

Verifying cuts as a tool for improving a classifier based on a decision tree

Łukasz Dydo, Jan G. Bazan, Sylwia Buregwa-Czuma,
 Wojciech Rząsa
 Interdisciplinary Centre for Computational Modelling,
 University of Rzeszow, Pigońia 1, 35-310 Rzeszow, Poland
 Email: {ldydo, bazan, sczuma, wrzasa}@ur.edu.pl

Andrzej Skowron
 Institute of Mathematics, The University of Warsaw
 Banacha 2, 02-097 Warsaw, Poland and
 Systems Research Institute Polish Academy of Sciences
 Newelska 6, 01-447 Warsaw, Poland
 Email: a.skowron@mimuw.edu.pl

Abstract—This article is a continuation of previous work, in which a new method of decision tree construction was presented. That method is based on the use of so-called verifying cuts, which can provide knowledge obtained from the attributes frequently eliminated when greedy methods of the choice of singleton best cuts are applied. Till now only one strategy of choosing verifying cuts was examined. It exploits a measure based on a number of pairs of objects discerned by a chosen cut. In this paper, we examine two additional measures used for determining the best verifying cuts. They are based on Gini's Index and Entropy. The paper includes the results of experiments that have been performed on data obtained from biomedical database and machine learning repositories.

I. INTRODUCTION

DECISION tree with verifying cuts [1] (denoted by v -tree) is a method of decision tree construction formed in response to the problem of classification data with a large number of attributes. Such data can contain a lot of attributes that bear similarity with respect to the quality of potential cuts but significantly different with respect to domain knowledge represented. In contrast to the method of classifier construction based on the decision tree with local discretization techniques known from literature (see, e.g., [2], [4], [8]), for which always singleton best cuts are used and therefore there are serious doubts as to the validity of such approach, v -tree uses the so-called verifying cuts. They are additional cuts, which enable to evaluate the quality of cuts in tree nodes during classification of objects. Our experiments conducted on real data and described in [1] have shown that data with numerous set of attributes constitutes a class of data where v -tree outperforms the conventional approach. However, only one technique of choosing verifying cuts was tested, namely the one based on the maximization of the number of discerned object pairs with different decision class membership. Accordingly, the following hypothesis arose. Perhaps, using another measures of determining the quality of verifying cuts v -tree can enhance its effectiveness. Another question is whether new techniques of choosing verifying cuts may be helpful in the case of input data with non-numerous set of attributes. In this paper two another techniques of verifying cuts construction based on the Entropy and the Gini's Index are tested. Furthermore, we conducted comparative experiments using these two approaches and the one described in [1].

II. CLASSICAL DISCRETIZATION TREE

Decision tree of the local discretization [2] is a technique of a binary tree construction based on supervised discretization which introduces iterative binary partitioning of data set into groups with respect to the value of certain attribute. This algorithm is well known from literature (see, e.g., [4], [8]), therefore we will refer to it as the *classical method*.

The greedy method of choosing a pair - an attribute and its value (for numeric attributes often called the cut), which are used in the process of data partitioning - is a key element of the discussed local discretization tree construction method and is taking into consideration decision attribute values of training objects. In construction of local discretization tree we decided to use two various measures of best cut, i.e., Information Gain and Gini's Index.

1) *Information Gain measure*: First method for calculating quality of cuts that was chosen for our research is Information Gain - approach used in C4.5 algorithm [9]. The method uses concept of Entropy which was described by Claude Shannon in his work on information theory [10]. In relation to construction of decision trees of the local discretization, this measure represents diversity of objects set that corresponds to particular node in tree. Thus, let X be the set of objects which comprises of two decision classes - C_0 and C_1 . Furthermore, $p_0 = \frac{|C_0|}{|X|}$ and $p_1 = \frac{|C_1|}{|X|}$ are the distribution of C_0 and C_1 in the set X . Therefore the entropy is calculated by the following expression: $Entropy(X) = -\sum_{i=0}^1 p_i * \log_2 p_i$. The quality of a binary partition, which is defined according to the cut value c in the set X of objects, is computed by Information Gain measure as follows.

$$Gain(c, X) = Entropy(X) - \sum_{i=0}^1 \frac{|X_i|}{|X|} * Entropy(X_i) \quad (1)$$

where X_i for $i = 0, 1$ are subsets of X , that corresponds to split which is defined by cut value c . The value of information gain is determined for all possible cuts and next a cut is greedily chosen which maximizes that measure. Surely, this method can be generalized to greater number of decision classes than 2.

2) *Gini's Index measure*: An alternative example of measuring the quality of cuts, that was used in our work, is the method used in CART algorithm [5] - Gini's Index. For indications such as in the previous paragraph, if X contains examples from classes C_0 and C_1 , the measure of diversity of X set is defined as $Gini(X) = 1 - \sum_{i=0}^1 p_i^2$ where p_i is the class distribution in X . Moreover, the quality of cut can be calculated as follows.

$$G(c, X) = Gini(X) - \sum_{i=0}^1 \frac{|X_i|}{|X|} * Gini(X_i) \quad (2)$$

As previously, the best cut is chosen greedily from all possible cuts. Furthermore, this approach can also be generalized to more than two decision classes.

Binary tree classifiers for which Information Gain or Gini's Index were used in the procedure of best cut finding we call in this paper the Entropy-C classifier and the Gini-C classifier, respectively.

III. DECISION TREE WITH VERIFYING CUTS

As in the previous article [1], the motivation for our work concerns the validity of classical approach used to data sets with large number of attributes. We recall that the method chooses only one split (for a single attribute) with the best quality based on the selected measure, at the given step of searching for optimal binary partitions. In such case, the method would greedily eliminate the information contained in attributes, which are similar in terms of quality of potential cuts, but are different with respect to domain knowledge, which they represent. The main idea presented in [1] is based on the fact that at a given stage of searching for partitions of a set of attributes, the family of k -binary verifying partitions is determined after construction of the optimal binary partition of a set of objects. Obviously, it refers to family of partitions which are similar to the optimal partition and concerns other attributes than the attributes used in the optimal partition. Moreover, the concept of similarity depends on measure which is used to determine the best split. Thus, in the case called *MaxDiscPair*, the similarity means to distinguish between set of pairs of objects of different decision classes as similar as possible to the optimal partition. Whereas in the case of measures based on Gini's Index and Information Gain, verifying partitions should separate objects in possible the same manner as the main split. The differences in selecting verifying cuts between all three measures are such that in case of discernibility-based measure we determine objects that are separated by both cuts simultaneously, while in case of Gini's Index and Information Gain based measures, the candidate for additional cut divides the set of object independently of the main cut.

The algorithm for construction of a decision tree with verifying cuts [1], has been enhanced to use all three measures. Assume that a decision table $\mathbf{A} = (U, A, d)$, a parameter k (in our experiments the value of k was empirically chosen) belonging to natural numbers, template T_p defined by the

optimal cut and template T_{p_i} (for $i = 1, \dots, k$) defined by the verifying cuts are given. Depending on a chosen measure the following criteria are optimized during the v-tree construction:

MaxDiscPair (see [1]) – criterion is maximized.

Entropy based measure – criterion is minimized:

$$EM(p_i) = \begin{cases} 0 & \text{for } \frac{|W|}{|\mathbf{A}|} \geq t_w \\ |ES(\mathbf{A}(T_p), \mathbf{A}(\neg T_p)) - ES(\mathbf{A}(T_{p_i}), \mathbf{A}(\neg T_{p_i}))| & \text{otherwise,} \end{cases}$$

where:

- W is a set of objects that at the same time are not matching patterns T_p and T_{p_i} as also patterns $\neg T_p$ and $\neg T_{p_i}$ (for $i = 1, \dots, k$),
- t_w is a fixed threshold (t_w was equal 0.1 and 0.05 in our experiments for "microarray" and "normal" data, respectively),
- $ES(\mathbf{A}(T_q), \mathbf{A}(\neg T_q)) = \frac{|\mathbf{A}(T_q)|}{|\mathbf{A}|} * Entropy(\mathbf{A}(T_q)) + \frac{|\mathbf{A}(\neg T_q)|}{|\mathbf{A}|} * Entropy(\mathbf{A}(\neg T_q))$ ($q \in \{p, p_1, \dots, p_k\}$) is the weighted sum of entropies of partitions p and p_i , respectively ($q \in \{p, p_1, \dots, p_k\}$).

Gini's Index based measure – criterion is minimized:

$$GM(p_i) = \begin{cases} 0 & \text{for } \frac{|W|}{|\mathbf{A}|} \geq t_w \\ |GS(\mathbf{A}(T_p), \mathbf{A}(\neg T_p)) - GS(\mathbf{A}(T_{p_i}), \mathbf{A}(\neg T_{p_i}))| & \text{otherwise} \end{cases},$$

where:

- W is a set of objects that at the same time are not matching patterns T_p and T_{p_i} as also patterns $\neg T_p$ and $\neg T_{p_i}$ (for $i = 1, \dots, k$),
- t_w is a fixed threshold (t_w was equal 0.1 and 0.05 in our experiments for "microarray" and "normal" data respectively),
- $GS(\mathbf{A}(T_q), \mathbf{A}(\neg T_q)) = \frac{|\mathbf{A}(T_q)|}{|\mathbf{A}|} * Gini(\mathbf{A}(T_q)) + \frac{|\mathbf{A}(\neg T_q)|}{|\mathbf{A}|} * Gini(\mathbf{A}(\neg T_q))$ is the weighted sum of ginis of partitions p and p_i , respectively ($q \in \{p, p_1, \dots, p_k\}$).

The stop condition mentioned in algorithm of v-tree construction is separation of all possible pairs of objects from different decision classes. It is worth pointing out, that the only part of the above algorithm, which would increase the time complexity compared to the classical algorithm from Section II is step 3. This step can be performed in time $O(n \cdot \log n \cdot m)$, where n is the number of objects and m is the number of attributes.

The determination of the best verification split for the symbolic attributes can be done in time $O(n \cdot l)$, where l is the number of values of symbolic attribute.

Below we present the algorithm for selection of verifying partition for the constructed earlier binary partition p . We assume that the verifying split is determined by a numerical attribute. For ease of discussion, we consider a situation that

there are only two decision classes C_0 and C_1 in the data. This method can be easily generalized to the case of more than two decision classes. The output of this algorithm is the computed collection of cuts that verify partition p .

Algorithm *Selection of verifying cut*

Step 1 Sort the values of the numerical attribute a .

Step 2 Browsing the a attribute values from the smallest to the largest, determine for each appearing cut c the following numbers and store them into a memory (about cuts) M :
 $V_L(a, c, C_0), V_L(a, c, C_1)$ - number of objects from decision class C_0 or C_1 with values of attribute a smaller then c ,
 $L(a, c, C_0, T_p), L(a, c, C_1, T_p)$ - number of objects from decision class C_0 or C_1 with values of attribute a smaller than c and at the same time matching the pattern T_p .

Step 3 Browsing the a attribute values from the highest to the lowest, determine for each appearing cut c the following numbers and place them in a memory (about cuts) M :
 $V_H(a, c, C_0), V_H(a, c, C_1)$ - number of objects from decision class C_0 or C_1 with values of a greater then or equal c ,
 $H(a, c, C_0, \neg T_p), H(a, c, C_1, \neg T_p)$ - number of objects from decision class C_0 or C_1 with values of attribute a greater than or equal to c and at the same time matching the pattern $\neg T_p$.

Step 4 Using information from the memory M , determine the quality of cuts on a in the manner that depends on measure selected for determining the quality of cuts:

MaxDiscPair (see [1])

Entropy:

1. determine the size of set W :

$$|W| = |\mathbf{A}| - (L(a, c, C_0, T_p) + L(a, c, C_1, T_p) + H(a, c, C_0, \neg T_p) + H(a, c, C_1, \neg T_p)),$$

2. discard cuts for which $\frac{|W|}{|\mathbf{A}|} > t_w$,

3. determine the sizes of tables designated by cut c :

$$|\mathbf{A}(T_c)| = V_L(a, c, C_0) + V_L(a, c, C_1),$$

$$|\mathbf{A}(\neg T_c)| = V_H(a, c, C_0) + V_H(a, c, C_1),$$

4. compute the weighted sum of entropies for partition designated by cut c from $ES(\mathbf{A}(T_c), \mathbf{A}(\neg T_c))$,

5. determine the optimum cutting such that the value of $|ES(\mathbf{A}(T_p), \mathbf{A}(\neg T_p)) - ES(\mathbf{A}(T_c), \mathbf{A}(\neg T_c))|$ is the smallest.

Gini:

1. determine the size of set W :

$$|W| = |\mathbf{A}| - (L(a, c, C_0, T_p) + L(a, c, C_1, T_p) + H(a, c, C_0, \neg T_p) + H(a, c, C_1, \neg T_p)),$$

2. discard cuts for which $\frac{|W|}{|\mathbf{A}|} > t_w$,

3. determine the sizes of tables designated by cut c :

$$|\mathbf{A}(T_c)| = V_L(a, c, C_0) + V_L(a, c, C_1),$$

$$|\mathbf{A}(\neg T_c)| = V_H(a, c, C_0) + V_H(a, c, C_1),$$

4. compute the weighted sum of Ginis for partition designated by cut c : $GS(\mathbf{A}(T_c), \mathbf{A}(\neg T_c))$,

5. determine the optimum cutting such that the value of $|GS(\mathbf{A}(T_p), \mathbf{A}(\neg T_p)) - GS(\mathbf{A}(T_c), \mathbf{A}(\neg T_c))|$ is smallest.

Assuming that the memory about cuts M is accessible in constant time, the above algorithms runs in time $O(n \cdot \log n)$, where n is the number of objects (due to the sorting of objects

on the basis of the a attribute).

The algorithm for an object classification, using a v-tree with verifying partitions was introduced in [1].

The classifiers constructed with the use of v-decision tree will be called here the *MaxDiscPair-V* classifier, *Entropy-V* classifier or *Gini-V* classifier – depending on used measure during construction, respectively. Note that the algorithm [1] to classify the object in the node utilizes a single tree only when all verifying cuts classify the object just as the main partition p . In other cases, the classification is done by both subtrees. Then the following two cases are considered. The first case refers to the situation when the two subtrees returned the same decision value. Then the value of the node is returned as the decision. The second case refers to a situation where one of the subtrees returned one decision value, and the second subtree the other one. Then that node returns a decision coming from the subtree, which is associated with a greater number of such verifying patterns that classify a test object for this tree. If the numbers of verifying cuts are equal, then the decision comes from subtree selected nondeterministically.

IV. EXPERIMENTS AND RESULTS

To verify the effectiveness of classifiers based on our approach, we have implemented classifiers based on the verifying cuts in the programming library CommoDM (Common Data Minning), which is a continuation of the RSES-lib library (forming the kernel of the RSES system [3]). The experiments have been performed on the data sets obtained from Kent Ridge Biomedical Dataset [7], UCI ML repository (see [11]) and website of The Elements of Statistical Learning book (Statweb)(see [6]). 6 data collections from the first source relates to microarray experiments and they are characterized by a large number of attributes. Our experiments were conducted on the merged original training and testing data sets. The objective of conducted experiments was to test the quality of the classification algorithms discussed in this paper. Table I presents the experimental results received for given data sets and two discretization methods (Entropy and Gini Index based ones) applied to classical tree and v-tree. The counterparts received for discretization method based on maximum number of discernible pairs is presented in [1].

For determining quality of classifiers we applied 10 fold cross-validation technique, which was repeated 10 times for every data set (i.e., 100 cycles of a train-and-test scheme was conducted). The final result of the algorithm is the average of 100 cycles. Popular parameters accuracy (ACC) and coverage (COV) were used to measure the classification success. It is easy to observe that in most cases better results were obtained when the v-tree classifier was applied, both for entropy and Gini's Index based discretization method. That observation is confirmed by the Wilcoxon mached pairs test with 0,05 level of significance in the following cases: (1) [ACC, Entropy-V classifier, num] > [ACC, Entropy-C classifier, num], i.e., the classification quality expressed by ACC coefficient and entropy based discretization method for v-tree is better than for c-tree when applied for data with numerous sets of

TABLE I
THE AVERAGE ACC AND COV WITH STD. DEV. OF EXPERIMENTS FOR C-TREE AND V-TREE AND 2 DISCRETIZATION METHODS

Method	Entropy-C classifier				Entropy-V classifier				Gini-C classifier				Gini-V classifier			
	Acc	Std dev	Cov	Std dev	Acc	Std dev	Cov	Std dev	Acc	Std dev	Cov	Std dev	Acc	Std dev	Cov	Std dev
lymphoma	0.788	0.041	0.945	0.022	0.836	0.043	1.0	0.0	0.795	0.042	0.943	0.021	0.845	0.047	1.0	0.0
leukemia	0.803	0.037	1.0	0.0	0.91	0.023	1.0	0.0	0.819	0.035	1.0	0.0	0.9	0.04	1.0	0.0
colon	0.75	0.045	1.0	0.0	0.756	0.033	1.0	0.0	0.766	0.03	1.0	0.0	0.765	0.045	1.0	0.0
lung	0.925	0.014	1.0	0.0	0.957	0.013	1.0	0.0	0.925	0.014	1.0	0.0	0.956	0.02	1.0	0.0
prostate	0.837	0.026	1.0	0.0	0.876	0.024	0.999	0.002	0.84	0.033	1.0	0.0	0.847	0.014	1.0	0.0
ovarian	0.976	0.004	1.0	0.0	0.981	0.004	0.999	0.002	0.976	0.004	1.0	0.0	0.98	0.006	1.0	0.0
audiology	0.625	0.025	0.74	0.017	0.538	0.039	0.996	0.004	0.66	0.02	0.827	0.026	0.618	0.021	1.0	0.0
biodeg	0.817	0.009	1.0	0.0	0.818	0.009	1.0	0.0	0.809	0.008	1.0	0.0	0.813	0.011	1.0	0.0
conn.bench	0.74	0.03	1.0	0.0	0.752	0.024	1.0	0.0	0.695	0.02	1.0	0.0	0.722	0.024	1.0	0.0
cylinder	0.708	0.015	0.813	0.011	0.73	0.013	1.0	0.0	0.703	0.014	0.811	0.014	0.74	0.015	1.0	0.0
dermatol.	0.945	0.007	1.0	0.001	0.954	0.008	1.0	0.0	0.939	0.005	0.998	0.001	0.952	0.006	1.0	0.0
mushroom	1.0	0.0	1.0	0.0	0.985	0.0	1.0	0.0	1.0	0.0	0.787	0.0	1.0	0.0	1.0	0.0
flags	0.629	0.023	1.0	0.0	0.632	0.019	0.999	0.002	0.605	0.017	1.0	0.0	0.609	0.019	1.0	0.0
ozone	0.953	0.003	0.843	0.004	0.96	0.002	1.0	0.0	0.947	0.004	0.822	0.002	0.96	0.003	1.0	0.0
parkinsons	0.865	0.016	1.0	0.0	0.873	0.029	1.0	0.0	0.86	0.025	1.0	0.0	0.882	0.025	1.0	0.0
SAheart	0.626	0.013	1.0	0.0	0.647	0.013	1.0	0.001	0.613	0.009	1.0	0.0	0.652	0.015	1.0	0.001
segmentat.	0.953	0.002	1.0	0.0	0.945	0.002	1.0	0.0	0.955	0.003	1.0	0.0	0.942	0.003	1.0	0.0
spam	0.921	0.002	1.0	0.0	0.915	0.003	1.0	0.0	0.913	0.002	1.0	0.0	0.893	0.003	1.0	0.0

attributes; (2) [ACC, Gini-V classifier, num] > [ACC, Gini-C classifier, num]; (3) [ACC*COV, Entropy-V classifier, num] > [ACC*COV, Entropy-C classifier, num]; (4) [ACC*COV, Gini-V classifier, num] > [ACC*COV, Gini-C classifier, num]; (5) [ACC*COV, Gini-V classifier, non-num] > [ACC*COV, Gini-C classifier, non-num];

We have also checked, separately for c-tree and v-tree classifiers, whether one of the three tested discretization methods leads to better classification quality. We used the Friedman test. It showed that none of the three methods has such property. Both Wilcoxon matched pairs test and Friedman test were used in the form implemented in Statistica program ver. 10.

V. CONCLUSION

In the paper, we presented Entropy based measure and Gini's Index based one applied to determining decision tree with verifying cuts classifier. We checked usefulness of those algorithms on - 18 input data sets. Experiments have confirmed (with statistical significance) that v-tree is relevant classifier for data with a large number of attributes. Used 12 input data with non-numerous set of attributes was too little family of data to express analogous observation when input data do not have really many attributes. Moreover, none of three methods of local discretization proved to be better than remaining ones. The novelty of the paper is important because the experimental results showed that the employment of the knowledge contained in the redundant attributes increases the quality of the classifiers not only for the previously used measure. The conducted experiments have proved the correctness of our assumptions that our method will be also effective for the use of new measures. We expect that the methods may be used in various fields.

ACKNOWLEDGEMENT

This work was partially supported by two following grants of the Polish National Science Centre: DEC-

2013/09/B/ST6/01568, DEC-2013/09/B/NZ5/00758, and also by the Centre for Innovation and Transfer of Natural Sciences and Engineering Knowledge of University of Rzeszów, Poland. Andrzej Skowron was also partially supported by the Polish National Science Centre (NCN) grants DEC-2011/01/D/ST6/06981, as well as by the Polish National Centre for Research and Development (NCBiR).

REFERENCES

- [1] Bazan, J., G., Bazan-Socha, S., Buregwa-Czuma, Dydo, L., Rzasa, W., Skowron, A.: A classifier based on a decision tree with verifying cuts. *Fundamenta Informaticae*, vol. 143, no. 1-2, pp. 1-18, 2016
- [2] Bazan, J.G., Bazan-Socha, S., Buregwa-Czuma, S., Pardel, P.W., Sokolowska, B.: Predicting the presence of serious coronary artery disease based on 24 hour Holter ECG monitoring. In: M. Ganzha, L. Maciaszek, M. Paprzycki (eds.), *Proceedings of the Federated Conference on Computer Science and Information Systems*, 2012, pp. 279-286, IEEE Xplore - digital library.
- [3] Bazan, J. G., Szczuka, M.: The Rough Set Exploration System. *Transactions on Rough Sets*, III, LNCS 3400, 2005, pp. 37-56.
- [4] Bazan, J. G., Nguyen, H. S., Nguyen, S. H., Synak, P., Wróblewski, J.: Rough set algorithms in classification problems. In: L. Polkowski, T. Y. Lin, S. Tsumoto (eds.), "Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems," *Studies in Fuzziness and Soft Computing*, Springer-Verlag/Physica-Verlag, vol. 56, 2000, pp. 49-88.
- [5] Breiman, L. et. al., *Classification and Regression Trees*. Wadsworth, Belmont, 1984.
- [6] The Elements of Statistical Learning repository, <http://statweb.stanford.edu/tibs/ElemStatLearn/datasets/>
- [7] Kent Ridge Biomedical Dataset repository, <http://datam.i2r.a-star.edu.sg/datasets/krbd/>
- [8] Nguyen, H. S.: Approximate Boolean Reasoning: Foundations and Applications in Data Mining, *Transactions on Rough Sets*, V, LNCS 4100, 2006, pp. 334-506.
- [9] Quinlan, J. R.: *C4.5: Programs for machine learning*, Morgan Kaufmann, San Mateo, California (1993)
- [10] Shannon, C.E.: A mathematical theory of communication, *Bell System Technical Journal*, 27 (1948), pp. 379-423.
- [11] UC Irvine Machine Learning Repository, <http://archive.ics.uci.edu/ml/>