Decision Trees with a Modal Flavor

Dario Della Monica[1], Giovanni Pagliarini[2,3],
Guido Sciavicco[3], and Ionel Eduard Stan[2,3]

[1] University of Udine
dario.dellamonica@uniud.it
[2] University of Parma
[3] University of Ferrara
{giovanni.pagliarini,guido.sciavicco**,ioneleduard.stan}@unife.it

Abstract. Symbolic learning is the sub-field of machine learning that deals with symbolic algorithms and models, which have been known for decades and successfully applied to a variety of contexts, and of which decision trees are the quintessential expression. The main limitation of current symbolic models is the fact that they are essentially based on classical propositional logic, which implies that data with an implicit dimensional component, such as temporal, e.g., time series, or spatial data, e.g., images, cannot be properly dealt with within the standard symbolic framework. In this paper, we show how propositional logic in decision trees can be replaced with the more expressive (propositional) modal logics, and we lay down the formal bases of modal decision trees by first systematically delineating interesting and well-known properties of propositional ones and then showing how to transfer these properties to the modal case.

Keywords: Machine learning · Decision trees · Modal logic · Learning from dimensional data

1 Introduction

The most iconic and fundamental separation between sub-fields of machine learning is the one between *functional* and *symbolic* learning. Functional learning is the process of learning a *function* that represents the theory underlying a certain phenomenon, while symbolic learning is the process of learning a *logical description* that represents that phenomenon.

Whether one or the other approach should be preferred raised a long-standing debate among experts, which roots in the fact that functional methods tend to be more versatile and statistically accurate than symbolic ones, while symbolic methods are able to extract models that can be interpreted, explained, and then enhanced using human-expert knowledge. These characteristics of symbolic methods, both for *political* reasons (consider, for instance, the recent General Data Protection Regulation (GDPR) of the European Union [13], that highlights

** Corresponding author.

the need for interpretable/explainable automatic learning-based decision-making processes, including those involving AI technologies) and *technical* ones (interpretable models are often easier to train, explore, integrate, and implement), are sometimes used as arguments for preferring a symbolic approach over a functional one. From a logical standpoint, canonical symbolic learning methods are all characterized by the use of propositional logic (they are, indeed, sometimes called *propositional* methods), and, among them, propositional decision trees are probably the best known.

The origin of modern decision trees dates back to the fifties [2]; a lot of work has been done since then, which includes, among others, [4, 8, 10, 11, 15–17], and decision tree models extracted using popular algorithms such as ID3, C4.5, and more recent ones, have been widely applied in the literature. Different decision tree models differ in their structure and the language on which they are based, but only slightly; from a structural point of view, it can be argued that virtually all such structures and learning algorithms stemmed, in some sense, from CART [4], which already contained all the fundamental ideas of decision trees.

Dimensional data, such as temporal or spatial data, cannot be dealt with in a proper, native way using propositional decision trees. The general to-go strategy to treat dimensional data with propositional models, such as decision trees, is to *flatten* the dimensional component, effectively hiding it. Flattening consists in massaging the dataset in such a way that dimensional attributes become scalar ones. As an example, a multivariate time series with n temporal attributes $A_1, \ldots, A_n$ can be transformed by applying one or more feature extractions function to all attributes, e.g., average, minimum, maximum, and the like, to obtain (a feature representation of) an instance $f_1(A_1), f_2(A_1), \ldots, f_1(A_2), f_2(A_2), \ldots,$ which can now be treated, for example, by a standard decision tree. A more general approach consists of applying the same strategy to different *windows* along all dimensions, e.g., intervals in the temporal case, rectangles in the spatial one, and so on, obtaining several new attributes for each original one and each feature extraction function. At the limit, each temporal (spatial, ...) point may become a window. As an example, a single-variate time series A with N ordered points ends up being represented as the (unordered) collection $A(1), A(2), \ldots, A(N)$. Such a representation is called *lagged* (for temporal data) or *flattened* (for spatial ones).

In this paper, we adopt a different point of view, aiming at laying down the formal bases of *modal symbolic learning*, by means of which dimensional datasets can be dealt with in a native way. To this end, we replace propositional logic by *propositional modal logic* (*modal logic* for short) and we enhance decision trees accordingly. Modal logic [3] generalizes propositional logic by allowing one to natively express the relationships that emerge among the different *worlds*, e.g., time points, time intervals, multi-dimensional areas, that contribute to describe real-world scenarios. Since modal logic can be declined into more practical languages, such as temporal and spatial logics, and dimensional data can be seen as modal data, modal symbolic learning is immediately applicable to the dimensional case. Moreover, this is not the only possible application, as modal data emerge in a natural way also from non-dimensional data, like, for instance, in textual and graph-based data.

Here, we introduce *modal decision trees*, and we systematically study their logical properties, specifically, correctness. Standard decision trees are, indeed, correct, although the nature of their presentation, mostly driven by applications, tends to hide their theoretical aspects. While we are not interested in studying efficient implementations of learning algorithms, the driving principle of the definition of modal decision trees is the preservation of the simplicity and interpretability that characterize propositional ones. As a result, modal decision tree learning algorithms can be implemented starting from any implementation of propositional ones, and working one's way up.

The paper is organized as follows. In Section 2, we provide some preliminary definitions and concepts. In Section 3, we define modal decision trees and study their properties. Then, in Section 4, we briefly show how modal decision trees can be applied to learn from dimensional data, before concluding.

2 Preliminaries

Let $\mathcal{P}$ be a set of *propositional letters*. The well-formed formulas of *modal logic* (*ML*) are obtained from the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi.$$

The other usual Boolean connectives can be derived from them, and, as standard, we use $\Box\varphi$ to denote $\neg\Diamond\neg\varphi$. The *modality* $\Diamond$ (resp., $\Box$) is usually referred to as *it is possible that* (resp., *it is necessary that*). Modal logic is considered as archetypical of (propositional) temporal, spatial, and spatio-temporal logics, and it is a non-conservative extension of *propositional logic* (*PL*). Its semantics is given in terms of Kripke models. A *Kripke model* $K = (W, R, V)$ *over* $\mathcal{P}$ consists of a (finite) set of *worlds* W, which contains a distinguished world w_0, called *initial world*, a binary *accessibility relation* $R \subseteq W \times W$, and a *valuation function* $V : W \to 2^{\mathcal{P}}$, which associates each world with the set of proposition letters that are true on it. The *truth* relation $K, w \Vdash \varphi$ for a model K and a world w in it is expresed by the following clauses:

$$\begin{array}{ll}
K, w \Vdash p & \text{iff } p \in V(w); \\
K, w \Vdash \neg\varphi & \text{iff } K, w \nVdash \varphi; \\
K, w \Vdash \varphi \wedge \psi & \text{iff } K, w \Vdash \varphi \text{ and } K, w \Vdash \psi; \\
K, w \Vdash \Diamond\varphi & \text{iff } \exists v \text{ s.t. } wRv \text{ and } K, v \Vdash \varphi.
\end{array}$$

We write $K \Vdash \varphi$ as an abbreviation for $K, w_0 \Vdash \varphi$.

The importance of modal logic comes from the fact that most classic temporal [5, 7, 14] and spatial logics [1, 9] stem from (generalizations of) modal logic. Therefore, the theory of modal logic and the tools built on it can be reused to cope with more practical situations.

We now introduce the notion of modal dataset and its associated problems.

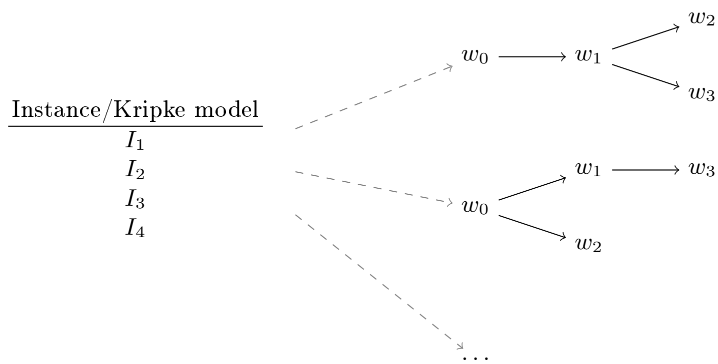


Fig. 1. An example of modal dataset with 4 instances, each described by a Kripke model.

Definition 1 (Modal dataset). *Let $\mathcal{P}$ be a set of proposition letters. A* modal dataset *$\mathcal{I} = \{I_1, \ldots, I_m\}$* over *$\mathcal{P}$ is a finite collection of m* instances*, each of which is a Kripke model over $\mathcal{P}$, and such that I, J are not bisimilar, for each $I, J \in \mathcal{I}$ with $I \neq J$, that is, there exists at least one formula $\varphi \in ML$ with $I \Vdash \varphi$ and $J \nVdash \varphi$. We say that $\mathcal{I}$ is* labeled *if it is equipped with a* labeling function *$L : \mathcal{I} \rightarrow \mathcal{C}$ which associates every instance with a* class *from a finite set $\mathcal{C} = \{C_1, \ldots, C_k\}$.*

In the static case, a dataset is usually defined as a collection $\mathcal{I} = \{I_1, \ldots, I_m\}$ of m instances described, each, by the value of n distinct attributes $\mathcal{A} = \{A_1, \ldots, A_m\}$. However, since each attribute A is associated to its finite domain $dom(A)$, that is, the finite set of all values taken by A across $\mathcal{I}$, the latter naturally induces a set of propositional letters:

$$\mathcal{P} = \{A \bowtie a \mid \bowtie \in \{<, \leq, =, \geq, >\}, A \in \mathcal{A}, a \in dom(A)\}.$$

Learning-wise, therefore, we can always define a static dataset as if the corresponding set of propositional letters is fixed.

A modal dataset immediately generalizes a static one, by postulating that instances are described by Kripke frames in which attributes change value across different worlds. There are several scenarios that can be naturally modeled by modal, non-static datasets, instead; by way of example, *dimensional* datasets are characterized by each attribute in each instance being described by a d-dimensional matrix (e.g., $d = 1$ in the temporal case, and $d = 2$ in the spatial case). In such cases, fixed a set of *feature extraction function(s)* $\mathcal{F} = \{f_1, \ldots, f_k\}$, the set of induced propositional letters becomes:

$$\mathcal{P} = \{f(A) \bowtie a \mid \bowtie \in \{<, \leq, =, \geq, >\}, A \in \mathcal{A}, a \in dom(A), f \in \mathcal{F}\}.$$

Dimensional datasets are not the only source of modal datasets; in fact, our definition of modal dataset is more general, and captures a wide range of practical situations.

In the static case two instances cannot be identical, that is, there must be a propositional formula that distinguishes them; at the modal level, this requirement translates into constraining every two instances to be non-bisimilar (see, again, [3]), that is, to be distinguishable by at least one modal formula.

In machine learning, several problems are associated to a labeled dataset $\mathcal{I}$. Among them, a fundamental and ubiquitous one is the *classification* problem, that is, the problem of synthesizing an algorithm (a *classifier*) that is able to classify the instances of an unlabeled dataset $\mathcal{J}$ of the same type as $\mathcal{I}$.

In the symbolic context, learning a classifier from a dataset requires extracting from it the logical property that define each class, that is, its *characteristic* formula. Then, instances are seen as models of the considered logical formalism and the classification task is performed via model checking an instance against characteristic formulas. Although, in principle, one can be interested in learning characteristic formulas of any logic in any dataset, to modal (resp., propositional) datasets it is natural to associate modal (resp., propositional) characteristic formulas.

Binary decision trees, which are typical classifiers, are binary trees whose leaves and edges are equipped with labels. Leaf labels identify the different classes an instance can belong to; edge labels are atomic logical elements which are then composed to obtain complex formulas in the considered logical formalism (in the propositional case, edge labels edges are literals and formulas are Boolean combinations). A tree associates a formula to every class it features (i.e., every label occurring in a leaf) and it classifies an instance into a class if and only if the instance satisfies the formula corresponding to that class. As there can be exponentially many leaves in a tree, the classification process can possibly require verifying the satisfaction of an instance against exponentially many formulas.

However, decision trees provide an efficient mechanism for classifying an instance that does not explore the entire tree: for every node, starting from the root and going down towards the leaves, the truth of the formula associated with that node is checked against the instance to be classified and, depending on the outcome the instance is passed to the right or the left child and the process is repeated. When a leaf is reached, the instance is classified into the class that labels that leaf. Summing up, the desired properties for a family $\mathcal{M}$ of decision trees include: *(i) correctness* (every tree classifies any given instance into exactly one class); *(ii) completeness* (for every formula φ of the considered formalism, there is a decision tree $\tau \in \mathcal{M}$ that realizes φ); and *(iii) efficiency* (a decision tree τ of height h must be able to classify an instance I by checking the truth of, at most, a number of formulas polynomial in h).

In the rest of this paper we consider the problem of designing modal decision trees in such a way to be correct, complete, and efficient with respect to modal logic.

3 Modal Decision Trees

Let $\tau = (V, E)$ be a full directed finite binary tree with vertexes in V and edges in E. We denote by $root(\tau)$ the root of τ, by $V^{\ell} \subseteq V$ the set of its *leaves*, and by V^{ι} the set of its *internal nodes* (that is, non-root and non-leaf nodes). For each non-leaf node ν we denote by $\swarrow(\nu)$ (resp., $\searrow(\nu)$) its *left* (resp. *right*) *child*, and by $\wr(\nu)$ its *parent*. Similarly, for a tree τ, we denote by $\swarrow(\tau)$ (resp., $\searrow(\tau)$) its *left* (resp. *right*) *subtree*. Finally, for a node ν, the set of its *ancestors* (ν included) is denoted by $\wr^*(\nu)$, where $\wr^*$ is the transitive and reflexive closure of $\wr$; we also define $\wr^+(\nu) = \wr^*(\nu) \setminus \{\nu\}$.

A *path* $\pi^\tau = \nu_0 \rightsquigarrow \nu_h$ *in tree* τ (or, simply, π, if τ is clear form the context) of *length* $h \geq 0$ from ν_0 to ν_h is a finite sequence of $h+1$ nodes $\nu_0, \ldots, \nu_h$ such that $\nu_i = \wr(\nu_{i+1})$, for each $i = 0, \ldots, h-1$. We denote by $\pi_1 \cdot \pi_2$ the operation of appending the path π_2 to the path π_1. We also say that a path $\nu_0 \cdot \nu_1 \rightsquigarrow \nu_h$ is *left* (resp., *right*) if $\nu_1 = \swarrow(\nu_0)$ (resp., $\nu_1 = \searrow(\nu_0)$). For a path $\pi = \nu_0 \rightsquigarrow \nu_h$, the set of its *improper prefixes* is denoted by $prefix(\pi)$, and if ν is a node in τ, π^τ_ν (or, simply, π_ν, if τ is clear from the context) denotes the unique path $root(\tau) \rightsquigarrow \nu$. Finally, a *branch* of τ is a path π^τ_ℓ (or, simply, π_ℓ, if τ is clear from the context) for some $\ell \in V^\ell$.

Definition 2 (modal decisions). *Fixed a modal dataset $\mathcal{I}$ over $\mathcal{P}$, the set of* decisions *is:*

$$\Lambda = \{\top, \bot, p, \neg p, \Diamond\top, \Box\bot \mid p \in \mathcal{P}\}.$$

We say that $p, \neg p$ are *propositional decisions*, while $\Diamond\top$ (resp., $\Box\bot$) are *modal existential* (resp., *modal universal*) ones, and we use the symbol $\lambda \in \Lambda$ to denote a decision. For each $\lambda \in \Lambda$, the decision that corresponds to its logical negation $\neg\lambda$ is univocally identified, so when $\lambda = \top$ (resp., $p, \Diamond\top$) we use $\neg\lambda$ to denote $\bot$ (resp., $\neg p, \Box\bot$), and vice versa.

Definition 3 (modal decision tree). *Fixed a propositional alphabet $\mathcal{P}$ and a set of classes $\mathcal{C}$, a* modal decision tree *τ* over *$\mathcal{P}$* and *$\mathcal{C}$ is a structure:*

$$\tau = (V, E, b, l, s),$$

where (V, E) is a full directed finite binary tree, $l : V^\ell \to \mathcal{C}$ is the leaf-labeling function*, $b : V^\iota \to V^\iota$ is the* back-edge function*, $s : E \to \Lambda$ is the* edge-labeling function*, and the following conditions hold:*

1. $\forall \nu, \nu' \in V.(b(\nu) = \nu' \to \nu' \in \wr^*(\nu))$;
2. $\forall \nu, \nu' \in V.((b(\nu) \neq \nu \wedge b(\nu') \neq \nu') \to b(\nu) \neq b(\nu'))$;
3. $\forall \nu, \nu', \nu'' \in V.((b(\nu) = \nu' \wedge \nu' \in \wr^+(\nu'') \wedge \nu'' \in \wr^+(\nu)) \to \nu' \in \wr^+(b(\nu'')))$;
4. $\forall (\nu, \nu') \in E.((s(\nu, \nu') \in \{\bot, \Box\bot\} \wedge \nu' \notin V^\ell) \to b(\nu') \neq \nu')$;
5. $\forall (\nu, \nu'), (\nu, \nu'') \in E.(s(\nu, \nu') = \neg s(\nu, \nu''))$.

For every $c \in \mathcal{C}$, we denote by $leaves^\tau(c)$ (or, simply, $leaves(c)$, when τ is clear from the context) the set of leaves of τ labeled with c.

A propositional decision tree is a modal decision tree in which edges are labeled with propositional decisions and the back-edge function plays no role (therefore, in propositional decision trees only condition 5 is still non-trivial); thus, propositional decision trees are a particular case of modal decision trees. In the following, we denote by $\mathcal{MDT}$ the *family* of modal decision trees (or *modal decision tree classification model*), and by $\mathcal{DT}$ its propositional counterpart (that is, the sub-family of $\mathcal{MDT}$ that only contains propositional trees). From now on, we use the term decision tree to refer to an element of either $\mathcal{DT}$ or $\mathcal{MDT}$.

We now show how a modal decision tree defines a modal formula for each of its classes. This is obtained by associating a formula to each branch, and then the formula of a class is the disjunction of all the formulas associated to branches whose leaf is labeled with that class. In the propositional case, each branch is associated to the conjunction of the labels that occur on its edges; as every propositional formula can be written in disjunctive normal form, propositional decision trees are complete with respect to propositional logic. Modal logic does not have a normal form that allows one to bound the nesting of modal operators, and this makes the construction of formulas more complicated. Let us first fix the following useful concepts.

Definition 4 (contributor, node agreement). *Given a decision tree τ and a path $\pi = \nu_0 \rightsquigarrow \nu_h$, with $h > 1$, the* contributor of π, *denoted $ctr(\pi)$, is defined as the only node ν_i in π such that $\nu_i \neq \nu_1$, $0 < i < h$, and $b(\nu_i) = \nu_1$, if it exists, and as ν_1 otherwise. Moreover, given two nodes $\nu_i, \nu_j \in \pi$, with $i, j < h$, we say that they* agree *if $\nu_{i+1} = \searrow(\nu_i)$ (resp., $\nu_{i+1} = \swarrow(\nu_i)$) and $\nu_{j+1} = \searrow(\nu_j)$ (resp., $\nu_{j+1} = \swarrow(\nu_j)$), and we denote this situation by $A(\nu_i, \nu_j)$, and that they* disagree *(denoted by $D(\nu_i, \nu_j)$), otherwise.*

To our purposes, we use the following grammar to generate formulas of ML:

$$\varphi ::= \lambda \mid \lambda \wedge (\varphi \wedge \varphi) \mid \lambda \rightarrow (\varphi \rightarrow \varphi) \mid \Diamond(\varphi \wedge \varphi) \mid \Box(\varphi \rightarrow \varphi),$$

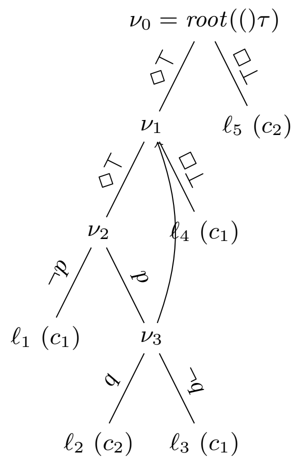
where $\lambda \in \Lambda$.

Definition 5 (implicative formulas). *We say that a modal formula φ is* implicative *if it has the form $\psi \rightarrow \xi$, or $\Box(\psi \rightarrow \xi)$, and we denote by Im the set of implicative formulas.*

As a matter of fact, in order to assign a formula to each leaf, and then to each class, we first associate a formula to every path (see Fig. 2 for an example).

Definition 6 (path-, leaf-, and class-formula). *Let τ be a decision tree. For each path $\pi = \nu_0 \rightsquigarrow \nu_h$ in τ, the* path-formula φ^τ_π *(or, simply, φ_π, when τ is clear from the context) is defined inductively as:*

- *If $h = 0$, then $\varphi_\pi = \top$.*
- *If $h = 1$, then $\varphi_\pi = s(\nu_0, \nu_1)$.*



$$
\begin{array}{lll}
\varphi_{\nu \rightsquigarrow \nu} & = \top, \forall \nu \in V^{\iota} & \\
\varphi_{\zeta(\nu) \rightsquigarrow \nu} & = s(\zeta(\nu), \nu), \forall \nu \in V \setminus \{\nu_0\} & \\
& & \\
\varphi_{\nu_1 \rightsquigarrow \ell_1} & = \Diamond(\top \wedge \neg p) & \\
\varphi_{\nu_0 \rightsquigarrow \nu_2} & = \Diamond(\top \rightarrow \Diamond\top) & \\
\varphi_{\nu_0 \rightsquigarrow \ell_1} & = \Diamond(\top \wedge \Diamond(\top \wedge \neg p)) & = \varphi_{\pi_{\ell_1}} \\
\varphi_{\nu_1 \rightsquigarrow \nu_3} & = \Box(\top \rightarrow p) & \\
\varphi_{\nu_0 \rightsquigarrow \nu_3} & = \Box(\top \rightarrow \Box(\top \rightarrow p)) & \\
\varphi_{\nu_0 \rightsquigarrow \ell_2} & = \Diamond(\Box(\top \rightarrow p) \wedge q) & = \varphi_{\pi_{\ell_2}} \\
\varphi_{\nu_0 \rightsquigarrow \ell_3} & = \Box(\Box(\top \rightarrow p) \rightarrow \neg q) & = \varphi_{\pi_{\ell_3}} \\
\varphi_{\nu_0 \rightsquigarrow \ell_4} & = \Box(\top \rightarrow \Box\bot) & = \varphi_{\pi_{\ell_4}} \\
& & \\
\varphi_{\ell} & = \bigwedge_{\pi \in prefix(\pi_\ell)} \varphi_\pi, \forall \ell \in V^{\ell} & \\
\varphi_{c_1} & = \varphi_{\ell_1} \vee \varphi_{\ell_3} \vee \varphi_{\ell_4} & \\
\varphi_{c_2} & = \varphi_{\ell_2} \vee \varphi_{\ell_5} &
\end{array}
$$

Fig. 2. On the left-hand side, an example of a modal decision tree; on the right-hand side, all relevant path-, leaf-, and class-formulas (φ_{ℓ_5} is included in the second group from the top).

– *If* $h > 1$, *let* $\lambda = s(\nu_0, \nu_1)$, $\pi_1 = \nu_1 \rightsquigarrow ctr(\pi)$, *and* $\pi_2 = ctr(\pi) \rightsquigarrow \nu_h$. *Then,*

$$
\varphi_\pi = \begin{cases}
\lambda \wedge (\varphi_{\pi_1} \wedge \varphi_{\pi_2}) & \textit{if} \;\; \lambda \neq \Diamond\top,\, A(\nu_0, ctr(\pi)),\, \textit{and}\; \varphi_{\pi_2} \notin Im, \\
 & \textit{or} \;\; \lambda \neq \Diamond\top,\, D(\nu_0, ctr(\pi)),\, \textit{and}\; \varphi_{\pi_2} \in Im; \\
\lambda \rightarrow (\varphi_{\pi_1} \rightarrow \varphi_{\pi_2}) & \textit{if} \;\; \lambda \neq \Diamond\top,\, D(\nu_0, ctr(\pi)),\, \textit{and}\; \varphi_{\pi_2} \notin Im, \\
 & \textit{or} \;\; \lambda \neq \Diamond\top,\, A(\nu_0, ctr(\pi)),\, \textit{and}\; \varphi_{\pi_2} \in Im; \\
\Diamond(\varphi_{\pi_1} \wedge \varphi_{\pi_2}) & \textit{if} \;\; \lambda = \Diamond\top,\, A(\nu_0, ctr(\pi)),\, \textit{and}\; \varphi_{\pi_2} \notin Im, \\
 & \textit{or} \;\; \lambda = \Diamond\top,\, D(\nu_0, ctr(\pi)),\, \textit{and}\; \varphi_{\pi_2} \in Im; \\
\Box(\varphi_{\pi_1} \rightarrow \varphi_{\pi_2}) & \textit{if} \;\; \lambda = \Diamond\top,\, D(\nu_0, ctr(\pi)),\, \textit{and}\; \varphi_{\pi_2} \notin Im, \\
 & \textit{or} \;\; \lambda = \Diamond\top,\, A(\nu_0, ctr(\pi)),\, \textit{and}\; \varphi_{\pi_2} \in Im.
\end{cases}
$$

Then, for each leaf $\ell \in V^{\ell}$, *the* leaf-formula φ_ℓ^τ *(or, simply* φ_ℓ, *when* τ *is clear from the context) is defined as:*

$$\varphi_\ell = \bigwedge_{\pi \in prefix(\pi_\ell)} \varphi_\pi.$$

Finally, for each class c, *the* class-formula φ_c^τ *(or, simply,* φ_c, *when* τ *is clear from the context), is defined as:*

$$\varphi_c = \bigvee_{\ell \in leaves(c)} \varphi_{\pi_\ell}.$$

Definition 7 (run). *Let* $\tau = (V, E, b, l, s)$ *be a modal decision tree,* ν *a node in* τ, *and* I *an instance in a modal dataset* $\mathcal{I}$. *Then, the* run of τ on I from ν, *denoted* $\tau(I, \nu)$, *is defined as:*

$$\tau(I, \nu) = \begin{cases} l(\nu) & \textit{if } \nu \in V^{\ell} \\ \tau(I, \swarrow(\nu)) & \textit{if } I \Vdash \varphi_{\pi_{\swarrow(\nu)}} \\ \tau(I, \searrow(\nu)) & \textit{if } I \Vdash \varphi_{\pi_{\searrow(\nu)}}. \end{cases}$$

The run of τ on I *(or the* class assigned to I by τ*), denoted* $\tau(I)$*, is defined as* $\tau(I, root(\tau))$.

Following the above definition, a modal decision tree classifies an instance using its class-formulas, and does so by checking, progressively, the path-formulas that contribute to build a leaf-formula, which, in turn, is one of the disjuncts that take part in a class-formula. Observe that, inter alia, this implies that propositional decision trees can be seen as particular cases of modal decision trees even from a semantic point of view: formulas of the type $\varphi_1 \wedge \varphi_2$ behave exactly as in the propositional case, while those of the type $\varphi_1 \rightarrow \varphi_2$, are such that their the antecedent is always included as a conjunct in their corresponding leaf-formula, effectively reducing it to a conjunction, as in the propositional case.

Now, on the one side, the efficiency of classification depends on how leaf-formulas are checked, while on the other side correctness and completeness depend on their semantics. Let us start by evaluating the efficiency of modal decision trees.

Definition 8 (efficiency). *We say that a decision tree τ of height h is* efficient *if and only if, for every dataset $\mathcal{I}$ and every instance $I \in \mathcal{I}$, it is the case that its run $\tau(I)$ can be computed in polynomial time with respect to h and to the size of I. A family of decision trees is* efficient *if and only all of its decision trees are efficient.*

The following result holds due to the fact that model checking an ML formula against a Kripke structure can be done in polynomial time in the sizes of the structure and the formula [6], and the fact that the size of the formula associated to a path is linear in the length of the path itself.

Theorem 1 (efficiency of $\mathcal{MDT}$). *The family $\mathcal{MDT}$ is efficient.*

Now, we want to prove that modal decision trees are correct.

Definition 9 (correctness). *We say that a decision tree τ is* correct *if and only if, for every dataset $\mathcal{I}$ and every instance $I \in \mathcal{I}$, it is the case that I satisfies exactly one of its class-formulas φ_c. A family of decision trees is* correct *if and only all of its decision trees are correct.*

The following lemma can be proved by induction on the lengths of the paths, and the correctness of $\mathcal{MDT}$ follows.

Lemma 1. *Let τ be a modal decision tree, and let $\pi_1 = \nu_0 \rightsquigarrow \nu_{h-1} \cdot \swarrow(\nu_{h-1})$ and $\pi_2 = \nu_0 \rightsquigarrow \nu_{h-1} \cdot \searrow(\nu_{h-1})$ be two paths. Then, $\varphi_{\pi_1} \leftrightarrow \neg\varphi_{\pi_2}$ is valid.*

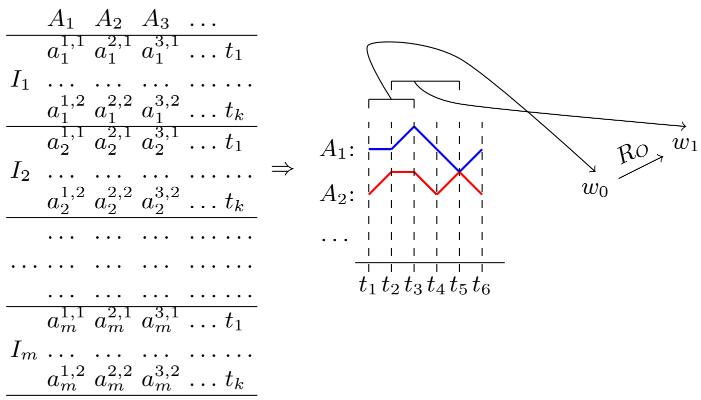


Fig. 3. Typical presentation of an implicit temporal data set.

Theorem 2 (correctness of $\mathcal{MDT}$). *The family $\mathcal{MDT}$ is correct.*

Finally, we discuss the completeness of modal decision trees with respect to modal logic.

Definition 10 (completeness). *We say that a family of decision trees is* strongly complete for *a logical formalism if and only if, for each of its formula φ, there is a decision tree τ and a class $c \in \mathcal{C}$ such that $\varphi_c \leftrightarrow \varphi$ is valid. We also say that a family of decision trees is* weakly complete for *a logical formalisms if and only if, for each of its formula φ, there is a decision tree τ and two classes $c, \bar{c} \in \mathcal{C}$ such that $\varphi_c \rightarrow \varphi$ and $\varphi_{\bar{c}} \rightarrow \neg\varphi$ are both valid.*

Modal decision trees are strongly complete with respect to propositional logic by definition, and weakly complete with respect to modal logic.

Lemma 2. *Let $\varphi \in ML$. Then, there exists a modal decision tree τ and two leaves $\ell_c, \ell_{\bar{c}} \in V^\ell$ such that $\varphi_{\pi_{\ell_c}} \leftrightarrow \varphi$ and $\varphi_{\pi_{\ell_{\bar{c}}}} \leftrightarrow \neg\varphi$ are both valid.*

Theorem 3 (completeness of $\mathcal{MDT}$). *The family $\mathcal{MDT}$ is strongly complete for PL and weakly complete for ML.*

4 Applications

To show the potential of modal symbolic learning, in this section we consider two representative learning situations: from temporal data and from spatial data. As we have observed, spatial/temporal datasets can be seen as modal ones, and modal logic can be declined into suitable spatial/temporal logics that are able to describe such data. An example of *dimensional* dataset in the temporal case

	temporal						spatial					
seed	propositional			modal			propositional			modal		
	acc.	sen.	spe.	acc.	sen.	spe.	acc.	sen.	spe.	acc.	sen.	spe.
1	79.17	79.17	95.83	88.89	88.89	97.78	59.58	59.58	96.33	79.58	79.58	98.14
2	83.33	83.33	96.67	88.89	88.89	97.78	62.50	62.50	96.59	79.58	79.58	98.14
3	80.56	80.56	96.11	93.06	93.06	98.61	63.75	63.75	96.70	67.92	67.92	97.08
4	77.78	77.78	95.56	91.67	91.67	98.33	62.50	62.50	96.59	79.58	79.58	98.14
5	84.72	84.72	96.94	91.67	91.67	98.33	62.92	62.92	96.63	75.83	75.83	97.80
6	77.78	77.78	95.56	88.89	88.89	97.78	57.08	57.08	96.10	71.25	71.25	97.39
7	83.33	83.33	96.67	84.72	84.72	96.94	71.25	71.25	97.39	80.00	80.00	98.18
8	80.56	80.56	96.11	91.67	91.67	98.33	62.92	62.92	96.63	75.83	75.83	97.80
9	80.56	80.56	96.11	84.72	84.72	96.94	58.75	58.75	96.25	77.08	77.08	97.92
10	75.00	75.00	95.00	87.50	87.50	97.50	62.92	62.92	96.63	79.58	79.58	98.14
avg.	80.27	80.27	96.05	89.16	89.16	97.83	62.42	62.42	96.58	76.62	76.62	97.87

Table 1. Test results: propositional versus modal learning from the public, 1-dimensional data set NATOPS (left), and from the public, 2-dimensional data set INDIAN PINES. Performances are reported in percentage points.

is given in Fig. 3 (left); here, m instances are described by several attributes, each one of which takes value on each of the time points that contribute to the description. Thus, this is a set of multi-variate time series; examples of real situations that can be described by sets of multi-variate time series range from hospitalized patients that are constantly monitored, to different sport activities described by the values of wearable sensors, to industrial machines whose behaviour is recorded over time.

In many such situations, the relevant information is not necessarily visible at time points, but rather at time intervals, and in many cases the information to be extracted is concerned with prolonged events that take place at the same, or overlapping, or separate times, which is, again, a situation that is more naturally described with intervals rather than points. One way to extract such information is considering the multi-variate time series that corresponds to each instance, as in Fig. 3 (right), and each interval that can be built on it. Each such interval is regarded as a world, as in Fig. 3 (right), and worlds are connected through interval-interval relations. Taking the standard approach to do so results in having 12 interval-interval relations, excluding equality, that is *meets* (R_A), *overlaps* (R_O), *begins* (R_B), *ends* (R_E), *during* (R_D), and *later* (R_L); in turn, these give rise to a multi-modal logic which is known as HS (from the authors that first introduced it, Halpern and Shoham [7]) which we can use to extract knowledge from a single-dimensional dataset. In Fig. 3 (right), we have shown the relation *overlaps* by way of example. In the spatial case, we can generalize both the definition of world and the relations between worlds, and devise a 2-dimensional version of HS, in order to apply the same idea.

We performed a simple classification experiment on two public datasets, using a prototype, simple version of $\mathcal{MDT}$ (available at [12]); besides being publicly available, the chosen datasets have been selected taking into account their number of attributes and instances, and their representativeness for temporal and spatial problems. The first dataset is temporal, and known as NATOPS. It contains data generated by sensors on the hands, elbows, wrists and thumbs, in all three coordinates, along the temporal axis, of subjects performing several repetitions of aircraft hand signals, chosen among the 24 most often used ones; the problem consists in recognizing the specific signal. The second one is spatial, known as INDIAN PINES, and contains an hyperspectral image over a single landscape in Indiana (US) with 145×145 pixels, each represented by 220 spectral reflectance bands, and classified into one or more of sixteen types of crops; the problem is to recognize the type of cultivation in each pixel. While it would be premature to draw any conclusions from a single group of experiments, we can already see the improvement that we can expect to observe stepping from a static to a modal approach in Tab.1. The results (accuracy, sensitivity, specificity) marked as *modal* are compared with those obtained with the same datasets using simple aggregating functions and propositional decision trees (*propositional*).

5 Conclusions

In this paper, we have shown how propositional decision trees can be generalized into modal decision trees. To this end, we have first highlighted the desirable properties of a family of decision trees in terms of efficiency of classification and logical properties, with respect to a given logical formalism. Then, we designed a family of efficient decision trees that is correct with respect to modal logic.

Application-wise, we have argued that, on the one side, different kinds of data are inherently non-propositional, including dimensional (temporal, spatial, spatial/temporal) data, graph-based data, and textual data, and that, on the other side, the logical formalisms that fit such cases are inherently modal. We considered two specific dimensional cases (a temporal one and a spatial one), and executed a learning experiment comparing the performances of propositional and modal decision trees on the same problem and under the same conditions. Temporal and spatial learning have been deeply studied in the machine learning literature; our purpose here is not that of comparing the performances of learning models in absolute terms, but to show the improvement that we can expect from introducing modal logic in symbolic learning schemata.

The current implementation of modal decision trees is simpler than the one presented in this paper. The problem of devising an efficient implementation of a learning algorithm that extracts full modal decision trees is still open. While the problem of extracting the optimal decision tree is knowingly NP-hard already at the propositional level, much work has been done on approximation algorithms; adapting such algorithms to this proposal, and studying their computational complexity, is an open issue as well.

Finally, decision trees are not the only symbolic learning classification method that can be generalized from the propositional to the modal case; the same can be done, at least, with rule-based systems and ensembles of trees, giving rise to what could be called *modal symbolic learning*.

References

1. Aiello, M., van Benthem, J.: A modal walk through space. Journal of Applied Non-Classical Logics **12**(3-4), 319–364 (2002)
2. Belson, W.A.: A technique for studying the effects of television broadcast. Journal of the Royal Statistical Society **5**(3), 195–202 (1956)
3. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic. Cambridge University Press (2001)
4. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and regression trees. Wadsworth Publishing Company (1984)
5. Clarke, E.M., Emerson, E.A.: Design and synthesis of synchronization skeletons using branching-time temporal logic. In: Proceedings of the Workshop on Logics of Programs. Lecture Notes in Computer Science, vol. 131, pp. 52–71. Springer (1981)
6. Clarke, E.M., Grumberg, O., Peled, D.A.: Model checking. MIT Press (2001)
7. Halpern, J.Y., Shoham, Y.: A propositional modal logic of time intervals. Journal of the ACM **38**(4), 935–962 (1991)
8. Kass, G.V.: An exploratory technique for investigating large quantities of categorical data. Journal of the Royal Statistical Society **29**(2), 119–127 (1980)
9. Lutz, C., Wolter, F.: Modal logics of topological relations. Logical Methods in Computer Science **2**(2), 1–41 (2006)
10. Messenger, R., Mandell, L.: A modal search technique for predictive nominal scale multivariate analysis. Journal of the American Statistical Association **67**(340), 768–772 (1972). https://doi.org/10.1080/01621459.1972.10481290
11. Morgan, J.N., Sonquist, J.A.: Problems in the analysis of survey data, and a proposal. Journal of American Statistical Association **58**(302), 415–434 (1963). https://doi.org/10.2307/2283276
12. Pagliarini, G., Manzella, F., Sciavicco, G., Stan, I.E.: ModalDecisionTrees.jl: Interpretable models for native time-series & image classification (v0.80) (2022), zenodo, doi: 10.5281/zenodo.7040420
13. Parliament and Council of the European Union: General data protection regulation (2016), https://gdpr-info.eu/
14. Pnueli, A.: The temporal logic of programs. In: Proceedings of the 18th Annual Symposium on Foundations of Computer Science (FOCS). p. 46–57. IEEE Computer Society (1977)
15. Quinlan, J.R.: Induction of decision trees. Machine Learning **1**, 81–106 (1986)
16. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann (1993)
17. Quinlan, J.R.: Simplifying decision trees. International Journal of Human-Computer Studies **51**(2), 497–510 (1999)