Knowledge Extraction with Interval Temporal Logic Decision Trees

Guido SCIAVICCO[1] guido.sciavicco@unife.it
Ionel Eduard STAN[1,2] ioneleduard.stan@unife.it

[1]Applied Computational Logic and Artificial Intelligence Laboratory,
Department of Mathematics and Computer Science,
University of Ferrara, Italy

[2]Department of Mathematical, Physical and Computer Sciences,
University of Parma, Italy

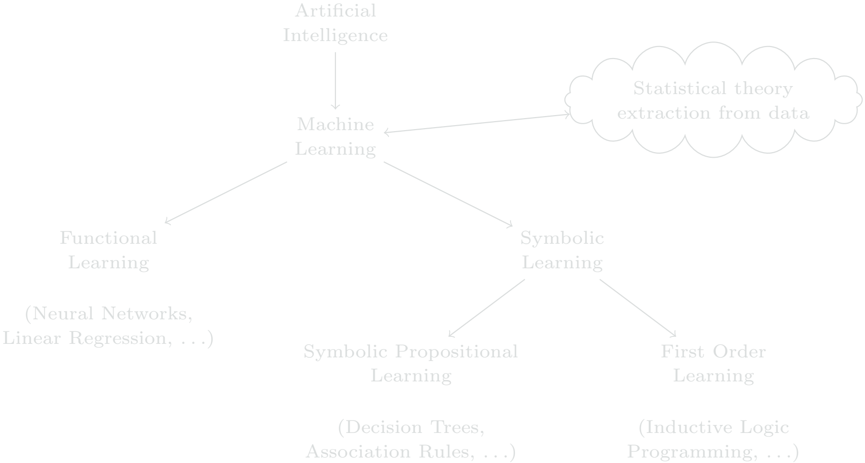
Introduction

Introduction

Machine Learning (ML) is the basis of modern Artificial Intelligence. From a symbolic point of view, it deals with the problem of inducing a model from available data.

Model induction, in opposition with model, or formula, deduction, is a statistical process, but, nonetheless, it can be logic-based.

Introduction



Artificial Intelligence

Machine Learning

Statistical theory extraction from data

Functional Learning

(Neural Networks, Linear Regression, . . .)

Symbolic Learning

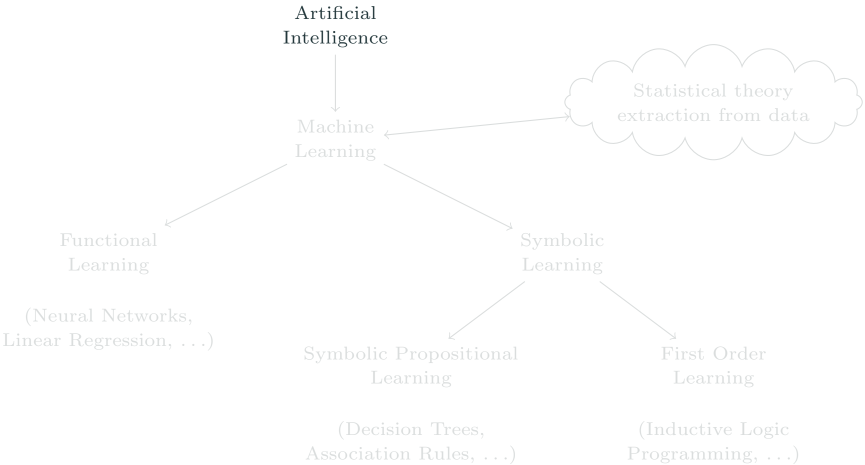
Symbolic Propositional Learning

(Decision Trees, Association Rules, . . .)

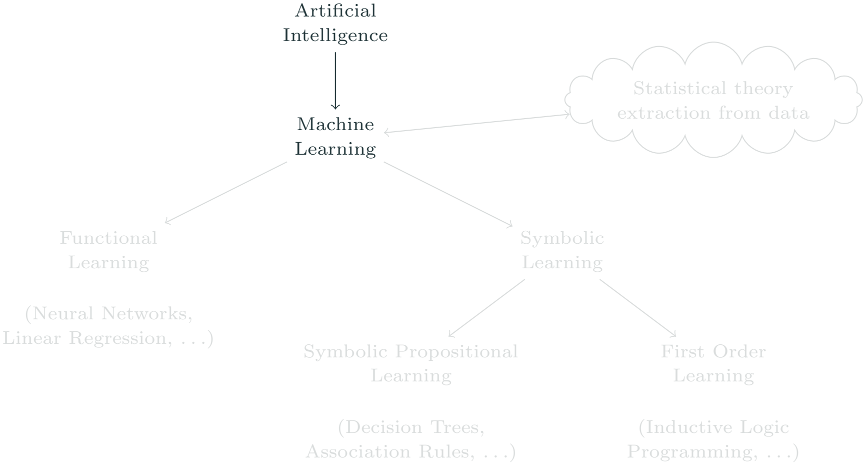
First Order Learning

(Inductive Logic Programming, . . .)

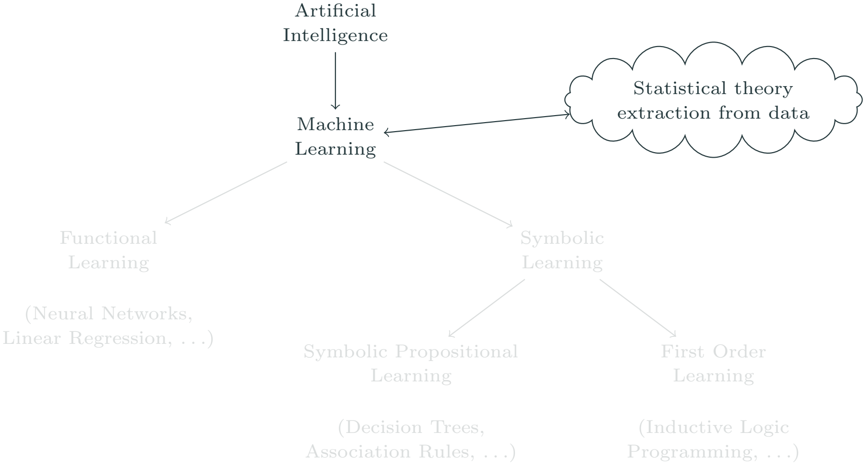
Introduction


Artificial Intelligence
Machine Learning
Statistical theory extraction from data
Functional Learning
(Neural Networks, Linear Regression, . . .)
Symbolic Learning
Symbolic Propositional Learning
(Decision Trees, Association Rules, . . .)
First Order Learning
(Inductive Logic Programming, . . .)

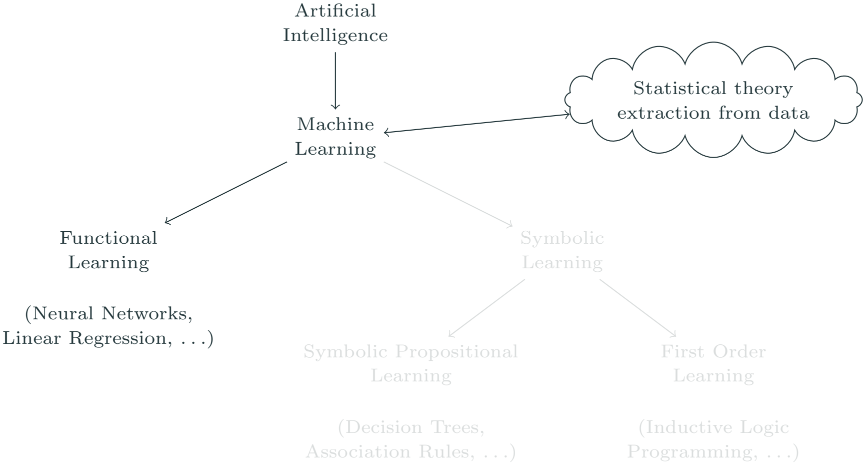
Introduction


Artificial Intelligence
Machine Learning
Statistical theory extraction from data
Functional Learning
(Neural Networks, Linear Regression, . . .)
Symbolic Learning
Symbolic Propositional Learning
(Decision Trees, Association Rules, . . .)
First Order Learning
(Inductive Logic Programming, . . .)

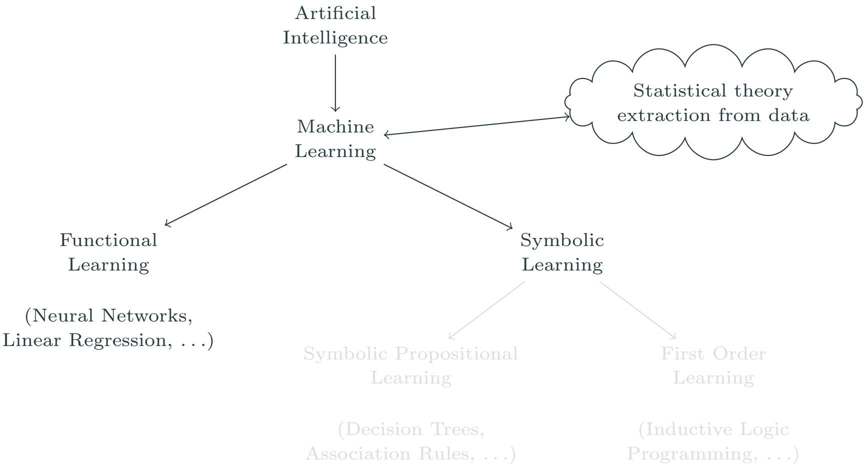
Introduction


Artificial Intelligence
Machine Learning
Statistical theory extraction from data
Functional Learning
(Neural Networks, Linear Regression, . . .)
Symbolic Learning
Symbolic Propositional Learning
(Decision Trees, Association Rules, . . .)
First Order Learning
(Inductive Logic Programming, . . .)

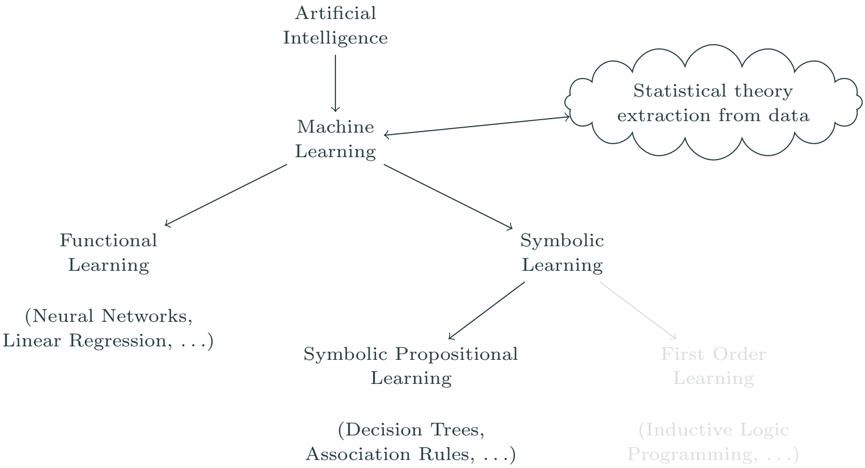
Introduction



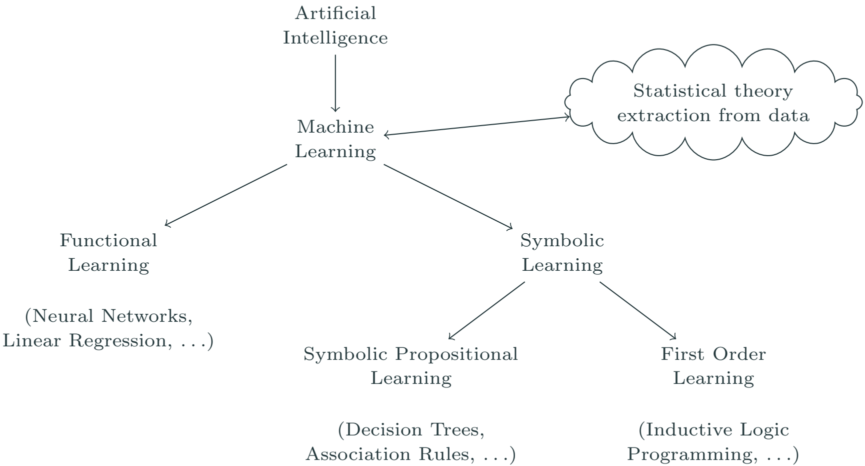
Introduction


Artificial Intelligence
Machine Learning
Statistical theory extraction from data
Functional Learning
(Neural Networks, Linear Regression, ...)
Symbolic Learning
Symbolic Propositional Learning
(Decision Trees, Association Rules, ...)
First Order Learning
(Inductive Logic Programming, ...)

Introduction


Artificial Intelligence
Machine Learning
Statistical theory extraction from data
Functional Learning
(Neural Networks, Linear Regression, . . .)
Symbolic Learning
Symbolic Propositional Learning
(Decision Trees, Association Rules, . . .)
First Order Learning
(Inductive Logic Programming, . . .)

Introduction



Artificial Intelligence

Machine Learning

Statistical theory extraction from data

Functional Learning

(Neural Networks, Linear Regression, . . .)

Symbolic Learning

Symbolic Propositional Learning

(Decision Trees, Association Rules, . . .)

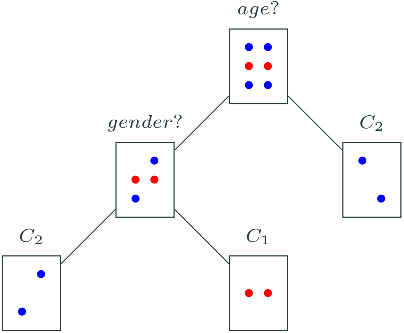
First Order Learning

(Inductive Logic Programming, . . .)

Introduction

Classic ML models are designed for static data (i.e., without no temporal component). Data sets are usually collections of instances, each represented by a set of attributes plus (in supervised learning models) a class.

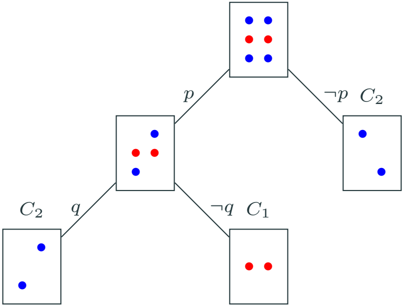
Most popular symbolic learning models are based on propositional logic, which is implicitly used to represent the results. The most striking example of this is decision tree learning:



Introduction

Classic ML models are designed for static data (i.e., without no temporal component). Data sets are usually collections of instances, each represented by a set of attributes plus (in supervised learning models) a class.

Most popular symbolic learning models are based on propositional logic, which is implicitly used to represent the results. The most striking example of this is decision tree learning:



which can be seen "logically" (p can be, e.g., *"age is over 30"*).

Introduction

But what if data is inherently temporal where each instance is a multivariate time series (i.e., a set of variables changing over time)?

Classification of multivariate time series is an active area of research across the scientific disciplines, such as air temperature in climate science, rates of inflation in economics, trends in infectious diseases in medicine, pronunciations of word signs in lingustics, sensor recordings of systems in aerospace engineering, among others.

Introduction

Types of learning schemas for (multivariate) time series:

Reference	Transformation	Distance-based	Feature-based	Function-based	Symbolic-based			Ontology	
					Tree-based	Rule-based	Ensemble-based	Point-based	Interval-based
(Kakizawa et. al., 1998)		✓		✓				✓	
(Diez et. al., 2001)			✓				✓	✓	
(Yamada et. al., 2003)		✓		✓				✓	
(Balakrishnan & Madigan, 2006)		✓			✓			✓	
(Bartocci et. al. 2014)	✓		✓			✓		✓	
(Baydogan & Runger, 2015)	✓		✓				✓	✓	
(Brunello et. al., 2019)	✓		✓		✓				✓
(Lucena-Sánchez et. al., 2019)	✓		✓			✓			✓

In this talk, we present a native, symbolic learning schemata for inducing temporal decision trees for multivariate time series classification where the learnt symbolic theory is expressed in the interval temporal logic HS.

A Theory of Static Decision Trees

A Theory of Static Decision Trees

Consider a labelled static data set $\mathcal{D} = \{D_1, \dots, D_m\}$ described by a set of attributes $\mathcal{A} = \{A_1, \dots, A_n\}$ and, w.l.o.g., associated with a set of classes $\mathcal{C} = \{Yes, No\}$. Let $dom(A)$ denote the domain of an attribute $A \in \mathcal{A}$.

The language of static decision trees encompasses a set of propositional decisions:

$$\mathcal{S} = \{A \bowtie a \mid A \in \mathcal{A}, a \in dom(A)\},$$

where $\bowtie \in \{\leq, =\}$.

W.l.o.g., binary static decision trees are formulas of the following grammar:

$$\hat{\varphi} ::= (S \wedge \hat{\varphi}) \uplus (\neg S \wedge \hat{\varphi}) \mid C,$$

where $C \in \mathcal{C}$ and $S \in \mathcal{S}$.

A decision S is interpreted over a single instance D using classical propositional logic: we say, e.g., that D satisfies the decision $A \leq a$ if A's value is less than or equal to $a \in dom(A)$ in D, and we use the symbol $D \models (A \leq a)$.

A Theory of Static Decision Trees

A decision tree is interpreted over a labelled data set $\mathcal{D}$ via the semantic relation $\hat{\models}_\theta$, which generalizes $\models$ from single instances to data sets: we need to define the notion of a data set satisfying $\hat{\varphi}$ with parameter θ, that is, $\mathcal{D} \hat{\models}_\theta \hat{\varphi}$.

The parameter θ formalizes the notion of how well a decision tree $\hat{\varphi}$ represents $\mathcal{D}$.

By comparing $C(D)$ (true class) and $\hat{\varphi}(D)$ (predicted class) for each instance $D \in \mathcal{D}$ we can compute the performance of a decision tree $\hat{\varphi}$ in terms of its confusion matrix:

	$C(D) = No$	$C(D) = Yes$
$\hat{\varphi}(D) = No$	True Negative (TN)	False Negative (FN)
$\hat{\varphi}(D) = Yes$	False Positive (FP)	True Positive (TP)

A Theory of Static Decision Trees

The root of a tree $\hat{\varphi}$ is associated with the data set $\mathcal{D}$ on which it is interpreted, and, in general, each node of the tree is associated with a subset $\mathcal{D}' \subset \mathcal{D}$ and a binary decision S.

A set $\mathcal{D}'$ is partitioned into two subsets $\mathcal{D}'_1$ and $\mathcal{D}'_2$, that contain, respectively the instances that satisfy S and those that do not.

From the leaves, one can inductively compute the confusion matrix of each node, and the confusion matrix of the root is the one we associate with the tree itself.

A Theory of Static Decision Trees

The rules for $\hat{\models}_\theta$ are now immediate:

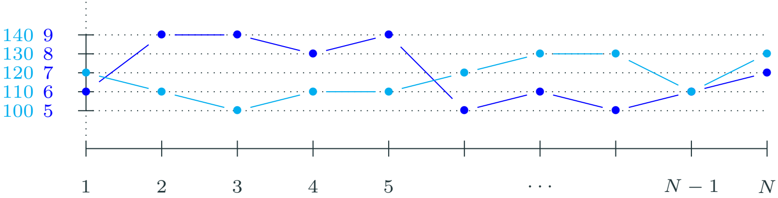
$$\begin{array}{lll}
\mathcal{D} \mathrel{\hat{\models}}_\theta No & \text{if} & \theta = \begin{array}{|c|c|}\hline |\mathcal{D}_{No}| & |\mathcal{D}| - |\mathcal{D}_{No}| \\ \hline 0 & 0 \\ \hline \end{array}\text{, where} \\
 & & \mathcal{D}_{No} = \{D \in \mathcal{D} \mid C(D) = No\}, \\
\mathcal{D} \mathrel{\hat{\models}}_\theta Yes & \text{if} & \theta = \begin{array}{|c|c|}\hline 0 & 0 \\ \hline |\mathcal{D}| - |\mathcal{D}_{Yes}| & |\mathcal{D}_{Yes}| \\ \hline \end{array}\text{, where} \\
 & & \mathcal{D}_{Yes} = \{D \in \mathcal{D} \mid C(D) = Yes\}, \\
\mathcal{D} \mathrel{\hat{\models}}_\theta (S \wedge \hat{\varphi}_1) \uplus (\neg S \wedge \hat{\varphi}_2) & \text{if} & \theta = \theta_1 + \theta_2, \mathcal{D}_1 \mathrel{\hat{\models}}_{\theta_1} \hat{\varphi}_1, \text{ and } \mathcal{D}_2 \mathrel{\hat{\models}}_{\theta_2} \hat{\varphi}_2, \text{ where} \\
 & & \mathcal{D}_1 = \{D \in \mathcal{D} \mid D \models S\}, \mathcal{D}_2 = \{D \in \mathcal{D} \mid D \models \neg S\}, \\
 & & \mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2, \text{ and } \mathcal{D}_1 \cap \mathcal{D}_2 = \emptyset.
\end{array}$$

$$\begin{array}{rccl}
\textbf{Dataset:} & \mathcal{D} & = & \{D_1, \ldots, D_m\} \\
\textbf{Attributes:} & \mathcal{A} & = & \{A_1, \ldots, A_n\} \\
\textbf{Decisions:} & \mathcal{S} & = & \{A \bowtie a\} \\
\textbf{Decision Trees:} & \hat{\varphi} & ::= & (S \wedge \hat{\varphi}) \mathbin{\ooalign{$\forall$\cr\hfil$|$\hfil}} (\neg S \wedge \hat{\varphi}) \mid C \\
\textbf{Learning Problem:} & \mathcal{D} & \hat{\models}_\theta & \hat{\varphi}
\end{array}$$

Time Series, Timelines and Interval Temporal Logic

Time Series

A time series is a set of variables that change over time, and they can be univariate or multivariate.



A labelled temporal data set $\mathcal{T} = \{T_1, \ldots, T_m\}$ is a set of temporal instances described by a set of temporal attributes $\mathcal{A} = \{A_1, \ldots, A_n\}$, each being a N-points time series, and, w.l.o.g., associated to a set of classes $\mathcal{C} = \{Yes, No\}$.

The multivariate time series classification is the problem of finding a formula (symbolic classification) or a function (functional classification) that associates multivariate time series to classes.

Timelines

Sciavicco et. al. (2019) proposed a discretization method for mapping a time series into a timeline. Intuitively, a timeline can be seen as the categorical counterpart of a numerical time series.

After the discretization of a time series that describe a continuous process, it makes little sense to model their values at each time-point, but, instead, they are naturally represented as interval-based ontology.

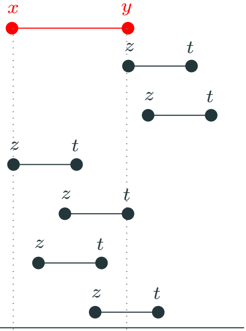
Therefore, if a static numerical data set is naturally expressed in propositional logic, a multivariate time series is naturally expressed in an interval temporal logic.

HS: The Modal Logic of Allen's Interval Relations

Let $[N]$ be an initial subset of $\mathbb{N}$ of length N. An interval over $[N]$ is an ordered pair $[x, y]$, where $x, y \in D$ and $x < y$. Let $\mathbb{I}([N])$ be the set of all intervals over $[N]$.

Excluding the equality, there are 12 binary ordering relations between 2 intervals on a linear order, often called Allen's interval relations, which give rise to corresponding unary modalities over frames where intervals are primitive entities.

HS: The Modal Logic of Allen's Interval Relations

HS modality	Definition w.r.t. interval structure			Example
$\langle A \rangle$ (after)	$[x,y] R_A [z,t]$	$\Leftrightarrow$	$y = z$	
$\langle L \rangle$ (later)	$[x,y] R_L [z,t]$	$\Leftrightarrow$	$y < z$	
$\langle B \rangle$ (begins)	$[x,y] R_B [z,t]$	$\Leftrightarrow$	$x = z \wedge t < y$	
$\langle E \rangle$ (ends)	$[x,y] R_E [z,t]$	$\Leftrightarrow$	$y = t \wedge x < z$	
$\langle D \rangle$ (during)	$[x,y] R_D [z,t]$	$\Leftrightarrow$	$x < z \wedge t < y$	
$\langle O \rangle$ (overlaps)	$[x,y] R_O [z,t]$	$\Leftrightarrow$	$x < z < y < t$	

For each modality $\langle X \rangle$, its transpose corresponds to the inverse $R_{\overline{X}}$ of R_X, i.e., $R_{\overline{X}} = (R_X)^{-1}$. HS formulas are defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle X \rangle \varphi \mid \langle \overline{X} \rangle \varphi,$$

where $p \in \mathcal{AP}$ (atomic proposition), and $X \in \{A, L, B, E, D, O\}$.

HS: The Modal Logic of Allen's Interval Relations

An interval model is a pair $T = \langle \mathbb{I}([N]), V \rangle^1$, where $V : \mathcal{AP} \to 2^{\mathbb{I}([N])}$ is a valuation function that assigns to each proposition $p \in \mathcal{AP}$ the set of interval $V(p)$ on which p holds.

The truth of a HS formula φ on an interval $[x, y]$ of an interval model T is defined by structural induction on formulas:

$$\begin{array}{lll}
T, [x, y] \Vdash p & \text{iff} & [x, y] \in V(p), \text{ for } p \in \mathcal{AP}, \\
T, [x, y] \Vdash \neg\psi & \text{iff} & T, [x, y] \nVdash \psi, \\
T, [x, y] \Vdash \psi \vee \xi & \text{iff} & T, [x, y] \Vdash \psi \text{ or } T, [x, y] \Vdash \xi, \\
T, [x, y] \Vdash \langle X \rangle\psi & \text{iff} & \exists [z, t] \text{ s.t. } [x, y] R_X [z, t] \text{ and } T, [z, t] \Vdash \psi, \\
T, [x, y] \Vdash \langle \overline{X} \rangle\psi & \text{iff} & \exists [z, t] \text{ s.t. } [x, y] R_{\overline{X}} [z, t] \text{ and } T, [z, t] \Vdash \psi.
\end{array}$$

[1] We deliberately use the symbol T to indicate both a timeline and a time series.

A Theory of Temporal Decision Trees

A Theory of Temporal Decision Trees

Now, consider a labelled temporal data set $\mathcal{T} = \{T_1, \ldots, T_m\}$ described by the time series $\mathcal{A} = \{A_1, \ldots, A_n\}$ and, as before, classified with classes $\mathcal{C} = \{Yes, No\}$.

Unlike the static case, we do not ask if $A \bowtie a$ only in the current interval, but also if there exists an interval, related to the current one, so that a decision becomes $\langle X \rangle (A \bowtie a)$. It follows that $\bowtie \in \{\leq, =, >\}$.

Moreover, we may relax the requirement $A \bowtie a$ over a given interval $[x, y]$ by asking that at least a certain fraction of the values of A in the interval $[x, y]$ meet the condition, denoted by $A \bowtie_\alpha a$ with $\alpha \in (0, 1] \subset \mathbb{R}$.

In some applications, trends are more important than values. From this, we denote by A^z the z-th discrete derivative of A; we identify A with A^0.

A Theory of Temporal Decision Trees

Thus, the language of temporal decision trees encompasses a set of temporal decisions:

$$\begin{array}{rcl}\mathcal{S} & = & \{\langle X\rangle(A^z \bowtie_\alpha a), \langle\overline{X}\rangle(A^z \bowtie_\alpha a) \mid X \in \mathcal{X}, A \in \mathcal{A}, a \in dom(A^z)\} \cup \\ & & \{A^z \bowtie_\alpha a \mid A \in \mathcal{A}, a \in dom(A^z)\},\end{array}$$

where $\mathcal{X} = \{A, L, B, E, D, O\}$ are interval operators of HS.

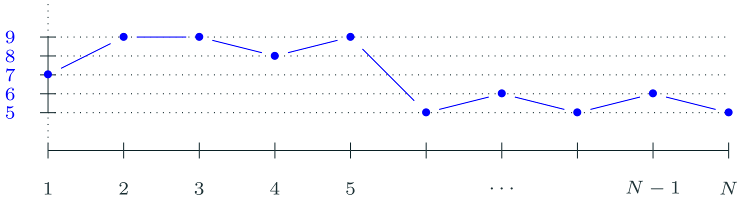
Temporal decision trees are formulas of the following grammar:

$$\hat{\varphi} ::= (S \wedge \hat{\varphi}) \veebar (\neg S \wedge \hat{\varphi}) \mid C,$$

where $C \in \mathcal{C}$ and $S \in \mathcal{S}$.

For a temporal decision $S \in \mathcal{S}$, we use the notation $T, [x, y] \Vdash S$ or $T \Vdash S$ when $[x, y]$ is the (current) reference interval of T.

A Theory of Temporal Decision Trees



In the above time series T:

- $T, [1,2] \Vdash \langle A \rangle (A >_{0.75} 8)$ because $\exists [2,5]$ such that $[1,2] R_A [2,5]$ and

$$\frac{|\{t \mid 2 \leq t \leq 5 \text{ and } A(t) > 8\}|}{5-2+1} = \frac{3}{4} = 0.75;$$

- $T, [3,5] \nVdash \langle L \rangle (A >_{0.2} 7)$ that is $T, [3,5] \Vdash [L](A \leq_{0.2} 7)$;
- $T, [N-1, N] \nVdash \langle \overline{L} \rangle (A \leq_{1.0} 4)$ that is $T, [N-1, N] \Vdash [\overline{L}](A >_{1.0} 4)$.

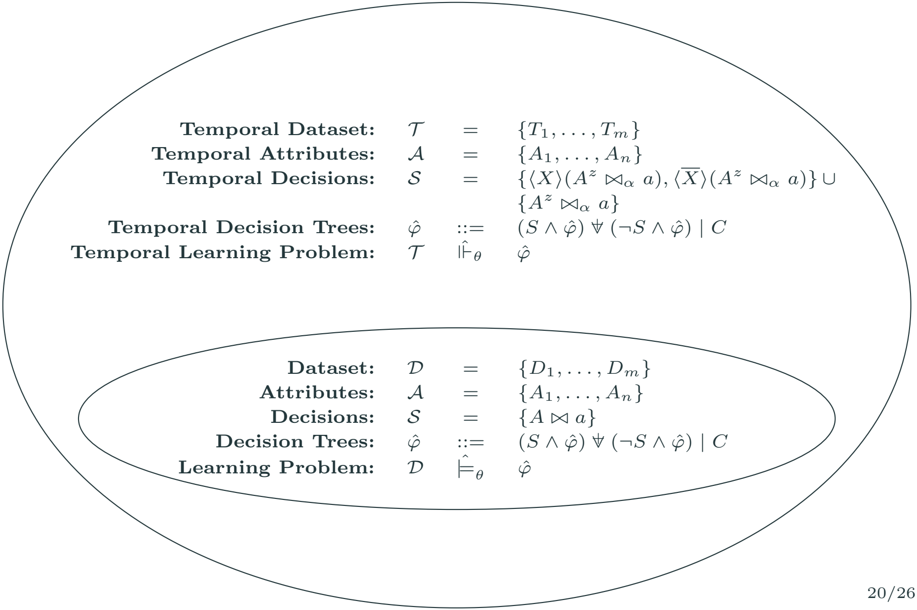
A Theory of Temporal Decision Trees

As before, for a labelled temporal data set $\mathcal{T}$, the rules for $\hat{\Vdash}_\theta$ are immediate:

$$\begin{array}{lll}
\mathcal{T} \,\hat{\Vdash}_\theta No & \text{if} & \theta = \begin{array}{|c|c|}\hline |\mathcal{T}_{No}| & |\mathcal{T}| - |\mathcal{T}_{No}| \\ \hline 0 & 0 \\ \hline \end{array}, \text{ where} \\
 & & \mathcal{T}_{No} = \{T \in \mathcal{T} \mid C(T) = No\}, \\
\mathcal{T} \,\hat{\Vdash}_\theta Yes & \text{if} & \theta = \begin{array}{|c|c|}\hline 0 & 0 \\ \hline |\mathcal{T}| - |\mathcal{T}_{Yes}| & |\mathcal{T}_{Yes}| \\ \hline \end{array}, \text{ where} \\
 & & \mathcal{T}_{Yes} = \{T \in \mathcal{T} \mid C(T) = Yes\}, \\
\mathcal{T} \,\hat{\Vdash}_\theta (S \wedge \hat{\varphi}_1) \forall (\neg S \wedge \hat{\varphi}_2) & \text{if} & \theta = \theta_1 + \theta_2, \mathcal{T}_1 \,\hat{\Vdash}_{\theta_1} \hat{\varphi}_1, \text{ and } \mathcal{T}_2 \,\hat{\Vdash}_{\theta_2} \hat{\varphi}_2, \text{ where} \\
 & & \mathcal{T}_1 = \{T \in \mathcal{T} \mid T \Vdash S\}, \mathcal{T}_2 = \{T \in \mathcal{T} \mid T \Vdash \neg S\}, \\
 & & \mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2, \text{ and } \mathcal{T}_1 \cap \mathcal{T}_2 = \emptyset.
\end{array}$$

Observe that a temporal decision S entails unique $\mathcal{T}_1$ and $\mathcal{T}_2$, but not unique (new) reference intervals for the time series of $\mathcal{T}_1$; however, this choice is implementative, not theoretical.

Mind Map



$$
\begin{array}{rccl}
\textbf{Temporal Dataset:} & \mathcal{T} & = & \{T_1, \ldots, T_m\} \\
\textbf{Temporal Attributes:} & \mathcal{A} & = & \{A_1, \ldots, A_n\} \\
\textbf{Temporal Decisions:} & \mathcal{S} & = & \{\langle X\rangle(A^z \bowtie_\alpha a), \langle \overline{X}\rangle(A^z \bowtie_\alpha a)\} \cup \\
 & & & \{A^z \bowtie_\alpha a\} \\
\textbf{Temporal Decision Trees:} & \hat{\varphi} & ::= & (S \wedge \hat{\varphi}) \mathbin{\ooalign{$\forall$\cr\hfil$\shortmid$\hfil}} (\neg S \wedge \hat{\varphi}) \mid C \\
\textbf{Temporal Learning Problem:} & \mathcal{T} & \hat{\Vdash}_\theta & \hat{\varphi}
\end{array}
$$

$$
\begin{array}{rccl}
\textbf{Dataset:} & \mathcal{D} & = & \{D_1, \ldots, D_m\} \\
\textbf{Attributes:} & \mathcal{A} & = & \{A_1, \ldots, A_n\} \\
\textbf{Decisions:} & \mathcal{S} & = & \{A \bowtie a\} \\
\textbf{Decision Trees:} & \hat{\varphi} & ::= & (S \wedge \hat{\varphi}) \mathbin{\ooalign{$\forall$\cr\hfil$\shortmid$\hfil}} (\neg S \wedge \hat{\varphi}) \mid C \\
\textbf{Learning Problem:} & \mathcal{D} & \hat{\models}_\theta & \hat{\varphi}
\end{array}
$$

Temporal J48

Entropy-based Learning

Hyafil and Rivest (1976) have proved that computing the optimal decision tree is an NP-hard problem, and for this reason sub-optimal approaches have been proposed.

Among others, to tackle such problem, entropy-based learning has been proposed by Quinlan (1986).

A Working Implementation: Temporal J48

WEKA offers the implementation of the algorithm C4.5 in Java, called J48.

A natural way to embed the theory of temporal decision trees into C4.5 is to extend the implementation of J48 into an algorithm which we call Temporal J48.

In addition to J48, Temporal J48 requires the following parameters:

- the value of α (which in this experiment we did not optimize),
- the value $z_{\max}$ (maximum discrete derivative which in this experiment was set to 0),
- the reference interval policy (how to choose the witnessing existential interval among many), and
- the subset of HS modalities that one allows during the learning phase.

Experiments and Results

Data Sets

Dataset	Train cases	Test cases	Channels	Length	Classes
AtrialFibrillation (AF)	24	6	2	150	3
FingerMovements (FM)	104	26	28	50	2
Libras (LI)	180	45	2	45	15
LSST (LS)	168	42	6	36	14
NATOPS (NA)	96	24	24	51	6
RacketSports (RS)	96	24	6	30	4
SelfRegulationSCP1 (S1)	96	24	6	150	2
SelfRegulationSCP1 (S2)	96	24	7	150	2
UWaveGestureLibrary (UW)	96	24	3	150	8

Table 1: A summary of resampled datasets from Bagnall et. al. (2018).

Results

Dataset	AF	FM	LI	LS	NA	RS	S1	S2	UW
J48 $1, 0, 0, 0$	83.33	50.00	40.00	30.95	79.17	70.83	66.67	50.00	66.67
J48 $1, 1, 0, 0$	83.33	42.31	51.11	30.95	75.00	87.50*	66.67	54.17	62.50
J48 $1, 1, 1, 1$	83.33	42.31	64.44	38.10	62.50	79.17	66.67	62.50	54.17
ED$_I$	83.33	76.92	86.67	42.86*	70.83	79.17	66.67	66.67	87.50
DTW$_I$	100.00*	65.38	91.11*	33.33	87.50*	75.00	66.67	66.67	91.67
DTW$_D$	83.33	57.69	91.11*	40.48	87.50*	83.33	83.33*	66.67	95.83*
T. J48 0.5	66.67	57.69	80.00	23.81	83.33	70.83	83.33*	54.17	62.50
T. J48 0.6	66.67	57.69	71.11	26.19	79.17	79.17	66.67	75.00*	58.33
T. J48 0.7	66.67	53.85	73.33	23.81	75.00	66.67	66.67	66.67	62.50
T. J48 0.8	83.33	80.77*	75.56	26.19	75.00	62.50	66.67	62.50	66.67
T. J48 0.9	66.67	80.77*	71.11	23.81	66.67	62.50	66.67	70.83	66.67

Table 2: Test results in terms of accuracy. Underlined results are the best ones in the group, and starred results are the absolute best ones.

Computational resources have been offered by the University of Udine, Italy, supported by the PRID project *Efforts in the uNderstanding of Complex interActing SystEms* (ENCASE)

Example of Interval-based Temporal Theory

```
<L> var5 <= -2.756591
|   <InvA> var5 <= 0.308951
|   |   <=> var2 > -0.916901
|   |   |   <InvB> var0 <= 2.832243: Badminton_Clear (6.0)
|   |   |   [InvB] var0 > 2.832243: Badminton_Smash (1.0)
|   |   [=] var2 <= -0.916901
|   |   |   <B> var3 <= -0.207743
|   |   |   |   <InvB> var0 > 4.115426
|   |   |   |   |   <D> var0 > 1.452113: Squash_ForehandBoast (3.0)
|   |   |   |   |   [D] var0 <= 1.452113: Squash_BackhandBoast (1.0)
|   |   |   |   [InvB] var0 <= 4.115426
|   |   |   |   |   <InvB> var0 <= -0.215688: Badminton_Smash (2.0)
|   |   |   |   |   [InvB] var0 > -0.215688: Badminton_Clear (3.0)
|   |   |   [B] var3 > -0.207743: Squash_ForehandBoast (14.0)
|   [InvA] var5 > 0.308951
|   |   <InvB> var5 <= -2.27452
|   |   |   <InvA> var0 <= -1.044682: Squash_BackhandBoast (3.0/1.0)
|   |   |   [InvA] var0 > -1.044682: Squash_ForehandBoast (7.0)
|   |   [InvB] var5 > -2.27452: Squash_BackhandBoast (21.0)
[L] var5 > -2.756591
|   <A> var0 <= 0.098773
|   |   <InvB> var0 > -0.960139
|   |   |   <B> var4 <= 0.625893: Badminton_Smash (16.0)
|   |   |   [B] var4 > 0.625893: Badminton_Clear (1.0)
|   |   [InvB] var0 <= -0.960139: Badminton_Clear (2.0)
|   [A] var0 > 0.098773
|   |   <L> var4 > 8.703901: Badminton_Smash (4.0)
|   |   [L] var4 <= 8.703901: Badminton_Clear (12.0)
```

Figure 1: One of the Temporal J48 models trained on RacketSports data set.

Conclusions

Conclusions

We have presented:

- the theory of static decision trees and its extension to the temporal case,
- a symbolic, interpretable method for classifying multivariate time series by means of temporal decision trees that implements the temporal theory,
- a comparison of our method against other methods that are known in literature.

We plan to:

- add pruning techniques to our method,
- optimize all parameters (e.g., α, subset of modalities of HS, etc.) in future experiments,
- extend the temporal theory not only for classification tasks but also for regression tasks,
- adapt other well-known symbolic learning schemas in the same way.

Backup Slides

Entropy

For a static data set $\mathcal{D} = \{D_1, \ldots, D_m\}$ with classes $\mathcal{C} = \{C_1, C_2\}$, the information conveyed (or entropy) of $\mathcal{D}$ is:

$$Info(\mathcal{D}) = -\sum_{i=1}^{2} \Big(\frac{|\{D \in \mathcal{D} \mid C(D) = C_i\}|}{|\mathcal{D}|} \cdot \log\big(\frac{|\{D \in \mathcal{D} \mid C(D) = C_i\}|}{|\mathcal{D}|} \big) \Big).$$

The entropy is inversely proportional to the purity degree of $\mathcal{D}$ w.r.t. the class values.

Observe that the above definition can be applied to a temporal data set $\mathcal{T}$ as well.

Entropy-based Learning: Static Data Set

Splitting, which is the main greedy operation in learning a DT, is defined over a specific attribute A and over $\bowtie \in \{\leq, =\}$. In particular, the entropy of splitting/partitioning $\mathcal{D}$ based on the static decision $A \bowtie a$ is defined as:

$$InfoSplit(A, a, \bowtie, \mathcal{D}) = \sum_{i=1}^{2} \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \cdot Info(\mathcal{D}_i),$$

where:

- $\mathcal{D}_1 = \{D \in \mathcal{D} \mid D \models A \bowtie a\}$, and
- $\mathcal{D}_2 = \{D \in \mathcal{D} \mid D \not\models A \bowtie a\}$.

The entropy of attribute A is defined as:

$$InfoAtt(A, \mathcal{D}) = \min_{\bowtie \in \{\leq, =\}, a \in dom(A)} \{InfoSplit(A, a, \bowtie, \mathcal{D})\},$$

and, finally, the information gain of A is defined as:

$$Gain(A, \mathcal{D}) = Info(\mathcal{D}) - InfoAtt(A, \mathcal{D}).$$

Entropy-based Learning: Temporal Data Set

C4.5 is designed to allow ID3 to cope with numerical data. Temporal C4.5 is the natural theoretical extension of C4.5 to deal with undiscretized time series.

Following the presentation, for a temporal data set $\mathcal{T}$, Temporal C4.5 has new parameters for a temporal decision $\langle X\rangle(A^z \bowtie_\alpha a)$, that is:

$$InfoSplit(A, X, a, \bowtie, \alpha, z, \mathcal{T}) = \sum_{i=1}^{2} \frac{|\mathcal{T}_i|}{|\mathcal{T}|} \cdot Info(\mathcal{T}_i),$$

where:

- $X \in \{A, L, B, E, D, O, \overline{A}, \overline{L}, \overline{B}, \overline{E}, \overline{D}, \overline{O}\} \cup \{eq\}$ (equality),
- $\mathcal{T}_1 = \{T \in \mathcal{T} \mid T, [x, y] \Vdash \langle X\rangle(A^z \bowtie_\alpha a)\}$,
- $\mathcal{T}_2 = \{T \in \mathcal{T} \mid T, [x, y] \nVdash \langle X\rangle(A^z \bowtie_\alpha a)\}$,
- $\alpha \in (0, 1] \subset \mathbb{R}$, and
- $z \geq 0$,

and:

$$InfoAtt(A, \mathcal{T}) = \min_{\substack{X \in \mathcal{X} \cup \{eq\}, \alpha \in (0,1],\\ 0 \leq z \leq z_{\max}, a \in dom(A)}} \{InfoSplit(A, X, a, \bowtie, \alpha, z, \mathcal{T})\}.$$