Proceedings of the
2nd Workshop on Artificial Intelligence and Formal Verification, Logics, Automata and Synthesis (OVERLAY),
September 25, 2020

Symbolic Learning with Interval Temporal Logic: the Case of Regression

*

Estrella Lucena-Sanchez[1], Guido Sciavicco[2], and Ionel Eduard Stan[3]

[1]University of Ferrara and Unversity of Modena and Reggio Emilia, Italy
[2]University of Ferrara, Italy
[3]University of Ferrara and University of Parma, Italy
[1]estrella.lucenasanchez@unife.it
[2]guido.sciavicco@unife.it
[3]ioneleduard.stan@unife.it

Abstract

Regression analysis is the statistical process used to estimate the relationship between a dependent variable and one or more independent variables. In machine learning, typical statistical approaches to regression such as linear regression are often replaced with symbolic learning, such as decision tree regression, to capture non-linear behaviour while keeping the interpretability of the results. For temporal series, regression is sometimes enhanced by using historical values of the independent variables. In this paper, we show how temporal regression can be handled by a symbolic learner based on interval temporal logic decision trees.

1 Introduction

A *multivariate time series* [2] is a set of variables that change over time. Each variable of a multivariate time series is an ordered collection of N real values, and, usually, in a multivariate time series one identifies one *dependent* variable B, whose values we are interested to predict, and the *independent* variables $A_1, \ldots, A_n$, whose values are used for the prediction. Multivariate time series emerge in many application contexts. The temporal history of some hospitalized patient can be described by the time series of the values of his/her temperature, blood pressure, and oxygenation; the pronunciation of a word in sign language can be described by the time series of the relative and absolute positions of the ten fingers w.r.t. some reference point; different sport activities can be distinguished by the time series of some relevant physical quantities. Having identified the dependent variable, the most relevant and interesting problem defined on a time series is *forecasting*, that is, the problem of predicting its future values, and it is typically solved by *regression*.

Regression is the statistical process used to capture the parameters that influence the relationship between independent and dependent variables. In the context of temporal regression with machine learning,

*The authors acknowledge the partial support by the Italian INDAM GNCS project *Strategic Reasoning and Automated Synthesis of Multi-Agent Systems*

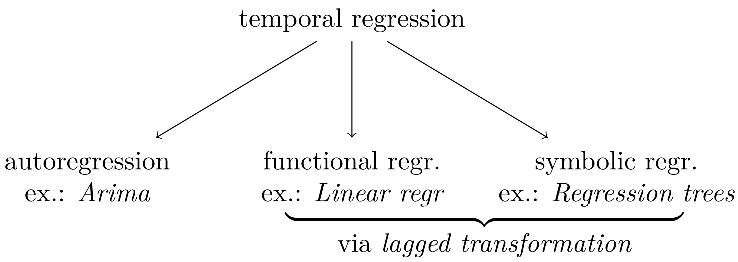


Figure 1: Approaches to temporal regression.

there are three main categories of approach. *Autoregression* is a set of statistical techniques developed mainly to model the idea that past values of the independent variable may influence, in several different ways, future values. These are techniques originally developed for univariate series, and later extended to the multivariate case. *Lagged variables* is the machine learning generalization to the latter idea, so to say, and it consists of transforming the original data set by adding virtual variables that represent the past values. For example, for a independent variable $A_i(t)$ (in which we make it explicit the temporal parameter), one adds the lagged variable $A_i(t-1)$ and $A_i(t-2), \ldots$, that correspond, respectively, to the value of A_i one, two, $\ldots$, units of time before. In this way, a static learner (where each instance of the data set is independent from any other) can be used on the transformed data set, giving rise to the possibility of using *functional* regression algorithms, such as linear regression and neural networks, among others, and *symbolic* regression algorithms, such as regression tree learning algorithms. While certain classical regression algorithms, such as linear regression or decision trees are interpretable by nature (that is, they return an explicit model), the possibility of understanding the underlying process may be hampered by the presence of lagged variables.

Regression trees were introduced in the CART (*classification and regression trees*) system in [3], but their practical introduction can be dated back to [7, 8]. The problem of extracting the *optimal* decision tree from a data set is NP-hard [6], which justifies the use of sub-optimal approaches. The *ID3* algorithm [7] is a greedy approach to the extraction of a decision tree, later extended and generalized in the algorithm C4.5 [9]. In this paper we sketch the theoretical basis of *temporal regression tree* learning, taking inspiration from [4], in which the idea of temporal decision tree learning was first introduced. The underlying idea is to generalize the concept of lagged data. Instead of transforming the original time series into a static data set by simply flattening the past values of the independent variables into atemporal instances, we produce an intermediate data set in which each instance is, itself, a multivariate time series of a fixed length l (which is the amount of the lag). In this way, to predict a certain value $B(t)$, we use the multivariate time series given by the values of the variables $A_1, \ldots, A_n$ at the times $t-l, t-(l-1), \ldots, t-1, t$, and we label it with the value $B(t)$, mimicking, in a way, a *moving window* approach. From [10, 4], we know that an interval temporal logic such as HS [5] is a very convenient language to describe time series. So, we use a greedy approach such as the one presented in [4], suitably adapted to the regression case, to learn a locally optimal regression tree.

2 Time Series and Interval Temporal Logic

Let $[N]$ an initial subset of $\mathbb{N}$ of length N. An *interval* over $[N]$ is an ordered pair $[x, y]$, where $x, y \in [N]$ and $x < y$, and we denote by $\mathbb{I}([N])$ the set of all intervals over $[N]$. If we exclude the identity relation, there are 12 different Allen's relations between two intervals in a linear order [1]: the six relations R_A (adjacent to), R_L (later than), R_B (begins), R_E (ends), R_D (during), and R_O (overlaps), depicted in Fig. 2, and their inverses, that is, $R_{\bar{X}} = (R_X)^{-1}$, for each $X \in \mathcal{X}$, where $\mathcal{X} = \{A, L, B, E, D, O\}$. Halpern and Shoham's modal logic of temporal intervals (HS) is defined from a set of propositional letters $\mathcal{AP}$, and by associating a universal modality $[X]$ and an existential one $\langle X \rangle$ to each Allen's relation R_X. Formulas of HS are obtained by:

HS	**Allen's relations**
$\langle A \rangle$	$[x, y]R_A[x', y'] \Leftrightarrow y = x'$
$\langle L \rangle$	$[x, y]R_L[x', y'] \Leftrightarrow y < x'$
$\langle B \rangle$	$[x, y]R_B[x', y'] \Leftrightarrow x = x', y' < y$
$\langle E \rangle$	$[x, y]R_E[x', y'] \Leftrightarrow y = y', x < x'$
$\langle D \rangle$	$[x, y]R_D[x', y'] \Leftrightarrow x < x', y' < y$
$\langle O \rangle$	$[x, y]R_O[x', y'] \Leftrightarrow x < x' < y < y'$

Graphical representation

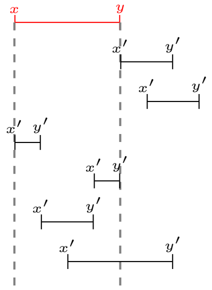


Figure 2: Allen's interval relations and HS modalities.

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle X \rangle\varphi \mid \langle \bar{X} \rangle\varphi, \tag{1}$$

where $p \in \mathcal{AP}$ and $X \in \mathcal{X}$. The other Boolean connectives and the logical constants, e.g., $\rightarrow$ and $\top$, as well as the universal modalities $[X]$, can be defined in the standard way, i.e., $[X]p \equiv \neg\langle X \rangle\neg p$. For each $X \in \mathcal{X}$, the modality $\langle \bar{X} \rangle$ (corresponding to the inverse relation $R_{\bar{X}}$ of R_X) is said to be the *transpose* of the modalities $\langle X \rangle$, and vice versa. The semantics of HS formulas is given in terms of *timelines* $T = \langle \mathbb{I}([N]), V \rangle$, where $V : \mathcal{AP} \rightarrow 2^{\mathbb{I}([N])}$ is a *valuation function* which assigns to each atomic proposition $p \in \mathcal{AP}$ the set of intervals $V(p)$ on which p holds. The *truth* of a formula φ on a given interval $[x, y]$ in an interval model T is defined by structural induction on formulas as follows:

$$\begin{array}{lll} T, [x, y] \Vdash p & \text{if} & [x, y] \in V(p), \text{ for } p \in \mathcal{AP}; \\ T, [x, y] \Vdash \neg\psi & \text{if} & T, [x, y] \not\Vdash \psi; \\ T, [x, y] \Vdash \psi \vee \xi & \text{if} & T, [x, y] \Vdash \psi \text{ or } T, [x, y] \Vdash \xi; \\ T, [x, y] \Vdash \langle X \rangle\psi & \text{if} & \text{there is } [w, z] \text{ s.t } [x, y]R_X[w, z] \text{ and } T, [w, z] \Vdash \psi; \\ T, [x, y] \Vdash \langle \bar{X} \rangle\psi & \text{if} & \text{there is } [w, z] \text{ s.t } [x, y]R_{\bar{X}}[w, z] \text{ and } T, [w, z] \Vdash \psi. \end{array} \tag{2}$$

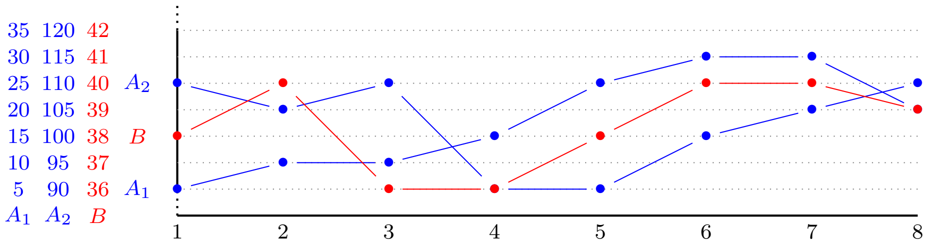
A time series can be described by formulas of HS, in which propositional letters represent *decisions* of the type $A_i \bowtie k$ ($\bowtie \in \{\leq, =, >\}$) for some value k. We can arbitrarily impose that a decision $A_i \bowtie k$ is *true* on an interval $[x, y]$ if and only if the ratio between number of time points between x and y having a value that satisfies $\bowtie k$ and the quantity $y - x$ is at least α, for a fixed α. From there, we can lift the evaluation to a formula. For example, in Fig. 3, the series T is such that $T, [3, 4] \Vdash [A](A_1 > 10)$, with $\alpha = 1$.

3 Interval Temporal Logic Regression Trees

Static regression learning is exemplified in Fig. 3, middle, left. If we want to construct a static regression model for the values of B, say, from the time point 3 to the time point 8, we build a data set with the values of A_1, A_2 at each of such points associated with the corresponding value of B. Static lagged regression improves upon static learning by associating also the values of A_1 and A_2 at $t - 3$, $t - 2$, and so on; a multivariate linear regression algorithm, or a regression tree, may benefit from this historical information to build a model for B. Since a static regression tree is an NP-hard problem, the most popular solution is to use a sub-optimal greedy strategy based on the following principles [7, 8]: given a data set $\mathcal{D} = \{I_1, \ldots, I_m\}$, it is defined the notion of *split*, by means of which $\mathcal{D}$ is partitioned into two sets $\mathcal{D}_1, \mathcal{D}_2$ on the basis of a decision (propositional formula) of the type $A_i \bowtie k$, for some attribute A_i and value k; in other words, for each instance I_j we check:

$$I_j \Vdash A_i \bowtie k$$

to decide if I_j belongs to $\mathcal{D}_1$ or $\mathcal{D}_2$. Then, $\mathcal{D}$, $\mathcal{D}_1$, and $\mathcal{D}_2$ are compared against each other establish the amount of information conveyed by that split; in regression problems, the information can simply be the



A_1	A_2	B
10	110	36
15	90	36
25	90	38
30	100	40
30	105	40
20	110	39

...	$A_1(t-1)$	$A_2(t-1)$	$A_1(t)$	$A_2(t)$	$B(t)$
...	10	105	10	110	36
...	10	110	5	100	36
...	5	100	5	110	38
...	5	110	15	115	40
...	15	115	20	115	40
...	20	115	25	105	39

$A_1[(t-2);(t-1);t]$	$A_2[(t-2);(t-1);t]$	B
5;10;10	110;105;110	36
10;10;15	105;110;90	36
25;5;5	95;100;110	38
5;5;15	100;110;115	40
5;15;20	110;115;115	40
15;20;25	115;115;105	39

Figure 3: A multivariate time series with three variables (top). Static regression (middle, left). Static lagged regression (middle, right). Multivariate time series regression (bottom).

variance of the predicted attribute in each data set. So, a simple greedy strategy can be devised: *(i)* at each step, if the stopping condition is not reached, find the attribute A_i and the value k that maximize the *information* conveyed by a split on that decision; *(ii)* split the current data set on the basis of the decision taken at the previous step; *(iii)* perform two recursive calls on the obtained data sets. At the end of this process, the entire model is described by a tree whose edges are labeled by decisions, and each branch can be seen as a propositional formula.

Here, we propose to design an algorithm capable to extract a regression tree after a transformation as in Fig. 3, bottom. For a fixed lag l, to each value $B(t)$, we associate the *finite time series* described by $A_i(t-l), A_i(t-l+1), \ldots$ for each i, and each represented by a string. We define a split based on a HS formula of the type $\langle X\rangle(A_i \bowtie k)$, which we call *atomic HS formulas*, where decisions have taken the place of propositional letters. Thus, fixed a lag l, at the beginning, the time series T with N points is replaced by a data set $\mathcal{T} = \{T_1, \ldots, T_{N-l+1}\}$ in which each instance is, itself, a multivariate time series labelled by a value of B and encompassing l points. Each of such series is initially associated to a reference interval generically denoted $[0, 1]$, and the main split operation now consists of checking:

$$T_j, [x, y] \Vdash \langle X\rangle(A_i \bowtie k).$$

By applying the same learning algorithm, then, one obtains a tree whose edges are labeled with atomic HS formulas instead of propositional formulas, and whose branches, then, can be read as formulas of HS. In this way, the obtained regression tree keeps the original non-linear (step-type) behaviour, but takes into account the recent history of a value to perform the prediction, and it does so natively. Initial experiments on real data sets seem to offer encouraging results.

4 Conclusions

We have sketched the theoretical bases for a temporal regression tree learning algorithm, based on the interval temporal logic HS, which allows us to natively take into account the past values of the independent variables and their mutual relationships.

References

[1] J. F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[2] G. Box, G. Jenkins, and G. Reinsel. *Time Series Analysis.* Wiley, 1970.

[3] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and Regression Trees.* Taylor&Francis, 1984.

[4] A. Brunello, G. Sciavicco, and I. E. Stan. Interval temporal logic decision tree learning. In *Proc. of the 16th European Conference on Logics in Artificial Intelligence (JELIA)*, volume 11468 of *Lecture Notes in Computer Science*, pages 778–793. Springer, 2019.

[5] J. Y. Halpern and Y. Shoham. A propositional modal logic of time intervals. *Journal of the ACM*, 38(4):935–962, 1991.

[6] L. Hyafil and R. L. Rivest. Constructing optimal binary decision trees is NP-complete. *Information Processing Letters*, 5(1):15–17.

[7] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[8] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, 1993.

[9] J. R. Quinlan. Simplifying decision trees. *International Journal of Human-Computer Studies*, 51(2):497–510, 1999.

[10] G. Sciavicco, I. E. Stan, and A. Vaccari. Towards a general method for logical rule extraction from time series. In *Proc. of the 8th International Work-Conference on the Interplay Between Natural and Artificial Computation (IWINAC)*, volume 11487 of *Lecture Notes in Computer Science*, pages 3–12. Springer, 2019.