Pattern Recognition 52 (2016) 33–45

Contents lists available at ScienceDirect

Pattern Recognition

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

Online active learning of decision trees with evidential data

Liyao Ma [a,*], Sébastien Destercke [b], Yong Wang [a]

[a] Department of Automation, University of Science and Technology of China, Hefei, China
[b] UMR7253 Heudiasyc, Centre de Recherches de Royallieu, Compiègne, France

ARTICLE INFO

Article history:
Received 2 April 2015
Received in revised form
28 September 2015
Accepted 19 October 2015
Available online 30 October 2015

Keywords:
Decision tree
Active learning
Evidential likelihood
Uncertain data
Belief functions

ABSTRACT

Learning from uncertain data has attracted increasing attention in recent years. In this paper, we propose a decision tree learning method that can not only handle uncertain data, but also reduce epistemic uncertainty by querying the most valuable uncertain instances within the learning procedure. Specifically, we use entropy intervals extracted from the evidential likelihood to query uncertain training instances when needed, with the goal to improve the selection of the splitting attribute. Experimental results under various conditions confirm the interest of the proposed approach.

1. Introduction

Decision trees, as one of the best-known approaches for classification, are widely used due to their good learning capabilities and simplicity to understand. However, classical decision trees can only handle certain data whose values are precisely known. Those uncertain instances, despite the fact that they may contain useful information, are usually ignored or removed by replacing them with precise instances when building decision trees [1], potentially leading to a loss of accuracy. Different approaches have been proposed to overcome this drawback, such as probabilistic decision trees developed by Quinlan [2] and Tsang et al. [3], fuzzy decision trees proposed by Yuan et al. [4] and Wang et al. [5] or uncertain decision trees proposed by Qin et al. [6] and Liang et al. [7].

The interest for integrating uncertain data in learning methods has been growing in the recent years [8–11]. While probability theory is still the most commonly used tool to model this uncertainty, various authors (see the special issue [12] and papers within it) have argued that probability cannot always adequately represent data uncertainty (often termed epistemic uncertainty). For instance, probabilistic modelling is unable to model faithfully set-valued observations. In this paper, we will work with a more general theory, the theory of belief functions (also called Dempster–Shafer theory or evidence theory) [13,14], which has the advantage to include both sets and probabilities as special cases.

Embedding belief functions within the learning of decision trees has already been investigated in the past. Elouedi et al. [15,16] discussed belief decision tree construction under the TBM model. Vannoorenberghe [17,18] concentrated on the aggregation of belief decision trees. Sutton-Charani et al. [19,20] proposed to estimate tree parameters by maximizing evidential likelihood function using the E^2M algorithm [21].

Although those proposals deal with uncertain data modelled by belief functions, none of them have looked at the issue of reducing data uncertainty through information querying. This is what we propose in this paper: to query uncertain data during the tree learning procedure, in order to improve its performances. In some sense, this idea is very close to the one of active learning [22], where the learning algorithm can achieve higher accuracies by selecting the most valuable unlabelled instances and querying their true labels. There are however two significant differences between our proposal and the usual active learning: we consider generic uncertain data (modelled by belief functions) and we query while learning the model (a process close to online active learning [23]), rather than "learning then querying". To our knowledge, this paper is the first to propose an evidential online active learning method. We do this by relying on the notion of evidential likelihood [24] to select the items to query in order to improve the split selection.

Apart from the query method mentioned above, our proposal also allows us to derive an alternative approach to learn decision trees from uncertain data: to choose the best split during the tree learning, we do not only consider the entropy value corresponding to the maximum likelihood, but extract entropy intervals from the evidential likelihood. The proposed approach extends both the

* Corresponding author. Postal address: 9-307 USTC, P.O.Box 4, Hefei 230027, China. Tel.: +86 18756916639.
E-mail addresses: liyma@mail.ustc.edu.cn (L. Ma), sebastien.destercke@hds.utc.fr (S. Destercke), yongwang@ustc.edu.cn (Y. Wang).

http://dx.doi.org/10.1016/j.patcog.2015.10.014
0031-3203/

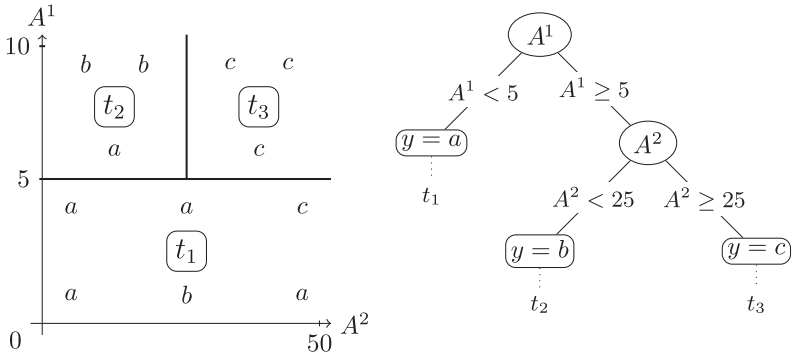

Fig. 1. Example of decision tree and associated partition.

classical C4.5 decision trees and the E^2M decision trees [19] applied to uncertain outputs (but certain inputs).

Section 2 recalls the necessary material on belief functions, classical decision trees and evidential likelihood. In Section 3, we discuss in detail the overall tree induction procedure, as well as how the data are queried during the learning. Section 4 details some experiments on classical UCI data sets, and compares the results of the proposed approach with decision trees without querying. Finally, conclusions are given in Section 5.

2. Settings and basic definitions

The purpose of a classification approach is to build a model $\mathcal{M}$ that maps a feature vector $\boldsymbol{x} = (x^1, \ldots, x^k) \in A^1 \times A^2 \times \cdots \times A^k$ taking its values on k attributes, to an output class $y \in \mathcal{C} = \{C_1, \ldots, C_\ell\}$ taking its value among ℓ classes. Each attribute $A^i = \{a_1^i, \ldots, a_{r_i}^i\}$ has r_i possible values. This model is then used to make predictions on new instances $\boldsymbol{x}$ whose classes are unknown. Typically this model $\mathcal{M}$ (a decision tree, a Bayes network, a logistic regression, …) is learned from a training set of precise data, denoted as

$$T = \begin{pmatrix} \boldsymbol{x_1}, y_1 \\ \vdots \\ \boldsymbol{x_n}, y_n \end{pmatrix} = \begin{pmatrix} x_1^1, \ldots, x_1^k, y_1 \\ \vdots \\ x_n^1, \ldots, x_n^k, y_n \end{pmatrix}.$$

However, in practical applications, it is possible to have uncertainty in the inputs (feature vectors) and/or the outputs (classification labels). This uncertainty is epistemic,[1] in the sense that a given x_j^i or y_i has a unique true value that may be ill-known. As recalled in the introduction, the adequacy of probability theory to model such uncertainty is questionable, hence in this paper we will model uncertainty by belief functions. We will consider the case where the input is certain and where only the output is uncertain (a classical assumption in active learning).

2.1. Belief functions

Let $\mathcal{C}$ be a finite space, called the frame of discernment, containing all the possible exclusive values that a variable (here, the output class y) can take. When the true value of y is ill-known, our uncertainty about it can be modelled by a mass function $m_y : 2^{\mathcal{C}} \to [0, 1]$, such that $m_y(\varnothing) = 0$ and

$$\sum_{E \subseteq \mathcal{C}} m_y(E) = 1. \tag{1}$$

A subset E of $\mathcal{C}$ is called a *focal set* of m_y if $m_y(E) > 0$. $m_y(E)$ can then be interpreted as the amount of evidence indicating that the true value is in E. The following typical mass functions show that this model extends both set-valued and probabilistic uncertainty models:

- a *vacuous* mass is such that $m_y(\mathcal{C}) = 1$. It represents total ignorance;
- a *Bayesian* mass is such that $m_y(E) > 0$ iff $|E| = 1$. It is equivalent to a probability distribution;
- a *logical (categorical)* mass is such that $m_y(E) = 1$ for some E. It is equivalent to the set E.

The associated belief and plausibility functions, which are in one-to-one relations with the mass function m_y, are defined as:

$$Bel_y(B) = \sum_{E \subseteq B} m_y(E), \tag{2}$$

$$Pl_y(B) = \sum_{E \cap B \neq \varnothing} m_y(E), \tag{3}$$

for all $B \subseteq \mathcal{C}$. The belief function measures how much event B is certain (it sums masses implying B), while the plausibility measures how much event B is consistent with available evidence. The function $pl_y : \mathcal{C} \to [0, 1]$ such that $pl_y(w) = Pl_y(\{w\})$ is called the contour function associated to m_y.

When uncertain outputs are modelled by mass functions, the training set becomes

$$T = \begin{pmatrix} \boldsymbol{x_1}, m_{y_1} \\ \vdots \\ \boldsymbol{x_n}, m_{y_n} \end{pmatrix} = \begin{pmatrix} x_1^1, \ldots, x_1^k, m_{y_1} \\ \vdots \\ x_n^1, \ldots, x_n^k, m_{y_n} \end{pmatrix}.$$

2.2. Decision trees

Decision trees [25] are common classifiers that induce a rooted tree structure, in which leaves (the terminal nodes) represent class labels and branches correspond to features with associated values leading to the nodes.

To be able to predict the class of an instance with k attributes, decision trees are induced top-down from a training set T. Every decision node (non-terminal node) is associated with a splitting attribute selected by a strategy that can be based on different algorithms and purity measures [26]. The selection and splitting process is then repeated recursively until a stopping criterion is met. The achieved decision tree then determines a partition of the instance space, and associates a class to each element of this partition. This means that each terminal node (or leaf) of a decision tree can be associated to an element of the partition. Fig. 1 shows a classical decision tree and the associated partition when $k = 2$ and $\mathcal{C} = \{a, b, c\}$.

Several algorithms have been proposed for decision tree learning, among which ID3 [25], C4.5 [27] and CART [28] are the most common ones. In this paper, we take the basic C4.5 algorithm

[1] By opposition to the so-called aleatory uncertainty, which concerns a stochastic behaviour.