



HAL
open science

Splitting Rules for Monotone Fuzzy Decision Trees

Christophe Marsala, Davide Petturiti

► **To cite this version:**

Christophe Marsala, Davide Petturiti. Splitting Rules for Monotone Fuzzy Decision Trees. EUSFLAT 2023: 13th Conference of the European Society for Fuzzy Logic and Technology, Sep 2023, Palma de Mallorca, Spain. pp.161-173, 10.1007/978-3-031-39965-7_14 . hal-04173012

HAL Id: hal-04173012

<https://hal.science/hal-04173012>

Submitted on 27 May 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Splitting rules for monotone fuzzy decision trees

Christophe Marsala¹ and Davide Petturiti²

¹ LIP6, Sorbonne Université CNRS, Paris, France
`christophe.marsala@lip6.fr`

² Dip. Economia, Università degli Studi di Perugia, Perugia, Italy
`davide.petturiti@unipg.it`

Abstract. This paper considers the problem of building monotone fuzzy decision trees when the attributes and the labeling function are in the form of partitions (in Ruspini’s sense) of totally ordered labels. We define a fuzzy version of Shannon and Gini rank discrimination measures, based on a definition of fuzzy dominance, to be used in the splitting phase of a fuzzy decision tree inductive construction algorithm. These extensions generalize the rank discrimination measures introduced in previous work. Afterwards, we introduce a new algorithm to build a fuzzy decision tree enforcing monotonicity and we present an experimental analysis on an artificial data set.

Keywords: Monotone fuzzy decision tree · Fuzzy rank discrimination measure · Totally ordered fuzzy partitions.

1 Introduction

Starting from the seminal paper [1], monotone classification has attracted increasing attention (see, e.g., [2, 3, 7, 18]) due to its capacity of modeling semantic concepts like preference, priority and importance. In turn, the possibility of incorporating linguistic or vague information, has naturally led to fuzzy monotone classification (see, e.g., [19, 22, 23]).

In this paper, we focus on fuzzy decision trees [11, 17, 24] in which we aim at enforcing monotonicity, relying on a set of training examples. We consider a learning problem where the data set consists of a finite number of objects described by m attributes a_j ’s, each referring to a totally ordered set of fuzzy labels (X_j, \leq) , together with a labeling function λ that refers to a totally ordered set of fuzzy classes (C, \leq) . We further assume that each attribute and the labeling function are fuzzy partitions in Ruspini’s sense [20]. Then, our goal is to build a fuzzy decision tree \mathcal{T} which encodes a labeling function $\lambda' : X_1 \times \dots \times X_m \rightarrow C$, that maps every m -tuple of attribute fuzzy labels to a fuzzy class and further satisfies monotonicity, that is

$$(x_1, \dots, x_m) \leq (y_1, \dots, y_m) \implies \lambda'(x_1, \dots, x_m) \leq \lambda'(y_1, \dots, y_m), \quad (1)$$

where $(x_1, \dots, x_m) \leq (y_1, \dots, y_m)$ stands for $x_j \leq y_j$, for $j = 1, \dots, m$.

In general, enforcing global monotonicity requires a pre-processing of the input data sets, so as to remove possible inconsistencies. Here, we face the problem by adopting a greedy approach: at each step of the building process we choose the attribute a_j “enforcing the most” a local form of monotonicity. This approach has been already exploited in previous work [12] in case of crisp data and relies on the introduction of *rank discrimination measures* (see also [8, 9]) that are inspired by classical Shannon and Gini measures. In this paper, we introduce fuzzy versions of measures introduced in [12], by relying on a suitable additive fuzzy preference structure without incomparability [5, 4, 21], used to model fuzzy dominant sets. We also propose an algorithm to build a monotone fuzzy decision tree by relying on the introduced *fuzzy rank discrimination measures* and we perform its analysis on an artificial data set.

The paper is structured as follows. In Section 2, we define Shannon and Gini fuzzy rank discrimination measures. Section 3 presents a greedy construction algorithm parameterized by the introduced fuzzy rank discrimination measures and shows an experimental analysis on an artificial data set. Finally, Section 4 gathers conclusions and future perspectives.

2 Fuzzy rank discrimination measures

In this section, after a recall on the background, fuzzy rank discrimination measures are presented. First of all, let us introduce the following notations:

- $\Omega = \{\omega_1, \dots, \omega_n\}$, a finite set of **objects**;
- $\mathcal{A} = \{a_1, \dots, a_m\}$, a finite set of **fuzzy attributes with totally ordered range of fuzzy labels**, where a_j refers to the set of labels $X_j = \{x_{j_1}, \dots, x_{j_{t_j}}\}$ and (X_j, \leq_{X_j}) is totally ordered;
- λ , a **fuzzy labelling function** referring to the set of **fuzzy classes** $C = \{c_1, \dots, c_k\}$ with (C, \leq_C) totally ordered.

To avoid cumbersome notation, in what follows we suppress the subscript X_j and C from the total orders \leq_{X_j} and \leq_C , relying on the context to clarify which relation we are referring to.

The case of crisp a_j and λ has been considered in [12]: in this case, a_j and λ correspond to (crisp) partitions of Ω :

$$\begin{aligned} a_j &= \{a_j = x_{j_s}\} = \{\omega_h \in \Omega : a_j(\omega_h) = x_{j_s}\} \mid x_{j_s} \in X_j\} \\ &= \{\chi_{\{a_j = x_{j_s}\}} : \Omega \rightarrow \{0, 1\} \mid x_{j_s} \in X_j\}, \\ \lambda &= \{\lambda = c_q\} = \{\omega_h \in \Omega : \lambda(\omega_h) = c_q\} \mid c_q \in C\} \\ &= \{\chi_{\{\lambda = c_q\}} : \Omega \rightarrow \{0, 1\} \mid c_q \in C\}, \end{aligned}$$

thus, they induce a (crisp) total preorder on Ω due to (X_j, \leq) and (C, \leq) :

$$R_{a_j}(\omega_i, \omega_h) = \begin{cases} 1 & \text{if } a_j(\omega_i) \leq a_j(\omega_h), \\ 0 & \text{otherwise.} \end{cases} \quad R_\lambda(\omega_i, \omega_h) = \begin{cases} 1 & \text{if } \lambda(\omega_i) \leq \lambda(\omega_h), \\ 0 & \text{otherwise.} \end{cases}$$

The **dominant sets** of ω_i generated by a_j or λ have characteristic functions

$$\chi_{[\omega_i]_{a_j}^{\leq}}(\cdot) = R_{a_j}(\omega_i, \cdot) \quad \chi_{[\omega_i]_{\lambda}^{\leq}}(\cdot) = R_{\lambda}(\omega_i, \cdot).$$

Therefore, $\mathcal{A} \cup \{\lambda\}$ can be regarded as a collection of (crisp) partitions. Moreover, R_{a_j} and R_{λ} form the preference structures $(P_{a_j}, I_{a_j}, R_{a_j})$ and $(P_{\lambda}, I_{\lambda}, R_{\lambda})$, where P_{a_j}, P_{λ} are strict preference relations, I_{a_j}, I_{λ} are indifference relations, and R_{a_j}, R_{λ} are weak preference relations.

In this paper, for the fuzzy case, we assume that both a_j and λ are fuzzy partitions (in the Ruspini's sense [20]) of Ω :

$$a_j = \left\{ \mu_{\{a_j=x_{j_s}\}} : \Omega \rightarrow [0, 1] \mid x_{j_s} \in X_j, \sum_{x_{j_s} \in X_j} \mu_{\{a_j=x_{j_s}\}}(\omega_i) = 1, \omega_i \in \Omega \right\},$$

$$\lambda = \left\{ \mu_{\{\lambda=c_q\}} : \Omega \rightarrow [0, 1] \mid c_q \in C, \sum_{c_q \in C} \mu_{\{\lambda=c_q\}}(\omega_i) = 1, \omega_i \in \Omega \right\}.$$

Therefore, in the fuzzy case, $\mathcal{A} \cup \{\lambda\}$ can be regarded as a collection of Ruspini's fuzzy partitions.

Example 1. Let $\Omega = \{\omega_1, \omega_2, \omega_3\}$ be three cars evaluated according to the fuzzy attributes and labeling function below, where orders express preferences:

- $a_1 = \text{comfort}$ with $X_1 = \{\text{low}, \text{medium}, \text{high}\}$ ordered as $\text{low} < \text{medium} < \text{high}$,
- $a_2 = \text{price}$ with $X_2 = \{\text{cheap}, \text{expensive}\}$ ordered as $\text{expensive} < \text{cheap}$,
- $\lambda = \text{appreciation}$ with $C = \{\text{low}, \text{high}\}$ ordered as $\text{low} < \text{high}$.

	comfort			price		appreciation	
Ω	low	medium	high	cheap	expensive	low	high
ω_1	0.3	0.1	0.6	0.6	0.4	0.7	0.3
ω_2	0.8	0.1	0.1	0.3	0.7	0.1	0.9
ω_3	0.2	0.2	0.6	0.7	0.3	0.2	0.8

◆

The totally ordered sets (X_j, \leq) and (C, \leq) induce a total order on the fuzzy labels $\{a_j = x_{j_s}\}$ and $\{\lambda = c_q\}$ that we wish to “transport” somehow to Ω : the best would be to obtain a fuzzy total T -preorder on Ω , where T is a t -norm [10]. In other terms, we search for a fuzzy counterpart of the relations R_{a_j} and R_{λ} defined in the non-fuzzy (crisp) case. At this aim we recall the definition of fuzzy total T -preorder given in [6].

Definition 1. A function $R : \Omega \times \Omega \rightarrow [0, 1]$ is a **fuzzy total T -preorder** for a t -norm T if it satisfies:

- (1) **(strong completeness)** $\max\{R(\omega_i, \omega_h), R(\omega_h, \omega_i)\} = 1$, for all $\omega_i, \omega_h \in \Omega$;

(2) (**T -transitivity**) $R(\omega_i, \omega_h) \geq T(R(\omega_i, \omega_l), R(\omega_l, \omega_h))$, for all $\omega_i, \omega_l, \omega_h \in \Omega$.

A fuzzy preference structure is generally a weaker notion than a fuzzy total T -preorder. To have “common properties” analogous to the crisp case, a φ -transformation of the Łukasiewicz t -norm T_L (whose dual t -conorm is S_L) must be used. Below we report the definition of additive fuzzy preference structure with no incomparability [5] (see also [4, 21]), where \cap_L and \cup_L refer to T_L and S_L , the superscript t denotes the transpose relation and co the complement.

Definition 2. A triple (P, I, R) of functions on Ω ranging in $[0, 1]$ are an **additive fuzzy preference structure with no incomparability** if:

- (1) P is irreflexive and I is reflexive;
- (2) P is T_L -asymmetric and I is symmetric;
- (3) $P \cap_L I = \emptyset$;
- (4) $co(P \cup_L I) = P^t$;
- (5) $R = P \cup_L I$, i.e., $R(\omega_i, \omega_h) = S_L(P(\omega_i, \omega_h), I(\omega_i, \omega_h)) = P(\omega_i, \omega_h) + I(\omega_i, \omega_h)$.

Inspired to fuzzy preference structures built in the comparison of independent random variables (see [13–15]), we can provide the following probabilistic interpretation of our setup. For a fixed $a_j \in \mathcal{A}$, for each $\omega_i \in \Omega$, we set

$$p_{i,s}^j = \mu_{\{a_j = x_{j_s}\}}(\omega_i), \quad s = 1, \dots, t_j$$

then we have

$$\frac{a_j(\omega_i) \begin{vmatrix} x_{j_1} & x_{j_2} & \dots & x_{j_{t_j}} \\ p_{i,1}^j & p_{i,2}^j & \dots & p_{i,t_j}^j \end{vmatrix}}{p_{i,s}^j} \begin{cases} p_{i,s}^j \geq 0, & s = 1, \dots, t_j, \\ \sum_{s=1}^{t_j} p_{i,s}^j = 1. \end{cases}$$

The evaluation $a_j(\omega_i)$ can be interpreted as a discrete random variable with assigned probability distribution.

Assumption 1 Since objects in Ω are assumed not to influence each other, then $\{a_j(\omega_1), \dots, a_j(\omega_n)\}$ can be considered as stochastically independent random variables.

The above probabilistic interpretation allows us to refer to a fuzzy stochastic preference [5]:

Fuzzy strict stochastic preference relation:

$$\tilde{P}_{a_j}(\omega_i, \omega_h) = \max\{\text{Prob}(a_j(\omega_i) < a_j(\omega_h)) - \text{Prob}(a_j(\omega_i) > a_j(\omega_h)), 0\},$$

Fuzzy stochastic indifference relation:

$$\tilde{I}_{a_j}(\omega_i, \omega_h) = 1 - |\text{Prob}(a_j(\omega_i) < a_j(\omega_h)) - \text{Prob}(a_j(\omega_i) > a_j(\omega_h))|,$$

Fuzzy weak stochastic preference relation:

$$\tilde{R}_{a_j}(\omega_i, \omega_h) = S_L(\tilde{P}_{a_j}(\omega_i, \omega_h), \tilde{I}_{a_j}(\omega_i, \omega_h)) = \tilde{P}_{a_j}(\omega_i, \omega_h) + \tilde{I}_{a_j}(\omega_i, \omega_h),$$

where

$$\text{Prob}(a_j(\omega_i) < a_j(\omega_h)) = \begin{cases} \sum_{s < q} p_{i,s}^j p_{h,q}^j & i \neq h, \\ 0 & i = h, \end{cases}$$

$$\text{Prob}(a_j(\omega_i) > a_j(\omega_h)) = \begin{cases} \sum_{s > q} p_{i,s}^j p_{h,q}^j & i \neq h, \\ 0 & i = h. \end{cases}$$

We now consider the properties of $(\tilde{P}_{a_j}, \tilde{I}_{a_j}, \tilde{R}_{a_j})$. The triple $(\tilde{P}_{a_j}, \tilde{I}_{a_j}, \tilde{R}_{a_j})$ is an **additive fuzzy preference structure with no incomparability** satisfying the following properties

- (**generalization**) if a_j is a crisp partition then $\tilde{R}_{a_j} = R_{a_j}$,
- (**reflexivity**) $\tilde{R}_{a_j}(\omega_i, \omega_i) = 1$, for all $\omega_i \in \Omega$,
- (**strong completeness**) $\max\{\tilde{R}_{a_j}(\omega_i, \omega_h), \tilde{R}_{a_j}(\omega_h, \omega_i)\} = 1$ for all $\omega_i, \omega_h \in \Omega$.

Therefore, we can define the **fuzzy dominant set generated by a_j** as

$$\mu_{[\omega_i]_{\tilde{a}_j}}(\cdot) = \tilde{R}_{a_j}(\omega_i, \cdot). \quad (2)$$

A natural question concerns the T -transitivity of \tilde{R}_{a_j} . The following example shows that generally \tilde{R}_{a_j} is not guaranteed to be T -transitive for a t -norm T , even though it may be the case.

Example 2. Take $\mathcal{A} = \{a_1, a_2\}$ with $X_1 = X_2 = \{1, 2, 3\}$ with the natural order of numbers, and $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$.

	$\{a_1 = 1\}$	$\{a_1 = 2\}$	$\{a_1 = 3\}$	\tilde{R}_{a_1}	ω_1	ω_2	ω_3	ω_4
ω_1	0.3	0.5	0.2	ω_1	1	1	1	0.61
ω_2	0.2	0.4	0.4	ω_2	0.78	1	0.82	0.46
ω_3	0.1	0.8	0.1	ω_3	0.91	1	1	0.46
ω_4	0.7	0.2	0.1	ω_4	1	1	1	1

It follows that \tilde{R}_{a_1} is not T_L -transitive and so it is not T -transitive for any Frank t -norm T (see [10]):

$$\begin{aligned} 0.46 &= \tilde{R}_{a_1}(\omega_3, \omega_4) < T_L(\tilde{R}_{a_1}(\omega_3, \omega_1), \tilde{R}_{a_1}(\omega_1, \omega_4)) \\ &= \max\{0.91 + 0.61 - 1, 0\} = 0.52. \end{aligned}$$

	$\{a_2 = 1\}$	$\{a_2 = 2\}$	$\{a_2 = 3\}$	\tilde{R}_{a_2}	ω_1	ω_2	ω_3	ω_4
ω_1	0.3	0.3	0.4	ω_1	1	1	0.87	0.87
ω_2	0.3	0.3	0.4	ω_2	1	1	0.87	0.87
ω_3	0.4	0.3	0.3	ω_3	1	1	1	1
ω_4	0.4	0.3	0.3	ω_4	1	1	1	1

On the other hand, we get that \tilde{R}_{a_2} is T_M -transitive and so it is T -transitive for any t -norm T . \blacklozenge

Analogously, we can define the **fuzzy dominant set generated by λ** as

$$\mu_{\widetilde{[\omega_i]_{\lambda}^{\leq}}}(\cdot) = \tilde{R}_{\lambda}(\omega_i, \cdot). \quad (3)$$

The previous discussion allows us to fuzzify the rank discrimination measures H_S^* and H_G^* introduced in [12] (see also [8, 9]). To this purpose:

- we fix the Łukasiewicz De Morgan triple $(T_L, S_L, 1 - x)$ for uniformity, to compute fuzzy set-theoretic operations;
- we use the sigma-count to compute fuzzy cardinalities.

Definition 3. Given a_j and λ we define

Fuzzy rank Shannon discrimination measure:

$$\tilde{H}_S^*(\lambda|a_j) = \sum_{i=1}^{|\Omega|} \frac{1}{|\Omega|} \left[-\log_2 \left(\frac{|\widetilde{[\omega_i]_{\lambda}^{\leq}} \cap_L \widetilde{[\omega_i]_{a_j}^{\leq}}|}{|\widetilde{[\omega_i]_{a_j}^{\leq}}|} \right) \right],$$

Fuzzy rank Gini discrimination measure:

$$\tilde{H}_G^*(\lambda|a_j) = \sum_{i=1}^{|\Omega|} \frac{1}{|\Omega|} \left[1 - \left(\frac{|\widetilde{[\omega_i]_{\lambda}^{\leq}} \cap_L \widetilde{[\omega_i]_{a_j}^{\leq}}|}{|\widetilde{[\omega_i]_{a_j}^{\leq}}|} \right) \right].$$

The following example shows the computation of measures \tilde{H}_S^* and \tilde{H}_G^* .

Example 3. Let $\mathcal{A} = \{a_1\}$ with $X_1 = \{1, 2, 3\}$ and $C = \{1, 2\}$ with the usual order of numbers and consider the following evaluations

	$\{a_1 = 1\}$	$\{a_1 = 2\}$	$\{a_1 = 3\}$	$\{\lambda = 1\}$	$\{\lambda = 2\}$
ω_1	0.7	0.3	0	0.5	0.5
ω_2	0.5	0.1	0.4	0.2	0.8
ω_3	0.8	0.1	0.1	0.1	0.9
ω_4	0.6	0.3	0.1	1	0

\tilde{R}_{a_1}	ω_1	ω_2	ω_3	ω_4
ω_1	1	1	0.93	1
ω_2	0.68	1	0.67	0.79
ω_3	1	1	1	1
ω_4	0.87	1	0.82	1

\tilde{R}_{λ}	ω_1	ω_2	ω_3	ω_4
ω_1	1	1	1	0.5
ω_2	0.7	1	1	0.2
ω_3	0.6	0.9	1	0.1
ω_4	1	1	1	1

Both \tilde{R}_{a_1} and \tilde{R}_{λ} are T_L -transitive and it holds that

$$\tilde{H}_S^*(\lambda|a_1) = 0.3582 \quad \tilde{H}_G^*(\lambda|a_1) = 0.2060. \quad \blacklozenge$$

3 Enforcing monotonicity in decision tree construction

In order to evaluate the introduced fuzzy rank discrimination measures, we propose the following algorithm for building a fuzzy decision tree, in which monotonicity between attributes and class labels is enforced in a greedy way. The algorithm is parameterized by the choice of $\tilde{H}^* \in \{\tilde{H}_S^*, \tilde{H}_G^*\}$. Since we deal with a recursive algorithm, we keep notation simple by referring to Ω and \mathcal{A} as those available at the current stage of recursion.

Starting from the original Ω and \mathcal{A} , that are assumed not to be empty, the algorithm proceeds recursively, until a leaf is created with a label in C . If we are in a stage of the recursion with set of objects Ω and set of attributes \mathcal{A} , we first check if a leaf can be created. The creation of a leaf is justified when a sufficient degree of uniformity on the class label is observed in the current Ω . To choose the label in C , we compute the local threshold for class labels

$$\alpha^C = \max_{c_q \in C} \sum_{\omega \in \Omega} \frac{1}{|\Omega|} \mu_{\{\lambda=c_q\}}(\omega), \quad (4)$$

which corresponds to the highest average membership value to a class label, of objects in the current Ω .

Next, for each $c_q \in C$, we compute the percentage of objects in Ω whose membership is greater than or equal to α^C as

$$f^{c_q} = \frac{|\{\omega \in \Omega : \mu_{\{\lambda=c_q\}}(\omega) \geq \alpha^C\}|}{|\Omega|}. \quad (5)$$

We avoid over-fitting by creating a leaf in case there is at least one class label c_q such that $f^{c_q} \geq \rho$, where ρ is a fixed hyper-parameter chosen from the beginning of the procedure. We choose the class label c_q with maximum percentage f^{c_q} . Possible ties are broken by choosing the greatest class label, according to the total order of C . If no leaf is created in the current stage, then we need to split the current Ω by choosing an element of the current \mathcal{A} . For that, we proceed by computing $\tilde{H}^*(\lambda|a)$, for all $a \in \mathcal{A}$, and by solving

$$a^* = \arg \min_{a \in \mathcal{A}} \tilde{H}^*(\lambda|a), \quad (6)$$

where ties are broken choosing randomly. Once the splitting attribute a^* has been chosen, a branch is created for every element of the corresponding set of labels X^* . Moreover, the following splitting threshold is computed

$$\alpha^{a^*} = \max_{x \in X^*} \sum_{\omega \in \Omega} \frac{1}{|\Omega|} \mu_{\{a^*=x\}}(\omega), \quad (7)$$

which, again, corresponds to the highest average membership value to an attribute label, of objects in the current Ω .

Next, for every $x \in X^*$, we form the set

$$\Omega^x = \{\omega \in \Omega : \mu_{\{a^*=x\}}(\omega) \geq \alpha^{a^*}\}, \quad (8)$$

and repeat the procedure recursively on Ω^x and $\mathcal{A} \setminus \{a^*\}$. We point out that for different $x, x' \in X^*$ it may happen $\Omega^x \cap \Omega^{x'} \neq \emptyset$.

We notice that the overlapping of splitting sets can affect labeling when $|\mathcal{A}| = 1$ and the creation of a leaf at current stage is not optimal. Indeed, in this case the only choice for the splitting attribute is the unique element a^* of \mathcal{A} , so, the computation of (6) can be skipped. Therefore, a leaf is directly created for each value of X^* and labeling is carried out, again maximizing the f^{c_q} 's. Nevertheless, this could lead to some non-monotonicities due to objects appearing in more than one splitting set. Hence, once the labeling of leaves is over, a possible relabeling is applied to enforce monotonicity in the generated leaves, by changing those leaves with lower value of f^{c_q} first. If the generated sub-tree has leaves with all equal labels, then it is replaced by a single leaf with the same label.

Algorithm 1 reports the pseudo-code of the procedure described above.

Algorithm 1 Construction of a fuzzy decision tree enforcing monotonicity

▷ *input*: $\mathcal{A}, \lambda, \Omega$, data set
 ▷ *input*: ρ , over-fitting hyper-parameter
 ▷ *output*: \mathcal{T} , tree of fuzzy labels

Compute the threshold α^C as in (4)
 Compute f^{c_q} as in (5), for all $c_q \in C$
if there is $c_q \in C$ such that $f^{c_q} \geq \rho$ **then**
 Create a leaf in \mathcal{T} choosing $c_q \in C$ with maximum f^{c_q} , possibly breaking ties
else if $|\mathcal{A}| = 1$ **then**
 for x in X^* **do**
 Create a leaf in \mathcal{T} for the branch x
 Compute Ω^x as in (8)
 Choose $c_q \in C$ with maximum f^{c_q} in Ω^x , possibly breaking ties
 end for
 if there are non-monotone leaves **then** relabel those with lower f^{c_q} first
 if all leaves have the same label **then** replace the sub-tree with a single leaf
else
 Determine the splitting attribute a^* as in (6), possibly breaking ties
 for x in X^* **do**
 Compute Ω^x as in (8)
 Call **Algorithm 1** on $\mathcal{A} \setminus \{a^*\}, \lambda, \Omega^x$, and ρ
 end for
end if

We test Algorithm 1 by considering an artificial data set described by attributes in the set $\mathcal{A} = \{a_1, a_2, a_3\}$ and a labeling function λ . We assume that the a_j 's and λ range in the interval $[0, 10]$ and each is fuzzyfied using the set of ordered labels $X_j = C = \{\text{low}, \text{medium}, \text{high}\}$ that correspond to the fuzzy partition (in Ruspini's sense) reported in Figure 1.

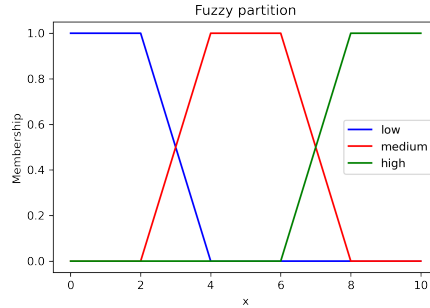


Fig. 1. Fuzzy partition {low, medium, high}.

We derive the description of $n = 1000$ objects by generating three independent random variables $A_j \sim \mathbf{Unif}([0, 10])$ and setting $\lambda = \max \left\{ A_1, \frac{A_2^2}{10}, \frac{A_3^3}{100} \right\}$, so as to range in $[0, 10]$ as well. Next, the realizations of each A_j and λ are fuzzyfied according to the fuzzy partition reported in Figure 1, so each object $\omega_i \in \Omega$ gives rise to a collection of probability distributions on the label set $X_j = C$. Table 1 shows the fuzzy partitions (2 decimal rounding) of Ω for the first 3 objects. We further fix the over-fitting parameter to $\rho = 70\%$.

Ω	a_1			a_2			a_3			λ		
	low	medium	high	low	medium	high	low	medium	high	low	medium	high
ω_1	0	0.82	0.18	0.65	0.35	0	1	0	0	0	0.82	0.18
ω_2	1	0	0	0	0	1	0	0	1	0	0.69	0.31
ω_3	0	0.97	0.03	0	0.35	0.65	0	1	0	0	0.97	0.03
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots

Table 1. Fuzzy partitions for the first three objects.

Below, we show the explicit execution of Algorithm 1 on the generated data set. To keep track of the evolution in the recursion, we add a subscript index starting at 0 to all quantities, related to the current level in the tree.

Therefore, we initially set $\Omega_0 := \Omega$ and $\mathcal{A}_0 := \mathcal{A}$, where $|\Omega_0| = 1000$. The terminal condition is not met since $f_0^{\text{low}} = 0.1600$, $f_0^{\text{medium}} = 0.4660$, $f_0^{\text{high}} = 0.4310$. Moreover, being $|\mathcal{A}_0| > 1$, since $H_S^*(\lambda|a_1) = 0.2475$, $H_S^*(\lambda|a_2) = 0.4547$, $H_S^*(\lambda|a_3) = 0.6274$, and $H_G^*(\lambda|a_1) = 0.1427$, $H_G^*(\lambda|a_2) = 0.2419$, $H_G^*(\lambda|a_3) = 0.3131$, both measures agree in selecting a_1 for splitting. Now, the set of objects Ω_0 is split in three subsets corresponding to labels in X_1 . The sets Ω_1^{low} , Ω_1^{medium} and Ω_1^{high} are not disjoint since $|\Omega_1^{\text{low}} \cap \Omega_1^{\text{medium}}| = 42$, $|\Omega_1^{\text{low}} \cap \Omega_1^{\text{high}}| = 0$, $|\Omega_1^{\text{medium}} \cap \Omega_1^{\text{high}}| = 42$.

We focus on Ω_1^{low} with $\mathcal{A}_1^{\text{low}} := \{a_2, a_3\}$, where $|\Omega_1^{\text{low}}| = 322$. The terminal condition is not met since $f_1^{\text{low}} = 0.4814$, $f_1^{\text{medium}} = 0.4006$, $f_1^{\text{high}} = 0.1615$.

Moreover, being $|\mathcal{A}_1^{\text{low}}| > 1$, since $H_S^*(\lambda|a_2) = 0.3001$, $H_S^*(\lambda|a_3) = 0.7780$, and $H_G^*(\lambda|a_2) = 0.1691$, $H_G^*(\lambda|a_3) = 0.3194$, both measures agree in selecting a_2 for splitting. Now, the set of objects Ω_1^{low} is split in three subsets corresponding to labels in X_2 . The sets $\Omega_2^{\text{low,low}}$, $\Omega_2^{\text{low,medium}}$ and $\Omega_2^{\text{low,high}}$ are not disjoint since $|\Omega_2^{\text{low,low}} \cap \Omega_2^{\text{low,medium}}| = 10$, $|\Omega_2^{\text{low,low}} \cap \Omega_2^{\text{low,high}}| = 0$, $|\Omega_2^{\text{low,medium}} \cap \Omega_2^{\text{low,high}}| = 12$.

We focus on Ω_1^{medium} with $\mathcal{A}_1^{\text{medium}} := \{a_2, a_3\}$, where $|\Omega_1^{\text{medium}}| = 440$. The terminal condition is not met since $f_1^{\text{low}} = 0$, $f_1^{\text{medium}} = 0.5955$, $f_1^{\text{high}} = 0.1227$. Moreover, being $|\mathcal{A}_1^{\text{medium}}| > 1$, since $H_S^*(\lambda|a_2) = 0.2505$, $H_S^*(\lambda|a_3) = 0.4605$, and $H_G^*(\lambda|a_2) = 0.1432$, $H_G^*(\lambda|a_3) = 0.2092$, both measures agree in selecting a_2 for splitting. Now, the set of objects Ω_1^{medium} is split in three subsets corresponding to labels in X_2 . The sets $\Omega_2^{\text{medium,low}}$, $\Omega_2^{\text{medium,medium}}$ and $\Omega_2^{\text{medium,high}}$ are not disjoint as $|\Omega_2^{\text{medium,low}} \cap \Omega_2^{\text{medium,medium}}| = 14$, $|\Omega_2^{\text{medium,low}} \cap \Omega_2^{\text{medium,high}}| = 0$, $|\Omega_2^{\text{medium,medium}} \cap \Omega_2^{\text{medium,high}}| = 18$.

We focus on Ω_1^{high} with $\mathcal{A}_1^{\text{high}} := \{a_2, a_3\}$, where $|\Omega_1^{\text{high}}| = 322$. The terminal condition is met since $f_1^{\text{low}} = 0$, $f_1^{\text{medium}} = 0$, $f_1^{\text{high}} = 0.7546$. Thus, since $f_1^{\text{high}} \geq \rho$, a leaf with label $\lambda = \text{high}$ is created.

We focus on $\Omega_2^{\text{low,low}}$ with $\mathcal{A}_2^{\text{low,low}} := \{a_3\}$, where $|\Omega_2^{\text{low,low}}| = 93$. The terminal condition is met since $f_2^{\text{low}} = 0.7097$, $f_2^{\text{medium}} = 0$, $f_2^{\text{high}} = 0$. Thus, since $f_2^{\text{low}} \geq \rho$, a leaf with label $\lambda = \text{low}$ is created.

We focus on $\Omega_2^{\text{low,medium}}$ with $\mathcal{A}_2^{\text{low,medium}} := \{a_3\}$, where $|\Omega_2^{\text{low,medium}}| = 140$. The terminal condition is not met since $f_2^{\text{low}} = 0.5071$, $f_2^{\text{medium}} = 0.4285$, $f_2^{\text{high}} = 0$. The splitting is made on a_3 and the sets $\Omega_3^{\text{low,medium,low}}$, $\Omega_3^{\text{low,medium,medium}}$, $\Omega_3^{\text{low,medium,high}}$ give rise to three leaves labelled, respectively, as $\lambda = \text{medium}$, $\lambda = \text{low}$, $\lambda = \text{medium}$. Thus, we relabel leaves by setting, respectively, $\lambda = \text{low}$, $\lambda = \text{low}$, $\lambda = \text{medium}$: the first label is indeed that with lower value of f^{c_q} .

We focus on $\Omega_2^{\text{low,high}}$ with $\mathcal{A}_2^{\text{low,high}} := \{a_3\}$, where $|\Omega_2^{\text{low,high}}| = 111$. The terminal condition is not met since $f_2^{\text{low}} = 0$, $f_2^{\text{medium}} = 0.5405$, $f_2^{\text{high}} = 0.4595$. The splitting is made on a_3 and the sets $\Omega_3^{\text{low,high,low}}$, $\Omega_3^{\text{low,high,medium}}$, $\Omega_3^{\text{low,high,high}}$ give rise to three leaves labelled, respectively, as $\lambda = \text{high}$, $\lambda = \text{medium}$, $\lambda = \text{medium}$. Thus, we relabel leaves by setting, respectively, $\lambda = \text{medium}$, $\lambda = \text{medium}$, $\lambda = \text{medium}$: the first label is indeed that with lower value of f^{c_q} . Therefore, we replace the built sub-tree with a single leaf labelled as $\lambda = \text{medium}$.

We focus on $\Omega_2^{\text{medium,low}}$ with $\mathcal{A}_2^{\text{medium,low}} := \{a_3\}$, where $|\Omega_2^{\text{medium,low}}| = 138$. The terminal condition is not met since $f_2^{\text{low}} = 0$, $f_2^{\text{medium}} = 0.6087$, $f_2^{\text{high}} = 0$. The splitting is made on a_3 and the sets $\Omega_3^{\text{medium,low,low}}$, $\Omega_3^{\text{medium,low,medium}}$, $\Omega_3^{\text{medium,low,high}}$ give rise to three leaves all labeled as $\lambda = \text{medium}$. Therefore, we replace the built sub-tree with a single leaf labelled as $\lambda = \text{medium}$.

We focus on $\Omega_2^{\text{medium,medium}}$ with $\mathcal{A}_2^{\text{medium,medium}} := \{a_3\}$, where $|\Omega_2^{\text{medium,medium}}| = 186$. The terminal condition is not met since $f_2^{\text{low}} = 0$, $f_2^{\text{medium}} = 0.6290$, $f_2^{\text{high}} = 0$. The splitting is made on a_3 and the sets $\Omega_3^{\text{medium,medium,low}}$, $\Omega_3^{\text{medium,medium,medium}}$, $\Omega_3^{\text{medium,medium,high}}$ give rise to three leaves all labeled as $\lambda = \text{medium}$. Therefore, we replace the built sub-tree with a single leaf labelled as $\lambda = \text{medium}$.

We focus on $\Omega_2^{\text{medium,high}}$ with $\mathcal{A}_2^{\text{medium,high}} := \{a_3\}$, where $|\Omega_2^{\text{medium,high}}| = 148$. The terminal condition is not met since $f_2^{\text{low}} = 0$, $f_2^{\text{medium}} = 0.4595$, $f_2^{\text{high}} = 0.4865$. The splitting is made on a_3 and the sets $\Omega_3^{\text{medium,high,low}}$, $\Omega_3^{\text{medium,high,medium}}$,

$\Omega_3^{\text{medium,high,high}}$ give rise to three leaves labeled, respectively, as $\lambda = \text{medium}$, $\lambda = \text{high}$, $\lambda = \text{high}$.

Figure 2 shows the monotone fuzzy decision tree \mathcal{T} obtained by applying Algorithm 1, where **low**, **medium**, and **high**, are abbreviated as **l**, **m**, and **h**, respectively. A direct inspection shows that the λ' encoded in \mathcal{T} satisfies (1).

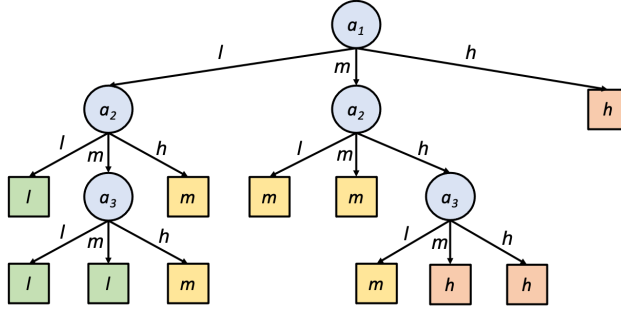


Fig. 2. Monotone fuzzy decision tree.

4 Conclusion

In this paper we consider the problem of building a fuzzy decision tree by enforcing monotonicity of the class label with respect to attributes labels, both assumed to range in totally ordered sets of labels. We propose two fuzzy versions of rank discrimination measures that generalize those proposed in [12], together with an associated construction algorithm. Due to space limitations we provided an experimental analysis on an artificial data set. A first line of future research consists in developing a hierarchical construction model of a general fuzzy rank discrimination measure in analogy with [12], and a systematic analysis of their analytical properties. Finally, we also plan to perform a deeper experimental analysis on real data: at this aim we point out the necessity of a suitable fuzzy non-monotonicity index, obtained, for instance, generalizing that in [16].

Acknowledgements Davide Petturiti is member of the GNAMPA-INdAM research group. This research was carried out during Davide Petturiti’s research stay at LIP6, Sorbonne Université.

References

1. Ben-David, A., Sterling, L., Pao, Y.: Learning and classification of monotonic ordinal concepts. *Computational Intell.* **5**(1), 45–49 (1989)

2. Cano, J.R., García, S.: Training set selection for monotonic ordinal classification. *Data & Knowledge Engin.* **112**, 94–105 (2017)
3. Cano, J.R., Gutiérrez, P., Krawczyk, B., Woźniak, M., García, S.: Monotonic classification: An overview on algorithms, performance measures and data sets. *Neurocomputing* **341**, 168–182 (2019)
4. De Baets, B., Fodor, J.: Principles of Fuzzy Preference Modelling and Decision Making, chap. Additive fuzzy preference structures: the next generation, pp. 15–25. Academia Press (2003)
5. Díaz, S., Montes, S., De Baets, B.: Transitivity bounds in additive fuzzy preference structures. *IEEE Trans. Fuzzy Systems* **15**(2), 275–286 (2007)
6. Fodor, J., Roubens, M.: Fuzzy Preference Modelling and Multicriteria Decision Support. Kluwer Academic Publishers, Dordrecht (1994)
7. González, S., Herrera, F., García, S.: Monotonic Random Forest with an Ensemble Pruning Mechanism based on the Degree of Monotonicity. *New Gener. Computing* **33**(4), 367–388 (2015)
8. Hu, Q., Che, X., Zhang, L., Zhang, D., Guo, M., Yu, D.: Rank entropy based decision trees for monotonic classification. *IEEE Trans. Knowledge and Data Engineering* **24**(11), 2052–2064 (2012)
9. Hu, Q., Guo, M., Yu, D., Liu, J.: Information entropy for ordinal classification. *Science China Inform. Sci.* **53**, 1188–1200 (2010)
10. Klement, E., Mesiar, R., Pap, E.: Triangular Norms, Trends in Logic, vol. 8. Kluwer Academic Publishers, Dordrecht/Boston/London (2000)
11. Marsala, C., Bouchon-Meunier, B.: An adaptable system to construct fuzzy decision trees. In: Proc. of the NAFIPS'99. pp. 223–227. New York, USA (1999)
12. Marsala, C., Petturiti, D.: Rank discrimination measures for enforcing monotonicity in decision tree induction. *Inform. Sci.* **291**, 143–171 (2015)
13. Martinetti, D., Montes, I., Díaz, S., Montes, S.: A study on the transitivity of probabilistic and fuzzy relations. *Fuzzy Sets Syst.* **184**(1), 156–170 (2011)
14. Martinetti, D., Montes, S., Díaz, S., De Baets, B.: On a correspondence between probabilistic and fuzzy choice functions. *Fuzzy Optim. Decision Making* **17**(3), 247–264 (2018)
15. Martinetti, D., Montes, S., Díaz, S., De Baets, B.: On the correspondence between reciprocal relations and strongly complete fuzzy relations. *Fuzzy Sets Syst.* **322**, 19–34 (2017)
16. Milstein, I., Ben-David, A., Potharst, R.: Generating noisy ordinal monotone datasets. *Artificial Intell. Research* **3**(1) (2014)
17. Olaru, C., Wehenkel, L.: A complete fuzzy decision tree technique. *Fuzzy Sets Syst.* **138**(2), 221–254 (2003)
18. Pei, S., Hu, Q.: Partially monotonic decision trees. *Inform. Sci.* **424**, 104–117 (2018)
19. Qian, Y., Xu, H., Liang, J., Liu, B., Wang, J.: Fusing monotonic decision trees. *IEEE Trans. Knowledge and Data Engineering* **27**(10), 2717–2728 (2015)
20. Ruspini, E.: A new approach to clustering. *Inform. and Control* **15**(1), 22–32 (1969)
21. Van de Walle, B., De Baets, B., Kerre, E.: Characterizable fuzzy preference structures. *Annals Oper. Research* **80**, 105–136 (1998)
22. Wang, J., Y. Qian, Y., Li, F., Liang, J., Ding, W.: Fusing fuzzy monotonic decision trees. *IEEE Trans. Fuzzy Systems* **28**(5), 887–900 (2020)
23. Wang, X., Zhai, J., Chen, J., Wang, X.: Ordinal decision trees based on fuzzy rank entropy. In: 2015 International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR). pp. 208–213 (2015)
24. Yuan, Y., Shaw, M.: Induction of fuzzy decision trees. *Fuzzy Sets Syst.* **69**, 125–139 (1995)