Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

The CART decision tree for mining data streams

Leszek Rutkowski [a,b,*], Maciej Jaworski [a], Lena Pietruczuk [a], Piotr Duda [a]

[a] *Institute of Computational Intelligence, Czestochowa University of Technology, ul. Armii Krajowej 36, 42-200 Czestochowa, Poland*
[b] *Information Technology Institute, Academy of Management, 90-113 Łódź, Poland*

ARTICLE INFO

Article history:
Received 13 March 2013
Received in revised form 28 August 2013
Accepted 30 December 2013
Available online 11 January 2014

Keywords:
Data steam
Decision tree
CART
Gini index
Gaussian approximation

ABSTRACT

One of the most popular tools for mining data streams are decision trees. In this paper we propose a new algorithm, which is based on the commonly known CART algorithm. The most important task in constructing decision trees for data streams is to determine the best attribute to make a split in the considered node. To solve this problem we apply the Gaussian approximation. The presented algorithm allows to obtain high accuracy of classification, with a short processing time. The main result of this paper is the theorem showing that the best attribute computed in considered node according to the available data sample is the same, with some high probability, as the attribute derived from the whole data stream.

1. Introduction

Among the plenty of techniques and methods used in machine learning or data mining, the classification seems to be one of the most important [14,21,31,35]. Let A^i denotes the set of possible values of attribute a^i, for $i = 1,\ldots,D$. The aim of the classification task is to find a classifier $h : A^1 \times \cdots \times A^D \to \{1,\ldots,K\}$ based on the training dataset $\mathbf{S} \subset A^1 \times \cdots \times A^D \times \{1,\ldots,K\}$. The dataset $\mathbf{S}$ consists of n elements $s_m = (v_m, k_m) = ([v_m^1,\ldots,v_m^D], k_m),\ m = 1,\ldots,n$, where

- $v_m^i \in A^i$ is the value of attribute a^i for data element s_m,
- $k_m \in \{1,\ldots,K\}$ is a class of data element s_m.

The classifier h is used to assign a class $k \in \{1,\ldots,K\}$ to unlabeled data elements $v \in A^1 \times \cdots \times A^D$. For static datasets a variety of classification methods have been proposed in literature. The most popular are neural networks [29,30], k-nearest neighbors [4] or decision trees [3,26,27], which are within the scope of this paper. The decision tree is a structure composed of nodes and branches. Terminal nodes are called leaves. To each node L_q, which is not a leaf, an appropriate splitting attribute a^i is assigned. The assignment of the attribute to the considered node is the crucial part of the decision tree construction algorithm. Usually the choice of the attribute is based on some impurity measure, calculated for the corresponding subset $\mathbf{S}_q$ of the training dataset $\mathbf{S}$. The impurity measure is used to calculate the split measure function for each attribute. According to the chosen attribute, the node is split into child nodes, which are connected with their parent nodes by branches. There exist two types of decision trees: binary and non-binary. In the case of non-binary tree, the node is split into as many children as the number of elements of set A^i. Each branch is labeled by a single value of attribute a^i. If the tree is binary, the node is split

[*] Corresponding author at: Institute of Computational Intelligence, Czestochowa University of Technology, ul. Armii Krajowej 36, 42-200 Czestochowa, Poland. Tel.: +48 34 32 50 546.

E-mail addresses: leszek.rutkowski@iisi.pcz.pl (L. Rutkowski), maciej.jaworski@iisi.pcz.pl (M. Jaworski), lena.pietruczuk@iisi.pcz.pl (L. Pietruczuk), piotr.duda@iisi.pcz.pl (P. Duda).

0020-0255/$ - see front matter © 2014 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2013.12.060

into two child nodes. The branches are labeled by some complementary subsets of A^i. According to the branches, the set $\mathbf{S}_q$ is partitioned into subsets, which then become the training subsets in the corresponding children nodes. Leaves serve to label unclassified data elements.

The existing algorithms for decision trees construction differ mainly in the two fields mentioned above: type of tree (binary or non-binary) and type of impurity measure. The ID3 algorithm [26], for example, produces non-binary trees. As the impurity measure the information entropy is applied. The split measure function, based on it, is called the information gain. An upgraded version of the ID3 algorithm, also based on the information entropy, is the C4.5 algorithm [27]. In this algorithm an additional function, called the split information, is proposed. It takes high values for attributes with large domains. As the split measure function in the C4.5 algorithm, the ratio of the information gain and the split information is used. In the CART algorithm [3] binary trees are constructed. The impurity measure is in the form of Gini index.

The algorithms mentioned above (ID3, C4.5 and CART) are designed for static datasets. They cannot be applied directly to data streams [1,2,7,11–13,24], which are of infinite size. Moreover, in case of data streams data elements income to the system continuously with very high rates. Additionally, the concept drift may occur [8,10,17,20,22,34], which means that the concept of data evolve in time. In the literature there are various approaches to deal with data streams. In recent decade an appropriate tool to solve data streams problems is incremental learning [6,15,25]. Among few characterizations of incremental learning presented in the literature we cite [15] stating that 'incremental learning should be capable of learning a new information and retaining the previously acquired knowledge, without having access to the previously seen data'. It is easily seen that the approach based on the decision trees possesses main features of the incremental learning.

In this paper we present a method to adapt the CART algorithm to deal with data streams. The main problem is to determine the best attribute in each node. Since it is not possible to compute the values of split measure based on infinite dataset, they should be estimated using the sample of data in considered node. Then, with some probability, one can say whether the best attribute according to this sample is also the best with respect to the whole stream. In the literature there are few approaches to solve this problem:

(a) The commonly known algorithm called 'Hoeffding's Tree' was introduced by P. Domingos and G. Hulten in [5]. The main mathematical tool used in this algorithm was the Hoeffding's Theorem [16] in the form:

Theorem 1. *If $X_1, X_2, \ldots, X_n$ are independent random variables and $a_i \leqslant X_i \leqslant b_i$ $(i = 1, 2, \ldots, n)$, then for $\epsilon > 0$*

$$P\{\overline{X} - E[\overline{X}] \geqslant \epsilon\} \leqslant e^{-2n^2\epsilon^2/\sum_{i=1}^{n}(b_i-a_i)^2}, \tag{1}$$

where $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$ and $E[\overline{X}]$ is expected value of $\overline{X}$.

For $a_i = a$ and $b_i = b$, $(i = 1, 2, \ldots, n)$ it states that after n observations the true mean of the random variable of range $R = b - a$ does not differ from the estimated mean by more than

$$\epsilon_H = \sqrt{\frac{R^2 \ln 1/\alpha}{2n}} \tag{2}$$

with probability $1 - \alpha$. However, we would like to emphasize that the Hoeffding's bound is wrong tool to solve the problem of choosing the best attribute to make a split in the node. This observation follows from the fact that the split measures, like information gain and Gini index, cannot be presented as a sum of elements and they are using only frequency of elements. Moreover, Theorem 1 is applicable only for numerical data. Therefore the idea presented in [5] violates the assumptions of Theorem 1 and the concept of Hoeffding Trees has no theoretical justification.

(b) In [33] the authors proposed new method in which they used the McDiarmid's inequality [23] instead of Hoeffding's bound as a tool for choosing the best attribute to make a split. First the function $f(\mathbf{S})$ was proposed as a difference between the values of Gini indices of two attributes

$$f(\mathbf{S}) = Gini_{a^x}(\mathbf{S}) - Gini_{a^y}(\mathbf{S}). \tag{3}$$

By applying the McDiarmid's inequality to independent elements in dataset $\mathbf{S}$ authors obtained the value of

$$\epsilon_M = 8\sqrt{\frac{\ln(1/\alpha)}{2n}}, \tag{4}$$

such that for any fixed α, if $f(\mathbf{S}) > \epsilon_M$, then with probability $1 - \alpha$ attribute a^x is better to make a split than attribute a^y.

(c) In [18] the authors proposed a method, based on the Multivariate Delta Method, for determining if the best attribute calculated from the data sample in considered node is also the best according to the whole stream. Even though the idea was valuable, the authors omitted the issue of calculating some of the necessary parameters what is not trivial. Therefore, the method does not have any practical application.