mathematics

Article

Toward Explainable Time-Series Numerical Association Rule Mining: A Case Study in Smart-Agriculture

Iztok Fister, Jr. [1,*], **Sancho Salcedo-Sanz** [2], **Enrique Alexandre-Cortizo** [2], **Damijan Novak** [1], **Iztok Fister** [1], **Vili Podgorelec** [1] and **Mario Gorenjak** [3]

[1] Faculty of Electrical Engineering and Computer Science, University of Maribor, 2000 Maribor, Slovenia; damijan.novak@um.si (D.N.); iztok.fister@um.si (I.F.); vili.podgorelec@um.si (V.P.)
[2] Department of Signal Processing and Communications, Universidad de Alcalá, 28805 Alcalá de Henares, Madrid, Spain; sancho.salcedo@uah.es (S.S.-S.); enrique.alexandre@uah.es (E.A.-C.)
[3] Center for Human Molecular Genetics and Pharmacogenomics, Faculty of Medicine, University of Maribor, Taborska Ulica 8, 2000 Maribor, Slovenia; mario.gorenjak@um.si
* Correspondence: iztok.fister1@um.si

Abstract

This paper defines time-series numerical association rule mining in smart-agriculture applications from an explainable-AI perspective. Two novel explainable methods are presented, along with a newly developed algorithm for time-series numerical association rule mining. Unlike previous approaches, such as fixed interval time-series numerical association, the proposed methods offer enhanced interpretability and an improved data science pipeline by incorporating explainability directly into the software library. The newly developed xNiaARMTS methods are then evaluated through a series of experiments, using real datasets produced from sensors in a smart-agriculture domain. The results obtained using explainable methods within numerical association rule mining in smart-agriculture applications are very positive.

Keywords: association rule mining; explainable artificial intelligence (XAI); numerical association rule mining; optimization algorithms

MSC: 68T05

Academic Editors: Florin Leon, Mircea Hulea and Marius Gavrilescu

Received: 4 June 2025
Revised: 24 June 2025
Accepted: 26 June 2025
Published: 28 June 2025

Citation: Fister, I., Jr.; Salcedo-Sanz, S.; Alexandre-Cortizo, E.; Novak, D.; Fister, I.; Podgorelec, V.; Gorenjak, M. Toward Explainable Time-Series Numerical Association Rule Mining: A Case Study in Smart-Agriculture. *Mathematics* **2025**, *13*, 2122. https://doi.org/10.3390/math13132122

1. Introduction

Numerical association rule mining methods have recently attracted the interest of researchers due to their easy applicability and good performance in many different prediction problems [1]. These methods, which are based particularly on stochastic population-based nature-inspired algorithms, show efficiency when searching for association rules in large datasets, mainly due to the stochastic nature of population-based algorithms, which have several advantages over the deterministic methods, e.g., Apriori methods, which search the whole search space in a deterministic way [2–4]. Additionally, a very positive point of these methods is that they can handle both numerical and categorical attributes concurrently. Thus, they do not involve any specific discretization steps in the data preprocessing stages. The importance of this research area has already been covered in several review papers published recently [5,6]. The main issue with evolving numerical association rule mining lies mainly in the number of identified (mined) association rules, which can go beyond thousands of rules, making them non-explainable AI approaches in general.

Canonical numerical association rule mining using population-based, nature-inspired algorithms, is often used for the mining of association rules in classical datasets [7–9] and different applications. Recently, new methods have been proposed, also intended for the mining of time-series datasets [10]. Time-series association rule mining also has great potential in areas where we deal with time data, especially when the data are acquired from Internet of Things (IoT) sensors. One of these areas is smart agriculture, where agricultural processes are monitored increasingly with IoT sensors. A great potential also comes with the introduction of Industry 5.0, along with the corresponding branches, e.g., Agriculture 5.0. On the other hand, Explainable Artificial Intelligence (XAI) [11] is an established field with a vibrant community that has developed a variety of very successful approaches to explain and interpret the predictions of complex machine learning models [12]. The importance of the visualization and explanation of rules becomes even greater in time-series numerical mining. One of the first steps in this direction is represented by the approach proposed in [13], which enhances the interpretability of complex association rules derived from time series, aligning with explainable AI principles. The method was tested using agricultural time-series data and demonstrates promising potential for smart-agriculture applications.

In this paper, we go a step further by introducing two novel post hoc explainable methods, i.e., xNiaARMTS, which are capable of either graphical or non-graphical explainable data analysis. The first proposed explainable method enables users to obtain deeper insights into the behavior of attributes arising in the antecedent and consequent of time-series association rules. On the other hand, the second explainable method developed here, where the stability of time-sequences is estimated, is devoted to statistical stationary analysis. Graphical explainable data analysis is applied to the analysis of trends, seasonality, and cycle behavior of the time series, while the non-graphical method highlights the same data by producing summary statistics and analysis [14].

The main contributions of this work are the following:

- Contribution 1: A new method for time-series numerical association rule mining is developed, called segmented interval time-series numerical association rule mining.
- Contribution 2: Two additional novel explainable methods (xNiaARMTS) are proposed for the analysis of time-series association rules.
- Contribution 3: Nature-inspired Association Rule Mining for Time Series (NiaARMTS) software is developed and published.
- Contribution 4: Extensive experimental work is conducted in order to show the advantages and disadvantages of the proposed explainable methods in a smart-agriculture application.

The remainder of the paper is structured as follows: Section 2 deals with the materials and methods needed to understand the subjects that follow. In Section 3, the proposed explainable methods for mined time-series association rules xNiaARMTS are described in detail. The experimental work and analysis of the results are described in Section 4. The paper is closed in Section 5, where the performed work is summarized and potential directions for future work are outlined.

2. Materials and Methods

2.1. Numerical Association Rule Mining

The classical Apriori algorithm for the mining of transaction databases is designed to work with categorical attributes only. These attributes need to be discretized if the numerical attributes are to be processed using this algorithm. However, when Numerical Association Rule Mining (NARM) is applied, this phase can be avoided. The algorithms for NARM typically work with the attributes represented as either numerical intervals or sets of categorical values.

NARM is defined formally as follows: Let us assume that a set of features ($I = \{A_1^{(.)}, \dots, A_M^{(.)}\}$ of M) called items and a set of transactions ($TD = \{t_1, \dots, t_N\}$) called the database are given, where each feature ($A_i^{(.)}$) is of a specific type ($(.) = \{(Cat), (Num)\}$) (either categorical or numerical attribute type) and each transaction in TD has a unique transaction ID and contains a subset of items in I. Then, the association rule is an implication:

$$X \implies Y, \tag{1}$$

where it holds that $X \subseteq I$, $Y \subseteq I$, and $X \cap Y = \emptyset$. The following measures have been devised for assessing the quality of the association rule:

$$supp(X \implies Y) = \frac{\text{number of transactions containing } X \text{ and } Y}{N}, \tag{2}$$

and

$$conf(X \implies Y) = \frac{supp(X \cup Y)}{supp(X)}, \tag{3}$$

where N denotes the number of transactions and M is the number of features in the transaction database ($|A^{(\cdot)}|$).

During the mining process, only association rules that satisfy the following relations are mined:

$$supp(X \implies Y) \geq S_{min} \text{ and } conf(X \implies Y) \geq C_{min},$$

where the S_{min} variable denotes minimum support and the C_{min} variable represents the minimum confidence. Actually, these variables determine that only association rules with support and confidence higher than the corresponding threshold values are to be taken into consideration, respectively.

2.2. Segmented Interval Time-Series Numerical Association Rule Mining

Usually, we are not interested in all time series but only a part of them. A partial sequence of a time series is also called a segment derived from the time-series matrix (Z), defined as follows:

$$Z = \begin{pmatrix} ts_1^{(1)}, & \dots & ts_i^{(1)}, & \dots, & ts_j^{(1)} & \dots, & ts_T^{(1)}, \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ ts_1^{(k)}, & \dots & \mathbf{ts}_{t_s}^{(k)} & , \dots, & ts_j^{(k)}, & \dots & ts_T^{(k)} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ ts_1^{(l)} & , \dots & ts_i^{(l)} & , \dots, & \mathbf{ts}_{t_e}^{(l)}, & \dots & ts_T^{(l)}, \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ ts_1^{(N)}, & \dots & ts_i^{(N)}, & \dots, & ts_j^{(N)} & \dots & ts_T^{(N)}. \end{pmatrix}, \tag{4}$$

where the $t \in [1, T]$ parameter denotes the time-series counter and T is the length of each episode (also a row of the time-series matrix), while each element of the time series denotes a set of features:

$$ts = \{A_1, A_2, \dots, A_M\}, \tag{5}$$

where M is the total number of features, whose attributes are typically obtained from various sensors. The segment is determined with its starting point (t_s—at the time when element $ts_i^{(k)}$ has arisen) and ending point (t_e—at the time when element $ts_j^{(l)}$ has arisen). In general, segment S represents the part of time series Z within the time interval of $[t_s, t_e]$; in other words,

$$S = (ts_{t_s}, ts_{t_{s+1}}, \dots, ts_{t_e}). \tag{6}$$

Let us mention that segment S is denoted with a border line in Equation (4).

3. Explainability in Rule-Based Time-Series Mining

Explainable artificial intelligence is a growing field that seeks to make the decision-making processes of complex models more transparent and interpretable. In the context of time-series rule mining using NiaARMTS, we focus on post hoc explainability, i.e., analyzing and understanding the logic and behavior of discovered rules after they are mined. In line with this, two methods (xNiaARMTS) are developed.

Each Time-series Association Rule (TAR) in NiaARMTS represents a temporal implication:

$$[t_s, t_e] : \underbrace{A_1^{(.)} \wedge A_2^{(.)} \wedge \cdots \wedge A_m^{(.)}}_{X(\text{also LHS})} \implies \underbrace{C_1^{(.)} \wedge C_2^{(.)} \wedge \cdots \wedge C_n^{(.)}}_{Y(\text{also RHS})},$$

where the interval of $[t_s, t_e]$ determines the time interval valid for the mined rule and $A_i^{(.)}$ and $C_j^{(.)}$ are feature-based constraints on either numerical (Num) or categorical (Cat) attributes (denoted by $(.)$) belonging to either the antecedent (i.e., left-hand side (LHS)) or consequent (i.e., right-hand side (RHS)) of the definite TAR.

Example 1. *Let us suppose a time series with the following features of numerical attributes (Let us emphasize that each element on the LHS or RHS is not a simple discrete item but a constrained feature derived from a time-series segment):*

$$I = \{'temp', 'hum', 'moist', 'light'\},$$

where all the attributes of the features are obtained from the corresponding agriculture sensors for temperature, humidity, moisture, and light during the period between 10 September 2024 and 11 September 2024 from 19:59:29 until 06:18:13. Then, the potential rule mined by NiaARMTS is expressed as follows:

$$[(\text{10 September 2024, 19:59:29}), (\text{11 September 2024, 06:18:13})] : \\ 'temp'[24.2925, 26.2961] \implies 'moist'[2015.6761, 2377.9884].$$

Let us suppose a moisture sensor that returns the resistance between the sensor probes in the interval of $[0, 5000]$. *If the soil is too wet, the resistance is less than 2500. If the soil is too dry, the resistance is higher than 3750. The ideal soil moisture is obtained if the resistance is between 2500 and 3750. In our case, the soil moisture was between 2015.6761 and 2377.9884, which means that the soil was too dry. The conclusion of the observed TAR is that if the temperature is high (e.g., between* 24.3 °C *and* 26.3 °C*), the soil becomes wet.*

To help users understand why a rule holds, how reliable it is, and what makes it distinctive, we introduce two explainable methods:

- Attribute criticality analysis;
- Rule stability assessment.

Before we take a closer look at these methods, the metrics for estimating the quality of the mined TAR used by the method are illustrated in more detail.

3.1. Explainability Metrics for xNiaARMTS

The explainable methods mentioned above also need the metrics necessary for estimating the quality of the mined TARs. In this study, we introduce the following five metrics:

- Coverage;
- Inclusion;
- Amplitude;
- Inverse frequency;
- Generalized constraint tightness.

While the first two metrics are suitable for estimating the mixed types of features arising in TAR, the third is dedicated to estimating the numerical attributes, and the last one is for estimating the discrete attributes. The explainability metrics mentioned above are discussed in more detail in the remainder of the paper.

3.1.1. Coverage Metric

Coverage (also called cover or LHS support) is the support of the left-hand side of the TAR, i.e., supp(X). It represents a measure of how often the rule is applied during the time-series segment. However, the calculation of the metric is different for categorical and numerical attributes. The coverage metric of a specific attribute ($A_i^{(\cdot)}$) for $i = 1, \dots, M$ is defined simply as a ratio between the sum of the attribute and the time-series element of segment $ts \in S$ and the length of time-series segment $|S|$:

$$cover(A_i^{(\cdot)}) = \frac{1}{|S|} \sum_{\forall ts^{(\cdot)} \in S} match(A_i^{(\cdot)}, ts), \tag{7}$$

In other words, the function expressed as $match(A_i^{(\cdot)}, a^{(\cdot)})$ is defined as follows for $\forall a^{(\cdot)} \in ts$:

$$match(A_i^{(\cdot)}, ts) = \begin{cases} 1, & \begin{cases} \text{if } I.a^{(Cat)} = I.A_i^{(Cat)} \wedge a^{(Cat)} = A_i^{(Cat)}, \\ \text{if } I.a^{(Num)} = I.A_i^{(Num)} \wedge a^{(Num)} \in A_i^{(Num)}[Lb, Ub], \end{cases} \\ 0, & \text{otherwise}, \end{cases} \tag{8}$$

The function returns a value of one in cases in which the categorical attribute from the definite time series ($a^{(Cat)} \in ts$) matches the categorical attribute from the TAR ($A_i^{(Cat)} \in X$), or the numerical value of the attribute ($a^{(Num)} \in ts$) is drawn from the interval of $[Lb, Ub]$, as proposed by the corresponding antecedent of the TAR. Thus, the equivalence relation ($I.a^{(\cdot)} = I.A_i^{(\cdot)}$) ensures that the attribute relations are determined based on the same features.

3.1.2. Inclusion Metric

An inclusion metric estimates the number of distinct features used in the TAR (antecedent and consequent), normalized by the total number of features in the time-series segment determined by the interval of $[t_s, t_e]$. Mathematically, the metric is defined as follows:

$$incl(A_i^{(\cdot)}, [t_s, t_e] : X \implies Y) = \frac{|X| + |Y|}{M}, \qquad \text{if } A_i^{(\cdot)} \in X \cup Y, \tag{9}$$

where $|X|$ and $|Y|$ denote the number of features in the antecedent and the number of features in the consequent, respectively, and M is the total number of features within the time-series segment. Indeed, the metric estimates how many attributes emerge in the specific TAR. The higher the value of the metric, the more interesting the rule can be. As is evident from Equation (9), the metric is defined on the TAR level, not on the attribute level. Indeed, the metric is defined only for TARs that include the particular attribute.

3.1.3. Amplitude Metric

A measure of constraint tightness for a definite numerical feature ($A_i^{(Num)}$) is defined as follows for $\forall A_i^{(\cdot)} \in X \cup Y$:

$$ampl(A_i^{(num)}) = 1 - \sum_{\forall A^{(Num)}{}_i \in X \cup Y} \frac{A_i^{(Num)}.Ub - A_i^{(Num)}.Lb}{\max_{\forall ts \in S} ts.a^{(Num)} - \min_{\forall ts \in S} ts.a^{(Num)}}, \tag{10}$$

where the sum of ratios is calculated between the width of the numeric attribute interval ($A_i^{(Num)}[Lb, Ub]$) and the difference between the maximum and minimum values of the corresponding numeric attribute ($ts.a^{(Num)}$) found during the observed time-series segment in the interval of $[t_s, t_e]$. The sum converges to a value of 1, which means, theoretically, that the total interval of observed values is covered by only one TAR. This is in contrast with NARM, where we want to find intervals of numerical attributes that highlight the relationships between attributes and that are the most crucial. Smaller amplitudes suggest more precise, selective conditions.

3.1.4. Inverse Frequency Metric

Let us emphasize that the amplitude metric is defined for numerical features only. Therefore, the inverse frequency measure is defined as the constraint tightness for categorical features. First, the frequency of a categorical attribute is defined as follows:

$$ifreq(A_i^{(Cat)}) = 1 - \frac{1}{|S|} \sum_{\forall ts \in S} \begin{cases} 1, & \text{if } A_i^{(Cat)} \in X \cup Y, \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where $|S|$ denotes the number of elements in time-series segment S. The frequency is calculated as the representation of the specific categories within the segment. Because we prefer the presence of rare categories, the inverse value of the frequency metric is applied to estimate the specific TAR as, simply incrementing the frequencies of all attributes arising in the TAR, regardless of whether they arise in the antecedent or consequent.

3.1.5. Generalized Constraint Tightness Metric

Typically, in an explainability analysis, we are interested in metrics that are independent of the type of attributes arising in the TAR. While the coverage and inclusion metrics satisfy this precondition, this is not true for the amplitude and inverse frequency metrics. Fortunately, the amplitude and inverse frequency metrics are dedicated to estimating the constraint tightness of different attribute types. To find the general metric for the constraint tightness, we need to integrate both metrics presented by Equations (10) and (11) into one metric.

This integration is performed by the introduction of the so-called generalized constraint tightness metric, which is defined as follows:

$$ctigh(A_i^{(\cdot)}) = \begin{cases} ampl(A_i^{(Num)}), & \text{if } isNum(A_i^{(\cdot)}), \\ ifreq(A_i^{(Cat)}), & \text{otherwise,} \end{cases} \tag{12}$$

where the calculation of constraint tightness is conducted according to the corresponding type of attribute. Hence, a function ($isNum(A_i^{(\cdot)})$) returns a true value when the attribute ($A_i^{(\cdot)}$) is of the numerical type and a false value otherwise.

3.2. Attribute Criticality Analysis

The goal of this analysis is to quantify the individual contribution of each attribute to a rule—whether in the antecedent or the consequent. This allows users to focus on the most informative features and interpret the rules with greater precision.

3.2.1. Antecedent Criticality

Each antecedent condition (A_i) of any type is evaluated using the following score:

$$ScoreA(A_i) = \alpha \cdot cover(A_i) + \beta \cdot incl(A_i, [t_s, t_e] : X \implies Y) + \gamma \cdot ctight(A_i), \quad (13)$$

where

- $cover(A_i)$ is the empirical frequency with which A_i is satisfied in the dataset;
- $incl(A_i), [t_s, t_e] : X \implies Y)$ captures a semantic or logical overlap between the attribute (A_i) and the rule's consequent (optional) (These metrics demand the corresponding time-series association rule);
- $ctigh(A_i)$ returns either the width of the numerical condition $|A_i.Ub - A_i.Lb|$ or the class of the categorical attribute.

By default, we use the following weighting values: $\alpha = 0.5$, $\beta = 0.3$, and $\gamma = 0.2$. Although we agree that a formal sensitivity analysis or ablation study could provide deeper insights into the impact of these weights, such an investigation was beyond the scope of this paper, which primarily aimed to introduce and validate the proposed explainable framework. Therefore, the default values of these weights were employed in our study.

Interpretation

High-scoring attributes are those that are rare, selective, and aligned with the outcome. This scoring helps distinguish core drivers from noisy or redundant features.

3.2.2. Consequent Criticality

Similarly, for each consequent attribute C_j of any type, we define:

$$ScoreC(C_j) = \lambda \cdot (1 - cover(C_j)) + \mu \cdot ctigh(C_j). \quad (14)$$

Here, the lower coverage and tighter bounds indicate a more informative or surprising consequence. The recommended weights are $\lambda = 0.4$ and $\mu = 0.6$.

Interpretation

High-scoring attributes are those that are rare, selective, and aligned with the outcome. This scoring helps distinguish core drivers from noisy or redundant features.

Example 2. *Let us suppose a time-series segment with four features, expressed as follows:*

$$I = \{'temp', 'hum', moist, 'light', 'weather'\},$$

where the first four incorporate attributes of the numerical type, while the last one is of the categorical type. The attributes represent measurements obtained from either agricultural sensors (temperature, humidity, moisture, and light) or a meteorological station (weather). The data from the time-series segment were collected from 20:16:21 to 20:17:51 on 8 September 2024 (Table 1).

Based on the time-series segment, the following association rule was mined by NiaARMTS:

$$[(8\text{ September }2024,\ 20{:}16{:}21),(8\text{ September }2024,\ 20{:}17{:}51)] : \\ 'weather'(cloudy) \wedge 'light'[0.0,\ 7.0] \wedge 'moist'[2340.0.0,\ 2341.0] \implies \\ 'temp'[28.3,\ 28.4] \wedge' hum'[60.2,\ 60.3].$$

The results of the attribute critical analysis are illustrated in Table 2, where the values of all three metrics are included. As is evident from the table, all the conditions match only the 'weather' feature due to the same attribute being set in all rows of the selected segment. The 'hum' feature demonstrates strong discriminative power,due to a tight numerical range (high amplitude), but its coverage was the lowest, i.e., it captured only half of the elements within the observed segment. The 'temp' feature, as a consequent, ranks lower than the 'hum' feature due to its lower tight constraint, although it covers 80 % of the time-series elements within the segment. The inclusion metric, which is consistent across features, reflects that three out of five available dataset features are used in the antecedent.

Table 1. Analyzed time-series segment.

Seq.	'temp'	'hum'	'moist'	'light'	'weather'	Timestamp
13	28.5	58.3	2338	11.67	cloudy	8 September 2024 20:16:21
14	28.5	59.7	2342	10.83	cloudy	8 September 2024 20:16:31
15	28.4	60.3	2340	15.83	cloudy	8 September 2024 20:16:41
16	28.4	58.6	2339	0.00	cloudy	8 September 2024 20:16:51
17	28.4	58.3	2340	0.00	cloudy	8 September 2024 20:17:01
18	28.3	60.3	2343	0.00	cloudy	8 September 2024 20:17:11
19	28.3	60.2	2340	0.00	cloudy	8 September 2024 20:17:21
20	28.3	60.2	2340	0.00	cloudy	8 September 2024 20:17:31
21	28.3	60.2	2340	0.00	cloudy	8 September 2024 20:17:41
22	28.3	60.6	2340	0.00	cloudy	8 September 2024 20:17:51

Table 2. Explanation of rule components.

Part	Rank	Feature	Coverage	Inclusion	Tightness	Score
Antecedent	1	'weather'	1.00	1.00	0.00	0.8000
	2	'light'	0.70	1.00	0.56	0.7616
	3	'moist'	0.60	1.00	0.80	0.7600
Consequent	1	'hum'	0.50	n/a	0.96	0.7283
	2	'temp'	0.80	n/a	0.50	0.6516

3.3. Rule Stability Analysis

Time-series rules are inherently sensitive to their time windows. A rule that only holds in a very specific time interval may not be generalized. To assess this, we define a stability metric as follows: Let $I = [t_s, t_e]$ be the original time interval where the rule holds. Additionally, the following two evaluation windows are indicated:

$$I^- = [t_s - \delta, t_e - \delta], \quad I = [t_s, t_e], \quad I^+ = [t_s + \delta, t_e + \delta],$$

where the δ parameter determines the offset from the original evaluation window to the past I^- and the future I^+ evaluation windows. For each window, we re-evaluate whether the rule's antecedent and consequent still co-occur. The co-occurrence is estimated using standard support and confidence metrics, which are redefined as follows:

$$supp(I : X \implies Y) = \frac{\text{Number of time series containing both } X \text{ and } Y}{\text{Total number of time series in segment } S}, \tag{15}$$

and

$$conf(I : X \implies Y) = \frac{supp(I : X \cup Y)}{supp(I : X)}, \tag{16}$$

where attributes X and Y hold within time interval I. Then, the quality of the TAR within the observed interval (I) is expressed as follow:

$$\nabla(I : X \implies Y) = \frac{supp(I : X \implies Y) + conf(I : X \implies Y)}{2}. \tag{17}$$

Let us mention that our intention in using Equation (17) was not to introduce a new theoretical metric but, rather, to provide a practical and interpretable way to evaluate the temporal consistency of a rule. Finally, the stability score is defined as follows:

$$Stability(I : X \Rightarrow Y) = \sqrt{(\nabla(I^- : X \Rightarrow Y) - \nabla(I : X \Rightarrow Y))^2 + (\nabla(I^+ : X \Rightarrow Y) - \nabla(I : X \Rightarrow Y))^2}, \tag{18}$$

where $Stability(I : X \implies Y)$ is an indicator function that calculates Euclidean distances between the original and either past or future evaluation windows and returns values within the interval of $[0.0, 1.0]$. The rule holds in the observed evaluation window if the function returns values greater than a definite confidence interval pre-defined by the user. The purpose of using Euclidean distance in Equation (18) is not to compute spatial distance but, rather, to quantify temporal variability in rule quality, i.e.,

- The function measures how much the rule's support and confidence vary when the time window is shifted.
- A lower stability value implies that the rule maintains similar quality (i.e., consistency) across time, suggesting it is more robust and generalizable.
- A higher value indicates greater sensitivity to the specific time segment, potentially pointing to overfitting or transient behavior.

Interpretation

A high stability score indicates that the rule is temporally robust, i.e., it persists even with small shifts. Unstable rules may be overfit to transient anomalies.

3.4. Discussion

Together, these two methods support a richer understanding of the rules discovered by NiaARMTS. Users are not only shown *what* patterns exist but also *why* they occur, *how reliably* they generalize, and *what makes them unique*. These capabilities align with the key objectives of explainable AI: transparency, user trust, and decision support in high-stakes domains such as healthcare, finance, and industrial monitoring. By offering insights into the causal and statistical underpinnings of identified patterns, new methods (xNiaARMTS) integrated into the NiaARMTS framework foster deeper user engagement and critical evaluation of the results. This is particularly valuable in contexts where understanding the rationale behind decisions can influence policy or operational protocols—for example, the protocol for watering a garden in the context of smart agriculture. Moreover, the system's emphasis on uniqueness helps reduce noise and redundancy in pattern discovery, allowing users to focus on the most salient insights. The combination of statistical rigor and semantic clarity supports a broader range of users, including those without deep technical expertise and domain-specific knowledge.

4. Experiments and Results

4.1. Dataset

The intention in the development of the proposed explainable methods (xNiaARMTS) was to support a smart agriculture process that relies on a real dataset collected using the LilyGO T-Call ESP32 microcontroller, which was applied to monitor the parameters of Aloe vera plants using IoT agriculture sensors. The hardware schematics and source code used to collect and store data are available in the corresponding repository (https://github.com/firefly-cpp/t-call-esp32-data-collection, accessed on 3 June 2025). The data were sent to our data collection framework called **succulent** (https://github.com/firefly-cpp/succulent, accessed on 3 June 2025) using POST requests. The entire dataset used in this study can be accessed in the following repository: https://github.com/firefly-cpp/smart-agriculture-datasets (accessed on 3 June 2025).

A summary of the dataset is presented in Table 3. As is evident in the table, the dataset consists of time-series elements that include four features, as follows:

$$I = ('temp','hum','moist','light').$$

Let us emphasize that the attributes of all features are of the numerical type, and obtained from the corresponding IoT agriculture sensors. On the other hand, our proposed segmented interval time-series numerical association method builds upon and significantly extends NARM to handle the temporal dimension inherent in time-series data. This includes mechanisms for segmentation, interval detection, and incorporating temporal patterns—features not present in the original NARM framework. Thus, while TS-NARM can be seen as a conceptual extension of NARM, the underlying differences in data structure and mining objectives make a direct comparison technically inappropriate.

Table 3. Characteristics of the 'Data Collection with LilyGO T-Call in Smart Agriculture' dataset.

Dataset Name	September 2024
Number of Instances	145,607
Number of Features	4
Feature Type	Numerical
Missing Values	No
Start Data Point	9 September 2024 00:00:01
End Data Point	25 September 2024 23:59:56
Sampling Rate	Approximately every 5 s
Device	LilyGO T-Call ESP32
Plant Species	Aloe Vera

4.2. Hardware and Software

The whole experiment proceeded in two parts. In the first part, we used NiaARMTS (https://github.com/firefly-cpp/NiaARMTS, accessed on 3 June 2025) software for the mining of TARs, as described in the previous sections. In the second part, the newly proposed explainable methods (xNiaARMTS) were integrated into the NiaARMTS software, then run on randomly selected identified association rules. Table 4 presents the hardware and software environment as applied in our experimental work.

Table 4. Hardware and software configuration for experiments.

Component	Specification
Processor (CPU)	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz
RAM	8 Gb
Operating System	Fedora 42
Programming Language	Python 3.13.3
Libraries	NiaARMTS 0.2.0

Since NiaARMTS can deal with any of the algorithms that are included in the NiaPy framework [15], we decided to generate rules using the PSO algorithm, which was already confirmed as a very efficient algorithm in our previous experiments [10]. While we acknowledge the potential of other metaheuristics, such as Genetic Algorithms (GAs) [16], our goal in this paper was not to perform a comprehensive comparison of optimization techniques but to focus on demonstrating the explainability of the proposed time-series numerical association rule-mining approach. The control parameters of the PSO algorithm applied during the tests are presented in Table 5. Let us emphasize that we used parameter settings that had already proven effective in our previous studies to ensure a stable and consistent evaluation environment. Given the focus on explainability rather than exhaustive optimization, we did not conduct a full sensitivity analysis in this work. However, we agree that this would be a valuable direction for future research, especially when prioritizing performance fine-tuning.

Table 5. Control Parameters of Particle Swarm Optimization (PSO).

Parameter	Selected Value(s)
Swarm Size ($n_{\text{particles}}$)	40
Cognitive Coefficient (c_1)	2.0
Social Coefficient (c_2)	2.0
Total Function Evaluations	10,000

Indeed, all the features are of the numerical type. The whole experimental work was dedicated to testing both explainable methods, consequently conducted in two steps as follows. In the first step, the following three scenarios were defined:

- Scenario A;
- Scenario B;
- Scenario C.

In each of the mentioned scenarios, the TARs were selected randomly and explained by the proposed attribute criticality analysis metrics. In the second step, the rule stability was tested using the selected TAR and explained by the proposed explanation metrics.

Let us emphasize that a direct comparison between the proposed TS-NARM method and the traditional Apriori algorithm is not feasible or meaningful due to fundamental differences in the nature and scope of the problems they address; therefore, no such comparison is included in the experimental study.

4.3. Results of the Attribute Criticality Analysis

In the first experiment, the attribute criticality analysis was conducted using three randomly selected TARs (i.e., Scenarios A–C) mined by the NiaARMTS software. In the remainder of the paper, the explainable data analysis according to different scenarios is described in detail.

4.3.1. Scenario A

In the first scenario, TAR-1 was analyzed, which is expressed as follows:

$$[(21\text{ September }2024,\ 13{:}37{:}36),(24\text{ September }2024,\ 17{:}37{:}29)]:$$
$$'\text{hum}'[52.48,\ 62.91]\wedge'\text{light}'[624.46,\ 892.87]\implies'\text{temp}'[21.14,\ 23.23].$$

TAR-1 captured more than 3 days and 4 h and highlights how a conjunction of features, humidity, and lights influence the temperature. The attribute criticality analysis metrics are illustrated in Table 6, from which it can be seen that the 'hum' feature has the highest coverage (i.e., $cover(\text{TAR-1}) = 0.93$). This means that the attribute is presented in more than 90 % of the antecedent of rules arising during the time-series segment. The rule incorporates three of the maximum four features (i.e., $incl(\text{TAR-1}) = 0.75$), while the amplitude is higher than 55 % and, thus, covers more than half of the border values within the domain of the attribute's feasible values in the segment. The 'light' attribute, on the other hand, is the second-ranked attribute in the antecedent. This attribute is distinguished by a higher amplitude, but it emerged rarely in the time-series elements within the observed segment. The 'temp' attribute, arising in the consequent, exposed the lower coverage by evaluating its consequent criticality due to the longer time interval in which the observations were performed (i.e., more than three days). The temperature in such long time intervals is subject to huge temperature changes.

Table 6. Attribute criticality analysis of TAR-1.

Rule Part	Rank	Feature	Coverage	Inclusion	Amplitude		Score
					Metric	Range	
Antecedent	1	'hum'	0.93	0.75	0.55	[40.2, 63.4]	0.8018
	2	'light'	0.07	0.75	0.88	[0.0, 2326.7]	0.4380
Consequent	1	'temp'	0.09	n/a	0.48	[20.3, 24.3]	0.3243

The results of the attribute criticality analysis are illustrated graphically in Figure 1, where the attributes of the antecedent and consequent rule parts are presented in a stacked bar plot. In the diagram, each of the observed metrics is drawn as a bar plot in different colors, while the corresponding areas reflect their magnitudes.

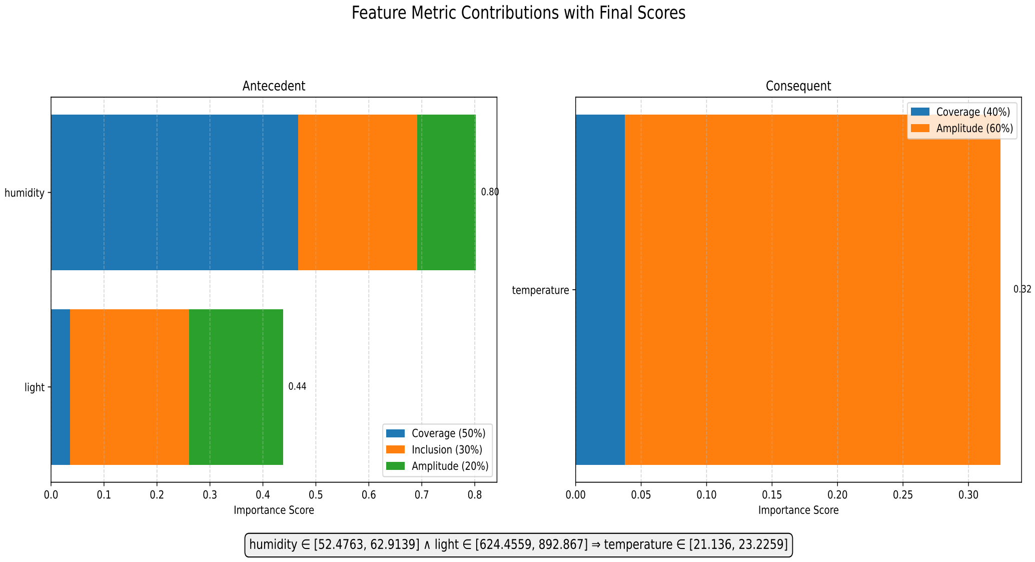

Figure 1. Visualization of attribute criticality analysis metrics for TAR-1.

In summary, the 'hum' attribute had the biggest impact on rule criticality due to a greater area being dedicated to the blue-colored bar.

4.3.2. Scenario B

The purpose of Scenario B was to analyze TAR-2, which is introduced as follows:

$$[(\text{14 September 2024},\ \text{21:43:07}), (\text{15 September 2024},\ \text{12:07:09})]: \\ \text{'light'}[110.38,\ 213.33] \wedge \text{'temp'}[22.71,\ 23.71] \implies \text{'hum'}[43.72,\ 53.67].$$

Indeed, the main characteristic of the rule is that it captures relatively short time intervals because the duration of the corresponding segment is less than one day (exactly 14 h, 24 min, and 2 s). Thus, it was expected that the observed criticality metrics could be more reliable, especially for the moisture feature.

The analytical results of the attribute criticality analysis are presented in Table 7, from which it is evident that our hypothesis about attribute reliability holds. Humidity remains the most critical attribute of TAR-2, where the most crucial metric coverage approaches the maximum value. The value of this metric was also increased substantially by the temperature attribute. The 'light' attribute remains less critical, especially because the observed interval captures time from the beginning of the night until noon the next day. Consequently, a suitable sensor measures light in almost all the measurement ranges (e.g., from 0 to 10,000 LUX) that are hard to cover with the appropriate time interval by mining with NiaARMTS.

Table 7. Attribute criticality analysis of TAR-2.

Rule Part	Rank	Feature	Coverage	Inclusion	Amplitude		Score
					Metric	Range	
Antecedent	1	'light'	0.0216	0.75	0.87	[0.0, 918.33]	0.4095
	2	'temp'	0.0002	0.75	0.55	[20.9, 23.1]	0.3342
Consequent	1	'hum'	0.9305	n/a	0.65	[28.0, 56.4]	0.4176

The graphical results of the attribute criticality analysis are illustrated in Figure 2, where the criticality of the attributes in TAR-2 is depicted by stacked bar plots. The larger the area of the bar, the more critical the observed attribute. In this sense, the most critical attribute in the antecedent is 'temp', while the 'hum' attribute remains the most critical in the consequent.

4.3.3. Scenario C

The attribute criticality analysis of TAR-3 was the most complex due to the appearance of all features in this rule and the longer duration of the time-series segment (precisely 4 days, 10 h 28, and min 54 s). The rule was mined by NiaARMTS, highlighting a relationship between the antecedent 'light' attribute and the remainder of the attributes emerging in the consequent. It is expressed as follows:

$$[(\text{16 September 2024},\ \text{21:58:18}), (\text{21 September 2024},\ \text{07:27:12})]: \\ \text{'light'}[227.73, 716.13] \implies \\ \text{'temp'}[24.19,\ 36.93] \wedge \text{'hum'}[36.93, 41.09] \wedge \text{'moist'}[1666.32,\ 1974.21].$$

The analytical results of the attribute criticality analysis of TAR-3 are depicted in Table 8, from which it is evident that the 'hum' attribute in the consequent achieved a total score of 0.9556; consequently, it is presented in all the elements of the observed time-series segment. This behavior indicates that the weather might be stable in this segment.

The ’moist’ attribute also achieved a good value regarding the coverage metric, but the results were not so brilliant regarding the inclusion metric. In contrast, the ’term’ attribute is better regarding inclusion but worse regarding the coverage metric.

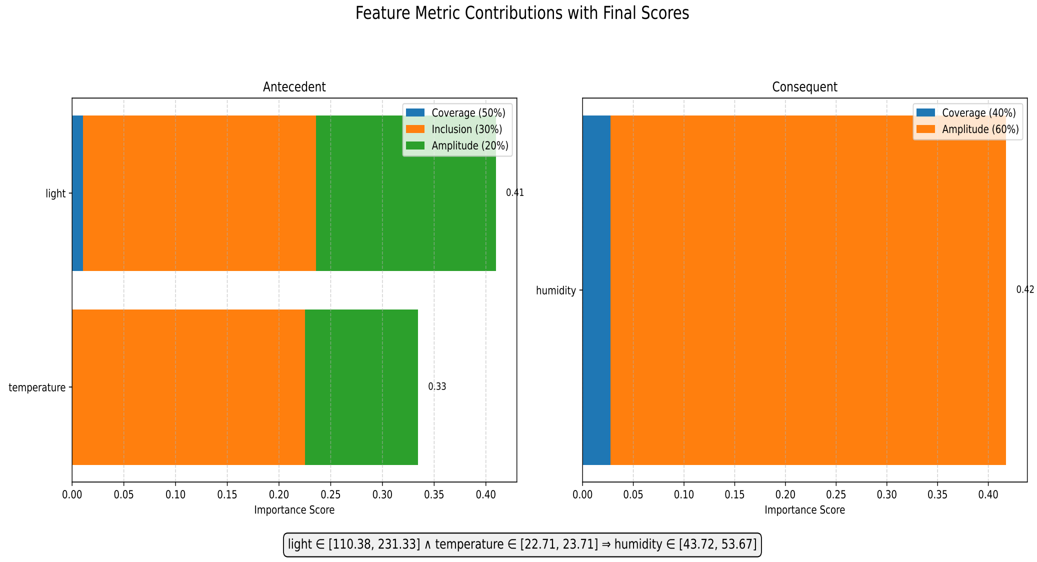

Figure 2. Visualization of attribute criticality analysis metrics for TAR-2.

Table 8. Attribute criticality analysis of TAR-3.

Rule Part	Rank	Feature	Coverage	Inclusion	Amplitude		Score
					Metric	Range	
Antecedent	1	’light’	0.24	1.00	0.69	[0.0, 1552.5]	0.5580
Consequent	1	’hum’	1.00	n/a	0.93	[6.3, 62.8]	0.9556
	2	’temp’	0.18	n/a	0.72	[19.3, 25.7]	0.5032
	3	’moist’	0.49	n/a	0.33	[1550.0, 2010.0]	0.3926

’Light’ is the only attribute in the antecedent and showed good results regarding the inclusion and amplitude metrics. In contrast, the coverage metric is crucially dependent on the longer time-series segments, where days and nights (and, consequently, the intensity of the light) are exchanged four times.

The graphical results for TAR-3 are also interesting, as presented in Figure 3, where the most criticality in the total scores for attributes is in the inclusion metric in the antecedent and the amplitude attribute in the consequent.

4.4. Results of the Rule Stability Analysis

TAR-4 was randomly selected for the rule stability analysis. The rule is described as follows:

$$[(18\text{ September }2024,\ 13{:}18),(20\text{ September }2024,\ 13{:}18)]:$$
$$\text{'temp'}[22.34,\ 24.83] \implies \text{'light'}[0.00,\ 871.11].$$

As indicated by the definition of TAR-4, the association rule captures a time-series segment with a duration of 2 days (i.e., 48 h). To make the experiment as fair as possible, the offset from the original time interval was also set to this duration, i.e., $\delta = 48$ h.

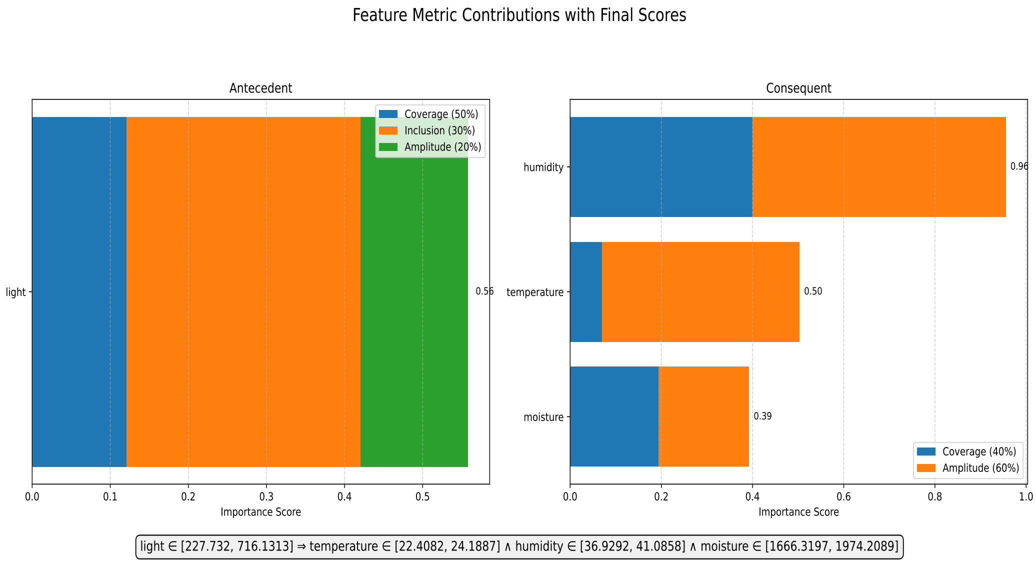

Figure 3. Visualization of attribute criticality analysis metrics for TAR-3.

The results of the stability analysis are illustrated in Table 9, which represents the three evaluation windows (I^-, I, and I^+) needed for a stability calculation, together with their corresponding starting and ending timestamps and support and confidence metrics; finally, the quality metric (∇) is expressed as the average of two calculated metrics. Based on the Euclidean distances between the mentioned evaluation windows, the stability function was calculated according to Equation (18) as $Stability(\text{TAR-4}) = 0.0955$.

In summary, the performed stability analysis showed that the calculated stability function is within the confidence interval of 90 %. Therefore, we can accept the hypothesis that the observed time-series association rule (TAR-4) is stable because a stability value below 0.1 (e.g., 0.0955 for TAR-4) is considered excellent and indicates strong temporal consistency.

The graphical results of the stability analysis are illustrated in Figure 4, where the diagram is divided into two parts: (1) the stacked bar plot, and (2) the group-ordered bar plot. The first bar plot represents the widths of the particular evaluation windows, while the second is focused on the particular metrics on which the stability calculation was founded, i.e., support, confidence, and quality. As is evident from the first bar plot, the evaluation windows are of the same duration. The second bar plot is more expressive and shows only the minor deviations between the calculated metrics.

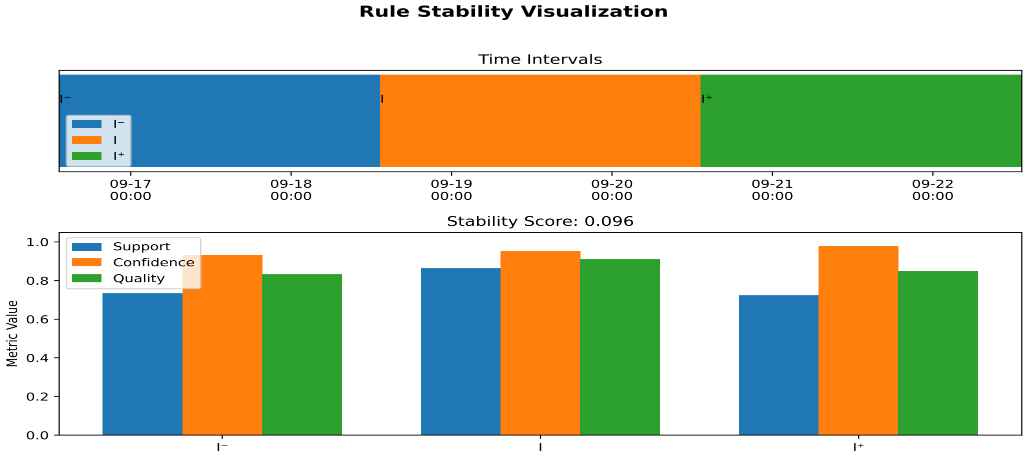

Figure 4. Visualization of rule stability analysis for TAR 4.

Table 9. Stability analysis for TAR 4 ($\nabla = 0,0955$, $\delta = 48$ h).

Window	Start	End	Support	Confidence	Quality
I^-	16 September 2024 13:18	18 September 2024 13:18	0.7324	0.9334	0.8329
I	18 September 2024 13:18	20 September 2024 13:18	0.8643	0.9532	0.9088
I^+	20 September 2024 13:18	22 September 2024 13:18	0.7229	0.9787	0.8508

5. Conclusions

In this paper, we have demonstrated the effectiveness of integrating explainable methods into numerical association rule mining for time-series data. Two explainable methods (xNiaARMTS) have been proposed for explainable data analysis for time-series numerical association rules from two points of view. The first explainable method focuses on attribute criticality analysis, while the second is based on a statistical stationary analysis of the time-series data.

Although the proposed explainable methods (xNiaARMTS) can be used generally, they were employed specifically to agricultural time-series data collected using several IoT sensors during long enough periods and processed by NiaARMTS. Only parts of the time-series data (the so-called time-series segment) were entered into the explainable data analysis due to the easier identification of the trends, seasonality, and cycle behavior of the time series. The proposed methods (xNiaARMTS) show promising results in smart-agriculture applications, highlighting the importance of advancing not only algorithm development but also post hoc explainable NARM methods.

This paper provides a bridge between the theoretical aspects and practical demonstrations of the study in the following ways:

- Real-world data: Our experimental evaluations were conducted on real-world time-series data collected from a smart-agriculture environment, which ensures method’s applicability.
- Open-source implementation: The complete implementation of our proposed method is publicly available in a GitHub repository that is accompanied by detailed documentation and numerous usage examples to facilitate use in the real world.
- Visualization tools: We developed and presented several visualization techniques to effectively interpret the discovered numerical association rules, enhancing the practical interpretability of the results.
- Application-driven: This study is grounded in a clear, real-world problem, i.e., in smart agriculture.

In future lines of research, we plan to experiment with different weights/offsets, which play an important role in both proposed methods, to enhance their performance on different prediction problems. Beyond the smart-agriculture domain, the proposed methods (xNiaARMTS) have potential applications involving time-series sensor data, such as predictive maintenance in industrial systems, health monitoring through the use of wearable devices, energy consumption analysis in smart grids, sports training data analysis, and environmental monitoring. These domains similarly benefit from interpretable rule mining to support decision-making processes.

Author Contributions: Conceptualization, I.F.J. and S.S.-S.; methodology, I.F.J., S.S.-S., D.N., I.F., V.P. and M.G.; software, I.F.J.; validation, D.N., I.F., E.A.-C. and M.G.; investigation, I.F.J. and S.S.-S.; data curation, I.F.J.; writing—original draft preparation, I.F.J. and S.S.-S.; writing—review and editing, D.N., I.F., E.A.-C. and M.G.; supervision, S.S.-S., I.F. and M.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially supported by project PID2023-150663NB-C21 of the Spanish Ministry of Science and Innovation (MICINN).

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Varol Altay, E.; Alatas, B. Performance analysis of multi-objective artificial intelligence optimization algorithms in numerical association rule mining. *J. Ambient Intell. Humaniz. Comput.* **2020**, *11*, 3449–3469. [CrossRef]
2. Altay, E.V.; Alatas, B. Intelligent optimization algorithms for the problem of mining numerical association rules. *Phys. A Stat. Mech. Its Appl.* **2020**, *540*, 123142. [CrossRef]
3. Badhon, B.; Kabir, M.M.J.; Xu, S.; Kabir, M. A survey on association rule mining based on evolutionary algorithms. *Int. J. Comput. Appl.* **2021**, *43*, 775–785. [CrossRef]
4. Alatas, B.; Akin, E.; Karci, A. MODENAR: Multi-objective differential evolution algorithm for mining numeric association rules. *Appl. Soft Comput.* **2008**, *8*, 646–656. [CrossRef]
5. Telikani, A.; Gandomi, A.H.; Shahbahrami, A. A survey of evolutionary computation for association rule mining. *Inf. Sci.* **2020**, *524*, 318–352. [CrossRef]
6. Kaushik, M.; Sharma, R.; Peious, S.A.; Shahin, M.; Yahia, S.B.; Draheim, D. A systematic assessment of numerical association rule mining methods. *SN Comput. Sci.* **2021**, *2*, 348. [CrossRef]
7. Minaei-Bidgoli, B.; Barmaki, R.; Nasiri, M. Mining numerical association rules via multi-objective genetic algorithms. *Inf. Sci.* **2013**, *233*, 15–24. [CrossRef]
8. Beiranvand, V.; Mobasher-Kashani, M.; Bakar, A.A. Multi-objective PSO algorithm for mining numerical association rules without a priori discretization. *Expert Syst. Appl.* **2014**, *41*, 4259–4273. [CrossRef]
9. Altay, E.V.; Alatas, B. Differential evolution and sine cosine algorithm based novel hybrid multi-objective approaches for numerical association rule mining. *Inf. Sci.* **2021**, *554*, 198–221. [CrossRef]
10. Fister, I., Jr.; Fister, D.; Fister, I.; Podgorelec, V.; Salcedo-Sanz, S. Time series numerical association rule mining variants in smart agriculture. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 16853–16866. [CrossRef]
11. Arrieta, A.B.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; García, S.; Gil-López, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115. [CrossRef]
12. Holzinger, A.; Goebel, R.; Fong, R.; Moon, T.; Müller, K.R.; Samek, W. xxAI—Beyond Explainable Artificial Intelligence. In *xxAI—Beyond Explainable AI: International Workshop, Proceedings of the Conjunction with ICML 2020, Vienna, Austria, 18 July 2020*; Revised and Extended Papers; Holzinger, A., Goebel, R., Fong, R., Moon, T., Müller, K.R., Samek, W., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 3–10. [CrossRef]
13. Fister Jr, I.; Fister, I.; Podgorelec, V.; Salcedo-Sanz, S.; Holzinger, A. NarmViz: A novel method for visualization of time series numerical association rules for smart agriculture. *Expert Syst.* **2024**, *41*, e13503. [CrossRef]
14. Kamath, U.; Liu, J. Introduction to Interpretability and Explainability. In *Explainable Artificial Intelligence: An Introduction to Interpretable Machine Learning*; Springer International Publishing: Cham, Switzerland, 2021; pp. 1–26. [CrossRef]
15. Vrbančič, G.; Brezočnik, L.; Mlakar, U.; Fister, D.; Fister, I. NiaPy: Python microframework for building nature-inspired algorithms. *J. Open Source Softw.* **2018**, *3*, 613. [CrossRef]
16. Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 1994.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.