behavior of the attribute selection measures on the growing, pruning and execution of the DT over the *Census Income database*. Finally, we conclude the work with a discussion on related work.

2. EXPERIMENTS

All experiments on *Census Income database* were conducted on a PC AMD Duron 995 MHz CPU with 512 MB RAM, running Windows XP and the code was written in C. For the performance tests we use software developed by C. Borgelt [3].

Census Income database is similar with *Adult Census database*, because both of them contain demographical and occupational variables. However, this database is much more larger (taking into account both the number of features and the number of cases) and much more detailed (*i.e.* the standard variables for census such as the industry code or the occupation are registered to a much more detailed level in this database) [2].

Census Income database is found in UCI Knowledge Discovery in Database Archive [13] and it contains data from some censuses from the Los Angeles and Long Beach regions during the years 1970, 1980, 1990 [16]. The owner is U.S. Census Bureau, United Stated Department of Commerce. The database was donated on March, 7th 2000 by Terran Lane and Ronny Kohavi [15].

Data characteristics. The data contain 40 attributes, continuous (7) and nominal (33) of occupational and demographical type, plus the class label (*income*). Total number of cases: 299.285. Number of training cases: 199.523, out of which duplicated or contradictory cases: 46.716. Number of testing cases: 99.762, out of which duplicated or contradictory cases: 20.936.

The prediction task is to determine the *income* for a person represented by a record in the dataset. The incomes were divided at the 50K$ level in order to state a problem of binary classification. The training file has over 100 MB, and the test file has over 50MB.

There has been used 29 attribute selection measures on which the splitting of a node of the DT has to be realized. Attribute selection measures used for induction, pruning and execution of DT are: information gain (*infgain*) [21] [9] [26], balanced information gain (*infgbal*), information gain ratio (*infgr*) [25] [26], symmetric information gain ratio 1 (*infsgr1*), symmetric information gain ratio 2 (*infsgr2*) [22], quadratic information gain (*qigain*), balanced quadratic information gain (*qigbal*), quadratic information gain ratio (*qigr*), symmetric quadratic information gain ratio 1 (*qisgr1*), symmetric quadratic information gain ratio 2 (*qisgr2*), Gini index (*gini*) [5] [28], symmetric Gini index (*ginisym*) [29], modified Gini index (*ginimod*), RELIEF measure (*relief*) [18] [17], sum of weighted differences (*wdiff*), χ^2 (*chi2*), normalized χ^2 (*chi2nrm*), weight of evidence (*wevid*) [19] [23], relevance (*relev*) [1] , Bayesian-Dirichlet/K2 metric (*bdm*), modified Bayesian-Dirichlet/K2 metric (*bdmod*) [10] [6] [12], reduction of description length - relative frequency (*rdlrel*), reduction of description length - absolute frequency (*rdlabs*), stochastic complexity (*stoco*) [20] [27], specificity gain (*spcgain*), balanced specificity gain (*spcgbal*), specificity gain ratio (*spcgr*), symmetric specificity gain ratio 1 (*spcsgr1*), symmetric specificity gain ratio 2 (*spcsgr2*)[4].

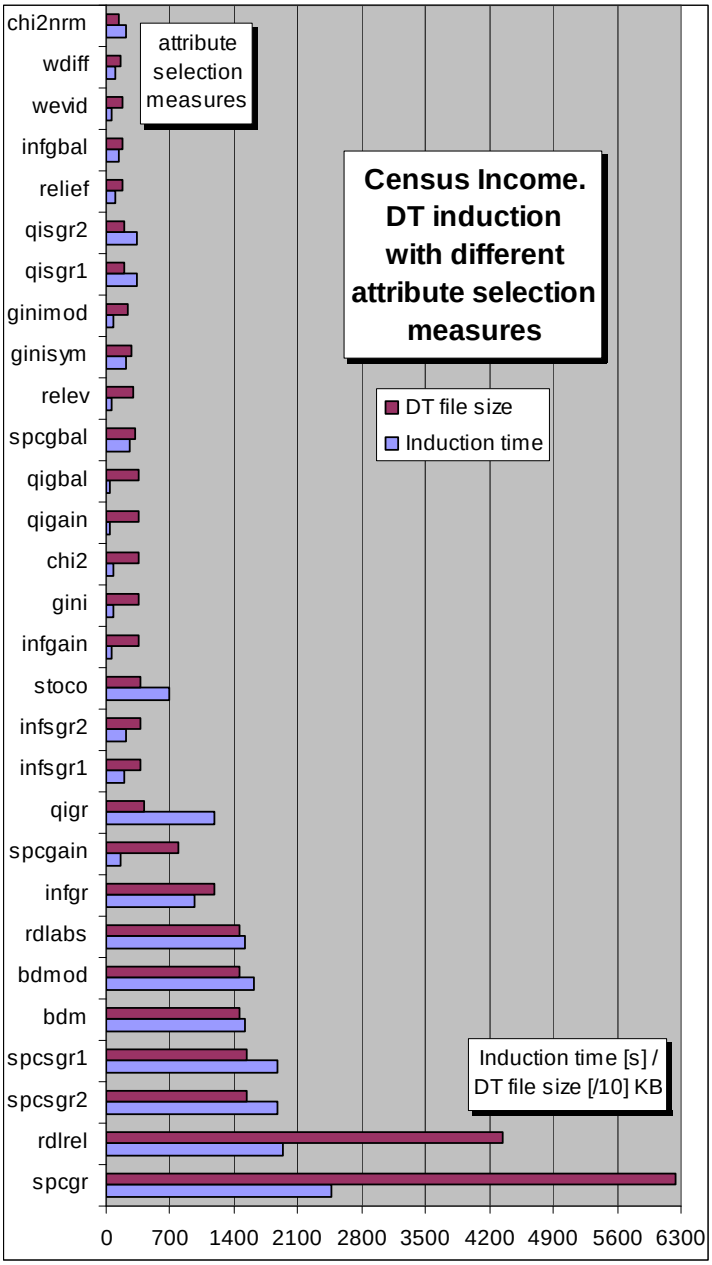
II.1. *The DT induction with 29 different measures*

Statistics	Attributes #	Nodes #	Levels #/DT height	Induction time [s]	DT file size [KB]
average:	38.55	21557.45	312.17	636.18	8866.52
maximum:	42 (spcgain)	60718 (spcgr)	1297 (spcgr)	2466.08 (spcgr)	62398 (spcgr)
minimum:	29 (stoco)	9555 (wevid)	13 (qigain, qigbal)	47.63 (qigbal)	1326 (chi2nrm)
standard deviation:	2.87	11700.67	441.67	750.99	13359.65

Table II.1. DT induction

From the values of Table II.1 and from the associated chart we conclude that there are three groups for the performances realized by the 29 measures. The first group realizes very good performances for the DT growth time and for the size of the file that contains the unpruned DT. In this group there are the following 16 measures: *infgain, gini, chi2, qigain, qigbal, spcgbal, relev, ginisym, ginimod, qisgr1, qisgr2, relief, infgbal, wevid, wdiff, chi2nrm*. Next group contains 11 measures with medium values for the 2 features presented in the chart: *spcsgr2, spcsgr1, bdm,*

bdmod, rdlabs, infgr, spcgain, qigr, infsgr1, infsgr2, stoco. The weakest performances are noticed to be presented in a last group of two measures: *rdlrel* and *spcgr*.



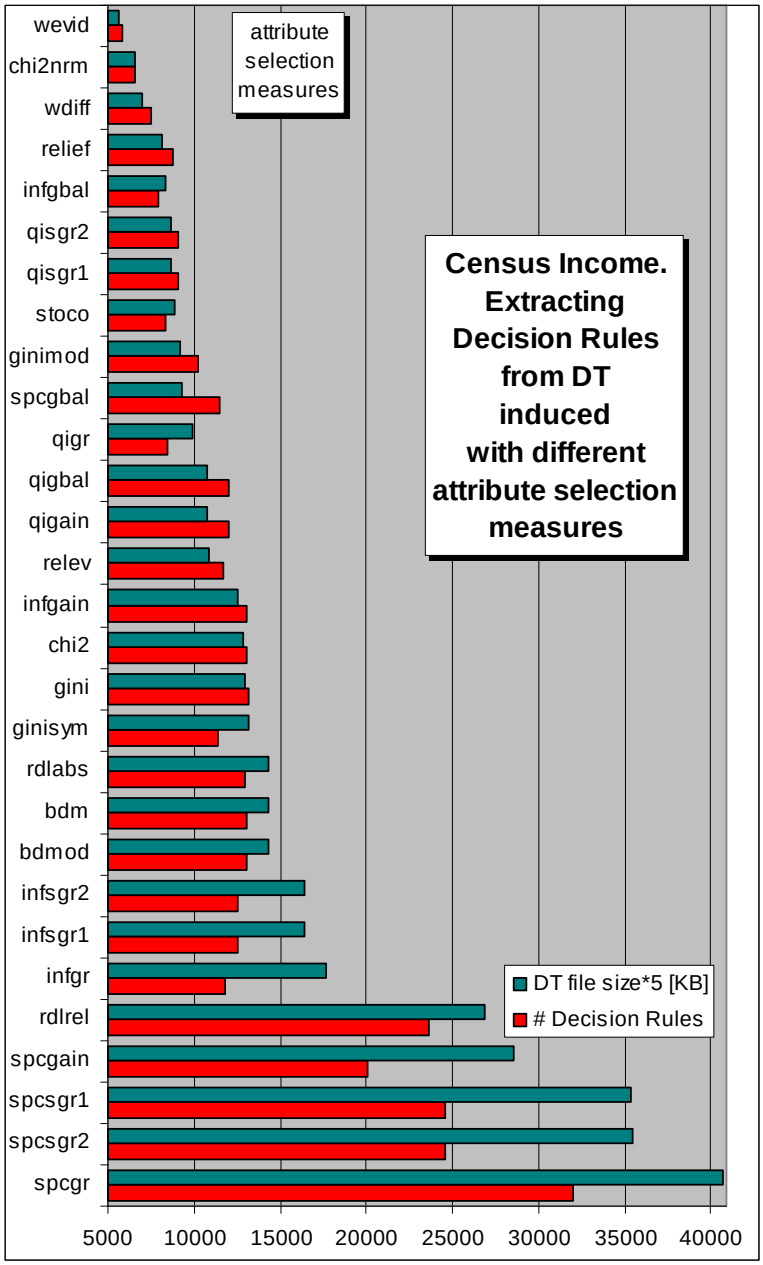
The correlation coefficient between the size of the file containing unpruned DT and the necessary time for the building of DT is 0.810 showing a powerful correlation between these features.

II.1.1. The decision rules extraction from the unpruned DT

Statistics	Decision rules #	Building rules file time [s]	Reading DT file time[s]	Decision rules file size [KB]
average:	13119.76	1.18	1.41	2995.38
maximum:	31976 (spcgr)	5.21 (spcgr)	8.42 (spcgr)	8137 (spcgr)
minimum:	5852 (wevid)	0.31 (wevid)	0.28 (chi2nrm)	1120 (wevid)
standard deviation:	6117.01	0.95	1.93	1860.32

Table II.2. Decision rules extraction from unpruned DT

As seen from Table II.2, which presents the performances obtained at the extraction of the decision rules from unpruned DT for each of the 29 measures, and from the associated chart which shows the performance for only two features: the decision rules number and the size of the file which contains these decision rules, the best values for the decision rules number are gained by a number of 24 measures*: infgr, infsgr1, infsgr2, bdmod, bdm, rdlabs, ginisym, gini, chi2, infgain, relev, qigain, qigbal, qigr, spcgbal, ginimod, stoco, qisgr1, qisgr2, infgbal, relief, wdiff, chi2nrm, wevid,* for which also the size of the file containing these decision rules is relatively small. A number of 5 measures (*spcgr, spcsgr2, spcsgr1, spcgain, rdlrel*) realizes weaker performances though.

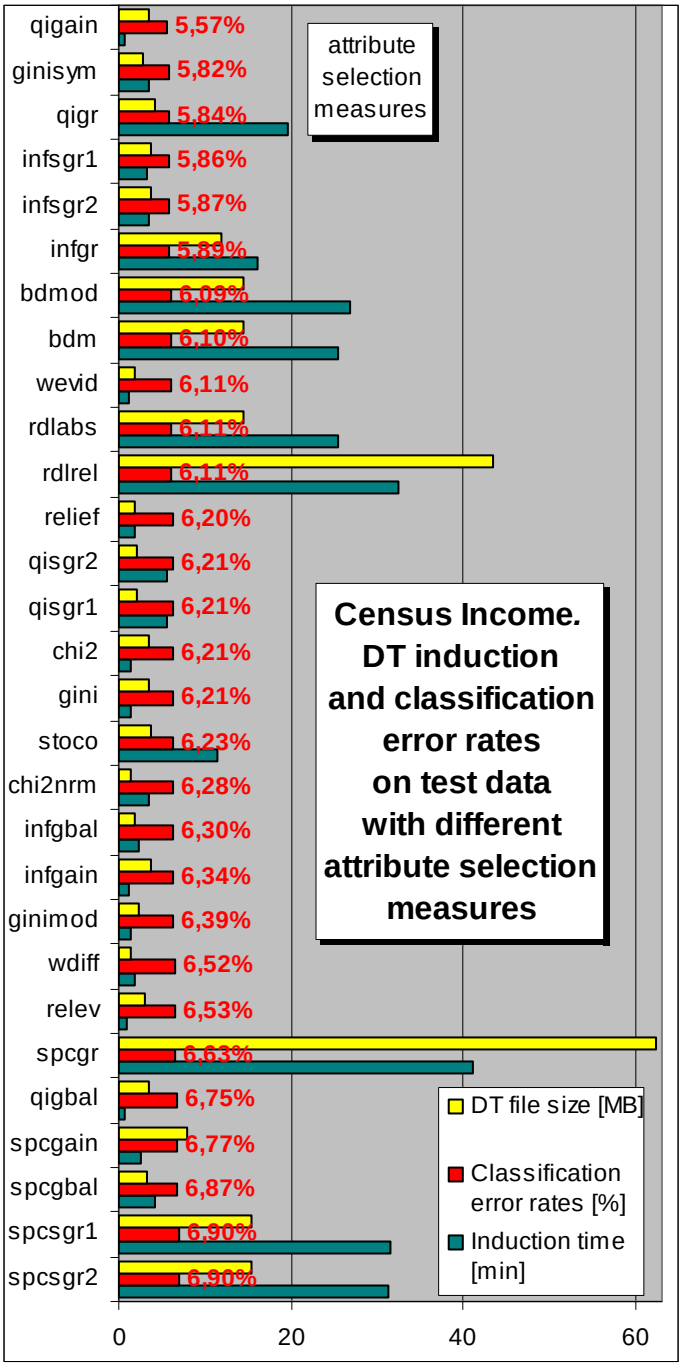


The correlation coefficient between the file size that contains the decision rules and the number of decision rules for unpruned DT is 0.971 showing a powerful correlation between these features.

II.1.2. The unpruned DT execution on test data

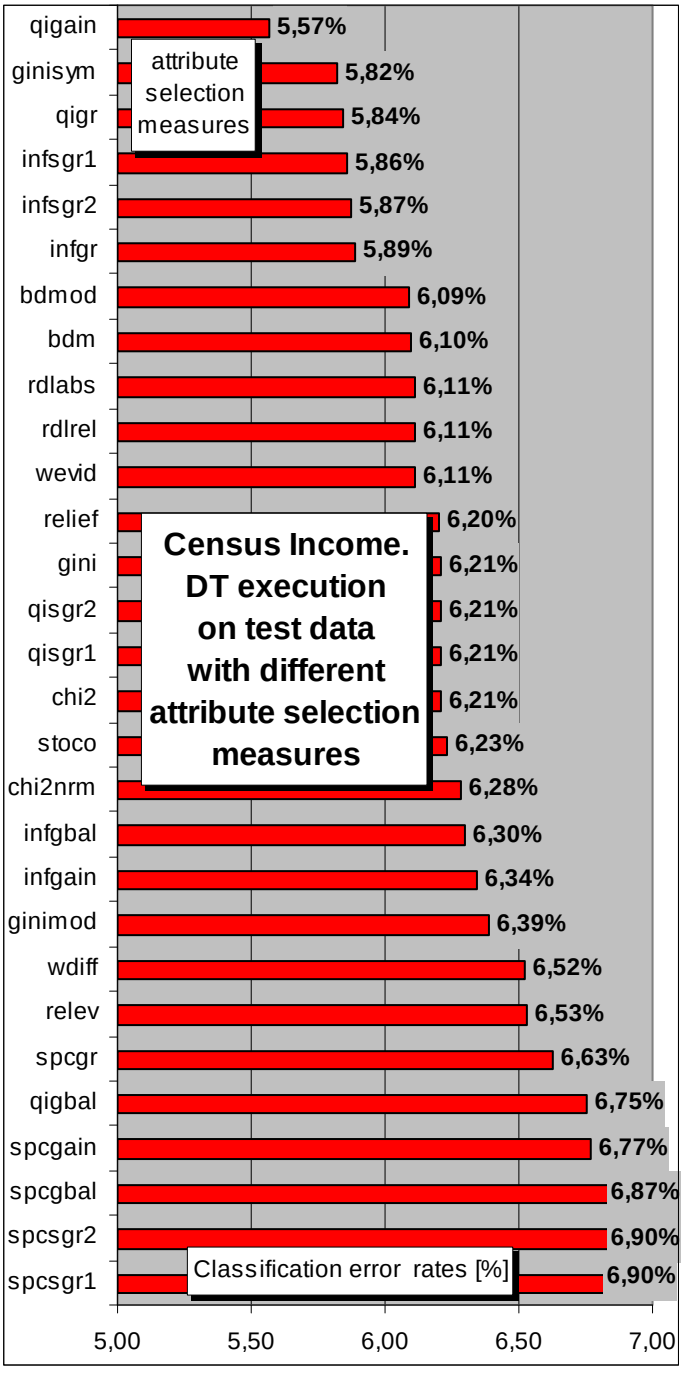
Statistics	Errors #	Error rate [%]
average:	6254.86	6.27
maximum:	6888 (spcsgr1)	6.90 (spcsgr1, spcsgr2)
minimum:	5560 (qigain)	5.57 (qigain)
standard deviation:	348.05	0.35

Table II.3. The unpruned DT execution on test data



For the unpruned DT, the best value for the classification error rate on the test data is obtained by *qigain* measure (5.57%), and the poorest performance is obtained by *spcsgr1* and *spcsgr2*

measures (6.90%). In fact, the amount of misclassified cases is not similar for the two measures, thus the *spcsgr1* measure classifies incorrectly 6888 cases, while the *spcsgr2* measure classifies incorrectly 6883 cases, but because of the approximations, the classification error rate appears similar for the two measures. Conclusively, the poorest performance for the classification rate error on the test data on *Census Income database*, for the unpruned DT is fulfilled by *spcsgr1* measure.

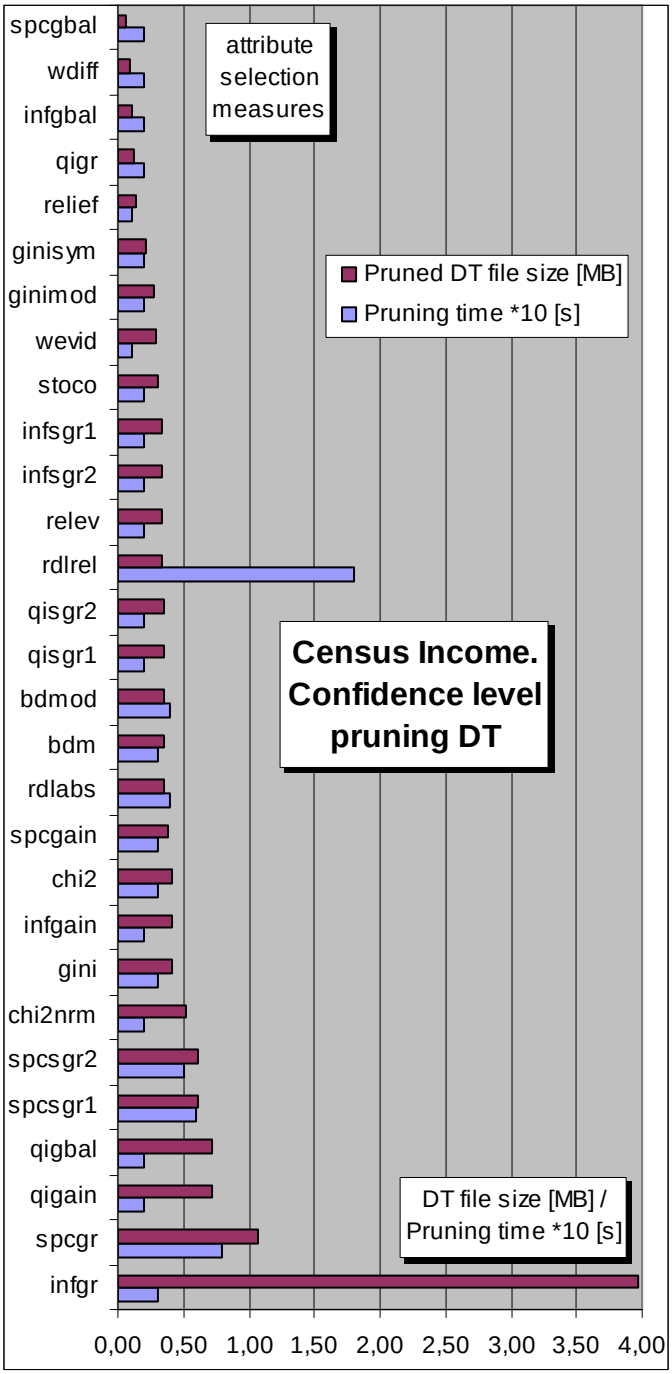


The correlation coefficient between the classification error rate and the DT induction time is 0.164 and the correlation coefficient between the classification error rate and the DT file size is 0.173. These poor values indicate a very small dependence between the classification error rate, on the one size, and the DT induction time and DT file size, on the other size.

II.2. *Confidence level pruning DT*

Statistics	Pruning parameter	Attribute #	Nodes #	Levels #	DT file pruning time [s]	Pruned DT file size [KB]
average:	0.82	30.07	1968.97	65.86	0.03	500
Maximum:	0.99 (infgr)	39 (rdlrel)	3641 (infgr)	861 (infgr)	0.18 (rdlrel)	3962 (infgr)
Minimum:	0.58 (qigain, qigbal)	13 (spcgbal)	470 (chi2nrm)	12 (qigain, qigbal)	0.01 (relief, wevid)	64 (spcgbal)
standard deviation:	0.10	5.64	810.12	161.38	0.03	699.68

Table II.4. Confidence level pruning DT



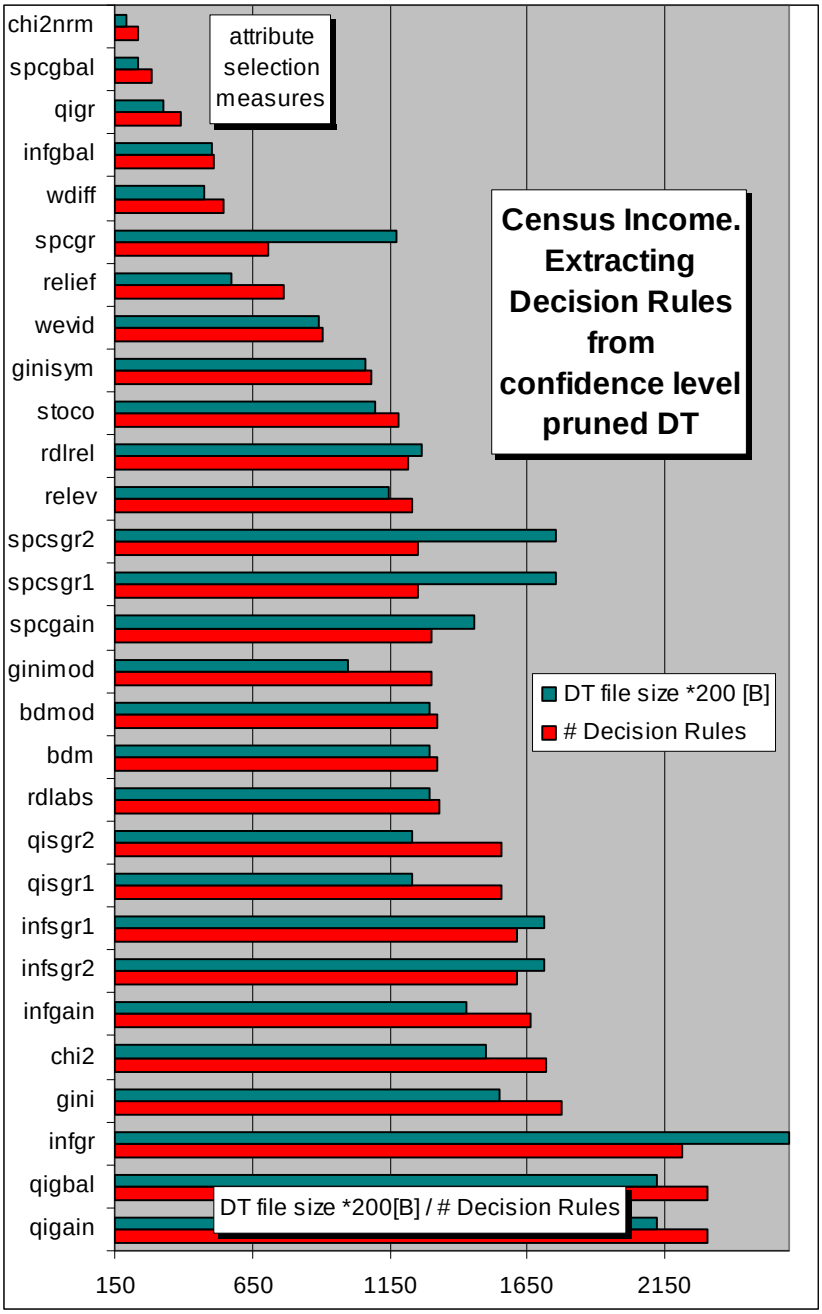
The correlation coefficient between the file size which contains pruned DT and the DT file pruning time is 0.074 indicating a poor correlation between these features.

Excepting two measures: *rdlrel* (which needs more time than the other measures for DT pruning: 0.18 seconds unlike the average of all 29 measures of 0.03 seconds and the minimum value of 0.01 seconds) and *infgr* (which builds a very large pruned DT file: 3962 KB, unlike the average of all 29 measures: 500 KB, or the minimum value: 64 KB), all the other 27 measures have approximately equal values for the DT file pruning time and for the pruned DT file size.

II.2.1. The decision rules extraction from the confidence level pruned DT

Statistics	Decision rules #	Building rules file time [s]	Reading DT file time[s]	Decision rules file size [KB]
average:	1257	0.07	0.06	248.10
maximum:	2301 (qigbal)	0.21 (infgr)	0.43 (infgr)	520 (infgr)
minimum:	236 (chi2nrm)	0.02 (chi2nrm)	0.01 (chi2nrm, spcgbal)	38 (chi2nrm)
Standard deviation:	557.14	0.03	0.07	115.85

Table II.5. The decision rules extraction from the confidence level pruned DT



The differences between the numbers of decision rules, which the 29 measures need for the construction of the classifier is very big. The maximum value of the decision rules number is almost 10 times bigger than the minimum value. The maximum number of decision rules is 2301, the minimum number is 236 and the standard deviation is 557.14.

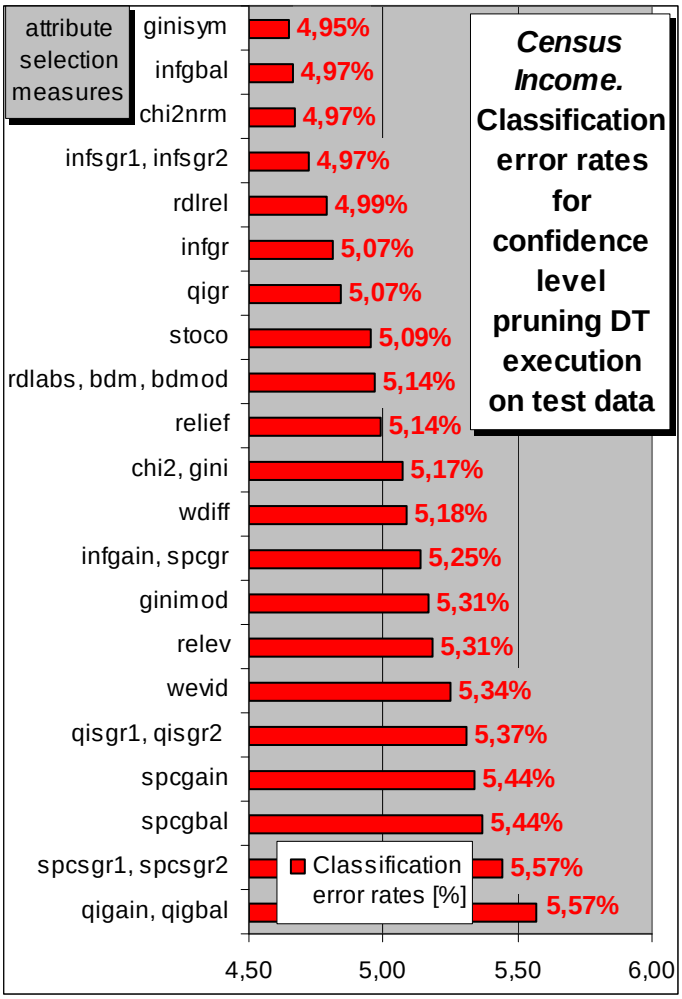
Unlike the unpruned DT, which presents a maximum number of decision rules, namely 31976 (14 times bigger), a minimum number of 5852 (25 times bigger) and a standard deviation of 6117.01 (11 times bigger), we can say that the performance "decision rules number" has obvious improved. Also, as we will next see, it has improved through the confidence level pruning the classifier accuracy.

The correlation coefficient between the confidence level pruned DT file size and the decision rules number is 0.917, thus indicating a strong correlation between these features.

II.2.2. The confidence level pruned DT execution on test data

Statistics	Errors #	Error rate [%]
average:	5063.72	5.07
maximum:	5560 (qigain, qigbal)	5.57 (qigain, qigbal)
minimum:	4634 (ginisym)	4.65 (ginisym)
standard deviation:	276.01	0.27

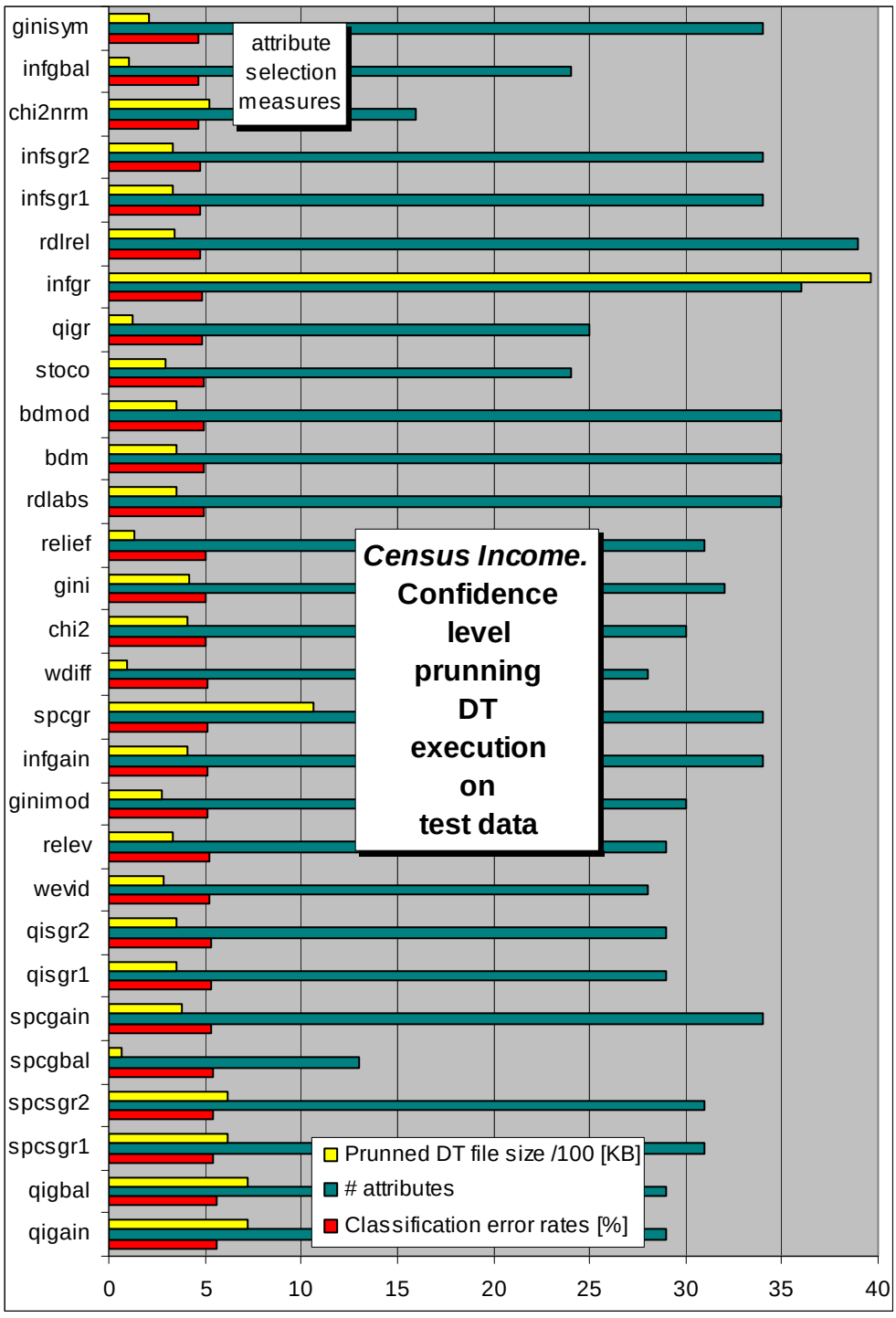
Table II.6. The confidence level pruned DT execution on test data



For the confidence level pruning DT, one can obtain the best value for the classification error rate: 4.65% (*ginisym* measure). The worst value (maximum) for the classification error rate (5.57%, obtained by two measures: *qigain* and *qigbal*) was the best value of unpruned DT.

At unpruned DT the highest value of the classification error rate was 6.90%. We also perceive a reduction for the classification error rate standard deviation with the confidence level pruning of DT, from 0.35 to 0.27.

The correlation coefficient between the classification error rate and the pruning parameter is higher, -0.749, indicating the existence of quite powerful dependence between these features. This matter can be perceived as: high value of the pruning parameter implies a small value of the classification error rate (*i.e.* a high value of the classification accuracy). So, we can conclude that for high values of the pruning parameter we can obtain a better accuracy of DT classifier.



The correlation coefficient between the classification error rate and the confidence level pruned DT file size is very small, -0.048, about the same as the correlation coefficient between the classification error rate and the number of attributes necessary for pruned DT, -0.14; which suggests the fact that there is no evident dependence between the confidence level pruned DT file size and the classification error rate, as the number of attributes used at pruned DT building doesn't have an influence on the classification error rate.

II.3. *Pessimistic pruning DT*

Statistics	Attribute #	Nodes #	Levels #	DT file pruning time [s]	Pruned DT file size [KB]
average:	35.28	7795.76	59.45	0.02	1659.48
maximum:	41 (rdlrel)	9691 (stoco)	423 (infgr)	0.07 (spcgr)	7064 (spcgr)
minimum:	24 (spcgbal)	5894 (wevid)	13 (qigain, qigbal)	0.01 (chi2nrm, gini, ginisym, infgain, infgbal, relev, relief, wdiff, wevid, qigbal, qisgr2)	829 (relief)
standard deviation:	3.95	882.41	90.71	0.02	1174.20

Table II.7. Pessimistic pruning DT

Through the pessimistic pruning DT it has been established a decrease of the DT file size. Thus, the maximum value of the file size decreases from 62398 KB to 7064 KB (9 times), the minimum value decreases a bit, from 1326 KB to 829 KB (1.5 times) and the average value from 8866.52 KB to 1659.48 KB (5 times). The standard deviation decreases much more for the DT file size, from 13359.65 to 1174.20 (11 times).

The tree's height has at unpruned DT a maximum value of 1297 levels which through the pessimistic pruning of DT decreases to 423 (3 times). The minimum value for unpruned DT is of 13 levels; we can notice that it is not modified through the pessimistic pruning of DT. However, the average decreases from 312.17 levels to 59.45 levels (5 times). For this performance the standard deviation decreases from 441.67 to 90.71 (5 times).

The number of tree's nodes has at unpruned DT a maximum value of 60718 nodes which through the pessimistic pruning of DT decreases to 9691 (6 times).

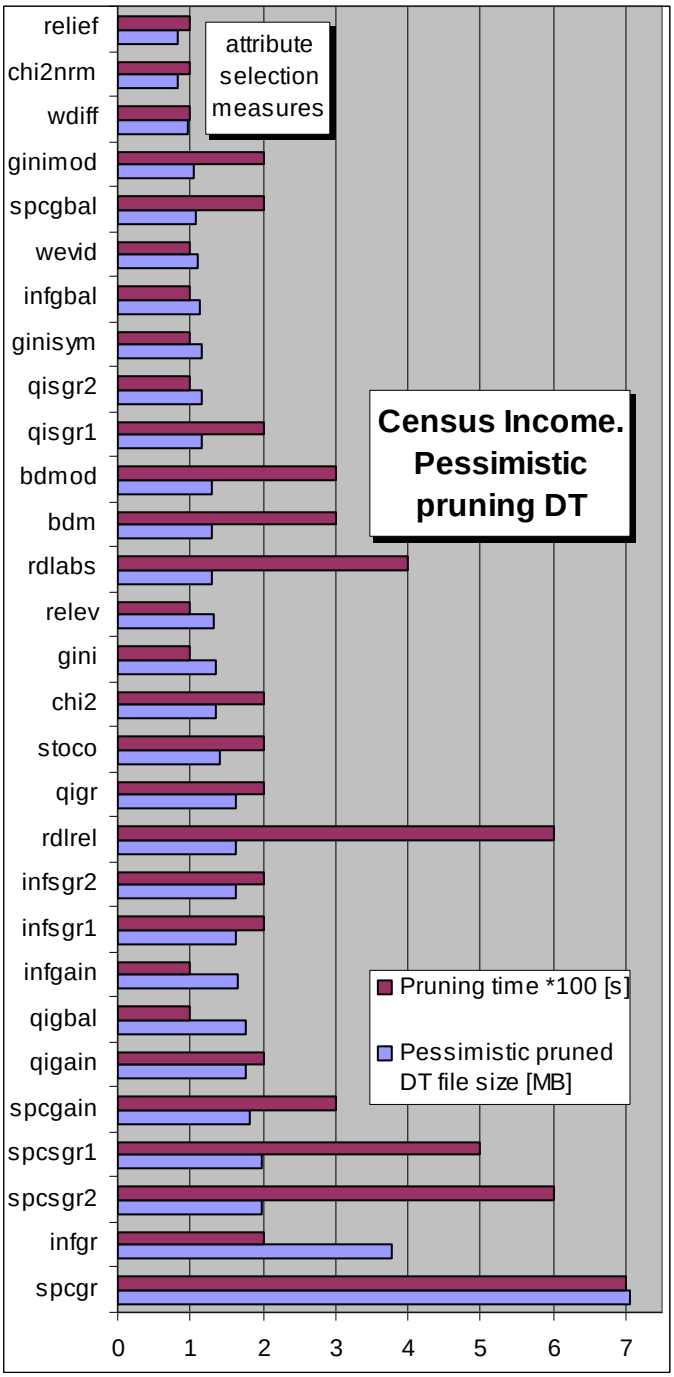
The minimum value of unpruned DT is of 9555 nodes; we can notice that it decreased through the pessimistic pruning of DT to a value of 5894 nodes (almost 2 times). However, the average decreases from 21557.45 nodes to 7795.76 nodes (almost 3 times). For this performance, the standard deviation decreases from 11700.67 to 882.41 (over 13 times).

The tree's height presents for confidence level pruned DT a maximum value of 861 levels, for which the pessimistic pruned DT is of 423 levels (two times smaller). The minimum value for confidence level pruned DT is of 12 levels; at the pessimistic pruned DT this value is slightly bigger: 13 levels. The average decreases from 65.86 levels (confidence level pruning) to 59.45 levels (pessimistic pruning). For this performance the standard deviation decreases from 161.38 (confidence level pruning) to 90.71 (pessimistic pruning).

The nodes number of the tree presents for confidence level pruned DT a maximum value of 3641 nodes which at the pessimistic pruning DT is of 9691 (almost 3 times higher). The minimum value for confidence level pruned DT is of 470 nodes, but for pessimistic pruning the minimum value of the number of nodes is of 5894 nodes (almost 13 times higher). The average also increases from 1969.97 nodes to 7795.76 nodes (almost 4 times). For this performance the standard deviation slightly increases from 810.12 to 882.41.

Conclusively, we may say that for the *Census Income database*, the pessimistic pruning method builds less deeper DT, but with more nodes than the confidence level pruning method.

For the pruning of the DT with the pessimistic pruning method, correlation coefficient between the pruning time and the size of the file that contains pruned DT is of 0.590 showing that



there is a clear dependence between these two features. This fact is explainable by idea that a DT of a large size implies a long time for the building of the file which contains it.

II.3.1. The decision rules extraction from the pessimistic pruned DT

Statistics	Decision rules #	Building rules file time [s]	Reading DT file time[s]	Decision rules file size [KB]
average:	4662.93	0.45	0.36	1012.21
maximum:	5644 (qigbal)	1.16 (spcgr)	1.13 (spcgr)	1506 (infgr)
minimum:	3433 (qigr)	0.23 (qigbal)	0.16 (chi2nrm)	684 (wevid)
standard deviation:	557.85	0.21	0.21	210.52

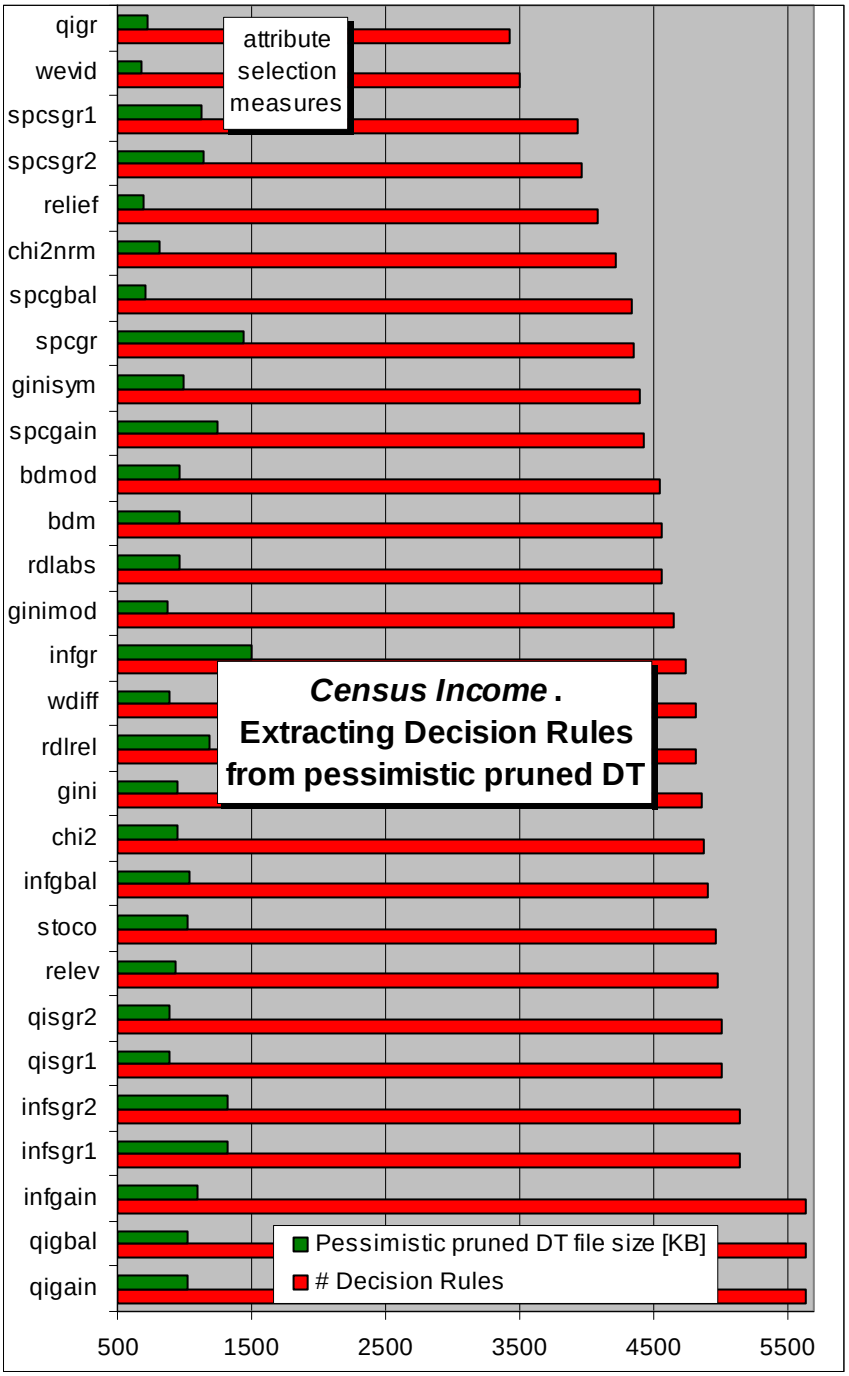
Table II.8. The decision rules extraction from the pessimistic pruned DT

For the decision rules extraction from the pessimistic pruned DT, the correlation coefficient between the size of the file containing pruned DT and the decision rules number is pretty small: 0.325, indicating a poor positive correlation between these two features.

The differences between the decision rules number which require the 29 measures needed to construct the classifier are relatively small. The maximum value of decision rules number (5644) is about 1.5 times larger than the minimal value (3433). The standard deviation is 557. 85.

Towards to the unpruned DT, presenting a maximal decision rules number of 31976 (almost 6 times larger), a minimum number of 5852 (almost 2 times more) and a standard deviation of 6117.01 (11 times more), we can say that the performance of the decision rules number has increased significantly with the pessimistic pruning of DT. As we are about to see, the accuracy of the classifier it has also improved through the pessimistic pruning.

Towards to the confidence pruning DT, which presented a maximum decision rules number of 2301 (over 2 times less), a minimum number of 236 (over 14 times less) and a standard deviation of 557.14 (almost the same), we can say that the performance of the decision rules number is better for

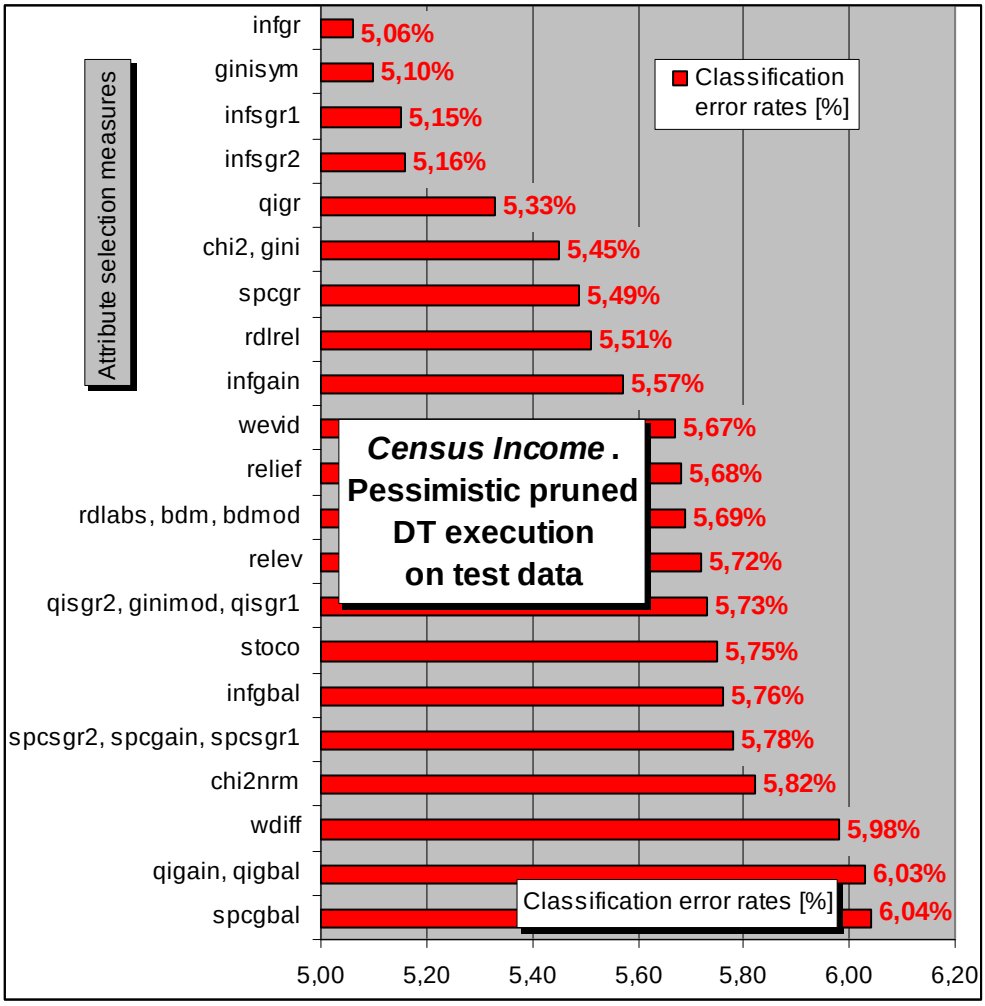


the confidence level pruned DT towards the pessimistic pruned DT. We will notice that we can say the same thing about the most important performance of the classifier: the classification accuracy on the test data.

II.3.2. The pessimistic pruned DT execution on test data

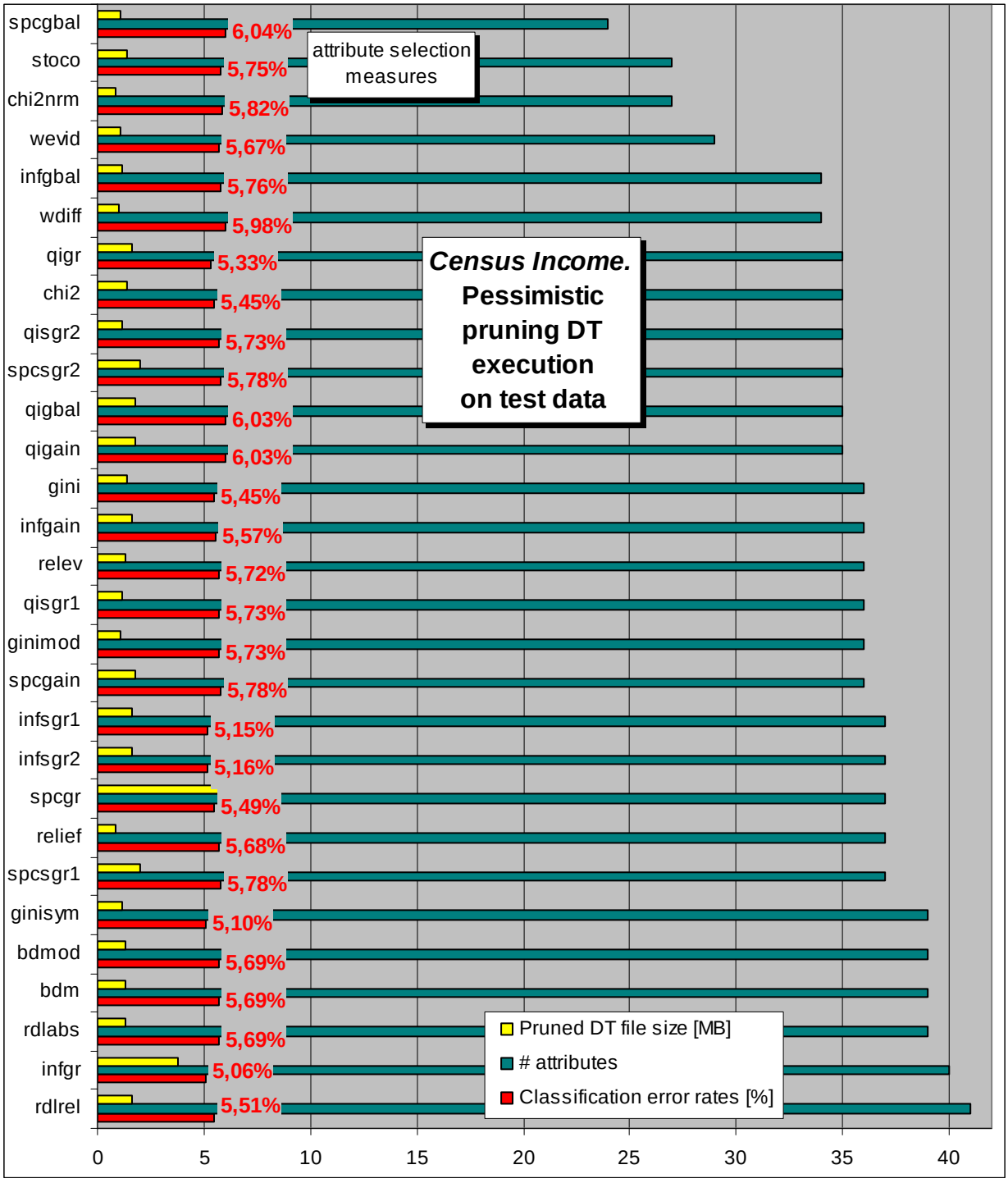
Statistics	Errors #	Error rate [%]
average:	5619.62	5.63
maximum:	6029 (spcgbal)	6.04 (spcgbal)
minimum:	5051 (infgr)	5.06 (infgr)
Standard deviation:	270.13	0.27

Table II.9. The pessimistic pruned DT execution on test data



At the DT execution on test data, data completely unknown at the DT training, the correlation coefficient between the classification error rate and the number of necessary attributes for the induction of pessimistic pruned DT is -0.478, showing a relatively opposite dependence between the classification error rate and the number of attributes, the more attributes are needed, the smaller the classification error rate is, a small number of necessary attributes for the pruned DT induction implies a raised classification error rate.

Instead, the correlation coefficient between classification error rate and the pruned DT file size, is smaller, but maintaining the negative sign: -0.269, showing a very weak opposite dependence between pessimistic pruned DT classification error rate and the pessimistic pruned DT file size.



It is proved that the best performance at the pessimistic pruned DT execution on test data is realized by the *infgr* measure (5.06%). But even the less better performance, realized by the *spcgbal* measure (6.04%), is just a little bit higher.

II.4. *Summarized table with decision rules number and classification error rate for the three types of DT*

Statistics	Unpruned DT		Confidence level pruned DT		Pessimistic pruned DT	
	Rules #	Classification error rate [%]	Rules #	Classification error rate [%]	Rules #	Classification error rate [%]
Average:	**6254.86**	**6.27**	**5063.72**	**5.07**	**5619.62**	**5.63**
maximum:	**6888** (spcsgr1)	**6.90** (spcsgr1)	5560 (qigain, qigbal)	5.57 (qigain, qigbal)	6029 (spcgbal)	6.04 (spcgbal)
minimum:	5560 (qigain)	5.57(qigain)	**4634 (ginisym)**	**4.65 (ginisym)**	5051 (infgr)	5.06(infgr)
standard deviation:	348.05	0.35	276.01	0.27	270.13	0.27

Table II.10. Summarized table with decision rules number and classification error rate for the three types of DT

The best performance for the classification error rate on the *Census Income database*, is obtained by confidence level pruned DT through *ginisym* measure (4.65%). This measure, also for the same confidence level pruned DT, obtains even the smallest decision rules number (4634). The weakest performance for the classification error rate on the *Census Income database*, is obtained by unpruned DT through *spcsgr1* measure (6.90%). This measure, also for the same unpruned DT, obtains even the highest decision rules number (6888).

Let's notice that the *qigain* measure which, for the unpruned DT obtains the best performance for the classification error rate (5.57%) and the smallest decision rules number (5560), when pruning the DT with the confidence level pruning method does not improve at all the performance, but it preserves the previous one (which, comparatively with the performances obtained by the other measures means the weakest performance) and, for the pessimistic pruned DT, it deteriorates the performance.

By calculating the correlation coefficients between the three classification error rate groups corresponding to the three types of DT (unpruned, confidence level pruned and pessimistic pruned) for all the 29 measures, it results that all the three values exceed 0.500, which means a clear dependence between these values. Thus the correlation coefficient between classification error rate for unpruned DT and confidence level pruned DT is 0.539, correlation coefficient between unpruned DT and pessimistic pruned DT is 0.547, scarcely higher, indicating a tighter dependence between this two features than previous dependence, and the correlation coefficient between confidence level pruned DT and pessimistic pruned DT is 0.664, indicating a higher correlation between these two features.

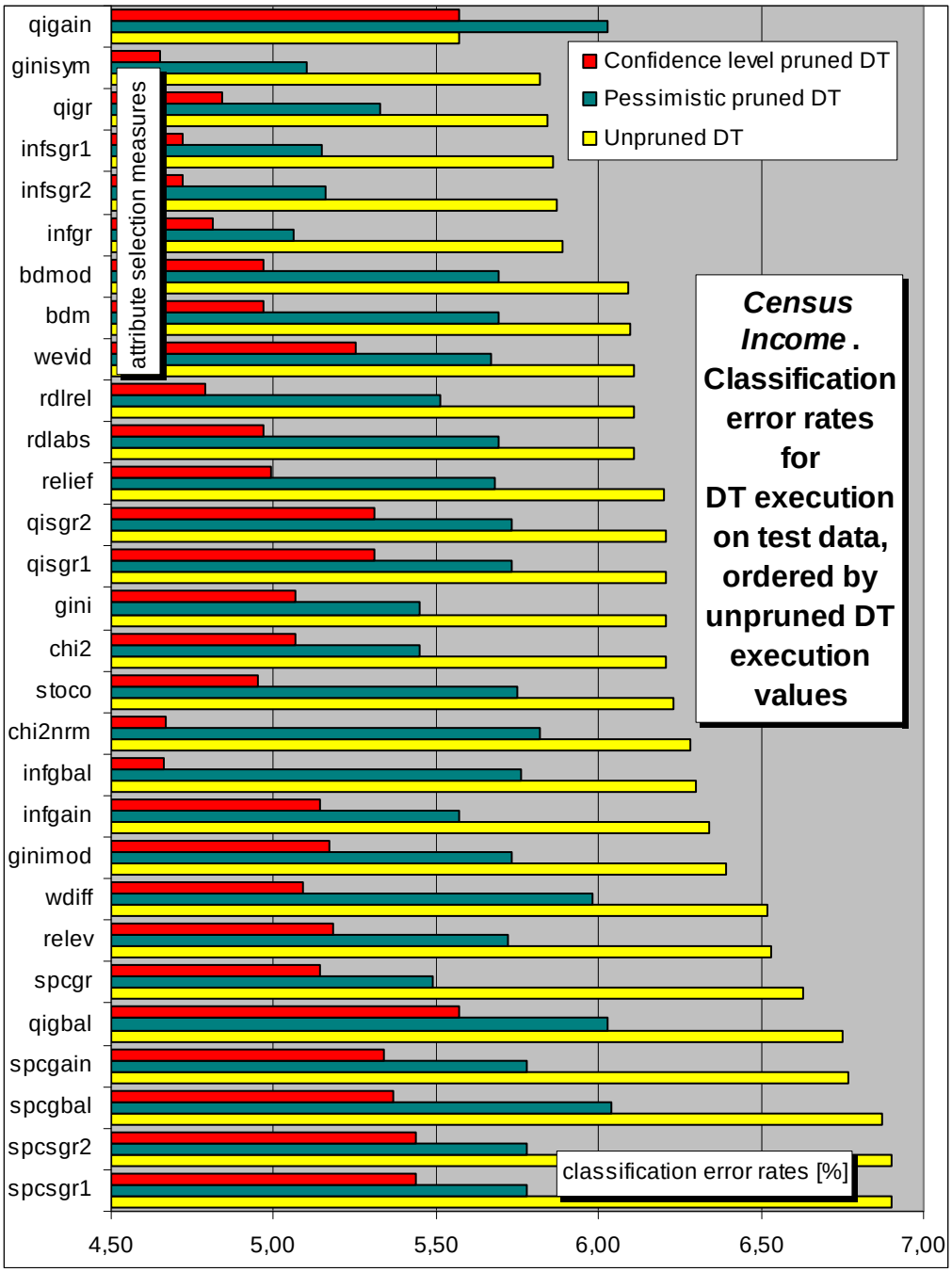
The value of the correlation coefficient between the classification error rates and the decision rules number for each attribute selection measure of the 29 measures and every DT (unpruned, confidence level pruned and pessimistic pruned) is 1.000.

The very good correlation to absolutely all measures shows that the small values for the decision rules involve small values for the classification error rate, as it can be noticed on the confidence level pruned DT (the average of decision rules number is 5063.72, and the average of classification error rate is 5.07%); average values for the decision rules number involve average values for the classification error rate, as it is noticed on the pessimistic pruned DT (the average of decision rules number is 5619.62, and the average of the classification error rate is 5.63%); and big values for the decision rules number are related to big values for the classification error rate as we notice on the unpruned DT (the average of the decision rules number is 6254.86, and the average of classification error rate is 6.27%).

The smallest values for the classification error rate are systematically obtained by confidence level pruned DT, the average of the classification error rate for the 29 attribute selection measures is 5.07%.

On the second place, with average values for the classification error rate, stands pessimistic pruned DT; the average of the classification error rate for all 29 attribute selection measures is 5.63%.

The biggest values for the classification error rate on *Census Income database* are made by unpruned DT; the average error classification rate for all 29 attribute selection measures is 6.27%.



The standard deviation, representing the spreading of the values of the classification error rate, is the same (0.27) for both the confidence level pruning method and pessimistic pruning method. Whereas, standard deviation for the unpruned DT classification error rate is higher (0.35). Taking into account the values obtained for the classification error rate at the three types of DT, we can say that throughout the pruning of DT, the accuracy of the classification improves and the spreading of the values of the classification error rate diminishes (the standard deviation decreases).

3. CONCLUSIONS AND RELATED WORK

The experiments accomplished targeted the growing, the pruning and the execution of the unpruned and the pruned DT on the test data. We tried to study the behavior of DT grown with 29 different attribute selection measures and in the same time the classification accuracy on the test

data of these trees. In our experiment we use *Census Income database* from UCI Knowledge Discovery in Database Archive.

1. In the documentation of the *Census Income database* (http://kdd.ics.uci.edu/databases/census-income/census-income.names) the following values for the classification error rate obtained by the various algorithms are presented:

#	Algorithm	Classification error rate
1.	C4.5	4.8%
2.	C5.0	4.7%
3.	C5.0 rules	4.7%
4.	**C5.0 boosting**	**4.6%**
5.	Naïve-Bayes	23.2%

Table III.1. Classification error rate values from literature

We can say that the value we encounter (4.65%) equals with the best value obtained by the other algorithms, taking into account that in the documentation is being offered for the classification error rate value only the first digit after the dot.

2. In [8] it is shown that the running of the Weka implementation for the Naïve Bayes algorithm on this database finalizes with the error: "out of memory", whereas the implementation of AirlDM with INDUS provides an accuracy of 76.2174% (*i.e.* error rate is 23.7826%).

3. In [14] the original values of the training records number (199523) and of the test records number (99762), have been modified to 249285 and, respectively to 50000. It is obvious that augmentation of the training records number, followed by a decreasing of the test records number, can only lead to the improvement of the classifier's accuracy. Thus are achieved the following classification error rates: 4.69% (for the C4.5 algorithm), 4.72% (for the BC4.5 algorithm), 5.44% (for the C4.5C algorithm), "out of memory" (for the BC4.5C algorithm), 4.46% (for the ADTree algorithm), 4.62% (for the OTC algorithm). The best value, gained through the process previously explained, is 4.46% , slightly smaller than our best value: 4.65%, achieved on the original values of the training and test records number.

4. In [11] the original values of the training records number (199523) and of the test records number (99762) have been modified to 224285 and, respectively to 50000; 25000 records have been used for validation. It is obvious that an increase in the training records number followed by a decrease of test records number can only lead to the improvement of the classifier's accuracy (*i.e.* a lower classification error rate). Thus, on the modified values for the distribution of records number, is obtained a classification error rate of 4.43% (for the LogitBoost algorithm), slightly smaller value than our best value: 4.65%, achieved on the original values of the training and test records number.

5. In [24] the original values for the number of training cases vs. the number of testing cases are being kept, whereas the value obtained for the classification accuracy on the test data is being presented only graphically. Examining the chart we can undoubtedly say that this value is under 95%, it is around 94%. So this means a classification error rate upper to 5%, maybe even 6%. The certain performance is lower then the best value for the classification error rate obtained by us, namely 4.65%.

6. The experiments made with XMiner over *Census Income database* in [7] don't provide any value for the classification accuracy.

7. In [2] the classification accuracy isn't mentioned as well, whereas it is affirmed that the running under the Apriori algorithm has failed providing the error „out of memory".

To conclude, the best performance obtained by the measures used in our experiments on *Census Income database*, is of 4.65% (classification error rate on the test data) and 4634 (decision rules number), achieved by the *ginisym* measure.

ACKNOWLEDGMENTS

We want to note the assistance we received from Hettich, S. and Bay, S. D., Irvine, CA: University of California, Department of Information and Computer Science.

4. REFERENCES

[1] P., W., Baim, *A method for attribute selection in inductive learning systems*, in "IEEE Trans. on PAMI", 1988, vol.10, pp. 888-896.

[2] S., D., Bay, *Multivariate Discretization for Set Mining*, in "Knowledge and Information Systems", New York, 2001, vol. 3 , Issue 4, pp. 491 – 512.

[3] C., Borgelt, http://fuzzy.cs.uni-magdeburg.de/~borgelt/dtree.html.

[4] C., Borgelt, R., Kruse, *Evaluation Measures for Learning Probabilistic and Possibilistic Networks*, in "Proc. of the FUZZ-IEEE'97", Barcelona, Spain, 1997, vol. 2, pp. 669–676.

[5] L., Breiman, J., Friedman, R., Olshen, C., Stone, *Classification and Regression Trees*, Stanford University and the University of California, Berkeley, 1984.

[6] W., Buntine, *Theory Refinement on Bayesian Networks*, in "Proc. 7th Conf. on Uncertainty in Artificial Intelligence", Morgan Kaufman, Los Angeles, CA, 1991, pp. 52–60.

[7] T., Calders, B., Goethals, M., Mampaey, *Mining Itemsets in the Presence of Missing Values*, in Z. Cho, R., L., Wainwright, H., Haddad, S., Y., Shin, Y., W., Koo, (eds.), Proceedings 22nd Annual ACM Symposium on Applied Computing (SAC '07, Seoul, Korea, March 11-15, 2007), ACM Press, 2007.

[8] D., Caragea, *Learning classifiers from distributed, semantically heterogeneous, autonomous data sources*, Iowa State University, Ames, Iowa, 2004, p. 162 , pp. 162-163.

[9] C., K., Chow, C., N., Liu, *Approximating Discrete Probability Distributions with Dependence Trees*, in "IEEE Trans. on Information Theory", IEEE, 1968,14(3), pp. 462–467.

[10] G., F., Cooper, E., Herskovits, *A Bayesian Method for the Induction of Probabilistic Networks from Data*, in "Machine Learning", Kluwer Academic Publishers, 1992, vol. 9, pp. 309–347.

[11] E., Frank, G., Holmes, R., Kirkby, M., Hall, *Racing Committees for Large Datasets*, Lecture Notes In Computer Science, Proceedings of the 5th International Conference on Discovery Science, Springer-Verlag, London, 2002, vol. 2534, pp.153–164,

[12] D., Heckerman, D., Geiger, D., M., Chickering, *Learning Bayesian Networks: The Combination of Knowledge and Statistical Data*, in "Machine Learning", Kluwer Academic Publishers, 1995, vol. 20, pp. 197–243.

[13] S., Hettich, S., D., Bay, (1999). The UCI KDD Archive [http://kdd.ics.uci.edu]. Irvine, CA: University of California, Department of Information and Computer Science.

[14] G., Holmes, R., Kirkby, B., Pfahringer, *Mining Data Streams Using Option Trees*, Workshop on Knowledge Discovery in Data Streams, 15th European Conference on Machine Learning (ECML), Pisa, 2004.

[15] http://kdd.ics.uci.edu/databases/census-income/census-income.data.html.

[16] http://kdd.ics.uci.edu/databases/census-income/census-income.html.

[17] K., Kira, L., Rendell, *A practical approach to feature selection*, in "Proc. Intern. Conf. on Machine Learning", D. Sleeman, P. Edwards (eds.), Morgan Kaufmann, Aberdeen, July 1992, pp. 249-256.

[18] I., Kokonenko, *Estimating Atributes: Analysis and extensions of RELIEF*, in "Proc. European Conf. on Machine Learning", L. De Raedt, F. Bergadano (eds.), Springer Verlag, Catania, April 1994, pp. 171-182.

[19] I., Kokonenko, *On Biases in Estimating Multi-Valued Attributes*, in "Proc. 1st Int. Conf. on Knowledge Discovery and Data Mining", Montreal, 1995, pp. 1034–1040.

[20] R., E., Krichevsky, V., K., Trofimov, *The Performance of Universal Coding*, in "IEEE Trans. on Information Theory", 1983, 27(2), pp. 199–207.

[21] S., Kullback, R., A., Leibler, *On Information and Sufficiency*, in "Ann. Math. Statistics", 1951, vol. 22, pp. 79–86.

[22] R., L., de Mantaras, *A Distance-based Attribute Selection Measure for Decision Tree Induction*, in "Machine Learning", Kluwer Academic Publishers, Boston, 1991, vol. 6, pp. 81–92.
[23] D., Michie, *Personal Models of Rationality*, in "Journal of Statistical Planning and Inference", Special Issue on Foundations and Philosophy of Probability and Statistics, 1990, vol. 21, pp. 381-399.
[24] N., C., Oza, S., Russell, *Experimental Comparisons of Online and Batch Versions of Bagging and Boosting*, in *The 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Morgan Kaufmann, San Francisco, California, 2001, pp. 359–364.
[25] J., R., Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufman, 1993.
[26] J., R., Quinlan, *Induction of Decision Trees*, in "Machine Learning", 1986, vol. 1, pp. 81–106.
[27] J., Rissanen, *Stochastic Complexity*, in "Journal of the Royal Statistical Society" (Series B), 1987, vol. 49, pp. 223-239.
[28] L., Wehenkel, *On Uncertainty Measures Used for Decision Tree Induction*, in "Proc. of the International Congress on Information Processing and Management of Uncertainty in Knowledge based Systems", IPMU96, 1996, pp. 413-418.
[29] X., Zhou, T., S., Dillon, *A statistical-heuristic Feature Selection Criterion for Decision Tree Induction*, in "IEEE Trans. on Pattern Analysis and Machine Intelligence", PAMI-13, 1991, pp. 834–841.

amount of the figures: 0
amount of the diagrams: 13
amount of the tables: 11

Paper received 2008-07-18